Efficient secret key reusing attribute-based encryption from lattices

Xinyuan Qian
School of Computer Science and Technology, University of Chinese Academy of Sciences, China
Wenyuan Wu
Chongqing Institute of Green and Intelligent Technology, Chinese Academy of Sciences, China

Abstract

Attribute-based encryption (ABE) schemes by lattices are likely to resist quantum attacks, and can be widely applied to many Internet of Thing or cloud scenarios. One of the most attractive feature for ABE is the ability of fine-grained access control which provides an effective way to ensure data security. In this work, we propose an efficient ciphertext policy attribute-based encryption scheme based on hardness assumption of LWE. Being different from other similar schemes, a user's secret key can only be generated once only and it can be used to decrypt ciphertext under different access policies by making combinations of secret key fragments. Specially, we propose a method for binding users' secret keys with their attributes and identities, which solves the collusion attack problem. The security of the scheme is proved to be selective secure under the LWE assumption.

1 Introduction

Public-key cryptography solves the problem of key distribution and management of symmetric-key system, and has become a main technique in secure communication. However, the cumbersome mechanism of traditional public-key system needs plenty of certificate exchanges, which is not suitable for newly thriveed applications, such as cloud and Internet of Thing.

Attribute-based encryption (ABE) is a cryptographic primitive which provides one-to-many encryption mechanism with fine-grained access control, eliminating certificates and producing a far simpler infrastructure. ABE are assorted into key-policy ABE (KP-ABE) and ciphertext-policy ABE (CP-ABE). These two variants of ABE, were extended by [15] and [5] proposed in 2006 and 2007, respectively. However, CP-ABE has gained much more attention than KP-ABE [37], because, in a CP-ABE scheme, the access policy can be specified by users rather than the authority, and this flexibility makes CP-ABE more practical to be used in real senarios.

CP-ABE schemes based on tradition assumptions have been well developed and can be divided into nine subcategories with regard to basic functionality [5, 12], revocation [4, 18], accountability [21, 25], policy hiding [19, 28], policy updating [23,26], multi-authority [9,10], hierarchy [20, 31], offline computation [17, 38], and outsourced computation [16, 29].

However, all the constructions listed above are vulnerable to quantum attacks. Lattice-based cryptography is proven to be secure under all the known quantum algorithms, but there are only a few ciphertext policy schemes that are constructed by lattices. In 2011, Agrawal introduced Shamir's secret-sharing scheme to lattices and construct a special case of KP-ABE supporting threshold access policies [3]. Inspired by Agrawal's work, some ciphertext-policy schemes from lattices are proposed using Shamir secret-sharing method [24, 36]. Also in 2011, Zhang et al. gave a lattice-based ciphertext-policy attribute-based encryption that supports and-gates on positive and negative attributes [35], but the policy are not flexible. Later on, based on Zhang's scheme, the work of [22, 34] make improvement in scaling parameters of [35] and add more functions to it. There are also some and-gates supported CP-ABE schemes [32,33], but the access control methods are different. In 2020, the first CP-ABE scheme from lattices supporting circuit was proposed in [7], but the large parameters restricts its application. In addtion, some schemes are from a more lightweight ideal lattice classes, Ring-Learning with Errors [1, 11, 13].

All the ciphertext policy schemes from lattices adopt the main techniques from the identity-based lattice encryption schemes of [2, 8, 14]. Their basic encryption structure, public-key dual cryptosystem, requires the use of trapdoor generation and preimage sampling function. On the one hand, the trapdoor in these funtions is proven to be large enough to ensure the security of lattice-based schemes, but this will enlarge the overall parameters, leading schemes unacceptably efficient and practical [37]. On the other hand, the secret key is sampled randomly from discrete gaussian distribution for a certain policy, so it cannot decode the ciphertext encrypted under another polices. Hence, it remains to find an efficient and practical construction method which have smaller parameters with the ability of secret key reusing.

Our contribution. In this paper, we construct a ciphertext policy attribute-based encryption scheme under the learning with error assumption, which support and-gates structure. Compared to all the previous ciphertext policy schemes from lattices, our scheme has a completely different construction, and thanks to this, the secret key can be reused for different policies, which is considered to be more practical than previous schemes. Our scheme is proven to be secure against chosen plaintext attack in the selective access structure model under the learning with error assumption.

The basic idea of our scheme is that each user in the system is defined by an "identity" matrix $\boldsymbol{F}_i$, and each attribute has a secret vector $\boldsymbol{g}_j$. All matrices and vectors are unique; therefore we can compute many special secret key fragments $(\boldsymbol{F}_i\boldsymbol{g}_j)$ as part of user's secret key. For an identical attribute, different users have different fragment value, so they cannot conspire together. This prevents collusion attacks in our scheme. In addition, a user can reuse their secret key by combining different fragments associated with the policy or access structure.

Moreover, our basic construction only encrypt the ciphertext under limited policies, but we later extend the access structure to support flexible access policies. In addition, we also show that the efficiency can be improved by introducing plaintext expansion method and compression algorithm. The efficiency and communication cost are compared within several similar and-gate supported ciphertext policy schemes from lattices, and the results show that our scheme not only has the advantage in the efficiencies for encryption and decryption, but also achieves shorter lattice dimension, public key, secret key, and lower ciphertext expansion rate. The new construction and smaller parameters make our scheme having both theoretical and practical merits.

2 Preliminaries

2.1 Notation

The set of intergers (real numbers) is denoted by $\mathbb{Z}$ ($\mathbb{R}$, resp.). The function $log\cdot$ denotes the natural logarithm. Let $\mathcal{D}$ denote a distribution over some finite set S. Then, we use $x \leftarrow \mathcal{D}$ to denote the fact that x is chosen from the distribution $\mathcal{D}$. $x \leftarrow U(S)$ is simply used to denote that x is chosen from the uniform distribution over S. We denote column vectors and matrices in bold, respectively by bold lowercase (e.g. $\mathbf{s}$) and uppercase (e.g. $\mathbf{A}$). We denote $\lfloor\cdot\rfloor$ as rounding down to the nearest integer. We denote l_2 and l_∞ norm by $||\cdot||$ and $||\cdot||_\infty$ respectively. For matrices $\mathbf{X} \in \mathbb{Z}_q^{n\times m}$ and $\mathbf{Y} \in \mathbb{Z}_q^{n\times \bar{m}}$, we denote $(\mathbf{X},\mathbf{Y}) \in \mathbb{Z}_q^{n\times(m+\bar{m})}$ as the concatenation of the columns of $\mathbf{X}$ followed by the columns of $\mathbf{Y}$. Similarly, for matrices $\mathbf{X} \in \mathbb{Z}_q^{n\times m}$ and $\mathbf{Y} \in \mathbb{Z}_q^{\bar{n}\times m}$, $(\mathbf{X};\mathbf{Y}) \in \mathbb{Z}_q^{(n+\bar{n})\times m}$ is denoted as the concatenation of the rows of $\mathbf{X}$ followed by the rows of $\mathbf{Y}$. We use $\mathcal{S} \vdash \mathcal{T}$ to denote an attribute set $\mathcal{S}$ satisfies an access structure $\mathcal{T}$. Let $|\mathcal{U}|$ denote the number of element in set $\mathcal{U}$.

The natural security parameter throughout the paper is n, and all other quantities are implicit functions of n. Let $poly(n)$ denote an unspecified function $f(n) = O(n^c)$ for some constant c. We use standard notation O to classify the growth of functions. We say a function $f(n)$ is negligible if for every $c > 0$, there exists a number N such that $f(n) < 1/n^c$ for all $n > N$. We use $negl(n)$ to denote a negligible function of n, and we say a probability is overwhelming if it is $1-negl(n)$.

2.2 Definition of The CP-ABE and Security Model

2.2.1 Definition

A ciphertext policy attribute-based encryption (CP-ABE) scheme normally consists of four algorithms $\mathcal{ABE} = \{Setup, KeyGen, Enc, Dec\}$. In this paper, we add *PKGen* algorithm to generated public key, rather than use *Setup* algorithm, and *SKGen* algorithm is equivalent to *KeyGen*.

(1)$Setup(\lambda,\mathcal{R}) \rightarrow msk$: Given a security parameter λ and an attributmske algorithm generates and returns a master secret key msk. The central authority can use msk to generate users' secret keys.

(2)$SKGen(msk,\mathcal{U}) \rightarrow sk_{\mathcal{U}}$: The algorithm takes as input the master key and msk and an use's attribute set $\mathcal{U} \subseteq \mathcal{R}$, and returns a secret key $sk_{\mathcal{U}}$.

(3)$PKGen(msk,\mathcal{T}) \rightarrow pk$: Taking as input the master secret key msk and an access structure $\mathcal{T}$, the algorithm returns a public key pk for $\mathcal{T}$.

(4)$Enc(pk,\mathcal{T},M) \rightarrow C$: Given a public key pk, an access structure $\mathcal{T}$ and a message M, it returns the ciphertext C.

(5)$Dec(C,sk_{\mathcal{U}}) \rightarrow res$: This algorithm takes a secret key $sk_{\mathcal{U}}$ and a ciphertext C as input, it first check whether the attribute set of $sk_{\mathcal{U}}$ satisfies the access structure $\mathcal{T}$ in C. The algorithm returns $\perp$ if not. Otherwise, it outputs the decryption result.

The correctness can be defined as follows.

Definition 1. *For any message $M \in \{0,1\}^*$, access structure $\mathcal{T}$ and attribute $\mathcal{U} \subseteq \mathcal{R}$ that $\mathcal{U} \vdash \mathcal{R}$, we require that $Dec(Enc(pk,\mathcal{T},M),sk_{\mathcal{U}}) = M$ with overwhelming probability.*

2.2.2 Security Model

The security model for CP-ABE should be against chosen-plaintext attack under the selective attribute model (sCPA), where the adversary chooses a challenge access structure and gives it to the challenger. Then the formal game model of our scheme is described below:

Init: The adversary $\mathcal{A}$ specifies a challenge access structure $\mathcal{T}^*$ and sends it to the challenger.

Setup: The challenger runs the *Setup* and *PKGen* algorithm, and sends pk to the adversary $\mathcal{A}$ and keeps the master secret key msk.
Phase 1: The adversary can make a number of private key queries on different attribute sets except for the attribute set $\mathcal{U}$ that satisfies $\mathcal{T}^*$.
Challenge: The adversary chooses two messages M_0,M_1 satisfing $|M_0| = |M_1|$ and gives it to challenger, who will later randomly choose one bit $b \in \{0,1\}$. It runs the *Enc* algorithm to compute $C^* = Enc(pk, \mathcal{T}^*, M_b)$, and sends C^* to adversary.
Phase 2: Similar as Phase 1.
Guess: $\mathcal{A}$ outputs a bit b'.

Definition 2. *A CP-ABE scheme is said to be secure against selective chosen plaintext attacks (sCPA) if any probably polynomial-time adversary $\mathcal{A}$ making some secret-key queries can win the given game with a negligible advantage* $Adv^{ind-scpa}_{\mathcal{ABE},\mathcal{A}}(\lambda) = |Pr[b = b'] - \frac{1}{2}|$.

2.3 Lattices

Let $H = \mathbb{R}^m$. A lattice is a discrete subgroup of H. For a basis $\mathbf{B} = \{\mathbf{b}_1, \mathbf{b}_2 \cdots, \mathbf{b}_n\} \in H^n$, we denote $\mathcal{L}(B)$ as the lattice generated by $\mathbf{B}$ through the following fomula:

$$\mathcal{L}(\mathbf{B}) = \{\sum_{i=1}^{n} x_i \mathbf{b}_i | x_i \in \mathbb{Z}\}$$

2.4 Learning with Errors (LWE)

The LWE problem proposed by Regev [30] can be reduced by a quantum algorithm to some standard lattices problems (i.e., SIVP) in the worst case. That is to say that even if we use the advanced quantum computing technology, it is still hard to solve LWE problem in regular time. We use the decisional version of the LWE problem (denoted by DLWE$_{n,q,\chi}$).

Definition 3. *[2] Denote by $\mathbb{T} = \mathbb{R}/\mathbb{Z}$ the group of reals $[0,1)$ with addition modulo 1. Denote by Ψ_α the distribution over $\mathbb{T}$ of a normal variable with mean 0 and standard deviation $\alpha/\sqrt{2\pi}$ then reduced modulo 1. Denote by $\lfloor x \rceil = \lfloor x + \frac{1}{2} \rfloor$ the nearest integer to the real $x \in \mathbb{R}$. Denote by $\bar{\Psi}_\alpha$ the discrete distribution over $\mathbb{Z}_q$ of the random variable $\lfloor qX \rceil \mod q$ where the random variable $X \in \mathbb{T}$ has distribution Ψ_α.*

According to complementary error function (erfc), a normal variable with variance σ^2 is within distance $t \cdot \sigma$ of its mean, with overwhelming probability at least $1 - 1/t \cdot \exp(-t^2/2)$.

Let $m = poly(n)$ and let $\mathbf{e} = (e_1, \cdots, e_m) \leftarrow \bar{\Psi}^m_\alpha$ denote variables drawn from distribution $\bar{\Psi}_\alpha$ over $\mathbb{Z}_q$ independently. We know that each $||e_i|| \le \alpha q \omega(\sqrt{\log m})$ holds with probability negligible to 1, so $||\mathbf{e}|| \le \alpha q \sqrt{m} \omega(\sqrt{\log m})$ holds with overwhelming probability.

Definition 4. $(DLWE_{n,q,\chi}.)$ *Given $\mathbf{A} \in \mathbb{Z}_q^{m \times n}$ a uniformly random matrix, and $\mathbf{b} = \mathbf{As} + \mathbf{e}$, where $\mathbf{s} \leftarrow U(\mathbb{Z}_q^n)$ and $\mathbf{e} \leftarrow \chi^m$, distinguish $(\mathbf{A}, \mathbf{b} = \mathbf{As} + \mathbf{e})$ from $(\mathbf{A}', \mathbf{b}')$ drawn from the uniform distribution over $\mathbb{Z}_q^{m \times n} \times \mathbb{Z}_q^m$.*

Regev gave a quantum reduction from the LWE problem for certain noise distribution χ, denoted $\bar{\Psi}_\alpha$, to the worst-case SIVP and GapSIVP. This reduction from LWE problem to the worst case of these two problems requires $q > 2\sqrt{n}/\alpha$. Also in that work, he proved that, LWE and its DLWE are polynomially equivalent, for prime $p = poly(n)$. Therefore, with these conditions , we can say that an adversary $\mathcal{A}$ cannot solve the DLWE$_{n,q,\chi}$ problem if his advantage DLWE$_{n,q,\chi}Adv[\mathcal{A}] = |Pr[\mathcal{A}^{(\mathbf{A},\mathbf{b})} = 1] - Pr[\mathcal{A}^{(\mathbf{A}',\mathbf{b}')} = 1]|$ is negligible for a random $\mathbf{s} \in \mathbb{Z}_q^n$.

2.5 Useful Facts

Lemma 1. *(leftover hash lemma.) Let $\lambda \in \mathbb{N}$, $n \in \mathbb{N}$, and $m \ge n \log q + \omega(\log n)$ where $q > 2$ is a prime. Let $\mathbf{A} \leftarrow U(\mathbb{Z}_q^{m \times n})$ be a uniform random matrix, and let $\mathbf{r} \leftarrow \{0,1\}^m$ and $\mathbf{y} \leftarrow U(\mathbb{Z}_q^n)$. Then, the distribution $(\mathbf{A}, \mathbf{r}^T\mathbf{A})$ is statistically close to the distribution $(\mathbf{A}, \mathbf{y})$.*

Similarly, let $\mathbf{r} \leftarrow \{0,1\}^m$, $\mathbf{t} \leftarrow U(\mathbb{Z}_q^m)$ and $z \leftarrow U(\mathbb{Z}_q)$. The distribution $(\mathbf{t}, \mathbf{r}^T\mathbf{t})$ is also statistically close to the distribution $(\mathbf{r}, z)$.

Definition 5. *([2].)Let $\mathbf{e}$ be a vector in $\mathbb{Z}^m$ and let $\mathbf{y} \leftarrow \bar{\Psi}^{\mathrm{m}}_\alpha$. Then the quantity $|\mathbf{e}^T\mathbf{y}|$ treated as an integer in $[0, q-1]$ satisfies*

$$|\mathbf{e}^T\mathbf{T}| \le ||\mathbf{e}|| q\alpha\omega(\sqrt{\log m}) + ||\mathbf{e}||\sqrt{m}/2$$

with all but negligible probability in m. In particularly, if $x \leftarrow \bar{\Psi}_\alpha$ is treated as an integer in $[0, q-1]$ then $|x| \le q\alpha\omega(\sqrt{\log m}) + 1/2$ with all but negligible probability in m.

3 A CP-ABE Scheme Based on LWE

We use and-gates as our access structure to guarantee the fine-grained access control. To illustrate our scheme clearly, we only show how one and-gate structure works in our scheme.

3.1 Bingding Method

The main technique for binding method is the identity matrix. This random matrix is not public, and can only be computed by the system. The result of multiplications of identity matrix and the secret values for attributes are also random. For one thing, this can bind the user's attributes and identity with their secret key. For another, it can cover the values of identity matrix and secret vector. With this binding method, we can "personalize" user's secret key, which prevent the collusion attacks. Next, we show how an identity matrix $\boldsymbol{F}_{id}$ is sampled.

We now define a function $IDSamp(\boldsymbol{A}, \boldsymbol{B})$ to be used in the next section. It takes two element $\boldsymbol{A} \in \mathbb{Z}_q^{l \times n}$ and $\boldsymbol{B} \in \mathbb{Z}_q^{l \times n}$, where $l < n$, and outputs a matrix $\boldsymbol{F}_{id} \in \mathbb{Z}_q^{n\times n}$, which satisfies $\boldsymbol{AF}_i = \boldsymbol{B}$. Thus the equation can be seen as:

$$(\boldsymbol{C}|\boldsymbol{D}) \cdot \left(\frac{\boldsymbol{X}}{\boldsymbol{Y}}\right) = \boldsymbol{B},$$

where $\boldsymbol{C} \in \mathbb{Z}_q^{l\times l}$, $\boldsymbol{D} \in \mathbb{Z}_q^{(n-l)\times l}$, $\boldsymbol{X} \in \mathbb{Z}_q^{l\times n}$ and $\boldsymbol{Y} \in \mathbb{Z}_q^{(n-l)\times n}$. We first choose $\boldsymbol{Y} \leftarrow U(\mathbb{Z}_q^{(n-l)\times n})$, and we can compute $\boldsymbol{X} = \boldsymbol{C}^{-1} \cdot (\boldsymbol{B} - \boldsymbol{DY})$. Finally, we get a matrix $\boldsymbol{F}_{id}$ by putting $\boldsymbol{X}$ and $\boldsymbol{Y}$ together.

We know that half of the identity matrix $\boldsymbol{F}_{id}$ is independently chosen from uniform distribution, and the other half is computed. So the real solution space for $\boldsymbol{F}_{id}$ is $q^{(n-l)\times n}$. When we chose $l = \frac{n}{2}$, the success probability for adversary to find the solution is $\frac{1}{q^{n/2\times n}}$ without knowing any information.

3.2 Our Scheme

In this subsection, we present our CP-ABE scheme based on LWE problem using and-gate structure. In our scheme, we denote the set of all attributes as $\mathcal{R}$. We represent $\mathcal{R}$ as $\{1, \cdots, |\mathcal{R}|\}$, without loss of generality, where $|\mathcal{R}|$ is the number of attributes in $\mathcal{R}$. One user's' attribute set is $\mathcal{U} \in \mathcal{R}$.

Our scheme includes five algorithms *Setup*, *PKGen*, *SKGen*, *Enc*, and *Dec*, and is defined in algorithm 1- 5 below, which is parameterized by lattice dimension m, modulus q, and σ that determines the error distribution χ. Usually, all these parameters are functions of security parameter n, and they will be instantiated later. All the additions here are performded in $\mathbb{Z}_q$.

Algorithm 1 $Setup(n,m,q,\mathcal{R}) \rightarrow (pp,msk)$:

```
$l = n/2$, $k = m/l$
$\boldsymbol{A}_0 \leftarrow U(\mathbb{Z}_q^{l\times n})$, $\boldsymbol{B}_1 \leftarrow U(\mathbb{Z}_q^{l\times n})$
$\boldsymbol{H} = (\boldsymbol{H}_1;\boldsymbol{H}_2;\cdots;\boldsymbol{H}_k) \leftarrow U(\mathbb{Z}_q^{kl\times l})$
compute $\boldsymbol{A} = (\boldsymbol{A}_1;\boldsymbol{A}_2;\cdots;\boldsymbol{A}_k) \in \mathbb{Z}_q^{m\times n}$, where $\boldsymbol{A}_i = \boldsymbol{H}_i\boldsymbol{A}_0 \in \mathbb{Z}_q^{l\times n}$
$\boldsymbol{D} = (\boldsymbol{D}_2;\boldsymbol{D}_3;\cdots;\boldsymbol{D}_k) \leftarrow U(\mathbb{Z}_q^{(k-1)n\times n})$
$\boldsymbol{P} \leftarrow IDSamp(\boldsymbol{A}_1,\boldsymbol{B}_1)$
compute $\boldsymbol{B}_i = \boldsymbol{A}_i\boldsymbol{PD}_i, i = 2,\cdots,k$, and get $\boldsymbol{B} = (\boldsymbol{B}_1;\boldsymbol{B}_2;\cdots;\boldsymbol{B}_k) \in \mathbb{Z}_q^{m\times n}$
for every $i \in \mathcal{R}$ do
    choose a secret vector $\boldsymbol{g}_i \leftarrow U(\mathbb{Z}_q^n)$
end for
we have $\mathcal{G} = \{\boldsymbol{g}_i\}_{i\in\mathcal{R}}$
compute $\boldsymbol{E}_i = \boldsymbol{B}_i\boldsymbol{D}_i^{-1}$, $i = 2,\cdots,k$, set $\boldsymbol{E}_1 = \boldsymbol{B}_1$, and finally get $\boldsymbol{E} = (\boldsymbol{E}_1;\boldsymbol{E}_2;\cdots;\boldsymbol{E}_k) \in \mathbb{Z}_q^{m\times n}$
return $(pp = (\boldsymbol{A},\boldsymbol{B}_1,\boldsymbol{E})$, $msk = \mathcal{G})$
```

Algorithm 2 $SKGen\ (pp,msk,\mathcal{U}) \rightarrow sk_{\mathcal{U}}$:

```
compute $\boldsymbol{F}_{id} \leftarrow IDSamp(\boldsymbol{A}_1,\boldsymbol{B}_1)$
compute $\boldsymbol{F}_{id}^{-1}$, and set $sk_{\mathcal{U}} = \{\boldsymbol{F}_{id}^{-1}\boldsymbol{g}_i\}_{i\in\mathcal{U}\subseteq\mathcal{R}}$
return $sk_{\mathcal{U}} = \{\boldsymbol{F}_{id}^{-1}\boldsymbol{g}_i\}_{i\in\mathcal{U}}$
```

Algorithm 3 $PKGen(pp,msk,\mathcal{T}) \rightarrow pk$:

```
for every $i \in \mathcal{T}$ do
    find its secret value $\boldsymbol{g}_i \in \mathcal{G}$ (represented as $\boldsymbol{g}_{i,\mathcal{T}}$ in the next step)
    construct the decrypting secret key for $\mathcal{T}$: $\boldsymbol{s} = \sum_{i=1}^{|\mathcal{T}|} \boldsymbol{g}_{i,\mathcal{T}}$
end for
randomly sample $\boldsymbol{e} \leftarrow \chi^n$, and compute $\boldsymbol{t} = \boldsymbol{As} + \boldsymbol{e}$
return $pk = (\boldsymbol{t},\mathcal{T},\boldsymbol{E})$
```

3.3 Correctness

Now we give proof of the correctness for our scheme.

$$\begin{aligned}
b &= c_1 - \boldsymbol{c}_0 \cdot sk^* \\
&= \boldsymbol{r}^T(\boldsymbol{As}+\boldsymbol{e}) - \boldsymbol{r}^T\boldsymbol{E} \cdot \sum_{i=1}^{|\mathcal{T}|} \boldsymbol{F}_{id}^{-1}\boldsymbol{g}_{i,\mathcal{T}} + M\lfloor q/2 \rfloor \\
&= \boldsymbol{r}^T(\boldsymbol{As}+\boldsymbol{e}) - (\sum_{i=2}^{k} \boldsymbol{r}_i^T\boldsymbol{B}_i\boldsymbol{D}_i^{-1} + \boldsymbol{r}_1^T\boldsymbol{B}_1) \cdot \sum_{i=1}^{|\mathcal{T}|} \boldsymbol{F}_{id}^{-1}\boldsymbol{g}_{i,\mathcal{T}} + M\lfloor q/2 \rfloor \\
&= \boldsymbol{r}^T\boldsymbol{As} + \boldsymbol{re} - \sum_{i=2}^{k} \boldsymbol{r}_i^T\boldsymbol{A}_i\boldsymbol{F}_{id}\boldsymbol{D}_i\boldsymbol{D}_i^{-1}\boldsymbol{F}_{id}^{-1} \sum_{i=1}^{|\mathcal{T}|} \boldsymbol{g}_{i,\mathcal{T}} - \boldsymbol{r}_1\boldsymbol{A}_1\boldsymbol{F}_{id}\boldsymbol{F}_{id}^{-1} \cdot \\
&\quad \sum_{i=1}^{|\mathcal{T}|} \boldsymbol{g}_{i,\mathcal{T}} + M\lfloor q/2 \rfloor \\
&= \boldsymbol{r}^T\boldsymbol{As} + \boldsymbol{re} - (\sum_{i=1}^{k} \boldsymbol{r}_i^T\boldsymbol{A}_i)\boldsymbol{s} + M\lfloor q/2 \rfloor \\
&\approx M\lfloor q/2 \rfloor (\bmod q)
\end{aligned}$$

It suffices to set the parameters so that with overwhelming probability,

$$|\boldsymbol{r}^T\boldsymbol{e}| < q/4 \tag{1}$$

3.4 Security Analysis

In this subsection, we prove the security of our CP-ABE scheme in the selective model in Definition 1.

Theorem 1. *For properly choosen n,m,q,α, let $\chi = \bar{\Psi}_\alpha$. Then if $LWE_{q,\chi}$ is hard, our CP-ABE scheme is secure against selective chosen plaintext attack (sCPA).*

Proof. In order to prove the sCPA security of the CP-ABE scheme described above, we use a sequence of games starting from the original sCPA game described in Definition 1. Suppose there exists a probabilistic polynomial time adversary $\mathcal{A}$, and an sCPA challenger $\mathcal{C}$. Let $\mathcal{A}$'s advantage in our scheme be $Adv_{\mathcal{CP}-\mathcal{ABE}}^{ind-scpa}[\mathcal{A}]$.

Algorithm 4 $Enc(pk, \mathcal{T}, M) \rightarrow C$:

```
randomly sample $\boldsymbol{r} = (\boldsymbol{r}_1; \cdots; \boldsymbol{r}_k) \leftarrow \{0,1\}^m$
compute $\boldsymbol{c}_0 = \boldsymbol{r}^T \boldsymbol{E} = \sum_{i=1}^{k} \boldsymbol{r}_i^T \boldsymbol{E}_i \in \mathbb{Z}_q^n$
compute $c_1 = \boldsymbol{r}^T \boldsymbol{t} + M\lfloor q/2 \rfloor$
return $C = (\boldsymbol{c}_0, c_1, \mathcal{T})$
```

Algorithm 5 $Dec(C, sk_{\mathcal{U}}) \rightarrow res$:

```
if $\mathcal{U}$ does not match $\mathcal{T}$ then
    output $\perp$
else
    construct decrypting secret key for $\mathcal{T}$:$sk^* = \sum_{i=1}^{|\mathcal{T}|} \boldsymbol{F}_{id}^{-1} \boldsymbol{g}_{1,\mathcal{T}} \in \mathbb{Z}_q^n$
    compute $b = c_1 - \boldsymbol{c}_0 \cdot sk^* \in \mathbb{Z}_q$, if $|b - \lfloor \frac{1}{2} \rfloor| \leq \lfloor \frac{1}{4} \rfloor \in \mathbb{Z}_q$,set $res = 1$, otherwise $res = 0$
end if
return $res$
```

To ensure that $\mathcal{A}$ breaks the selective chosen plaintext security of our CP-ABE scheme with negligible advantage, we show that $\mathcal{A}$ cannot distinguish between the games.

Game 0. This is the original sCPA game from Definition 1 between the attacker $\mathcal{A}$ and the challenger $\mathcal{C}$.

In the **Phase 1** of this game, let $\boldsymbol{F}_{(i)}$ and $\mathcal{U}_i$ denote the value of random matrix generated from *SKGen* algorithm and the user's attribute set asked in the i th query respectively, and let N denote the total number of queries. Therefore, $\mathcal{A}$ has a set of secret key $\mathcal{S} = \{\{\boldsymbol{F}_{(1)}^{-1}\boldsymbol{g}_i\}_{i\in\mathcal{U}_1}, \{\boldsymbol{F}_{(2)}^{-1}\boldsymbol{g}_i\}_{i\in\mathcal{U}_2}, \cdots, \{\boldsymbol{F}_{(N)}^{-1}\boldsymbol{g}_i\}_{i\in\mathcal{U}_N}\}$.

First, we prove the security of a secret key for a user. Because every asked attribute set $\mathcal{U}_i \in \mathcal{S}$ do not satisfy $\mathcal{T}$, the adversary $\mathcal{A}$ cannot constructe the decrypting secret key sk^* for $\mathcal{T}$ using any user's secret key $\{\boldsymbol{F}_{(i)}^{-1}\boldsymbol{g}_j\}_{j\in\mathcal{U}_i}$. In the i query, the identity matrix $\boldsymbol{F}_{(i)} = (\boldsymbol{X}_i, \boldsymbol{Y}_i) \in \mathbb{Z}_q^{n\times n}$, where the solution space for $\boldsymbol{F}_{(i)}$ is $q^{n/2\times n}$. The adversary $\mathcal{A}$ cannot enumerate the identity matrix for a certain user in polynomial time. In addition, any secret key fragment $\boldsymbol{F}_{(i)}^{-1}\boldsymbol{g}_i$ looks random and uniform, so it is also impossible for $\mathcal{A}$ to get $\boldsymbol{F}_{(i)}^{-1}$, which means that $\mathcal{A}$ cannot get more information to attack our scheme. In fact, the adversary $\mathcal{A}$ can also gauss the decrypting secret key to attack, but the successful probability is smaller than $\frac{1}{2}$; therefore, $\mathcal{A}$ would rather gauss the message bit and the successful probability is $\frac{1}{2}$.

On the other hand, since that all the $\mathcal{U}_i$ in $\mathcal{S}$ cannot satisfy the access structure $\mathcal{T}$, the secret $\boldsymbol{s}$ for $\mathcal{T}$ cannot be constructed by any single secret key $\{\boldsymbol{F}_{(j)}^{-1}\boldsymbol{g}_i\}_{i\in\mathcal{U}_j} \in \mathcal{S}$ ($|\mathcal{U}_i| < n$). In addition, the attack cannot succeed by scraping together several secret key fragments from different secret keys, because every secret key is personalized thanks to matrix $\boldsymbol{F}_{(i)}$. With the restriction $|\mathcal{U}_i| < n$, the adversary $\mathcal{A}$ cannot obtain more information about any $\boldsymbol{F}_{(i)}$ or $\boldsymbol{g}_i$.

What should be mentioned is that even if $\mathcal{A}$ get an identity matrix $\boldsymbol{F}_{(i)}$, still he cannot decrypt the ciphertext, because the secret key is binded with a certain user. Only $\mathcal{A}$ get the secret key of that user can he compute the values of *msk*.

Thus, the advantage of $\mathcal{A}$ in *Game 0* is the same as that in our scheme $Adv_{Game0}[\mathcal{A}] = Adv_{\mathcal{CP-ABE}}^{ind-scpa}[\mathcal{A}]$.

Game 1. Recall that the public key of the scheme is $pk = (\boldsymbol{t}, \mathcal{T}, \boldsymbol{E})$, with $\boldsymbol{t}$ generated by computing $\boldsymbol{As} + \boldsymbol{e}$, where $\boldsymbol{A} = \boldsymbol{HA}_0$ and $\boldsymbol{s} = \sum_{i=1}^{|\mathcal{T}|} \boldsymbol{g}_{i,\mathcal{T}}$. We know that $\boldsymbol{t} = \boldsymbol{HA}_0\boldsymbol{s} + \boldsymbol{e}$, where $\boldsymbol{H}$ is independently chosen from randomly uniform distribution, so every coefficient in $\boldsymbol{A}_0\boldsymbol{s}$ looks random and uniform. According to LWE assumption, the distribution of $\boldsymbol{t}$ also looks uniform.

Game 1 is identical to *Game 0* except that the $\boldsymbol{t}$ in public key is always chosen as a random independent element in $\mathbb{Z}_q^m$. Hence, in $\mathcal{A}$'s view, he cannot distinguish the distribution between $(\boldsymbol{H}, \boldsymbol{HA}_0\boldsymbol{s} + \boldsymbol{e})$ and $(\boldsymbol{H}, \boldsymbol{v})$, where $\boldsymbol{v} \leftarrow U(\mathbb{Z}_q^n)$, according to the lemma 1, so the distinguishing problem is as hard as LWE problem. The advantage for $\mathcal{A}$ to break D-LWE problem is represented as $\mathrm{DLWE}_{n,q,\chi}Adv[\mathcal{A}]$, which is considered to be negligible. Thus, the advantage difference of $\mathcal{A}$ between *Game 1* and *Game 0* is $|Adv_{Game1}[\mathcal{A}] - Adv_{Game0}[\mathcal{A}]| \leq \mathrm{DLWE}_{n,q,\chi}Adv[\mathcal{A}]$.

Game 2. In the last game, we change how the challenge ciphertext is built. Previously, we compute the ciphertext $(\boldsymbol{c}_0, c_1) = (\boldsymbol{r}^T\boldsymbol{E}, \boldsymbol{r}^T\boldsymbol{t} + M\lfloor q/2 \rfloor)$. The ciphertext now is choosen from uniform distribution $(\boldsymbol{c}_0', c_1') \leftarrow \mathbb{Z}_q^m \times \mathbb{Z}_q$ randomly and independently. Since the challenge ciphertext is always a fresh element in the ciphertext space, $\mathcal{A}$'s advantage in this game is zero ($Adv_{Game2}[\mathcal{A}] = 0$). It remains to show that *Game 1* and *Game 2* are computationally indistinguishable for $\mathcal{A}$, which we do by appling leftover hash lemma.

Note that the distribution $(\boldsymbol{t}, \boldsymbol{r}^T\boldsymbol{t})$ is statistically close to the uniform distribution by appling the lemma 1. In $\mathcal{A}$'s view, $c_1 = \boldsymbol{r}^T\boldsymbol{t} + M\lfloor q/2 \rfloor$ also looks uniform and he cannot distinguish between c_1 and c_1'. Let $\varepsilon_1(n)$ be a negligible probabilistic function in n for the adversary $\mathcal{A}$ to distinguish between c_1 and c_1'. Similarly, in $\mathcal{A}$'s view, he cannot distinguish between $\boldsymbol{c}_0$ and $\boldsymbol{c}_0'$, and the negligible probabilistic function is $\varepsilon_2(n)$. Therefore, the advantage difference of $\mathcal{A}$ between *Game 2* and *Game 1* is $|Adv_{Game2}[\mathcal{A}] - Adv_{Game1}[\mathcal{A}]| \leq \varepsilon_1(n) + \varepsilon_2(n)$.

Combining all the equalities and inequalities, we have

$$\begin{aligned}
&Adv_{\mathcal{CP-ABE}}^{ind-scpa}[\mathcal{A}] \\
&= |Adv_{Game2}[\mathcal{A}] - Adv_{Game1}[\mathcal{A}] + Adv_{Game1}[\mathcal{A}] - Adv_{Game0}[\mathcal{A}]| \\
&\leq |Adv_{Game2}[\mathcal{A}] - Adv_{Game1}[\mathcal{A}]| + |Adv_{Game1}[\mathcal{A}] - Adv_{Game0}[\mathcal{A}]| \\
&\leq \mathrm{DLWE}_{n,q,\chi}Adv[\mathcal{A}] + \varepsilon_1(n) + \varepsilon_2(n) \\
&= negl(n).
\end{aligned}$$

Thus, we have proven theorem 2.

3.5 Parameter Choices

In this subsection, we set the appropriate parameters to ensure that the our scheme's correctness and security. The parameters should be set to meet the following conditions with overwhelming probability.

- The hardness of LWE requires $\alpha q > 2\sqrt{n}$ (see Section 2.4).
- To cover the ciphertext $\boldsymbol{c}_0$ in *Dec* phase, the leftover hash lemma requires $m \geq n \log q + \omega(\log n)$ (see lemma 1).
- To ensure the correctness of decrypted result with overwhelming probability, we need $||\boldsymbol{r}^T\boldsymbol{e}|| < q/4$.

The basic parameters are $l = n/2$ and $k = m/l$. Since we have $||\boldsymbol{r}|| \leq \sqrt{m}$ (see Section 2.4), $||\boldsymbol{e}|| \leq q\alpha\omega(\sqrt{\log m}) + 1/2$ and $|\boldsymbol{r}^T\boldsymbol{e}| \leq ||\boldsymbol{r}||q\alpha\omega(\sqrt{\log m}) + ||\boldsymbol{r}||/2$ by lemma 2 , we can obtain the scheme's error $||\boldsymbol{r}^T\boldsymbol{e}|| \leq \sqrt{m}q\alpha\omega(\sqrt{\log m}) + \sqrt{m}/2 \leq q/4$.

Let δ be real such that $n^{1+\delta} > n \log q + \omega(\log n)$. To satisfy all the conditions above, m, q, α are determined as follows:

$$m = n^{1+\delta}$$

$$q = 2m\omega(\sqrt{\log m})$$

$$\alpha = (\sqrt{m}\omega(\sqrt{\log m}))^{-1}$$

4 Extension and Improvement

4.1 Access Control Extension

Our scheme supports and-gates as the access structure, and it can be extended to represent any policy using the same method in [32]. In fact, any policy represented as logical formula $\mathcal{T}$ can be converted into a disjunctive normal form (DNF), $\mathcal{T} = \mathcal{T}_1 \vee \cdots \vee \mathcal{T}_t$. The DNF is a disjunction of conjunctive clause, where every conjunctive clause $\mathcal{T}_i = (A_{i,1} \wedge \cdots \wedge A_{i,|\mathcal{T}_i|})$ can be represented as an and-gate access structure in our scheme. Therefore, we need to construct t secrets and public keys and encrypt the message t times accordingly.

However, the DNF represented by and-gates would significantly increase the number of access structures, due to the logical operations OR, and this may make the size of ciphertext unacceptably large. To address this problem, our access structure can be adjusted to access tree supporting AND and OR operations. We provide a diagram to illustrate the process of secret construction and secret re-building.

As depicted in Fig. 1, there are three clients, *Alice*, *Bob*, and *Carl*, whose "identities" are $\boldsymbol{F}_A$, $\boldsymbol{F}_B$, and $\boldsymbol{F}_C$, and four attributes, *Teacher*, *CS*, *Student*, and *CE*, which are associated with four secret vectors $\boldsymbol{g}_1$, $\boldsymbol{g}_2$, $\boldsymbol{g}_3$, $\boldsymbol{g}_4$. We specify a policy $\boldsymbol{P} = \mathit{Teacher} \vee (\mathit{CS} \wedge \mathit{Student})$, and the corresponding tree structure is shown in the figure, where each node is numbered. In the *PKGen* phase, we need the tree structure $\mathcal{T}$ to generate the secret vector of the root node from the bottom to top. The computing rules are that the value of OR node equals to each of its child nodes and the value of AND node equals to the sum of its child nodes' value and additional value. To be more specific, the system find the secret vectors for each leaf node's attribute to represent the node value, and compute as many node values as possible. For example, the leaf nodes 2, 4, and 5 have values $\boldsymbol{g}_1$, $\boldsymbol{g}_2$, and $\boldsymbol{g}_3$ respectively. The node 1 (root node) is OR, and the value of it should be the same as nodes 2 and 3. Hence, the secret $\boldsymbol{s}$ for the structure equals $\boldsymbol{g}_1$, and so does node 3. Finall, compute an additional value $\boldsymbol{r} = \boldsymbol{g}_1 - \boldsymbol{g}_2 - \boldsymbol{g}_3$ for node 3 and it will be used in the *Dec* phase.

Alice's secret key $\boldsymbol{F}_A^{-1}\boldsymbol{g}_1$ directly contains the secret $\boldsymbol{s} = \boldsymbol{g}_1$, so she has the access to decrypt the ciphertext. As for user Bob, he gets his secret key $\boldsymbol{sk}_{Bob} = \{\boldsymbol{F}_B^{-1}\boldsymbol{g}_2, \boldsymbol{F}_B^{-1}\boldsymbol{g}_3, \boldsymbol{F}_B^{-1}\boldsymbol{r}\}$ in the *SKGen* phase. He computes the value for node 3 first $\boldsymbol{F}_B^{-1}\boldsymbol{g}_1 = \boldsymbol{F}_B^{-1}\boldsymbol{r} + \boldsymbol{F}_B^{-1}\boldsymbol{g}_2 + \boldsymbol{F}_B^{-1}\boldsymbol{g}_1 = \boldsymbol{F}_B^{-1}(\boldsymbol{r} + \boldsymbol{g}_1 + \boldsymbol{g}_2)$. The root node is OR node, so the Bob's secret for decryption equals to the value for node 3; hence he also has the access for decryption. The analysis is the same for Carl, who does not have access to decrypt.

Both the above access control methods proposed above have pros and cons, and they should be applied according to the real situations. The first DNF-based method is suitable for scenarios that has many simple policies. A user's secret key need to be computed only one time and it can decrypt all conditionally-satisfied ciphertexts. It facilitates the management for secret keys, and the size of ciphertext will be acceptable thanks to the simple policy. However, the policy complexity in this method will lead to a exponential growth in the number of and-gates, which will further increase ciphertext size to a large extent, leading the scheme impractical. The second tree-based method can be used in scenarios where the policies can be extremely complex, because no matter how complex it can be, the message only need to be encrypted only once, which can save much computing and restorage resources.

4.2 Improvement

In this subsection, we introduce two methods to improve our scheme.

4.2.1 Plaintext Expansion Method.

This method is used to improve the efficiency of *Enc* and *Dec* algorithms by expanding the plaintext space, and also can reduce the ciphertext.

Note that in the *Enc* algorithm, $c_1 = \boldsymbol{r}^T\boldsymbol{t} + M\lfloor q/2 \rfloor$, where $M \in \{0, 1\}$. In fact, only if we ensure that $M \cdot X < q$, the *Dec* algorithm can output the correct result. We change the generation of ciphertext block by computing $c_1 = \boldsymbol{r}^T\boldsymbol{t} + M\lfloor q/2^d \rfloor$,

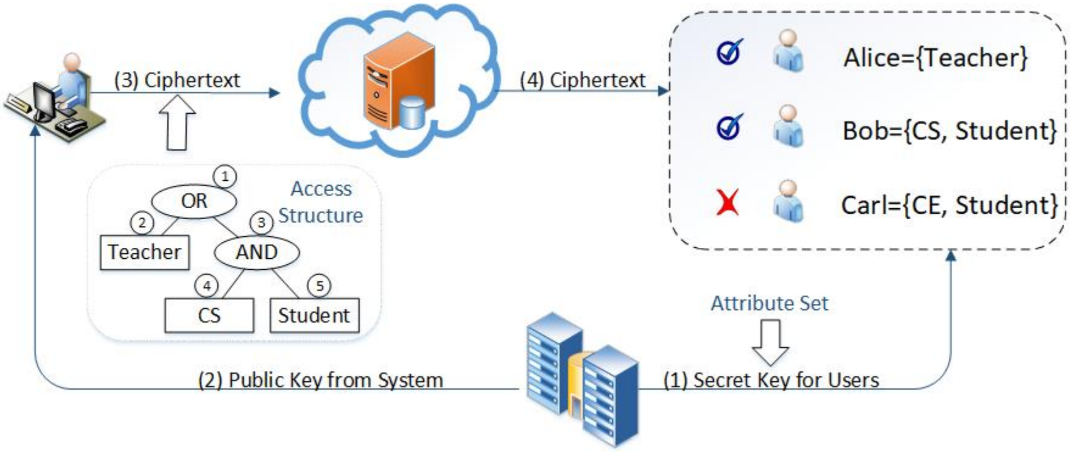

Figure 1: Access Structure

and the plaintext space is enlarged to $M \in \{0, 1, \cdots, 2^d - 1\}$. That is to say one c_1 can contain more message bits. Similarly, we should change the final step of Dec algorithm $res = \lfloor b / \lfloor \frac{q}{2^d} \rfloor \rceil$. In addition, the error in b should be less than $q/2^{d+1}$ for the correctness of decryption.

4.2.2 Compression and Decrompression.

Compression technology is introduced in Kyber scheme [6] to reduce the size of public key and ciphertext and communication costs to some extent. Two functions $Compress(x,q,d)$ and $Decompress(y,q,d)$ are defined as follows:

$Compress(x,q,d)$: Input $x \in \mathbb{Z}_q$, $d < \lceil \log q \rceil$ and output an integer $y = \lceil (2^d/q) \cdot x \rfloor \ mod^+ \ 2^d$.

$Deompress(y,q,d)$: Input $y = Compress(x,q,d)$ and output an integer $x' = \lceil (q/2^d) \cdot y \rfloor$.

The public key and ciphertext are compressed before transmitting, and they should be decompressed before encryption and decryption. When we compress and decompress an integer $x \in \mathbb{Z}_q$ in order, it might generate some errors $err = |x' - x| \ mod \ q \leq B_q = \lceil \frac{q}{2^{d+1}} \rfloor$, and it will further influence the scheme's parameters. These two functions can be used in a vector $\boldsymbol{x} \in \mathbb{Z}_q^k$, and the procedure is applied to each coefficient individually.

5 Performance

5.1 Efficiency Analysis

In this subsection, we compare the system efficiency and the efficiency of encryption and decryption of our scheme.

All the previous CP-ABE schemes from lattices are constructed using main technique in [2, 14, 27] to build lattice trapdoors, but the security of secret key in a attribute-based scheme relies on trapdoor quality, which is inefficiently large, leading to the growth of other parameters's size. The security of secret key in our scheme depends on the solution space, which is too large too gauss a certain value for the secret key or the secret selected by the system in advance, and compared to others, the secret key can be reused, which is practical; therefore, we argue that our scheme has advantages in generating the secret keys.

We choose six similar CP-ABE schemes using the same access structure, and-gates, to make comparisons. We represent the schemes in [22, 32–35] as ZJ11, W13-1, W13-2, ZX15, WZZ18 and Li19 respectively. These schemes' efficiencies of encryption and decryption algorithms are compared in the Table 1. In the table, the operation means a multiplition between two numbers, which is considered to dominate most of the time. The parameter k_0 is the constant coefficient for matrix inversion, and $m \geq n \log q$.

Table 1: Encryption and Decryption Efficiency Comparison.

Scheme	Operations for encryption per bit	Operations for decryption per bit
ZJ11	$(2\lvert\mathcal{R}\rvert - \lvert\mathcal{T}\rvert)mn + n$	$(\lvert\mathcal{R}\rvert + 1)m$
W13-1	$m^2 + 2mn + n$	$2m$
W13-2	$k_0 m^3 + mn + n$	m
ZX15	$(2\lvert\mathcal{R}\rvert + 1)mn + n$	$(\lvert\mathcal{R}\rvert + 1)m$
WZZ18	$2m^2 + 3mn + n$	$4m$
Li19	$(\lvert\mathcal{R}\rvert + 2)mn + n$	$\lvert\mathcal{R}\rvert m$
Ours	$(m+1)n$	n

From Table 1, we can see that our scheme is more efficient than the previous cryptosystems in terms of encryption and decryption efficiency.

5.2 Communication Cost Comparison

We also give comparisons on communication cost with the above schemes in Table 2. The parameter s is a large constant, and N is the maximum number of users.

Table 2: Comparing with Related Schemes.

Scheme	Lattice Dimension (m)	*pk* Size	*sk* Size	Ciphertext Expansion Rate
ZJ11	$\geq 6n \log q$	$(2\lvert\mathcal{R}\rvert+1)^2mn+n$	$m\lvert\mathcal{R}\rvert$	$2m\lvert\mathcal{R}\rvert+m-\lvert\mathcal{U}\rvert m$
W13-1	$\geq 6n \log q$	$3mn+n+sn$	$2m$	$2m+1$
W13-2	$\geq 6n \log q$	$mn+sm^2+n$	$2m$	$2m+1$
ZX15	$\geq 6n \log q$	$(2\lvert\mathcal{R}\rvert+1)^2mn+n$	$m\lvert\mathcal{R}\rvert$	$2m\lvert\mathcal{R}\rvert+m+1$
WZZ18	$\geq 5n \log q$	$(5m+c+1)\lvert\mathcal{R}\rvert n$	$4m\lvert\mathcal{R}\rvert$	$3m\lvert\mathcal{R}\rvert+1$
Li19	$\geq 6n \log q$	$4\lvert\mathcal{R}\rvert^2mn+n$	$m\lvert\mathcal{R}\rvert$	$2m\lvert\mathcal{R}\rvert+1$
Ours	$\geq n \log q$	$m+3mn$	$n\lvert\mathcal{U}\rvert$	$n+1$

From Table 2, we can see that our scheme achieves a better performance with with less lattice dimension, public key and secret key sizes, and ciphertext expansion rate.

We next compare several lattice-based CP-ABE schemes in terms of security and functionality. These schemes are [1, 7, 11, 13, 22, 32–36] and are represented as ZJ11, ZJ12, W13, FS15, ZX15, CZZ17, WZZ17, Li19, BV20 and Aff20. From the Table 3, we can see that our scheme is the only one that make secret keys reusable and be secure against the selective chosen plaintext attack under learning with error assumption.

6 Conclusion

We construct an efficient and secure ABE scheme by lattices supporting secret key reusing. The basic access structure in our scheme is and-gate and we extend it to support expressing any policies using two methods. In addition, we also introduce two methods to improve the efficiencies of encryption and decryption and reduce the communication cost. Under the $(\mathbb{Z}_q, n, \chi)$-LWE assumption, we have proven the security of our system in selective security model using game sequence method. How to construct an adaptively secure ABE and how to extend the functionality, such as attribute revocation, needs further study.

References

[1] Eric Affum, Xiasong Zhang, Xiaofen Wang, and John Bosco Ansuura. Efficient lattice cp-abe ac scheme supporting reduced-obdd structure for ccn/ndn. *Symmetry*, 12(1):166, 2020.

[2] Shweta Agrawal, Dan Boneh, and Xavier Boyen. Efficient lattice (h) ibe in the standard model. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 553–572. Springer, 2010.

[3] Shweta Agrawal, Xavier Boyen, Vinod Vaikuntanathan, Panagiotis Voulgaris, and Hoeteck Wee. Fuzzy identity based encryption from lattices. *IACR Cryptol. ePrint Arch.*, 2011:414, 2011.

[4] Nuttapong Attrapadung and Hideki Imai. Conjunctive broadcast and attribute-based encryption. In *International conference on pairing-based cryptography*, pages 248–265. Springer, 2009.

[5] John Bethencourt, Amit Sahai, and Brent Waters. Ciphertext-policy attribute-based encryption. In *2007 IEEE symposium on security and privacy (SP'07)*, pages 321–334. IEEE, 2007.

[6] Joppe Bos, Léo Ducas, Eike Kiltz, Tancrède Lepoint, Vadim Lyubashevsky, John M Schanck, Peter Schwabe, Gregor Seiler, and Damien Stehlé. Crystals-kyber: a CCA-secure module-lattice-based KEM. In *2018 IEEE European Symposium on Security and Privacy (EuroS&P)*, pages 353–367. IEEE, 2018.

[7] Zvika Brakerski and Vinod Vaikuntanathan. Lattice-inspired broadcast encryption and succinct ciphertext-policy ABE. *IACR Cryptol. ePrint Arch.*, 2020:191, 2020.

[8] David Cash, Dennis Hofheinz, Eike Kiltz, and Chris Peikert. Bonsai trees, or how to delegate a lattice basis. In *Annual international conference on the theory and applications of cryptographic techniques*, pages 523–552. Springer, 2010.

[9] Melissa Chase. Multi-authority attribute based encryption. In *Theory of cryptography conference*, pages 515–534. Springer, 2007.

[10] Melissa Chase and Sherman SM Chow. Improving privacy and security in multi-authority attribute-based encryption. In *Proceedings of the 16th ACM conference on Computer and communications security*, pages 121–130, 2009.

[11] Zehong Chen, Peng Zhang, Fangguo Zhang, and Jiwu Huang. Ciphertext policy attribute-based encryption supporting unbounded attribute space from r-lwe. *TIIS*, 11(4):2292–2309, 2017.

[12] Ling Cheung and Calvin Newport. Provably secure ciphertext policy ABE. In *Proceedings of the 14th ACM*

Table 3: Comparison of Security and Function.

Scheme	Assumption	Security	Access Control Structure	Operation	*sk* Reusing
ZJ11	LWE	sCPA	AND-gates	AND	No
ZJ12	LWE	sCPA	TS	THRESHOLD	No
W13	LWE	sCPA	AND-gates	AND	No
FS15	R-LWE	sCPA	LSSS	AND	No
ZX15	LWE	sCPA	AND-gates	AND	No
CZZ17	R-LWE	sCPA	TS	THRESHOLD	No
WZZ17	LWE	sCCA	AND-gates	AND,NOT	No
Li19	LWE	sCPA	AND-gates	AND	No
BV20	LWE	NG	Circuit	AND,NOT	No
Aff20	R-LWE	sCPA	TS	ALL	No
Ours	LWE	sCPA	AND-gates	AND	Yes

conference on Computer and communications security, pages 456–465, 2007.

[13] Tan Soo Fun and Azman Samsudin. Lattice ciphertext-policy attribute-based encryption from ring-lwe. In *2015 International Symposium on Technology Management and Emerging Technologies (ISTMET)*, pages 258–262. IEEE, 2015.

[14] Craig Gentry, Chris Peikert, and Vinod Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In *Proceedings of the fortieth annual ACM symposium on Theory of computing*, pages 197–206, 2008.

[15] Vipul Goyal, Omkant Pandey, Amit Sahai, and Brent Waters. Attribute-based encryption for fine-grained access control of encrypted data. In *Proceedings of the 13th ACM conference on Computer and communications security*, pages 89–98, 2006.

[16] Matthew Green, Susan Hohenberger, Brent Waters, et al. Outsourcing the decryption of abe ciphertexts. In *USENIX security symposium*, volume 2011, 2011.

[17] Susan Hohenberger and Brent Waters. Online/offline attribute-based encryption. In *International workshop on public key cryptography*, pages 293–310. Springer, 2014.

[18] Junbeom Hur and Dong Kun Noh. Attribute-based access control with efficient revocation in data outsourcing systems. *IEEE Transactions on Parallel and Distributed Systems*, 22(7):1214–1221, 2010.

[19] Junzuo Lai, Robert H Deng, and Yingjiu Li. Fully secure cipertext-policy hiding cp-abe. In *International conference on information security practice and experience*, pages 24–39. Springer, 2011.

[20] Jiguo Li, Qihong Yu, and Yichen Zhang. Hierarchical attribute based encryption with continuous leakage-resilience. *Information Sciences*, 484:113–134, 2019.

[21] Jin Li, Kui Ren, Bo Zhu, and Zhiguo Wan. Privacy-aware attribute-based encryption with user accountability. In *International Conference on Information Security*, pages 347–362. Springer, 2009.

[22] Juyan Li, Chunguang Ma, and Kejia Zhang. A novel lattice-based CP-ABPRE scheme for cloud sharing. *Symmetry*, 11(10):1262, 2019.

[23] Xiaohui Liang, Zhenfu Cao, Huang Lin, and Jun Shao. Attribute based proxy re-encryption with delegating capabilities. In *Proceedings of the 4th International Symposium on Information, Computer, and Communications Security*, pages 276–286, 2009.

[24] Yuan Liu, Licheng Wang, Lixiang Li, and Xixi Yan. Secure and efficient multi-authority attribute-based encryption scheme from lattices. *IEEE Access*, 7:3665–3674, 2018.

[25] Zhen Liu, Zhenfu Cao, and Duncan S Wong. White-box traceable ciphertext-policy attribute-based encryption supporting any monotone access structures. *IEEE Transactions on Information Forensics and Security*, 8(1):76–88, 2012.

[26] Song Luo, Jianbin Hu, and Zhong Chen. Ciphertext policy attribute-based proxy re-encryption. In *International Conference on Information and Communications Security*, pages 401–415. Springer, 2010.

[27] Daniele Micciancio and Chris Peikert. Trapdoors for lattices: Simpler, tighter, faster, smaller. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 700–718. Springer, 2012.

[28] Takashi Nishide, Kazuki Yoneyama, and Kazuo Ohta. Attribute-based encryption with partially hidden encryptor-specified access structures. In *International conference on applied cryptography and network security*, pages 111–129. Springer, 2008.

[29] Go Ohtake, Reihaneh Safavi-Naini, and Liang Feng Zhang. Outsourcing scheme of abe encryption secure against malicious adversary. *Computers & Security*, 86:437–452, 2019.

[30] Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. *Journal of the ACM (JACM)*, 56(6):1–40, 2009.

[31] Guojun Wang, Qin Liu, Jie Wu, and Minyi Guo. Hierarchical attribute-based encryption and scalable user revocation for sharing data in cloud servers. *computers & security*, 30(5):320–331, 2011.

[32] Shangping Wang, Xia Zhang, and Yaling Zhang. Efficient revocable and grantable attribute-based encryption from lattices with fine-grained access control. *IET Information Security*, 12(2):141–149, 2018.

[33] Yongtao Wang. Lattice ciphertext policy attribute-based encryption in the standard model. *IJ Network Security*, 16(6):444–451, 2014.

[34] Fugeng Zeng and Chunxiang Xu. A novel model for lattice-based authorized searchable encryption with special keyword. *Mathematical Problems in Engineering*, 2015, 2015.

[35] Jiang Zhang and Zhenfeng Zhang. A ciphertext policy attribute-based encryption scheme without pairings. In *International Conference on Information Security and Cryptology*, pages 324–340. Springer, 2011.

[36] Jiang Zhang, Zhenfeng Zhang, and Aijun Ge. Ciphertext policy attribute-based encryption from lattices. In *Proceedings of the 7th ACM Symposium on Information, Computer and Communications Security*, pages 16–17, 2012.

[37] Yinghui Zhang, Robert H Deng, Shengmin Xu, Jianfei Sun, Qi Li, and Dong Zheng. Attribute-based encryption for cloud computing access control: A survey. *ACM Computing Surveys (CSUR)*, 53(4):1–41, 2020.

[38] Yinghui Zhang, Axin Wu, and Dong Zheng. Efficient and privacy-aware attribute-based data sharing in mobile cloud computing. *Journal of Ambient Intelligence and Humanized Computing*, 9(4):1039–1048, 2018.