Copyright ⓒ2017 KSII

Ciphertext policy attribute-based encryption supporting unbounded attribute space from R-LWE

Zehong Chen[1,2], Peng Zhang[1,2] , Fangguo Zhang[3,4] and Jiwu Huang[1]
[1] College of Information Engineering, Shenzhen University, Shenzhen 518060, China
[e-mail: zhchen@szu.edu.cn, zhangp@szu.edu.cn, jwhuang@szu.edu.cn]
[2] ATR Key Laboratory of National Defense Technology, Shenzhen University, Shenzhen 518060, China
[3] School of Data and Computer Science, Sun Yat-sen University, Guangzhou 510006, China
[e-mail: isszhfg@mail.sysu.edu.cn]
[4] Guangdong Key Laboratory of Information Security, Guangzhou 510006, China
*Corresponding author: Peng Zhang

Received November 2, 2016; revised January 14, 2017; accepted February 14, 2017; published April 30, 2017

Abstract

Ciphertext policy attribute-based encryption (CP-ABE) is a useful cryptographic technology for guaranteeing data confidentiality but also fine-grained access control. Typically, CP-ABE can be divided into two classes: small universe with polynomial attribute space and large universe with unbounded attribute space. Since the learning with errors over rings (R-LWE) assumption has characteristics of simple algebraic structure and simple calculations, based on R-LWE, we propose a small universe CP-ABE scheme to improve the efficiency of the scheme proposed by Zhang *et al*. (AsiaCCS 2012). On this basis, to achieve unbounded attribute space and improve the expression of attribute, we propose a large universe CP-ABE scheme with the help of a full-rank differences function. In this scheme, all polynomials in the R-LWE can be used as values of an attribute, and these values do not need to be enumerated at the setup phase. Different trapdoors are used to generate secret keys in the key generation and the security proof. Both proposed schemes are selectively secure in the standard model under R-LWE. Comparison with other schemes demonstrates that our schemes are simpler and more efficient. R-LWE can obtain greater efficiency, and unbounded attribute space means more flexibility, so our research is suitable in practices.

Keywords: Ciphertext policy attribute-based encryption, R-LWE, unbounded, small universe, large universe

The work of this paper is supported by the National Natural Science Foundation of China (61402293), and the Science & Technology Innovation Projects of Shenzhen, China (ZDSYS20140430164957660, JCYJ20140418095735596, JCYJ2016 0307150216309, GJHZ20160226202520268).

https://doi.org/10.3837/tiis.2017.04.025 **ISSN : 1976-7277**

1. Introduction

Attribute-based encryption (ABE) has attracted much attention because it can achieve flexible one-to-many encryption, provide the ability to encrypt the data without knowing the specific information of receivers and realize fine-grained access control to encrypted data [1, 2]. In 2007, Bethencourt *et al*. [3] introduced a variant of ABE called ciphertext policy ABE (CP-ABE), which enables data owners to freely define access structures over attribute sets and encrypt the data under the structures. Since then, CP-ABE is widely used in many scenarios, such as access control for data stored on the cloud [4, 5] and secure online social networks [6, 7] and so on.

Over the past few years, many ABE/CP-ABE schemes have been proposed to achieve various functional purposes. Cheung and Newport [8] proposed a CP-ABE scheme in which access structures are AND-gates, and the scheme is proved to be secure under the decisional bilinear Diffie-Hellman (DBDH) assumption. Hur [9] presented a CP-ABE scheme to achieve immediate user revocation. An access control (CP-ABE) scheme is proposed by Zhang *et al*. [10] to realize both user revocability and attribute update, and the scheme is secure to the DBDH assumption. Liu *et al*. [11] presented a hierarchical ABE scheme from the learning with errors (LWE) assumption. Zhu *et al*. [12] constructed an ABE scheme from the learning with errors over rings (R-LWE) assumption. Fun and Samsudin [13] presented a CP-ABE scheme from R-LWE, which applied the linear secret sharing scheme to express an access structure.

However, once public parameters have been set in the setup phase, they are bounded in the whole encryption system for the above schemes. Especially, a data owner cannot set flexible and arbitrary access structures. Lewko *et al*. [14] solved this problem by introducing fresh local randomness at the phases of key generation and encryption, and divided ABE schemes into two classes: small universe and large universe. In a small universe ABE scheme, the size of the attribute space is polynomially bounded in the security parameter, and the size of the public parameters grows linearly with the number of attributes, such as [3, 8-11]. In a large universe ABE scheme, the attribute universe is exponentially large, and the public parameters do not impose additional restrictions on the functionality of the scheme, thus “unbounded” is achieved. Okamoto and Takashima [15] proposed first fully secure unbounded ABE scheme, and proved that the scheme is secure under the decisional linear assumption in the standard model. Rouselakis and Waters [16] described how to construct an unbounded CP-ABE scheme over prime order groups. Li *et al*. [17] constructed an unbounded multi-authority CP-ABE scheme with no needless restriction on the public parameters. Agrawal *et al*. [18] combined a small universe ABE scheme with a compatible standard model identity-based encryption scheme to construct a large universe ABE scheme from LWE. Zhang *et al*. [19] proposed a large universe CP-ABE scheme from LWE, and proved that the scheme is selectively secure against chosen plaintext attacks (CPA).

The above ABE schemes from LWE have the characteristic of simple algebraic structure, but these schemes are not efficient enough in practices because of an intrinsic quadratic computation overhead when using LWE. Thereby, Lyubashevsky *et al*. [20] introduced the LWE assumption over rings (R-LWE) whose distribution is pseudorandom. They also proved that the security of the R-LWE assumption can be reduced to the hardness of SVP in the worst case on ideal lattices. Cryptographic schemes based on R-LWE have many advantages, such as fast implementation and small public key size, ciphertext size and secret key size. As far as we know, there are two papers on ABE schemes based on R-LWE: one is proposed by Zhu *et al*. [12] and the other is proposed by Fun and Samsudin [13]. However, the scheme proposed by Zhu *et al*. [12] does not satisfy CPA security. Here is the reason: assume $C = PK \cdot r + pe + M$ is the ciphertext which contains the plaintext *M*, where *PK* and *p* are public. The adversary randomly chooses two equal length messages M_0 and M_1, and sends them to the challenger. The challenger picks $\vartheta \in \{0,1\}$ randomly and encrypts the message M_ϑ to construct a challenge ciphertext C^*, then sends C^* to the adversary. The adversary obtains C^* and computes $(C^* \bmod p) \bmod (PK \bmod p)$, then he can distinguish which message is encrypted. This explains that the scheme in [12] is not correct. Similar attack also exists in the CP-ABE scheme proposed by Fun and Samsudin [13].

1.1 Our contribution

Because the R-LWE assumption can use the fast Fourier transform (FFT) to compute the product of polynomials in the rings, encryption schemes based on R-LWE are more efficient than those based on LWE in the similar framework. In this paper, we first propose a small universe CP-ABE scheme from R-LWE for threshold access structure. We apply a Gaussian sampling algorithm over rings to generate a secret key whose size is reduced by nearly half compare with the first scheme in [19]. In the decryption phase, the Lagrange interpolation coefficients are used to reconstruct the secret embedded in the ciphertext. Making convenience of computation, we need to clear the denominators of the Lagrange interpolation coefficients. In order to do this, we take a sufficiently large constant which is associated with the number of system attributes multiplies the results derived from the sampling algorithm.

Then, we extend the proposed scheme to a large universe scheme supporting unbounded attribute space with the help of a full-rank differences (FRD) function, which means that the public parameters do not impose additional restriction on the values of attributes used for key generation and encryption. Two lattices called left lattice and right lattice are used in the key generation of the proposed scheme and its security proof, respectively. We apply the FRD function to map the coefficient vector of each attribute value to matrix, then combine a trapdoor of the left lattice to generate a secret key for a user in the large universe scheme such that the distribution of the secret key is statistically close to the discrete Gaussian distribution, and in the security proof, a trapdoor of the right lattice is used to generate a secret key for the adversary. In the encryption phase, we apply a low norm randomization matrix to ensure that attacks cannot distinguish between pseudorandom and true randomness. Both schemes are secure against CPA in the selective model. Moreover, compared with the schemes in [19], our small universe scheme has shorter public key, secret key and ciphertext sizes, and needs fewer operations for encryption per bit; our large universe scheme also has shorter public key size and needs fewer operations for encryption per bit, while the secret key and ciphertext sizes are equal to those in [19].

1.2. Organization

This paper is organised as follows. In Section 2, we introduce the preliminaries. In Section 3, we propose a small universe CP-ABE scheme from R-LWE and analyze its security. In Section 4, we propose a large universe CP-ABE scheme from R-LWE and also analyze its security. We compare our schemes with the existing CP-ABE schemes based on the LWE assumption in Section 5. Finally, we summarize the results of the paper in Section 6.

2. Preliminaries

2.1. Notations

Let bold lowercase and capital letters denote vectors and matrices, respectively. Let $\mathbb{Z}$ and $\mathbb{R}$ represent sets of integers and real numbers, and n be a power of 2. Let *poly*(n) represent an indeterminate function $g(n) = O(n^c)$ for a certain constant c. Let $q = 1 \bmod 2n$ represent a sufficiently large public prime modulus, $\mathbb{Z}_q$ denote a set of integers modulo q, $\mathbb{Z}[x]$ be a set of polynomials with integer coefficients. We take $f(x) = x^n + 1 \in \mathbb{Z}[x]$, which is irreducible over the rational field. Let $R = \mathbb{Z}[x]/<f(x)>$ be ring of integer polynomials modulo $f(x)$. Let $R_q = \mathbb{Z}_q[x]/<f(x)>$ be ring of integer polynomials modulo both $f(x)$ and q. Let $R_q^{\times}$ be a set of invertible polynomials in R_q. Unless stated otherwise, we let $\overline{\gamma}_\alpha \subset R_q$ represent the error distribution, which is defined in [21].

For two matrices $\boldsymbol{X} \in \mathbb{R}^{m_1 \times n}$ and $\boldsymbol{Y} \in \mathbb{R}^{m_2 \times n}$, $(\boldsymbol{X};\ \boldsymbol{Y}) \in \mathbb{R}^{(m_1+m_2)\times n}$ is the concatenation of the rows of $\boldsymbol{X}$ and $\boldsymbol{Y}$. Let $a = a_0 + a_1 x + \cdots + a_{n-1} x^{n-1} \in R_q$ and $\boldsymbol{x} = (x_1,\ x_2,\ \cdots,\ x_m)^{\mathrm{T}} \in R_q^m$. Define $rot_f(a) = (a,\ ax \bmod f,\ \cdots,\ ax^{n-1} \bmod f)^{\mathrm{T}} \in R_q^n$ and $Rot_f(\boldsymbol{x}) = (rot_f(x_1);\ rot_f(x_2);\ \cdots;\ rot_f(x_m)) \in R_q^{mn}$. Let $\boldsymbol{a} = (a_0,\ a_1,\ \cdots,\ a_{n-1}) \in \mathbb{Z}_q^n$ represent the coefficient vector of a, and let $\|\cdot\|$ and $\|\cdot\|_\infty$ denote the Euclidean norm and infinity norm, respectively. Then $\|a\|$ and $\|a\|_\infty$ can be denoted as $\sqrt{a_0^2 + a_1^2 + \cdots + a_{n-1}^2}$ and $\max\limits_{0 \le i \le n-1}(|a_i|)$, respectively.

$\boldsymbol{e} = Map(\boldsymbol{x}) \in \mathbb{Z}^{mn}$ is a column vector generated by concatenating coefficients of x_i $(1 \le i \le m)$ in sequence, and $\boldsymbol{x} = Map^{-1}(\boldsymbol{e}) \in R^m$ is the inverse process of Map. $(\{-1, +1\}^n)^{m\times m}$ is a matrix with m rows and m columns, of which the elements whose coefficients are −1 or 1 are chosen from R_q. $\boldsymbol{X}$=$Trans_{V\to M}(\boldsymbol{x}) \in \mathbb{Z}_q^{m\times n}$ is a $m\times n$ matrix whose rows are comprised of coefficient vectors of x_i $(1 \le i \le m)$, and $\boldsymbol{x}$=$Trans_{M\to V}(\boldsymbol{X}) \in R_q^m$ is a m-dimensional vector by viewing elements of each row in $\boldsymbol{X}$ as coefficients of a polynomial in R_q.

$a \leftarrow R_q$ is used to represent that a is uniformly selected in R_q at random. When we say $x \leftarrow \overline{\gamma}_\alpha$, we mean that x is a 'small' random error term chosen from $\overline{\gamma}_\alpha$ uniformly.

2.2. Lattices

Definition 1. There are n linear independent vectors $\boldsymbol{a}_1,\ \boldsymbol{a}_2,\ \cdots,\ \boldsymbol{a}_n \in \mathbb{R}^n$, let $\boldsymbol{A} = [\boldsymbol{a}_1,\ \boldsymbol{a}_2,\ \cdots,\ \boldsymbol{a}_n]$, the lattice Λ generated by $\boldsymbol{A}$ has the following form:

$$\Lambda = \mathcal{L}(\boldsymbol{A}) = \{\sum_{i=1}^{n} x_i \boldsymbol{a}_i \mid x_i \in \mathbb{Z},\ 1 \le i \le n\}$$

where $\boldsymbol{a}_1,\ \boldsymbol{a}_2,\ \cdots,\ \boldsymbol{a}_n$ is a basis of $\mathcal{L}$, and its rank is n.

Definition 2. For q prime, $\boldsymbol{A} \in \mathbb{Z}_q^{n\times m}$ and $u \in \mathbb{Z}_q^n$, define:

$$\Lambda_q^{\perp}(\boldsymbol{A}) = \{e \in \mathbb{Z}^m \ \ s.t.\ \boldsymbol{A}e = 0 (\bmod\ q)\}$$

$$\Lambda_q^{u}(\boldsymbol{A}) = \{e \in \mathbb{Z}^m \ \ s.t.\ \boldsymbol{A}e = u (\bmod\ q)\}$$

2.3. Discrete Gaussians

Definition 3. Let $\rho_s(\boldsymbol{x})$ denote the standard n-dimensional Gaussian distribution with center 0 and variance s, such as $\rho_s(\boldsymbol{x}) = \exp(-\pi \frac{\|\boldsymbol{x}\|^2}{s^2})$. For a given lattice Λ and $s > 0$, the sum $\rho_s(\Lambda) = \sum_{\boldsymbol{x}\in\Lambda} \rho_s(\boldsymbol{x})$ is finite, then define the lattice Gaussian distribution $D_{\Lambda,s}$ as

$$\exists \boldsymbol{y} \in \Lambda,\ D_{\Lambda,s}(\boldsymbol{y}) = \frac{\rho_s(\boldsymbol{y})}{\rho_s(\Lambda)}$$

Lemma 1. ([18]) For any lattice Λ of integer dimension m and any two reals $\varepsilon \in (0,\ 1)$ and $\sigma \ge \omega(\sqrt{\log m})$, we have

$$\Pr\{\boldsymbol{x} \sim D_{\Lambda,\sigma} :\ \|\boldsymbol{x}\| > \sqrt{m}\sigma\} \le \frac{1+\varepsilon}{1-\varepsilon} 2^{-m}$$

2.4. The R-LWE hardness assumption

Definition 4. For a uniformly random element $s \in R_q$ (secret) and an error distribution $\overline{\gamma}_\alpha$ over R_q, a sample from the R-LWE distribution $A_{s,\overline{\gamma}_\alpha}$ is generated by selecting $b \leftarrow R_q$ and an error term $x' \leftarrow \overline{\gamma}_\alpha$, and outputting $(b,\ bs + x') \in R_q \times R_q$.

Definition 5. ([22]) The Decisional R-LWE assumption is defined as follows: consider a prime $q = 1 \bmod 2n$ and an error distribution $\overline{\gamma}_\alpha$ over R_q, a decisional R-LWE assumption instance consists of access to an unspecified challenge oracle O which is either a truly random sampling oracle $O_\$$ or a pseudo-random sampling oracle O_s, where $O_\$$ outputs truly uniform random samples from $R_q \times R_q$, and O_s outputs samples $(b_i,\ b_i s + x_i)$ according to the R-LWE distribution. We say that an adversary A can solve the decisional R-LWE assumption if A's advantage $Adv(A) = |\Pr[A^{O_s} = 1] - \Pr[A^{O_\$} = 1]|$ is non-negligible for a random $s \in R_q$.

Assuming that the worst case γ-Ideal-SVP cannot be efficiently solved by using quantum algorithms for small γ, Steinfeld [21] showed that the R-LWE problem is hard (see Theorem 1).

Theorem 1. Assume that $\alpha q = \omega(n\sqrt{(\log n)})$ and $\alpha \in (0,\ 1)$ and $q = poly(n)$. There is a randomized polynomial time quantum reduction from γ-Ideal-SVP to R-LWE$_{q,\alpha}$,with $\gamma = \omega(n^{1.5} \log n) / \alpha$.

Lemma 2. ([21]) Assume that $\alpha q \ge \sqrt{n}$. For any $r \in R$, we have

$$\Pr_{y \leftarrow \overline{\gamma}_\alpha}[\|yr\|_\infty \ge \alpha q \omega(\log n) \cdot \|r\|] \le n^{-\omega(1)}$$

2.5. Important algorithms

2.5.1. Preimage sampling algorithm

RingSamplePre $(Rot_f^{\mathrm{T}}(\boldsymbol{a}),\ T_{\boldsymbol{a}},\ u,\ \sigma)$. Let q be a prime and $\boldsymbol{a}$ be a m-dimensional vector in R_q^m. On input a row vector $Rot_f^{\mathrm{T}}(\boldsymbol{a})$ with short trapdoor basis $T_{\boldsymbol{a}}$, a target image $u \in R_q$ and a Gaussian parameter σ, output $\boldsymbol{e} \in \mathrm{Z}^{mn}$ which sampled from a distribution statistically close to $D_{\mathrm{Z}^{mn},\sigma}$.

2.5.2. Trapdoor generation algorithm

The following theorem is used by Yang *et al.* [22] to generate the trapdoor over ideal lattices, we call the theorem as the trapdoor generation algorithm over rings, short for **RingGenTrap**.

Theorem 2. ([22]) There is a PPT algorithm with the following properties. It takes as inputs n, r > 0, q, $m_1 \in \mathbb{Z}$, $m_2 \in \mathbb{Z}$, a degree n polynomial $f \in \mathbb{Z}[x]$ and random vector $\boldsymbol{g}_1 \in (R_q^{\times})^{m_1}$. It also takes as inputs $\boldsymbol{u}_i \in R^{m_1}(0 \le i \le m_2)$ whose coefficients are selected from $D_{\mathbb{Z}^{m_1 \times n},\sigma}$. Let $f = \prod_{i \le t} f_i$ be its factorization over $\mathbb{Z}_q$, $\kappa = \lceil 1 + \log q \rceil$, $\Delta = (\prod_{i \le t}(1 + (q/3^r)^{\deg(f_i)}) - 1)^{1/2}$ and $m = m_1 + m_2$. Compute $\boldsymbol{g}_2 = (\boldsymbol{u}_1^{\mathrm{T}}\boldsymbol{g}_1, \boldsymbol{u}_2^{\mathrm{T}}\boldsymbol{g}_1, \cdots, \boldsymbol{u}_{m_2}^{\mathrm{T}}\boldsymbol{g}_1)$. The algorithm succeeds with probability $\ge 1 - p$, where $p = (1 - \prod_{i \le t}(1 - q^{-\deg(f_i)}))^{\sigma}$, when it does, it outputs $\boldsymbol{g} = (\boldsymbol{g}_1; \boldsymbol{g}_2) \in R_q^m$ and a (trapdoor) basis $\boldsymbol{S} \in \mathbb{Z}^{mn \times mn}$ of the lattice $\Lambda_q^{\perp}(Rot_f^{\mathrm{T}}(\boldsymbol{g}))$, such that:

1. The distance to uniformity of $\boldsymbol{g}$ is at most $p + m_2\Delta$;
2. $\|\boldsymbol{S}\| \le \sqrt{2n(9r+\sigma)}$.

2.5.3. Secret key extraction algorithm

ExtractLeft $(\boldsymbol{a}, \boldsymbol{b}, T_{\boldsymbol{a}}, u, \sigma)$. On input $\boldsymbol{a} \in R_q^m$ and a trapdoor $T_{\boldsymbol{a}}$ of $\Lambda_q^{\perp}(Rot_f^{\mathrm{T}}(\boldsymbol{a}))$, $\boldsymbol{b} \in R_q^m$, $u \in R_q$, $\sigma \ge \|T_{\boldsymbol{a}}\| \omega(\sqrt{\log m})$, then do:

1. Randomly sample a vector $\boldsymbol{e}_2' \in \mathbb{Z}^{mn}$ distributed statistically close to $D_{\mathbb{Z}^{mn},\sigma}$ and compute $\boldsymbol{e}_2 = Map^{-1}(\boldsymbol{e}_2') \in R^m$;
2. Run **RingSamplePre** $(Rot_f^{\mathrm{T}}(\boldsymbol{a}), T_{\boldsymbol{a}}, u_1, \sigma)$ to get $\boldsymbol{e}_1' \in \mathbb{Z}^{mn}$ and compute $\boldsymbol{e}_1 = Map^{-1}(\boldsymbol{e}_1') \in R^m$, where $u_1 = u - \boldsymbol{b}^{\mathrm{T}}\boldsymbol{e}_2 \in R_q$;
3. Output $\boldsymbol{e} = (\boldsymbol{e}_1; \boldsymbol{e}_2) \in R^{2m}$.

ExtractRight $(\boldsymbol{a}, \boldsymbol{b}, T_{\boldsymbol{b}}, R, u, \sigma)$. On input $\boldsymbol{a} \in R_q^m$, $\boldsymbol{b} \in R_q^m$ and a trapdoor $T_{\boldsymbol{b}}$ of $\Lambda_q^{\perp}(Rot_f^{\mathrm{T}}(\boldsymbol{b}))$, $R \in (\{-1,+1\}^n)^{m \times m}$, $u \in R_q$ and $\sigma \ge \|T_{\boldsymbol{a}}\| \omega(\sqrt{\log m})$, then do:

1. Set $\boldsymbol{d} = (\boldsymbol{a}; \boldsymbol{R}^{\mathrm{T}}\boldsymbol{a} + \boldsymbol{b}) \in R^{2m}$;
2. Construct a trapdoor $T_{\boldsymbol{d}}$ of $\Lambda_q^{\perp}(Rot_f^{\mathrm{T}}(\boldsymbol{d}))$, which is similar to [23];
3. Run **RingSamplePre** $(Rot_f^{\mathrm{T}}(\boldsymbol{d}), T_{\boldsymbol{d}}, u, \sigma)$ to get $\boldsymbol{e}' \in \mathbb{Z}^{2mn}$ and compute $\boldsymbol{e} = Map^{-1}(\boldsymbol{e}') \in R^{2m}$;
4. Output $\boldsymbol{e}$.

2.6. CP-ABE

Definition 6. (CP-ABE [3]) A CP-ABE scheme consists of the following algorithms:

- *Setup*(1^{κ}) → (*PK*, *MSK*). The *Setup* algorithm inputs a security parameter κ and produces the public key *PK* and master key *MSK*.
- *KeyGen*(*MSK*, *S*) → *SK*. The *KeyGen* algorithm inputs *MSK* and an attribute set *S* that depict the key, then produces a secret key *SK*.
- *Encrypt*(*PK*, *T*, *M*) → *CT*. The *Encrypt* algorithm inputs *PK*, an access structure *T* and a message *M*. It encrypts *M* and produces a ciphertext *CT*. Suppose *CT* implicitly contains *T*.
- *Decrypt*(*PK*, *SK*, *CT*) → *M*. The *Decrypt* algorithm inputs *PK*, *SK* and *CT*. It decrypts *CT* and outputs the message *M* if and only if *S* satisfies *T*.

2.7. CPA security game

The selective CPA security game for the CP-ABE scheme can be described like [19]. Before the *Setup* phase, the adversary *A* needs to state a challenge access structure T^* that he wishes to be challenged upon. Detailed steps of the selective security game are described in the following.

- *Initialization*. *A* picks T^* and sends it to the challenger *C*.
- *Setup*. *C* performs the *Setup* algorithm and gives *PK* to *A*.
- *Phase* 1. *A* submits an attribute set *S* for secret key query. The restriction is that *S* dose not satisfy T^*. For *S*, *C* performs the *KeyGen* algorithm and sends the secret key *SK* to *A*.
- *Challenge*. *A* sends two equal length messages M_0 and M_1 to *C*. *C* randomly selects one bit $\vartheta \in \{0, 1\}$, and encrypts M_ϑ by running the *Encrypt* algorithm under T^*. Then, *C* returns a challenge ciphertext CT^* to *A*.
- *Phase* 2. The same as *Phase* 1.
- *Guess*. *A* produces a guess ϑ' of ϑ.

  In the above selective security game, $A's$ advantage is

$$Adv(A) = |\Pr(\vartheta' = \vartheta) - 1/2|$$

Definition 9. A CP-ABE scheme is said to be selective CPA secure if all the PPT adversaries have at most a negligible advantage in the above game.

3. A small universe CP-ABE scheme from R-LWE

Based on the R-LWE problem, we propose a small universe CP-ABE scheme which is denoted as $\text{CP-ABE}^s_{\text{R-LWE}}$. For simplicity, we suppose there exists l normal attributes $L = \{1, 2, \cdots, l\}$ in the system. In $\text{CP-ABE}^s_{\text{R-LWE}}$, there is an access structure (T, t) being embedded in the ciphertext, which means that anyone who has t attributes in T can decrypt the ciphertext to obtain the right message, where $T \subseteq L$ and $t \in \mathbb{Z}$ is a threshold whose maximum value is d $(t \le d \le l)$. In order to decrypt correctly, we need to introduce d default attributes $D = \{l+1, \cdots, l+d\}$ into our system, these default attributes should be handled in the *Setup* phase, and all users have to add these attributes to generate their secret keys. The data owner also needs to add some default attributes to generate the ciphertext according to the access structure. Let $P = ((l+d)!)^2$. Now, detailed steps of $\text{CP-ABE}^s_{\text{R-LWE}}$ are described as follows.

- $Setup(1^\kappa) \to (PK, MSK)$. On input a security parameter κ which is a power of 2, do:
  1. For each $i \in L \cup D$, use the **RingGenTrap** algorithm to select a vector $\boldsymbol{b}_i \in R_q^m$ and a trapdoor $T_{\boldsymbol{b}_i}$ of $\Lambda_q^\perp(Rot_f^{\mathrm{T}}(\boldsymbol{b}_i))$.
  2. Set $u = Pu'$ by randomly selecting $u' \leftarrow R_q$.
  3. Output the public key $PK = \{\{\boldsymbol{b}_i \mid i \in L \cup D\}, u\}$ and master key $MSK = \{T_{\boldsymbol{b}_i} \mid i \in L \cup D\}$.
- $KeyGen(PK, MSK, U) \to SK$. On input the public key *PK*, the master key *MSK* and an attribute set *U*, do:
  1. Let $u' = u/P$ and $U' = U \cup D$.
  2. Select a uniformly random polynomial $p(y) = u' + \sum_{j=1}^{d} t_j y^j$ of degree d, where $t_j \leftarrow R_q$.

3. For each $i \in U'$, set $u_i = p(i) \in R_q$ and perform **RingSamplePre** $(Rot_f^{\mathrm{T}}(\boldsymbol{b}_i),\ T_{\boldsymbol{b}_i},\ u_i,\ \sigma)$ to obtain $\boldsymbol{e}_i'' \in \mathrm{Z}^{mn}$, then compute $\boldsymbol{e}_i' = Map^{-1}(\boldsymbol{e}_i'')$ and $\boldsymbol{e}_i = P\boldsymbol{e}_i'$ $(i.e.,\ \boldsymbol{b}_i^{\mathrm{T}}\boldsymbol{e}_i = Pu_i)$.
4. Output the secret key $SK = \{\boldsymbol{e}_i \mid i \in U'\}$.

- *Encrypt* $(PK,\ (T,\ t),\ \boldsymbol{m}) \to CT$. On input the public key *PK*, an access structure $(T,\ t)$ $(1 \le t \le \min\{|T|, d\})$ and a message $\boldsymbol{m} = (m_0,\ m_1,\ \cdots,\ m_{n-1})$, where $m_i \in \{0,\ 1\}$. Here, $\boldsymbol{m}$ can be viewed as a coefficient vector of a polynomial $m(x) = m_0 + m_1 x + \cdots + m_{n-1}x^{n-1} \in R_q$ (m for short), then do:
  1. Choose a uniformly random element $s \leftarrow R_q$.
  2. Let $T' = T \cup \{l+1, \cdots, l+d-t+1\}$.
  3. Set $c' = su + x' + \lfloor q/2 \rfloor m$, where $x' \leftarrow \overline{\gamma}_\alpha$.
  4. For each $i \in T'$, compute $\boldsymbol{c}_i = \boldsymbol{b}_i s + \boldsymbol{x}_i$, where $\boldsymbol{x}_i \leftarrow \overline{\gamma}_\alpha^m$.
  5. Output the ciphertext $CT = \{c', \{\boldsymbol{c}_i \mid i \in T'\}\}$.
- *Decrypt* $(PK,\ SK,\ CT) \to \boldsymbol{m}$. On input the public key *PK*, the secret key *SK* and the ciphertext *CT*, then do:
  1. If $|U \cap T| < t$, output $\perp$. Otherwise, we have $|U' \cap T'| \ge d+1$. Randomly choose a subset $I \subseteq U' \cap T'$ with $|I| = d+1$.
  2. For each $i \in I$, compute $K_i = \boldsymbol{e}_i^{\mathrm{T}}\boldsymbol{c}_i$.
  3. Compute
  $$K = \sum_{i \in I} L_i(0) K_i$$
  4. Compute $z = c' - K = z_0 + z_1 x + \cdots + z_{n-1}x^{n-1}$.
  5. For $i = 0,\ 1,\ \cdots,\ n-1$, if $|z_i| < q/4$, then output $m_i = 0$, otherwise, output $m_i = 1$.

3.1. Correctness and parameter setting

In this subsection, we show that our construction is correct.

For each $i \in I$, compute

$$K_i = \boldsymbol{e}_i^{\mathrm{T}}\boldsymbol{c}_i = \boldsymbol{e}_i^{\mathrm{T}}(\boldsymbol{b}_i \cdot s + \boldsymbol{x}_i) = (\boldsymbol{b}_i^{\mathrm{T}}\boldsymbol{e}_i)^{\mathrm{T}} s + \boldsymbol{e}_i^{\mathrm{T}}\boldsymbol{x}_i = Pu_i s + \boldsymbol{e}_i^{\mathrm{T}}\boldsymbol{x}_i$$

We have

$$\sum_{i \in L} L_i(0) u_i = i = \sum_{i=1}^{l} L_i(0) p(i) = u'$$

Then,

$$K = \sum_{i \in I} L_i(0) K_i = \sum_{i \in I} L_i(0)(Pu_i s + \boldsymbol{e}_i^{\mathrm{T}}\boldsymbol{x}_i) = us + \sum_{i \in L} L_i(0)\boldsymbol{e}_i^{\mathrm{T}}\boldsymbol{x}_i$$

Finally, compute

$$\begin{aligned} z &= c' - K \\ &= us + x' + \lfloor q/2 \rfloor m - us - \sum_{i \in I} L_i(0)\boldsymbol{e}_i^{\mathrm{T}}\boldsymbol{x}_i \\ &= \lfloor q/2 \rfloor m + x' - \sum_{i \in I} L_i(0)\boldsymbol{e}_i^{\mathrm{T}}\boldsymbol{x}_i \\ &\approx \lfloor q/2 \rfloor m \end{aligned}$$

To decrypt the ciphertext correctly, it requires that the absolute value of each coefficient of the error term $(x' - \sum_{i \in I} L_i(0)\boldsymbol{e}_i^{\mathrm{T}}\boldsymbol{x}_i)$ is less than $q/4$ with overwhelming probability. Here, we only need to compute $\|(x'-\sum_{i\in I} L_i(0)\boldsymbol{e}_i^{\mathrm{T}}\boldsymbol{x}_i)\|_\infty < q/4$. Then

$$\|(x'-\sum_{i\in I} L_i(0)\boldsymbol{e}_i^{\mathrm{T}}\boldsymbol{x}_i)\|_\infty \le \|x'\|_\infty + \|\sum_{i\in I} L_i(0)\boldsymbol{e}_i^{\mathrm{T}}\boldsymbol{x}_i\|_\infty$$

By Lemma 2, we have $\|x'\|_\infty < \alpha q\omega(\log n)$. Let $\boldsymbol{x}_i = (x_{i1}, x_{i2}, \cdots, x_{im})^{\mathrm{T}}$ and $\boldsymbol{e}_i'^{\mathrm{T}} = (e'_{i1}, e'_{i2}, \cdots, e'_{im})$. Then

$$\|\boldsymbol{e}_i'^{\mathrm{T}}\boldsymbol{x}_i\|_\infty \le \sum_{j=1}^{m}\|e'_{ij}x_{ij}\|_\infty < m\alpha q\omega(\log n)\cdot\|e'_{ij}\|$$

From Lemma 1, $\|e'_{ij}\| < \sigma\sqrt{n}$. Hence,

$$\|\boldsymbol{e}_i'^{\mathrm{T}}\boldsymbol{x}_i\|_\infty < m\alpha q\sigma\sqrt{n}\omega(\log n)$$

Let $\alpha q = n\sqrt{\log n}$. Since $|PL_j(0)| \le (n!)^3$, we have

$$\|x'-\sum_{i\in I} L_i(0)\boldsymbol{e}_i'^{\mathrm{T}}\boldsymbol{x}_i\|_\infty < \alpha q\omega(\log n) + (d+1)((l+d)!)^3 m\alpha q\sigma\sqrt{n}\omega(\log n)$$

$$< 4((l+d)!)^4 n^{2.5}\sigma\log^{1.5} n$$

We set $q \ge 16((l+d)!)^4 n^{2.5}\sigma\log^{1.5} n$ to ensure correctness. Simultaneously, other parameters are set as:

- $m=\kappa$ and $n=m/2$.
- $\sigma = m\cdot\omega(\log m)$. In the *KeyGen* algorithm, $\boldsymbol{e}_i'$ obeys Gaussian distributions with center 0 and variance σ , while $\boldsymbol{e}_i = P\boldsymbol{e}_i'$ obeys Gaussian distributions with center 0 and variance $P^2\sigma$, and $P^2\sigma$ also satisfies $P^2\sigma = m\cdot\omega(\log m)$.
- $l = n^{\varepsilon}$ for a certain constant $\varepsilon \in (0,\ 1/2)$.
- Due to $d \le l$, we have $(l+d)! \le (2l)! \le (2l)^{2l} = 2^{2l\log 2l}$, thus $\alpha = n\sqrt{\log n}/q$ $=1/(2^{8n^{\varepsilon}\log 2n^{\varepsilon}+6}\cdot poly(n))$.

Combining above parameter setting and Theorem 1, we get security under the hardness of $2^{O(n^{\varepsilon}\log 2n^{\varepsilon})}$-approximating Ideal-SVP applying algorithms that run in time $2^{O(n^{\varepsilon}\log 2n^{\varepsilon})}$.

3.2. Security analysis

Now, we reduce CPA security of $\text{CP-ABE}^{s}_{\text{R-LWE}}$ to the decisional R-LWE assumption.

Theorem 3. If there exists a PPT adversary *A* can win the $\text{CP-ABE}^{s}_{\text{R-LWE}}$ scheme with non-negligible advantage $\varepsilon > 0$, then there is a PPT algorithm *B* that can solve the decisional R-LWE assumption with the same advantage.

Proof. Recall from Definition 5 that a R-LWE assumption instance is provided as a sampling oracle O which can be either a truly random sampling oracle $O_\$$ or a pseudo-random sampling oracle O_s for a certain secret $s \in R_q$. The simulator *B* uses *A* to distinguish the two, and does as follows:

- ➢ *Instance*: *B* requests from *O* and obtains $(l+d)m+1$ R-LWE samples $(v_0, w_0) \in R_q \times R_q$, $(v_i^j, w_i^j) \in R_q \times R_q$, $(1 \le i \le l+d,\ 1 \le j \le m)$, where v_0 implies that there exists $v_0' \in R_q$ such that $v_0 = Pv_0'$.
- ➢ *Initialization*: *A* picks a challenge access structure $(T^*,\ t^*)$ $(1 \le t^* \le \min\{|T^*|, d\})$ and sends it to *A*.

- *Setup*: After receiving $(T^*,\ t^*)$, *B* generates the public key *PK* as:
  1. Let $T' = T^* \cup \{l+1,\ \cdots,\ l+d+1-t^*\}$.
  2. Set $u = v_0$ and $\boldsymbol{b}_i = (v_i^1,\ v_i^2,\ \cdots,\ v_i^m)^{\mathrm{T}}$ for each $i \in T'$.
  3. For each $i \in (L \cup D) \setminus T'$, use the **RingGenTrap** algorithm to choose $\boldsymbol{b}_i \in R_q^m$ and a trapdoor $T_{\boldsymbol{b}_i}$ of $\Lambda_q^{\perp}(Rot_f^{\mathrm{T}}(\boldsymbol{b}_i))$.

  Finally, *B* sends $PK = \{\{\boldsymbol{b}_i \mid i \in L \cup D\},\ u\}$ to *A*, and keeps $\{T_{\boldsymbol{b}_i} \mid i \in L \cup D\}$ secret.
- *Phase* 1 : *A* can ask *B* to get the secret key *SK* corresponds to any attribute set *U*, where $|U \cap T^*| < t^*$. *B* creates *SK* as follows.
  1. Let $U' = U \cup D$ and $|U \cap T^*| \le t^* - 1$. Then we have $|U' \cap T'| \le d$. Assume that $|U' \cap T'| = \eta$ and the first η attributes of U' are the same as T'.
  2. Represent the shares of u/P as $p(y) = u/P + \sum_{j=1}^{d} t_j y^j$, where $t_1,\ t_2,\ \cdots,\ t_d \leftarrow R_q$ are variables.
  3. For each $i \in U' \cap T'$, sample $\boldsymbol{e}_i'' \leftarrow D_{\mathbb{Z}^{mn},\sigma}$, compute $\boldsymbol{e}_i' = Map^{-1}(\boldsymbol{e}_i'')$ and $u_i = \boldsymbol{b}_i^{\mathrm{T}} \boldsymbol{e}_i'$. Then set $p(i) = u_i$ and $\boldsymbol{e}_i = P\boldsymbol{e}_i'$ for every $i \in U' \cap T'$.
  4. Since $\eta \le d$, randomly choose $d-\eta$ shares $u_{\eta+1},\ u_{\eta+2},\ \cdots,\ u_d \leftarrow R_q$ and set $p(i) = u_i$ $(i = \eta+1,\ \eta+2,\ \cdots,\ d)$. Then the values for $t_1,\ t_2,\ \cdots,\ t_d$ are determined. This determines all $|U'|$ shares $p(1),\ \cdots,\ p(|U'|)$.
  5. For each $i \in U' \setminus T'$, perform **RingSamplePre** $(\boldsymbol{b}_i,\ T_{\boldsymbol{b}_i},\ u_i,\ \sigma)$ to obtain $\boldsymbol{e}_i''$, then compute $\boldsymbol{e}_i = P \cdot Map^{-1}(\boldsymbol{e}_i'')$.

  At last, *B* sends $SK = \{\boldsymbol{e}_i \mid i \in U'\}$ to *A*.
- *Challenge*: *A* sends messages $\boldsymbol{m}_0$ and $\boldsymbol{m}_1$ to *B*, where $\boldsymbol{m}_0 = (m_{00},\ m_{01},\ \cdots,\ m_{0,n-1})$, $\boldsymbol{m}_1 = (m_{10},\ m_{11},\ \cdots,\ m_{1,n-1})$, $m_{i,j} \in \{0,\ 1\}$, $i = 0,\ 1$; $j = 0,\ 1,\ \cdots,\ n-1$. After receiving the messages, *B* picks $\vartheta \in \{0,\ 1\}$ at random and constructs the challenge ciphertext CT^* as:

  – Set $c' = w_0 + \lfloor q/2 \rfloor m_\vartheta$;

  – For each $i \in T'$, set $\boldsymbol{c}_i = (w_i^1,\ w_i^2,\ \cdots,\ w_i^m)$.

  B sends $CT^* = \{c', \{\boldsymbol{c}_i \mid i \in T'\}\}$ to *A*.
- *Phase* 2 : The same as *Phase* 1.
- *Guess*: *A* produces a guess ϑ' of ϑ. *B* applies *A*'s guess to determine an answer on the R-LWE oracle: if $\vartheta' = \vartheta$, *B* produces "R-LWE", otherwise it produces "truly random".

If *O* is a R-LWE oracle for a certain secret $s \in R_q$, we show that CT^* is a valid ciphertext for *s* as follows: $c' = w_0 + \lfloor q/2 \rfloor m_\vartheta = v_0 s + x' + \lfloor q/2 \rfloor m_\vartheta$, and $\boldsymbol{c}_i = (w_i^1,\ w_i^2,\ \cdots,\ w_i^m) = \boldsymbol{b}_i s + \boldsymbol{x}_i$ for each $i \in T'$. If the probability that *A* guesses the right ϑ is $\varepsilon + 1/2$, then *B* can win the game with the same probability.

If *O* is $O_\$$, the ciphertext CT^* is completely random from *A*'s view, thus the probability that *A* guesses the right ϑ is $1/2$, *B* also has the same probability to win the game.

Therefore, if *A* can win the above security game with non-negligible advantage $\varepsilon > 0$, then *B* can solve the decisional R-LWE assumption with the same advantage

$$\begin{aligned} Adv(B) &= |\Pr[B^{O_s}=1] - \Pr B^{O_\$}=1]| \\ &= |\varepsilon + 1/2 - 1/2| \\ &= \varepsilon \end{aligned}$$

4. A large-universe CP-ABE scheme from R-LWE

The proposed scheme in Section 3 is bounded in selecting parameters for key generation and encryption once the public parameters have been set. And the expression of attribute is not flexible enough. In order to support unbounded attribute space and improve the expressiveness of attribute, we combine the above scheme and some fixed FRD function to construct a large universe CP-ABE scheme which is denoted as $\text{CP-ABE}^{l}_{\text{R-LWE}}$.

Definition 10. [23] Let q be a prime. If there is a function $H: \mathbb{Z}_q^n \to \mathbb{Z}_q^{n\times n}$ satisfies:

- for all the different $u,\ v \in \mathbb{Z}_q^n$, the matrix $H(u)-H(v) \in \mathbb{Z}_q^{n\times n}$ is full rank;
- H is computable in polynomial time in $n\log q$.

Then we call H is an encoding with full-rank differences (FRD).

Suppose there exists l normal attributes $L=\{attr_1,\ attr_2,\ \cdots,\ attr_l\}$ in the system. Each polynomial in R_q is the possible value of $attr_i$. Let $t_i << attr_i$ denote $t_i \in R_q$ is a value of $attr_i$, where $i=1,\ 2,\ \cdots,\ l$. Now, detailed steps of $\text{CP-ABE}^{l}_{\text{R-LWE}}$ are described as follows.

➢ *Setup*$(1^\kappa) \to (PK,\ MSK)$. On input a security parameter κ which is a power of 2, do:
   1. Use the **RingGenTrap** algorithm to select a vector $\boldsymbol{a} \in R_q^m$ and a trapdoor $T_{\boldsymbol{a}}$ of $\Lambda_q^{\perp}(Rot_f^{\mathrm{T}}(\boldsymbol{a}))$.
   2. For each $i \in L \cup D$, select a uniformly random vector $\boldsymbol{b}_i \leftarrow R_q^m$.
   3. Select a uniformly random vector $\boldsymbol{b}=(b_1,\ b_2,\ \cdots,b_m) \leftarrow R_q^m$.
   4. Set $u = Pu'$ by selecting $u' \leftarrow R_q$ at random.
   5. Select a FDR function $H: \mathbb{Z}_q^n \to \mathbb{Z}_q^{n\times n}$.
   6. Output the public key $PK=\{\boldsymbol{a},\ \boldsymbol{b},\ \{\boldsymbol{b}_i \mid i \in L\cup D\},\ u,\ H\}$ and master key $MSK=\{T_{\boldsymbol{a}}\}$.

➢ *KeyGen*$(PK,\ MSK,\ U) \to SK$. On input the public key *PK*, the master key *MSK* and an attribute set $U=\{t_i \mid t_i << attr_i\}$, do:
   1. Let $u' = u/P$ and $U' = U \cup D$.
   2. Compute $\boldsymbol{B} = Trans_{V\to M}(\boldsymbol{b})$.
   3. Select a uniformly random polynomial $p(y) = u' + \sum_{j=1}^{d} t_j y^j$ of degree d, where $t_j \leftarrow R_q$.
   4. For each $i \in U$, compute $\boldsymbol{h}_i = Trans_{M\to V}(\boldsymbol{B}H^{\mathrm{T}}(\boldsymbol{t}_i)) \in R_q^m$, set $u_i = p(i) \in R_q$ and $E_i = (\boldsymbol{a};\ \boldsymbol{b}_i + \boldsymbol{h}_i)$, then perform **ExtractLeft** $(\boldsymbol{a},\ \boldsymbol{b}_i+\boldsymbol{h}_i,\ T_{\boldsymbol{a}},\ u_i,\ \sigma)$ to obtain $\boldsymbol{e}_i' \in R^{2m}$, and compute $\boldsymbol{e}_i = P\boldsymbol{e}_i'$.
   5. For every $i \in D$, set $u_i = p(i) \in R_q$ and $E_i = (\boldsymbol{a};\ \boldsymbol{b}+\boldsymbol{b}_i) \in R_q^{2m}$, then perform **ExtractLeft** $(\boldsymbol{a},\ \boldsymbol{b}\ +\boldsymbol{b}_i,\ T_{\boldsymbol{a}},\ u_i,\ \sigma)$ to obtain $\boldsymbol{e}_i' \in R^{2m}$, and compute $\boldsymbol{e}_i = P\boldsymbol{e}_i'$.
   6. Output the secret key $SK=\{\boldsymbol{e}_i \mid i \in U'\}$.

- $Encrypt(PK,\ (T,\ t),\ \boldsymbol{m}) \rightarrow CT$. On input the public key *PK*, an access structure $(T,\ t)$ and a message $\boldsymbol{m} = (m_0,\ m_1,\ \cdots,\ m_{n-1})$, where $T = \{t_i \mid t_i << attr_i\}$, $1 \le t \le \min\{|T|,\ d\}$ and $m_i \in \{0,\ 1\}$. Here, $\boldsymbol{m}$ is viewed as a coefficient vector of a polynomial $m(x) = m_0 + m_1 x + \ \cdots + m_{n-1} x^{n-1} \in R_q$ (m for short), do:
  1. Choose a uniformly random element $s \leftarrow R_q$.
  2. Compute $\boldsymbol{B} = Trans_{V \rightarrow M}(\boldsymbol{b})$.
  3. Let $V = \{l+1, \cdots, l+d-t+1\}$ and $T' = T \cup V$.
  4. Set $c' = us + x' + \lfloor q/2 \rfloor m$, where $x' \leftarrow \overline{\gamma}_\alpha$.
  5. Set $\boldsymbol{c}_0 = \boldsymbol{a}s + \boldsymbol{x}$, where $\boldsymbol{x} \leftarrow \overline{\gamma}_\alpha^m$.
  6. For each $i \in T$, compute $\boldsymbol{h}_i = Trans_{M \rightarrow V}(\boldsymbol{B}H^{\mathrm{T}}(\boldsymbol{t}_i)) \in R_q^m$, randomly select $\boldsymbol{R}_i \leftarrow (\{-1,+1\}^n)^{m \times m}$ and compute $\boldsymbol{c}_i = (\boldsymbol{b}_i + \boldsymbol{h}_i)s + \boldsymbol{R}_i^{\mathrm{T}} \boldsymbol{x}$.
  7. For each $i \in V$, select $\boldsymbol{R}_i \leftarrow (\{-1,+1\}^n)^{m \times m}$ at random and compute $\boldsymbol{c}_i = (\boldsymbol{b} + \boldsymbol{b}_i)s + \boldsymbol{R}_i^{\mathrm{T}} \boldsymbol{x}$.
  8. Output the ciphertext $CT = \{T, c', \boldsymbol{c}_0, \{\boldsymbol{c}_i \mid i \in T'\}\}$.
- $Decrypt(PK,\ SK,\ CT) \rightarrow \boldsymbol{m}$. On input the public key *PK*, the secret key *SK* and the ciphertext *CT*, then do:
  1. If $|U \cap T| < t$, output $\perp$. Otherwise, we have $|U \cap T| \ge t$, due to $|U' \cap T'| \ge d+1$. Randomly choose a subset $I \in U' \cap T'$ with $|I| = d+1$. Let S_1 denote the subscript set of normal attributes in *I*, and S_2 denote the default attributes in *I*. For convenience, we set $I = S_1 \cup S_2$.
  2. For each $i \in I$, compute $K_i = \boldsymbol{e}_i^{\mathrm{T}}(\boldsymbol{c}_0;\ \boldsymbol{c}_i)$.
  3. Compute
  $$K = \sum_{i \in I} L_i(0) K_i$$
  4. Compute $z = c' - K = z_0 + z_1 x + \cdots + z_{n-1} x^{n-1}$.
  5. For $i = 0,\ 1,\ \cdots,\ n-1$, if $|z_i| < q/4$, then output $m_i = 0$, otherwise, output $m_i = 1$.

4.1. Correctness

In this subsection, we show that the $\text{CP-ABE}^{l}_{\text{R-LWE}}$ scheme is correct.

Suppose $|S_1| = t$, $|S_2| = d+1-t$. For each $i \in S_1$, compute

$$\begin{aligned} K_i &= \boldsymbol{e}_i^{\mathrm{T}} \begin{pmatrix} \boldsymbol{c}_0 \\ \boldsymbol{c}_i \end{pmatrix} \\ &= \boldsymbol{e}_i^{\mathrm{T}} \begin{pmatrix} \boldsymbol{a}s + \boldsymbol{x} \\ (\boldsymbol{b}_i + \boldsymbol{h}_i)s + \boldsymbol{R}_i^{\mathrm{T}} \boldsymbol{x} \end{pmatrix} \\ &= \boldsymbol{e}_i^{\mathrm{T}}(E_i s) + \boldsymbol{e}_i^{\mathrm{T}} \begin{pmatrix} \boldsymbol{x} \\ \boldsymbol{R}_i^{\mathrm{T}} \boldsymbol{x} \end{pmatrix} \\ &= P u_i s + \boldsymbol{e}_i^{\mathrm{T}} \begin{pmatrix} \boldsymbol{x} \\ \boldsymbol{R}_i^{\mathrm{T}} \boldsymbol{x} \end{pmatrix} \end{aligned}$$

For each $i \in S_2$, compute

$$\begin{aligned} K_i &= \boldsymbol{e}_i^{\mathrm{T}} \begin{pmatrix} \boldsymbol{c}_0 \\ \boldsymbol{c}_i \end{pmatrix} \\ &= \boldsymbol{e}_i^{\mathrm{T}} \begin{pmatrix} \boldsymbol{a}s + \boldsymbol{x} \\ (\boldsymbol{b} + \boldsymbol{b}_i)s + \boldsymbol{R}_i^{\mathrm{T}} \boldsymbol{x} \end{pmatrix} \\ &= \boldsymbol{e}_i^{\mathrm{T}} (E_i s) + \boldsymbol{e}_i^{\mathrm{T}} \begin{pmatrix} \boldsymbol{x} \\ \boldsymbol{R}_i^{\mathrm{T}} \boldsymbol{x} \end{pmatrix} \\ &= P u_i s + \boldsymbol{e}_i^{\mathrm{T}} \begin{pmatrix} \boldsymbol{x} \\ \boldsymbol{R}_i^{\mathrm{T}} \boldsymbol{x} \end{pmatrix} \end{aligned}$$

For every $i \in I$, let $\xi_i = K_i - P u_i s$. Then,

$$K = \sum_{i \in I} L_i(0) K_i = \sum_{i \in I} L_i(0)(P u_i s + \xi_i) = us + \sum_{i \in I} L_i(0) \xi_i$$

Finally, compute

$$\begin{aligned} z &= c' - K \\ &= us + x' + \lfloor q/2 \rfloor m - us - \sum_{i \in I} L_i(0) \xi_i \\ &= \lfloor q/2 \rfloor m + x' - \sum_{i \in I} L_i(0) \xi_i \\ &\approx \lfloor q/2 \rfloor m \end{aligned}$$

As Subsection 3.1, we can select appropriate parameters to satisfy

$$\| x' - \sum_{i \in I} L_i(0) \xi_i \|_\infty < q/4$$

4.2. Security analysis

In this subsection, we prove the security of $\text{CP-ABE}^{l}_{\text{R-LWE}}$ in the selective model in Subsection 2.7.

Theorem 4. If there exists a PPT adversary A can win the $\text{CP-ABE}^{l}_{\text{R-LWE}}$ scheme with non-negligible advantage $\varepsilon > 0$, then there is a PPT algorithm B that can solve the decisional R-LWE assumption with the same advantage.

Proof. Recall from Definition 5 that a R-LWE assumption instance is provided as a sampling oracle O which can be either a truly random sampling oracle $O_\$$ or a pseudo-random sampling oracle O_s for a certain secret $s \in R_q$. The simulator B uses A to distinguish the two, and does:

- *Instance*: B requests from O and obtains $m+1$ R-LWE samples $(v_i,\ w_i) \in R_q \times R_q$ $(0 \le i \le m)$, where v_0 implies that there exists $v_0' \in R_q$ such that $v_0 = P v_0'$.
- *Initialization*: A sends a challenge access structure $(T^*,\ t^*)$ to A, where $T^* = \{t_i^* \mid t_i^* << attr_i\}$ and $1 \le t^* \le \min\{|T^*|,\ d\}$.
- *Setup*: After receiving $(T^*,\ t^*)$, B generates the public key PK as follows:
  1. Let $V = \{l+1,\ \cdots,\ l+d+1-t^*\}$ and $T' = T^* \cup V$.
  2. Set $u = v_0$ and $\boldsymbol{a} = (v_1,\ v_2,\ \cdots,\ v_m)^{\mathrm{T}} \in R_q^m$.
  3. Use the **RingGenTrap** algorithm to choose $\boldsymbol{b} = (b_1,\ b_2,\ \cdots,\ b_m) \in R_q^m$ and a trapdoor $T_{\boldsymbol{b}}$ of $\Lambda_q^{\perp}(Rot_f^{\mathrm{T}}(\boldsymbol{b}))$. Simultaneously, compute $\boldsymbol{B} = Trans_{V \to M}(\boldsymbol{b})$.

4. For each $i \in T^*$, compute $\boldsymbol{h}_i = Trans_{M \to V}(\boldsymbol{B}H^{\mathrm{T}}(\boldsymbol{t}_i)) \in R_q^m$, select $\boldsymbol{R}_i \in (\{-1,\ +1\}^n)^{m \times m}$ at random and set $\boldsymbol{b}_i = \boldsymbol{R}_i^{\mathrm{T}}\boldsymbol{a} - \boldsymbol{h}_i$.
5. For every $i \in L \setminus T^*$, select $\boldsymbol{R}_i \in (\{-1,\ +1\}^n)^{m \times m}$ randomly and set $\boldsymbol{b}_i = \boldsymbol{R}_i^{\mathrm{T}}\boldsymbol{a}$.
6. For each $i \in V$, select $\boldsymbol{R}_i \in (\{-1,\ +1\}^n)^{m \times m}$ at random and set $\boldsymbol{b}_i = \boldsymbol{R}_i^{\mathrm{T}}\boldsymbol{a} - \boldsymbol{b}$.
7. For each $i \in D \setminus V$, randomly choose $\boldsymbol{R}_i \in (\{-1,\ +1\}^n)^{m \times m}$ and set $\boldsymbol{b}_i = \boldsymbol{R}_i^{\mathrm{T}}\boldsymbol{a}$.

Finally, *B* sends $PK = \{\boldsymbol{a},\ \boldsymbol{b},\ \{\boldsymbol{b}_i \mid i \in L \cup D\},\ u\}$ to *A*, and keeps $(T_{\boldsymbol{b}},\ \{\boldsymbol{R}_i \mid i \in L \cup D\})$ secret.

- *Phase* 1: *A* can ask *B* to get the secret key *SK* corresponds to any attribute set $U = \{t_i \mid t_i << attr_i\}$, where $|U \cap T^*| < t^*$. *B* creates *SK* as follows.
  1. Let $U' = U \cup D$ and $|U \cap T^*| \le t^* - 1$. Then we have $|U' \cap T'| \le d$. Assume that $|U' \cap T'| = \eta$ and the first η attributes of U' are the same as T'.
  2. Represent the shares of u / P as $p(y) = u / P + \sum_{j=1}^{d} t_j y^j$, where $t_1,\ t_2,\ \cdots,\ t_d \leftarrow R_q$ are variables.
  3. For each $i \in U$, compute $\boldsymbol{h}_i = Trans_{M \to V}(\boldsymbol{B}H^{\mathrm{T}}(\boldsymbol{t}_i)) \in R_q^m$, define $E_i = (\boldsymbol{a};\ \boldsymbol{b}_i + \boldsymbol{h}_i)$, sample $\boldsymbol{e}_i'' \leftarrow D_{\mathbb{Z}^{2mn},\sigma}$, compute $\boldsymbol{e}_i' = Map^{-1}(\boldsymbol{e}_i'')$ and $u_i = E_i^{\mathrm{T}}\boldsymbol{e}_i'$. Then set $p(i) = u_i$ and $\boldsymbol{e}_i = P\boldsymbol{e}_i'$ for every $i \in U$.
  4. For each $i \in V$, define $E_i = (\boldsymbol{a};\ \boldsymbol{b} + \boldsymbol{b}_i)$, sample $\boldsymbol{e}_i'' \leftarrow D_{\mathbb{Z}^{2mn},\sigma}$, compute $\boldsymbol{e}_i' = Map^{-1}(\boldsymbol{e}_i'')$ and $u_i = E_i^{\mathrm{T}}\boldsymbol{e}_i'$. Then set $p(i) = u_i$ and $\boldsymbol{e}_i = P\boldsymbol{e}_i'$ for every $i \in V$.
  5. Since $\eta \le d$, randomly choose $d - \eta$ shares $u_{\eta+1},\ u_{\eta+2},\ \cdots,\ u_d \leftarrow R_q$ and set $p(i) = u_i$ $(i = \eta+1,\ \eta+2,\ \cdots,\ d)$. Then the values for $t_1,\ t_2,\ \cdots,\ t_d$ are determined. This determines all $|U'|$ shares $p(1),\ \cdots,\ p(|U'|)$.
  6. For every $i \in D \setminus V$, take $E_i = (\boldsymbol{a};\ \boldsymbol{b} + \boldsymbol{b}_i)$ and perform **ExtractRight** $(\boldsymbol{a},\ \boldsymbol{b},\ \boldsymbol{R}_i,\ T_{\boldsymbol{b}},\ u_i,\ \sigma)$ to obtain $\boldsymbol{e}_i'$, then compute $\boldsymbol{e}_i = P\boldsymbol{e}_i'$.

  At last, *B* sends $SK = \{\boldsymbol{e}_i \mid i \in U'\}$ to *A*.
- *Challenge*: *A* sends messages $\boldsymbol{m}_0$ and $\boldsymbol{m}_1$ to *B*, where $\boldsymbol{m}_0 = (m_{00},\ m_{01},\ \cdots,\ m_{0,n-1})$, $\boldsymbol{m}_1 = (m_{10},\ m_{11},\ \cdots,\ m_{1,n-1})$, $m_{i,j} \in \{0,\ 1\}$, $i = 0,\ 1$; $j = 0,\ 1,\ \cdots,\ n-1$. After receiving the messages, *B* picks $\vartheta \in \{0,\ 1\}$ at random and constructs the challenge ciphertext CT^* as:

  – Let $\boldsymbol{w} = (w_1,\ w_2,\ \cdots,\ w_m)$;
  – Set $c' = w_0 + \lfloor q/2 \rfloor m_\vartheta$ and $\boldsymbol{c}_0 = \boldsymbol{w}$;
  – If $i \in T'$, set $\boldsymbol{c}_i = \boldsymbol{R}_i^{\mathrm{T}}\boldsymbol{w}$.

  B sends $CT^* = \{c',\ \boldsymbol{c}_0,\ \{\boldsymbol{c}_i \mid i \in T'\}\}$ to *A*.
- *Phase* 2: The same as *Phase* 1.
- *Guess*: *A* produces a guess ϑ' of ϑ. *B* applies *A*'s guess to determine an answer on the R-LWE oracle: if $\vartheta' = \vartheta$, *B* produces "R-LWE", otherwise it produces "truly random".

5. Performance analysis

We compare the performance of our schemes with the existing schemes in [19]. We denote the first scheme in [19] as $\text{CP-ABE}_{\text{LWE}}^{s}$, and the second one as $\text{CP-ABE}_{\text{LWE}}^{l}$. In order to make the analysis more understandable, new notations used in comparison are shown as follows:

l : the number of the attribute universe

k : the number of attributes a user has

t : the number of the least attributes to decrypt the ciphertext

θ : the number of attributes appeared in the access structure

δ : a real such that $n^{1+\delta} > \lceil (n+1)\log q + \omega(\log n) \rceil$ (in [19])

Table 1 shows the performance comparison results among our schemes and the ones in [19], each scheme is compared in terms of *PK* size, *SK* size, *message* size, *CT* size, multiplications in encryption per bit (MEPB), multiplications in decryption per bit (MDPB), underlying hardness assumption (UHA), principle of operation (POO) and worst case problem (WCP). All sizes are in bits. For convenience, we let $v = \theta + l - t + 1$ and $w = \log q$.

Table 1. Performance comparison among our schemes and the ones in [19]

Schemes	Small universe		Large universe	
	$\text{CP-ABE}^s_{\text{LWE}}$	$\text{CP-ABE}^s_{\text{R-LWE}}$	$\text{CP-ABE}^l_{\text{LWE}}$	$\text{CP-ABE}^l_{\text{R-LWE}}$
PK size	$[(12l+12)n^{1+\delta}+1]nw$	$(4l\ n+1)nw$	$[(12l+12)n^{1+\delta}+1]nw$	$[(4l+4)n+1]nw$
SK size	$(2k+2l)n^{1+\delta}w$	$(2k+2l)n^2w$	$(2k+2l)n^{1+\delta}w$	$(4k+4l)n^2w$
Message size	$\{0,1\}$	$\{0,1\}^n$	$\{0,1\}$	$\{0,1\}^n$
CT size	$[(v+1)n^{1+\delta}+1]w$	$(2vn+1)nw$	$[(v+1)n^{1+\delta}+1]w$	$[(2v+2)n+1]nw$
MEPB	$O(vn^{2+2\delta})$	$O(vn\log n)$	$O(vn^{3+3\delta})$	$O(vn^2\log n)$
MDPB	$O(l\ n^{1+\delta})$	$O(l\ n\log n)$	$O(l\ n^{1+\delta})$	$O(l\ n\log n)$
UHA	LWE	R-LWE	LWE	R-LWE
POO	Matrix operation	FFT	Matrix operation	FFT
WCP	$SIVP_\gamma$	$\gamma - Ideal - SVP$	$SIVP_\gamma$	$\gamma - Ideal - SVP$

As shown in **Table 1**, compared with $\text{CP-ABE}^s_{\text{LWE}}$ and $\text{CP-ABE}^l_{\text{LWE}}$ in [19], *PK* size in our schemes are reduced nearly 3*n* times when encrypting messages of the same size, our schemes also have smaller *SK* size and *CT* size. Our schemes require less computation in the encryption and decryption phases, mainly because our scheme is constructed based on the R-LWE assumption, which can use FFT to improve the efficiency of encryption from $O(vn^2)$ to $O(vn\log n)$ and decryption from $O(l\ n^2)$ to $O(l\ n\log n)$. Especially, MEPB in $CP-ABE^s_{R-LWE}$ is $O(vn\log n)$, which is much more less than that in $\text{CP-ABE}^s_{\text{LWE}}$. In **Table 1**, $\text{CP-ABE}^s_{\text{LWE}}$ and $\text{CP-ABE}^l_{\text{LWE}}$ are secure under the LWE assumption, which can be reduced to SIVP_γ on arbitrary lattices; while our schemes are secure under the R-LWE assumption, which also can be reduced to γ-Ideal-SVP on ideal lattices. As a whole, our schemes are secure and more efficient than $\text{CP-ABE}^s_{\text{LWE}}$ and $\text{CP-ABE}^l_{\text{LWE}}$ in [19].

6. Conclusion

Based on the R-LWE assumption, a small universe CP-ABE scheme is proposed, which has a flexible and simple threshold access structure. On this basis, we proposed a large universe CP-ABE scheme from R-LWE with the help of a FRD function, which can achieve unbounded attribute space and enhance the expressiveness of attribute. Both schemes are proved to be secure under the R-LWE assumption. Moreover, we compared our schemes with the schemes in [19], and then found that ours are more efficient and have shorter public key, secret key and ciphertext sizes.

References

[1] S. Gorbunov, V. Vaikuntanathan and H. Wee, "Attribute-based encryption for circuits," *Journal of ACM*, vol. 62, no. 6, pp. 45:1–45:33, 2015. Article(CrossRefLink)

[2] A. Sahai and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in *Proc. of 13th ACM Conference on Computer and Communications Security*, pp. 89–98, October 30 - November 3, 2006. Article(CrossRefLink)

[3] J. Bethencourt, A. Sahai and B. Waters, "Ciphertext policy attribute-based encryption," in *Proc. of 2007 IEEE Symposium on Security and Privacy*, pp. 321-334, May 20-23, 2007. Article(CrossRefLink)

[4] H. Li, D. Liu and K. Alharbi, "Enabling fine-grained access control with efficient attribute revocation and policy updating in smart grid," *KSII Transactions on Internet and Information Systems*, vol. 9, no. 4, pp. 1404-1423, 2015. Article(CrossRefLink)

[5] L. Zhang and Y. Hu, "New constructions of hierarchical attribute-based encryption for fine-grained access control in cloud computing," *KSII Transactions on Internet and Information Systems*, vol. 7, no. 5, pp. 1343–1356, 2013. Article(CrossRefLink)

[6] X. Gao, C. Ma, P. Zhao and L. Xiao, "Fine-grained access control scheme for social network with transitivity," *Journal of Computer Applications*, vol. 33, no. 1, pp. 8-11, 2013. Article(CrossRefLink)

[7] C. Li, X. Yang, S. Zhou, Y. Li and C. Wang, "A fined-grained cryptograph access control scheme for social network," *Computer Engineering*, vol. 41, no. 2, pp. 117–121, 2015. Article(CrossRefLink)

[8] L. Cheung and C. Newport, "Provably secure ciphertext policy ABE," in *Proc. of 14th ACM Conference on Computer and Communications Security*, pp. 456–465, October 29 - November 2, 2007. Article(CrossRefLink)

[9] J. Hur, "Improving security and efficiency in attribute-based data sharing," *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, no. 10, pp. 2271–2282, 2013. Article(CrossRefLink)

[10] P. Zhang, Z. Chen, K. Liang, S. Wang and T. Wang, "A cloud-based access control scheme with user revocation and attribute update," in *Proc. of 21st Australasian Conference on Information Security and Privacy*, pp. 1–16, July 4-6, 2016. Article(CrossRefLink)

[11] X. Liu, J. Ma, J. Xiong, Q. Li, T. Zhang and H. Zhu, "Threshold attribute-based encryption with attribute hierarchy forlattices in the standard model," *IET Information Security*, vol. 8, no. 4, pp. 217–223, 2014. Article(CrossRefLink)

[12] W. Zhu, J. Yu, T. Wang, P. Zhang and W. Xie, "Efficient attribute-based encryption from R-LWE," *Chinese Journal of Electronics*, vol. 23, no. 4, pp. 778–782, 2014.

[13] T.S. Fun and A. Samsudin, "Lattice ciphertext-policy attribute-based encryption from ring-LWE," in *Proc. of 2015 International Symposium on Technology Management and Emerging Technologies (ISTMET)*, pp. 258–262, Augest 25-27, 2015. Article(CrossRefLink)

[14] A. Lewko and B. Waters, "Unbounded HIBE and attribute based encryption," in *Proc. of 30th Annual International Conference on Theory and Applications of Cryptographic Techniques: Advances in Cryptology*, pp. 547–567, May 15-19, 2011. Article(CrossRefLink)

[15] T. Okamoto and K. Takashima, "Fully secure unbounded inner-product and attribute-based encryption," in *Proc. of 18th International Conference on the Theory and Application of Cryptology and Information Security*, pp. 349–366, December 2-6, 2012. Article(CrossRefLink)

[16] Y. Rouselakis and B. Waters, "Practical constructions and new proof methods for large universe attribute-based encryption," in *Proc. of 20th ACM Conference on Computer and Communications Security*, pp. 463–474, November 4-8, 2013. Article(CrossRefLink)

[17] Q. Li, J. Ma, R. Li, J. Xiong and X. Liu, "Provably secure unbounded multi-authority ciphertext-policy attribute-based encryption," *Security and Communication Networks*, vol. 8, pp. 4098–4109, 2015. Article(CrossRefLink)
[18] S. Agrawal, X. Boyen, V. Vaikuntanathan, P. Voulgaris and H. Wee, "Functional encryption for threshold functions (or fuzzy IBE) from lattices," in *Proc. of 15th International Conference on Practice and Theory in Public Key Cryptography*, pp. 280–297, May 21-23, 2012. Article(CrossRefLink)
[19] J. Zhang, Z. Zhang and A. Ge, "Ciphertext policy attribute-based encryption from lattices," in *Proc. of 7th ACM Symposium on Information, Computer and Communications Security*, pp. 16–17, May 2-4, 2012. Article(CrossRefLink)
[20] V. Lyubashevsky, C. Peikert and O. Regev, "On ideal lattices and learning with errors over rings," in *Proc. of 29th Annual International Conference on Theory and Applications of Cryptographic Techniques*, pp. 1–23, May 30-June 3, 2010. Article(CrossRefLink)
[21] D. Stehl´e and R. Steinfeld, "Making NTRU as secure as worst-case problems over ideal lattices," in *Proc. of 30th Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pp.27–47, May 15-19, 2011. Article(CrossRefLink)
[22] X. Yang, L.Wu, M. Zhang and X. Chen, "An efficient CCA-secure cryptosystem over ideal lattices from identity-based encryption," *Computers & Mathematics with Application*, vol. 65, no. 9, pp. 1254–1263, 2013. Article(CrossRefLink)
[23] S. Agrawal, D. Boneh and B. Xavier, "Efficient lattice (H)IBE in the standard model," in *Proc. of 29th Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 553–572, May 30-June 3, 2010. Article(CrossRefLink)

Zehong Chen received her M.S. degree in applied mathematics from Minnan Normal University in 2014. Now she is currently pursuing the Ph.D. degree with the College of Information Engineering, Shenzhen University, China. Her research interests include lattice-based cryptography and information security

Peng Zhang received her M.S. degree and Ph.D. degree in signal and information processing from Shenzhen University in 2008 and 2011, respectively. Now she is a lecturer of College of Information Engineering, Shenzhen University. Her current research interests include cryptography, information and network security.

Fangguo Zhang received his Ph.D. degree from the School of Communication Engineering, Xidian University in 2001. He is currently a Professor at the School of Information Science and Technology of Sun Yat-sen University, China. He is the co-director of Guangdong Key Laboratory of Information Security Technology. His research mainly focuses on cryptography and its applications. Specific interests are elliptic curve cryptography, secure multiparty computation, anonymmity and privacy.

Jiwu Huang received the B.S. degree from Xidian University, Xi'an, China, in 1982; the M.S. degree from Tsinghua University, Beijing, China, in 1987; and the Ph.D. degree from Institute of Automation, Chinese Academy of Sciences, Beijing, in 1998. He is a Professor with College of Information Engineering, Shenzhen University, Shenzhen, China, and also with Shenzhen Key Laboratory of Media Security. His research interests include multimedia forensics and security