Journal of Cyber Security
DOI: 10.32604/jcs.2022.027693
Article

Tech Science Press

A Certificateless Homomorphic Encryption Scheme for Protecting Transaction Data Privacy of Post-Quantum Blockchain

Meng-Wei Zhang[1], Xiu-Bo Chen[1], Haseeb Ahmad[2], Gang Xu[3,4,*] and Yi-Xian Yang[1]

[1]Information Security Center, State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing, 100876, China
[2]Department of Computer Science National Textile University, Faisalabad, 37610, Pakistan
[3]School of Information Science and Technology, North China University of Technology, Beijing, 100144, China
[4]Beijing Key Laboratory of Security and Privacy in Intelligent Transportation, Beijing Jiaotong University, Beijing, 100044, China
*Corresponding Author: Gang Xu. Email: gangxu_bupt@163.com
Received: 05 March 2022; Accepted: 08 April 2022

Abstract: Blockchain has a profound impact on all areas of society by virtue of its immutability, decentralization and other characteristics. However, blockchain faces the problem of data privacy leakage during the application process, and the rapid development of quantum computing also brings the threat of quantum attack to blockchain. In this paper, we propose a lattice-based certificateless fully homomorphic encryption (LCFHE) algorithm based on approximate eigenvector firstly. And we use the lattice-based delegate algorithm and preimage sampling algorithm to extract part of the private key based on certificateless scheme, which is composed of the private key together with the secret value selected by the user, thus effectively avoiding the problems of certificate management and key escrow. Secondly, we propose a post-quantum blockchain transaction privacy protection scheme based on LCFHE algorithm, which uses the ciphertext calculation characteristic of homomorphic encryption to encrypt the account balance and transaction amount, effectively protecting the transaction privacy of users and having the ability to resist quantum attacks. Finally, we analyze the correctness and security of LCFHE algorithm, and the security of the algorithm reduces to the hardness of learning with errors (LWE) hypothesis.

Keywords: Blockchain; homomorphic encryption; lattice; privacy protection

1 Introduction

In 2008, Nakamoto [1] introduced bitcoin, also known as cryptocurrency or electronic cash. Its transaction records are in blockchain. As a public distributed ledger, blockchain has the characteristics of decentralization, non-tampering and traceability, which has attracted more and more attention and research from all walks of life.

With the rapid development of blockchain technology, the problem of sensitive data privacy disclosure is becoming more and more prominent. First of all, in blockchain transactions, the transaction record of the whole network is open to all nodes in the blockchain, and the transaction amount of users on the blockchain ledger is stored in plaintext, which causes the problem of privacy disclosure.

There have been some researches on blockchain data privacy protection [2–7]. Among them, the homomorphic encryption technology, as a special encryption scheme, can realize that the decrypted result of the encrypted data obtained after calculation is the same as the result obtained by performing the calculation on plaintext, which can significantly improve the security of user data privacy. However, the existing technology still uses classic cryptographic algorithms, and the traditional solutions designed based on the difficult assumptions of number theory are no longer safe in the quantum environment. The privacy and security issues of sensitive data leakage and quantum attacks on the blockchain urgently need specific and effective privacy protection schemes to protect. As one of the most important post-quantum cryptographic algorithms, lattice cryptography has gradually become a research hotspot in cryptography.

At present, there is no quantum algorithm for solving difficult problems on the lattice, and the lattice cipher is simple to calculate and has high security. However, lattice cryptographic algorithms are mostly designed based on the traditional PKI cryptosystem or based on the identity-based cryptosystem, and there will be problems with certificate management and key escrow. In 2003, Al-Riyami et al. [8] proposed the idea of certificateless public key cryptosystem. In the certificateless system, the user's private key is composed of a part of the private key generated by KGC and a random number selected by the user. KGC cannot know the complete private key information, so the above problems can be avoided, which gives us a lot of inspiration.

In this paper, we propose a post-quantum blockchain privacy protection scheme based on lattice homomorphic encryption. The main contributions of this paper are as follows:

(1) We propose a lattice-based certificateless fully homomorphic encryption (LCFHE) algorithm. The security of the algorithm is based on LWE difficult problem and can resist quantum computing attacks. In addition, the algorithm can solve the key escrow and certificate management problems effectively based on the certificateless system.
(2) Based on the LCFHE algorithm, we propose a post-quantum blockchain transaction data protection scheme, which effectively protects the user's transaction privacy by homomorphically encrypting the available balance and transaction amount during the transaction process.
(3) We analyze the algorithm and prove that the algorithm satisfies the correctness and has the security of chosen-plaintext attack.

2 Related Work

In view of security issues such as exposure of sensitive transaction data faced by blockchain, relevant researchers have done a lot of studies [9–16]. In 2013, Maxwell proposed the CoinJoin mechanism [17]. The core idea of the mechanism is to merge multiple transactions into one transaction, so as to hide the corresponding relationship between the input and output parties. However, the credibility of participating nodes cannot be guaranteed during the process, so attackers can realize DoS attacks at low cost. In 2014, Ruffing et al. [18] proposed a completely decentralized CoinShuffle protocol for bitcoin. But it requires all users to be online at the same time, which does not solve the DoS attack. In 2014, Sasson et al. [19] proposed Zerocash protocol which hides the amount of each transaction in addition to providing anonymous function. However, the process of generating a proof is

very slow. Homomorphic encryption, as a ciphertext processing technology, can avoid the disclosure of privacy in the process of data verification or operation, and has been widely concerned by researchers.

Homomorphic encryption can achieve the same decryption result of ciphertext calculation as that of plaintext calculation, which can significantly improve the security of user data privacy. In 2009, Gentry [20] proposed the first homomorphic encryption algorithm, which realized homomorphic encryption using Bootstrapping. In 2011, Brakerski et al. implemented holomorphic encryption [21] using LWE hypothesis for the first time and [22] under RLWE hypothesis, which is called BGV algorithm. It is the first hierarchical holomorphic encryption algorithm without bootstrapping technology. In 2013, Gentry et al. [23] realized fully homomorphic encryption for the first time by using the method of approximate feature vector, which is the most classical Gentry-Sahai-Waters (GSW) scheme at present. Although the biggest advantage of this scheme is that it no longer relies on public key calculation in homomorphic operation. In fact, the size of its public and private keys is not significantly different from many other algorithms. Homomorphic encryption has been studied, but the current homomorphic encryption schemes generally have the problems of large key space, high operation difficulty and low efficiency, so it is urgent to put forward a more high-performance homomorphic encryption scheme.

In order to solve the problem of exposing the privacy of transaction data, some researchers study the block chain data privacy protection scheme based on homomorphic encryption. In 2015, Cheon et al. [24] from South Korea proposed a ciphertext processing framework based on the homomorphic encryption scheme proposed by Brakerski et al. [21] to realize the homomorphic calculation of ciphertext data. However, the homomorphic encryption algorithm used in the scheme has problems such as large public key size and large computation. In 2019, Yu et al. [25] in China proposed a blockchain privacy protection method based on Paillier homomorphic encryption, but it has the problem of limited calculation. In 2020, Wang et al. [26] in China used Paillier homomorphic encryption combined with zero-knowledge proof to realize transaction protection in view of the security problems of traditional Internet of vehicles devices, but the scheme had problems of large key space and low computational efficiency. At present, most of the existing blockchain transaction privacy protection schemes use Paillier homomorphic encryption algorithm, but it cannot resist quantum attacks. How to make use of lattice cryptography theory [27–30] in the blockchain system to study the block chain sensitive data security processing scheme based on lattice homomorphic encryption algorithm is an urgent problem to be solved.

In view of the problem of exposing sensitive transaction data of blockchain, this project conducts in-depth research on high-performance homomorphic encryption algorithm based on lattice cryptography, mode exchange, key exchange and other technologies on blockchain. In addition, a lattice-fully homomorphic crypto blockchain scheme that can implicitly process sensitive transaction data is constructed to realize the privacy protection of blockchain sensitive data security processing.

3 Preliminaries

3.1 Lattice

Definition 1: Given n linearly independent vectors $b_1, b_2, \ldots, b_n \in \mathbb{R}^m$, the lattice $\mathcal{L}$ is generated as the set

$$\mathcal{L}(b_1, b_2, \ldots, b_n) = \left\{ \sum x_i \cdot b_i | x_i \in \mathbb{Z} \right\} \tag{1}$$

and $b_1, b_2, \ldots, b_n$ are called a set of basis of lattice $\mathcal{L}$, n is the rank of the lattice $\mathcal{L}$, m is the dimension of lattice $\mathcal{L}$, $m \geq n$. If $n = m$, the lattice $\mathcal{L}$ is called full-rank lattice. If B is defined as an $m \times n$ dimensional

matrix composed of $b_1, b_2, \ldots, b_n$, the lattice $\mathcal{L}$ generated by B is shown as follows:

$$\mathcal{L}(b_1, b_2, \ldots, b_n) = \left\{ \sum x_i \cdot b_i | x_i \in \mathbb{Z} \right\} \tag{2}$$

Definition 2: Given modulus q, dimension m, n, matrix $A \in \mathbb{Z}_q^{m \times n}$, vector $A \in \mathbb{Z}_q^{m \times n}$, and three integer lattices are defined as follows:

$$\Lambda_q^{\perp}(A) = \{v \in \mathbb{Z}^m \ s.t. \ A \cdot v = 0 \ (mod \ q)\} \tag{3}$$

$$\Lambda_q^{u}(A) = \{v \in \mathbb{Z}^m \ s.t. \ A \cdot v = u \ (mod \ q)\} \tag{4}$$

$$\Lambda_q(A) = \left\{v \in \mathbb{Z}^m \ s.t. \exists s \in \mathbb{Z}_q^n, \ A^T \cdot s = v \ (mod \ q)\right\} \tag{5}$$

Definition 3(LWE): Given a parameter n $\geq$ 1, χ is the Gaussian noise distribution over $\mathbb{Z}_q^m$, with modulus q $\geq$ 2. The probability distribution $A_{s,\chi}$ is obtained by selecting the matrix $A \in \mathbb{Z}_q^{n \times m}$ and the vector s $\in \mathbb{Z}_q^n$ randomly and uniformly, extracting the noise vector $e \in \mathbb{Z}_q^m$ from the gaussian noise distribution χ, and producing $\left(A, A^T s + e\right)$. There is the following definition:

(1) ***SLWE:*** Given the polynomial samples $\left(A, A^T s + e\right)$ of the distribution $A_{s,\chi}$, output s $\in \mathbb{Z}_q^n$ with a non-negligible probability.
(2) ***DLWE:*** Determine whether a sample $\left(A, A^T s + e\right)$ is obtained by the above algorithm or is randomly selected from the uniform distribution on $\mathbb{Z}_q^{n \times m} \times \mathbb{Z}_q^m$.

The SLWE problem and the DLWE problem can be mutually regulated.

3.2 Lattice Sampling Algorithms

Proposition 1: Given prime $q \geq 3$, $m \geq 5nlogq$, the polynomial-time algorithm $TrapGen(q, n, m)$ can output (A, S) with $A \in \mathbb{Z}_q^{m \times n}$, $S \in \mathbb{Z}^{m \times n}$. S is a basis of the lattice $\Lambda_q^{\perp}(A)$ and $\|S\| \leq O\left(\sqrt{nlogq}\right)$.

Proposition 2: Given a matrix $A \in \mathbb{Z}_q^{m \times n}$, a basis T_A of $\Lambda_q^{\perp}(A)$, a vector $u \in \mathbb{Z}_q^n$, the polynomial-time algorithm $SamplePre(A, T_A, u, \sigma)$ can output a matrix $x \in \mathbb{Z}^m$ that satisfies $A \cdot x = u \ (mod \ q)$. The distribution of x is closing to $D_{\Lambda_q^u(A),\sigma}$.

3.3 Homomorphic Encryption

A homomorphic (public key) encryption algorithm HE = (He.keygen, He.enc, He.dec, He.eval) is a polynomial time algorithm (n is the security parameter) as follows:

(1) ***Key generation algorithm:*** $(pk, evk, sk) \leftarrow$ HE.Keygen (1^n), output public key pk, homomorphic calculation key evk, private key sk;
(2) ***Encryption algorithm:*** $c \leftarrow \text{HE.Enc}_{pk}(m)$, using public key pk, encrypts single bit information $m \in \{0, 1\}$ to generate ciphertext c;
(3) ***Decryption algorithm:*** $m \leftarrow \text{HE.Dec}_{sk}(c)$ uses the private key sk to decrypt the ciphertext c and recover the information $m \in \{0, 1\}$;
(4) ***Homomorphic calculation algorithm:*** $c_f \leftarrow \text{HE.Eval}_{evk}(f, c_1, \ldots, c_l)$, using homomorphism to calculate the key evk, applying the function $f : \{0, 1\}^l \rightarrow \{0, 1\}$ to $c_1, \ldots, c_l$, output the ciphertext c_f. He.eval is divided into homomorphic addition $c_{\text{add}} \leftarrow \text{HE.Add}_{evk}(c_1, c_2)$ and homomorphic multiplication $c_{\text{mult}} \leftarrow \text{HE.Mult}_{evk}(c_1, c_2)$.

4 Lattice-based Certificateless Fully Homomorphic Encryption (LCFHE)

We design a lattice-based certificateless homomorphic encryption algorithm, in which the private key consists of part of the KGC private key and the secret value randomly selected by the user. The specific steps of the algorithm are as follows:

(1) ***Setup*** $\left(1^{\lambda}, 1^{L}\right)$***:***
   a) Input the safety parameter λ and circuit depth L, and select the parameters $\mathrm{n} = \mathrm{n}(\lambda, L)$, $\mathrm{m} = O(nlogq)$, $\mathrm{q} = \mathrm{q}(n)$, $\sigma_1 = \sigma_1(\lambda, L)$, $\sigma_2 = \sigma_2(\lambda, L), \sigma_3 = \sigma_3(\lambda, L), \mathrm{N} = (2\mathrm{m}+1) \cdot \mathrm{logq}$.
   b) Execute the trapdoor generation algorithm $\mathrm{TranGen}(\mathrm{q}, \mathrm{n}, \mathrm{m})$, and output a matrix $\mathrm{A} \in \mathbb{Z}_q^{n*m}$ that obeys an approximate random distribution and a basis $T_A \in \mathbb{Z}_q^{m*m}$ of the lattice $\Lambda_q^{\perp}(\mathrm{A})$, and $\left\|\tilde{T}_A\right\| \leq O\left(\sqrt{nlogq}\right)$.
   c) Select secure Hash functions $\mathrm{H}_1 : \{0,1\}^* \rightarrow \mathbb{Z}_q^{m*m}$, $\mathrm{H}_2 : \mathbb{Z}_q^{m*m} \rightarrow \mathbb{Z}_q^n$, $\mathrm{H}_3 : \mathbb{Z}_q^{n*m} \rightarrow \mathbb{Z}_q^{(2m+1)\times N}$.
   d) Output public parameters $\mathrm{pp} = (\mathrm{A}, \mathrm{H}_1, \mathrm{H}_2)$, System master key $\mathrm{msk} = T_A$.

(2) ***Extract*** $(\boldsymbol{pp}, \boldsymbol{msk}, \boldsymbol{id})$***:***
   a) KGC generates part of the private key for the user and calculates $\mathrm{R} = \mathrm{H}_1(\mathrm{id})$, $Q_{id} = A \cdot R^{-1}$, executes the Lattice delegate algorithm $\mathrm{BasisDel}(\mathrm{A}, \mathrm{R}, T_A, \sigma_1)$, output matrix $T_{id} \in \mathbb{Z}_q^{m*m}$.
   b) Calculate $g = \mathrm{H}_2(\mathrm{R})$, and execute the sampling algorithm $\mathrm{SamplePre}(Q_{id}, T_{id}, g, \sigma_2)$, output a vector $\mathrm{y} \in \mathbb{Z}_q^m$ m with a statistical distribution close to $D_{\Lambda_q^g(A),\sigma_2}$, and $Q_{id} \cdot y = \mathrm{g}$ mod q, and part of the private key is y.

(3) ***SecretGen*** $(\boldsymbol{pp}, \boldsymbol{id})$***:*** The user randomly selects the secret vector $\mathrm{h} \leftarrow D_{\mathbb{Z}^m,\sigma_3}$.

(4) ***KeyGen*** $(\boldsymbol{pp}, \boldsymbol{id}, \boldsymbol{d})$:
   a) Output public key $pk = Q_{id}$.
   b) Output private key $\mathrm{sk} = \mathrm{s} = \begin{pmatrix} -y \\ h \\ 1 \end{pmatrix} \in \mathbb{Z}_q^{2m+1}$.

(5) ***Enc*** $(\boldsymbol{pp}, \boldsymbol{pk}, \boldsymbol{\mu})$***:*** Input the plaintext μ, public key $pk = Q_{id}$, sample error vectors E_1, E_2, $E_3 \in D_{\mathbb{Z},\alpha q}^{m\times N}$, generate ciphertext matrix $C = \mu \cdot H_3(Q_{id}) + \begin{pmatrix} Q_{id}^T W + E_1^T \\ E_2^T \\ g^T W \end{pmatrix} \in \mathbb{Z}_q^{(2m+1)*N}$.

(6) ***Dec*** $(\boldsymbol{pp}, \boldsymbol{sk}, \boldsymbol{C})$: Let c_i be the ith row of C, and compute $x_i \leftarrow \langle s, c_i \rangle$, output $\mu' = \lfloor x_i / s_i \rceil$.

(7) ***Evaluate*** $(\boldsymbol{pp}, \boldsymbol{id}, \boldsymbol{C}_1, \boldsymbol{C}_2)$***:***
   a) $\mathrm{Add}(\mathrm{pp}, \mathrm{id}, \mathrm{C}_1, \mathrm{C}_2)$:

$$C_{Add} = \mathrm{C}_1 + \mathrm{C}_2 \in \mathbb{Z}_q^{(2m+1)*N} \tag{6}$$

   b) $\mathrm{Mult}(\mathrm{pp}, \mathrm{id}, \mathrm{C}_1, \mathrm{C}_2)$:

$$C_{Mult} = \mathrm{C}_1 \cdot H_3(Q_{id})^{-1} \cdot \mathrm{C}_2 \in \mathbb{Z}_q^{(2m+1)*N} \tag{7}$$

5 Security Proof and Analysis

5.1 Correctness

According to the above encryption algorithm, we have

$$s^T \cdot C = s^T \cdot \begin{pmatrix} Q_{id}^T W + E_1^T \\ E_2^T \\ g^T W \end{pmatrix} + \mu \cdot s^T \cdot H_3(Q_{id}) = \mu \cdot s^T \cdot H_3(Q_{id}) - y^T \cdot E_1^T + h^T \cdot E_2^T = \mu \cdot s^T \cdot H_3(Q_{id}) + p^T \tag{8}$$

let the error vector be $p^T = -y^T \cdot E_1^T + h^T \cdot E_2^T$, the bound of error is

$$\begin{aligned}\left\|p^T\right\| &= \left\|-y^T \cdot E_1^T + h^T \cdot E_2^T\right\| \le \|y\| \cdot \left\|E_1^T\right\| + \|h\| \cdot \left\|E_2^T\right\| \le \sigma_2\sqrt{m} \cdot \alpha q\sqrt{m} + \sigma_3\sqrt{m} \cdot \alpha q\sqrt{m} \\ &= ((\sigma_2 + \sigma_3)\,\mathrm{m} + 1) \cdot \alpha q \end{aligned} \tag{9}$$

where $\|y\| \le \sigma_2\sqrt{m}$, $\|h\| \le \sigma_3\sqrt{m}$. Therefore, the algorithm satisfies the correctness.

5.2 Homomorphic Operation

For the additive homomorphism, we have

$$\begin{aligned} s^T \cdot C_{Add} &= s^T \cdot (\mathrm{C}_1 + \mathrm{C}_2) = \mu_1 \cdot s^T \cdot M + p_1^T + \mu_2 \cdot s^T \cdot M + p_2^T \\ &= (\mu_1 + \mu_2) \cdot s^T \cdot M + p_1^T + p_2^T = (\mu_1 + \mu_2) \cdot \ \mathrm{v} + \mathrm{p}_{Add} \end{aligned} \tag{10}$$

Therefore, the additive homomorphism satisfies the correctness. In the same way, it can be proved that multiplication satisfies correctness.

$$\begin{aligned} s^T \cdot C_{Mult} &= s^T \cdot \mathrm{C}_1 \cdot H_3(Q_{id})^{-1} \cdot \mathrm{C}_2 = \left(p_1^T + \mu_1 \cdot s^T \cdot H_3\left(Q_{id}\right)\right) \cdot H_3(Q_{id})^{-1} \cdot \mathrm{C}_2 \\ &= p_1^T \cdot H_3(Q_{id})^{-1} \cdot \mathrm{C}_2 + \mu_1 \cdot p_2^T + \mu_1\mu_2 \cdot s^T \cdot H_3\left(Q_{id}\right) = \mu_1\mu_2 \cdot s^T \cdot H_3\left(Q_{id}\right) + \mathrm{p}_{Mult} \end{aligned} \tag{11}$$

5.3 Security Proof

Conclusion: Based on the LWE difficulty hypothesis, given the security parameters required by the above LCFHE scheme, if the LWE difficulty hypothesis is true, the LCFHE algorithm we proposed is chosen-plaintext attack security, and the proof is as follows.

Proof: Let A be a probabilistic polynomial time attacker, and the challenge ciphertext can be distinguished by probability under the chosen-plaintext attack security model. Specific steps are as follows:

(1) ***Setup:*** Input each security parameter, challenger generates public parameter pp and system master key msk.

(2) ***Queries1:*** The challenger executes KeyGen to obtain the public and private key pairs, which are returned to A.

(3) ***Challenge:*** After the query, A outputs two different plaintext μ_1 and μ_2. The challenger randomly selects a bit $k \in \{0, 1\}$ to generate the public key pk^* that has not been queried. After encryption, the challenge ciphertext is output $C_k = \begin{pmatrix} Q_k^T W + e_1^T \\ e_2^T \\ \mu_k^T W + e_3^T \end{pmatrix} + \mu_k \cdot M$.

(4) ***Queries2:*** Similar to Queries1, but the challenge public key cannot be queried.

(5) ***Output:*** A outputs the guess result $k' \in \{0, 1\}$, if A guesses correctly, output 1, if A guesses incorrectly, output 0.

We build a distributed discriminator D to distinguish $\left(Q_k, Q_k^T w_i + e_1^T\right)$ and $U\left(\mathbb{Z}_q^{n\times m} \times \mathbb{Z}_q^m\right)$, where $U\left(\mathbb{Z}_q^{n\times m} \times \mathbb{Z}_q^m\right)$ is randomly selected from uniform distribution on $\mathbb{Z}_q^{n\times m} \times \mathbb{Z}_q^m$. As we know, after getting the plaintext from A, D randomly selects $k \in \{0, 1\}$ and challenges the ciphertext to C_k. If D has the advantage of distinguishing ciphertext C_0 and C_1 with probability of ε, then it can also distinguish $\left(Q_k, Q_k^T w_i + e_1^T\right)$ and $U\left(\mathbb{Z}_q^{n\times m} \times \mathbb{Z}_q^m\right)$, then D can successfully solve LWE difficult problems, which is contrary to the fact. Therefore, LCHFE scheme has the security of indistinguishability under chosen-plaintext attack (IND-CPA).

5.4 Efficiency Analysis

As shown in Tab. 1, we compared the efficiency of scheme [31] and scheme [32] with the LCFHE algorithm proposed in this paper. In this scheme, the dimension m of the lattice is reduced to $2n \log q$. Analysis shows that when the m is reduced, the corresponding Gaussian parameter will become smaller, which can effectively improve the efficiency of the trapdoor generation algorithm and the sampling algorithm. In the comparison of public key size, it can be seen that the LCFHE scheme has a smaller public key size, requires less space to store the public key, and has higher encryption efficiency. In addition, in terms of the selection of public key cryptosystem, both schemes [31] and [32] adopt identity-based encryption system, which will lead to the third party KGC completely controlling the complete private key. This will cause key escrow problem, which will threaten the security of user's private key and encrypted information. The LCFHE algorithm is based on the certificateless public key encryption system, and the private key is composed of the third party KGC and the random value selected by the user, which effectively solves the key escrow problem and greatly improves the security of the scheme.

Table 1: Efficiency comparison

Schemes	m	Public key size	Cryptography
ABB10 [31]	$6n \log q$	$(mn^2 + 2mn) \log q$	IBE
Wang16 [32]	$5n \log q$	$(mn + n^2 + n) \log q$	IBE
Our scheme	$2n \log q$	$mn \log q$	CLE

6 Post-quantum Blockchain Transaction Data Protection Scheme Based on LCFHE

According to the lattice-based certificateless homomorphic encryption algorithm mentioned above, we design a post-quantum blockchain transaction data protection scheme. There are three roles in the scheme: key generation center, transaction node and validation node. Specific definitions are as follows:

KGC: Used to generate system public parameters, system master keys, and user private keys.

Transaction node: The node on the blockchain for transactions, which are divided into the transaction sender and transaction receiver.

Validation node: The network-wide verification node on the blockchain that is responsible for verifying the transaction information and maintaining the public ledger. We designed the public ledger to store the real balance of the network accounts as homomorphic encrypted ciphertext.

The cryptic transaction scheme design of blockchain based on LCFHE mainly includes the following three parts: hiding transaction amount, transaction verification and updating account available balance. Compared with blockchain transactions without privacy protection, the research focus and difficulty lies in how to verify the ciphertext transaction after the encrypted transaction amount and dynamically update the available balance of the encrypted account in real time. The following are designed for these three parts respectively.

In blockchain hidden transactions, the amount to be hidden mainly consists of two parts: one is the balance of each account stored in the blockchain public ledger, and the other is the transaction amount in a transaction. The available balance exists in two forms: one is ciphertext stored in the global ledger after being encrypted by the LCFHE algorithm, and the other is plaintext stored locally only for the

user to see personally. And there is Available balance = Remaining balance + Transaction amount. For each available balance, remaining balance and transaction amount stored in the blockchain public ledger, the LCFHE algorithm is used to encrypt and hide. As shown in Fig. 1, the scheme process is as follows:

(1) ***Initialization phase:*** According to the first three stages of LCFHE algorithm, KGC selects parameters, implements trap door generation algorithm, selects Hash function, and generates system public parameter pp and system master key msk.
(2) ***Partial key extraction phase:*** KGC generates part of private keys d_s and d_r for transaction sender S and transaction receiver R respectively.
(3) ***Secret value selection phase:*** The transaction node, including the transaction initiator and the transaction receiver, selects their own secret value h_s, and h_r is saved locally.
(4) ***Key generation phase:*** Perform the fourth step of the LCFHE algorithm to generate transaction nodes, including the transaction initiator and the transaction receiver, and obtain their respective public and private key pairs of homomorphic encryption (pk_s, sk_s), (pk_r, sk_r).
(5) ***Transaction data encryption stage:*** The available balance of the entire network transaction account is homomorphically encrypted using the user's homomorphic public key and stored in the public ledger. The available balances of the initiator and receiver of this transaction are A_s and A_r respectively after homomorphic encryption. The available balance is equal to the sum of the remaining balance and the transaction amount. In order to complete the secret transaction, the available amount of the transaction initiator is divided into two parts, the remaining balance and the transaction amount, and the homomorphic public key of the transaction sender is used as the remaining balance and the transaction The amount is homomorphically encrypted to obtain R_s, T_s. Next, use the homomorphic public key of the transaction receiver to homomorphically encrypt the transaction amount to obtain T_r, generate transaction data (R_s, T_s, T_r), and send the transaction data to the entire network for verification.
(6) ***Transaction verification stage:*** The verification nodes of the entire network receive the homomorphic encrypted transaction data, verify the correctness of the transaction according to the homomorphic nature of the algorithm, that is $A_s = R_s + T_s$, and use the least necessary zero-knowledge proof to verify the correctness of the transaction. The legality of T_s is verified, that is, the real amount is not less than zero, and should be less than the available amount (This is not the key research content of this paper, and will not be specified here). After the verification is successful, the verification node modifies the available balance ciphertext data of the corresponding transaction sender and receiver on the public ledger.
(7) ***Update the balance phase:*** The sender of the transaction updates its available balance. The transaction receiver uses the homomorphic private key to decrypt the transaction amount and update its available balance.
(8) ***Re-verification phase:*** After the transaction is over, the verification node requires the user to review the user's local available balance. If the homomorphic encrypted available balance is inconsistent with the real balance ciphertext of the public ledger, the transaction is determined to be invalid and the verification node cancels the transaction, the data of the public ledger is withdrawn to the state before the transaction. And this user cannot make the next transaction, which effectively prevents the user from changing the local available balance at will.

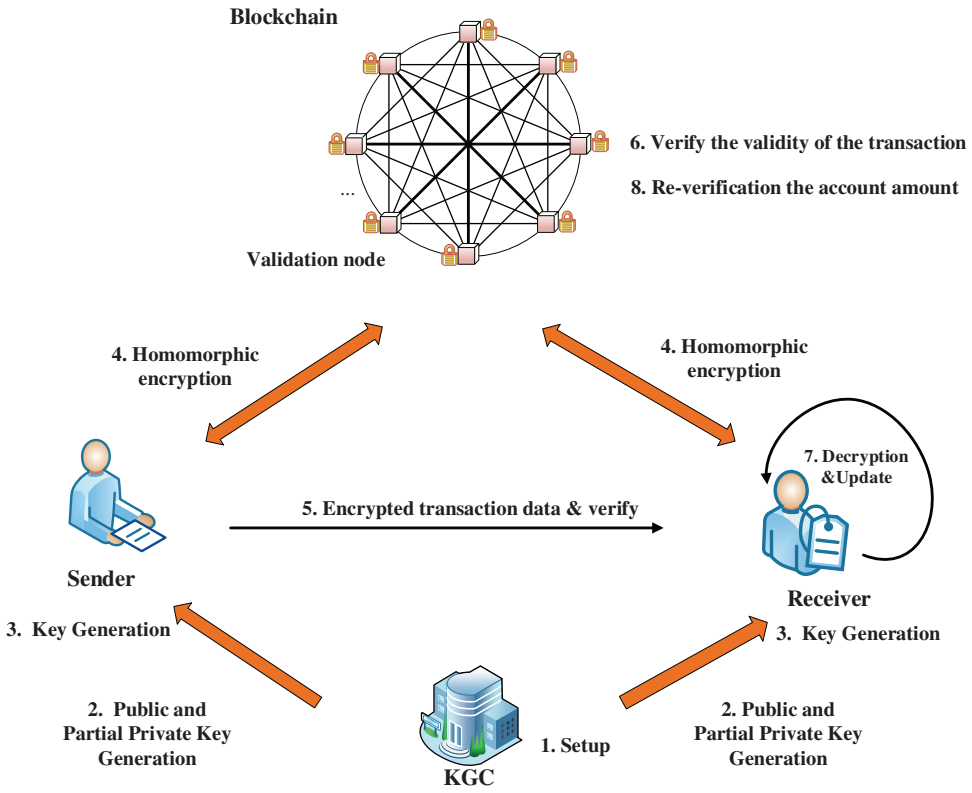

Figure 1: Post-quantum blockchain transaction data protection scheme based on LCFHE

7 Conclusion

In this paper, a new lattice-based certificateless homomorphic encryption LCFHE algorithm based on approximate eigenvectors is proposed, which can satisfy the correctness of addition homomorphism and multiplication homomorphism. Certificateless system uses the advantages and disadvantages of certificate-based system and identity-based system to avoid certificate management and key escrow. Among them, lattice delegation algorithm and original image sampling algorithm are used to extract part of the private key, and the secret value selected by the user is combined to form the private key, which protects the security of the user private key. In addition, the correctness and security of LCFHE algorithm are analyzed, and it is proved that the algorithm satisfies the correctness and the security of selecting plaintext attack. Based on LCFHE algorithm, this paper puts forward the quantum block chain transaction privacy protection scheme, calculated by use of homomorphic encryption cipher decryption and clear again the calculation results of the same features, the account of the available amount into the remaining amount and transaction, and the user's real balance stored encrypted in a public books, and the remaining amount in the trading and transaction amount for homomorphic encryption, It is invisible to other nodes, which effectively protects users' transaction privacy and can resist quantum attacks.

In the next step, we will further study the application of zero-knowledge proof and other technologies to blockchain privacy protection to further improve the security and practicability of the scheme. The content proposed in this paper can provide new ideas for post-quantum blockchain research and promote the development of post-quantum blockchain.

Funding Statement: This work is supported by NSFC (Grant Nos. 92046001, 61671087, 61962009, 61971021), the Fundamental Research Funds for Beijing Municipal Commission of Education, the Scientific Research Launch Funds of North China University of Technology, and Beijing Urban Governance Research Base of North China University of Technology.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," Decentralized Business Review, 2008. [Online]. Available: https://bitcoin.org/bitcoin.pdf.
[2] R. Mercer, "Privacy on the blockchain: Unique ring signatures," arXiv preprint arXiv, 2016. [Online]. Available: https://arxiv.org/pdf/1612.01188.pdf.
[3] C. Y. Li, Y. Tian, X. B. Chen and J. Li, "An efficient anti-quantum lattice-based blind signature for blockchain-enabled systems," *Information Sciences*, vol. 546, pp. 253–264, 2020.
[4] P. Mundhe, V. K. Yadav and A. Singh, "Ring signature-Based conditional privacy-Preserving authentication in VANETs," *Wireless Personal Communications*, vol. 114, no. 1, pp. 853–881, 2020.
[5] G. G. Dagher, J. Mohler and M. Milojkovic, "Ancile: Privacy-preserving framework for access control and interoperability of electronic health records using blockchain technology," *Sustainable Cities and Society*, vol. 39, pp. 283–297, 2018.
[6] A. Sahai, H. Seyalioglu and B. Waters, "Dynamic credentials and ciphertext delegation for attribute-based encryption," in *Annual Cryptology Conf.*, Berlin, Heidelberg, Springer, pp. 199–217, 2012.
[7] J. Xu, K. Xue and S. H. Li, "Healthchain: A blockchain-based privacy preserving scheme for large-scale health data," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 8770–8781, 2019.
[8] S. S. Al-Riyami and K. G. Paterson, "Certificateless public key cryptography," in *Int. Conf. on the Theory and Application of Cryptology and Information Security*, Berlin, Heidelberg, Springer, pp. 452–473, 2003.
[9] J. A. Kassem, S. Sayeed and H. Marco-Gisbert, "DNS-IdM: A blockchain identity management system to secure personal data sharing in a network," *Applied Sciences*, vol. 9, no. 15, pp. 2953, 2019.
[10] A. Sahai and B. Waters, "Fuzzy identity-based encryption," in *Annual Int. Conf. on the Theory and Applications of Cryptographic Techniques*, Berlin, Heidelberg, Springer, pp. 457–473, 2005.
[11] S. Gusmeroli, S. Piccione and D. Rotondi, "A Capability-based security approach to manage access control in the internet of things," *Mathematical and Computer Modelling*, vol. 58, no. 5–6, pp. 1189–120, 2013.
[12] H. Es-Samaali, A. Outchakoucht and J. P. Leroy, "A Blockchain-based access control for big data," *International Journal of Computer Networks and Communications Security*, vol. 5, no. 7, pp. 137, 2017.
[13] C. Y. Li, M. X. Dong and J. Li, "Healthchain: Secure EMRs management and trading in distributed healthcare service system," *IEEE Internet of Things Journal*, vol. 8, no. 9, pp. 7192–7202, 2020.
[14] T. Feng, H. Pei, R. Ma, Y. Tian and X. Feng, "Blockchain data privacy access control based on searchable attribute encryption," *Computers, Materials & Continua*, vol. 66, no. 1, pp. 871–890, 2021.
[15] X. Peng, J. Zhang, S. Zhang, W. Wan, H. Chen *et al.,* "A secure signcryption scheme for electronic health records sharing in blockchain," *Computer Systems Science and Engineering*, vol. 37, no. 2, pp. 265–281, 2021.
[16] S. R. Khonde and V. Ulagamuthalvi, "Blockchain: Secured solution for signature transfer in distributed intrusion detection system," *Computer Systems Science and Engineering*, vol. 40, no. 1, pp. 37–51, 2022.
[17] G. Maxwell, "CoinJoin: Bitcoin privacy for the real world," *Post on Bitcoin Forum*, vol. 3, pp. 110, 2013.
[18] T. Ruffing, P. Moreno-Sanchez and A. Kate, "Coinshuffle: Practical decentralized coin mixing for bitcoin," in *European Symp. on Research in Computer Security*, Cham, Springer, pp. 345–364, 2014.
[19] E. B. Sasson, A. Chiesa and C. Garman, "Zerocash: Decentralized anonymous payments from bitcoin," in *2014 IEEE Symp. on Security and Privacy*, America, IEEE, pp. 459–474, 2014.

[20] C. Gentry, "Fully homomorphic encryption using ideal lattices," in *Proc. of the Forty-First Annual ACM Symp. on Theory of Computing*, America, pp. 169–178, 2009.
[21] Z. Brakerski and V. Vaikuntanathan, "Efficient fully homomorphic encryption from (standard) LWE," *SIAM Journal on Computing*, vol. 43, no. 2, pp. 831–871, 2014.
[22] Z. Brakerski and V. Vaikuntanathan, "Fully homomorphic encryption from ring-LWE and security for key dependent messages," in *Advances in Cryptology—CRYPTO*, Berlin, Heidelberg, Springer, pp. 505–524, 2011.
[23] C. Gentry, A. Sahai and B. Waters, "Homomorphic encryption from learning with errors: Conceptually-simpler, asymptotically-faster, attribute-based," in *Annual Cryptology Conf.*, Berlin, Heidelberg, Springer, pp. 75–92, 2013.
[24] J. H. Cheon, M. Kim and M. Kim, "Search-and-compute on encrypted data," in *Int. Conf. on Financial Cryptography and Data Security*, Berlin, Heidelberg, Springer, pp. 142–159, 2015.
[25] P. Yu, S. F. Zhang and J. Zhong, "Block-chain privacy protection based on fully homomorphic encryption," in *Proc. of the 2019 3rd Int. Conf. on Innovation in Artificial Intelligence*, America, pp. 239–242, 2019.
[26] R. J. Wang, Y. C. Tang, W. Q. Zhang and F. L. Zhang, "Privacy protection scheme for internet of vehicles based on homomorphic encryption and block chain technology," *Chinese Journal of Netword and Information Security*, vol. 6, no. 1, pp. 46–53, 2020.
[27] V. Lyubashevsky, "Fiat-shamir with aborts: Applications to lattice and factoring-based signatures," in *International Conf. on the Theory and Application of Cryptology and Information Security*, Berlin, Heidelberg, Springer, pp. 598–616, 2009
[28] D. Cash, D. Hofheinz and E. Kiltz, "Bonsai trees, or how to delegate a lattice basis," in *Annual Int. Conf. on the Theory and Applications of Cryptographic Techniques*, Berlin, Heidelberg, Springer, pp. 523–552, 2010.
[29] D. Micciancio and C. Peikert, "Trapdoors for lattices: Simpler, tighter, faster, smaller," in *Annual Int. Conf. on the Theory and Applications of Cryptographic Techniques*, Berlin, Heidelberg, Springer, pp. 700–718, 2012.
[30] L. Mei, C. Xu, L. Xu, X. Yu and C. Zuo, "Verifiable identity-based encryption with keyword search for iot from lattice," *Computers, Materials & Continua*, vol. 68, no. 2, pp. 2299–2314, 2021.
[31] S. Agrawal, D. Boneh and X. Boyen, "Efficient lattice (H)IBE in the standard model, proceedings of eurocrypt," in *Annual Int. Conf. on the Theory and Applications of Cryptographic Techniques*, Berlin, Heidelberg, Springer, pp. 553–572, 2010.
[32] F. H. Wang, Z. H. Liu and C. X. Wang, "Full secure identity-based encryption scheme with short public key size over lattices in the standard model," *International Journal of Computer Mathematics*, vol. 93, no. 6, pp. 854–863, 2016.