Lattice-based Revocable Identity-based Encryption with Bounded Decryption Key Exposure Resistance

Atsushi Takayasu[1,3] and Yohei Watanabe[2,3]

[1] The University of Tokyo, Tokyo, Japan,
[2] The University of Electro-Communications, Tokyo, Japan,
[3] National Institute of Advanced Industrial Science and Technology, Tokyo, Japan
takayasu@mist.i.u-tokyo.ac.jp

Abstract. A revocable identity-based encryption (RIBE) scheme, proposed by Boldyreva et al, provides a revocation functionality for managing a number of users dynamically and efficiently. To capture a realistic scenario, Seo and Emura introduced an additional important security notion, called *decryption key exposure resistance* (DKER), where an adversary is allowed to query short-term decryption keys. Although several RIBE schemes that satisfy DKER have been proposed, all the lattice-based RIBE schemes, e.g., Chen et al.'s scheme, do not achieve DKER, since they basically do not have *the key re-randomization property*, which is considered to be an essential requirement for achieving DKER. In particular, in every existing lattice-based RIBE scheme, an adversary can easily recover plaintexts if the adversary is allowed to issue even a single short-term decryption key query. In this paper, we propose a new lattice-based RIBE scheme secure against exposure of a-priori bounded number of decryption keys (for every identity). We believe that this bounded notion is still meaningful and useful from a practical perspective. Technically, to achieve the bounded security without the key re-randomization property, key updates in our scheme are short vectors whose corresponding syndrome vector changes in each time period. For this approach to work correctly and for the scheme to be secure, *cover free families* play a crucial role in our construction.

1 Introduction

1.1 Background

Identity-based encryption (IBE) is currently one of the central cryptographic primitives. IBE allows any strings to be used as public keys, and therefore is an advanced form of public-key encryption (PKE). The first practical IBE was proposed by Boneh and Franklin [9] from bilinear groups. Since then, several IBE schemes have been proposed including ones from lattices [1, 6, 10, 11, 14, 20, 24, 38–40]. Lattice-based schemes are believed to resist quantum attacks and the average-case security is guaranteed by the worst-case lattice assumptions.

Although IBE is known as an important cryptographic primitive, IBE itself has not been used that much than PKE in modern society. One main reason for the situation is inefficient revocation procedures of ordinary IBE schemes. Revocation functionality is necessary to handle users in cryptographic schemes since malicious users should be immediately driven out from the schemes, and even honest users should be revoked if their keys get compromised. In the PKE setting, the validity of public keys are guaranteed by certificates issued by public-key infrastructures (PKIs). Therefore, users can be easily revoked by invalidating the corresponding certificate. On the other hand, IBE does not have such a revocation procedure due to the absence of PKIs. Boneh and Franklin [9] mentioned the following naive and inefficient revocation procedure: The lifetime of the system is divided into discrete time periods. In every time period, a key generation center (KGC) generates secret keys for each non-revoked user and sends the new keys to the corresponding users.

Later, Boldyreva et al. [7] proposed a pairing-based IBE scheme with efficient revocation, which is called a *revocable IBE* (RIBE) scheme by utilizing the spirit of fuzzy IBE constructions and a subset cover framework called the complete subtree (CS) method. They significantly improved the efficiency of revocation procedures from linear to logarithmic in the number of all users. Specifically, they considered two kinds of keys: a *long-term secret key* and a *short-term decryption key*. In every time-period, the KGC generates update information called a *key update*, and broadcasts it. Each non-revoked user can derive a decryption key for each time period from his long-term secret key and the key update for the corresponding time period, while revoked users cannot compute their decryption keys. After the proposal, Libert and Vergnaud [28] proposed the first adaptively secure pairing-based RIBE scheme. The first lattice-based RIBE scheme was proposed by Chen et al. [16] in the selective security model. The idea of these constructions follow Boldyreva et al.'s one.

Human errors seem never to be eliminated, and therefore a key exposure problem is unavoidable. In the context of RIBE, Seo and Emura [34] pointed out that Boldyreva et al.'s security model did not capture such a realistic threat, and they first realized an RIBE scheme with *decryption-key exposure resistance* (DKER) from bilinear groups. An RIBE scheme with DKER, DKER RIBE for short, guarantees that the security is not compromised even if polynomially many short-term decryption keys are leaked. Boneh-Franklin's naive solution captures DKER, whereas the previous RIBE schemes [7, 28, 16] are vulnerable against decryption key exposure. Hence, DKER seems to be a natural security requirement for RIBE. Although the construction idea is almost the same as Boldyreva et al.'s one [7], Seo and Emura [34] made use of *the key re-randomization property* for proving the stronger security (i.e., security with DKER). Since the proposal, DKER has become the standard security notion of RIBE. Indeed, several DKER RIBE schemes [18, 23, 26, 27, 33, 35, 36] have been proposed.

However, no lattice-based DKER RIBE schemes have been proposed thus far; existing lattice-based RIBE schemes [16, 17, 30][4]do not satisfy DKER. In particular, Chen et al.'s RIBE scheme immediately becomes insecure even with an adversary's single short-term decryption key query. Hence, the limitation does not stem from proof techniques. Actually, all the existing DKER RIBE schemes satisfy the key re-randomization property, which is used for preventing adversaries from obtaining critical information from decryption key queries. Since the current lattice-based RIBE construction does not satisfy the property, we should explore new approaches to construct DKER RIBE schemes.

1.2 Our Contributions

In this paper, we construct the first lattice-based RIBE scheme that is resilient to decryption key exposure. To be precise, we should note that our scheme is secure when adversaries are allowed to query a-priori bounded number of short-term decryption keys, which is denoted by Q, for the target identity. Therefore, we call our proposal *B-DKER RIBE*, where B-DKER stands for *bounded DKER*. Decryption key exposure is mainly caused by human errors. The leakage might happen, however we can assume that it rarely happens. Hence, although the security of B-DKER RIBE is weaker than DKER RIBE, we believe that our security model is sufficient for practical use. Even if a number of decryption keys are exposed, our scheme is secure by setting sufficiently large Q, whereas Chen et al.'s scheme is insecure in such a case (in particular, even in the case that $Q = 1$).

One may think that the notion of B-DKER RIBE is similar to that of bounded-collusion IBE [21] or k-resilient IBE [22]. However, we emphasize that there is a major gap between them from the practical aspect. In the bounded-collusion IBE, the number of secret key extraction queries is a-priori bounded, whereas our definition allows unbounded collusion, i.e., an adversary can unboundedly issue secret key extraction queries and decryption key queries except for the target identity. Practically, in the bounded-collusion IBE scenario, an adversary might collude with the larger number of users than the a-priori bounded number. The KGC may be unaware of the behind-the-scenes collusion, and thus the system would not be refreshed before breaking it. On the one hand, in the B-DKER RIBE scenario, it would appear that decryption key exposures happen only through human errors or some accident. That is, the leakage cannot be controlled by adversaries. The KGC may notice the fact of leakage from users who are honest but leaked their keys, and therefore the KGC can keep the scheme secure by refreshing it at some point.

To obtain (a kind of) DKER for lattice-based RIBE is the main contribution of this paper. Our scheme has a different flavor from the template RIBE

[4] Cheng and Zhang [17] proposed the first adaptively secure lattice-based RIBE scheme, however, their security proofs contain unavoidable bugs. Therefore, there are no adaptively secure lattice-based RIBE schemes even without DKER. See Section 6 for the detail.

construction due to Boldyreva et al. [7] (and therefore Chen et al. [16]) in the sense that each key update corresponds to distinct syndrome vectors in each time period. Although the modification causes several troubles, cover free families (CFFs) enable us to resolve them with longer secret keys (see Section 1.3 for details). For simplicity, we discuss our construction in the selective security model throughout the paper. We believe that it enables readers to understand our technique easily. In addition, as side contributions, we obtain the following improvements although they are not very technical: smaller parameters by utilizing Micciancio-Peikert's gadget matrix [29], the first semi-adaptively secure lattice-based RIBE, the first anonymous RIBE scheme that is resilient to decryption key exposure. They will be discussed in Section 6. Notice that in the semi-adaptive security model, the adversary issues the challenge identity and the challenge time period just after receiving a public parameter.

1.3 Our Approach

In this section, we show a brief overview of Chen et al.'s lattice-based RIBE construction [16] and our modification to the scheme to achieve B-DKER. The public parameter of Chen et al.'s RIBE scheme consists of three matrices $\mathbf{A}_0, \mathbf{A}_1, \mathbf{A}_2$ and a syndrome vector $\mathbf{u}$ along with a gadget matrix[5] $\mathbf{G}$ that was introduced in [29]. The ciphertext of a plaintext $M \in \{0,1\}$ for an identity ID and a time period T is

$$[\mathbf{A}_0|\mathbf{A}_1 + H(\mathsf{ID})\mathbf{G}|\mathbf{A}_2 + H(\mathsf{T})\mathbf{G}]^T \mathbf{s} + \mathsf{noise} \quad \text{and} \quad \mathbf{u}^T\mathbf{s} + M\left\lfloor \frac{q}{2} \right\rfloor + \mathsf{noise}$$

where $\mathbf{s}$ is a random secret vector and $H()$ is a public hash function. Each user has a long-term secret key $\mathbf{e}'$ whereas KGC broadcasts a key update $\tilde{\mathbf{e}}$ in each time period such that

$$[\mathbf{A}_0|\mathbf{A}_1 + H(\mathsf{ID})\mathbf{G}]\,\mathbf{e}' = \mathbf{u}' \qquad \text{and} \qquad [\mathbf{A}_0|\mathbf{A}_2 + H(\mathsf{T})\mathbf{G}]\,\tilde{\mathbf{e}} = \tilde{\mathbf{u}} \qquad (1)$$

for some random syndrome vectors $\mathbf{u}'$ and $\tilde{\mathbf{u}}$. If the user is non-revoked, these two syndrome vectors satisfy $\mathbf{u}' + \tilde{\mathbf{u}} = \mathbf{u}$. The short-term decryption key $\mathbf{e}$ for $(\mathsf{ID}, \mathsf{T})$ is their concatenation $\mathbf{e} := (\mathbf{e}', \tilde{\mathbf{e}})$.

As opposed to an ordinary IBE, the RIBE simulator should create a long-term secret key $\mathbf{e}'$ for the target identity ID^* and a key update $\tilde{\mathbf{e}}$ for the challenge time period T^*. Chen et al. resolved the problem by utilizing a Gaussian sampling algorithm in a clever way. If we do not care about DKER, the simulator should create either a secret key $\mathbf{e}'$ for the target identity ID^* or a key update $\tilde{\mathbf{e}}$ for the target time period T^*. Then, the simulator picks $\mathbf{e}'$ or $\tilde{\mathbf{e}}$ in advance and sets $\mathbf{u}'$ or $\tilde{\mathbf{u}}$ according to the equation (1). Notice that the simulator can create long-term secret keys and key updates for all the other $\mathsf{ID} \neq \mathsf{ID}^*$ and $\mathsf{T} \neq \mathsf{T}^*$ since it has a trapdoor.

[5] Although the gadget matrix was not used by Chen et al. [16], it is well known that the parameters can be reduced by utilizing the matrix.

In short, to obtain DKER, the challenge ciphertext for the target $(\mathsf{ID}^*, \mathsf{T}^*)$ should not be decrypted by using a key update for T^* and short-term decryption keys for $(\mathsf{ID}^*, \mathsf{T})$ such that $\mathsf{T} \neq \mathsf{T}^*$. However, since Chen et al.'s short-term decryption key is a simple concatenation, the target decryption key for $(\mathsf{ID}^*, \mathsf{T}^*)$ can be recovered even with a single decryption key for $(\mathsf{ID}^*, \mathsf{T})$. Since there is a concrete attack, the limitation is not due to proof techniques but the construction. In other words, the simulator should create both short-term decryption keys $\mathbf{e}'$ for $(\mathsf{ID}^*, \mathsf{T})$ such that $\mathsf{T} \neq \mathsf{T}^*$ and key updates $\tilde{\mathbf{e}}$ for T^* during the simulation. However, once the simulator uses a Gaussian sampling algorithm and sets $\mathbf{e}'$, the corresponding syndrome $\mathbf{u}'$ is fixed. Then, the simulator cannot create $\tilde{\mathbf{e}}$ for $\tilde{\mathbf{u}}$ such that $\mathbf{u}' + \tilde{\mathbf{u}} = \mathbf{u}$. If lattice-based RIBE scheme supports the key re-randomization property, we can avoid the problem as Seo-Emura [34], however, it does not. We will discuss the fact in Section 6.

To resolve the problem, we employ a novel RIBE construction. A starting point of our modification is that our key update $\tilde{\mathbf{e}}$ for a time period T satisfies

$$[\mathbf{A}_0 | \mathbf{A}_2 + H(\mathsf{T})\mathbf{G}]\, \tilde{\mathbf{e}} = \tilde{\mathbf{u}}_{\mathsf{T}}$$

where the corresponding syndrome vector $\tilde{\mathbf{u}}_{\mathsf{T}}$ changes in each time period. The property directly suggests that decryption keys for $(\mathsf{ID}^*, \mathsf{T})$ such that $\mathsf{T} \neq \mathsf{T}^*$ are useless to recover a decryption key for the target $(\mathsf{ID}^*, \mathsf{T}^*)$. However, a new problem occurs by the construction. Since a secret key $\mathbf{e}'$ corresponds to a fixed syndrome vector $\mathbf{u}'$, even non-revoked users cannot derive well-formed decryption keys such that $\mathbf{u}' + \tilde{\mathbf{u}}_{\mathsf{T}} = \mathbf{u}$ for all time periods with their secret keys and key updates. To overcome the issue, in our scheme, each user ID has multiple d secret keys $\mathbf{e}'_1, \ldots, \mathbf{e}'_d$ such that

$$[\mathbf{A}_0 | \mathbf{A}_1 + H(\mathsf{ID})\mathbf{G}]\, \mathbf{e}'_1 = \mathbf{u}'_1, \ldots, [\mathbf{A}_0 | \mathbf{A}_1 + H(\mathsf{ID})\mathbf{G}]\, \mathbf{e}'_d = \mathbf{u}'_d.$$

A naive approach for the scheme to work correctly is that we use each $\mathbf{e}'_\ell$ in each time period. However, the modification makes the scheme too inefficient since the number of secret keys d has to be at least larger than the maximum time period and results in super-polynomial. To reduce the size, we set $\mathbf{u} - \tilde{\mathbf{u}}_{\mathsf{T}}$ as a subset sum of $\mathbf{u}'_1, \ldots, \mathbf{u}'_d$ so that non-revoked users can produce well-formed decryption keys with smaller d. The resulting decryption key is a concatenation of the corresponding subset sum of $\mathbf{e}'_1, \ldots, \mathbf{e}'_d$ and the key update $\tilde{\mathbf{e}}$. The simulator utilizes a Gaussian sampling algorithm to create $d-1$ secret key elements $\mathbf{e}'_1, \ldots, \mathbf{e}'_d$ except $\mathbf{e}'_{\ell^*}$ for ID^* and a key update $\tilde{\mathbf{e}}$ for T^* along with their corresponding syndrome vectors, then answers decryption key queries for $(\mathsf{ID}^*, \mathsf{T})$ such that $\mathsf{T} \neq \mathsf{T}^*$. The remaining syndrome vector $\mathbf{u}'_{\ell^*}$ is directly fixed. If $\mathbf{e}'_{\ell^*}$ is not used to answer Q decryption key queries, the approach goes well.

For the above construction to become a provably secure practical RIBE scheme whose adversary is allowed to query Q decryption keys, there are the following three requirements: (1) the number of secret keys d is at most polynomially bounded, (2) a subset sum of $\mathbf{u}_1, \ldots, \mathbf{u}_d$ produces distinct vectors whose number is larger than the maximum time period, (3) there is at least one secret key $\mathbf{e}'_{\ell^*}$ that is not used to answer arbitrary Q decryption key queries. Therefore, we use CFFs so that the resulting scheme satisfies all the above requirements.

2 Preliminaries

Notation, "Probabilistic polynomial-time" is abbreviated as "PPT". We denote $[a, b]$ by a set $\{a, a+1, \ldots, b\}$ for any integers $a, b \in \mathbb{N}$ such that $a \leq b$. We sometimes write $[d]$ as $[1, d]$ for simplicity. Let a bold capital $\mathbf{A}$ and a bold lower $\mathbf{a}$ denote a matrix and a column vector respectively. Let $\mathbf{A}^T$ and $\mathbf{a}^T$ denote their transposes. If we write $(y_1, y_2, \ldots, y_m) \leftarrow \mathcal{A}(x_1, x_2, \ldots, x_n)$ for an algorithm $\mathcal{A}$ having n inputs and m outputs, it means to input $x_1, x_2, \ldots, x_n$ into $\mathcal{A}$ and to get the resulting output $y_1, y_2, \ldots, y_m$. We write $(y_1, y_2, \ldots, y_m) \leftarrow \mathcal{A}^{\mathcal{O}}(x_1, x_2, \ldots, x_n)$ to indicate that an algorithm $\mathcal{A}$ that is allowed to access an oracle $\mathcal{O}$ takes $x_1, x_2, \ldots, x_n$ as input and outputs $(y_1, y_2, \ldots, y_m)$. If $\mathcal{X}$ is a set, we write $x \xleftarrow{\$} \mathcal{X}$ to mean the operation of picking an element x of $\mathcal{X}$ uniformly at random. We use λ as a security parameter. For sufficiently large λ, a function $\mathsf{negl} : \mathbb{R} \to \mathbb{R}$ is negligible if $\mathsf{negl}(\lambda) < 1/p(\lambda)$ for any polynomial $p(\lambda)$. Let X and Y be two random variables taking values in some finite set Ω. Statistical distance is defined as $\Delta(X; Y)$, as $\Delta(X; Y) := \frac{1}{2} \sum_{s \in \Omega} |\Pr[X = s] - \Pr[Y = s]|$. For sets of random variables X and Y, we say that X and Y are statistically close if $\Delta(X; Y)$ is negligible.

Cover Free Families. We define a cover free family (CFF), which is a core building block in our construction, as follows.

Definition 1 (Cover Free Families [19]). *Let α, d, Q be positive integers, and $\mathcal{F} := \{\mathcal{F}_\mu\}_{\mu \in [\alpha]}$ be a family of subsets of $[d]$, where every $|\mathcal{F}_\mu| = w$. $\mathcal{F}$ is said to be w-uniform Q-cover-free if it holds that $\bigcup_{j=1}^{Q} \mathcal{F}_{i_j} \not\supset \mathcal{F}_{i_{Q+1}}$ for any $\mathcal{F}_{i_1}, \mathcal{F}_{i_2}, \ldots, \mathcal{F}_{i_{Q+1}} \in \mathcal{F}$ such that $\mathcal{F}_{i_k} \neq \mathcal{F}_{i_\ell}$ for any distinct $k, \ell \in [Q+1]$.*

Lemma 1 ([25]). *There is a deterministic polynomial time algorithm $\mathsf{CFF.Gen}$ that, on input of positive integers α and Q, returns $d \in \mathbb{N}$ and a family $\mathcal{F} = \{\mathcal{F}_\mu\}_{\mu \in [\alpha]}$, such that $\mathcal{F}$ is Q-cover free over $[d]$ and w-uniform, where $d \leq 16Q^2 \log \alpha$ and $w = d/4Q$.*

KUNode Algorithm. To reduce costs of a revocation process, we use a binary tree structure and apply the following KUNode algorithm as in the previous RIBE schemes [7, 28, 34]. $\mathsf{KUNode}(\mathsf{BT}, \mathsf{RL}, \mathsf{T})$ takes as input a binary tree BT, a revocation list RL, and a time period $\mathsf{T} \in \mathcal{T}$, and outputs a set of nodes. When η is a non-leaf node, then we write η_L and η_R as the left and right child of η, respectively. When η is a leaf node, $\mathsf{Path}(\mathsf{BT}, \eta)$ denotes the set of nodes on the path from η to the root. Each user is assigned to a leaf node. If a user who is assigned to η is revoked on a time period $\mathsf{T} \in \mathcal{T}$, then $(\eta, \mathsf{T}) \in \mathsf{RL}$. $\mathsf{KUNode}(\mathsf{BT}, \mathsf{RL}, \mathsf{T})$ is executed as follows. It sets $\mathcal{X} := \emptyset$ and $\mathcal{Y} := \emptyset$. For any $(\eta_i, \mathsf{T}_i) \in \mathsf{RL}$, if $\mathsf{T}_i \leq \mathsf{T}$ then it adds $\mathsf{Path}(\mathsf{BT}, \eta_i)$ to $\mathcal{X}$ (i.e., $\mathcal{X} := \mathcal{X} \cup \mathsf{Path}(\mathsf{BT}, \eta_i)$). Then, for any $\eta \in \mathcal{X}$, if $\eta_L \notin \mathcal{X}$, then it adds η_L to $\mathcal{Y}$. If $\eta_R \notin \mathcal{X}$, then it adds η_R to $\mathcal{Y}$. Finally, it outputs $\mathcal{Y}$ if $\mathcal{Y} \neq \emptyset$. If $\mathcal{Y} = \emptyset$, then it adds root to $\mathcal{Y}$ and outputs $\mathcal{Y}$.

Lattices. An m-dimensional integer lattice is an additive discrete subgroup of $\mathbb{Z}^m$. For positive integers q, n, m, a matrix $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$, and a vector $\mathbf{u} \in \mathbb{Z}_q^m$,

the m-dimensional integer (shifted) lattices $\Lambda_q^{\perp}(\mathbf{A})$ and $\Lambda_q^{\mathbf{u}}(\mathbf{A})$ are defined as $\Lambda_q^{\perp}(\mathbf{A}) := \{\mathbf{e} \in \mathbb{Z}^m : \mathbf{Ae} = \mathbf{0} \bmod q\}, \Lambda_q^{\mathbf{u}}(\mathbf{A}) := \{\mathbf{e} \in \mathbb{Z}^m : \mathbf{Ae} = \mathbf{u} \bmod q\}$. The lattice $\Lambda_q^{\mathbf{u}}(\mathbf{A})$ is a shift of the lattice $\Lambda_q^{\perp}(\mathbf{A})$; if $\mathbf{t} \in \Lambda_q^{\mathbf{u}}(\mathbf{A})$ then $\Lambda_q^{\mathbf{u}}(\mathbf{A}) = \Lambda_q^{\perp}(\mathbf{A}) + \mathbf{t}$. Let $\mathbf{T_A} \in \mathbb{Z}^{m \times m}$ be a basis of a lattice $\Lambda_q^{\perp}(\mathbf{A})$. Then $\mathbf{T_A} \in \mathbb{Z}^{m \times m}$ is also a basis of a lattice $\Lambda_q^{\perp}(\mathbf{HA})$ for a full rank $\mathbf{H} \in \mathbb{Z}_q^{n \times n}$.

Matrix Norms. For a vector $\mathbf{u}$, we let $\|\mathbf{u}\|$ denote its L_2 norm. For a matrix $\mathbf{R} \in \mathbb{Z}^{k \times m}$, we define the following three norms:

- $\|\mathbf{R}\|$ denotes the L_2 length of the longest column of $\mathbf{R}$.
- $\|\mathbf{R}\|_{\mathsf{GS}} = \|\tilde{\mathbf{R}}\|$ where $\tilde{\mathbf{R}}$ is the Gram-Schmidt orthogonalization of $\mathbf{R}$.
- $\|\mathbf{R}\|_2$ is defined as $\|\mathbf{R}\|_2 = \sup_{\|\mathbf{x}\|=1} \|\mathbf{Rx}\|$.

Note that $\|\mathbf{R}\|_{\mathsf{GS}} \leq \|\mathbf{R}\| \leq \|\mathbf{R}\|_2 \leq \sqrt{k}\|\mathbf{R}\|$ and that $\|\mathbf{R} \cdot \mathbf{S}\|_2 \leq \|\mathbf{R}\|_2 \cdot \|\mathbf{S}\|_2$.

Gaussian Distributions. Let $\mathcal{D}_{\Lambda,\sigma,\mathbf{c}}$ denote the discrete gaussian distribution over Λ with center $\mathbf{c}$ and a parameter σ. If $\mathbf{c} = 0$, we omit the subscript and denote $\mathcal{D}_{\Lambda,\sigma}$. We summarize some basic properties of discrete Gaussian distributions.

Lemma 2 ([20]). *Let Λ be an m-dimensional lattice. Let $\mathbf{T}$ be a basis for Λ, and suppose $\sigma \geq \|\mathbf{T}\|_{\mathsf{GS}} \cdot \omega(\sqrt{\log m})$. Then* $\Pr[\|\mathbf{x}\| > \sigma\sqrt{m} : \mathbf{x} \leftarrow \mathcal{D}_{\Lambda,\sigma}] \leq \mathsf{negl}(m)$.

Lemma 3 ([20]). *Let n and q be positive integers with q prime, and let $m \geq 2n \log q$. Then for all but a $2q^{-n}$ fraction of all $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ and for any $\sigma \geq \omega(\sqrt{\log m})$, the distribution of $\mathbf{u} = \mathbf{Ae} \bmod q$ is statistically close to uniform over $\mathbb{Z}_q^n$ where $\mathbf{e} \leftarrow \mathcal{D}_{\mathbb{Z}^m,\sigma}$. Furthermore, the conditional distribution of $\mathbf{e}$ given $\mathbf{Ae} = \mathbf{u} \bmod q$ is exactly $\mathcal{D}_{\Lambda_q^{\mathbf{u}}(\mathbf{A}),\sigma}$.*

Sampling Algorithms.

Lemma 4. *Let $n, m, q > 0$ be positive integers with q prime. There are probabilistic polynomial time algorithms such that*

- *([13]):* $\mathsf{SampleGaussian}(\mathbf{T}, \sigma) \rightarrow \mathbf{e}$
  a randomized algorithm that, given a basis $\mathbf{T}$ for an m-dimensional lattice Λ and a parameter $\sigma \geq \|\mathbf{T}\|_{\mathsf{GS}} \cdot \omega(\sqrt{\log m})$ as inputs, then outputs $\mathbf{e}$ which is distributed according to $\mathcal{D}_{\Lambda,\sigma}$.
- *([4, 5, 29]):* $\mathsf{TrapGen}(q, n, m) \rightarrow (\mathbf{A}, \mathbf{T_A})$
  a randomized algorithm that, when $m \geq 2n\lceil \log q \rceil$, outputs a full rank matrix $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ and a basis $\mathbf{T_A} \in \mathbb{Z}^{m \times m}$ for $\Lambda_q^{\perp}(\mathbf{A})$ such that $\mathbf{A}$ is statistically close to uniform and $\|\mathbf{T_A}\|_{\mathsf{GS}} = O(\sqrt{n \log q})$ with overwhelming probability in n.
- *([14]):* $\mathsf{SampleLeft}(\mathbf{A}, \mathbf{F}, \mathbf{u}, \mathbf{T_A}, \sigma) \rightarrow \mathbf{e}$
  a randomized algorithm that, given a full rank matrix $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$, a matrix $\mathbf{F} \in \mathbb{Z}_q^{n \times m}$, a vector $\mathbf{u} \in \mathbb{Z}_q^n$, a basis $\mathbf{T_A}$ for $\Lambda_q^{\perp}(\mathbf{A})$, and a Gaussian parameter $\sigma > \|\mathbf{T_A}\|_{\mathsf{GS}} \cdot \omega(\sqrt{\log m})$ as inputs, then outputs a vector $\mathbf{e} \in \mathbb{Z}_q^{2m}$ sampled from a distribution that is statistically close to $\mathcal{D}_{\Lambda_q^{\mathbf{u}}(\mathbf{A}|\mathbf{F}),\sigma}$.

- *([1]):* $\mathsf{SampleRight}(\mathbf{A}, \mathbf{G}, \mathbf{R}, \mathbf{u}, \mathbf{T_G}, \sigma) \to \mathbf{e}$ *where* $\mathbf{F} = \mathbf{AR} + \mathbf{G}$ *a randomized algorithm that, given full rank matrices* $\mathbf{A}, \mathbf{G} \in \mathbb{Z}_q^{n \times m}$*, a matrix* $\mathbf{R} \in \mathbb{Z}^{m \times m}$*, a vector* $\mathbf{u} \in \mathbb{Z}_q^n$*, a basis* $\mathbf{T_G}$ *of* $\Lambda_q^\perp(\mathbf{G})$*, and a Gaussian parameter* $\sigma > \|\mathbf{T_G}\|_{\mathsf{GS}} \cdot \|\mathbf{R}\| \cdot \omega(\sqrt{\log m})$ *as inputs, then outputs a vector* $\mathbf{e} \in \mathbb{Z}_q^{2m}$ *sampled from a distribution that is statistically close to* $\mathcal{D}_{\Lambda_q^{\mathbf{u}}(\mathbf{A}|\mathbf{F}),\sigma}$.
- *([29]): Let* $m > n\lceil \log q \rceil$*. Then there is 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* $\mathbf{T_G} \in \mathbb{Z}_q^{m \times m}$ *with* $\|\mathbf{T_G}\|_{\mathsf{GS}} \leq \sqrt{5}$.

We sometimes call $\mathbf{G}$ a gadget matrix that enables us to reduce several parameters. We use $\mathsf{SampleGaussian}(\mathbf{T}, \sigma)$ only for sampling a distribution $\mathcal{D}_{\mathbb{Z}^m,\sigma}$. For the purpose, we always use a standard basis for $\mathbb{Z}^m$ as $\mathbf{T}$. Hence, we omit the basis and write $\mathsf{SampleGaussian}(\sigma)$ throughout the paper.

To obtain a lower bound of σ, we will use the following fact.

Lemma 5 ([1]). *Let* $\mathbf{R}$ *be a* $m \times m$ *matrix chosen at random from* $\{-1, 1\}^{m \times m}$*. Then there is a universal constant* C *such that* $\Pr[\|\mathbf{R}\| > C\sqrt{m}] < e^{-m}$.

Randomness Extraction.

Lemma 6 ([1]). *Suppose that* $m > (n+1)\log_2 q + \omega(\log n)$ *and that* $q > 2$ *is prime. Let* $\mathbf{R}$ *be an* $m \times k$ *matrix chosen uniformly in* $\{-1, 1\}^{m \times k}$ *where* $k = k(n)$ *is polynomial in* n*. Let* $\mathbf{A}$ *and* $\mathbf{B}$ *be matrices chosen uniformly in* $\mathbb{Z}_q^{n \times m}$ *and* $\mathbb{Z}_q^{n \times k}$ *respectively. Then, for all vectors* $\mathbf{e} \in \mathbb{Z}_q^m$*, the distribution* $(\mathbf{A}, \mathbf{AR}, \mathbf{R}^T\mathbf{e})$ *is statistically close to the distribution* $(\mathbf{A}, \mathbf{B}, \mathbf{R}^T\mathbf{e})$.

Encoding Identities as Matrices.

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

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

Learning with Errors (LWE). For $\alpha \in (0, 1)$ and an integer $q > 2$, let $\bar{\Psi}_\alpha$ denote the probability distribution over $\mathbb{Z}_q$ obtained by choosing $x \in \mathbb{R}$ according to the normal distribution with mean 0 and standard deviation $\alpha/2\sqrt{\pi}$, then output $\lfloor qx \rceil$. The security of our RIBE scheme is reduced to the following LWE assumption.

Assumption 1 (Learning with Errors (LWE) Assumption [32]) *For integers* $n, m = m(n)$*,* $\alpha \in (0, 1)$ *such that a prime* $q = q(n) > 2$ *and* $\alpha q > 2\sqrt{n}$*, define the distribution:* $\mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n \times m}, \mathbf{s} \xleftarrow{\$} \mathbb{Z}_q^n, \mathbf{x} \xleftarrow{\$} \bar{\Psi}_\alpha^m, \mathbf{v} \xleftarrow{\$} \mathbb{Z}_q^m$*. We assume that for any PPT algorithm* $\mathcal{A}$ *(with output in* $\{0, 1\}$*),* $Adv_{\mathcal{A}}^{LWE} := \left|\Pr\left[\mathcal{A}(\mathbf{A}, \mathbf{A}^T\mathbf{s} + \mathbf{x}) = 1\right] - \Pr\left[\mathcal{A}(\mathbf{A}, \mathbf{v}) = 1\right]\right|$ *is negligible in the security parameter* n.

Regev [32] showed that (through a quantum reduction) the LWE problem is as hard as approximating the worst-case GapSVP to $\tilde{O}(n/\alpha)$ factors. Peikert [31], Brakerski et al. [13] showed analogous results through classical reductions.

3 B-DKER RIBE

$\mathcal{M}$, $\mathcal{I}$, and $\mathcal{T}$ denote sets of plaintexts, IDs, and time-periods, respectively. Throughout this paper, we consider a single bit scheme, i.e., $\mathcal{M} := \{0, 1\}$.

An RIBE scheme Π consists of seven-tuple algorithms (SetUp, PKG, KeyUp, DKG, Enc, Dec, Revoke) defined as follows:

- $(\mathsf{PP}, \mathsf{MK}, \mathsf{RL}, \mathsf{st}) \leftarrow \mathsf{SetUp}(\lambda, N)$: A probabilistic algorithm for setup. It takes a security parameter λ and the number of users N as input and outputs a public parameter PP, a master secret key MK, an initial revocation list $\mathsf{RL} = \emptyset$ and a state st.
- $(\mathsf{SK}_{\mathsf{ID}}, \mathsf{st}) \leftarrow \mathsf{PKG}(\mathsf{PP}, \mathsf{MK}, \mathsf{ID}, \mathsf{st})$: An algorithm for private key generation. It takes PP, MK, an identity $\mathsf{ID} \in \mathcal{I}$, and st as input and outputs a secret key $\mathsf{SK}_{\mathsf{ID}}$ and updated state information st.
- $\mathsf{KU}_{\mathsf{T}} \leftarrow \mathsf{KeyUp}(\mathsf{PP}, \mathsf{MK}, \mathsf{T}, \mathsf{RL}, \mathsf{st})$: An algorithm for key update generation. It takes PP, MK, a time-period $\mathsf{T} \in \mathcal{T}$, a current revocation list RL, and state st as input, and then outputs a key update KU_{T}.
- $\mathsf{DK}_{\mathsf{ID},\mathsf{T}}$ or $\perp \leftarrow \mathsf{DKG}(\mathsf{PP}, \mathsf{SK}_{\mathsf{ID}}, \mathsf{KU}_{\mathsf{T}})$: A probabilistic algorithm for decryption key generation. It takes PP, $\mathsf{SK}_{\mathsf{ID}}$ and KU_{T} as input and then outputs a decryption key $\mathsf{DK}_{\mathsf{ID},\mathsf{T}}$ at T or $\perp$ if ID has been revoked by T.
- $\mathsf{CT}_{\mathsf{ID},\mathsf{T}} \leftarrow \mathsf{Enc}(\mathsf{PP}, \mathsf{ID}, \mathsf{T}, M)$: A probabilistic algorithm for encryption. It takes PP, $\mathsf{ID} \in \mathcal{I}$, and $\mathsf{T} \in \mathcal{T}$, and a plaintext $M \in \mathcal{M}$ as input and then outputs a ciphertext $\mathsf{CT}_{\mathsf{ID},\mathsf{T}}$.
- M or $\perp \leftarrow \mathsf{Dec}(\mathsf{PP}, \mathsf{DK}_{\mathsf{ID},\mathsf{T}}, \mathsf{CT}_{\mathsf{ID},\mathsf{T}})$: A deterministic algorithm for decryption. It takes PP, $\mathsf{DK}_{\mathsf{ID},\mathsf{T}}$ and $\mathsf{CT}_{\mathsf{ID},\mathsf{T}}$ as input and then outputs M or $\perp$.
- $\mathsf{RL} \leftarrow \mathsf{Revoke}(\mathsf{PP}, \mathsf{ID}, \mathsf{T}, \mathsf{RL}, \mathsf{st})$: An algorithm for revocation. It takes $(\mathsf{ID}, \mathsf{T}) \in \mathcal{I} \times \mathcal{T}$, the current revocation list RL, and a state st as input and then outputs an updated revocation list RL.

In the above model, we assume that Π meets the following correctness property: For all security parameter $\lambda \in \mathbb{N}$, all $(\mathsf{PP}, \mathsf{MK}, \mathsf{RL}, \mathsf{st}) \leftarrow \mathsf{SetUp}(\lambda, N)$, all $M \in \mathcal{M}$, all $\mathsf{ID} \in \mathcal{I}$, all $\mathsf{T} \in \mathcal{T}$, if ID has not been revoked by $\mathsf{T} \in \mathcal{T}$, it holds that $M = \mathsf{Dec}(\mathsf{DKG}(\mathsf{PP}, \mathsf{PKG}(\mathsf{PP}, \mathsf{MK}, \mathsf{ID}, \mathsf{st}), \mathsf{KeyUp}(\mathsf{PP}, \mathsf{MK}, \mathsf{T}, \mathsf{RL}, \mathsf{st})), \mathsf{Enc}(\mathsf{PP}, \mathsf{ID}, \mathsf{T}, M))$.

Throughout this paper, we consider the following security notion called *indistinguishability from random against selective chosen plaintext attacks and Q-bounded decryption key exposure* (IND-sRID-Q-CPA). That is, we define indistinguishability from random against CPA adversaries taking into account *Q-bounded DKER*, which is a weaker notion than original (unbounded) DKER [34]. Q-bounded DKER guarantees that the RIBE scheme is secure even if at most Q decryption keys per user leaked, whereas unbounded DKER allows any number of decryption-key leakage. In our security model, a CPA adversary is allowed to obtain at most Q decryption keys of the target user ID^*, and tries to distinguish between the challenge ciphertext and a random element in the ciphertext space. Therefore, our security model also implies anonymity.

Definition 3 (IND-sRID-Q-CPA). *For any a-priori fixed Q ($:= poly(\lambda)$), an RIBE scheme Π is said to satisfy IND-sRID-Q-CPA security if for all*

PPT adversaries $\mathcal{A}$, $Adv^{IND\text{-}Q\text{-}CPA}_{\Pi,\mathcal{A}}(\lambda, N)$ is negligible in λ. For a PPT adversary $\mathcal{A}$, we define $\mathcal{A}$'s advantage against IND-sRID-Q-CPA security by $Adv^{IND\text{-}Q\text{-}CPA}_{\Pi,\mathcal{A}}(\lambda) := |\Pr[Exp^{IND\text{-}Q\text{-}CPA}_{\Pi,\mathcal{A}}(\lambda) = 1] - 1/2|$, where $Exp^{IND\text{-}Q\text{-}CPA}_{\Pi,\mathcal{A}}(\lambda)$ is defined by the following experiment:

$$
\begin{aligned}
Exp^{IND\text{-}Q\text{-}CPA}_{\Pi,\mathcal{A}}(\lambda) : & (\mathsf{ID}^*, \mathsf{T}^*, state_1) \leftarrow \mathcal{A}(\mathsf{find}, \lambda) \\
& (\mathsf{PP}, \mathsf{MK}, \mathsf{RL}, \mathsf{st}) \leftarrow \mathsf{SetUp}(\lambda, N) \\
& (M^*, state_2) \leftarrow \mathcal{A}^{\mathcal{O}}(\mathsf{find}, \mathsf{PP}, state_1) \\
& \mathsf{CT}_0 \leftarrow \mathsf{Enc}(\mathsf{PP}, \mathsf{ID}^*, \mathsf{T}^*, M^*), \quad \mathsf{CT}_1 \xleftarrow{\$} \mathcal{C}_\lambda, \quad b \xleftarrow{\$} \{0,1\} \\
& b' \leftarrow \mathcal{A}^{\mathcal{O}}(\mathsf{guess}, \mathsf{CT}_b, state_2) \\
& \textit{Return } 1 \textit{ if } b' = b; \textit{ otherwise, return } 0
\end{aligned}
$$

where $\mathcal{C}_\lambda$ is a ciphertext space which is determined by the security parameter λ. Here, $\mathcal{O}$ is a set of oracles $\{PKG(\cdot), KeyUp(\cdot), Revoke(\cdot,\cdot), DKG(\cdot,\cdot)\}$ defined as follows.

***PKG*$(\cdot)$:** *For a query $\mathsf{ID} \in \mathcal{I}$, it stores and returns $\mathsf{PKG}(\mathsf{PP}, \mathsf{MK}, \mathsf{ID}, \mathsf{st})$.*
***KeyUp*$(\cdot)$:** *For a query $T \in \mathcal{T}$, it stores and returns $\mathsf{KeyUp}(\mathsf{PP}, \mathsf{MK}, \mathsf{T}, \mathsf{RL}, \mathsf{st})$.*
***Revoke*$(\cdot,\cdot)$:** *For a query $(\mathsf{ID}, \mathsf{T}) \in \mathcal{I} \times \mathcal{T}$, it updates a revocation list RL by running $\mathsf{Revoke}(\mathsf{PP}, \mathsf{ID}, \mathsf{T}, \mathsf{RL}, \mathsf{st})$.*
***DKG*$(\cdot,\cdot)$:** *For a query $(\mathsf{ID}, \mathsf{T}) \in \mathcal{I} \times \mathcal{T}$, it returns $\mathsf{DKG}(\mathsf{PP}, \mathsf{SK}_{\mathsf{ID}}, \mathsf{KU}_{\mathsf{T}})$ and stores it unless it is $\perp$.*

$\mathcal{A}$ is allowed to access the above oracles with the following restrictions.

1. *KeyUp$(\cdot)$ and Revoke$(\cdot,\cdot)$ can be queried at a time period which is later than or equal to that of all previous queries.*
2. *Revoke$(\cdot,\cdot)$ cannot be queried at a time period T after issuing T to KeyUp$(\cdot)$.*
3. *If ID^* was issued to PKG$(\cdot)$ at T', then $(\mathsf{ID}^*, \mathsf{T})$ must be issued to Revoke$(\cdot,\cdot)$ such that $\mathsf{T}' \leq \mathsf{T} \leq \mathsf{T}^*$.*
4. *DKG$(\cdot,\cdot)$ cannot be queried at T before issuing T to KeyUp$(\cdot)$.*
5. *$(\mathsf{ID}^*, \mathsf{T}^*)$ cannot be issued to DKG$(\cdot,\cdot)$.*
6. *If $(\mathsf{ID}^*, \mathsf{T})$'s such that $\mathsf{T} \neq \mathsf{T}^*$ were issued to DKG$(\cdot,\cdot)$ more than Q times, then $(\mathsf{ID}^*, \mathsf{T})$ must be issued to Revoke$(\cdot,\cdot)$ such that $\mathsf{T} \leq \mathsf{T}^*$.*

4 Construction

In this section, we show the construction of our lattice-based B-DKER RIBE scheme that utilizes CFFs.

– $\mathsf{SetUp}(\lambda, N)$: On input a security parameter λ and a maximal number N of users, set the parameters q, n, m, σ, α. Then, use the $\mathsf{TrapGen}(q, n, m)$ algorithm to select $\mathbf{A}_0 \in \mathbb{Z}_q^{n \times m}$ with a basis $\mathbf{T}_{\mathbf{A}_0}$ for $\Lambda_q^{\perp}(\mathbf{A}_0)$. Select $\mathbf{A}_1, \mathbf{A}_2 \xleftarrow{\$} \mathbb{Z}_q^{n \times m}$ and $\mathbf{u} \leftarrow \mathbb{Z}_q^n$. Choose an FRD map H as in Definition 2. Run $(w, d, \mathcal{F}) \xleftarrow{\$} \mathsf{CFF.Gen}(|\mathcal{T}|, Q)$ and finally output

$$\mathsf{PP} := (H, \mathbf{A}_0, \mathbf{A}_1, \mathbf{A}_2, \mathbf{u}), \qquad \mathsf{MK} := \mathbf{T}_{\mathbf{A}_0},$$

$\mathsf{st} := \mathtt{BT}$, and $\mathsf{RL} := \emptyset$.

- $\mathsf{PKG}(\mathsf{PP}, \mathsf{MK}, \mathsf{ID}, \mathsf{st})$: Parse st as $\mathtt{BT}$. Randomly choose an unassigned leaf η from $\mathtt{BT}$, and store $\mathsf{ID} \in \mathbb{Z}_q^n$ in the leaf η. For each node $\theta \in \mathsf{Path}(\mathtt{BT}, \eta)$, perform the following steps: Recall $\{\mathbf{u}'_{\theta,\ell}\}_{\ell\in[d]}$ if it was defined. Otherwise, choose $\mathbf{u}'_{\theta,1}, \ldots, \mathbf{u}'_{\theta,d} \xleftarrow{\$} \mathbb{Z}_q^n$ and store them in θ. For every $\ell \in [d]$, sample $\mathbf{e}'_{\theta,\ell} \leftarrow \mathsf{SampleLeft}(\mathbf{A}_0, \mathbf{F}_{\mathsf{ID}}, \mathbf{u}'_{\theta,\ell}, \mathbf{T}_{\mathbf{A}_0}, \sigma)$ where $\mathbf{F}_{\mathsf{ID}} = \mathbf{A}_1 + H(\mathsf{ID})\mathbf{G} \in \mathbb{Z}_q^{n\times m}$. Finally output
$$\mathsf{SK}_{\mathsf{ID}} = \left(\left\{\theta, \{\mathbf{e}'_{\theta,\ell}\}_{\ell\in[d]}\right\}_{\theta\in\mathsf{Path}(\mathtt{BT},\eta)}\right).$$
- $\mathsf{KeyUp}(\mathsf{PP}, \mathsf{MK}, \mathsf{T}, \mathsf{RL}, \mathsf{st})$: For each node $\theta \in \mathsf{KUNode}(\mathtt{BT}, \mathsf{RL}, \mathsf{T})$, perform the following steps: Recall $\{\mathbf{u}'_{\theta,\ell}\}_{\ell\in[d]}$ if it was defined. Otherwise, choose $\mathbf{u}'_{\theta,1}, \ldots, \mathbf{u}'_{\theta,d} \xleftarrow{\$} \mathbb{Z}_q^n$ and store them in θ. Sample $\tilde{\mathbf{e}}_\theta \leftarrow \mathsf{SampleLeft}(\mathbf{A}_0, \mathbf{F}_{\mathsf{T}}, \tilde{\mathbf{u}}_\theta, \mathbf{T}_{\mathbf{A}_0}, \sigma)$ where $\mathbf{F}_{\mathsf{T}} = \mathbf{A}_2 + H(\mathsf{T})\mathbf{G}$ and $\tilde{\mathbf{u}}_\theta = \mathbf{u} - \sum_{\ell\in\mathcal{F}_{\mathsf{T}}} \mathbf{u}'_{\theta,\ell}$. Output
$$\mathsf{KU}_{\mathsf{T}} = \left(\{\theta, \tilde{\mathbf{e}}_\theta\}_{\theta\in\mathsf{KUNode}(\mathtt{BT},\mathsf{RL},\mathsf{T})}, \mathcal{F}_{\mathsf{T}}\right),$$
where for simplicity we here assume $\mathcal{F}_{\mathsf{T}}$ is a d-bit string such that ℓ-th bit is one for $\ell \in \mathcal{F}_{\mathsf{T}}$ and other bits are zero.
- $\mathsf{DKG}(\mathsf{PP}, \mathsf{SK}_{\mathsf{ID}}, \mathsf{KU}_{\mathsf{T}})$: Parse $\mathsf{SK}_{\mathsf{ID}}$ and KU_{T} as $\left\{\theta, \left\{\mathbf{e}'_{\theta,\ell}\right\}_{\ell\in[d]}\right\}_{\theta\in\Theta_{\mathrm{SK}}}$ and $\{\theta, \tilde{\mathbf{e}}_\theta\}_{\theta\in\Theta_{\mathrm{KU}}}$, respectively. Output $\perp$ if $\Theta_{\mathrm{SK}} \cap \Theta_{\mathrm{KU}} = \emptyset$. Otherwise, for some $\theta \in \Theta_{\mathrm{SK}} \cap \Theta_{\mathrm{KU}}$, compute $\mathbf{e}_\theta = \sum_{\ell\in\mathcal{F}_{\mathsf{T}}} \mathbf{e}'_{\theta,\ell}$ and output $\mathsf{DK}_{\mathsf{ID},\mathsf{T}} = (\mathbf{e}_\theta, \tilde{\mathbf{e}}_\theta)$.
- $\mathsf{Enc}(\mathsf{PP}, \mathsf{ID}, \mathsf{T}, M)$: To encrypt a bit $M \in \{0,1\}$, it runs the following steps: Set $\mathbf{F}_{\mathsf{ID},\mathsf{T}} = [\mathbf{A}_0|\mathbf{F}_{\mathsf{ID}}|\mathbf{F}_{\mathsf{T}}] \in \mathbb{Z}_q^{n\times 3m}$. Choose $\mathbf{s} \xleftarrow{\$} \mathbb{Z}_q^n$ and $\mathbf{R}_{\mathsf{ID}}, \mathbf{R}_{\mathsf{T}} \xleftarrow{\$} \{-1,1\}^{m\times m}$. Choose noise $x \leftarrow \bar{\Psi}_\alpha$ and a noise vector $\mathbf{y} \leftarrow \bar{\Psi}_\alpha^m$ and set $\mathbf{z}_{\mathsf{ID}} = \mathbf{R}_{\mathsf{ID}}^T\mathbf{y} \in \mathbb{Z}_q^m$, $\mathbf{z}_{\mathsf{T}} = \mathbf{R}_{\mathsf{T}}^T\mathbf{y} \in \mathbb{Z}_q^m$. Set
$$c_0 = \mathbf{u}^T\mathbf{s} + x + M\left\lfloor\frac{q}{2}\right\rfloor \in \mathbb{Z}_q, \qquad \mathbf{c} = \mathbf{F}_{\mathsf{ID},\mathsf{T}}^T\mathbf{s} + \begin{bmatrix}\mathbf{y}\\ \mathbf{z}_{\mathsf{ID}}\\ \mathbf{z}_{\mathsf{T}}\end{bmatrix} \in \mathbb{Z}_q^{3m}.$$
Output the ciphertext $\mathsf{CT}_{\mathsf{ID},\mathsf{T}} := (c_0, \mathbf{c}) \in \mathbb{Z}_q \times \mathbb{Z}_q^{3m}$.
- $\mathsf{Dec}(\mathsf{PP}, \mathsf{DK}_{\mathsf{ID},\mathsf{T}}, \mathsf{CT}_{\mathsf{ID},\mathsf{T}})$: It runs the following steps: Parse $\mathbf{c}$ as $\begin{bmatrix}\mathbf{c}_0\\ \mathbf{c}_1\\ \mathbf{c}_2\end{bmatrix}$ where $\mathbf{c}_i \in \mathbb{Z}_q^m$. Compute $c' = c_0 - \mathbf{e}_\theta^T\begin{bmatrix}\mathbf{c}_0\\ \mathbf{c}_1\end{bmatrix} - \tilde{\mathbf{e}}_\theta^T\begin{bmatrix}\mathbf{c}_0\\ \mathbf{c}_2\end{bmatrix} \in \mathbb{Z}_q$. Compare c' and $\lfloor\frac{q}{2}\rfloor$ treating them as integers in $\mathbb{Z}$. If they are close, i.e., if $|c' - \lfloor\frac{q}{2}\rfloor| < \lfloor\frac{q}{4}\rfloor$, output 1, otherwise output 0.
- $\mathsf{Revoke}(\mathsf{ID}, \mathsf{T}, \mathsf{RL}, \mathsf{st})$: Add $(\mathsf{ID}, \mathsf{T})$ to RL, and output the updated RL.

Parameters and Correctness. We use the following lemma to bound the noise.

Lemma 7 ([1]). *Let* $\mathbf{e}$ *be some vector in* $\mathbb{Z}^m$ *and let* $\mathbf{y} \leftarrow \bar{\Psi}_\alpha$. *Then the quantity* $|\langle \mathbf{e}, \mathbf{y} \rangle|$ *when treated as an integer in* $(-q/2, q/2]$ *satisfies* $|\langle \mathbf{e}, \mathbf{y} \rangle| \leq \|\mathbf{e}\| q\alpha \cdot \omega(\sqrt{\log m}) + \|\mathbf{e}\|\sqrt{m}/2$.

We have during decryption,

$$w = c_0 - \mathbf{e}_\theta^T \begin{bmatrix} \mathbf{c}_0 \\ \mathbf{c}_1 \end{bmatrix} - \tilde{\mathbf{e}}_\theta^T \begin{bmatrix} \mathbf{c}_0 \\ \mathbf{c}_2 \end{bmatrix} = M \left\lfloor \frac{q}{2} \right\rfloor + \underbrace{x - \mathbf{e}_\theta^T \begin{bmatrix} \mathbf{y} \\ \mathbf{R}_{\mathsf{ID}}^T \mathbf{y} \end{bmatrix} - \tilde{\mathbf{e}}_\theta^T \begin{bmatrix} \mathbf{y} \\ \mathbf{R}_{\mathsf{T}}^T \mathbf{y} \end{bmatrix}}_{\text{error term}}.$$

Then, the error term can be bounded as follows.

Lemma 8. *The norm of the error term is bounded by* $wq\sigma m\alpha \cdot \omega(\sqrt{\log m}) + O(w\sigma m^{3/2})$ *with high probability.*

Proof. Let $\mathbf{e}_\theta = (\mathbf{e}_{\theta,1} | \mathbf{e}_{\theta,2})$ and $\tilde{\mathbf{e}}_\theta = (\tilde{\mathbf{e}}_{\theta,1} | \tilde{\mathbf{e}}_{\theta,2})$ with $\mathbf{e}_{\theta,1}, \mathbf{e}_{\theta,2}, \tilde{\mathbf{e}}_{\theta,1}, \tilde{\mathbf{e}}_{\theta,2} \in \mathbb{Z}^m$. Then the error term is

$$x - \mathbf{e}_\theta^T \begin{bmatrix} \mathbf{y} \\ \mathbf{R}_{\mathsf{ID}}^T \mathbf{y} \end{bmatrix} - \tilde{\mathbf{e}}_\theta^T \begin{bmatrix} \mathbf{y} \\ \mathbf{R}_{\mathsf{T}}^T \mathbf{y} \end{bmatrix} = x - \left(\mathbf{e}_{\theta,1} + \tilde{\mathbf{e}}_{\theta,1} + \mathbf{R}_{\mathsf{ID}} \mathbf{e}_{\theta,2} + \mathbf{R}_{\mathsf{T}} \tilde{\mathbf{e}}_{\theta,2}\right)^T \mathbf{y}.$$

From Lemma 2, we have $\|\mathbf{e}'_{\theta,\ell}\| \leq \sigma\sqrt{2m}$ and $\|\tilde{\mathbf{e}}_\theta\| \leq \sigma\sqrt{2m}$ with high probability. The former bounds imply that $\|\mathbf{e}_\theta\| \leq \sum_{\ell \in \mathcal{F}_{\mathsf{T}}} \|\mathbf{e}'_{\theta,\ell}\| \leq w\sigma\sqrt{2m}$. Here, we use the fact that CFF is w-uniform. By Lemma 5, $\|\mathbf{R}_{\mathsf{ID}}\| \leq O(\sqrt{m})$ and $\|\mathbf{R}_{\mathsf{T}}\| \leq O(\sqrt{m})$ with high probability. Then, $\|\mathbf{e}_{\theta,1} + \tilde{\mathbf{e}}_{\theta,1} + \mathbf{R}_{\mathsf{ID}}\mathbf{e}_{\theta,2} + \mathbf{R}_{\mathsf{T}}\tilde{\mathbf{e}}_{\theta,2}\| \leq \|\mathbf{e}_{\theta,1}\| + \|\tilde{\mathbf{e}}_{\theta,1}\| + \|\mathbf{R}_{\mathsf{ID}}\mathbf{e}_{\theta,2}\| + \|\mathbf{R}_2\tilde{\mathbf{e}}_{\theta,2}\| \leq O(w\sigma m)$. Then, by Lemma 7, the error term is bounded by

$$|x| + \left|\left(\mathbf{e}_{\theta,1} + \tilde{\mathbf{e}}_{\theta,1} + \mathbf{R}_{\mathsf{ID}}\mathbf{e}_{\theta,2} + \mathbf{R}_2\tilde{\mathbf{e}}_{\theta,2}\right)^T \mathbf{y}\right| \leq w\sigma m q\alpha \cdot \omega(\sqrt{\log m}) + O(w\sigma m^{3/2}).$$

□

Now, for the scheme to work correctly, the following conditions should hold, taking n to be the security parameter:

- the error term is less than $q/5$ with high probability, i.e., $\alpha < [w\sigma m \cdot \omega(\sqrt{\log m})]^{-1}$ and $q = \Omega(w\sigma m^{3/2})$,
- that $\mathsf{TrapGen}$ can operate, i.e., $m > 2n \log q$,
- that σ is sufficiently large for $\mathsf{SampleLeft}$ and $\mathsf{SampleRight}$, i.e., $\sigma > \|\mathbf{T}_{\mathbf{G}}\|_{\mathsf{GS}} \cdot \|\mathbf{R}_{\mathsf{ID}}\| \cdot \omega(\sqrt{\log m}) = \sqrt{m} \cdot \omega(\sqrt{\log m})$,
- that Regev's reduction applies, i.e., $q > 2\sqrt{n}/\alpha$,

Hence, we set the parameters (q, m, σ, α) as follows:

$$m = 2n^{1+\delta}, \qquad q = wm^2 \cdot \omega(\sqrt{\log n}),$$
$$\sigma = \sqrt{m} \cdot \omega(\sqrt{\log n}), \qquad \alpha = \left[wm^{3/2} \cdot \omega(\sqrt{\log n})\right]^{-1},$$

and round up m to the nearest larger integer and q to the nearest larger prime. Here we assume that δ is such that $n^\delta > \lceil \log q \rceil = O(\log n)$.

5 Security

In this section, we prove the security of our scheme in Section 4.

Theorem 1. *If the LWE assumption holds and the underlying CFF is Q-cover-free and w-uniform, then the proposed RIBE scheme in Section 4 with the parameters set as above is IND-sRID-Q-CPA secure. In particular, if there exists an adversary $\mathcal{A}$ attacking IND-sRID-Q-CPA security of the RIBE scheme, then there exists an adversary $\mathcal{B}$ against the LWE assumption with advantage $Adv_{\mathcal{B}}^{LWE} \geq \frac{1}{w} Adv_{\Pi,\mathcal{A}}^{IND\text{-}Q\text{-}CPA}(\lambda) - \mathsf{negl}(\lambda)$.*

Due to the page limitation, we omit some detailed discussion of the following proof. Especially, we focus on the part that differs from Chen et al.'s proof [16].

Proof. The proof proceeds in a sequence of games where the first game is the same as IND-sRID-Q-CPA game. In the last game, the challenge ciphertext is a uniform random element in the ciphertext space, hence, the advantage of a PPT adversary $\mathcal{A}$ is zero. Let E_i denote the event that $\mathcal{A}$ wins the game, i.e., $b' = b$, in **Game** i. Then, $\mathcal{A}$'s advantage in **Game** i is $\left|\Pr[E_i] - \frac{1}{2}\right|$.

Let ID^* denote the challenge identity. The simulator $\mathcal{B}$ guesses an adversarial type among the following two types:

- **Type-I adversary**: ID^* will be revoked before T^*. Hence, $\mathcal{A}$ may issue a secret key extraction query for $\mathsf{SK}_{\mathsf{ID}^*}$ or decryption key queries $\mathsf{DK}_{\mathsf{ID}^*,\mathsf{T}}$ for $\mathsf{T} \neq \mathsf{T}^*$ more than Q times.
- **Type-II adversary**: ID^* will not be revoked before T^*. Hence, $\mathcal{A}$ may issue decryption key queries $\mathsf{DK}_{\mathsf{ID}^*,\mathsf{T}}$ for $\mathsf{T} \neq \mathsf{T}^*$ at most Q times.

$\mathcal{B}$ guesses the types of the adversary with probability 1/2. If the guess is not correct, $\mathcal{B}$ aborts the game and output a random bit. We separate the description of **Game** 2 against the Type-I and Type-II adversary. Other games are the same for both types of the adversary.

Game^{real}: This is the original IND-sRID-Q-CPA game between an adversary $\mathcal{A}$ against our scheme and an IND-RID-Q-CPA challenger.

Game 0: The game is the same as **Game**^{real} except that at the beginning of the game, the challenger guesses an index $\ell^* \in \mathcal{F}_{\mathsf{T}^*}$ such that the secret key element $\mathbf{e}'_{\theta,\ell^*}$ is not used to answer the first Q decryption key queries $\mathsf{DK}_{\mathsf{ID}^*,\mathsf{T}}$ by $\mathcal{A}$, and assume that the guess is right. If the guess is not correct, $\mathcal{B}$ aborts the game and output a random bit.

Obviously, the challenger's guess is right with probability $1/w$. In other words, the reduction loss is w, which is polynomial in the security parameter. Note that in the rest of the proof, the challenger knows the index ℓ^*. The guess is crucial to answer ID's decryption keys in **Game** 2 against the Type II adversary.

Game 1: In **Game** 0, the PP contains random matrices $\mathbf{A}_0, \mathbf{A}_1, \mathbf{A}_2$ in $\mathbb{Z}_q^{n\times m}$. At the challenge phase, the challenger generates a ciphertext $\mathsf{CT}_{\mathsf{ID}^*,\mathsf{T}^*}$. We let $\mathbf{R}_{\mathsf{ID}^*}$ and $\mathbf{R}_{\mathsf{T}^*}$ denote random matrices generated for the creation of the challenge

ciphertext. As the proof of Agrawal et al. [1], **Game** 1 is the same as **Game** 0 except that we change the creations of $\mathbf{A}_1$ and $\mathbf{A}_2$ in the PP. The challenger chooses $\mathbf{R}_{\mathsf{ID}^*}$ and $\mathbf{R}_{\mathsf{T}^*}$, which will be used to create the challenge ciphertext $\mathsf{CT}_{\mathsf{ID}^*,\mathsf{T}^*}$, at the setup phase and construct matrices $\mathbf{A}_1$ and $\mathbf{A}_2$ as

$$\mathbf{A}_1 \leftarrow \mathbf{A}_0\mathbf{R}_{\mathsf{ID}^*} - H(\mathsf{ID}^*)\mathbf{G} \quad \text{and} \quad \mathbf{A}_2 \leftarrow \mathbf{A}_0\mathbf{R}_{\mathsf{T}^*} - H(\mathsf{T}^*)\mathbf{G}.$$

The remainder of the game is unchanged. In $\mathcal{A}$'s view, **Game** 1 and **Game** 0 are statistically indistinguishable from Lemma 6.

Game 2: In **Game** 1, $\{\mathbf{u}'_{\theta,\ell}\}_{\ell\in[d]}$ are independently random vectors in $\mathbb{Z}_q^n$, and the challenger samples $\{\mathbf{e}'_{\theta,\ell}\}_{\ell\in[d]}$ and $\tilde{\mathbf{e}}_\theta$ using $\mathsf{SampleLeft}$. **Game** 2 is the same as **Game** 1 except that, for each node θ, we change the distributions of $\{\mathbf{u}'_{\theta,\ell}\}_{\ell\in[d]}$, the secret key $\{\mathbf{e}'_{\theta,\ell}\}_{\ell\in[d]}$ for ID^*, and the key update $\tilde{\mathbf{e}}_\theta$ for T^* so that $\mathcal{B}$ can create the keys without using the trapdoor $\mathbf{T}_{\mathbf{A}_0}$. In this game, the distributions differ against the type of adversaries. We use **Game** 2-I and **Game** 2-II to denote the games.

Type-I Adversary: The modification of **Game** 2-I is similar to Chen et al.'s one [16]. By definition, the challenger should answer $\mathsf{SK}_{\mathsf{ID}^*}$ and $\mathsf{DK}_{\mathsf{ID}^*,\mathsf{T}}$ queries only for the nodes $\theta \in \mathsf{Path}(\eta^*)$, where η^* is a randomly selected leaf which ID^* will be assigned to. By definition of Type-I adversary, since ID^* will be revoked before T^*, the challenger should answer $\mathsf{KU}_{\mathsf{T}^*}$ queries only for the nodes $\theta \notin \mathsf{Path}(\eta^*)$. Hence, there are no nodes θ that the challenger should answer key queries for both ID^* and T^*. Then, in **Game** 2-I, we change the distributions as follows:

- Sample independently random $\mathbf{e}'_{\theta,\ell} \leftarrow \mathsf{SampleGaussian}(\sigma)$ and set $\mathbf{u}'_{\theta,\ell} = [\mathbf{A}_0|\mathbf{F}_{\mathsf{ID}^*}]\mathbf{e}_{\theta,\ell}$ for $\ell \in [d]$ and $\theta \in \mathsf{Path}(\eta^*)$,
- Sample $\tilde{\mathbf{e}}_\theta \leftarrow \mathsf{SampleGaussian}(\sigma)$ and set $\tilde{\mathbf{u}}_\theta = [\mathbf{A}_0|\mathbf{F}_{\mathsf{T}^*}]\tilde{\mathbf{e}}_\theta$ for $\theta \notin \mathsf{Path}(\eta^*)$. Set $\mathbf{u}'_{\theta,\ell}$ for $\ell \in [d] \setminus \{\ell^*\}$ as independently random vectors in $\mathbb{Z}_q^n$. Then, set $\mathbf{u}'_{\theta,\ell^*} = \mathbf{u} - \tilde{\mathbf{u}}_\theta - \sum_{\ell\in\mathcal{F}_{\mathsf{T}^*}\setminus\{\ell^*\}} \mathbf{u}'_{\theta,\ell}$.

Although we use ℓ^*, which the challenger guessed in **Game** 0, to create $\{\mathbf{u}'_{\theta,\ell}\}_{\ell\in[d]}$ for $\theta \notin \mathsf{Path}(\eta^*)$, the role can be replaced by any $\ell \in \mathcal{F}_{\mathsf{T}^*}$. Then, the challenger responds to $\mathcal{A}$'s key queries as follows:

- $\mathsf{SK}_{\mathsf{ID}}$ queries for $\mathsf{ID} \neq \mathsf{ID}^*$ and KU_{T} queries for $\mathsf{T} \neq \mathsf{T}^*$ are unchanged,
- answers $\mathsf{SK}_{\mathsf{ID}^*}$ queries using the above $\left\{\mathbf{e}'_{\theta,\ell}\right\}_{\ell\in[d]}$,
- answers $\mathsf{KU}_{\mathsf{T}^*}$ queries using the above $\tilde{\mathbf{e}}_\theta$,
- answers $\mathsf{DK}_{\mathsf{ID},\mathsf{T}}$ queries by using the above $\mathsf{SK}_{\mathsf{ID}}$ and KU_{T}.

Notice that we do not use the trapdoor $\mathbf{T}_{\mathbf{A}_0}$ to create $\mathsf{SK}_{\mathsf{ID}^*}$ and $\mathsf{KU}_{\mathsf{T}^*}$.

As Chen et al. [16], we can show that **Game** 2-I is statistically indistinguishable from **Game** 1 with high probability. In **Game** 1, $\{\mathbf{u}'_{\theta,\ell}\}_{\ell\in[d]}$ are independently random vectors in $\mathbb{Z}_q^n$, and since $\{\mathbf{e}'_{\theta,\ell}\}_{\ell\in[d]}$ for ID^* and $\tilde{\mathbf{e}}_\theta$ for T^* are sampled from $\mathbf{e}_{\theta,\ell} \leftarrow \mathsf{SampleLeft}(\mathbf{A}_0, \mathbf{F}_{\mathsf{ID}^*}, \mathbf{u}'_{\theta,\ell}, \mathbf{T}_{\mathbf{A}_0}, \sigma)$ and $\tilde{\mathbf{e}}_\theta \leftarrow \mathsf{SampleLeft}(\mathbf{A}_0, \mathbf{F}_{\mathsf{T}^*}, \tilde{\mathbf{u}}_\theta, \mathbf{T}_{\mathbf{A}_0}, \sigma)$ where $\tilde{\mathbf{u}}_\theta = \mathbf{u} - \sum_{\ell\in\mathcal{F}_{\mathsf{T}^*}} \mathbf{u}'_{\theta,\ell}$, the distributions

are statistically close to $\mathcal{D}_{\Lambda_q^{\mathbf{u}'_{\theta,\ell}}([\mathbf{A}_0|\mathbf{F}_{\mathsf{ID}^*}]),\sigma}$ and $\mathcal{D}_{\Lambda_q^{\tilde{\mathbf{u}}_\theta}([\mathbf{A}_0|\mathbf{F}_{\mathsf{T}^*}]),\sigma}$, respectively. In **Game** 2-I, $\{\mathbf{e}_{\theta,\ell}\}_{\ell\in[d]}$ for ID^* and $\tilde{\mathbf{e}}_\theta$ for T^* are sampled from $\mathcal{D}_{\mathbb{Z}^{2m},\sigma}$ from the property of $\mathsf{SampleGaussian}$. Hence, by Lemma 3, the distribution of each $\{\mathbf{u}'_{\theta,\ell}\}_{\ell\in[d]}$ and $\tilde{\mathbf{u}}_\theta$ in **Game** 2-I is statistically close to uniform over $\mathbb{Z}_q^n$, respectively. Furthermore, the conditional distribution of each $\{\mathbf{e}'_{\theta,\ell}\}_{\ell\in[d]}$ and $\tilde{\mathbf{e}}_\theta$ given $\{\mathbf{u}'_{\theta,\ell}\}_{\ell\in[d]}$ and $\tilde{\mathbf{u}}_\theta$ is statistically close to $\mathcal{D}_{\Lambda_q^{\mathbf{u}'_{\theta,\ell}}([\mathbf{A}_0|\mathbf{F}_{\mathsf{ID}^*}]),\sigma}$ and $\mathcal{D}_{\Lambda_q^{\tilde{\mathbf{u}}_\theta}([\mathbf{A}_0|\mathbf{F}_{\mathsf{T}^*}]),\sigma}$, respectively. Hence, **Game** 2-I is statistically indistinguishable from **Game** 1 in $\mathcal{A}$'s view.

Type-II adversary. The modification of **Game** 2-II is the most technical part of this paper. In this game, the distributions of $\mathbf{u}'$, $\{\mathbf{e}'_{\theta,\ell}\}_{\ell\in[d]}$, and $\tilde{\mathbf{e}}_\theta$ for $\theta \notin \mathsf{Path}(\eta^*)$ are the same as **Game** 2-I, however, we change the distributions of those for $\theta \in \mathsf{Path}(\eta^*)$. As opposed to the case of **Game** 2-I, the challenge ID^* will not be revoked in the challenge time period T^*. Since there are nodes θ which the simulator should create both the secret key $\{\mathbf{e}_{\theta,\ell}\}_{\ell\in[d]}$ for ID^* and the key update $\tilde{\mathbf{e}}_\theta$ for T^*, the previous approach is insufficient. In **Game** 2-II, we change the distributions for $\theta \in \mathsf{Path}(\eta^*)$ as follows:

- Sample independently random $\mathbf{e}'_{\theta,\ell} \leftarrow \mathsf{SampleGaussian}(\sigma)$ and set $\mathbf{u}'_{\theta,\ell} = [\mathbf{A}_0|\mathbf{F}_{\mathsf{ID}^*}]\mathbf{e}'_{\theta,\ell}$ for $\ell \in [d] \setminus \{\ell^*\}$,
- Sample $\tilde{\mathbf{e}}_\theta \leftarrow \mathsf{SampleGaussian}(\sigma)$ and set $\tilde{\mathbf{u}}_\theta = [\mathbf{A}_0|\mathbf{F}_{\mathsf{T}^*}]\tilde{\mathbf{e}}_\theta$. It immediately means that $\mathbf{u}'_{\theta,\ell^*} = \mathbf{u} - \tilde{\mathbf{u}}_\theta - \sum_{\ell\in\mathcal{F}_{\mathsf{T}^*}\setminus\{\ell^*\}} \mathbf{u}'_{\theta,\ell}$.

Then, the challenger responds to $\mathcal{A}$'s key queries as follows:

- $\mathsf{SK}_{\mathsf{ID}}$ queries for $\mathsf{ID} \neq \mathsf{ID}^*$ and KU_{T} queries for $\mathsf{T} \neq \mathsf{T}^*$ are unchanged,
- answers $\mathsf{KU}_{\mathsf{T}^*}$ queries using the above $\tilde{\mathbf{e}}_\theta$,
- answers $\mathsf{DK}_{\mathsf{ID},\mathsf{T}}$ queries for $\mathsf{ID} \neq \mathsf{ID}^*$ by using the above $\mathsf{SK}_{\mathsf{ID}}$ and KU_{T},
- answers $\mathsf{DK}_{\mathsf{ID}^*,\mathsf{T}}$ queries using the above $\{\mathbf{e}'_{\theta,\ell}\}_{\ell\in[d]}$ and KU_{T}.

The challenger can respond to all key queries by $\mathcal{A}$ using the key creation algorithms. Although the challenger can create all the other keys, it cannot create the secret key element $\mathbf{e}'_{\theta,\ell^*}$ for ID^*. However, it does not matter since the maximum number of $\mathsf{DK}_{\mathsf{ID}^*,\mathsf{T}}$ queries by $\mathcal{A}$ is bounded up to Q times by the definition of Type II adversary. Moreover, thanks to the property of CFFs and the guess ℓ^* in **Game** 0, we know that $\mathbf{e}_{\theta,\ell^*}$ is not used to respond to $\mathsf{DK}_{\mathsf{ID}^*,\mathsf{T}}$ queries. As in **Game** 2-I, **Game** 2-II is statistically indistinguishable from **Game** 1 in $\mathcal{A}$'s view by Lemma 3.

Game 3: In **Game** 2, a matrix $\mathbf{A}_0$ is generated by $\mathsf{TrapGen}$ and its trapdoor $\mathbf{T}_{\mathbf{A}_0}$ is used to respond to $\mathcal{A}$'s key queries for $\mathsf{ID} \neq \mathsf{ID}^*$ and $\mathsf{T} \neq \mathsf{T}^*$. **Game** 3 is the same as **Game** 2 except that we sample $\mathbf{A}_0$ as a random matrix in $\mathbb{Z}_q^{n\times m}$. From the property of $\mathsf{TrapGen}$, matrices generated by the algorithm are statistically close to random matrices in $\mathbb{Z}_q^{n\times m}$. Hence, the distributions of PP between **Game** 2 and **Game** 3 are statistically indistinguishable. Observe that

$$[\mathbf{A}_0|\mathbf{F}_{\mathsf{ID}}] := [\mathbf{A}_0|\mathbf{A}_1 + H(\mathsf{ID})\mathbf{G}] = [\mathbf{A}_0|\mathbf{A}_0\mathbf{R}_{\mathsf{ID}^*} + (H(\mathsf{ID}) - H(\mathsf{ID}^*))\,\mathbf{G}]\,,$$

$$[\mathbf{A}_0|\mathbf{F}_{\mathsf{T}}] := [\mathbf{A}_0|\mathbf{A}_2 + H(\mathsf{T})\mathbf{G}] = [\mathbf{A}_0|\mathbf{A}_0\mathbf{R}_{\mathsf{T}^*} + (H(\mathsf{T}) - H(\mathsf{T}^*))\,\mathbf{G}]\,.$$

Due to the property of gadget matrix, we know a trapdoor $\mathbf{T}_{\mathbf{G}}$ which is also a trapdoor for $(H(\mathsf{ID})-H(\mathsf{ID}^*))\mathbf{G}$ and $(H(\mathsf{T})-H(\mathsf{T}^*))\mathbf{G}$ if $\mathsf{ID} \neq \mathsf{ID}^*$ and $\mathsf{T} \neq \mathsf{T}^*$, since $H(\mathsf{ID})-H(\mathsf{ID}^*)$ and $H(\mathsf{T})-H(\mathsf{T}^*)$ in $\mathbb{Z}_q^{n\times n}$ are full rank. Since the trapdoor is public, one may think that it can be used by anyone, however, the knowledge of secret $\mathbf{R}_{\mathsf{ID}^*}$ and $\mathbf{R}_{\mathsf{T}^*}$ are required to use $\mathsf{SampleRight}$.

Then, the challenger responds to $\mathcal{A}$'s key queries as follows:

- $\mathsf{SK}_{\mathsf{ID}^*}$ queries and $\mathsf{KU}_{\mathsf{T}^*}$ queries are unchanged,
- answers $\mathsf{SK}_{\mathsf{ID}}$ queries for $\mathsf{ID} \neq \mathsf{ID}^*$ by $\mathbf{e}'_{\theta,\ell}$ where $\mathbf{e}'_{\theta,\ell} \leftarrow \mathsf{SampleRight}(\mathbf{A}_0, \mathbf{G}, \mathbf{R}_{\mathsf{ID}^*}, \mathbf{u}'_{\theta,\ell}, \mathbf{T}_{\mathbf{G}}, \sigma)$,
- answers KU_{T} queries for $\mathsf{T} \neq \mathsf{T}^*$ by $\tilde{\mathbf{e}}_\theta$ where $\tilde{\mathbf{e}}_\theta \leftarrow \mathsf{SampleRight}(\mathbf{A}_0, \mathbf{G}, \mathbf{R}_{\mathsf{T}^*}, \tilde{\mathbf{u}}_\theta, \mathbf{T}_{\mathbf{G}}, \sigma)$,
- answers $\mathsf{DK}_{\mathsf{ID},\mathsf{T}}$ queries by using the above $\mathsf{SK}_{\mathsf{ID}}$ and KU_{T}.

Due to the property of $\mathsf{SampleRight}$, the distributions of $\mathbf{e}'_{\theta,\ell}$ and $\tilde{\mathbf{e}}_\theta$, which are the differences from **Game** 2, are statistically close to $\mathcal{D}_{\Lambda_q^{\mathbf{u}'_{\theta,\ell}}([\mathbf{A}_0|\mathbf{F}_{\mathsf{ID}^*}]),\sigma}$ and $\mathcal{D}_{\Lambda_q^{\tilde{\mathbf{u}}_\theta}([\mathbf{A}_0|\mathbf{F}_{\mathsf{T}^*}]),\sigma}$. As a result, **Game** 3 is statistically indistinguishable from **Game** 2 in $\mathcal{A}$'s view.

$\mathbf{Game}^{final}$: $\mathbf{Game}^{final}$ is the same as **Game** 3 except that the challenge ciphertext $\mathsf{CT}_{\mathsf{ID}^*,\mathsf{T}^*}$ is always chosen as a random independent element in the ciphertext space $\mathbb{Z}_q \times \mathbb{Z}_q^{3m}$. Since the challenge ciphertext is always a fresh random element in the ciphertext space, $\mathcal{A}$'s advantage in this game is zero.

If there exists a PPT adversary $\mathcal{A}$ to distinguish between $\mathbf{Game}^{final}$ and **Game** 3, then there exists another adversary $\mathcal{B}$ to solve LWE problem. Therefore, $\left|\Pr[E_3] - \frac{1}{2}\right| = |\Pr[E_3] - \Pr[E_{final}]| \leq Adv_{\mathcal{B}}^{LWE}$. Since the proof is the standard technique of lattice-based cryptography, we omit it.

Thus, we complete the proof. □

6 Discussion

To conclude this paper, we give some further comments and open questions of this research.

Key Re-randomization. As mentioned in the introduction, the key re-randomization property is crucial for constructing all the previous (pairing-based) DKER RIBE schemes. One may think that lattice-based RIBE schemes can be easily modified to support the key re-randomization property with $\mathbf{T}_{[\mathbf{A}_0|\mathbf{F}_{\mathsf{ID}}]}$, which is a short basis of $\Lambda_q^{\perp}([\mathbf{A}_0|\mathbf{F}_{\mathsf{ID}}])$, as secret keys or $\mathbf{T}_{[\mathbf{A}_0|\mathbf{F}_{\mathsf{T}}]}$, which is a short basis of $\Lambda_q^{\perp}([\mathbf{A}_0|\mathbf{F}_{\mathsf{T}}])$, as key updates. These bases are used to support delegation in the context of hierarchical IBE [2, 14]. Indeed, the bases enable any users of RIBE scheme to re-randomize their decryption keys and the scheme to be decryption key exposure resistant. However, the approach is not applicable to the RIBE setting. If a user ID has his own secret key $\mathbf{T}_{[\mathbf{A}_0|\mathbf{F}_{\mathsf{ID}}]}$, he can produce the well-formed decryption key $\mathbf{e}$ such that $[\mathbf{A}_0|\mathbf{F}_{\mathsf{ID}}|\mathbf{F}_{\mathsf{T}}]\mathbf{e} = \mathbf{u}$ for

any time periods T without key updates. Hence, KGC cannot revoke any users. For the same reason, constructing lattice-based revocable hierarchical IBE is a major open problem that seems very hard to be solved.

Insecurity of Cheng-Zhang's RIBE Scheme [17]. Cheng and Zhang claimed that their proposed RIBE scheme with the subset difference (SD) method is the first adaptively secure one with smaller key updates. However, there are critical bugs in their security proof, i.e., Game 3 in the proof of their Theorem 1. Here, we follow the notation from [17], e.g., id and t. In their Game 3, the simulator aborts the game if $h_{\mathsf{id}^*} = 0$, where $h_{()}$ is a certain function, to answer secret key extraction queries. In addition, the simulator also aborts the game if $h_{\mathsf{id}^*} \neq 0$ to create a challenge ciphertext. Hence, the game never ends. Note that the same holds for the target time period t^*.

One may think that Chen et al.'s Gaussian sampling technique [16], which we also used, can be used to fix the bugs. However, it is not the case. Furthermore, Cheng-Zhang's RIBE scheme is not secure even in the selective security model. The difficulty comes from the SD method which they used to revoke users. The SD method is another subset cover framework and it enables us to reduce the size of key updates. Notice that the subset cover framework which Chen et al. [16] and we used in this paper is the CS method. If we modify Cheng-Zhang's RIBE scheme in the selective security model, the secret key $\mathbf{e}'$ and the key update $\tilde{\mathbf{e}}$ satisfy the following equations:

$$[\mathbf{A}_0|\mathbf{A}_1 + H(\mathsf{id})\mathbf{G}]\,\mathbf{e}' = \mathbf{u}' \qquad \text{and} \qquad [\mathbf{A}_0|\mathbf{A}_2 + H(\mathsf{t})\mathbf{G}]\,\tilde{\mathbf{e}} = \tilde{\mathbf{u}}.$$

The main difference between the SD method and the CS method is the restriction of syndrome vectors $\mathbf{u}'$ and $\tilde{\mathbf{u}}$. In the security proof, the simulator should create both the secret key $\mathbf{e}'$ for the target id^* and the key update $\tilde{\mathbf{e}}$ for the target t^*. As opposed to the CS method case, if we use the SD method, the simulator should create both $\mathbf{e}'$ and $\tilde{\mathbf{e}}$ for the same syndrome vector $\mathbf{u}' = \tilde{\mathbf{u}}$ even without DKER. Since we cannot create the keys by using the trapdoor $\mathbf{T}_{\mathbf{G}}$, we try to create them by using a Gaussian sampling algorithm. Once the simulator uses a Gaussian sampling algorithm to sample $\mathbf{e}'$ for the target id^*, then the corresponding syndrome vector $\mathbf{u}' = \tilde{\mathbf{u}}$ is fixed. Therefore, the simulator cannot create $\tilde{\mathbf{e}}$ for the target t^* by using a Gaussian sampling algorithm. Therefore, a construction of lattice-based RIBE with the SD method even in the selective security model and even without DKER is an interesting open problem.

Gadget Matrix. If we do not use CFFs in our scheme, i.e., $w = d = 1$, then the scheme is an RIBE scheme without DKER. However, our parameters are better than Chen et al.'s [16]. Notice that q and σ in our scheme are smaller than those in [16]. The improvement stems from the gadget matrix $\mathbf{G}$ due to Micciancio and Peikert [29], hence it is not the technical contribution of this paper.

Semi-adaptive Security. If we replace the hash function $\mathbf{F}_{\mathsf{ID}} = \mathbf{A}_1 + H(\mathsf{ID})\mathbf{G}$ of Agrawal et al. [1] by that of adaptively secure schemes [6, 10, 11, 14, 20, 24, 38–40], our scheme achieves semi-adaptive security[6], where an adversary issues

[6] Notice that we do not have to replace $\mathbf{F}_{\mathsf{T}} = \mathbf{A}_2 + H(\mathsf{T})\mathbf{G}$ by adaptively secure ones. Since the maximum time period is polynomially bounded, $|\mathcal{T}|$ security loss

the target $(\mathsf{ID}^*, \mathsf{T}^*)$ in advance of any key queries. What is required to prove the security of lattice-based RIBE is trapdoors that can sample short vectors $\mathbf{e}'_{\theta,\ell}$ for $\mathsf{ID} \neq \mathsf{ID}^*$ and $\tilde{\mathbf{e}}$ for $\mathsf{T} \neq \mathsf{T}^*$ according to discrete Gaussian distributions, where all the lattice-based IBE schemes have. However, it is insufficient to construct adaptively secure RIBE even without DKER. In the RIBE setting, we have to set all $\mathbf{u}'_{\theta,\ell}$ in advance of any key queries, then we use $\mathbf{F}_{\mathsf{ID}^*}$, or equivalently ID^*, for the computations. It means that the simulator has to know ID^* at that time. To avoid the obstacle, we should develop new lattice-based RIBE constructions, which are different from Chen et al.'s [16], or it may be equivalent to new lattice-based fuzzy IBE constructions, which are different from Agrawal et al.'s [3].

One may think that adaptively secure IBE is more than enough to construct semi-adaptively secure RIBE. However, we do not know how to construct semi-adaptively secure lattice-based IBE that is more efficient than adaptively secure ones. We think that the construction should be an interesting open problem in this research topic.

Anonymous (B-)DKER RIBE. Our scheme is the first anonymous (B-)DKER RIBE that is resilient to decryption key exposure. As in lattice-based IBE schemes (e.g., [1]) and Chen et al.'s RIBE scheme [16], since pairing-based anonymous IBE [12] does not support the key re-randomization property, an existing anonymous RIBE scheme [15] is insecure if an adversary is allowed to query even a single decryption key. Since the spirit of our construction is the use of distinct $\tilde{\mathbf{u}}$'s for each time period and the concrete construction with CFFs, we did not use specific techniques for lattices. Therefore, we believe that our approach enables one to construct pairing-based anonymous B-DKER RIBE.

Acknowledgement. We would like to thank Shantian Cheng and Juanyang Zhang for their sincere discussion with us. We would like to thank Shuichi Katsumata for his helpful comments. Atsushi Takayasu was (during the submission) and Yohei Watanabe is supported by a JSPS Fellowship for Young Scientists. This research was supported by JST CREST Grant Number JPMJCR14D6, Japan, JSPS KAKENHI Grant Number JP14J08237 and JP17K12697.

References

1. Agrawal, S., Boneh, D., Boyen, X.: Efficient lattice (H)IBE in the standard model. In: Gilbert, H. (ed.) EUROCRYPT 2010. LNCS 6110, pp. 553–572. Springer (2010)
2. Agrawal, S., Boneh, D., Boyen, X.: Lattice basis delegation in fixed dimension and shorter-ciphertext hierarchical IBE. In: Rabin, T. (ed.) CRYPTO 2010. LNCS 6223, pp. 98–115. Springer (2010)
3. Agrawal, S., Boyen, X., Vaikuntanathan, V., Voulgaris, P., Wee, H.: Functional encryption for threshold functions (or fuzzy IBE) from lattices. In: Fischlin, M.,

enables us to guess the target time period T^*. Indeed, Seo-Emura [34] constructed adaptively secure DKER RIBE scheme by combining the Waters IBE [37] for ID and the Boneh-Boyen IBE [8] for T.

Buchmann, J.A., Manulis, M. (eds.) PKC 2012. LNCS 7293, pp. 280–297. Springer (2012)
4. Ajtai, M.: Generating hard instances of the short basis problem. In: Wiedermann, J., van Emde Boas, P., Nielsen, M. (eds.) ICALP'99. LNCS 1644, pp. 1–9. Springer (1999)
5. Alwen, J., Peikert, C.: Generating shorter bases for hard random lattices. Theory Comput. Syst. 48(3), 535–553 (2011)
6. Apon, D., Fan, X., Liu, F.: Fully-secure lattice-based IBE as compact as PKE. IACR Cryptology ePrint Archive 2016, 125 (2016)
7. Boldyreva, A., Goyal, V., Kumar, V.: Identity-based encryption with efficient revocation. In: Ning, P., Syverson, P.F., Jha, S. (eds.) CCS 2008. pp. 417–426. ACM (2008)
8. Boneh, D., Boyen, X.: Efficient selective identity-based encryption without random oracles. J. Cryptology 24(4), 659–693 (2011)
9. Boneh, D., Franklin, M.K.: Identity-based encryption from the Weil pairing. SIAM J. Comput. 32(3), 586–615 (2003)
10. Boyen, X.: Lattice mixing and vanishing trapdoors: A framework for fully secure short signatures and more. In: Nguyen, P.Q., Pointcheval, D. (eds.) PKC 2010. LNCS 6056, pp. 499–517. Springer (2010)
11. Boyen, X., Li, Q.: Towards tightly secure lattice short signature and ID-based encryption. In: Cheon, J.H., Takagi, T. (eds.) ASIACRYPT 2016. LNCS 10032, pp. 404–434 (2016)
12. Boyen, X., Waters, B.: Anonymous hierarchical identity-based encryption (without random oracles). In: Dwork, C. (ed.) CRYPTO 2006. LNCS 4117, pp. 290–307. Springer (2006)
13. Brakerski, Z., Langlois, A., Peikert, C., Regev, O., Stehlé, D.: Classical hardness of learning with errors. In: Boneh, D., Roughgarden, T., Feigenbaum, J. (eds.) STOC'13. pp. 575–584. ACM (2013)
14. Cash, D., Hofheinz, D., Kiltz, E., Peikert, C.: Bonsai trees, or how to delegate a lattice basis. J. Cryptology 25(4), 601–639 (2012)
15. Chen, J., Lim, H.W., Ling, S., Su, L., Wang, H.: Anonymous and adaptively secure revocable IBE with constant size public parameters. CoRR abs/1210.6441 (2012)
16. Chen, J., Lim, H.W., Ling, S., Wang, H., Nguyen, K.: Revocable identity-based encryption from lattices. In: Susilo, W., Mu, Y., Seberry, J. (eds.) ACISP 2012. LNCS 7372, pp. 390–403. Springer (2012)
17. Cheng, S., Zhang, J.: Adaptive-ID secure revocable identity-based encryption from lattices via subset difference method. In: Lopez, J., Wu, Y. (eds.) ISPEC 2015. LNCS 9065, pp. 283–297. Springer (2015)
18. Emura, K., Seo, J.H., Youn, T.: Semi-generic transformation of revocable hierarchical identity-based encryption and its DBDH instantiation. IEICE Transactions 99-A(1), 83–91 (2016)
19. Erdös, P., Frankl, P., Füredi, Z.: Families of finite sets in which no set is covered by the union of r others. Israel Journal of Mathematics 51(1), pp. 79–89 (1985)
20. Gentry, C., Peikert, C., Vaikuntanathan, V.: Trapdoors for hard lattices and new cryptographic constructions. In: Dwork, C. (ed.) STOC'08. pp. 197–206. ACM (2008)
21. Goldwasser, S., Lewko, A.B., Wilson, D.A.: Bounded-collusion IBE from key homomorphism. In: Cramer, R. (ed.) TCC 2012. LNCS 7194, pp. 564–581. Springer (2012)

22. Heng, S., Kurosawa, K.: k-resilient identity-based encryption in the standard model. In: Okamoto, T. (ed.) CT-RSA 2004. LNCS 2964, pp. 67–80. Springer (2004)
23. Ishida, Y., Watanabe, Y., Shikata, J.: Constructions of cca-secure revocable identity-based encryption. In: Foo, E., Stebila, D. (eds.) ACISP 2015. LNCS 9144, pp. 174–191. Springer (2015)
24. Katsumata, S., Yamada, S.: Partitioning via non-linear polynomial functions: More compact IBEs from ideal lattices and bilinear maps. In: Cheon, J.H., Takagi, T. (eds.) ASIACRYPT 2016. LNCS 10032, pp. 682–712 (2016)
25. Kumar, R., Rajagopalan, S., Sahai, A.: Coding constructions for blacklisting problems without computational assumptions. In: Wiener, M.J. (ed.) CRYPTO '99. LNCS 1666, pp. 609–623. Springer (1999)
26. Lee, K.: Revocable hierarchical identity-based encryption with adaptive security. IACR Cryptology ePrint Archive 2016, 749 (2016)
27. Lee, K., Lee, D.H., Park, J.H.: Efficient revocable identity-based encryption via subset difference methods. IACR Cryptology ePrint Archive 2014, 132 (2014)
28. Libert, B., Vergnaud, D.: Adaptive-ID secure revocable identity-based encryption. In: Fischlin, M. (ed.) CT-RSA 2009. LNCS 5473, pp. 1–15. Springer (2009)
29. Micciancio, D., Peikert, C.: Trapdoors for lattices: Simpler, tighter, faster, smaller. In: Pointcheval, D., Johansson, T. (eds.) EUROCRYPT 2012. LNCS 7237, pp. 700–718. Springer (2012)
30. Nguyen, K., Wang, H., Zhang, J.: Server-aided revocable identity-based encryption from lattices. In: Foresti, S., Persiano, G. (eds.) CANS 2016. LNCS 10052, pp. 107–123 (2016)
31. Peikert, C.: Public-key cryptosystems from the worst-case shortest vector problem: extended abstract. In: Mitzenmacher, M. (ed.) STOC'09. pp. 333–342. ACM (2009)
32. Regev, O.: On lattices, learning with errors, random linear codes, and cryptography. In: Gabow, H.N., Fagin, R. (eds.) STOC'05. pp. 84–93. ACM (2005)
33. Seo, J.H., Emura, K.: Revocable hierarchical identity-based encryption. Theor. Comput. Sci. 542, 44–62 (2014)
34. Seo, J.H., Emura, K.: Revocable identity-based cryptosystem revisited: Security models and constructions. IEEE Trans. Information Forensics and Security 9(7), 1193–1205 (2014)
35. Seo, J.H., Emura, K.: Revocable hierarchical identity-based encryption via history-free approach. Theor. Comput. Sci. 615, 45–60 (2016)
36. Watanabe, Y., Emura, K., Seo, J.H.: New revocable IBE in prime-order groups: Adaptively secure, decryption key exposure resistant, and with short public parameters. In: Handschuh, H. (ed.) CT-RSA 2017. LNCS 10159, pp. 432–449. Springer (2017)
37. Waters, B.: Efficient identity-based encryption without random oracles. In: Cramer, R. (ed.) EUROCRYPT 2005. LNCS 3494, pp. 114–127. Springer (2005)
38. Yamada, S.: Adaptively secure identity-based encryption from lattices with asymptotically shorter public parameters. In: Fischlin, M., Coron, J. (eds.) EUROCRYPT 2016. LNCS 9666, pp. 32–62. Springer (2016)
39. Yamada, S.: Asymptotically compact adaptively secure lattice IBEs and verifiable random functions via generalized partitioning techniques. IACR Cryptology ePrint Archive 2017, 096 (2017)
40. Zhang, J., Chen, Y., Zhang, Z.: Programmable hash functions from lattices: Short signatures and IBEs with small key sizes. In: Robshaw, M., Katz, J. (eds.) CRYPTO 2016. LNCS 9816, pp. 303–332. Springer (2016)