Attribute Based Encryption with Efficient Revocation from Lattices

Kang Yang[1], Guohua Wu[2], Chengcheng Dong[1], Xingbing Fu[2,3], Fagen Li[4], Ting Wu[2]
(Corresponding author: Guohua Wu)

School of Computer Science and Technology, Hangzhou Dianzi University[1]
Hangzhou, Zhejiang Province, China
School of Cyberspace, Hangzhou Dianzi University[2]
Hangzhou, Zhejiang Province, China
Lab of Security Insurance of Cyberspace[3]
Sichuan Province
School of Computer Science and Engineering, University of Electronic Science and Technology of China[4]
Chengdu, China
(Email: drwuguohua@163.com)

(Received Sept. 7, 2018; Revised and Accepted Jan. 7, 2019; First Online Jan. 24, 2019)

Abstract

Attribute-based encryption (ABE) can be used in many cloud storage and computing applications, and it is an attractive alternative to identity-based encryption. The feature of the ABE is that it provides a flexible mechanism to achieve fine-grained access control. The revocable ABE (RABE) is an extension of the ABE. The attribute revocation is essential because of the factors such as changes of user's attributes, key exposures and key loss. In this paper, we propose a revocable ciphertext policy ABE (CP-RABE) scheme from lattices, which supports flexible access control and efficient revocation. In our scheme, a binary tree with an attribute revocation list is used to revoke attributes, and key update is logarithmically related to the number of each user attribute. Finally, the security of the scheme is proved to be selective-attribute secure in the standard model and security can be reduced to hardness of learning with error assumption.

Keywords: Attribute Based Encryption; Attribute Revocation; Binary Tree; Lattice Based Cryptography

1 Introduction

Attribute based encryption first proposed by Sahai and Waters [25] is a type of public key encryption [9, 28], and it provides a flexible mechanism with fine-grained access control. In attribute based encryption schemes [19], both the ciphertext and the key are associated with a set of attributes. According to the contents to be encrypted or the receiver's attributes, a sender can specify the access control policy such that only users whose attributes satisfy the access control policy can decrypt the encrypted ciphertext. Attribute based encryption is classified as key policy attribute based encryption (KP-ABE) [21] and ciphertext policy attribute based encryption (CP-ABE) [10,20]. In a KP-ABE scheme, the private key is associated with an access policy, and the ciphertext is associated with a set of attributes. On the contrary, in a CP-ABE scheme, the ciphertext is associated with an access policy, and the private key is associated with a set of attributes. In general, a CP-ABE scheme is more flexible than a KP-ABE scheme, since the data sender can specify the access policy when encrypting the message, instead of the key authority setting policy when user's key is extracted.

In recent years, researchers have proposed various ABE schemes [11, 27, 30]. Meanwhile, the attribute based encryption schemes from lattices [3,6,7,12,13,17] have got a great deal of attention from the cryptographic researcher. The constructions of lattice based encryption schemes are highly efficient, and its operation is fast and secure. Moreover, the lattice based encryption schemes are considered to be resistant to quantum attacks since there is no known algorithm which can break the lattice based encryption schemes.

When ABE schemes are used in practical scenarios, due to factors such as changes of user permissions and key exposures [26], it is inevitable to consider the issue of attribute revocation. The revocable attribute based encryption can be used for fine-grained access control of encrypted data in cloud computing [18] or Internet of Things. The ABE revocation scheme was first proposed in [23] where a key authority establishes an attribute revocation list and sets a valid period for user's attributes. The attribute revocation list is periodically updated according to the expiration date, and the scheme obtains the revocation of attributes by updating the latest version of attributes. According to the scope of attribute revocation, ABE revocation schemes mainly include three types: users' revocation, revocation of users' part attributes and revocation of system attributes. With no affect to other users, the users' revocation is the revocation on all attributes contained in the attribute set of the given user, while the revocation of users' part attributes is that the user's part attributes are revoked, while the remaining attributes are not revoked. The certain user who is performed the revocation operation loses the permissions corresponding to the attributes which are revoked, but remaining users still hold the permissions of these attributes. However, when the revocation operation is about the system attributes, all users will lose the permissions of revoked attributes. According to different revocation performers, the current ABE revocation schemes are divided into two types: direct revocation and indirect revocation. The direct revocation is performed by the sender, who directly adds the revocation list of the user when encrypting message, which obtains the revocation of attributes. However, the indirect revocation is performed by the key authority, who periodically updates the unrevoked user's key. Only can the unrevoked user's key be updated, and the unrevoked user decrypt the ciphertext with the new key, while the revoked user will not be able to receive the updates, which will result in the invalidation of his key.

Inspired by the revocable identity based encryption scheme [8], this paper achieves an attribute based encryption scheme from lattices that supports user's attribute revocation. The scheme is an indirect revocation scheme and has been proved to be selective-attribute secure in the standard model.

Our contributions. We propose a revocable attribute based encryption scheme, which supports flexible threshold access control [29]. The following building blocks are used in our scheme: (1) Based on Chen et al.'s scheme [8], we propose an lattice based attribute based encryption scheme, and the scheme empolys a binary tree to support attribute revocation. It achieves flexible threshold access control and increases expressiveness of the scheme. (2) Using the Shamir secret sharing scheme [4] to recover key, our scheme chooses a random polynomial, and associates each attribute with a component of the key. (3) Our scheme proposes tuples $(key, value)$ associated with all nodes of a binary tree [5], and achieves the user's attribute revocation by updating key. The binary tree improves the efficiency of key update and makes the workload of key update logarithmically related with the maximal number of each user's attributes.

From four aspects, we have compared our scheme and other schemes in **Table 1**.

Table 1: Feature comparisons

Scheme	Attribute based	Quantum security	Revocable	Standard model
Agrawal et al. [1]	no	yes	no	no
Agrawal et al. [2]	no	yes	no	yes
Zhang et al. [29]	yes	yes	no	yes
Chen et al. [8]	no	yes	yes	yes
Sahai et al. [24]	yes	no	yes	yes
Hur et al. [16]	yes	no	yes	yes
Our scheme	yes	yes	yes	yes

Our Techniques. In our construction, each user is associated with a binary tree and the user's attributes are associated with the leaf nodes of the binary tree, numbering all nodes of the binary tree from 1 to ξ as shown in **Figure** 1. We use tuples $(key, value)$ to store some specific information for each node of the binary tree. The key is the number of the node and $value$ is the set of attribute's leaf nodes owned by the user j when we consider the current node as the root node. When the attribute i of the user j is revoked, all nodes on the path from the root node to the leaf node are added to the revocation list RL_j. Traversing the path, adding the revoked node to the set S_1 and the unrevoked children of the revoked node to the set S_2. When obtaining the decryption key, we need to determine whether the elements' number of intersection of the set of all $value$ in the set S_2 and the set of attributes in the ciphertext policy W is equal or greater than the system threshold k. If so, the user j can obtain the decryption key. If not, return $\perp$. The time complexity of key update can be reduced to a logarithmic relation with the maximal number of each user's attributes.

We define a system attribute set $\mathcal{Q} = \{1, 2, \ldots, f\}$ and a default attribute set $\mathcal{M} = \{f+1, f+2, \ldots, f+l\}$, let $\mathcal{Q}^{'} = \mathcal{Q} \cup \mathcal{M}$. When a user with an attribute set $\mathcal{G}$ is added to the system, where $\mathcal{G} \subset \mathcal{Q}$, the scheme at random chooses a vector $\mathbf{u} = (u_1, \ldots, u_n) \in \mathbb{Z}_q^n$, and let $\mathcal{G}^{'} = \mathcal{G} \cup \mathcal{M}$. Applying Shamir's secret sharing scheme, the vector $\mathbf{u}$ is divided into i parts, where $i \in \mathcal{G}^{'}$, and each vector $\hat{\mathbf{u}}_i$ is at random divided into two vector $\hat{\mathbf{u}}_{i,1}, \hat{\mathbf{u}}_{i,2}$ that are associated with attributes and time, respectively. The sender sets an access structure W and a threshold k to encrypt a message, and let $W^{'} = W \cup \{f+1, f+2, \ldots, f+l+1-k\}$. When decrypting, if there are unrevoked attributes in the user's attribute set $\mathcal{G}$ and $| \mathcal{G} \cap W |< k$, the user can't obtain the decryption key. If $| \mathcal{G} \cap W |\geq k$, the user can obtain the decryption key to decrypt the ciphertext. Then $| \mathcal{G}^{'} \cap W^{'} |\geq l+1$, choose a subset P such that $| P |= l+1$. Finally, we show that our scheme is secure in the standard model.

Related work. There are many attribute based encryption schemes that support attribute revocation [14–16, 21, 24], most of which are indirect revocation schemes. We introduce three typical works as follows.

- Goyal et al. [14] limited the validity of the key by adding an extra expiration attribute to each user and achieved the attribute revocation by updating the key of the expiration attribute. However, the key authority needs to regularly distribute keys to users who have not been revoked permissions. The workload of its key authority is linear in the number of users in the system, and it also requires a secure channel between the key authority and each user.

- Hur et al. [16] proposed an attribute based encryption scheme that supports immediate revocation in the context of outsourcing ciphertext. In their scheme, the sender sends the ciphertext to data outsourcing server, and data outsourcing server reencrypts the ciphertext. Only can users whose attributes have not been revoked obtain the updated key and decrypt the new ciphertext. However, the scheme has expensive cost on key maintenances and cannot resist quantum attacks.

- Using a binary tree, Sahai et al. [24] set each user to be associated with leaf nodes such that the complexity of key update is logarithmical in the number of users in the system. Combining the nature of "ciphertext delegation", an efficient encryption scheme with attribute revocation is proposed. The key authority only needs to periodically send update key to the receiver to obtain attribute revocation, which reduces the workload of key update.

However, these schemes are all built on the traditional bilinear pairing. Bilinear pairing has its own fatal flaw, and if quantum computers are invented, cryptographic schemes based on bilinear pairing will no longer be secure. Chen et al. [8] applied the work of Sahai et al. [24] to lattices and proposed a revocable identity based encryption scheme.

2 Preliminaries

2.1 Notation

We use lowercase boldface alphabet for vectors such as $\mathbf{e}$; uppercase boldface alphabet for matrices such as $\mathbf{A}$; lowercase regular alphabet for scalars such as l. q represents a prime number, $\mathbb{R}$ represents a real number set and $\mathbb{Z}$ represents an integer set. For the positive integer f, $[f]$ denotes $(1,\ldots,f)$ and the system security parameter is n. The length of a matrix is the length of its longest vector norm: $\|\mathbf{X}\| = max\|\mathbf{x}_i\|$. ϵ: $\mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$ is a negligible function if $\epsilon(\lambda)$ is smaller than all polynomial fractions for sufficiently large λ, we call an event non-negligible if its probability is $1-\epsilon(\lambda)$.

2.2 Syntax of CP-RABE

Definition 1. *A revocable ciphertext policy lattice based attribute based encryption scheme* $\mathbf{CP-RABE}$ = { **Setup**, **PriKeyGen**, **KeyUpd**, **DecKeyGen**, **Enc**, **Dec**, **AttRev** } *consists of the following seven probabilistic polynomial time(PPT) algorithms:*

RABE.**Setup**$(1^\lambda, \mathcal{Q}, N) \rightarrow (pp, msk, RL_j)$. *The algorithm takes a security parameter* λ*, an attribute set* $\mathcal{Q}$ *and a maximum number of users* N *as input, and returns a public parameters* pp*, a master key* msk *and revocation lists* $RL_j,\ j \in N$.

RABE.**PriKeyGen**$(\mathrm{pp}, \mathrm{msk}, \mathcal{G}) \rightarrow \mathrm{SK}_{\mathcal{G}}$. *The algorithm takes the master key* msk*, the public parameters* pp *and an attribute set* $\mathcal{G} \in \mathcal{Q}$ *as input, and returns a private key* $\mathrm{SK}_{\mathcal{G}}$ *associated with the attribute set* $\mathcal{G}$.

RABE.**KeyUpd**$(\mathrm{pp}, \mathrm{msk}, t, RL_j) \rightarrow \mathrm{KU}_t$.*The algorithm takes the public parameters* pp*, the master key* msk*, an update time* $t \in \mathcal{T}$ *and revocation lists* RL_j *as input, and returns a key update* KU_t.

RABE.**DecKeyGen**$(\mathrm{SK}_{\mathcal{G}}, \mathrm{KU}_t, (W, k)) \rightarrow \mathrm{DK}_{\mathcal{G},t}$.*The algorithm takes a private key* $\mathrm{SK}_{\mathcal{G}}$*, a key update* KU_t*, an access structure* W*, and a system threshold* k *as input, and returns a decription key* $\mathrm{DK}_{\mathcal{G},t}$ *indicating the user has enough attributes to decrypt or a special symbol* $\perp$ *meaning that some attributes of the user are revoked.*

RABE.**Enc**$(\mathrm{pp}, (W, k), t, M) \rightarrow \mathrm{CT}_{W,t}$. *The algorithm takes an access structure* W*, a threshold* k*, the public parameters* pp*, a message* $M \in M_0$ *and an encryption time* $t \in \mathcal{T}$ *as input, and returns a ciphertext* $\mathrm{CT}_{W,t}$.

RABE.**Dec**$(\mathrm{DK}_{\mathcal{G},t}, \mathrm{CT}_{W,t}) \rightarrow M$. *The algorithm takes the decryption key* $\mathrm{DK}_{\mathcal{G},t}$ *and ciphertext* $\mathrm{CT}_{W,t}$ *as input, and returns the decryption message* M.

RABE.**RevListUpd**$(\mathcal{G}, t, RL_j) \rightarrow \widetilde{RL}_j$. *The algorithm takes an attribute set* $\mathcal{G}$*, a revocation time* $t \in \mathcal{T}$ *and revocation lists* RL_j *as input, and returns updated revocation lists* $\widetilde{RL}_j$.

In order to ensure the validity of the time t, the message M, and the attribute set $\mathcal{G}$, all $t \in \mathcal{T}$, $M \in M_0$, and $\mathcal{G} \in \mathcal{Q}$. The algorithms **Setup**, **PriKeyGen**, **KeyUpd**, and **RevListUpd** are run by the key authority, the algorithm **Enc** is run by the sender, and the algorithms **DecKeyGen** and **Dec** are run by the receiver.

2.3 Security Model of CP-RABE

The security model of the CP-RABE scheme under the selective-attribute and chosen plaintext attack(*IND-sAtt-CPA*) will be given below. In the model, the adversary needs to provide a challenge access structure before the system is set up. For example, the adversary chooses an access structure (W^*, k^*) before obtaining the private keys in *Phase* 1, then the attribute set $\mathcal{G}$ choosed by the adversary must satisfy $\mathcal{G} \subsetneq W^*$. The security game between an adversary $\mathcal{A}$ and a challenger $\mathcal{C}$ is as follows:

Init: The adversary $\mathcal{A}$ announces to a challenger $\mathcal{C}$ a revocation list RL_j on time period t^* and the challenge access structure (W^*, k^*).

Setup: The challenger $\mathcal{C}$ uses the **Setup** algorithm to generate the public parameters pp and the master key msk, sends the public parameters pp to the adversary $\mathcal{A}$, and holds the master key msk by himself.

Phase 1: The adversary $\mathcal{A}$ arbitrarily chooses the attribute set $\mathcal{G} = \{a_i | a_i \notin W^*\}$ and initiates a request to the challenger $\mathcal{C}$ to get the private key. The challenger $\mathcal{C}$ runs the **KeyGen** algorithm to answer the adversary's request. The adversary $\mathcal{A}$ initiates a request for updating the private key to the challenger $\mathcal{C}$ according to the revocation list RL_j, and the challenger $\mathcal{C}$ runs the **KeyUpd** algorithm to answer the adversary's request. The adversary $\mathcal{A}$ is allowed to query only during time periods are increased.

Challenge: The adversary $\mathcal{A}$ sends two message bits m_0 and m_1 to the challenger $\mathcal{C}$. The challenger $\mathcal{C}$ performs a fair coin-toss and chooses $b \in \{0, 1\}$ to run the **Enc** algorithm, and sends the challenge ciphertext to the adversary $\mathcal{A}$.

Phase 2: Similar as *Phase* 1, the adversary $\mathcal{A}$ continues to make a request to the challenger $\mathcal{C}$.

Guess: The adversary $\mathcal{A}$ guesses $b' \in \{0, 1\}$. If $b' = b$, then the adversary $\mathcal{A}$ succeeds in attacks.

The advantage of the adversary's success in the game is defined as

$$Adv^{IND-sAtt-CPA}_{\mathcal{CP}-\mathcal{RABE},\mathcal{A}}(\nu) = |Pr[b = b'] - \frac{1}{2}|,$$

where the probability depends on the probability distribution of random parameters and internal random coin tosses.

Definition 2. *A* $\mathrm{CP-RABE}$ *scheme is said to be secure against IND-sAtt-CPA secure if the advantage* $\mathrm{Adv}^{\mathrm{IND-sAtt-CPA}}_{\mathcal{CP}-\mathcal{RABE},\mathcal{A}}(\nu)$ *is a negligible function in* ν *for all polynomial time adversary* $\mathcal{A}$.

3 Background

We describe the required background knowledge as follows.

3.1 Integer Lattices

Definition 3. ([1], Definition 2). *Given any* m *linearly independent vectors* $\mathbf{a}_1$, $\mathbf{a}_2$, $\ldots$, $\mathbf{a}_m \in \mathbb{Z}^m$, *we call linear combinations of their integral coefficients as* $\mathcal{L}(\mathbf{A})$, *where* $\mathbf{A} = \{\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_m\}$, *then:*

$$\Lambda := \mathcal{L}(\mathbf{A}) := \{\mathbf{y} \in \mathbb{R}^m \ s.t. \ \exists \ \mathbf{c} = (c_1, \ldots, c_n) \in \mathbb{Z}^n, \ \mathbf{y} = \mathbf{A}\mathbf{s} = \Sigma_{i=1}^{n} c_i a_i\}.$$

Definition 4. *For a matrix* $\mathbf{B} \in \mathbb{Z}_q^{n\times m}$, *a vector* $\mathbf{u} \in \mathbb{Z}_q^n$, *and a prime number* q, *then:*

$$\Lambda_q^{\perp}(\mathbf{B}) = \{\mathbf{s} \in \mathbb{Z}^m \quad s.t. \quad \mathbf{B}\mathbf{s} = 0 \ (mod \ q)\}$$
$$\Lambda_q^{\mathbf{u}}(\mathbf{B}) = \{\mathbf{s} \in \mathbb{Z}^m \quad s.t. \quad \mathbf{B}\mathbf{s} = \mathbf{u} \ (mod \ q)\}.$$

3.2 Trapdoors for Lattices

Theorem 1. **[22].** *Let a prime* $q \geq 2$, $m > 5n\log_2 q$, *and a positive integer* n, *there is a PPT algorithm* **TrapdoorGen**(q, n), *output a matrix* $\mathbf{B} \in \mathbb{Z}_q^{n\times m}$ *and* $T_{\mathbf{B}} \in \mathbb{Z}_q^{m\times m}$, *where* $\mathbf{B}$ *is statistically uniform on* $\mathbb{Z}_q^{n\times m}$, $T_{\mathbf{B}}$ *is the base of the lattice* $\Lambda_q^{\perp}(\mathbf{B})$ *and* $\| \widetilde{T_{\mathbf{B}}} \| \leq O(\sqrt{n\log_2 q})$.

3.3 Discrete Gaussians

Definition 5. **[1].** *For any real number* $\mathbf{r} > 0$, *Gaussian function with* r *as the parameter and* $\mathbf{c}$ *as the center on* $\mathbb{R}^n$, *defined as follows:*

$$\forall x \in \mathbb{R}^n, \ \rho_{r,\mathbf{c}}(\mathbf{x}) = exp(-\pi\tfrac{\|x-\mathbf{c}\|^2}{r^2}).$$

When $\mathbf{c}$ *is the origin or* $r = 1$, *the subscript can be omitted.*

For any $\mathbf{c} \in \mathbb{R}^n$, *the real* $r > 0$ *and* n*-dimensional lattice* $\mathcal{L}$, *the discrete Gaussian distribution on lattices is defined as:*

$$\forall \mathbf{y} \in \mathcal{L} \ , \ \mathcal{D}_{\mathcal{L},r,\mathbf{c}}(\mathbf{x}) = \tfrac{\rho_{r,\mathbf{c}}(\mathbf{x})}{\rho_{r,\mathbf{c}}(\mathcal{L})}$$

For any countable set $\mathbf{B}$, $\rho_{r,\mathbf{c}}(\mathbf{B}) = \Sigma_{\mathbf{x}\in\mathbf{B}}\rho_{r,\mathbf{c}}(\mathbf{x})$.

3.4 Sampling Algorithms

SampleLeft algorithm [29]. *SampleLeft*$(\mathbf{A}$, $\mathbf{B}$, $\mathbf{T_A}$, $\mathbf{u}$, $s) \mapsto \mathbf{e}$. *Given a full rank matrix* $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$, *a matrix* $\mathbf{B} \in \mathbb{Z}_q^{n\times m_1}$, *a basis* $\mathbf{T_A}$ *for* $\Lambda_q^{\perp}(\mathbf{A})$, *a vector* $\mathbf{u} \in \mathbb{Z}_q^n$, *and a Gaussian parameter* $s > \|\widetilde{\mathbf{T_A}}\| \cdot \omega(\sqrt{\log(m + m_1)})$, *outputs a vector* $\mathbf{e} \in \mathbb{Z}^{m+m_1}$ *sampled from a distribution statistically close to* $D_{\Lambda_q^{\mathbf{u}}([\mathbf{A}\|\mathbf{B}]),s}$.
SampleRight algorithm [29].*SampleRight* $(\mathbf{A}, \mathbf{B}$, $\mathbf{R}$, $\mathbf{T_G}$, $\mathbf{u}$, $s) \mapsto \mathbf{e}$. *Given a full rank matrix* $\mathbf{A}, \mathbf{B} \in \mathbb{Z}_q^{n\times m}$, *a matrix* $\mathbf{R} \in \mathbb{Z}^{m\times m}$, *a vector* $\mathbf{u} \in \mathbb{Z}_q^n$, *a basis* $\mathbf{T_B}$ *for* $\Lambda_q^{\perp}(\mathbf{B})$, *and a Gaussian parameter* $s > \|\widetilde{\mathbf{T_B}}\| \cdot \|\mathbf{R}\| \cdot \omega(\sqrt{\log m})$ *outputs a vector* $\mathbf{e} \in \mathbb{Z}^{2m}$ *sampled from a distribution statistically close to* $D_{\Lambda_q^{\mathbf{u}}([\mathbf{A}\|\mathbf{A}\mathbf{R}+\mathbf{B}]),\sigma}$.

3.5 The LWE Hardness Assumption

Definition 6. **[1].** *Decisional Learning With Errors (DLWE). Let* q *be a prime number and* n *be a positive integer. For any* $a > 0$, *define 0 is the center of* Ψ_a, *and the normal distribution on* $[0, 1)$ *with variance* $a/\sqrt{2\pi}$, *the discrete distribution on the corresponding* $\mathbb{Z}_q$ *is* $\overline{\Psi}_a$. *Suppose the learning with error* χ *on* $\mathbb{Z}_q$, *define the distribution* $\mathbf{A}_{s,\chi}$ *on* $(\mathbf{u}_i, v_i) = (\mathbf{u}_i, \mathbf{u}_i^T s + x_i) \in \mathbb{Z}_q^n \times \mathbb{Z}_q$, *where* $\mathbf{u}_i \in \mathbb{Z}_q$ *is a randomly selected vector,* $x_i \in \mathbb{Z}_q$ *is independently selected according to the distribution* χ. *The decision* $(\mathbb{Z}_q, n, \chi) - LWE$ *is to distinguish between the pseudo-random distribution and the true random distribution on* $\mathbf{A}_{s,\chi}$ *and* $\mathbb{Z}_q^n \times \mathbb{Z}_q$.

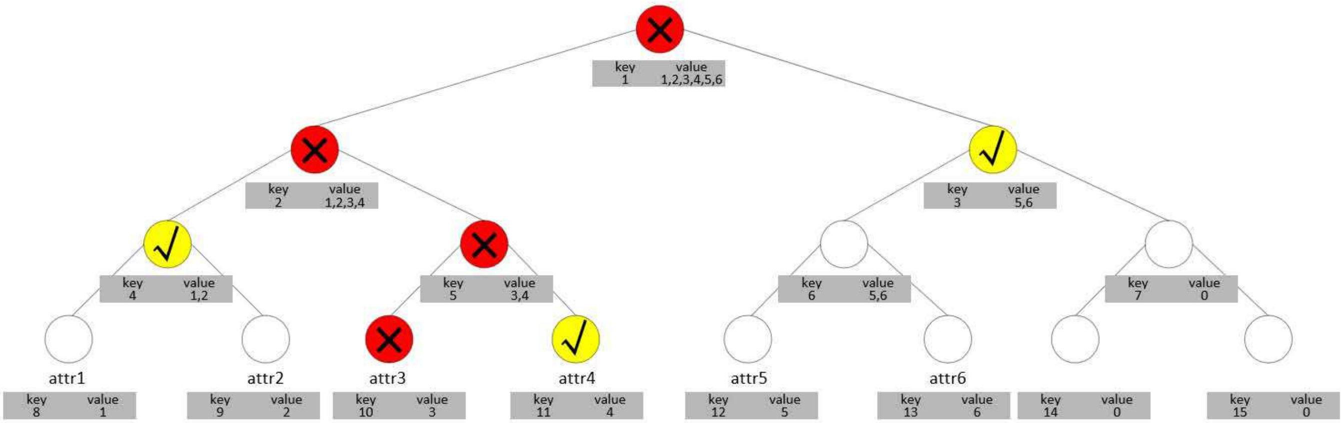

Figure 1: Description of the KUNodes algorithm on binary tree

3.6 Encoding Attributes and Time as Matrices

Definition 7. [8]. *Let q be a prime number, m be a positive integer. A full rank difference (FRD) map function $\mathbf{H} : \mathbb{Z}_q^m \longrightarrow \mathbb{Z}_q^{m\times m}$, It has two properties. One is that for all different $i, t \in \mathbb{Z}_q^m$, the matrix $\mathbf{H}(i) - \mathbf{H}(t) \in \mathbb{Z}_q^{m\times m}$ is full rank; the other is that H is computable in the polynomial time of $O(m \log q)$.*

3.7 The Binary Tree Data Structure

Our construction takes advantage of binary tree to support attribute revocation, as with [5, 8]. In our scheme, the user j is associated with a binary tree BT_j. Each attribute i of the user j is associated with a leaf node, the path $Path(i)$ denotes a set of all nodes from the leaf node i to the root node. All nodes are associated with tuples $(key, value)$, the key is set to the number of the node, and the $value$ is the set of attribute's leaf nodes owned by the user j when we consider the current node as the root node. The $value$ is 0 when the current node doesn't have any attribute leaf nodes. If ξ is an intermediate node, ξ_l and ξ_r represent the left and right child node of the node ξ, respectively. t_i is the revocation time of the attribute i. S_1 is the set of all nodes in $Path(i)$ whose attribute i was revoked after time t, and S_2 is the set of non-revoked child nodes whose attribute i was revoked after time t.

For leaf node $(i, t_i) \in RL_j$ of the user j, if all nodes $\xi \in RL_j$ in $Path(i)$, then add $Path(i)$ to the set S_1. For all nodes $\xi \in S_1$, if $\xi_l \notin S_{1,j}$, then add $value_{\xi_l}$ to the set S_2. If $\xi_r \notin S_{1,j}$, then add $value_{\xi_r}$ to the set S_2. If the set S_2 is empty, then add the root node to the set S_2. By running the **KUNodes** algorithm, all parent nodes of the revoked node are revoked. The algorithm outputs all non-revoked child nodes of the revoked node, indicating that the user's attributes were not revoked at the time t. The **KUNodes** algorithm that obtains the attribute revocation is as follows:

```
**KUNodes**$(BT_j, RL_j, t)$
    $S_1, S_2 \leftarrow \emptyset$
    $\forall(i, t_i) \in RL_j$
        if $t_i \leq t$ then add $Path(i)$ to $S_1$
    $\forall\xi \in S_1$
        if $\xi_l \notin S_1$ then add $value_{\xi_l}$ to $S_2$
        if $\xi_r \notin S_1$ then add $value_{\xi_r}$ to $S_2$
    if $\mid S_2 \mid= 0$ then add root to $S_2$
    Return $S_2$
```

We give an example to illustrate our attribute revocation method as follows.

As shown in **Figure** 1, the nodes 2 to 15 are associated with a tuple $(key, value)$, respectively. The $value$ of the node 2 is the set of the attributes $attr1, attr2, attr3$ and $attr4$ because these attribute nodes are the leaf nodes of the node 2, the $value$ of other nodes is calculated in the same way. Assume the user 1 owns the attributes $attr1, attr2, attr3, attr4, attr5, attr6$. When $attr3$ is revoked, the nodes 1, 2, 5 and 10 are added to the set X, and the $value = 1, 2$, $value = 4$, $value = 5, 6$ are added to the set Y, so **KUNodes**$(BT_j, RL_j, t) \rightarrow Y = \{1, 2, 4, 5, 6\}$. Assume the access structure $W = \{1, 2, 3\}$ and the system threshold $k = 3$, then

$$IN = W \cap \textbf{KUNodes}(BT_j, RL_j, t) = \{1,2\}$$

Because of $\mid IN \mid < k$, it means that the user 1 has not decryption permissions.

4 A New CP-RABE Scheme from Lattices

In this section, we propose a revocable ciphertext policy lattice based attribute based encryption scheme. Unlike previous bilinear pairing based cryptographic schemes, the scheme is built on the mathematical structure of lattices. For convenience, it is assumed that there are f attributes in the system, and $\mathcal{Q} = \{1, 2, ..., f\}$, representing a set of all attributes. The ciphertext CT is associated with an access policy (W, k), where $W \subset \mathcal{Q}$ is an attribute set, an integer k represents a threshold which is up bounded by a system parameter l, and $D = ((f+l)!)^2$. The access policy (W, k) indicates that the scheme can be successfully decrypted when the set of attributes associated with the key intersects W by more than or equal to k.

RABE.**Setup**$(1^\lambda, \mathcal{Q}, N)$. On input a security parameter λ, a system attribute set $\mathcal{Q} = \{1, 2, ..., f\}$ and a maximum number of users N in the system, as described in the construction framework of Section 2.2. The system sets the parameters $q, m, n, l, \sigma, \alpha$. Do:

1) Select a default set of attributes $\mathcal{M} = \{f+1, f+2, ..., f+l\}$, let $\mathcal{Q}^{'} = \mathcal{Q} \cup \mathcal{M}$.
2) Call the **TrapdoorGen(q,n)** algorithm to generate a uniformly random matrix $\mathbf{A}_0 \in \mathbb{Z}_q^{n\times m}$ with a basis $\mathbf{T}_{\mathbf{A}_0} \in \mathbb{Z}_q^{m\times m}$ for a lattice $\mathbf{\Lambda}_q^{\perp}(\mathbf{A}_0)$ such that $\| \widetilde{\mathbf{T}_{\mathbf{A}_0}} \| \leq \mathcal{O}(n\sqrt{\log q})$.
3) Select uniformly random matrices $\mathbf{B_1}$, $\mathbf{B_2}, \mathbf{C_1}, \mathbf{C_2} \in \mathbb{Z}_q^{n\times m}$ and a vector $\mathbf{u} = (u_1, \ldots, u_n) \in \mathbb{Z}_q^n$. For each $i \in \mathcal{Q}^{'}$, $\mathbf{a}_i \in \mathbb{Z}_q^n$ is randomly selected. Select a FRD map H.
4) Let RL_j be initially an empty list, where $j \in N$, and BT_j be a binary tree.
5) Output revocation lists RL_j, the public parameters pp, and the master key msk,

$$\text{pp} = (\mathbf{A_0}, \mathbf{B_1}, \mathbf{B_2}, \mathbf{C_1}, \mathbf{C_2}, \{\mathbf{a}_i\}_{i\in\mathcal{R}'}, \mathbf{u}, \text{H}),$$
$$\text{msk} = \{\mathbf{T}_{\mathbf{A}_0}\}.$$

RABE.**PriKeyGen**$(\text{pp}, \text{msk}, \mathcal{G})$. On input the public parameters pp, the master key msk and a user attribute set $\mathcal{G} \subset \mathcal{Q}$, and let $\mathcal{G}^{'} = \mathcal{G} \cup \mathcal{M}$. Do:

1) For $i = 1, 2, \ldots, n$, randomly choose degree d polynomial $p_i(x) \in \mathbb{Z}_q[x]$ such that $p_i(0) = u_i$. For each attribute in $i \in \mathcal{G}^{'}$, let $\hat{\mathbf{u}}_i = (p_1(i), \ldots, p_n(i))^T \in \mathbb{Z}_q^n$. Note that, for any subset $P \subseteq \mathcal{G}^{'} with\ |P| = l+1$, we have $\mathbf{u} = \Sigma_{i\in P} L_i \cdot \hat{\mathbf{u}}_i$, where the Lagrangian coefficient $L_i = \frac{\Pi_{j\in J, j\neq i} - j}{\Pi_{j\in J, j\neq i} i - j}$.
2) All nodes of the binary tree are sequentially numbered from the root node, the number of the root node is 1 and the number of the ξ-th node is ξ. For any node ξ in $Path(i)$ of the current user, $\hat{\mathbf{u}}_{i,\xi,1} \in \mathbb{Z}_q^n$ are randomly choosed, and let $\hat{\mathbf{u}}_{i,\xi,2} = \hat{\mathbf{u}}_i - \hat{\mathbf{u}}_{i,\xi,1}$, it is stored in the node ξ.
3) Calculate **SampleLeft**$(\mathbf{A}_0,\ \text{H}(\mathbf{a}_i)\ \mathbf{C_1} + \mathbf{B}_1,\ \mathbf{T}_{\mathbf{A}_0},\ \sigma, \hat{\mathbf{u}}_{i,\xi,1}) \to \mathbf{e}_{i,\xi,1}$, output node private key $\text{SK}_\mathcal{G}$,

$$\text{SK}_\mathcal{G} = (\xi, \mathbf{e}_{i,\xi,1}, value_\xi)_{\xi \in Path(i)}$$

RABE.**KeyUpd**$(\text{pp}, \text{msk}, t, RL_j)$. On input the public parameters pp, the master key msk, a key update time t and revocation lists RL_j, we think t as a vector $\mathbf{t} \in \mathbb{Z}_q^n$. Do:

1) For any node $\xi \in$ **KUNodes** $(BT_j,\ RL_j,\ t)$, if $\hat{\mathbf{u}}_{i,\xi,1}, \hat{\mathbf{u}}_{i,\xi,2}$ are not defined, then generate $\hat{\mathbf{u}}_{i,\xi,1}$, $\hat{\mathbf{u}}_{i,\xi,2}$ according to the **PriKeyGen**$(\text{pk}, \text{msk}, \mathcal{G})$ algorithm.
2) Calculate **SampleLeft**$(\mathbf{A}_0,\ \text{H}(\mathbf{t})\mathbf{C_2} + \mathbf{B}_2,\ \mathbf{T}_{\mathbf{A}_0},\ \sigma,\ \hat{\mathbf{u}}_{i,\xi,2}) \to \mathbf{e}_{i,\xi,2}$, where H is a function that maps time t to an n×m matrix. Output an updated key:

$$\text{KU}_t = (\xi, \mathbf{e}_{i,\xi,2}, value_\xi)_{\xi \in \mathbf{KUNodes}(BT, RL_j, t)}.$$

RABE.**DecKeyGen**$(\text{SK}_\mathcal{G}, \text{KU}_\text{t}, (W, k))$. On input two sets $\text{SK}_\mathcal{G} = \{(x, \mathbf{e}_{i,x,1}, value_x)\}_{x\in S_1}$ and $\text{KU}_\text{t} = \{(y, \mathbf{e}_{i,y,2}, value_y)\}_{y\in S_2}$, where S_1 represents nodes contained in path $Path(i)$ of S_1, $i \in \mathcal{G}$, $j \in N$, and S_2 represents unrevoked children of revoked nodes. Do:

1) If the elements' number of intersection of the set of all $value$ in the set S_2 and the set of attributes in the ciphertext policy W is equal or greater than the system threshold k, it means that the user has decryption permissions, then let $\text{DK}_{\mathcal{G},\text{t}} = (\mathbf{e}_{i,x_j,1}, \mathbf{e}_{i,y_j,2})$. If the intersection is less than the system threshold k, it means that the user has not decryption permissions, then output $\text{DK}_{\mathcal{G},\text{t}} = \perp$.
2) For the user $j \in N$ with decryption permissions and the attribute $i \in \mathcal{G}$, there are some of the same nodes between each $Path(i)$ of S_1 and the set S_2, i.e., for the user j, the algorithm finds components of $\text{SK}_\mathcal{G}$ and KU_t such that $\mathbf{F}_i\mathbf{e}_{i,1} + \mathbf{F}_t\mathbf{e}_{i,2} = \hat{\mathbf{u}}_i$ since they are in the same node(*The matrix* $\mathbf{F}_i$ *and* $\mathbf{F}_t$ *will be introduced in the encryption phase*). Because of $x_j = y_j$ in the previous step, we can omit x_j, y_j, then $\text{DK}_{\mathcal{G},\text{t}} = (\mathbf{e}_{i,1}, \mathbf{e}_{i,2})$.

RABE.**Enc**$(\text{pp}, (W, k), t, M)$. On input the public key $\text{pp} = (\mathbf{A}_0, \mathbf{B}_1, \mathbf{B}_2, \mathbf{C_1}, \mathbf{C_2}, \{\mathbf{a}_i\}_{i\in\mathcal{R}'}, \mathbf{u}, \text{H})$, an access structure W, a threshold k, satisfy $1 \leq k \leq min(\mid W \mid, l)$, a message bit m and a time $\mathbf{t} \in \mathbb{Z}_q^n$. Let $W^{'} = W \cup \{f+1, f+2, \ldots, f+l+1-k\}$ and $D = ((f+l)!)^2$. Do:

1) Construct

$$\mathbf{F}_i = (\mathbf{A}_0 \mid \text{H}(\mathbf{a}_i)\mathbf{C_1} + \mathbf{B}_1) \in \mathbb{Z}_q^{n\times 2m},$$
$$\mathbf{F}_t = (\mathbf{A}_0 \mid \text{H}(\mathbf{t})\mathbf{C_2} + \mathbf{B}_2) \in \mathbb{Z}_q^{n\times 2m}$$
$$\mathbf{F}_{i,t} = (\mathbf{A}_0 \mid \text{H}(\mathbf{a}_i)\mathbf{C_1} + \mathbf{B}_1 \mid \text{H}(\mathbf{t})\mathbf{C_2} + \mathbf{B}_2) \in \mathbb{Z}_q^{n\times 3m}$$

2) Choose a uniformly random $\mathbf{s} \in \mathbb{Z}_q^n$.
3) Choose a noise $x \xleftarrow{\overline{\Psi}_\alpha} \mathbb{Z}_q$, a noise vectors $\mathbf{y} \xleftarrow{\overline{\Psi}_\alpha^m} \mathbb{Z}_q^m$.
4) For each attribute $i \in W^{'}$, randomly choose two matrices $\mathbf{R}_{i,1}, \mathbf{R}_{i,2} \in \{-1, 1\}^{m\times m}$, calculate $\mathbf{r}_{i,1} \leftarrow \mathbf{R}_{i,1}^T\mathbf{y} \in \mathbb{Z}_q^m, \mathbf{r}_{i,2} \leftarrow \mathbf{R}_{i,2}^T\mathbf{y} \in \mathbb{Z}_q^m$.
5) Output ciphertext $\text{CT}_{W,t} = (c_0, \mathbf{c}_i) \in \mathbb{Z}_q \times \mathbb{Z}_q^{3m}$, where

$$c_0 \leftarrow \mathbf{u}^T\mathbf{s} + Dx + M\lfloor \tfrac{q}{2} \rfloor \in \mathbb{Z}_q,$$
$$\mathbf{c}_i \leftarrow \mathbf{F}_{i,t}^T\mathbf{s} + D(\mathbf{y}, \mathbf{r}_{i,1}, \mathbf{r}_{i,2})^T \in \mathbb{Z}_q^{3m}$$

RABE.Dec$(\mathrm{DK}_{\mathcal{G},t}, \mathrm{CT}_{W,t})$. On input a decryption key $\mathrm{DK}_{\mathcal{G},t}$ and a ciphertext $\mathrm{CT}_{W,t}$. The user's attribute set $\mathcal{G}$ is associated with $\mathrm{DK}_{\mathcal{G},t}$, and the access structure W is associated with $\mathrm{CT}_{W,t}$. If $\mid \mathcal{G} \cap W \mid < k$, then return $\perp$; otherwise, let $\mathcal{G}^{'} = \mathcal{G} \cup \mathcal{M}$, $W^{'} = W \cup \{f+1, f+2, \ldots, f+l+1-k\}$. Since $\mid \mathcal{G} \cap W \mid \geq k$, there is $\mid \mathcal{G}^{'} \cap W^{'} \mid \geq l+1$. Choose a subset P of $\mid \mathcal{G}^{'} \cap W^{'} \mid$ such that $\mid P \mid = l+1$. Do:

1) Parse c_i as

$$\begin{bmatrix} \mathbf{c}_{i,0} \\ \mathbf{c}_{i,1} \\ \mathbf{c}_{i,2} \end{bmatrix} = \begin{bmatrix} \mathbf{A}_0^T \mathbf{s} \\ (\mathrm{H}(\mathbf{a}_i)\mathbf{C_1} + \mathbf{B}_1)^T \mathbf{s} \\ (\mathrm{H}(\mathbf{t})\mathbf{C_2} + \mathbf{B}_2)^T \mathbf{s} \end{bmatrix} + D \begin{bmatrix} \mathbf{y} \\ \mathbf{r}_{i,1} \\ \mathbf{r}_{i,2} \end{bmatrix},$$

where $\mathbf{c}_{i,0}, \mathbf{c}_{i,1}, \mathbf{c}_{i,2} \in \mathbb{Z}_q^m$.

2) Compute

$$\begin{aligned} \mathbf{c}_i^{'} &= \mathbf{e}_{i,1}^T \begin{bmatrix} \mathbf{c}_{i,0} \\ \mathbf{c}_{i,1} \end{bmatrix} + \mathbf{e}_{i,2}^T \begin{bmatrix} \mathbf{c}_{i,0} \\ \mathbf{c}_{i,2} \end{bmatrix} \\ &= \hat{\mathbf{u}}_i^T \mathbf{s} + D(\mathbf{e}_{i,1}^T \begin{bmatrix} \mathbf{y} \\ \mathbf{r}_{i,1} \end{bmatrix} + \mathbf{e}_{i,2}^T \begin{bmatrix} \mathbf{y} \\ \mathbf{r}_{i,2} \end{bmatrix}) \in \mathbb{Z}_q^n. \end{aligned}$$

3) According to the Lagrangian interpolation formula in Shamir's secret-sharing scheme $\mathbf{u} = \Sigma_{i \in P} L_i \cdot \hat{\mathbf{u}}_i$ to recover

$$\mathbf{c}^{''} = \mathbf{u}^T \mathbf{s} + \sum_{i \in P} DL_i(\mathbf{e}_{i,1}^T \begin{bmatrix} \mathbf{y} \\ \mathbf{r}_{i,1} \end{bmatrix} + \mathbf{e}_{i,2}^T \begin{bmatrix} \mathbf{y} \\ \mathbf{r}_{i,2} \end{bmatrix}),$$

where the Lagrangian coefficient $L_i = \frac{\Pi_{j \in J, j \neq i} - j}{\Pi_{j \in J, j \neq i} i - j}$.

4) Compute $\mathbf{c}^{''}$, if $\mid c_0 - \mathbf{c}^{''} - \lfloor \frac{q}{2} \rfloor \mid < \frac{q}{4}$, return $M = 1$, otherwise return 0.

RABE.RevListUpd$(\mathcal{G}, t, RL_j)$.On input an attribute set $\mathcal{G}$, a time t and a revocation list $RL_j, j \in N$, the algorithm adds attribute set $\mathcal{G}$ and time t of all nodes associated with attribute i to the revocation list RL_j, and returns the revocation list $\widetilde{RL_j}$.

4.1 Correctness and Parameters

When the user's attributes satisfy the threshold access control policy W, that is, $\mid \mathcal{G} \cap W \mid \geq k$, we have $\mid \mathcal{G}^{'} \cap W^{'} \mid \geq l+1$. Choose a set of attributes with $l+1$ legal attributes. For each attribute i, we have:

$$\begin{aligned} \mathbf{c}_i^{'} &= \mathbf{e}_{i,1}^T \begin{bmatrix} \mathbf{c}_{i,0} \\ \mathbf{c}_{i,1} \end{bmatrix} + \mathbf{e}_{i,2}^T \begin{bmatrix} \mathbf{c}_{i,0} \\ \mathbf{c}_{i,2} \end{bmatrix} \\ &= \mathbf{e}_{i,1}^T \begin{bmatrix} \mathbf{A}_0^T \mathbf{s} + D\mathbf{y} \\ (\mathrm{H}(\mathbf{a}_i)\mathbf{C_1} + \mathbf{B}_1)^T \mathbf{s} + D\mathbf{r}_{i,1} \end{bmatrix} \\ &\quad + \mathbf{e}_{i,2}^T \begin{bmatrix} \mathbf{A}_0^T \mathbf{s} + D\mathbf{y} \\ (\mathrm{H}(\mathbf{t})\mathbf{C_2} + \mathbf{B}_2)^T \mathbf{s} + D\mathbf{r}_{i,2} \end{bmatrix} \\ &= \mathbf{e}_{i,1}^T \begin{bmatrix} \mathbf{A}_0^T \\ (\mathrm{H}(\mathbf{a}_i)\mathbf{C_1} + \mathbf{B}_1)^T \end{bmatrix} \mathbf{s} \\ &\quad + \mathbf{e}_{i,2}^T \begin{bmatrix} \mathbf{A}_0^T \\ (\mathrm{H}(\mathbf{t})\mathbf{C_2} + \mathbf{B}_2)^T \end{bmatrix} \mathbf{s} \\ &\quad + D(\mathbf{e}_{i,1}^T \begin{bmatrix} \mathbf{y} \\ \mathbf{r}_{i,1} \end{bmatrix} + \mathbf{e}_{i,2}^T \begin{bmatrix} \mathbf{y} \\ \mathbf{r}_{i,2} \end{bmatrix}) \\ &= \mathbf{e}_{i,1}^T \mathbf{F}_i^T \mathbf{s} + \mathbf{e}_{i,2}^T \mathbf{F}_t^T \mathbf{s} + D(\mathbf{e}_{i,1}^T \begin{bmatrix} \mathbf{y} \\ \mathbf{r}_{i,1} \end{bmatrix} + \mathbf{e}_{i,2}^T \begin{bmatrix} \mathbf{y} \\ \mathbf{r}_{i,2} \end{bmatrix}) \\ &= (\mathbf{F}_i e_{i,1} + \mathbf{F}_t e_{i,2})^T \mathbf{s} + D(\mathbf{e}_{i,1}^T \begin{bmatrix} \mathbf{y} \\ \mathbf{r}_{i,1} \end{bmatrix} + \mathbf{e}_{i,2}^T \begin{bmatrix} \mathbf{y} \\ \mathbf{r}_{i,2} \end{bmatrix}) \\ &= \hat{\mathbf{u}}_i^T \mathbf{s} + D(\mathbf{e}_{i,1}^T \begin{bmatrix} \mathbf{y} \\ \mathbf{r}_{i,1} \end{bmatrix} + \mathbf{e}_{i,2}^T \begin{bmatrix} \mathbf{y} \\ \mathbf{r}_{i,2} \end{bmatrix}) \in \mathbb{Z}_q^n. \end{aligned}$$

According to the Lagrangian interpolation formula in Shamir secret sharing scheme $\mathbf{u} = \Sigma_{i \in P} L_i \cdot \hat{\mathbf{u}}_i$ to recover

$$\begin{aligned} \mathbf{c}^{''} &= \sum_{i \in P} L_i \mathbf{c}_i^{'} \\ &= \sum_{i \in P} L_i(\hat{\mathbf{u}}_i^T \mathbf{s} + D(\mathbf{e}_{i,1}^T \begin{bmatrix} \mathbf{y} \\ \mathbf{r}_{i,1} \end{bmatrix} + \mathbf{e}_{i,2}^T \begin{bmatrix} \mathbf{y} \\ \mathbf{r}_{i,2} \end{bmatrix})) \\ &= \mathbf{u}^T \mathbf{s} + \sum_{i \in P} DL_i(\mathbf{e}_{i,1}^T \begin{bmatrix} \mathbf{y} \\ \mathbf{r}_{i,1} \end{bmatrix} + \mathbf{e}_{i,2}^T \begin{bmatrix} \mathbf{y} \\ \mathbf{r}_{i,2} \end{bmatrix}) \in \mathbb{Z}_q^n \end{aligned}$$

then

$$\begin{aligned} w &= c_0 - \mathbf{c}^{''} \\ &= \mathbf{u}^T \mathbf{s} + Dx + M \lfloor \frac{q}{2} \rfloor \\ &\quad - \mathbf{u}^T \mathbf{s} - \sum_{i \in J} DL_i(\mathbf{e}_{i,1}^T \begin{bmatrix} \mathbf{y} \\ \mathbf{r}_{i,1} \end{bmatrix} + \mathbf{e}_{i,2}^T \begin{bmatrix} \mathbf{y} \\ \mathbf{r}_{i,2} \end{bmatrix}) \\ &= M \lfloor \frac{q}{2} \rfloor + Dx - \sum_{i \in J} DL_i(\mathbf{e}_{i,1}^T \begin{bmatrix} \mathbf{y} \\ \mathbf{r}_{i,1} \end{bmatrix} + \mathbf{e}_{i,2}^T \begin{bmatrix} \mathbf{y} \\ \mathbf{r}_{i,2} \end{bmatrix}) \\ &= M \lfloor \frac{q}{2} \rfloor + error \in \mathbb{Z}_q. \end{aligned}$$

As in [29], we need to set the parameters m, q, α, σ to ensure that the error term < q/5:

$$\begin{aligned} &q = \alpha m(\|R\| + 1)(f+1)((f+l)!)^4 \cdot \omega(\sqrt{2m}) \\ &m = 6n^{1+\delta}, \qquad \sigma = m \cdot \omega(\sqrt{\log 2n}) \\ &\alpha = (\sigma\sqrt{m}(\|R\| + 1)(f+1)((f+l)!)^4 \cdot \omega\sqrt{m})^{-1} \end{aligned}$$

and round up q to the nearest larger prime number, and m to the nearest larger integer. Here we assume that δ is such that $n^{1+\delta} > \lceil (n+1)\log q + \omega(\log n) \rceil$.

4.2 Security Analysis

Under the LWE assumption in the standard model, we prove that our construction is secure, and the specific process is as follows.

Theorem 2. *If there is a PPT adversary $\mathcal{A}$ with advantage $\epsilon > 0$ against the selective security game for the RABE scheme described above, then there is a PPT algorithm $\mathcal{B}$, which decides the LWE problem with advantage $\epsilon/2$.*

Proof. Suppose that the adversary $\mathcal{A}$ has a probability polynomial time algorithm that can selectively attack the scheme, the adversary breaks through the above scheme with advantage ε, then we construct an algorithm $\mathcal{B}$ that can distinguish the decision $(\mathbb{Z}_q, n, \chi) - LWE$ problem

Table 2: Efficiency comparison

	Chen's scheme [8]	Zhang's scheme [29]	Our scheme
Private Key Size	$O(\log N)\cdot\widetilde{O}(n^{\varepsilon+1})$	$\widetilde{O}(n)$	$O(\log^2 M)\cdot\widetilde{O}(n^{\eta+1})$
Key Update Size	$r_u \log\frac{N}{r_u}\cdot\widetilde{O}(n^{\delta+1})$	–	$r_a \log\frac{M}{r_a}\cdot\widetilde{O}(n^{\delta+1})$
Public Key Size	$\widetilde{O}(n^{\varepsilon+2})$	$\widetilde{O}(n^{\delta+2})$	$\widetilde{O}(n^{\delta+\eta+2})$
Cipertext Size	$\widetilde{O}(n^{\varepsilon+1})$	$\widetilde{O}(n^{\delta+1})$	$\widetilde{O}(n^{\delta+\eta+1})$

with advantage ε. Recall *Definition* 6 provides an instance of the LWE problem as a sample oracle $\mathcal{O}$, for some secret key $s\in\mathbb{Z}_q^n$, which can be either truly random $\mathcal{O}_\$$ or noisy pseudorandom $\mathcal{O}_s$. The simulator $\mathcal{B}$ uses the adversary $\mathcal{A}$ to distinguish between the two, and proceeds as follows:

Instance. $\mathcal{B}$ requests from $\mathcal{O}$ and receives a fresh pair $(\mathbf{u}_i, v_i)\in\mathbb{Z}_q^n\times\mathbb{Z}_q$, for each $i=0,\ldots,m$.

Init. The adversary $\mathcal{A}$ announces to $\mathcal{B}$ the challenge access structure (W^*, k^*) and a revocation list RL_j on time period t^*.

Setup. The simulator $\mathcal{B}$ sets the public parameters as follows:

1) Let an attribute set $\mathcal{Q}=\{1,2,\ldots,f\}$, a default attribute set $\mathcal{M}=\{f+1,f+2,\ldots,f+l\}$ and $\mathcal{Q}'=\mathcal{Q}\cup\mathcal{M}$.
2) The adversary submits an access control policy $(W^*,\ k^*)$ to $\mathcal{B}$, where $1\le k^*\le min(|W^*|,l)$. Let $W'=W^*\cup\{f+1,f+2,\ldots,f+l+1-k^*\}$.
3) After the simulator $\mathcal{B}$ receives $(W^*,\ k^*)$, the $f+l$ uniformly random matrixs $\mathbf{a}_i^*$ are chosen. Using $\mathbf{TrapdoorGen}(q,n)$ to generate $(\mathbf{C}_1,\ \mathbf{T}_{\mathbf{C}_1})$, $(\mathbf{C}_2,\ \mathbf{T}_{\mathbf{C}_2})$, For $i\in W'$, the simulator $\mathcal{B}$ randomly choose $\mathbf{R}_{i,1}^*,\ \mathbf{R}_{i,2}^*\in\{-1,1\}^{m\times m}$, calculate $\mathbf{B}_1=\mathbf{A}_0\mathbf{R}_{i,1}^*-\mathrm{H}(\mathbf{a}_i^*)\mathbf{C}_1$, $\mathbf{B}_2=\mathbf{A}_0\mathbf{R}_{i,2}^*-\mathrm{H}(\mathbf{t}^*)\mathbf{C}_2$ and give the public parameters $pp=(\mathbf{A_0},\mathbf{B_1},\ \mathbf{B_2},\mathbf{C_1},\ \mathbf{C_2},\{\mathbf{a}_i^*\}_{i\in\mathcal{Q}'},\mathbf{u},\mathrm{H})$ to $\mathcal{A}$.

Phase 1. The simulator $\mathcal{B}$ can use the trapdoor $\mathbf{T}_{\mathbf{C}_1}$,$\mathbf{T}_{\mathbf{C}_2}$ to respond to private key queries:

1) When the adversary's query attributes $\mathcal{G}\in\mathcal{Q}$ satisfies the access control policy (W^*,k^*), $\mathcal{B}$ returns $\perp$.
2) When the adversary's query attributes $\mathcal{G}\in\mathcal{Q}$ doesn't satisfy the access control policy (W^*,k^*), i.e., $|\mathcal{G}\cap W^*|\le k^*-1$, let $\mathcal{G}'=\mathcal{G}\cup\{f+1,f+2,\ldots,f+l\}$. $|\mathcal{G}'\cap W'|\le d$ because of $W'=W\cup\{f+1,f+2,\ldots f+l+1-k\}$. Choose a subset $\hat{\mathcal{G}}$, satisfy $(\mathcal{G}'\cap W')\subseteq\hat{\mathcal{G}}\subset\mathcal{G}'$, $\mid\hat{\mathcal{G}}\mid=l$.
3) For $i\in\hat{\mathcal{G}}$, define $\mathbf{F}_i=(\mathbf{A}_0\mid\mathrm{H}(\mathbf{a}_i)\mathbf{C_1}+\mathbf{B}_1)$, $\mathbf{F}_t=(\mathbf{A}_0\mid\mathrm{H}(\mathbf{t})\mathbf{C_2}+\mathbf{B}_2)$. Choose $\mathbf{e}_{i,1},\mathbf{e}_{i,2}\leftarrow\mathcal{D}_{\mathbb{Z}_q^{2m},\sigma}$, Calculate $\hat{\mathbf{u}}_{i,1}=\mathbf{F}_i\mathbf{e}_{i,1},\hat{\mathbf{u}}_{i,2}=\mathbf{F}_t\mathbf{e}_{i,2}$, $\hat{\mathbf{u}}_i=\hat{\mathbf{u}}_{i,1}+\hat{\mathbf{u}}_{i,2}$.
4) Choose n polynomials of degree d, that is, $p_1(x),\ldots,p_n(x)\in\mathbb{Z}_q[x]$ such that $\mathbf{u}=(p_1(0),\ldots,p_n(0))^T$. For every $i\in\hat{\mathcal{G}}$, $\hat{\mathbf{u}}_i=(p_1(x),\ldots,p_n(x))^T$, we can recover the polynomial $p_1(x),\ldots,p_n(x)\in\mathbb{Z}_q[x]$ by using the Lagrange interpolation formula.
5) If $i\in\mathcal{G}'/\hat{\mathcal{G}}$, that is $i\notin W'$, then

$$\begin{aligned}\mathbf{F}_i &= (\mathbf{A}_0\mid\mathrm{H}(\mathbf{a}_i)\mathbf{C_1}+\mathbf{B}_1)\\ &= (\mathbf{A}_0\mid\mathbf{A}_0\mathbf{R}_{i,1}^*+(\mathrm{H}(\mathbf{a}_i)-\mathrm{H}(\mathbf{a}_i^*))\mathbf{C_1}),\\ \mathbf{F}_t &= (\mathbf{A}_0\mid\mathrm{H}(\mathbf{t})\mathbf{C_2}+\mathbf{B}_2)\\ &= (\mathbf{A}_0\mid\mathbf{A}_0\mathbf{R}_{i,2}^*+(\mathrm{H}(\mathbf{t})-\mathrm{H}(\mathbf{t}^*))\mathbf{C_2}).\end{aligned}$$

There is the FRD's definition in Section 3.6, $(\mathrm{H}(\mathbf{a}_i)-\mathrm{H}(\mathbf{a}_i^*))$ and $(\mathrm{H}(\mathbf{t})-\mathrm{H}(\mathbf{t}^*))$ are all full rank matrix. Therefore, $\mathbf{T}_{\mathbf{C}_1}$ and $\mathbf{T}_{\mathbf{C}_2}$ are also trapdoors for $\Lambda_q^\perp(\mathbf{C'}_1)$ and $\Lambda_q^\perp(\mathbf{C'}_2)$ respectively, where $\mathbf{C'}_1=(\mathrm{H}(\mathbf{a}_i)-\mathrm{H}(\mathbf{a}_i^*))\mathbf{C}_1$ and $\mathbf{C'}_2=(\mathrm{H}(\mathbf{t})-\mathrm{H}(\mathbf{t}^*))\mathbf{C}_2$. Run the SampleRight algorithm:

$$\begin{aligned}\mathbf{e}_{i,1} &\leftarrow \mathrm{SampleRight}(\mathbf{A}_0,\mathbf{C'}_1,\mathbf{R}_{i,1}^*,\mathbf{T}_{\mathbf{C}_1},\hat{\mathbf{u}}_{i,1},\sigma),\\ \mathbf{e}_{i,2} &\leftarrow \mathrm{SampleRight}(\mathbf{A}_0,\mathbf{C'}_2,\mathbf{R}_{i,2}^*,\mathbf{T}_{\mathbf{C}_2},\hat{\mathbf{u}}_{i,2},\sigma),\end{aligned}$$

return private key $(\mathbf{e}_{i,1},\mathbf{e}_{i,2})$.

Challenge. The adversary sends two message bits $M_0,M_1\in\{0,1\}$ and an access structure W^* to the simulator $\mathcal{B}$ and $\mathcal{B}$ randomly choose $b\in\{0,1\}$, calculate $c_0=Dv_0+M_b\lfloor q/2\rfloor\in\mathbb{Z}_q$, $\mathbf{v}_i=(v_1,v_2,\ldots,v_m)^T\in\mathbb{Z}_q^m$. For $i\in W'$, calculate $\mathbf{c}_{i,1}=D(\mathbf{R}_{i,1}^*)^T\mathbf{v}_i$, $\mathbf{c}_{i,2}=D(\mathbf{R}_{i,2}^*)^T\mathbf{v}_i$. Return challenge ciphertext

$$c^*=(c_0,\{\mathbf{c}_{i,1}\}_{i\in W'},\{\mathbf{c}_{i,2}\}_{i\in W'},W^*,\mathbf{t}^*).$$

Phase 2. Similar as Phase 1, the adversary $\mathcal{A}$ continues to initiate a request to $\mathcal{B}$.

Guess. The adversary $\mathcal{A}$ outputs a guess b'. The simulator $\mathcal{B}$ uses the guess to determine an answer on the LWE oracle: Output yes if b' = b, else output no.

For each $i \in W'$, we have

$$\begin{aligned}
\mathbf{c}_{i,1} &= D(\mathbf{R}^*_{i,1})^T\mathbf{v}_i \\
&= D(\mathbf{R}^*_{i,1})^T(\mathbf{A}_0^T\mathbf{s} + \mathbf{y}) \\
&= (\mathbf{A}_0\mathbf{R}^*_{i,1})^T(D\mathbf{s}) + D(\mathbf{R}^*_{i,1})^T\mathbf{y} \\
&= (\mathrm{H}(\mathbf{a}^*_i)\mathbf{C}_1 + \mathbf{B}_1)^T(D\mathbf{s}) + D(\mathbf{R}^*_{i,1})^T\mathbf{y} \\
\mathbf{c}_{i,2} &= D(\mathbf{R}^*_{i,2})^T\mathbf{v}_i \\
&= D(\mathbf{R}^*_{i,2})^T(\mathbf{A}_0^T\mathbf{s} + \mathbf{y}) \\
&= (\mathbf{A}_0\mathbf{R}^*_{i,2})^T(D\mathbf{s}) + D(\mathbf{R}^*_{i,2})^T\mathbf{y} \\
&= (\mathrm{H}(\mathbf{t}^*)\mathbf{C}_2 + \mathbf{B}_2)^T(D\mathbf{s}) + D(\mathbf{R}^*_{i,2})^T\mathbf{y}
\end{aligned}$$

Because the adversary could not obtain the $\{\mathbf{R}^*_{i,1}, \mathbf{R}^*_{i,2}\}_{i\in\mathcal{Q}'}$ from the public key, the adversary cannot distinguish actual ciphertext distribution from O_s or $O_\$$. If the adversary can have a non-negligible probability to guess the value of b, then there is an algorithm to solve the **LWE** problem. □

5 Performance Evaluation

We give an efficiency comparison with other schemes in **Table** 2. Here, $\mathcal{M}$ is the number of attributes, N is the number of users, r_u denotes the number of revoked users, r_a denotes the number of revoked attributes, δ is a small constant such that $\delta < 1/2$, ε is a small constant and $n^\varepsilon > O(\log N)$, η is a small constant and $n^\eta > O(\log \mathcal{M})$. Compared with Chen's scheme that can only achieve one-to-one communication, our scheme can achieve one-to-many communication. Compared with Zhang's scheme, our scheme supports attribute revocation.

6 Conclusions

In this paper, we propose a ciphertext policy attribute-based encryption scheme from lattices with efficient attribute revocation, which resists quantum attacks. The scheme builds a binary tree structure to update the legitimate user's key, and obtains the attribute revocation. We prove that our construction is secure against selective-attribute attacks in the standard model and security can be reduced to hardness of learning with error assumption. Although our scheme achieves a flexible threshold access control, how to construct a more complex access structure (such as access tree structure, circuit structure, etc.) is the work that will be carried out in the next step. In addition, how to design a scheme against adaptive attacks is also our future work.

Acknowledgments

This work was supported by Department of Education of Zhejiang Province of China (No.Y201636547), higher education research of Hangzhou Dianzi University in 2017, Key Research Project of Zhejiang Province (No.2017C01062), Guangxi Key Laboratory of Cryptography and Information Security (No.GCIS201718), the Opening Project of Guangdong Provincial Key Laboratory of Information Security Technology(No.2017B030314131-05), the Fund of Lab of Security Insurance of Cyberspace, Sichuan Province (No.szjj2017-055), research on the innovation of training mode of the first level subject of Cyberspace Security (No.YB201767), and cyberspace subject of Hangzhou Dianzi University (No.GK168800225075).

References

[1] S. Agrawal, D. Boneh, and X. Boyen, "Lattice basis delegation in fixed dimension and shorter-ciphertext hierarchical IBE," in *Advances in Cryptology (CRYPTO'10), 30th Annual Cryptology Conference*, pp. 98–115, Aug. 2010.

[2] S. Agrawal, X. Boyen, V. Vaikuntanathan, P. Voulgaris, and H. Wee, *Functional encryption for threshold functions (or fuzzy IBE) from lattices.* Darmstadt, Germany: Springer Berlin Heidelberg, 2012.

[3] D. Apon, X. Fan, and F. H. Liu, "Deniable attribute based encryption for branching programs from LWE," in *Theory of Cryptography - 14th International Conference (TCC'16)*, pp. 299–329, 2016.

[4] R. Bendlin and I. Damgård, "Threshold decryption and zero-knowledge proofs for lattice-based cryptosystems," in *Theory of Cryptography, 7th Theory of Cryptography Conference (TCC'10)*, pp. 201–218, 2010.

[5] A. Boldyreva, V. Goyal, and V. Kumar, "Identity-based encryption with efficient revocation," in *Proceedings of the 2008 ACM Conference on Computer and Communications Security (CCS'08)*, pp. 417–426, 2008.

[6] X. Boyen, *Attribute-Based Functional Encryption on Lattices*, Tokyo, Japan: Springer Berlin Heidelberg, 2013.

[7] X. Boyen and Q. Y. Li, *Attribute-Based Encryption for Finite Automata from LWE*, Kanazawa, Japan: Springer International Publishing, 2015.

[8] J. Chen, H. W. Lim, S. Ling, H. X. Wang, and K. Nguyen, "Revocable identity-based encryption from lattices," in *17th Australasian Conference on Information Security and Privacy (ACISP'12)*, pp. 390–403, 2012.

[9] J. S. Chen, C. Y. Yang, and M. S. Hwang, "The capacity analysis in the secure cooperative communication system," *International Journal of Network Security*, vol. 19, no. 6, pp. 863–869, 2017.

[10] P. S. Chung, C. W. Liu, and M. S. Hwang, "A study of attribute-based proxy re-encryption scheme in cloud environments," *International Journal of Network Security*, vol. 16, no. 1, pp. 1–13, 2014.

[11] X. B. Fu, X. Y. Nie, T. Wu, and F. G. Li, "Large universe attribute based access control with efficient decryption in cloud storage system," *Journal of Systems and Software*, vol. 135, pp. 157–164, 2018.

[12] S. Gorbunov, V. Vaikuntanathan, and H. Wee, "Attribute-based encryption for circuits," in *Symposium on Theory of Computing Conference (STOC'13)*, pp. 545–554, 2013.

[13] S. Gorbunov and D. Vinayagamurthy, *Riding on Asymmetry: Efficient ABE for Branching Programs*, New Zealand: Springer Berlin Heidelberg, 2015.

[14] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in *Proceedings of the 13th ACM Conference on Computer and Communications Security (CCS'06)*, pp. 89–98, Oct. 2006.

[15] M. A. Hamza, J. F. Sun, X. Y. Nie, Z. Q. Qin, and H. Xiong, "Revocable abe with bounded ciphertext in cloud computing," *International Journal of Network Security*, vol. 19, no. 6, pp. 973–983, 2017.

[16] J. Hur and K. N. Dong, "Attribute-based access control with efficient revocation in data outsourcing systems," *IEEE Transactions on Parallel and Distributed Systems*, vol. 22, no. 7, pp. 1214–1221, 2011.

[17] V. Kuchta and O. Markowitch, "Multi-authority distributed attribute-based encryption with application to searchable encryption on lattices," in *Paradigms in Cryptology (Mycrypt'16). Malicious and Exploratory Cryptology - Second International Conference*, pp. 409–435, 2016.

[18] C. Mao L. Liu, Z. Cao, "A note on one outsourcing scheme for big data access control in cloud," *International Journal of Electronics and Information Engineering*, vol. 9, no. 1, pp. 29–35, 2018.

[19] C. C. Lee, P. S. Chung, and M. S. Hwang, "A survey on attribute-based encryption schemes of access control in cloud environments," *International Journal Network Security*, vol. 15, no. 4, pp. 231–240, 2013.

[20] M. Aref M. Bayat, "An attribute based key agreement protocol resilient to kci attack," *International Journal of Electronics and Information Engineering*, vol. 2, no. 1, pp. 10–20, 2015.

[21] H. Ma, T. Peng, and Z. H. Liu, "Directly revocable and verifiable key-policy attribute-based encryption for large universe," *International Journal of Network Security*, vol. 19, no. 2, pp. 272–284, 2017.

[22] D. Micciancio and C. Peikert, *Trapdoors for Lattices: Simpler, Tighter, Faster, Smaller*, Cambridge, UK: Springer Berlin Heidelberg, 2012.

[23] M. Pirretti, P. Traynor, P. D. McDaniel, and B. Waters, "Secure attribute-based systems," in *Proceedings of the 13th ACM Conference on Computer and Communications Security (CCS'06)*, pp. 99–112, 2006.

[24] A. Sahai, H. Seyalioglu, and B. Waters, "Dynamic credentials and ciphertext delegation for attribute-based encryption," *IACR Cryptology ePrint Archive*, vol. 2012, pp. 437, 2012.

[25] A. Sahai and B. Waters, "Fuzzy identity-based encryption," in *Advances in Cryptology (EUROCRYPT'05), 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 457–473, Denmark, May 2005.

[26] J. Singh, "Cyber-attacks in cloud computing: A case study," *International Journal of Electronics and Information Engineering*, vol. 1, no. 2, pp. 78–87, 2014.

[27] Y. H. Wang, Y. Q. Liu, and K. Wang, "A secure and efficient ciphertext encryption scheme based on attribute and support strategy dynamic update via hybrid encryption method," *International Journal of Network Security*, vol. 20, no. 5, pp. 907–913, 2018.

[28] N. I. Wu and M. S. Hwang, "Development of a data hiding scheme based on combination theory for lowering the visual noise in binary images," *Displays*, vol. 49, pp. 116–123, 2017.

[29] J. Zhang, Z. F. Zhang, and A. J. Ge, "Ciphertext policy attribute-based encryption from lattices," in *Symposium on Information, Compuer and Communications Security (ASIACCS'12)*, pp. 16–17, 2012.

[30] L. Y. Zhang and H. J. Yin, "Recipient anonymous ciphertext-policy attribute-based broadcast encryption," *International Journal of Network Security*, vol. 20, no. 1, pp. 168–176, 2018.

Biography

Kang Yang is currently pursuing his master's degreee in the computer science and technology, Hangzhou Dianzi University. His research interests include cloud computing security and cryptography.

Guohua Wu is a professor at Hangzhou Dianzi University, and he received the Ph.D. degree from Zhejiang University in 1998. His research interests include cryptography and information hiding.

Chengcheng Dong is currently pursuing his master's degreee in the computer science and technology, Hangzhou Dianzi University. His research interests include cloud computing security and cryptography.

Xingbing Fu is a lecturer, and he received the Ph.D. degree from University of Electronic Science and Technology of China (UESTC) in 2016. His research interests include cloud computing and cryptography.

Fagen Li is a professor, and he received the Ph.D. degree in Cryptography from Xidian University, Xi'an, P.R. China in 2007. His research interests include cryptography and network security.

Ting Wu received the Ph.D. degree in Shandong University in 2002. He is a Professor at Hangzhou Dianzi University. His research interests include cryptography and information security.