Directly revocable ciphertext-policy attribute-based encryption from lattices

Fei Meng

School of Mathematics, Shandong University, Jinan Shandong 250100, China
menegfei_sdu@163.com

Abstract. Attribute-based encryption (ABE) is a promising type of cryptosystem achieving fine-grained access control on encrypted data. Revocable attribute-based encryption (RABE) is an extension of ABE that provides revocation mechanisms when user's attributes change, key exposure, and so on. In this paper, we propose two directly revocable ciphertext-policy attribute-based encryption (DR-ABE) schemes from lattices, which support flexible threshold access policies on multi-valued attributes, achieving user-level and attribute-level user revocation, respectively. Specifically, the revocation list is defined and embedded into the ciphertext by the message sender to revoke a user in the user-level revocable scheme or revoke some attributes of a certain user in the attribute-level revocable scheme. We also discuss how to outsource decryption and reduce the workload for the end user. Our schemes are proved to be secure in the standard model, assuming the hardness of the learning with errors (LWE) problem.

Keywords: Access control · Attribute-based encryption · Direct revocation · Decryption outsourcing · Lattice-based cryptosystem.

1 Introduction

Attribute-based encryption, first proposed by Sahai and Waters [29], is a cryptographic primitive providing encryption mechanism with fine-grained access control. In 2006, Goyal et al. [16] extended the idea of ABE and classified ABE as key-policy ABE (KP-ABE) [7, 17] and ciphertext-policy ABE (CP-ABE) [8, 34]. In a KP-ABE scheme, the private key of a user is associated with an access policy, while the ciphertext is associated with a set of attributes. On the contrary, in a CP-ABE scheme, the private key of a user is associated with a set of attributes, and the ciphertext is associated with an access policy. Generally, CP-ABE is more flexible than KP-ABE, since the former allows users to set their access policies when encrypting messages. In order to resist against quantum attacks, many attribute-based encryption schemes from lattices [37, 31, 5, 11, 10] have emerged.

In practical applications, one of the challenges of ABE is to revoke users or their attributes to change users' access rights when user's attributes change, key exposure, and so on. The revocation mechanism in ABE can be roughly divided

into two types: user-level user revocation [18, 35, 32] and attribute-level user revocation [20]. In user-level user revocation, when a user leaves the system, he/she should be revoked and can't decrypt any ciphertext. In attribute-level user revocation, when some attributes of a user are removed, he/she will lose the authorities corresponding to these attributes.

The methods for revocation can be divided into two types: indirect revocation [26, 28, 13] and direct revocation [6, 24, 21]. In indirect revocation schemes, the authority needs to master the revocation list, and issues key update for non-revoked users regularly. In addition, all non-revoked users need to communicate with the authority and update their decryption keys periodically as well. However, in direct revocation schemes, the revocation list is defined by the message sender , who "embeds" it into the ciphertext during encryption. Therefore, the authority does not need to generate and issue key update. In this paper, we only focus on direct revocation.

1.1 Motivation

Wang et al. [33] and Yang et al. [36] proposed indirectly revocable CP-ABE schemes from lattices. Both of their schemes have achieved attribute-level user revocation. However, Wang et al. [33] does not resist to collusion attacks, that is, two users who do not satisfy the access structure can successfully decrypt the ciphertext through cooperation. In Yang et al. [36], they built N user binary trees $\{\mathsf{BT}_i\}_{i\in[1,N]}$, where N is the maximum number of users. Each binary tree has M leaf nodes and the each attribute is assigned to a leaf node in the binary tree, where M is the number of attributes in the system. To revoke r' attributes of a user, the authority actually needs to issue $M - r'$ (rather than $r' \log \frac{M}{r'}$ as they claimed) associated key update in the key updating phase, since each attribute is assigned a different secret-shared key. In other words, they didn't actually take advantage of the binary-tree data structure to reduce the burden of the authority during key updating phase as [26, 28, 13].

1.2 Our contributions

This paper proposes two directly revocable ciphertext-policy attribute-based encryption (DR-ABE) schemes from lattices. One achieves user-level user revocation, while the other achieves attribute-level user revocation. Both schemes support flexible threshold access policies on multi-valued attributes. The size of public key of our schemes can be reduced in the random oracle model. The main advantages of our DR-ABE schemes are as follows:

direct revocation: the revocation list is embedded into the ciphertext by the message sender ; the authority does not have to generate and issue key update; all non-revoked users do not need to communicate with the authority to update their decryption keys.

user-level and attribute-level revocation: We provide two DR-ABE schemes with user-level and attribute-level revocation respectively. We use different

techniques to construct these two schemes because the method of constructing user-level scheme cannot be directly extended to attribute-level scheme.

fine-grained access control: our schemes support flexible threshold access policies on multi-valued attributes.

collusion resistance: users in the system cannot combine their information together to illegitimately gain unauthorized data through collaboration.

resistant against quantum attacks: the security of our schemes are reduced to the learning with errors (LWE) problem.

decryption outsourced: most computational overhead of end user in our DR-ABE schemes can be outsourced to a third party (Section 7).

In Table 1, we compare our schemes with other lattice-based ABE and revocable ABE schemes.

Table 1. comparison with other schemes

	multi-valued	direct/indirect	collusion resistance	Security model	Dec Outsourced
[37]	yes	—	yes	reasonable	no
[33]	yes	indirect	no	unreasonable	no
[36]	no	indirect	yes	unreasonable	no
Ours1	yes	direct	yes	reasonable	yes
Ours2	yes	direct	yes	reasonable	yes

Note that Zhang et al. [37] did not consider revocation. Wang et al. [33] and Yang et al. [36] achieve attribute-level user revocation. In the security model of [33], after submitting the challenge access structure $\mathbb{A}^*$ and challenge revocation list $\mathsf{RL}^* = \{\mathsf{RL}_i^*\}$, the adversary can only issue key generation queries $(\mathsf{id}, S = \{att_i\}_{i\in I})$ under the restriction $S \not\models \mathbb{A}^*$, while in [36], there is a stricter restriction $att_i \notin \mathbb{A}^*$. However, these restrictions are unreasonable. Because the private key of the key generation query (id, S) should be given to the adversary as long as the non-revoked attribute set $S_{\mathsf{id},\mathsf{RL}^*} = \{att_i \in S \mid \mathsf{id} \notin \mathsf{RL}_i^*, i \in I\}$ does not satisfies $\mathbb{A}^*$, which is the case in our security model for DR-ABE with attribute-level revocation. In other words, Wang et al. [33] and Yang et al. [36] didn't take into account all the key queries that an adversary could issue. While both of our schemes have considered all the situations of the key generation queries from the adversary. In Section 7, we discuss how to outsource most computational overhead of end user to an honest-but-curious third party.

2 Preliminaries

For notational convenience, we sometimes regard a matrix as simply a set of its column vectors. For a matrix $\mathbf{T}$, let $\|\mathbf{T}\|$ denote the L_2 length of its longest column, i.e., $\|\mathbf{T}\| := \max_i \|\mathbf{t}_i\|$; let $s_1(\mathbf{T})$ denote the largest singular value of $\mathbf{T}$, i.e., $s_1(\mathbf{T}) := \sup_{\|\mathbf{u}\|=1} \|\mathbf{T}\mathbf{u}\|$. Further, if the columns of $\mathbf{T} = \{\mathbf{t}_1, \cdots, \mathbf{t}_k\}$ are

linearly independent, let $\widetilde{\mathbf{T}} := \{\tilde{\mathbf{t}}_1, \cdots, \tilde{\mathbf{t}}_k\}$ denote the Gram-Schmidt orthogonalization of vectors $\mathbf{t}_1, \cdots, \mathbf{t}_k$ taken in that order. For two matrices $\mathbf{X} \in \mathbb{R}^{n \times m_1}$ and $\mathbf{Y} \in \mathbb{R}^{n \times m_2}$, let $(\mathbf{X} \| \mathbf{Y}) \in \mathbb{R}^{n \times (m_1 + m_2)}$ denote the concatenation of the columns of $\mathbf{X}$ followed by the columns of $\mathbf{Y}$. For two matrices $\mathbf{X} \in \mathbb{R}^{n_1 \times m}$ and $\mathbf{Y} \in \mathbb{R}^{n_2 \times m}$, let $(\mathbf{X}; \mathbf{Y}) \in \mathbb{R}^{(n_1 + n_2) \times m}$ denote the concatenation of the rows of $\mathbf{X}$ followed by the rows of $\mathbf{Y}$.

For non-negative integers $i < j$, Let $[i, j]$ denote the set $\{i, i+1, \ldots, j\}$. If S is an attribute set and $\mathbb{A}$ is an access structure, then $S \models \mathbb{A}$ means that S satisfies $\mathbb{A}$. If S is a finite set then $x \leftarrow S$ is the operation of choosing an element uniformly at random from S. For a probability distribution $\mathcal{D}$, $x \leftarrow \mathcal{D}$ denotes the operation of choosing an element according to $\mathcal{D}$. If γ is either an algorithm nor a set then $x \leftarrow \gamma$ is a simple assignment statement.

The natural security parameter throughout this paper is n. A function $f(n)$ is *negligible*, denoted as $\mathsf{negl}(n)$, if for every $c > 0$, there exists an n_c such that $f(n) < 1/n^c$ for all $n > n_c$. We say that a probability is overwhelming if it is $1 - \mathsf{negl}(n)$. An algorithm is probabilistic polynomial-time (PPT) computable if it is modeled as a probabilistic Turing machine whose running time is bounded by some polynomial function.

2.1 Directly revocable attribute-based encryption

A directly revocable ciphertext-policy attribute-based encryption (DR-ABE) scheme with user-level (*resp.* attribute-level) user revocation consists of the following four algorithms $\{\mathsf{Setup}, \mathsf{Keygen}, \mathsf{Enc}, \mathsf{Dec}\}$.

$\mathsf{Setup}(n, \mathcal{R}, N)$**:** This algorithm takes as input a security parameter n, a system attribute set $\mathcal{R}$ and a maximal number of users N in the system, returns a public key PK and a master secret key MSK.

$\mathsf{Keygen}(\mathsf{PK}, \mathsf{MSK}, \mathsf{id}, S)$**:** This algorithm takes as input a public key PK, a master secret key MSK, an identity id, and an attribute set $S = \{att_i\} \subseteq \mathcal{R}$ for the user with identity id, returns a private key $\mathsf{sk}_{S,\mathsf{id}}$.

$\mathsf{Enc}(\mathsf{PK}, \mathbb{A}, \mathsf{RL}, M)$**:** This algorithm takes as input a public key PK, an access structure $\mathbb{A} = (W = \{att_j\}_{j \in J}, t)$, a revocation list RL (*resp.* a family of attribute revocation lists $\mathsf{RL} = \{\mathsf{RL}_j\}_{j \in J}$, where RL_j consisting of identities whose j-th attribute is revoked), and a message M, returns a ciphertext C.

$\mathsf{Dec}(\mathsf{PK}, \mathsf{sk}_{S,\mathsf{id}}, C)$**:** This algorithm takes as input a public key PK, a private key $\mathsf{sk}_{S,\mathsf{id}}$ of identity id with attribute set $S = \{att_i\}$ and a ciphertext C encrypted under access structure $\mathbb{A}$ and RL, it first checks whether $S \models \mathbb{A}$ and $\mathsf{id} \notin \mathsf{RL}$ (*resp.* whether the set of non-revoked attributes of the identity id, $S_{\mathsf{id},\mathsf{RL}} = \{att_i \in S \mid \mathsf{id} \notin \mathsf{RL}_i\} \models \mathbb{A}$). If not, the algorithm returns a special symbol $\perp$ indicating decryption failure. Otherwise, it returns a message M.

Note that for DR-ABE scheme with attribute-level revocation, it is reasonable that the message sender only need to consider attribute revocation lists associated with his/her access structure.

2.2 Security model for DR-ABE

We now describe the selective security model for the DR-ABE scheme with user-level (*resp.* attribute-level) user revocation. The security model is described by the following game between a challenger $\mathcal{C}$ and an adversary $\mathcal{A}$.

Init. The adversary $\mathcal{A}$ chooses an access structure $\mathbb{A}^* = (W^*, t^*)$ with $W^* = \{att_j^*\}_{j\in J^*}$ and a revocation list RL^* (*resp.* a family of attribute revocation lists $\mathsf{RL}^* = \{\mathsf{RL}_j^*\}_{j\in J^*}$), and submits them to the challenger $\mathcal{C}$.

Setup. $\mathcal{C}$ runs the Setup algorithm, gives the public key PK to $\mathcal{A}$ and keeps the master secret key MSK private.

Phase 1. $\mathcal{A}$ can adaptively make a number of key generation queries (id, S), where $S = \{att_i\}_{i\in I}$. The restriction is that if $S \models W^*$, then $\mathsf{id} \in \mathsf{RL}^*$ (*resp.* the non-revoked attribute set $S_{\mathsf{id},\mathsf{RL}^*} = \{att_i \in S \mid \mathsf{id} \notin \mathsf{RL}_i^*, i \in I\}$ does not satisfies $\mathbb{A}^*$).

Challenge. $\mathcal{A}$ submits two equal length messages $M_0 \neq M_1$. The challenger $\mathcal{C}$ flips a random coin $b \in \{0,1\}$, computes $C^* = \mathsf{Enc}(\mathsf{PK}, \mathbb{A}^*, \mathsf{RL}^*, M_b)$, and gives C^* to $\mathcal{A}$.

Phase 2. It is the same as in **Phase 1**.

Guess. $\mathcal{A}$ output a guess $b' \in \{0,1\}$ for b.

The advantage of adversary $\mathcal{A}$ in the above game is defined as

$$\mathsf{Adv}_{\mathcal{A}}(\lambda) = \mid \Pr[b = b'] - 1/2 \mid . \tag{1}$$

Definition 1. *A directly revocable ciphertext-policy attribute-based encryption scheme is secure if the advantage* $\mathsf{Adv}_{\mathcal{A}}(\lambda)$ *is negligible in* λ *for all polynomial time adversary* $\mathcal{A}$.

2.3 Full Rank Difference Encoding (FRD)

In our construction and proof of security, we need an encoding function $H : \mathbb{Z}_q^n \to \mathbb{Z}_q^{n\times n}$ to map attributes in $\mathbb{Z}_q^n$ to matrices in $\mathbb{Z}_q^{n\times n}$.

Definition 2. *[1, 14] Let* q *be a prime and* n *a positive integer. We say that a function* $H : \mathbb{Z}_q^n \to \mathbb{Z}_q^{n\times n}$ *is an encoding with full-rank difference (FRD) if:*

1. *for all distinct* $\mathbf{x}, \mathbf{y} \in \mathbb{Z}_q^n$*, the matrix* $H(\mathbf{x}) - H(\mathbf{y})$ *is full rank.*
2. $\mathsf{G}_{\mathrm{FRD}}$ *is computable in polynomial time.*

2.4 The binary-tree data structure

Our construction makes use of the binary-tree data structure, as with [13, 9, 30, 23, 19]. This structure uses a node selection algorithm called $\mathsf{KUNodes}$. In the algorithm, we use the following notations: BT denotes a binary-tree. root denotes the root node of BT. θ denotes a node in the binary tree and ν emphasizes that the node θ is a leaf node. The set $\mathsf{Path}(\mathsf{BT}, \nu)$ stands for the collection of nodes

on the path from the leaf ν to the root (including ν and the root). If θ is a non-leaf node, then θ_ℓ, θ_r denote the left and right child of θ, respectively. The KUNodes algorithm takes as input a binary tree BT, a revocation list RL, and outputs a set of nodes Y, which is the smallest subset of nodes that contains an ancestor of all the leaf nodes corresponding to non-revoked indexes. The description of the KUNodes algorithm is as follows:

$\mathsf{KUNodes}(\mathsf{BT}, \mathsf{RL})$:
$X, Y \leftarrow \emptyset$; $\forall \nu \in \mathsf{RL}$,add $\mathsf{Path}(\mathsf{BT}, \nu)$ to X;
$\forall \theta \in X$: if $\theta_\ell \notin X$ then add θ_ℓ to Y, if $\theta_r \notin X$ then add θ_r to Y;
If $Y = \emptyset$ then add root to Y; Return Y.

3 Background on lattices

Let $\mathbf{B} = \{\mathbf{b}_1 \cdots \mathbf{b}_m\} \subset \mathbb{R}^m$ consists of m linearly independent vectors. The m-dimensional full-rank lattice Λ generated by the *basis* $\mathbf{B}$ is the set $\Lambda = \mathcal{L}(\mathbf{B}) := \{\sum_{i=1}^{m} x_i \mathbf{b}_i \mid x_i \in \mathbb{Z}\}$. For any positive integers n, m and $q \geq 2$, a matrix $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ and a vector $\mathbf{u} \in \mathbb{Z}_q^n$, we define $\mathcal{L}_q^{\perp}(\mathbf{A}) := \{\mathbf{z} \in \mathbb{Z}^m : \mathbf{A} \cdot \mathbf{z} = \mathbf{0}_n \mod q\}$ and $\mathcal{L}_q^{\mathbf{u}}(\mathbf{A}) := \{\mathbf{z} \in \mathbb{Z}^m : \mathbf{A} \cdot \mathbf{z} = \mathbf{u} \mod q\}$.

3.1 Discrete Gaussian

Let Λ be an m-dimensional lattice. For any vector $\mathbf{c} \in \mathbb{R}^m$ and any parameter $\sigma \in \mathbb{R}_{>0}$, define $\rho_{\sigma,\mathbf{c}}(\mathbf{x}) = \exp(-\pi \frac{\|\mathbf{x}-\mathbf{c}\|^2}{\sigma^2})$ and $\rho_{\sigma,\mathbf{c}}(\Lambda) = \sum_{\mathbf{x} \in \Lambda} \rho_{\sigma,\mathbf{c}}(\mathbf{x})$. The *discrete Gaussian distribution* over Λ with center $\mathbf{c}$ and Gaussian parameter σ is $\mathcal{D}_{\Lambda,\sigma,\mathbf{c}} = \frac{\rho_{\sigma,\mathbf{c}}(\mathbf{y})}{\rho_{\sigma,\mathbf{c}}(\Lambda)}$ for $\forall \mathbf{y} \in \Lambda$. If $\mathbf{c} = \mathbf{0}$, we conveniently use ρ_σ and $\mathcal{D}_{\Lambda,\sigma}$. In the following, we summarize some basic properties of the discrete Gaussian distribution.

Lemma 1. *[15] Let n, m, q be positive integers with $m > n$, $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ be a matrix, $\mathbf{u} \in \mathbb{Z}_q^n$ be a vector, $\mathbf{T}_\mathbf{A}$ be a basis for $\Lambda = \mathcal{L}_q^{\mathbf{u}}(\mathbf{A})$ and $\sigma \geq \|\widetilde{\mathbf{T}_\mathbf{A}}\| \cdot \omega(\sqrt{\log m})$. Then $\Pr[\|\mathbf{x}\| > \sigma\sqrt{m} : \mathbf{x} \leftarrow \mathcal{D}_{\Lambda,\sigma}] \leq \mathsf{negl}(n)$.*

Lemma 2. *[15] Let $n, m, q > 0$ be positive integers with $m \geq 2n\lceil \log q \rceil$ and q a prime. Let σ be any positive real such that $\sigma \geq \omega(\sqrt{\log m})$. Then for $\mathbf{A} \leftarrow \mathbb{Z}_q^{n \times m}$ and $\mathbf{e} \leftarrow \mathcal{D}_{\mathbb{Z}^m,\sigma}$, the distribution of $\mathbf{u} = \mathbf{A}\mathbf{e} \mod q$ is statistically close to uniform over $\mathbb{Z}_q^n$. Furthermore, for a fixed $\mathbf{u} \in \mathbb{Z}_q^n$, the conditional distribution of $\mathbf{e} \leftarrow \mathcal{D}_{\mathbb{Z}^m,\sigma}$, given $\mathbf{A}\mathbf{e} = \mathbf{u} \mod q$ for a uniformly random $\mathbf{A}$ in $\mathbb{Z}_q^{n \times m}$ is $\mathcal{D}_{\Lambda_q^{\mathbf{u}}(\mathbf{A}),\sigma}$ with all but negligible probability.*

3.2 Trapdoors for lattices

We review two trapdoor generation algorithms in the following lemma. The first algorithm generates a matrix $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ that is statistically close to uniform, together with a short trapdoor basis for the associated lattice $\Lambda_q^{\perp}(\mathbf{A})$. The second algorithm generates a basis for the lattice $\Lambda_q^{\perp}(\mathbf{G})$, where $\mathbf{G}$ is what they call the primitive matrix.

Lemma 3. *[4, 22, 3] Let $n, m, q > 0$ be positive integers with $m \geq 2n\lceil \log q \rceil$ and q a prime. Then, we have:*

- *[4, 22, 3] a PPT algorithm* $\mathsf{TrapGen}$ *that outputs a pair $(\mathbf{A}, \mathbf{T_A}) \in \mathbb{Z}_q^{n\times m} \times \mathbb{Z}^{m\times m}$ such that $\mathbf{A}$ is full rank and statistically close to uniform and $\mathbf{T_A}$ is a basis for $\Lambda_q^{\perp}(\mathbf{A})$ satisfying $\|\widetilde{\mathbf{T_A}}\| \leq O(\sqrt{n \log q})$.*
- *[22] a fixed full rank matrix $\mathbf{G} \in \mathbb{Z}_q^{n\times m}$ such that the lattice $\Lambda_q^{\perp}(\mathbf{G})$ has a publicly known basis $\mathbf{T_G} \in \mathbb{Z}^{m\times m}$ with $\|\widetilde{\mathbf{T_G}}\| \leq \sqrt{5}$.*

3.3 Sampling algorithms

The following $\mathsf{SampleLeft}$ [12, 1] and $\mathsf{SampleRight}$ [1] algorithms will be used to sample short vectors in our construction and in the simulation, respectively.

Lemma 4. *Let integers $q > 2$ and $m > n$. There is an efficient PPT algorithm $\mathsf{SampleLeft}(\mathbf{A}, \mathbf{B}, \mathbf{u}, \mathbf{T_A}, \sigma)$ takes as input a full rank matrix $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$, a matrix $\mathbf{B} \in \mathbb{Z}_q^{n\times \bar{m}}$, a vector $\mathbf{u} \in \mathbb{Z}_q^n$, a basis $\mathbf{T_A} \in \mathbb{Z}^{m\times m}$ of $\Lambda_q^{\perp}(\mathbf{A})$, and a Gaussian parameter $\sigma > \|\widetilde{\mathbf{T_A}}\| \cdot \omega(\sqrt{\log(m+\bar{m})})$, outputs a vector $\mathbf{e} \in \mathbb{Z}^{m+\bar{m}}$ distributed statistically close to $\mathcal{D}_{\Lambda_q^{\mathbf{u}}([\mathbf{A}\|\mathbf{B}]),\sigma}$.*

Lemma 5. *Let integers $q > 2$ and $m > n$. There is an efficient PPT algorithm $\mathsf{SampleRight}(\mathbf{A}, \mathbf{B}, \mathbf{R}, \mathbf{u}, \mathbf{T_B}, \sigma)$ takes as input matrices $\mathbf{A}, \mathbf{B} \in \mathbb{Z}_q^{n\times m}$, where $\mathbf{B}$ is full rank, a uniform random matrix $\mathbf{R} \in \mathbb{Z}^{m\times m}$, a vector $\mathbf{u} \in \mathbb{Z}_q^n$, a basis $\mathbf{T_B}$ of $\Lambda_q^{\perp}(\mathbf{B})$, a Gaussian parameter $\sigma > \|\widetilde{\mathbf{T_B}}\| \cdot s_1(\mathbf{R}) \cdot \omega(\sqrt{\log m})$, outputs a vector $\mathbf{e} \in \mathbb{Z}^{2m}$ distributed statistically close to $\mathcal{D}_{\Lambda_q^{\mathbf{u}}([\mathbf{A}\|\mathbf{AR}+\mathbf{B}]),\sigma}$.*

3.4 Useful facts.

To prove correctness and security of our construction, we need more lemmas from [1] as follows.

Lemma 6. *Let $\mathbf{R}$ be a $m \times m$ matrix chosen at random from $\{-1, 1\}^{m\times m}$, then there exists a universal constant C such that $\Pr[s_1(\mathbf{R}) > C\sqrt{m}] < e^{-m}$.*

Lemma 7. *Suppose that q is a prime and that $m > (n+1)\log q + \omega(\log n)$. Let $\mathbf{A}, \mathbf{B}$ be matrices chosen uniformly in $\mathbb{Z}_q^{n\times m}$ and let $\mathbf{R}$ be an $m \times m$ matrix chosen uniformly in $\{-1, 1\}^{m\times m} \mod q$. Then, for all vectors $\mathbf{w}$ in $\mathbb{Z}_q^m$, the distribution $(\mathbf{A}, \mathbf{AR}, \mathbf{R}^{\top}\mathbf{w})$ is statistically close to the distribution $(\mathbf{A}, \mathbf{B}, \mathbf{R}^{\top}\mathbf{w})$.*

3.5 The LWE hardness assumption

Security of our construction reduces to the learning with errors (LWE) problem defined by Regev [27].

Definition 3. *Consider a prime q, a positive integer n, and a distribution χ over $\mathbb{Z}_q$, all public. An $(\mathbb{Z}_q, n, \chi)$-LWE problem instance consists of access to an unspecified challenge oracle $\mathcal{O}$, being, either, a noisy pseudo-random sampler $\mathcal{O}_{\mathbf{s}}$ carrying some constant random secret key $\mathbf{s} \in \mathbb{Z}_q^n$, or, a truly random sampler $\mathcal{O}_{\$}$, whose behaviors are respectively as follows:*

$\mathcal{O}_{\mathbf{s}}$: *outputs samples in $\mathbb{Z}_q^n \times \mathbb{Z}_q$ of the form $(\mathbf{u}_i, v_i) = (\mathbf{u}_i, \mathbf{u}_i^\top \mathbf{s} + x_i)$, where, $\mathbf{s} \in \mathbb{Z}_q^n$ is a uniformly distributed persistent value invariant across invocations, $x_i \in \mathbb{Z}_q$ is a fresh sample from χ, and $\mathbf{u}_i$ is uniform in $\mathbb{Z}_q^n$.*

$\mathcal{O}_{\$}$: *outputs truly uniform random samples from $\mathbb{Z}_q^n \times \mathbb{Z}_q$.*

The $(\mathbb{Z}_q, n, \chi)$-LWE problem allows repeated queries to the challenge $\mathcal{O}$. We say that an algorithm $\mathcal{A}$ decides the $(\mathbb{Z}_q, n, \chi)$-LWE problem if $|\Pr[\mathcal{A}^{\mathcal{O}_{\mathbf{s}}} = 1] - \Pr[\mathcal{A}^{\mathcal{O}_{\$}} = 1]|$ is non-negligible for a random $\mathbf{s} \in \mathbb{Z}_q^n$.

Regev [27] and Peikert [25] showed that for certain noise distribution χ, denoted $\bar{\Psi}_\alpha$, the LWE problem is hard.

Definition 4. *Consider a real number $\alpha = \alpha(n) \in (0, 1)$ and a prime q. Let $\mathbb{T} := \mathbb{R}/\mathbb{Z}$ be the group of reals $[0, 1)$ with addition modulo 1. Define by Ψ_α the distribution over $\mathbb{T}$ of a normal variable with mean 0 and standard deviation $\alpha/\sqrt{2\pi}$, reduced modulo 1, i.e.,*

$$\forall r \in [0, 1), \Psi_\alpha(r) := \sum_{k=-\infty}^{\infty} \frac{1}{\alpha} \cdot \exp(-\pi(\frac{r-k}{\alpha})^2).$$

We denote by $\bar{\Psi}_\alpha$ the discrete distribution *over $\mathbb{Z}_q$ of the random variable $\lfloor q \cdot X_{\Psi_\alpha} \rceil \mod q$, where the random variable $X_{\Psi_\alpha} \in \mathbb{T}$ has distribution Ψ_α.*

Lemma 8. *Consider $\alpha = \alpha(n) \in (0, 1)$ and a prime $q = q(n)$ such that $\alpha q > 2\sqrt{n}$. If there exists an efficient (possibly quantum) algorithm solves $(\mathbb{Z}_q, n, \bar{\Psi}_\alpha)$-LWE problem, then there exists an efficient quantum algorithm for approximating SIVP in the ℓ_2 norm, in the worst case, to within $\tilde{O}(n/\alpha)$ factors.*

The following lemma about the distribution $\bar{\Psi}_\alpha$ will be used to analyze the correctness of our constructions in Section 4 and 5.

Lemma 9. *[1] Let $\mathbf{e}$ be some vector in $\mathbb{Z}^m$ and let $\mathbf{x} \leftarrow \bar{\Psi}_\alpha^m$. Then the quantity $|\mathbf{e}^\top \mathbf{x}|$ treated as an integer in $[0, q-1]$ satisfies*

$$|\mathbf{e}^\top \mathbf{x}| \leq \|\mathbf{e}\| q\alpha\omega(\sqrt{\log m}) + \|\mathbf{e}\|\sqrt{m}/2$$

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

4 DR-ABE with user-level revocation

In this section, we propose a DR-ABE scheme from lattices, which supports user-level revocation and flexible threshold access policies on multi-valued attributes. The main ideas behind our construction can be described as follows. We assign identity id to a leaf node ν_{id} in the binary tree BT. Then we store the attribute set S of id in every node $\theta \in \mathsf{path}(\mathsf{BT}, \nu_{id})$: For each θ, the random vector $\mathbf{u}$ in the public key is secret-shared into vectors $\{\hat{\mathbf{u}}_{\theta,i}\}$, where $\hat{\mathbf{u}}_{\theta,i}$ is associated with attribute att_i. If $\mathsf{id} \notin \mathsf{RL}$ and $S \models \mathbb{A}$, then there exists a node $\theta^* \in \mathsf{path}(\mathsf{BT}, \nu_{id}) \cap \mathsf{KUNodes}(\mathsf{BT}, \mathsf{RL})$, and $\mathbf{u}$ can be recovered using $\{\hat{\mathbf{u}}_{\theta^*,i}\}$.

For convenience, it is assumed that there are ℓ attributes in our system, and the i-th attribute is associated with a value space $\mathcal{R}_i \subseteq \mathbb{Z}_q^n \setminus \{\mathbf{0}\}$. Let $\mathcal{R} = \mathcal{R}_1 \times \cdots \times \mathcal{R}_\ell$ denote the attribute space. We also define d default attributes $\{\ell+1, \cdots, \ell+d\}$. Let $\mathcal{I} = \{1, \ldots, \ell+d\}$ and let $\mathcal{I}_1 = \{1, \ldots, \ell\}$, $\mathcal{I}_2 = \{\ell+1, \ldots, \ell+d\}$, $D = ((\ell+d)!)^2$.

$\mathsf{Setup}(n, \mathcal{R}, N)$**:** On input a security parameter n, a system attribute set $\mathcal{R} = \mathcal{R}_1 \times \cdots \times \mathcal{R}_\ell$ and a maximal number of users N in the system, this algorithm sets the primitive matrix $\mathbf{G}$ (with public trapdoor $\mathbf{T_G}$, see Lemma 3) and the parameters q, m, α, σ as specified in Section 4.3. Then it performs as follows:

1. Run $(\mathbf{A}, \mathbf{T_A}) \leftarrow \mathsf{TrapGen}(n, m, q)$.
2. Choose $\mathbf{B}_i \leftarrow \mathbb{Z}_q^{n \times m}$ for $i \in \mathcal{I}$.
3. Choose $\mathbf{u} \leftarrow \mathbb{Z}_q^n$.
4. Choose a full-rank difference map $H : \mathbb{Z}_q^n \to \mathbb{Z}_q^{n \times n}$.
5. Build a binary tree BT with N leaf nodes. For each node $\theta \in \mathsf{BT}$, choose "identifier" $\mathbf{D}_\theta \leftarrow \mathbb{Z}_q^{n \times m}$.
6. Return $\mathsf{PK} = \{\mathbf{A}, \{\mathbf{B}_i\}_{i \in \mathcal{I}}, \mathbf{u}, H, \mathsf{BT}\}$ and $\mathsf{MSK} = \mathbf{T_A}$.

$\mathsf{Keygen}(\mathsf{PK}, \mathsf{MSK}, \mathsf{id}, S)$**:** On input the public key PK, the master secret key MSK, an identity id, and the attribute set $S = \{att_i\}_{i \in I}$ of id, where $I \subseteq \mathcal{I}_1$ and $att_i \in \mathcal{R}_i$, it goes as follows:

1. Pick an unassigned leaf node ν_{id} from BT and store id in that node. For each $\theta \in \mathsf{path}(\mathsf{BT}, \nu_{id})$, randomly choose n degree d polynomials $p_{\theta,1}(x), \ldots, p_{\theta,n}(x) \in \mathbb{Z}_q[x]$, such that $\mathbf{u} = (p_{\theta,1}(0), \ldots, p_{\theta,n}(0))^\top$. For each $i \in I \cup \mathcal{I}_2$, let $\hat{\mathbf{u}}_{\theta,i} = (p_{\theta,1}(i), \ldots, p_{\theta,n}(i))^\top$.
2. For each $\theta \in \mathsf{path}(\mathsf{BT}, \nu_{id})$, sample

$$\mathbf{e}_{\theta,i} \leftarrow \mathsf{SampleLeft}(\mathbf{A}, \mathbf{D}_\theta \| \mathbf{B}_i + H(att_i)\mathbf{G}, \hat{\mathbf{u}}_{\theta,i}, \mathbf{T_A}, \sigma)$$

for $i \in I$ and sample $\mathbf{e}_{\theta,i} \leftarrow \mathsf{SampleLeft}(\mathbf{A}, \mathbf{D}_\theta \| \mathbf{B}_i + \mathbf{G}, \hat{\mathbf{u}}_{\theta,i}, \mathbf{T_A}, \sigma)$ for $i \in \mathcal{I}_2$.

Let $\mathbf{E}_{\theta,i} = (\mathbf{A} \| \mathbf{D}_\theta \| \mathbf{B}_i + H(att_i)\mathbf{G})$ for $i \in I$ and $\mathbf{E}_{\theta,i} = (\mathbf{A} \| \mathbf{D}_\theta \| \mathbf{B}_i + \mathbf{G})$ for $i \in \mathcal{I}_2$, note that $\mathbf{E}_{\theta,i} \cdot \mathbf{e}_{\theta,i} = \hat{\mathbf{u}}_{\theta,i}$.

3. Return $\mathsf{sk}_{S,\mathsf{id}} = \left(\{\mathbf{e}_{\theta,i}\}_{\theta \in \mathsf{path}(\mathsf{BT}, \nu_{id}), i \in I \cup \mathcal{I}_2}\right)$ as the private key.

Note that for any $\theta \in \mathsf{path}(\mathsf{BT}, \nu_{id})$ and any subset $K \subseteq I \cup \mathcal{I}_2$ with $|K| = d+1$, we have $\mathbf{u} = \sum_{i \in K} L_i \cdot \hat{\mathbf{u}}_{\theta,i}$, where the Lagrange coefficient $L_i = \frac{\prod_{j \in K, j \neq i} -j}{\prod_{j \in K, j \neq i}(i-j)}$.

$\mathsf{Enc}(\mathsf{PK}, (W,t), \mathsf{RL}, M)$**:** On input a public key PK, an attribute set $W=\{att_j\}_{j\in J_1}$, an integer $1\le t\le \min(|W|,d)$, a revocation list RL consisting of revoked identities, and a message $M\in\{0,1\}$, it works as follows:

1. Choose $\mathbf{s}\leftarrow\mathbb{Z}_q^n$, compute $c_0=\mathbf{u}^\top\mathbf{s}+Dx_0+M\lfloor\frac{q}{2}\rfloor$ and $\mathbf{c}=\mathbf{A}^\top\mathbf{s}+D\mathbf{x}$, where $x_0\leftarrow\bar{\Psi}_\alpha$, $\mathbf{x}\leftarrow\bar{\Psi}_\alpha^m$.
2. For each $j\in J_1$, choose $\mathbf{R}_j\leftarrow\{-1,1\}^{m\times m}$, compute $\mathbf{c}_j=(\mathbf{B}_j+H(att_j)\mathbf{G})^\top\mathbf{s}+D\mathbf{R}_j^\top\mathbf{x}$.
3. Let $J_2=\{\ell+1,\ldots,\ell+d+1-t\}$ and for each $j\in J_2$, choose $\mathbf{R}_j\leftarrow\{-1,1\}^{m\times m}$, compute $\mathbf{c}_j=(\mathbf{B}_j+\mathbf{G})^\top\mathbf{s}+D\mathbf{R}_j^\top\mathbf{x}$.
4. For each $\theta\in\mathsf{KUNodes}(\mathsf{BT},\mathsf{RL})$, choose $\mathbf{R}_\theta\leftarrow\{-1,1\}^{m\times m}$, compute $\mathbf{c}_\theta=\mathbf{D}_\theta^\top\cdot\mathbf{s}+D\mathbf{R}_\theta^\top\cdot\mathbf{x}$.
5. Return $C=\big(c_0,\mathbf{c},\{\mathbf{c}_j\}_{j\in J_1\cup J_2},\{\mathbf{c}_\theta\}_{\theta\in\mathsf{KUNodes}(\mathsf{BT},\mathsf{RL})}\big)$ as the ciphertext.

$\mathsf{Dec}(\mathsf{PK},\mathsf{sk}_{S,\mathsf{id}},C)$**:** On input the public key PK, the private key $\mathsf{sk}_{S,\mathsf{id}}$ of identity id with attribute set $S=\{att_i\}_{i\in I}$, and a ciphertext C encrypted under access structure $(W=\{att_j\}_{j\in J_1},t)$ and revocation list RL.

1. If $|S\cap W|<t$ or $\mathsf{id}\in\mathsf{RL}$, return $\perp$;
2. Else, parse the private key $\mathsf{sk}_{S,\mathsf{id}}=\big(\{\mathbf{e}_{\theta,i}\}_{\theta\in\mathsf{path}(\mathsf{BT},\nu_{id}),i\in I\cup\mathcal{I}_2}\big)$ and $C=\big(c_0,\mathbf{c},\{\mathbf{c}_j\}_{j\in J_1\cup J_2},\{\mathbf{c}_\theta\}_{\theta\in\mathsf{KUNodes}(\mathsf{BT},\mathsf{RL})}\big)$. Since $\mathsf{id}\notin\mathsf{RL}$, there exists a $\theta\in\mathsf{path}(\mathsf{BT},\nu_{id})\cap\mathsf{KUNodes}(\mathsf{BT},\mathsf{RL})$. Let $S\cap W=\{att_i\}_{i\in K}$. Since $|K|\ge t$, there exists a set $K'\subseteq K\cup J_2$ with size $d+1$. For all $j\in K'$, compute $r_{\theta,j}=\mathbf{e}_{\theta,j}^\top(\mathbf{c};\mathbf{c}_\theta;\mathbf{c}_j)$ and $r_\theta=\sum_{j\in K'}L_jr_{\theta,j}$, where $L_j=\frac{\prod_{k\in K',k\neq j}-k}{\prod_{k\in K',k\neq j}(j-k)}$. Finally, compute $\hat{r}=c_0-r_\theta$. If $|\hat{r}-\lfloor\frac{q}{2}\rfloor|\le\lfloor\frac{q}{4}\rfloor$ in $\mathbb{Z}$, return 1, otherwise return 0.

4.1 Correctness

For $j\in K'\cap K$ and $\theta\in\mathsf{path}(\mathsf{BT},\nu_{\mathsf{id}})\cap\mathsf{KUNodes}(\mathsf{BT},\mathsf{RL})$, we have

$$\begin{aligned}(\mathbf{c};\mathbf{c}_\theta;\mathbf{c}_j)=\begin{bmatrix}\mathbf{c}\\ \mathbf{c}_\theta\\ \mathbf{c}_j\end{bmatrix}&=\begin{bmatrix}\mathbf{A}^\top\mathbf{s}+D\mathbf{x}\\ \mathbf{D}_\theta^\top\cdot\mathbf{s}+D\mathbf{R}_\theta^\top\cdot\mathbf{x}\\ (\mathbf{B}_j+H(att_j)\mathbf{G})^\top\mathbf{s}+D\mathbf{R}_j^\top\mathbf{x}\end{bmatrix}\\ &=(\mathbf{A}\|\mathbf{D}_\theta\|\mathbf{B}_j+H(att_j)\mathbf{G})^\top\mathbf{s}+D\begin{bmatrix}\mathbf{x}\\ \mathbf{R}_\theta^\top\mathbf{x}\\ \mathbf{R}_j^\top\mathbf{x}\end{bmatrix}.\end{aligned}$$

For $j\in K'\cap J_2$ and $\theta\in\mathsf{path}(\mathsf{BT},\nu_{\mathsf{id}})\cap\mathsf{KUNodes}(\mathsf{BT},\mathsf{RL})$, we have

$$(\mathbf{c};\mathbf{c}_\theta;\mathbf{c}_j)=\begin{bmatrix}\mathbf{c}\\ \mathbf{c}_\theta\\ \mathbf{c}_j\end{bmatrix}=\begin{bmatrix}\mathbf{A}^\top\mathbf{s}+D\mathbf{x}\\ \mathbf{D}_\theta^\top\cdot\mathbf{s}+D\mathbf{R}_\theta^\top\cdot\mathbf{x}\\ (\mathbf{B}_j+\mathbf{G})^\top\mathbf{s}+D\mathbf{R}_j^\top\mathbf{x}\end{bmatrix}=(\mathbf{A}\|\mathbf{D}_\theta\|\mathbf{B}_j+\mathbf{G})^\top\mathbf{s}+D\begin{bmatrix}\mathbf{x}\\ \mathbf{R}_\theta^\top\mathbf{x}\\ \mathbf{R}_j^\top\mathbf{x}\end{bmatrix}.$$

Denote $\mathbf{x}_{\theta,j}=(\mathbf{x};\mathbf{R}_\theta^\top\mathbf{x};\mathbf{R}_j^\top\mathbf{x})$, then $(\mathbf{c};\mathbf{c}_\theta;\mathbf{c}_j)=\mathbf{E}_{\theta,j}^\top\mathbf{s}+D\mathbf{x}_{\theta,j}$ for both cases. Thus, we have $r_{\theta,j}=\mathbf{e}_{\theta,j}^\top\cdot(\mathbf{c};\mathbf{c}_\theta;\mathbf{c}_j)=\mathbf{e}_{\theta,j}^\top\cdot(\mathbf{E}_{\theta,j}^\top\mathbf{s}+D\mathbf{x}_{\theta,j})=\hat{\mathbf{u}}_{\theta,j}^\top\mathbf{s}+Dy_{\theta,j}$, where $y_{\theta,j}=$

$\mathbf{e}_{\theta,j}^{\top} \cdot \mathbf{x}_{\theta,j}$. Hence, $r_\theta = \sum_{j \in K'} L_j r_{\theta,j} = \mathbf{u}^{\top}\mathbf{s} + y_\theta$, where $y_\theta = \sum_{j \in K'} DL_j y_{\theta,j}$. Finally, we have

$$\hat{r} = c_0 - r_\theta = Dx_0 - y_\theta + M \cdot \lfloor \frac{q}{2} \rfloor.$$

Now, we begin to bound $|Dx_0 - y_\theta|$. By Lemma 1 and 4, we have $\|\mathbf{e}_{\theta,j}\| \leq \sigma\sqrt{3m}$. Note that $\mathbf{e}_{\theta,j}^{\top} \cdot \mathbf{x}_{\theta,j} = \mathbf{e}_{\theta,j,0}^{\top} \cdot \mathbf{x} + \mathbf{e}_{\theta,j,1}^{\top} \cdot \mathbf{R}_\theta^{\top}\mathbf{x} + \mathbf{e}_{\theta,j,2}^{\top} \cdot \mathbf{R}_j^{\top}\mathbf{x}$, where $\mathbf{e}_{\theta,j}^{\top} = (\mathbf{e}_{\theta,j,0}^{\top}, \mathbf{e}_{\theta,j,1}^{\top}, \mathbf{e}_{\theta,j,2}^{\top})$. Since $\|\mathbf{e}_{\theta,j,0} + \mathbf{R}_\theta \mathbf{e}_{\theta,j,1} + \mathbf{R}_j \mathbf{e}_{\theta,j,2}\| \leq (s_1(\mathbf{R}_\theta) + s_1(\mathbf{R}_j) + 1) \cdot \sigma\sqrt{3m}$, by Lemma 9, we have $\mathbf{e}_{\theta,j}^{\top} \cdot \mathbf{x}_{\theta,j} \leq (s_1(\mathbf{R}_\theta) + s_1(\mathbf{R}_j) + 1) \cdot \sigma\sqrt{3m} \cdot (q\alpha\omega(\sqrt{\log m}) + \sqrt{m}/2)$. Applying Lemma 9 in [2], we have $DL_j \leq ((\ell + d)!)^4$. By Lemma 6, we have $s_1(\mathbf{R}_\theta) = O(\sqrt{m})$, $s_1(\mathbf{R}_j) = O(\sqrt{m})$. Thus, $|y_\theta| \leq (d + 1)((\ell + d)!)^4 \sigma O(m) \cdot (q\alpha\omega(\sqrt{\log m}) + \sqrt{m}/2)$. Therefore, we have $|Dx_0 - y_\theta| \leq ((\ell+d)!)^2(q\alpha\omega(\sqrt{\log m}) + 1/2) + |y_\theta| \leq \sigma q\alpha m(d+1)((\ell+d)!)^4\omega(\sqrt{\log m}) + \sigma(d+1)((\ell + d)!)^4 O(m^{3/2})$ by Lemma 9.

4.2 Security

In this section, we prove the security of our construction of DR-ABE scheme with user-level user revocation in the selective model in Definition 1.

Theorem 1. *For appropriate parameters n, m, q, σ, α, the above DR-ABE scheme with user-level user revocation is secure provided that the $(\mathbb{Z}_q, n, \bar{\Psi}_\alpha)$-LWE problem is hard.*

Proof. Suppose there exists a PPT adversary $\mathcal{A}$ breaks the security of our DR-ABE scheme with user-level user revocation with non-negligible probability, we can construct an algorithm $\mathcal{B}$ that solves the LWE problem with the same advantage.

Note that $\mathcal{B}$ has an oracle $\mathcal{O}(\cdot)$ and he want to determine whether it is a noisy pseudo-random sampler $\mathcal{O}_{\mathbf{s}^*}$ for some $\mathbf{s}^* \in \mathbb{Z}_q^n$ or a truly random sampler $\mathcal{O}_\$$. To this end, $\mathcal{B}$ proceeds as follows:

Init. $\mathcal{A}$ submits a challenge access structure $\mathbb{A}^* = (W^* = \{att_j^*\}_{j \in J_1^*}, t^*)$ and a challenge revocation list RL^* to $\mathcal{B}$, where $J_1^* \subseteq \mathcal{I}_1$ and $1 \leq t^* \leq \min(|W^*|, d)$. Let $J_2^* = \{\ell + 1, \ldots, \ell + d + 1 - t^*\}$ and let $J^* = J_1^* \cup J_2^*$.

Setup. After receiving $(W^* = \{att_j^*\}_{j \in J_1^*}, t^*)$ and RL^*, $\mathcal{B}$ samples $(\mathbf{u}, v_u) \in \mathbb{Z}_q^n \times \mathbb{Z}_q$ and $(\mathbf{A}, \mathbf{v}) \in \mathbb{Z}_q^{n \times m} \times \mathbb{Z}_q^m$ from $\mathcal{O}(\cdot)$, chooses an FRD map $H : \mathbb{Z}_q^n \to \mathbb{Z}_q^{n \times n}$, builds a binary tree BT with N leaf nodes.

- For each $j \in J_1^*$, $\mathcal{B}$ chooses $\mathbf{R}_j^* \leftarrow \{-1, 1\}^{m \times m}$, and computes $\mathbf{B}_j = \mathbf{A}\mathbf{R}_j^* - H(att_j^*)\mathbf{G}$.
- For each $j \in \mathcal{I}_1 \backslash J_1^*$, $\mathcal{B}$ chooses $\mathbf{R}_j^* \leftarrow \{-1, 1\}^{m \times m}$, and computes $\mathbf{B}_j = \mathbf{A}\mathbf{R}_j^* - H(\mathbf{0})\mathbf{G}$.
- For each $j \in J_2^*$, $\mathcal{B}$ chooses $\mathbf{R}_j^* \leftarrow \{-1, 1\}^{m \times m}$, computes $\mathbf{B}_j = \mathbf{A}\mathbf{R}_j^* - \mathbf{G}$.
- For each $j \in \mathcal{I}_2 \backslash J_2^*$, $\mathcal{B}$ chooses $\mathbf{R}_j^* \leftarrow \{-1, 1\}^{m \times m}$, computes $\mathbf{B}_j = \mathbf{A}\mathbf{R}_j^*$.
- For each $\theta \in \mathsf{BT}$, $\mathcal{B}$ chooses $\mathbf{R}_\theta^* \leftarrow \{-1, 1\}^{m \times m}$ and computes $\mathbf{D}_\theta = \mathbf{A}\mathbf{R}_\theta^*$ if $\theta \in \mathsf{KUNodes}(\mathsf{BT}, \mathsf{RL}^*)$ and $\mathbf{D}_\theta = \mathbf{A}\mathbf{R}_\theta^* + \mathbf{G}$ otherwise.

Finally, $\mathcal{B}$ sends the public key $\mathsf{PK} = \{\mathbf{A}, \{\mathbf{B}_i\}_{i\in\mathcal{I}}, \mathbf{u}, H, \mathsf{BT}\}$ to $\mathcal{A}$ and keeps $(\{\mathbf{R}_j^*\}_{j\in\mathcal{I}}, \{\mathbf{R}_\theta^*\}_{\theta\in\mathsf{BT}}, v_u, \mathbf{v})$ secret.

Phase 1 and 2. When $\mathcal{B}$ receives a key generation query (id, S) from $\mathcal{A}$, where $S = \{att_i\}_{i\in I}$, he outputs $\perp$ if $S \models (W^*, t^*)$ and $\mathsf{id} \notin \mathsf{RL}^*$. Otherwise, the adversary $\mathcal{B}$ picks an unassigned leaf node ν_{id} from BT and stores id in that node.

- For $\mathsf{id}\in\mathsf{RL}^*$, note that in this case $\mathsf{path}(\mathsf{BT}, \nu_{id})\cap\mathsf{KUNodes}(\mathsf{BT}, \mathsf{RL}^*) = \emptyset$. For each node $\theta \in \mathsf{path}(\mathsf{BT}, \nu_{id})$, $\mathcal{B}$ first picks n degree d polynomials $p_{\theta,1}(x), \ldots, p_{\theta,n}(x) \in \mathbb{Z}_q[x]$, such that $\mathbf{u} = (p_{\theta,1}(0), \ldots, p_{\theta,n}(0))^\top$. Then for each $i \in I \cup \mathcal{I}_2$, $\mathcal{B}$ sets $\hat{\mathbf{u}}_{\theta,i} = (p_{\theta,1}(i), \ldots, p_{\theta,n}(i))^\top$. Note that $\mathbf{E}_{\theta,i} = (\mathbf{A}\|\mathbf{A}\mathbf{R}_\theta^* + \mathbf{G}\|\mathbf{B}_i + H(att_i)\mathbf{G})$ for $i \in I$ and $\mathbf{E}_{\theta,i} = (\mathbf{A}\|\mathbf{A}\mathbf{R}_\theta^* + \mathbf{G}\|\mathbf{B}_i + \mathbf{G})$ for $i \in \mathcal{I}_2$. Now, for each $\theta \in \mathsf{path}(\mathsf{BT}, \nu_{id})$ and each $i \in I \cup \mathcal{I}_2$, $\mathcal{B}$ first chooses $\mathbf{e}''_{\theta,i} \leftarrow \mathcal{D}_{\mathbb{Z}^m,\sigma}$, computes $\hat{\mathbf{u}}''_{\theta,i} = (\mathbf{B}_i + H(att_i)\mathbf{G}) \cdot \mathbf{e}''_{\theta,i}$ if $i \in I$ and $\hat{\mathbf{u}}''_{\theta,i} = (\mathbf{B}_i + \mathbf{G}) \cdot \mathbf{e}''_{\theta,i}$ if $i \in \mathcal{I}_2$, runs $\mathbf{e}'_{\theta,i} \leftarrow \mathsf{SampleRight}(\mathbf{A}, \mathbf{G}, \mathbf{R}_\theta^*, \hat{\mathbf{u}}'_{\theta,i}, \mathbf{T}_\mathbf{G}, \sigma)$ where $\hat{\mathbf{u}}'_{\theta,i} = \hat{\mathbf{u}}_{\theta,i} - \hat{\mathbf{u}}''_{\theta,i}$, then sets $\mathbf{e}_{\theta,i} = (\mathbf{e}'_{\theta,i}\|\mathbf{e}''_{\theta,i})$.
- For $\mathsf{id} \notin \mathsf{RL}^*$ and $S \not\models (W^*, t^*)$, there exists a $\theta^* \in \mathsf{path}(\mathsf{BT}, \nu_{id}) \cap \mathsf{KUNodes}(\mathsf{BT}, \mathsf{RL}^*)$. For each $\theta \in \mathsf{path}(\mathsf{BT}, \nu_{id}) \setminus \{\theta^*\}$, $\mathcal{B}$ picks n degree d polynomials $p_{\theta,1}(x), \ldots, p_{\theta,n}(x)$ such that $\mathbf{u} = (p_{\theta,1}(0), \ldots, p_{\theta,n}(0))^\top$. Then it sets $\hat{\mathbf{u}}_{\theta,i} = (p_{\theta,1}(i), \ldots, p_{\theta,n}(i))^\top$ and generates $\mathbf{e}_{\theta,i} = (\mathbf{e}'_{\theta,i}\|\mathbf{e}''_{\theta,i})$ for $i \in I \cup \mathcal{I}_2$ by using the Gaussian sampling and the $\mathsf{SampleRight}$ algorithms according to the above process.
  For θ^*, let $S \cap W^* = \{att_j\}_{j\in K}$, then $|K| < t^*$, thus $|K \cup J_2^*| \le d$. $\mathcal{B}$ chooses a set K' such that $K \cup J_2^* \subseteq K' \subseteq I \cup \mathcal{I}_2$ and $|K'| = d$. For each $i \in K'$, $\mathcal{B}$ chooses $\mathbf{e}_{\theta^*,i} \leftarrow \mathcal{D}_{\mathbb{Z}^{3m},\sigma}$ and if $i \in I$, let $\mathbf{E}_{\theta^*,i} = (\mathbf{A}\|\mathbf{D}_{\theta^*}\|\mathbf{B}_i + H(att_i)\mathbf{G})$, else let $\mathbf{E}_{\theta^*,i} = (\mathbf{A}\|\mathbf{D}_{\theta^*}\|\mathbf{B}_i + \mathbf{G})$, then computes $\hat{\mathbf{u}}_{\theta^*,i} = \mathbf{E}_{\theta^*,i} \cdot \mathbf{e}_{\theta^*,i}$. Thus, we have $d+1$ n-dimensional vectors $\{\mathbf{u}, \{\hat{\mathbf{u}}_{\theta^*,i}\}_{i\in K'}\}$. By the Lagrange interpolation formula, we can recover polynomials $p_{\theta^*,1}(x), \ldots, p_{\theta^*,n}(x)$ such that $\mathbf{u} = (p_{\theta^*,1}(0), \ldots, p_{\theta^*,n}(0))^\top$, and for each $i \in K'$, $\hat{\mathbf{u}}_{\theta^*,i} = (p_{\theta^*,1}(i), \ldots, p_{\theta^*,n}(i))^\top$. Now, for each $i \in I\setminus(K'\cap I)$, if $i \in J_1^*$, we have $att_i \neq att_i^*$ and $\mathbf{E}_{\theta^*,i} = (\mathbf{A}\|\mathbf{D}_{\theta^*}\|\mathbf{A}\mathbf{R}_i^* + (H(att_i) - H(att_i^*))\mathbf{G})$, else we have $att_i \neq \mathbf{0}$ and $\mathbf{E}_{\theta^*,i} = (\mathbf{A}\|\mathbf{D}_{\theta^*}\|\mathbf{A}\mathbf{R}_i^* + (H(att_i) - H(\mathbf{0}))\mathbf{G})$. For each $i \in \mathcal{I}_2\setminus(K' \cap \mathcal{I}_2)$, note that we have $\mathbf{E}_{\theta^*,i} = (\mathbf{A}\|\mathbf{D}_{\theta^*}\|\mathbf{A}\mathbf{R}_i^* + \mathbf{G})$. Now, for $i \in I \cup \mathcal{I}_2$, $\mathcal{B}$ first chooses $\mathbf{e}''_{\theta^*,i} \leftarrow \mathcal{D}_{\mathbb{Z}^m,\sigma}$, computes $\hat{\mathbf{u}}''_{\theta^*,i} = \mathbf{D}_{\theta^*} \cdot \mathbf{e}''_{\theta^*,i}$ and $\hat{\mathbf{u}}'_{\theta^*,i} = \hat{\mathbf{u}}_{\theta^*,i} - \hat{\mathbf{u}}''_{\theta^*,i}$, then runs $(\mathbf{e}'_{\theta^*,i}, \mathbf{e}'''_{\theta^*,i}) \leftarrow \mathsf{SampleRight}(\mathbf{A}, \mathbf{G}, \mathbf{R}_i^*, \hat{\mathbf{u}}'_{\theta^*,i}, \mathbf{T}_\mathbf{G}, \sigma)$, and sets $\mathbf{e}_{\theta^*,i} = (\mathbf{e}'_{\theta^*,i}\|\mathbf{e}''_{\theta^*,i}\|\mathbf{e}'''_{\theta^*,i})$.

  In the end, $\mathcal{B}$ returns $\mathsf{sk}_{S,\mathsf{id}} = \big(\{\mathbf{e}_{\theta,i}\}_{\theta\in\mathsf{path}(\mathsf{BT},\nu_{id}),i\in I\cup\mathcal{I}_2}\big)$ to $\mathcal{A}$.

Challenge. When $\mathcal{A}$ submits two different messages $M_0, M_1 \in \{0,1\}$, the adversary $\mathcal{B}$ picks $b \in \{0,1\}$, computes $c_0 = Dv_u + M_b\lfloor q/2\rfloor$, $\mathbf{c} = D\mathbf{v}$. Then, $\mathcal{B}$ computes $\mathbf{c}_j = D(\mathbf{R}_j^*)^\top\mathbf{v}$ for each $j \in J_1^* \cup J_2^*$, and $\mathbf{c}_\theta = D(\mathbf{R}_\theta^*)^\top\mathbf{v}$ for each $\theta \in \mathsf{KUNodes}(\mathsf{BT}, \mathsf{RL}^*)$. Finally, $\mathcal{B}$ sends to $\mathcal{A}$ the ciphertext $C = \big(c_0, \mathbf{c}, \{\mathbf{c}_j\}_{j\in J_1^*\cup J_2^*}, \{\mathbf{c}_\theta\}_{\theta\in\mathsf{KUNodes}(\mathsf{BT},\mathsf{RL}^*)}\big)$.

Guess. $\mathcal{A}$ output a guess $b' \in \{0,1\}$ for b. If $b' = b$, $\mathcal{B}$ outputs 1, else outputs 0.

Note that by Lemma 8, the pair $(\mathbf{A}, \mathbf{u})$ is computationally indistinguishable from its distribution in the real attack. Applying Lemma 7, we know that $\{\mathbf{B}_i\}_{i\in\mathcal{I}}$ and $\{\mathbf{D}_\theta\}_{\theta\in\mathsf{BT}}$ are statistically close to uniform even given more information about $(\mathbf{R}_i^*)^\top\mathbf{x}$ and $(\mathbf{R}_\theta^*)^\top\mathbf{x}$, respectively. Hence, the distribution of the public key in the simulation is indistinguishable from that in the real attack, and $\mathcal{A}$ gains negligible information about $\{\mathbf{R}_i^*\}_{i\in\mathcal{I}}$ and $\{\mathbf{R}_\theta^*\}_{\theta\in\mathsf{BT}}$ from the public key. According to Lemma 2, 4 and 5, the output distribution of the key generation simulation using the $\mathsf{SampleRight}$ algorithm is statistical to that in the real attack.

If $\mathcal{O}(\cdot) = \mathcal{O}_{\mathbf{s}^*}$ for some $\mathbf{s}^*$, we claim that the challenge ciphertext C^* is a valid ciphertext for $\mathbf{s} = D\mathbf{s}^*$, $\{\mathbf{R}_i^*\}_{i\in J_1^*\cup J_2^*}$, and $\{\mathbf{R}_\theta^*\}_{\theta\in\mathsf{KUNodes}(\mathsf{BT},\mathsf{RL}^*)}$: Note that for each $j\in J_1^*$, $\mathbf{c}_j = D(\mathbf{R}_j^*)^\top\cdot(\mathbf{A}^\top\mathbf{s}^*+\mathbf{x}) = (\mathbf{A}\mathbf{R}_j^*)^\top\cdot(D\mathbf{s}^*)+D\cdot(\mathbf{R}_j^*)^\top\mathbf{x} = (\mathbf{B}_j + H(att_j^*)\mathbf{G})^\top\mathbf{s}+D(\mathbf{R}_j^*)^\top\mathbf{x}$. For each $j\in J_2^*$, $\mathbf{c}_j = D(\mathbf{R}_j^*)^\top\cdot(\mathbf{A}^\top\mathbf{s}^*+\mathbf{x}) = (\mathbf{A}\mathbf{R}_j^*)^\top\cdot(D\mathbf{s}^*)+D\cdot(\mathbf{R}_j^*)^\top\mathbf{x} = (\mathbf{B}_j+\mathbf{G})^\top\mathbf{s}+D(\mathbf{R}_j^*)^\top\mathbf{x}$. For each $\theta\in\mathsf{KUNodes}(\mathsf{BT},\mathsf{RL}^*)$, $\mathbf{c}_\theta = D(\mathbf{R}_\theta^*)^\top(\mathbf{A}^\top\mathbf{s}^*+\mathbf{x}) = (\mathbf{A}\mathbf{R}_\theta^*)^\top(D\mathbf{s}^*)+D(\mathbf{R}_\theta^*)^\top\mathbf{x} = \mathbf{D}_\theta^\top\cdot\mathbf{s}+D(\mathbf{R}_\theta^*)^\top\mathbf{x}$. Therefore, the ciphertext is the same as the view of $\mathcal{A}$ in the real attack.

Hence, if $\mathcal{A}$ guesses the right b with noticeable probability more than $1/2$, then $\mathcal{B}$ can succeed in its game with the same probability. Else if $\mathcal{O}(\cdot) = \mathcal{O}_\$$, then the ciphertexts c_0, $\mathbf{c}$, $\{\mathbf{c}_j\}_{j\in J_1^*\cup J_2^*}$, $\{\mathbf{c}_\theta\}_{\theta\in\mathsf{KUNodes}(\mathsf{BT},\mathsf{RL}^*)}$ are uniform, thus the probability of $\mathcal{A}$ guesses the right b is exactly $1/2$. In a word, if $\mathcal{A}$ breaks the security of our DR-ABE with user-level revocation, then $\mathcal{B}$ solves the underlying LWE problem. □

4.3 Parameters

In this section, we will instantiate the parameters to satisfy the correctness and security of DR-ABE with user-level revocation. In particular, we need to set parameters so that the following conditions hold with overwhelming possibility.

- For the algorithm $\mathsf{TrapGen}$, we need $m \geq 2n\lceil\log q\rceil$ (i.e., Lemma 3).
- For the algorithm $\mathsf{SampleLeft}$, we need $\sigma \geq O(\sqrt{n\log q})\cdot\omega(\sqrt{\log m})$ (i.e., Lemma 3, 4).
- For correctness, we need $|Dx_0 - y_\theta| \leq q/5$.
- For security proof, we need $\sigma \geq \sqrt{m}\cdot\omega(\sqrt{\log m})$ for the algorithm $\mathsf{SampleRight}$ (i.e., Lemma 3, 5, 6) and $m > (n+1)\log q + \omega(\log n)$ (i.e., Lemma 7).
- For the hardness of LWE, we need $\alpha q > 2\sqrt{n}$ (i.e., Lemma 8).

Assume that δ is a real such that $n^{1+\delta} > \lceil(n+1)\log q + \omega(\log n)\rceil$, m, σ, q, α are determined as follows:

- $m = 2n^{1+\delta}$,
- $\sigma = \sqrt{m}\cdot\omega(\sqrt{\log m})$,
- $q = \sigma m^{3/2}(d+1)((\ell+d)!)^4\omega(\sqrt{2\log m})$,
- $\alpha = \big(\sigma m(d+1)((\ell+d)!)^4\omega(\sqrt{\log m})\big)^{-1}$.

5 DR-ABE with attribute-level revocation

The idea of constructing a DR-ABE with user-level user revocation in Section 4 cannot be extended to constructing a DR-ABE with attribute-level user revocation directly for the following reason. Suppose we associate every attribute att_i with a binary tree BT_i of depth L. For each id, we link id to a leaf node $\nu_{\mathsf{id},i}$ of BT_i. Then, for each $l \in [L]$, the random vector $\mathbf{u}$ in the public key is secret-shared into vectors $\{\hat{\mathbf{u}}_{l,i}\}$, where $\hat{\mathbf{u}}_{l,i}$ is associate with the node of depth l in $\mathsf{path}(\mathsf{BT}, \nu_{id,i})$ of BT_i. Now, if the non-revoked attribute set $S_{\mathsf{id},\mathsf{RL}=\{\mathsf{RL}_i\}} = \{att_i \mid \mathsf{id} \notin \mathsf{RL}_i\}$ of id satisfies the access structure, then $\mathbf{u}$ should be recovered if the extension works. Now, for each $att_i \in S_{\mathsf{id},\mathsf{RL}}$, there exists a $\theta_i \in \mathsf{path}(\mathsf{BT}, \nu_{id,i}) \cap \mathsf{KUNodes}(\mathsf{BT}_i, \mathsf{RL}_i)$ and thus $\hat{\mathbf{u}}_{\theta_i,i}$ can be recovered. However, we cannot recover $\hat{\mathbf{u}}$ since θ_i may not be at the same depth.

In this section, we propose a DR-ABE scheme from lattices, which supports attribute-level user revocation and flexible threshold access policies on multi-valued attributes. The main ideas behind our construction can be described as follows. The random vector $\mathbf{u}$ in the public key is secret-shared into vectors $\{\hat{\mathbf{u}}_i\}$, where $\hat{\mathbf{u}}_i$ is associate with the i-th attribute att_i of the identity id. To revoke att_i of id, we further split each $\hat{\mathbf{u}}_i$ into two random vectors $\hat{\mathbf{u}}'_i$ and $\hat{\mathbf{u}}''_i$, corresponding to att_i and id respectively. If the att_i of id is revoked, $\hat{\mathbf{u}}''_i$, and therefore $\hat{\mathbf{u}}_i$, cannot be recovered. In this way, $\mathbf{u}$ can be recovered only if the set of non-revoked attributes of id satisfies the threshold access policy, thereby achieving the revocation of part attributes of id.

For convenience, we use the notations from Section 4.

$\mathsf{Setup}(n, \mathcal{R}, N)$**:** On input a security parameter n, a system attribute set $\mathcal{R} = \mathcal{R}_1 \times \cdots \times \mathcal{R}_\ell$ and a maximal number of users N in the system, this algorithm sets the primitive matrix $\mathbf{G}$ (with public trapdoor $\mathbf{T}_\mathbf{G}$, see Lemma 3) and the parameters q, m, α, σ as specified in Section 4.3. Then it performs as follows:

1. Run $(\mathbf{A}, \mathbf{T}_\mathbf{A}) \leftarrow \mathsf{TrapGen}(n, m, q)$.
2. Choose $\mathbf{B}_i \leftarrow \mathbb{Z}_q^{n \times m}$ for $i \in \mathcal{I}$.
3. Choose $\mathbf{u} \leftarrow \mathbb{Z}_q^n$.
4. Choose a full-rank difference map $H : \mathbb{Z}_q^n \to \mathbb{Z}_q^{n \times n}$.
5. Build a family of binary trees $\mathsf{BT} = \{\mathsf{BT}_i\}_{i \in \mathcal{I}_1}$, where each BT_i has N leaf nodes. For each $i \in \mathcal{I}_1$ and each node $\theta \in \mathsf{BT}_i$, choose "identifier" $\mathbf{D}_{i,\theta} \leftarrow \mathbb{Z}_q^{n \times m}$.
6. Return $\mathsf{PK} = \{\mathbf{A}, \{\mathbf{B}_i\}_{i \in \mathcal{I}}, \mathbf{u}, H, \mathsf{BT}\}$ and $\mathsf{MSK} = \mathbf{T}_\mathbf{A}$.

$\mathsf{Keygen}(\mathsf{PK}, \mathsf{MSK}, \mathsf{id}, S)$**:** On input the public key PK, the master secret key MSK, an identity id, and the attribute set $S = \{att_i\}_{i \in I}$ of id, where $I \subseteq \mathcal{I}_1$ and $att_i \in \mathcal{R}_i$, it goes as follows:

1. For $i \in [1, n]$, randomly choose degree d polynomial $p_i(x) \in \mathbb{Z}_q[x]$, such that $\mathbf{u} = (p_1(0), \ldots, p_n(0))^\top$. For each $i \in I \cup \mathcal{I}_2$, let $\hat{\mathbf{u}}_i = (p_1(i), \ldots, p_n(i))^\top$.
2. For each $i \in I$, pick an unassigned leaf node $\nu_{\mathsf{id},i}$ from BT_i and store id in that node. Choose $\hat{\mathbf{u}}'_i \leftarrow \mathbb{Z}_q^n$ and set $\hat{\mathbf{u}}''_i = \hat{\mathbf{u}}_i - \hat{\mathbf{u}}'_i$. Sample vector $\mathbf{e}'_i \leftarrow \mathsf{SampleLeft}(\mathbf{A}, \mathbf{B}_i + H(att_i)\mathbf{G}, \hat{\mathbf{u}}'_i, \mathbf{T}_\mathbf{A}, \sigma)$. Sample $\mathbf{e}''_{i,\theta} \leftarrow$

$\mathsf{SampleLeft}(\mathbf{A}, \mathbf{D}_{i,\theta}, \hat{\mathbf{u}}_i'', \mathbf{T}_{\mathbf{A}}, \sigma)$ for $\theta \in \mathsf{path}(\mathsf{BT}_i, \nu_{id,i})$.
Let $\mathbf{E}_i' = (\mathbf{A} \| \mathbf{B}_i + H(att_i)\mathbf{G})$ and $\mathbf{E}_{i,\theta}'' = (\mathbf{A} \| \mathbf{D}_{i,\theta})$, note that $\mathbf{E}_i' \cdot \mathbf{e}_i' = \hat{\mathbf{u}}_i'$ and $\mathbf{E}_i'' \cdot \mathbf{e}_{i,\theta}'' = \hat{\mathbf{u}}_i''$.

3. For each $i \in \mathcal{I}_2$, sample $\mathbf{e}_i \leftarrow \mathsf{SampleLeft}(\mathbf{A}, \mathbf{B}_i + \mathbf{G}, \hat{\mathbf{u}}_i, \mathbf{T}_{\mathbf{A}}, \sigma)$. Let $\mathbf{E}_i = (\mathbf{A} \| \mathbf{B}_i + \mathbf{G})$, note that $\mathbf{E}_i \cdot \mathbf{e}_i = \hat{\mathbf{u}}_i$.
4. Return $\mathsf{sk}_{S,\mathsf{id}} = \left(\{\mathbf{e}_i'\}_{i \in I}, \{\mathbf{e}_{i,\theta}''\}_{i \in I, \theta \in \mathsf{path}(\mathsf{BT}_i, \nu_{id,i})}, \{\mathbf{e}_i\}_{i \in \mathcal{I}_2} \right)$ as the private key.

Note that for any subset $K \subseteq I \cup \mathcal{I}_2$, $|K| = d+1$, we have $\mathbf{u} = \sum_{i \in K} L_i \cdot \hat{\mathbf{u}}_i$, where the Lagrange coefficient $L_i = \frac{\prod_{j \in K, j \neq i} -j}{\prod_{j \in K, j \neq i}(i-j)}$.

$\mathsf{Enc}(\mathsf{PK}, (W, t), \mathsf{RL}, M)$**:** On input a public key PK, an attribute set $W = \{att_j\}_{j \in J_1}$, an integer $1 \leq t \leq \min(|W|, d)$, a family of attribute revocation lists $\mathsf{RL} = \{\mathsf{RL}_j\}_{j \in J_1}$ where each RL_j consisting of identities whose j-th attribute is revoked and a message $M \in \{0,1\}$, it works as follows:

1. Choose $\mathbf{s} \leftarrow \mathbb{Z}_q^n$, compute $c_0 = \mathbf{u}^\top \mathbf{s} + Dx_0 + M\lfloor \frac{q}{2} \rfloor$ and $\mathbf{c} = \mathbf{A}^\top \mathbf{s} + D\mathbf{x}$, where $x_0 \leftarrow \bar{\Psi}_\alpha$, $\mathbf{x} \leftarrow \bar{\Psi}_\alpha^m$.
2. For each $j \in J_1$, choose $\mathbf{R}_j \leftarrow \{-1,1\}^{m \times m}$, compute $\mathbf{c}_j' = (\mathbf{B}_j + H(att_j)\mathbf{G})^\top \mathbf{s} + D\mathbf{R}_j^\top \mathbf{x}$.
3. For each $j \in J_1$ and each $\theta \in \mathsf{KUNodes}(\mathsf{BT}_j, \mathsf{RL}_j)$, choose $\mathbf{R}_{j,\theta} \leftarrow \{-1,1\}^{m \times m}$, compute $\mathbf{c}_{j,\theta}'' = \mathbf{D}_{j,\theta}^\top \cdot \mathbf{s} + D\mathbf{R}_{j,\theta}^\top \cdot \mathbf{x}$.
4. Let $J_2 = \{\ell + 1, \ldots, \ell + d + 1 - t\}$ and for each $j \in J_2$, choose $\mathbf{R}_j \leftarrow \{-1,1\}^{m \times m}$, compute $\mathbf{c}_j = (\mathbf{B}_j + \mathbf{G})^\top \mathbf{s} + D\mathbf{R}_j^\top \mathbf{x}$.
5. Return $C = \left(c_0, \mathbf{c}, \{\mathbf{c}_j'\}_{j \in J_1}, \{\mathbf{c}_{j,\theta}''\}_{j \in J_1, \theta \in \mathsf{KUNodes}(\mathsf{BT}_j, \mathsf{RL}_j)}, \{\mathbf{c}_j\}_{j \in J_2} \right)$ as the ciphertext.

$\mathsf{Dec}(\mathsf{PK}, \mathsf{sk}_{S,\mathsf{id}}, C)$**:** On input the public key PK, the private key $\mathsf{sk}_{S,\mathsf{id}}$ of identity id with attribute set $S = \{att_i\}_{i \in I}$ and a ciphertext C encrypted under access structure $(W = \{att_j\}_{j \in J_1}, t)$ and a family of revocation lists $\mathsf{RL} = \{\mathsf{RL}_j\}_{j \in J_1}$. Let $S_{\mathsf{id},\mathsf{RL}} = \{att_i \in S \mid \mathsf{id} \notin \mathsf{RL}_i, i \in I\}$.

1. If $|S_{\mathsf{id},\mathsf{RL}} \cap W| < t$, then return $\perp$;
2. Else, parse $\mathsf{sk}_{S,\mathsf{id}} = \left(\{\mathbf{e}_i'\}_{i \in I}, \{\mathbf{e}_{i,\theta}''\}_{i \in I, \theta \in \mathsf{path}(\mathsf{BT}_i, \nu_{id,i})}, \{\mathbf{e}_i\}_{i \in \mathcal{I}_2} \right)$ and $C = \left(c_0, \mathbf{c}, \{\mathbf{c}_j'\}_{j \in J_1}, \{\mathbf{c}_{j,\theta}''\}_{j \in J_1, \theta \in \mathsf{KUNodes}(\mathsf{BT}_j, \mathsf{RL}_j)}, \{\mathbf{c}_j\}_{j \in J_2} \right)$. Let $S_{\mathsf{id},\mathsf{RL}} \cap W = \{att_i\}_{i \in K}$. Since $|K| \geq t$, there exists a set $K' \subseteq K \cup J_2$ with size $d+1$. For all $j \in K' \cap K$, there exists a $\theta_j \in \mathsf{path}(\mathsf{BT}_j, \nu_{\mathsf{id},j}) \cap \mathsf{KUNodes}(\mathsf{BT}_j, \mathsf{RL}_j)$, compute $r_j = \mathbf{e}_j'^\top(\mathbf{c}; \mathbf{c}_j') + \mathbf{e}_{j,\theta_j}''^\top(\mathbf{c}; \mathbf{c}_{j,\theta_j}'')$. For all $j \in K' \cap J_2$, compute $r_j = \mathbf{e}_j^\top(\mathbf{c}; \mathbf{c}_j)$. Then, compute $r = \sum_{j \in K'} L_j r_j$, where $L_j = \frac{\prod_{k \in K', k \neq j} -k}{\prod_{k \in K', k \neq j}(j-k)}$. Finally, compute $\hat{r} = c_0 - r$. If $|\hat{r} - \lfloor \frac{q}{2} \rfloor| \leq \lfloor \frac{q}{4} \rfloor$ in $\mathbb{Z}$, return 1, otherwise return 0.

5.1 Correctness

For $j \in K' \cap K$ and $\theta_j \in \mathsf{path}(\mathsf{BT}_j, \nu_{\mathsf{id},j}) \cap \mathsf{KUNodes}(\mathsf{BT}_j, \mathsf{RL}_j)$, we have

$$\begin{aligned}(\mathbf{c}; \mathbf{c}'_j) = \begin{bmatrix}\mathbf{c} \\ \mathbf{c}'_j\end{bmatrix} &= \begin{bmatrix}\mathbf{A}^\top \mathbf{s} + D\mathbf{x} \\ (\mathbf{B}_j + H(att_j)\mathbf{G})^\top \mathbf{s} + D\mathbf{R}_j^\top \mathbf{x}\end{bmatrix} \\ &= (\mathbf{A} \| \mathbf{B}_j + H(att_j)\mathbf{G})^\top \mathbf{s} + D \begin{bmatrix}\mathbf{x} \\ \mathbf{R}_j^\top \mathbf{x}\end{bmatrix}.\end{aligned}$$

$$(\mathbf{c}; \mathbf{c}''_{j,\theta_j}) = \begin{bmatrix}\mathbf{c} \\ \mathbf{c}''_{j,\theta_j}\end{bmatrix} = \begin{bmatrix}\mathbf{A}^\top \mathbf{s} + D\mathbf{x} \\ \mathbf{D}_{j,\theta_j}^\top \cdot \mathbf{s} + D\mathbf{R}_{j,\theta_j}^\top \cdot \mathbf{x}\end{bmatrix} = (\mathbf{A} \| \mathbf{D}_{j,\theta_j})^\top \mathbf{s} + D \begin{bmatrix}\mathbf{x} \\ \mathbf{R}_{j,\theta_j}^\top \mathbf{x}\end{bmatrix}.$$

Thus,

$$\begin{aligned}r_j =& \mathbf{e}_j'^\top (\mathbf{c}; \mathbf{c}'_j) + \mathbf{e}''^\top_{j,\theta_j} (\mathbf{c}; \mathbf{c}''_{j,\theta_j}) \\ =& \mathbf{e}_j'^\top (\mathbf{A} \| \mathbf{B}_i + H(att_i)\mathbf{G})^\top \mathbf{s} + D\mathbf{e}_j'^\top (\mathbf{x}; \mathbf{R}_j^\top \mathbf{x}) + \mathbf{e}''^\top_{j,\theta_j} (\mathbf{A} \| \mathbf{D}_{j,\theta_j})^\top \mathbf{s} + D\mathbf{e}''^\top_{j,\theta_j} (\mathbf{x}; \mathbf{R}_{j,\theta_j}^\top) \\ =& \left(\hat{\mathbf{u}}_j'^\top + \hat{\mathbf{u}}_j''^\top\right) \mathbf{s} + D\left(\mathbf{e}_j'^\top \mathbf{x}'_j + \mathbf{e}''^\top_{j,\theta_j} \mathbf{x}''_{j,\theta_j}\right) \\ =& \hat{\mathbf{u}}_j^\top \mathbf{s} + D\left(\mathbf{e}_j'^\top \mathbf{x}'_j + \mathbf{e}''^\top_{j,\theta_j} \mathbf{x}''_{j,\theta_j}\right),\end{aligned}$$

where $\mathbf{x}'_j = (\mathbf{x}; \mathbf{R}_j^\top \mathbf{x})$ and $\mathbf{x}''_{j,\theta_j} = (\mathbf{x}; \mathbf{R}_{j,\theta_j}^\top \mathbf{x})$.

For $j \in K' \cap J_2$, we have

$$(\mathbf{c}; \mathbf{c}_j) = \begin{bmatrix}\mathbf{c} \\ \mathbf{c}_j\end{bmatrix} = \begin{bmatrix}\mathbf{A}^\top \mathbf{s} + D\mathbf{x} \\ (\mathbf{B}_j + \mathbf{G})^\top \mathbf{s} + D\mathbf{R}_j^\top \mathbf{x}\end{bmatrix} = (\mathbf{A} \| \mathbf{B}_j + \mathbf{G})^\top \mathbf{s} + D \begin{bmatrix}\mathbf{x} \\ \mathbf{R}_j^\top \mathbf{x}\end{bmatrix}.$$

Thus, $r_j = \mathbf{e}_j^\top (\mathbf{c}; \mathbf{c}_j) = \mathbf{e}_j^\top (\mathbf{A} \| \mathbf{B}_j + \mathbf{G})^\top \mathbf{s} + D\mathbf{e}_j^\top (\mathbf{x}; \mathbf{R}_j^\top \mathbf{x}) = \hat{\mathbf{u}}_j^\top \mathbf{s} + D\mathbf{e}_j^\top \mathbf{x}'_j$, where $\mathbf{x}'_j = (\mathbf{x}; \mathbf{R}_j^\top \mathbf{x})$.

Then, we have

$$\begin{aligned}r =& \sum_{j \in K'} L_j r_j \\ =& \sum_{j \in K' \cap K} L_j \left(\hat{\mathbf{u}}_j^\top \mathbf{s} + D\left(\mathbf{e}_j'^\top \mathbf{x}'_j + \mathbf{e}''^\top_{j,\theta_j} \mathbf{x}''_{j,\theta_j}\right)\right) + \sum_{j \in K' \cap J_2} L_j \left(\hat{\mathbf{u}}_j^\top \mathbf{s} + D\mathbf{e}_j^\top \mathbf{x}'_j\right) \\ =& \left(\sum_{j \in K'} L_j \hat{\mathbf{u}}_j^\top\right) \mathbf{s} + y = \mathbf{u}^\top \mathbf{s} + y,\end{aligned}$$

where $y = D\left(\sum_{j \in K'} L_j \mathbf{e}_j'^\top \mathbf{x}'_j + \sum_{j \in K' \cap K} L_j \mathbf{e}''^\top_{j,\theta_j} \mathbf{x}''_{j,\theta_j}\right)$.

Finally, we have

$$\hat{r} = c_0 - r = Dx_0 - y + M \cdot \lfloor \frac{q}{2} \rfloor.$$

Now, we begin to bound $|Dx_0 - y|$. By Lemma 1 and 4, we have $\|\mathbf{e}'_j\| \leq \sigma\sqrt{2m}$ and $\|\mathbf{e}''_{j,\theta_j}\| \leq \sigma\sqrt{2m}$. For $j \in K'$, $\mathbf{e}_j'^{\top}\mathbf{x}'_j = \mathbf{e}_{j,0}'^{\top}\mathbf{x} + \mathbf{e}_{j,1}'^{\top}\mathbf{R}_j^{\top}\mathbf{x}$, where $\mathbf{e}'_j = (\mathbf{e}'_{j,0}; \mathbf{e}'_{j,1})$. Since $\|\mathbf{e}'_{j,0} + \mathbf{R}_j\mathbf{e}'_{j,1}\| \leq (s_1(\mathbf{R}_j) + 1) \cdot \sigma\sqrt{2m}$, by Lemma 9, we have $|\mathbf{e}_j'^{\top}\mathbf{x}'_j| \leq (s_1(\mathbf{R}_j) + 1)\sigma\sqrt{2m}(q\alpha\omega(\sqrt{\log m}) + \sqrt{m}/2)$. Similarly, for $j \in K' \cap K$, we have $|\mathbf{e}_j''^{\top}\mathbf{x}''_{j,\theta_j}| \leq (s_1(\mathbf{R}_{j,\theta_j}) + 1)\sigma\sqrt{2m}(q\alpha\omega(\sqrt{\log m}) + \sqrt{m}/2)$. Applying Lemma 9 in [2], we have $DL_j \leq ((\ell + d)!)^4$. By Lemma 6, we have $s_1(\mathbf{R}_j), s_1(\mathbf{R}_{j,\theta_j}) = O(\sqrt{m})$. Thus, $|y| \leq 2(d+1)((\ell+d)!)^4\sigma O(m)\cdot(q\alpha\omega(\sqrt{\log m})+\sqrt{m}/2)$. Therefore, we have $|Dx_0 - y| \leq ((\ell + d)!)^2(q\alpha\omega(\sqrt{\log m}) + 1/2) + |y| \leq \sigma q\alpha m(d + 1)((\ell + d)!)^4\omega(\sqrt{\log m}) + \sigma(d + 1)((\ell + d)!)^4 O(m^{3/2})$ by Lemma 9.

5.2 Security

In this section, we prove the security of our DR-ABE scheme with attribute-level revocation in the selective model in Definition 1.

Theorem 2. *For appropriate parameters n, m, q, σ, α, the above DR-ABE scheme with attribute-level revocation is secure provided that the $(\mathbb{Z}_q, n, \bar{\Psi}_\alpha)$-LWE problem is hard.*

Proof. Suppose there exists a PPT adversary $\mathcal{A}$ breaks the security of our DR-ABE scheme with non-negligible probability, we can construct an algorithm $\mathcal{B}$ that solves the LWE problem with the same advantage.

Note that $\mathcal{B}$ has an oracle $\mathcal{O}(\cdot)$ and he want to determine whether it is a noisy pseudo-random sampler $\mathcal{O}_{\mathbf{s}^*}$ for some $\mathbf{s}^* \in \mathbb{Z}_q^n$ or a truly random sampler $\mathcal{O}_\$$. To this end, $\mathcal{B}$ proceeds as follows:

Init. $\mathcal{A}$ submits a challenge access structure $\mathbb{A}^* = (W^* = \{att_j^*\}_{j\in J_1^*}, t^*)$ and a family of challenge attribute revocation lists $\mathsf{RL}^* = \{\mathsf{RL}_j^*\}_{j\in J_1^*}$ to $\mathcal{B}$, where $J_1^* \subseteq \mathcal{I}_1$ and $1 \leq t^* \leq \min(|W^*|, d)$. Let $J_2^* = \{\ell + 1, \ldots, \ell + d + 1 - t^*\}$ and let $J^* = J_1^* \cup J_2^*$.

Setup. After receiving $(W^* = \{att_j^*\}_{j\in J_1^*}, t^*)$ and RL^*, $\mathcal{B}$ samples $(\mathbf{u}, v_u) \in \mathbb{Z}_q^n \times \mathbb{Z}_q$ and $(\mathbf{A}, \mathbf{v}) \in \mathbb{Z}_q^{n\times m} \times \mathbb{Z}_q^m$ from $\mathcal{O}(\cdot)$, chooses an FRD map $H : \mathbb{Z}_q^n \to \mathbb{Z}_q^{n\times n}$, builds a family of binary trees $\mathsf{BT} = \{\mathsf{BT}_i\}_{i\in\mathcal{I}_1}$, where each BT_i has N leaf nodes.

- For each $j \in J_1^*$ and each $\theta \in \mathsf{BT}_j$, $\mathcal{B}$ randomly chooses $\mathbf{R}_j^*, \mathbf{R}_{j,\theta}^* \leftarrow \{-1, 1\}^{m\times m}$, and computes $\mathbf{B}_j = \mathbf{A}\mathbf{R}_j^* - H(att_j^*)\mathbf{G}$, $\mathbf{D}_{j,\theta} = \mathbf{A}\mathbf{R}_{j,\theta}^*$ if $\theta \in \mathsf{KUNodes}(\mathsf{BT}_j, \mathsf{RL}_j^*)$, $\mathbf{D}_{j,\theta} = \mathbf{A}\mathbf{R}_{j,\theta}^* + \mathbf{G}$ if $\theta \in \mathsf{BT}_j \setminus \mathsf{KUNodes}(\mathsf{BT}_j, \mathsf{RL}_j^*)$.
- For each $j \in \mathcal{I}_1 \setminus J_1^*$ and each $\theta \in \mathsf{BT}_j$, $\mathcal{B}$ randomly chooses $\mathbf{R}_j^*, \mathbf{R}_{j,\theta}^* \leftarrow \{-1, 1\}^{m\times m}$, and computes $\mathbf{B}_j = \mathbf{A}\mathbf{R}_j^* - H(\mathbf{0})\mathbf{G}$ and $\mathbf{D}_{j,\theta} = \mathbf{A}\mathbf{R}_{j,\theta}^* + \mathbf{G}$.
- For each $j \in J_2^*$, $\mathcal{B}$ chooses $\mathbf{R}_j^* \leftarrow \{-1, 1\}^{m\times m}$, computes $\mathbf{B}_j = \mathbf{A}\mathbf{R}_j^* - \mathbf{G}$.
- For each $j \in \mathcal{I}_2 \setminus J_2^*$, $\mathcal{B}$ chooses $\mathbf{R}_j^* \leftarrow \{-1, 1\}^{m\times m}$, computes $\mathbf{B}_j = \mathbf{A}\mathbf{R}_j^*$.

Finally, $\mathcal{B}$ sends the public key $\mathsf{PK} = \{\mathbf{A}, \{\mathbf{B}_i\}_{i\in\mathcal{I}}, \mathbf{u}, H, \mathsf{BT}\}$ to $\mathcal{A}$ and keeps $(\{\mathbf{R}_j^*\}_{j\in\mathcal{I}}, \{\mathbf{R}_{j,\theta}^*\}_{j\in\mathcal{I}_1,\theta\in\mathsf{BT}_j}, v_u, \mathbf{v})$ secret.

Phase 1 and 2. When $\mathcal{B}$ receives a key generation query (id, S) from $\mathcal{A}$, where $S = \{att_i\}_{i\in I}$, he outputs $\perp$ if $S_{\mathsf{id},\mathsf{RL}^*} = \{att_i \in S \mid \mathsf{id} \notin \mathsf{RL}_i^*, i \in I\} \models$

(W^*, t^*). Otherwise, for each $i \in I$, $\mathcal{B}$ picks an unassigned leaf node $\nu_{id,i}$ from BT_i and stores id in that node. Let $S_{\mathsf{id},\mathsf{RL}^*} \cap W^* = \{att_j\}_{j\in K}$, we have $|K| < t^*$, thus $|K \cup J_2^*| \leq d$. Then $\mathcal{B}$ chooses a set K' such that $K \cup J_2^* \subseteq K' \subseteq I \cup \mathcal{I}_2$ and $|K'| = d$.
For each $j \in K'$:

- If $j \in I$, choose $\mathbf{e}'_j \leftarrow \mathcal{D}_{\mathbb{Z}^{2m},\sigma}$, let $\mathbf{E}'_j = (\mathbf{A}\|\mathbf{B}_j + H(att_j)\mathbf{G})$, then compute $\hat{\mathbf{u}}'_j = \mathbf{E}'_j \cdot \mathbf{e}'_j$.
  - If $j \in J_1^*$ and $\mathsf{id} \notin \mathsf{RL}_j^*$, there exists a $\theta_j^* \in \mathsf{path}(\mathsf{BT}_j, \nu_{id,j}) \cap \mathsf{KUNodes}(\mathsf{BT}_j, \mathsf{RL}_j^*)$. Choose $\mathbf{e}''_{j,\theta_j^*} \leftarrow \mathcal{D}_{\mathbb{Z}^{2m},\sigma}$, let $\mathbf{E}''_{j,\theta_j^*} = (\mathbf{A}\|\mathbf{D}_{j,\theta_j^*})$, and compute $\hat{\mathbf{u}}''_j = \mathbf{E}''_{j,\theta_j^*} \cdot \mathbf{e}''_{j,\theta_j^*}$. For each $\theta \in \mathsf{path}(\mathsf{BT}_j, \nu_{id,j}) \backslash \{\theta_j^*\}$, let $\mathbf{E}''_{j,\theta} = (\mathbf{A}\|\mathbf{D}_{j,\theta}) = (\mathbf{A}\|\mathbf{A}\mathbf{R}^*_{j,\theta} + \mathbf{G})$, and then sample $\mathbf{e}''_{j,\theta} \leftarrow \mathsf{SampleRight}(\mathbf{A}, \mathbf{G}, \mathbf{R}^*_{j,\theta}, \hat{\mathbf{u}}''_j, \mathbf{T}_{\mathbf{G}}, \sigma)$ such that $\mathbf{E}''_{j,\theta} \cdot \mathbf{e}''_{j,\theta} = \hat{\mathbf{u}}''_j$.
  - Else, pick $\hat{\mathbf{u}}''_j \leftarrow \mathbb{Z}_q^n$. For $\theta \in \mathsf{path}(\mathsf{BT}_j, \nu_{id,j})$, let $\mathbf{E}''_{j,\theta} = (\mathbf{A}\|\mathbf{D}_{j,\theta}) = (\mathbf{A}\|\mathbf{A}\mathbf{R}^*_{j,\theta} + \mathbf{G})$, sample $\mathbf{e}''_{j,\theta} \leftarrow \mathsf{SampleRight}(\mathbf{A}, \mathbf{G}, \mathbf{R}^*_{j,\theta}, \hat{\mathbf{u}}''_j, \mathbf{T}_{\mathbf{G}}, \sigma)$ such that $\mathbf{E}''_{j,\theta} \cdot \mathbf{e}''_{j,\theta} = \hat{\mathbf{u}}''_j$.

  Then $\mathcal{B}$ computes $\hat{\mathbf{u}}_j = \hat{\mathbf{u}}'_j + \hat{\mathbf{u}}''_j$.
- If $j \in \mathcal{I}_2$, choose $\mathbf{e}_j \leftarrow \mathcal{D}_{\mathbb{Z}^{2m},\sigma}$, let $\mathbf{E}_j = (\mathbf{A}\|\mathbf{B}_j + \mathbf{G})$, compute $\hat{\mathbf{u}}_j = \mathbf{E}_j \cdot \mathbf{e}_j$.

Let n degree d polynomials $p_1(x), \ldots, p_n(x)$ such that $\mathbf{u} = (p_1(0), \ldots, p_n(0))$, $\hat{\mathbf{u}}_j = (p_1(j), \ldots, p_n(j))$ for each $j \in K'$. Then we can recover polynomials $p_1(x), \ldots, p_n(x) \in \mathbb{Z}_q[x]$ by the Lagrange interpolation formula. Compute $\hat{\mathbf{u}}_j = (p_1(j), \ldots, p_n(j))$ for each $j \in (I \cup \mathcal{I}_2) \setminus K'$.
For each $j \in I \setminus (K' \cap I)$:

- If $j \in J_1^*$ and $att_j = att_j^*$, we have $\mathsf{id} \in \mathsf{RL}_j^*$. Choose $\mathbf{e}'_j \leftarrow \mathcal{D}_{\mathbb{Z}^{2m},\sigma}$, let $\mathbf{E}'_j = (\mathbf{A}\|\mathbf{B}_j + H(att_j^*)\mathbf{G}) = (\mathbf{A}\|\mathbf{A}\mathbf{R}_j^*)$, compute $\hat{\mathbf{u}}'_j = \mathbf{E}'_j \cdot \mathbf{e}'_j$ and $\hat{\mathbf{u}}''_j = \hat{\mathbf{u}}_j - \hat{\mathbf{u}}'_j$. For each $\theta \in \mathsf{path}(\mathsf{BT}_j, \nu_{id,j})$, let $\mathbf{E}''_{j,\theta} = (\mathbf{A}\|\mathbf{D}_{j,\theta}) = (\mathbf{A}\|\mathbf{A}\mathbf{R}^*_{j,\theta} + \mathbf{G})$, $\mathcal{B}$ can sample $\mathbf{e}''_{j,\theta} \sim \mathcal{D}_{\Lambda_q^{\hat{\mathbf{u}}''_j}(\mathbf{E}''_{j,\theta}),\sigma}$ by using the $\mathsf{SampleRight}$ algorithm.
- If $j \in J_1^*$, $att_j \neq att_j^*$ and $\mathsf{id} \notin \mathsf{RL}_j^*$, there exists a $\theta_j^* \in \mathsf{path}(\mathsf{BT}_j, \nu_{id,j}) \cap \mathsf{KUNodes}(\mathsf{BT}_j, \mathsf{RL}_j^*)$. Choose $\mathbf{e}''_{j,\theta_j^*} \leftarrow \mathcal{D}_{\mathbb{Z}^{2m},\sigma}$, let $\mathbf{E}''_{j,\theta_j^*} = (\mathbf{A}\|\mathbf{D}_{j,\theta_j^*})$, compute $\hat{\mathbf{u}}''_j = \mathbf{E}''_{j,\theta_j^*} \cdot \mathbf{e}''_{j,\theta_j^*}$. For $\theta \in \mathsf{path}(\mathsf{BT}_j, \nu_{id,j}) \backslash \{\theta_j^*\}$, let $\mathbf{E}''_{j,\theta} = (\mathbf{A}\|\mathbf{D}_{j,\theta}) = (\mathbf{A}\|\mathbf{A}\mathbf{R}^*_{j,\theta} + \mathbf{G})$, sample $\mathbf{e}''_{j,\theta} \leftarrow \mathsf{SampleRight}(\mathbf{A}, \mathbf{G}, \mathbf{R}^*_{j,\theta}, \hat{\mathbf{u}}''_j, \mathbf{T}_{\mathbf{G}}, \sigma)$ such that $\mathbf{E}''_{j,\theta} \cdot \mathbf{e}''_{j,\theta} = \hat{\mathbf{u}}''_j$. Then compute $\hat{\mathbf{u}}'_j = \hat{\mathbf{u}}_j - \hat{\mathbf{u}}''_j$, sample $\mathbf{e}'_j \sim \mathcal{D}_{\Lambda_q^{\hat{\mathbf{u}}'_j}(\mathbf{E}'_j),\sigma}$ by using $\mathsf{SampleRight}$ algorithm, where $\mathbf{E}'_j = (\mathbf{A}\|\mathbf{B}_j + H(att_j)\mathbf{G}) = (\mathbf{A}\|\mathbf{A}\mathbf{R}_j^* + (H(att_j) - H(att_j^*))\mathbf{G})$.
- Otherwise, choose $\hat{\mathbf{u}}''_j \leftarrow \mathbb{Z}_q^n$, compute $\hat{\mathbf{u}}'_j = \hat{\mathbf{u}}_j - \hat{\mathbf{u}}''_j$. For $\theta \in \mathsf{path}(\mathsf{BT}_j, \nu_{id,j})$, sample $\mathbf{e}''_{j,\theta} \leftarrow \mathsf{SampleRight}(\mathbf{A}, \mathbf{G}, \mathbf{R}^*_{j,\theta}, \hat{\mathbf{u}}''_j, \mathbf{T}_{\mathbf{G}}, \sigma)$ for $\mathbf{E}''_{j,\theta} = (\mathbf{A}\|\mathbf{A}\mathbf{R}^*_{j,\theta} + \mathbf{G})$. Then sample $\mathbf{e}'_j \sim \mathcal{D}_{\Lambda_q^{\hat{\mathbf{u}}'_j}(\mathbf{E}'_j),\sigma}$ by using the $\mathsf{SampleRight}$ algorithm, where $\mathbf{E}'_j = (\mathbf{A}\|\mathbf{A}\mathbf{R}_j^* + (H(att_j) - H(att_j^*))\mathbf{G})$ if $j \in J_1^*$, and $\mathbf{E}'_j = (\mathbf{A}\|\mathbf{A}\mathbf{R}_j^* + (H(att_j) - H(\mathbf{0}))\mathbf{G})$ if $j \notin J_1^*$.

For each $j \in \mathcal{I}_2 \setminus (K' \cap \mathcal{I}_2)$, let $\mathbf{E}_j = (\mathbf{A}\|\mathbf{B}_j + \mathbf{G}) = (\mathbf{A}\|\mathbf{A}\mathbf{R}_j^* + \mathbf{G})$, sample $\mathbf{e}_j \leftarrow \mathsf{SampleRight}(\mathbf{A}, \mathbf{G}, \mathbf{R}_j^*, \hat{\mathbf{u}}_j, \mathbf{T}_{\mathbf{G}}, \sigma)$.
Finally, $\mathcal{B}$ sends $\mathsf{sk}_{S,\mathsf{id}} = \left(\{\mathbf{e}'_i\}_{i\in I}, \{\mathbf{e}''_{i,\theta}\}_{i\in I, \theta\in\mathsf{path}(\mathsf{BT}_i,\nu_{id,i})}, \{\mathbf{e}_i\}_{i\in\mathcal{I}_2}\right)$ to $\mathcal{A}$.

Challenge. When $\mathcal{A}$ submits two different messages $M_0, M_1 \in \{0,1\}$, $\mathcal{B}$ flips a random coin $b \in \{0,1\}$, computes $c_0 = Dv_u + M_b\lfloor q/2 \rfloor$, $\mathbf{c} = D\mathbf{v}$. For each $j \in J_1^*$ and each $\theta \in \mathsf{KUNodes}(\mathsf{BT}_j, \mathsf{RL}_j^*)$, $\mathcal{B}$ computes $\mathbf{c}'_j = D(\mathbf{R}_j^*)^\top \mathbf{v}$ and $\mathbf{c}''_{j,\theta} = D(\mathbf{R}_{j,\theta}^*)^\top \mathbf{v}$. For each $j \in J_2^*$, $\mathcal{B}$ computes $\mathbf{c}_j = D(\mathbf{R}_j^*)^\top \mathbf{v}$. Finally, $\mathcal{B}$ sends the ciphertext $C^* = \left(c_0, \mathbf{c}, \{\mathbf{c}'_j\}_{j\in J_1^*}, \{\mathbf{c}''_{j,\theta}\}_{j\in J_1^*, \theta\in\mathsf{KUNodes}(\mathsf{BT}_j,\mathsf{RL}_j^*)}, \{\mathbf{c}_j\}_{j\in J_2^*}\right)$ to $\mathcal{A}$.

Guess. $\mathcal{A}$ output a guess $b' \in \{0,1\}$ for b. If $b' = b$, $\mathcal{B}$ outputs 1, else outputs 0.

Note that by Lemma 8, the pair $(\mathbf{A}, \mathbf{u})$ is computationally indistinguishable from its distribution in the real attack. Applying Lemma 7, we know that $\{\mathbf{B}_i\}_{i\in\mathcal{I}}$ and $\{\mathbf{D}_{i,\theta}\}_{i\in\mathcal{I}_1,\theta\in\mathsf{BT}_i}$ are statistically close to uniform even given more information about $(\mathbf{R}_i^*)^\top\mathbf{x}$ and $(\mathbf{R}_{i,\theta}^*)^\top\mathbf{x}$, respectively. Hence, the distribution of the public key in the simulation is indistinguishable from that in the real attack, and $\mathcal{A}$ gains negligible information about $\{\mathbf{R}_i^*\}_{i\in\mathcal{I}}$ and $\{\mathbf{R}_{i,\theta}^*\}_{i\in\mathcal{I}_1,\theta\in\mathsf{BT}_i}$ from the public key. According to Lemma 2, 4 and 5, the output distribution of the key generation simulation using the $\mathsf{SampleRight}$ algorithm is statistical to that in the real attack.

If $\mathcal{O}(\cdot) = \mathcal{O}_{\mathbf{s}^*}$ for some $\mathbf{s}^*$, we claim that the challenge ciphertext C^* is a valid ciphertext for $\mathbf{s} = D\mathbf{s}^*$, $\{\mathbf{R}_i^*\}_{i\in J_1^*\cup J_2^*}$, and $\{\mathbf{R}_{i,\theta}^*\}_{i\in J_1^*,\theta\in\mathsf{KUNodes}(\mathsf{BT}_i,\mathsf{RL}_i^*)}$: Note that for each $j\in J_1^*$ and each $\theta\in\mathsf{KUNodes}(\mathsf{BT}_j,\mathsf{RL}_j^*)$, $\mathbf{c}'_j = D(\mathbf{R}_j^*)^\top(\mathbf{A}^\top\mathbf{s}^* + \mathbf{x}) = (\mathbf{A}\mathbf{R}_j^*)^\top(D\mathbf{s}^*) + D(\mathbf{R}_j^*)^\top\mathbf{x} = (\mathbf{B}_j + H(att_j^*)\mathbf{G})^\top\mathbf{s} + D(\mathbf{R}_j^*)^\top\mathbf{x}$ and $\mathbf{c}''_{j,\theta} = D(\mathbf{R}_{j,\theta}^*)^\top(\mathbf{A}^\top\mathbf{s}^*+\mathbf{x}) = (\mathbf{A}\mathbf{R}_{j,\theta}^*)^\top(D\mathbf{s}^*) + D(\mathbf{R}_{j,\theta}^*)^\top\mathbf{x} = \mathbf{D}_{j,\theta}^\top\cdot\mathbf{s} + D(\mathbf{R}_{j,\theta}^*)^\top\mathbf{x}$. For each $j\in J_2^*$, $\mathbf{c}_j = D(\mathbf{R}_j^*)^\top(\mathbf{A}^\top\mathbf{s}^* + \mathbf{x}) = (\mathbf{A}\mathbf{R}_j^*)^\top(D\mathbf{s}^*) + D(\mathbf{R}_j^*)^\top\mathbf{x} = (\mathbf{B}_j + \mathbf{G})^\top\mathbf{s} + D(\mathbf{R}_j^*)^\top\mathbf{x}$. Therefore, the ciphertext is the same as the view of $\mathcal{A}$ in the real attack.

Hence, if $\mathcal{A}$ guesses the right b with noticeable probability more than $1/2$, then $\mathcal{B}$ can succeed in its game with the same probability. Else if $\mathcal{O}(\cdot) = \mathcal{O}_\$$, then the ciphertexts c_0, $\mathbf{c}$, $\{\mathbf{c}'_j\}_{j\in J_1^*}$, $\{\mathbf{c}''_{j,\theta}\}_{j\in J_1^*,\theta\in\mathsf{KUNodes}(\mathsf{BT}_j,\mathsf{RL}_j^*)}$, $\{\mathbf{c}_j\}_{j\in J_2^*}$ are uniform, thus the probability of $\mathcal{A}$ guesses the right b is exactly $1/2$. In a word, if $\mathcal{A}$ breaks the security of our DR-ABE, then $\mathcal{B}$ solves the underlying LWE problem. □

5.3 Parameters

The parameters are the same as those of Section 4.3.

6 Reducing the size of public key

Our DR-ABE scheme with user-level (*resp.* attribute-level) revocation has a relatively large public key, and its dependence on the number of users N in the system is due to the fact that each node θ in BT (*resp.* each BT_i) is associated with a uniform random matrix $\mathbf{D}_\theta \in \mathbb{Z}_q^{n\times m}$ (*resp.* $\mathbf{D}_{i,\theta} \in \mathbb{Z}_q^{n\times m}$). In fact, the size of the public key can be reduced in the random oracle model in a way similar to [21]: Let $\mathcal{H} : \{0,1\}^* \to \mathbb{Z}_q^{n\times m}$ be a random oracle. For each node θ in BT (*resp.* each BT_i), we obtain uniformly random matrix $\mathbf{D}_\theta$ (*resp.* $\mathbf{D}_{i,\theta}$) as

$\mathbf{D}_\theta := \mathcal{H}(\mathbf{A}, \{\mathbf{B}_j\}_{j\in\mathcal{I}}, \mathbf{u}, \theta)$ (*resp.* $\mathbf{D}_{i,\theta} := \mathcal{H}(\mathbf{A}, \{\mathbf{B}_j\}_{j\in\mathcal{I}}, \mathbf{u}, i, \theta)$). In the security proof, we first simulate the generation of $\mathbf{D}_\theta$ (*resp.* $\mathbf{D}_{i,\theta}$) as in the proof of Theorem 1 (*resp.* Theorem 2) then programs the random oracle $\mathcal{H}$ such that $\mathcal{H}(\mathbf{A}, \{\mathbf{B}_j\}_{j\in\mathcal{I}}, \mathbf{u}, \theta) = \mathbf{D}_\theta$ (*resp.* $\mathcal{H}(\mathbf{A}, \{\mathbf{B}_j\}_{j\in\mathcal{I}}, \mathbf{u}, i, \theta) = \mathbf{D}_{i,\theta}$).

7 Decryption outsourcing

To make our schemes more applicable for resource-limited end user, we modify our DR-ABE schemes to outsource most computational overhead of the end user to an honest-but-curious third party in the following manner: We add an extra dummy attribute *dummy* in the system. The Setup algorithm chooses an extra matrix $\bar{\mathbf{B}} \leftarrow \mathbb{Z}_q^{n\times m}$. To generate the private key for a user, the KGC splits the public vector $\mathbf{u}$ into $\bar{\mathbf{u}}, \hat{\mathbf{u}}$ such that $\mathbf{u} = \bar{\mathbf{u}} + \hat{\mathbf{u}}$, samples $\bar{\mathbf{e}} \leftarrow \mathsf{SampleLeft}(\mathbf{A}, \mathbf{B} + H(dummy)\mathbf{G}, \bar{\mathbf{u}}, \mathbf{T}_\mathbf{A}, \sigma)$, replaces $\mathbf{u}$ with $\hat{\mathbf{u}}$ in the original Keygen algorithm to get $sk_{S,\mathsf{id}}$, finally returns $sk_{S,\mathsf{id}}$ along with $\bar{\mathbf{e}}$ as the private key of the user. Moreover, we add an extra ciphertext corresponding with *dummy*, $\bar{\mathbf{c}} = (\mathbf{B} + H(dummy)\mathbf{G})^\top \mathbf{s} + D\bar{\mathbf{R}}^\top \mathbf{x}$, into the output of the original Enc algorithm. In this case, the end user can give $sk_{S,\mathsf{id}}$ to an untrusted third party to help decrypt the ciphertext except for $\bar{\mathbf{c}}$. The third party will return $\hat{\mathbf{u}}\mathbf{s} + \mathbf{e}$ and $\bar{\mathbf{c}}$ to the user, and the latter only need to deal with $\bar{\mathbf{c}}$ using $\bar{\mathbf{e}}$ to recover the message.

8 Conclusion

In this paper, we propose two directly revocable ciphertext-policy attribute-based encryption schemes from lattices. One achieves user-level user revocation, while the other achieves attribute-level user revocation. Both schemes inherit the main advantages of the direct revocation mechanism: the revocation list is defined by the message sender ; the authority does not need to generate and issue key update anymore. In addition, both schemes support flexible threshold access policies on multi-valued attributes. The size of public key of our schemes can be reduced in the random oracle model. Most part of the decryption work can be outsourced to a third party as well. Compared with other existing lattice-based revocable CP-ABE schemes, our schemes have reasonable security guarantee.

Acknowledgments

The authors are supported by National Cryptography Development Fund (Grant No. MMJJ20180210) and National Natural Science Foundation of China (Grant No. 61832012 and No. 61672019).

References

1. Shweta Agrawal, Dan Boneh, and Xavier Boyen. Efficient lattice (H)IBE in the standard model. In *Advances in Cryptology - EUROCRYPT 2010, 29th Annual*

International Conference on the Theory and Applications of Cryptographic Techniques, Monaco / French Riviera, May 30 - June 3, 2010. Proceedings, pages 553–572, 2010.

2. Shweta Agrawal, Xavier Boyen, Vinod Vaikuntanathan, Panagiotis Voulgaris, and Hoeteck Wee. Fuzzy identity based encryption from lattices. *IACR Cryptology ePrint Archive*, 2011:414, 2011.
3. Miklós Ajtai. Generating hard instances of the short basis problem. In *Automata, Languages and Programming, 26th International Colloquium, ICALP'99, Prague, Czech Republic, July 11-15, 1999, Proceedings*, pages 1–9, 1999.
4. Joël Alwen and Chris Peikert. Generating shorter bases for hard random lattices. *Theory Comput. Syst.*, 48(3):535–553, 2011.
5. Daniel Apon, Xiong Fan, and Feng-Hao Liu. Deniable attribute based encryption for branching programs from LWE. In *Theory of Cryptography - 14th International Conference, TCC 2016-B, Beijing, China, October 31 - November 3, 2016, Proceedings, Part II*, pages 299–329, 2016.
6. Nuttapong Attrapadung and Hideki Imai. Conjunctive broadcast and attribute-based encryption. In *Pairing-Based Cryptography - Pairing 2009, Third International Conference, Palo Alto, CA, USA, August 12-14, 2009, Proceedings*, pages 248–265, 2009.
7. Nuttapong Attrapadung, Benoît Libert, and Elie de Panafieu. Expressive key-policy attribute-based encryption with constant-size ciphertexts. In *Public Key Cryptography - PKC 2011 - 14th International Conference on Practice and Theory in Public Key Cryptography, Taormina, Italy, March 6-9, 2011. Proceedings*, pages 90–108, 2011.
8. John Bethencourt, Amit Sahai, and Brent Waters. Ciphertext-policy attribute-based encryption. In *2007 IEEE Symposium on Security and Privacy (S&P 2007), 20-23 May 2007, Oakland, California, USA*, pages 321–334, 2007.
9. Alexandra Boldyreva, Vipul Goyal, and Virendra Kumar. Identity-based encryption with efficient revocation. *IACR Cryptology ePrint Archive*, 2012:52, 2012.
10. Xavier Boyen. Attribute-based functional encryption on lattices. In *Theory of Cryptography - 10th Theory of Cryptography Conference, TCC 2013, Tokyo, Japan, March 3-6, 2013. Proceedings*, pages 122–142, 2013.
11. Xavier Boyen and Qinyi Li. Attribute-based encryption for finite automata from LWE. In *Provable Security - 9th International Conference, ProvSec 2015, Kanazawa, Japan, November 24-26, 2015, Proceedings*, pages 247–267, 2015.
12. David Cash, Dennis Hofheinz, Eike Kiltz, and Chris Peikert. Bonsai trees, or how to delegate a lattice basis. *J. Cryptology*, 25(4):601–639, 2012.
13. Jie Chen, Hoon Wei Lim, San Ling, Huaxiong Wang, and Khoa Nguyen. Revocable identity-based encryption from lattices. In *Information Security and Privacy - 17th Australasian Conference, ACISP 2012, Wollongong, NSW, Australia, July 9-11, 2012. Proceedings*, pages 390–403, 2012.
14. Ronald Cramer and Ivan Damgård. On the amortized complexity of zero-knowledge protocols. In *Advances in Cryptology - CRYPTO 2009, 29th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 16-20, 2009. Proceedings*, pages 177–191, 2009.
15. Craig Gentry, Chris Peikert, and Vinod Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In *Proceedings of the 40th Annual ACM Symposium on Theory of Computing, Victoria, British Columbia, Canada, May 17-20, 2008*, pages 197–206, 2008.

16. 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, CCS 2006, Alexandria, VA, USA, Ioctober 30 - November 3, 2006*, pages 89–98, 2006.
17. Susan Hohenberger and Brent Waters. Attribute-based encryption with fast decryption. In *Public-Key Cryptography - PKC 2013 - 16th International Conference on Practice and Theory in Public-Key Cryptography, Nara, Japan, February 26 - March 1, 2013. Proceedings*, pages 162–179, 2013.
18. Sonia Jahid, Prateek Mittal, and Nikita Borisov. Easier: encryption-based access control in social networks with efficient revocation. In *Proceedings of the 6th ACM Symposium on Information, Computer and Communications Security, ASIACCS 2011, Hong Kong, China, March 22-24, 2011*, pages 411–415, 2011.
19. Shuichi Katsumata, Takahiro Matsuda, and Atsushi Takayasu. Lattice-based revocable (hierarchical) IBE with decryption key exposure resistance. In *Public-Key Cryptography - PKC 2019 - 22nd IACR International Conference on Practice and Theory of Public-Key Cryptography, Beijing, China, April 14-17, 2019, Proceedings, Part II*, pages 441–471, 2019.
20. Huijie Lian, Qingxian Wang, and Guangbo Wang. Large universe ciphertext-policy attribute-based encryption with attribute level user revocation in cloud storage. *Int. Arab J. Inf. Technol.*, 17(1):107–117, 2020.
21. San Ling, Khoa Nguyen, Huaxiong Wang, and Juanyang Zhang. Revocable predicate encryption from lattices. In *Provable Security - 11th International Conference, ProvSec 2017, Xi'an, China, October 23-25, 2017, Proceedings*, pages 305–326, 2017.
22. Daniele Micciancio and Chris Peikert. Trapdoors for lattices: Simpler, tighter, faster, smaller. In *Advances in Cryptology - EUROCRYPT 2012 - 31st Annual International Conference on the Theory and Applications of Cryptographic Techniques, Cambridge, UK, April 15-19, 2012. Proceedings*, pages 700–718, 2012.
23. Juan Manuel González Nieto, Mark Manulis, and Dongdong Sun. Fully private revocable predicate encryption. In *Information Security and Privacy - 17th Australasian Conference, ACISP 2012, Wollongong, NSW, Australia, July 9-11, 2012. Proceedings*, pages 350–363, 2012.
24. Rafail Ostrovsky, Amit Sahai, and Brent Waters. Attribute-based encryption with non-monotonic access structures. In *Proceedings of the 2007 ACM Conference on Computer and Communications Security, CCS 2007, Alexandria, Virginia, USA, October 28-31, 2007*, pages 195–203, 2007.
25. Chris Peikert. Public-key cryptosystems from the worst-case shortest vector problem: extended abstract. In *Proceedings of the 41st Annual ACM Symposium on Theory of Computing, STOC 2009, Bethesda, MD, USA, May 31 - June 2, 2009*, pages 333–342, 2009.
26. Matthew Pirretti, Patrick Traynor, Patrick D. McDaniel, and Brent Waters. Secure attribute-based systems. In *Proceedings of the 13th ACM Conference on Computer and Communications Security, CCS 2006, Alexandria, VA, USA, Ioctober 30 - November 3, 2006*, pages 99–112, 2006.
27. Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. In *Proceedings of the 37th Annual ACM Symposium on Theory of Computing, Baltimore, MD, USA, May 22-24, 2005*, pages 84–93, 2005.
28. Amit Sahai, Hakan Seyalioglu, and Brent Waters. Dynamic credentials and ciphertext delegation for attribute-based encryption. In *Advances in Cryptology - CRYPTO 2012 - 32nd Annual Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2012. Proceedings*, pages 199–217, 2012.

29. Amit Sahai and Brent Waters. Fuzzy identity-based encryption. In *Advances in Cryptology - EUROCRYPT 2005, 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Aarhus, Denmark, May 22-26, 2005, Proceedings*, pages 457–473, 2005.
30. Jae Hong Seo and Keita Emura. Revocable identity-based encryption revisited: Security model and construction. In *Public-Key Cryptography - PKC 2013 - 16th International Conference on Practice and Theory in Public-Key Cryptography, Nara, Japan, February 26 - March 1, 2013. Proceedings*, pages 216–234, 2013.
31. Rotem Tsabary. Fully secure attribute-based encryption for t-cnf from LWE. In *Advances in Cryptology - CRYPTO 2019 - 39th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2019, Proceedings, Part I*, pages 62–85, 2019.
32. Hao Wang, Zhihua Zheng, Lei Wu, and Ping Li. New directly revocable attribute-based encryption scheme and its application in cloud storage environment. *Cluster Computing*, 20(3):2385–2392, 2017.
33. 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.
34. Brent Waters. Ciphertext-policy attribute-based encryption: An expressive, efficient, and provably secure realization. In *Public Key Cryptography - PKC 2011 - 14th International Conference on Practice and Theory in Public Key Cryptography, Taormina, Italy, March 6-9, 2011. Proceedings*, pages 53–70, 2011.
35. Zhiqian Xu and Keith M. Martin. Dynamic user revocation and key refreshing for attribute-based encryption in cloud storage. In *11th IEEE International Conference on Trust, Security and Privacy in Computing and Communications, TrustCom 2012, Liverpool, United Kingdom, June 25-27, 2012*, pages 844–849, 2012.
36. Kang Yang, Guohua Wu, Chengcheng Dong, Xingbing Fu, Fagen Li, and Ting Wu. Attribute based encryption with efficient revocation from lattices. *I. J. Network Security*, 22(1):161–170, 2020.
37. Jiang Zhang, Zhenfeng Zhang, and Aijun Ge. Ciphertext policy attribute-based encryption from lattices. In *7th ACM Symposium on Information, Compuer and Communications Security, ASIACCS '12, Seoul, Korea, May 2-4, 2012*, pages 16–17, 2012.