Research Article

Fucai Luo* Saif Al-Kuwari

Revocable attribute-based proxy re-encryption

https://doi.org/10.1515/jmc-2020-0039
received September 24, 2020; accepted April 5, 2021

Abstract: Attribute-based proxy re-encryption (ABPRE), which combines the notions of proxy re-encryption (PRE) and attribute-based encryption (ABE), allows a semi-trusted proxy with re-encryption key to transform a ciphertext under a particular access policy into a ciphertext under another access policy, without revealing any information about the underlying plaintext. This primitive is very useful in applications where encrypted data need to be stored in untrusted environments, such as cloud storage. In many practical applications, and in order to address scenarios where users misbehave or the re-encryption keys are compromised, an efficient revocation mechanism is necessary for ABPRE. Previously, revocation mechanism was considered in the settings of identity-based encryption (IBE), ABE, predicate encryption (PE), and broadcast PRE, but not ABPRE, which is what we set to do in this paper. We first formalize the concept of revocable ABPRE and its security model. Then, we propose a lattice-based instantiation of revocable ABPRE. Our scheme not only supports an efficient revocation mechanism but also supports polynomial-depth policy circuits and has short private keys, where the size of the keys is dependent only on the depth of the supported policy circuits. In addition, we prove that our scheme is selectively chosen-plaintext attack (CPA) secure in the standard model, based on the learning with errors assumption.

Keywords: attribute-based encryption, proxy re-encryption, revocable mechanism, attribute-based proxy re-encryption, learning with errors

MSC 2020: 11T71

1 Introduction

With the rapid spread of modern applications, such as cloud computing, the issues of data security and privacy attract increasing attention. Hence, various cryptographic primitives have been proposed to alleviate these problems. An important example of such potential primitives is proxy re-encryption (PRE), which was shown to be useful in many applications such as distributed file system [3], data storage [12] and publish/subscribe system [36]. PRE, initially introduced by Blaze et al. [6], is an attractive cryptographic primitive that allows a semi-trusted proxy with re-encryption key to efficiently convert a ciphertext encrypted for a delegator (e.g., Alice) into another ciphertext of the same message encrypted under a delegatee's (e.g., Bob's) key, without revealing the underlying plaintext and the private keys of the delegator and the delegatee. However, in the traditional PRE systems, the communication model is one-to-one (i.e., one delegator to one delegatee), which means that a message can be re-encrypted for only a single public key. This limits their utility in many applications, where the re-encryption may be used for arbitrary recipients. One such important application is data sharing in untrusted cloud storage. In a cloud storage system, data owners are often interested in sharing their encrypted data with users satisfying a specific access policy. To enable such granular data sharing requirement, Liang et al. [26] introduced the notion of

*** Corresponding author: Fucai Luo,** College of Science and Engineering, Ringgold Standard Institution, Hamad Bin Khalifa University, Qatar, Doha, Qatar, e-mail: lfucai@hbku.edu.qa
Saif Al-Kuwari: College of Science and Engineering, Hamad Bin Khalifa University, Education City, Doha, Qatar

attribute-based proxy re-encryption (ABPRE) that combines PRE with attribute-based encryption (ABE) and presented the first ABPRE scheme based on Augment Decisional Bilinear Diffie–Hellman problem.

Attribute-based proxy re-encryption. ABPRE is an extension of PRE with additional features that allow fine-grained and role-based access to encrypted data. Similar to the ABE system, there are two types of ABPRE: Key-Policy ABPRE (KP-ABPRE) and Ciphertext-Policy ABPRE (CP-ABPRE). In an ABPRE system, a semi-trusted proxy with access to a re-encryption key (generated by delegators) can transform ciphertexts for delegators satisfying an access policy (e.g., $f(x) = 0$[1] for some policy function f and attribute x) into ciphertexts for delegatees satisfying a new access policy (e.g., some policy function g which satisfies $g(x) = 1$). Since the introduction of ABPRE by Liang et al. [26], there have been many proposals for ABPRE schemes on different settings based on different hardness assumptions [17,18,25,26,43], but all of these schemes are based on classical number-theoretic assumptions, which are not quantum-resistant. The only exception is the CP-ABPRE scheme based on learning with errors (LWE) problem proposed by Li et al. [24], which is proven CPA secure in the selective security model. LWE problem was widely used to construct various quantum-resistant schemes due to its simple algebraic structure and the classical (quantum) reduction from some lattice problems (e.g., GapSVP), which were conjectured to be resistant against quantum attacks. However, to the best of our knowledge, the problem of constructing quantum-resistant KP-ABPRE schemes remains open.

1.1 Motivations

The ABPRE system achieves both delegation of decryption and fine-grained access control. However, in many practical applications, an efficient revocation mechanism is necessary for ABPRE. In a KP-ABPRE system, ciphertexts with respect to an attribute x can be decrypted by users who have policy functions f satisfying $f(x) = 0$, but not by users who have policy functions g satisfying $g(x) = 1$. With the re-encryption keys $rk_{f\to g}$ produced by the policy functions f satisfying $f(x) = 0$ for policy functions g satisfying $g(x) = 1$, the ciphertexts can be converted into the re-encrypted ciphertexts of the same messages that are decryptable under the policy functions g satisfying $g(x) = 1$. This all-or-nothing delegation of decryption is undesirable from the perspective of data senders because the data senders may not want some users[2] whose policy functions g' do not satisfy $g'(x') = 0$ for some attributes x' to access some encrypted data. In other words, it is desirable for data senders to be able to selectively revoke some users, without being constrained by delegation of decryption. Moreover, when some re-encryption keys are compromised, it is better to invalidate the re-encryption keys in order not to affect the decryption capabilities of delegators who generate the re-encryption keys.

However, user revocation is a challenge in many one-to-many and many-to-many communication systems. In attribute-based systems, this issue is difficult since each attribute is shared by multiple users; that is, revocation of a single user may affect others who share the same attributes. Moreover, user revocation in attribute-based systems needs to be flexible and support different granularities. That is, it may be required to revoke either the entire access privilege or just partial access right of the user, i.e., a subset of her attributes. In ABPRE systems, user revocation is even more difficult since it may affect the re-encryption keys and thus the corresponding delegators.

In general, there are two types of revocation mechanisms: indirect revocation [7], which requires the authority to master revocation list and periodically issue key updates for non-revoked users, and direct revocation, which does not require key updates. The latter has been discussed for ABE [4] and predicate encryption (PE) [33]. In this paper, we focus on direct revocation on ABPRE as it is useful to have an ABPRE scheme that supports fine-grained delegation of decryption and user revocation.

---

1 Hereafter, we use $f(x) = 0$ to denote the ability of decryption.

2 Or these users may become malicious in the view of the data senders.

1.2 Our results and techniques

We first formalize the notion of revocable KP-ABPRE and its security model. Our notion supports an efficient revocation mechanism while maintaining the functionality of KP-ABPRE. The security model takes into account all adversarial capabilities of the standard CPA security of ABPRE. In addition, we assume that the adversary is able to revoke users of his choice and has access to re-encryption keys generated for the revoked users. Then, we put forward an instantiation of revocable KP-ABPRE from lattices and prove that it is selectively CPA secure in the standard model as per our security definition. Moreover, if we do not revoke any user (let the revocation list be an empty set), our scheme would yield the first lattice-based KP-ABPRE scheme, which is, unlike previous KP-ABPRE schemes [17,18,25,26,43], quantum safe.

At a high level, we obtain our revocable KP-ABPRE scheme by applying the tree-based revocation technique proposed by Naor et al. [32] to the lattice-based KP-ABE scheme of Boneh et al. [8]. In particular, we first build a complete binary tree BT with N leaves, where N is the maximum expected number of users. Each node γ in the binary tree is associated with an "identifier" and each target user is assigned an index $I \in [N]$. The secret key sk_f for a policy function f, which is only associated with the policy function f, is generated using the master secret key, and the re-encryption key $rk_{f\to(g,I)}$ for a pair (g, I) is generated using the secret key sk_f. The re-encryption key $rk_{f\to(g,I)}$ consists of some matrices corresponding to all nodes in the tree path from I to the root. Then, when generating a ciphertext c of message μ with respect to an attribute x and revocation list RL, the sender generates two layers: one is associated with the attribute x, and the other is associated with the non-revoked users (the corresponding nodes are obtained using a node selection algorithm). With the secret key sk_f, we can correctly recover the underlying plaintext μ from the ciphertext c if $f(x) = 0$. With the re-encryption key $rk_{f\to(g,I)}$, we can convert the ciphertext c into a re-encrypted ciphertext $c_{f\to(g,I)}$, and we can correctly recover the plaintext μ from the ciphertext $c_{f\to(g,I)}$ with the secret key sk_g if $I \notin \mathsf{RL}$. We emphasize that RL is kept hidden in our scheme.

In terms of efficiency, our scheme supports polynomial-depth policy function and has short secret key, where the size of the key depends only on the depth of the supported policy function. Specifically, we obtain the following results: the size of the public parameters is $O(N)$, the size of keys (including the re-encryption key) is $O(\log N)$, the ciphertext has size $O\left(r \log \frac{N}{r}\right)$, where r is the number of revoked users, and the re-encrypted ciphertext has size $O\left(r \log \frac{N}{r} \log N\right)$. Indeed, the secret key is a single $2m \times 2m$ low-norm matrix. Since $m = \Theta(n \log q)$ and $\log q$ grows linearly with the depth of the policy circuit d, the size of the secret key grows as $O(d^2)$, which is independent of the size of the supported policy function.

1.3 Related work

Chen et al. [14] proposed the first lattice-based directly revocable IBE scheme, building upon the lattice-based IBE scheme in ref. [1]. Takayasu and Watanabe [39] proposed the first lattice-based directly revocable IBE with bounded decryption key exposure resistance (DKER), which is a security notion introduced for revocable IBE to guarantee that an exposure of a user's decryption key at some period of time will not compromise the confidentiality of ciphertexts that are encrypted for different time periods. Recently, Katsumata et al. [21] proposed the first lattice-based indirectly revocable IBE with DKER without relying on the key re-randomization property, which was used in the previous constructions [23,34,42] based on number-theoretic assumptions, e.g., bilinear maps and multilinear maps.

The first lattice-based indirectly revocable CP-ABE scheme was proposed by Wang et al. [40], which combines lattice-based revocable IBE [14] with the lattice-based CP-ABE scheme [41]. Based on the lattice-based revocable IBE of Chen et al. [14], Yang et al. [44] also proposed a lattice-based indirectly revocable CP-ABE scheme. In ref. [44], the authors adopted the threshold decryption technique of ref. [5] to recover the key, which makes their scheme support flexible threshold access control. Recently, Meng [29] pointed

out some security issues of the aforementioned two schemes and proposed two lattice-based directly revocable CP-ABE schemes, which support flexible threshold access policies on multi-valued attributes.

By applying the tree-based revocation technique of Naor et al. [32] to the LWE-based PE scheme proposed by Agrawal et al. [2], Ling et al. [27] proposed the first lattice-based directly revocable PE scheme and its server-aided variant [28], where most of the computations of the users are delegated to an untrusted server. The functionality of the untrusted server in ref. [28] is to help the key generation center (KGC) achieve user revocation and reduce the users' computational burden by converting the ciphertext into a "partially decrypted ciphertext." However, the semi-trusted proxy in our revocable ABPRE system is used to achieve fine-grained delegation of decryption and help the data senders (not KGC) realize user revocation.

Recently, Ge et al. [16] applied the revocation mechanism proposed for identity-based broadcast encryption (IBBE) by Susilo et al. [38] to identity-based broadcast proxy re-encryption (IB-BPRE) and proposed the first revocable IB-BPRE scheme based on bilinear pairings. BPRE was originally proposed by Chu [15] to handle the one-to-many communication model, so that the system does not need to generate a re-encryption key for each delegatee in a specific group, but only needs to generate a broadcast re-encryption key. This notion is similar to ABPRE, which focuses on many-to-many communication model.

1.4 Organization

The rest of the paper is organized as follows: in Section 2, we give the required background on lattices, including LWE, lattice trapdoors, matrix embeddings, and the complete subtree (CS) method. The definition of revocable KP-ABPRE including its syntax and security model are provided in Section 3. In Section 4, we propose our revocable KP-ABPRE scheme in the standard model based on the LWE problem and give a full security proof. Finally, we conclude the paper with several open problems in Section 5.

2 Preliminaries

We use lower-case bold letter to denote vector $\mathbf{x}$ and upper-case bold letter to denote matrix $\mathbf{A}$. The ith component of any set r is represented by r_i. Throughout the paper, we consider truncated discrete Gaussian distribution $\mathcal{D}_{\sigma,\mathbb{Z}^m}$ and we let $[n] \triangleq \{1,\ldots,n\}$. Let $\mathbf{A}^T$ (resp. $\mathbf{x}^T$) be the transpose of $\mathbf{A}$ (resp. $\mathbf{x}$). For any vector $\mathbf{b} \in \mathbb{Z}^m$, we use $\|\mathbf{b}\|$ to denote its ℓ_2 length, i.e., $\|\mathbf{b}\| \coloneqq \sqrt{\sum_{i=1}^{m}\mathbf{b}_i^2}$. The norm of any matrix $\mathbf{A} \in \mathbb{Z}^{m\times k}$ is represented by $\|\mathbf{A}\|$ which denotes the ℓ_2 length of its longest column vector, we use $\|\mathbf{A}\|_{\mathrm{GS}}$ to denote the norm of its Gram–Schmidt (GS) orthogonalization, and we define $\|\mathbf{A}\|_2 \coloneqq \sup_{\|\mathbf{e}\|=1}\|\mathbf{Ae}\|$. Then, we have $\|\mathbf{A}\|_{\mathrm{GS}} \le \|\mathbf{A}\| \le \|\mathbf{A}\|_2 \le \sqrt{m}\cdot\|\mathbf{A}\|$ and $\|\mathbf{AB}\|_2 \le \|\mathbf{A}\|_2\cdot\|\mathbf{B}\|_2$ for any $\mathbf{B} \in \mathbb{Z}^{k\times k'}$.

2.1 Lattice background

An m-dimensional integer lattice Λ is a discrete additive subgroup of $\mathbb{R}^m$. A q-ary integer lattice and a "shifted" integer lattice are defined as follows.

Definition 2.1. For $q \ge 2$, $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$, and $\mathbf{u} \in \mathbb{Z}_q^n$, we define

$$\Lambda_q^{\perp}(\mathbf{A}) = \{\mathbf{x} \in \mathbb{Z}^m : \mathbf{Ax} = 0 \bmod q\}.$$
$$\Lambda_q^{\mathbf{u}}(\mathbf{A}) = \{\mathbf{x} \in \mathbb{Z}^m : \mathbf{Ax} = \mathbf{u} \bmod q\}.$$

Note that if $\mathbf{y} \in \Lambda_q^{\mathbf{u}}(\mathbf{A})$, then $\Lambda_q^{\mathbf{u}}(\mathbf{A}) = \Lambda_q^{\perp}(\mathbf{A}) + \mathbf{y}$.

Lemma 2.2. [8,19,31] *Given positive integers* n, $q > 2$, *and* $m > n$. *Given* $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ *and a short basis* $\mathbf{T_A}$ *for lattice* $\Lambda_q^{\perp}(\mathbf{A})$, *for any* $\sigma \geq \|\mathbf{T_A}\|_{\mathrm{GS}} \cdot \omega(\sqrt{\log m})$, $\mathbf{u} \in \mathbb{Z}_q^n$ *and* $\mathbf{D} \in \mathbb{Z}_q^{n\times m}$, *we have*

1. $\Pr[\mathbf{x} \leftarrow \mathcal{D}_{\sigma,\Lambda_q^{\mathbf{u}}(\mathbf{A})} | \|\mathbf{x}\| > \sqrt{m} \cdot \sigma] \leq \exp(-m/2)$.
2. *There exists a probabilistic polynomial time* (*PPT*) *algorithm* ***SamplePre***$(\mathbf{A}, \mathbf{T_A}, \mathbf{D}, \sigma)$ *which outputs a matrix* $\mathbf{X} \in \Lambda_q^{\mathbf{D}}(\mathbf{A})$ *distributed statistically close to* $\mathcal{D}_{\sigma,\Lambda_q^{\mathbf{D}}(\mathbf{A})}$.
3. $\Pr[\mathbf{R} \leftarrow \mathcal{D}_{\sigma,\Lambda_q^{\mathbf{D}}(\mathbf{A})} | \|\mathbf{R}\|_2 > m \cdot \sigma] \leq \exp(-m^2/2)$.
4. $\Pr[\mathbf{S} \leftarrow \{-1, 1\}^{m\times m} | \|\mathbf{S}\|_2 > 20\sqrt{m}] \leq \exp(-m/2)$.

Lemma 2.3. [1] *Given* $q > 2$ *and* $m > (n + 1)\log q + \omega(\log n)$. *For some polynomial* $k = k(n)$, *choose three uniformly random matrices* $\mathbf{U} \in \{-1, 1\}^{m\times k}$, $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$, *and* $\mathbf{B} \in \mathbb{Z}_q^{n\times k}$. *For all vectors* $\mathbf{r} \in \mathbb{Z}_q^m$, *the distributions* $(\mathbf{A}, \mathbf{AU}, \mathbf{U}^T\mathbf{r})$ *and* $(\mathbf{A}, \mathbf{B}, \mathbf{U}^T\mathbf{r})$ *are statistically indistinguishable*.

Definition 2.4. If $\Pr_{e\leftarrow\chi_n}[|e| > B] \leq 2^{-\tilde{\Omega}(n)}$, the distribution ensemble $\{\chi_n\}_{n\in\mathbb{N}}$ is called B-bounded over integers LWE.

LWE. Given $n, q \geq 1$, $m \geq O(n \log q)$, and a distribution $\chi = \chi(n)$ over $\mathbb{Z}$, the $\mathrm{LWE}_{n,q,m,\chi}$ problem is defined to distinguish between the following two distributions:

$$(\mathbf{A}, \mathbf{A}^T\mathbf{s} + \mathbf{e}) \quad \text{and} \quad (\mathbf{A}, \mathbf{u}),$$

where $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$, $\mathbf{s} \in \mathbb{Z}_q^n$, $\mathbf{e} \in \chi^m$, $\mathbf{u} \in \mathbb{Z}_q^m$ are independently sampled.

Lemma 2.5. [11, 35] *Given* $q = q(n) \leq 2^n$ *and* B*-bounded distribution* $\chi = \chi(n)$ *where* $B = B(n)$, $q/B \geq 2^{n^{\varepsilon}}$, *for all* $\varepsilon > 0$, *we have that the* $\mathrm{LWE}_{n,q,m,\chi}$ *problem is as hard as the quantum hardness of* $\mathrm{SIV}\,\mathrm{P}_\gamma$ *and the classical hardness of* $\mathrm{GapSV}\,\mathrm{P}_\gamma$ *where* $\gamma = 2^{\Omega(n^{\varepsilon})}$.

2.2 Lattice trapdoors and matrix embeddings

Gadget matrix. For integers $q \geq 2$ and $n \geq 1$, Micciancio and Peikert [30] defined a special matrix (known as gadget matrix) as $\mathbf{G} := \mathbf{I}_n \otimes \mathbf{g} \in \mathbb{Z}_q^{n\times M}$ for $M = n\lceil \log q \rceil$ and $\mathbf{g} := (1, 2, \ldots, 2^{\lceil \log q \rceil - 1}) \in \mathbb{Z}_q^{\lceil \log q \rceil}$, and defined the inversion function as $\mathbf{G}^{-1} : \mathbb{Z}_q^{n\times M} \to \{0, 1\}^{M\times M}$. Hence, given any matrix $\mathbf{A} \in \mathbb{Z}_q^{n\times M}$, we have $\mathbf{G} \cdot \mathbf{G}^{-1}(\mathbf{A}) = \mathbf{A}$, where $\|\mathbf{G}^{-1}(\mathbf{A})\|_2 \leq M$ (by Claim 2.3 in ref. [8]). In addition, $\mathbf{G} \in \mathbb{Z}_q^{n\times M}$ can be extended to a matrix $\widetilde{\mathbf{G}} \in \mathbb{Z}_q^{n\times M'}$ for $M' > M$ (e.g., by padding zero) and the corresponding inversion function $\widetilde{\mathbf{G}}^{-1}$ is defined in a similar way.

Definition 2.6. ($\mathbf{G}$-trapdoor, [30]) Let $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ and $\mathbf{G} \in \mathbb{Z}_q^{n\times w}$ be a gadget matrix, where $m \geq w \geq n$. A matrix $\mathbf{R} \in \mathbb{Z}^{(m-w)\times w}$ satisfying

$$\mathbf{A}\begin{bmatrix}\mathbf{R}\\ \mathbf{I}_w\end{bmatrix} = \mathbf{HG}$$

is called a $\mathbf{G}$-trapdoor for $\mathbf{A}$, where $\mathbf{H} \in \mathbb{Z}_q^{n\times n}$ viewed as the tag of the trapdoor is invertible.

Based on the notion of $\mathbf{G}$-trapdoor, Micciancio and Peikert [30] gave the following lemma showing that a good basis for $\Lambda_q^{\perp}(\mathbf{A})$ can be obtained from the $\mathbf{G}$-trapdoor $\mathbf{R}$.

Lemma 2.7. [30] *Given a matrix* $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ *along with a* $\mathbf{G}$*-trapdoor* $\mathbf{R} \in \mathbb{Z}^{(m-w)\times w}$ *and a tag* $\mathbf{H} \in \mathbb{Z}_q^{n\times n}$*. Let* $\mathbf{S} \in \mathbb{Z}^{w\times w}$ *be any basis for* $\Lambda_q^{\perp}(\mathbf{G})$*. Then the matrix*

$$\mathbf{S}_{\mathbf{A}} = \begin{bmatrix} \mathbf{I}_{m-w} & \mathbf{R} \\ \mathbf{0} & \mathbf{I}_w \end{bmatrix} \begin{bmatrix} \mathbf{I}_{m-w} & \mathbf{0} \\ \mathbf{W} & \mathbf{S} \end{bmatrix}$$

is a basis for $\Lambda_q^{\perp}(\mathbf{A})$*, where* $\mathbf{W} \in \mathbb{Z}^{w\times(m-w)}$ *is an arbitrary solution to* $\mathbf{GW} = -\mathbf{H}^{-1}\mathbf{A}(\mathbf{I}_{m-w}|\mathbf{0})^T \bmod q$ *and* $\|\mathbf{S}_{\mathbf{A}}\|_{\mathrm{GS}} \le \|\mathbf{S}\|_{\mathrm{GS}} \cdot (\|\mathbf{R}\|_2 + 1)$.

Throughout the paper we need the following algorithms which show the properties of lattice trapdoors.

Lemma 2.8. [1,13] *Given* $n \ge 1$, $q \ge 2$, $\bar{m}, m', \tilde{m} \ge n$, *and* $m = \Theta(n \log q)$, *we have the following polynomial-time algorithms*:

- *There is a PPT algorithm* ***TrapGen***$(1^n, 1^m, q)$ *that outputs a matrix* $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ *distributed statistically close to uniform and a short basis* $\mathbf{T}_{\mathbf{A}} \in \mathbb{Z}^{m\times m}$ *for* $\Lambda_q^{\perp}(\mathbf{A})$ *where* $\|\mathbf{T}_{\mathbf{A}}\|_{\mathrm{GS}} \le O(\sqrt{n \log q})$.
- *There is a PPT algorithm* ***SampleBasisLeft***$(\mathbf{A}, \mathbf{B}, \mathbf{T}_{\mathbf{A}}, s)$ *that, given matrices* $\mathbf{A}, \mathbf{B} \in \mathbb{Z}_q^{n\times m}$*, a short basis* $\mathbf{T}_{\mathbf{A}}$ *for lattice* $\Lambda_q^{\perp}(\mathbf{A})$ *and* $s \ge \|\mathbf{T}_{\mathbf{A}}\|_{\mathrm{GS}} \cdot \omega(\sqrt{\log m})$*, outputs a basis* $\mathbf{T}_{(\mathbf{A}|\mathbf{B})}$ *for* $\Lambda_q^{\perp}(\mathbf{A}|\mathbf{B})$ *distributed statistically close to* $(\mathcal{D}_{s,\Lambda_q^{\perp}(\mathbf{A}|\mathbf{B})})^m$.
- *There is a PPT algorithm* ***SampleBasisRight***$(\mathbf{A}, \mathbf{G}, \mathbf{S}, \mathbf{T}_{\mathbf{G}}, s)$ *that, given matrices* $\mathbf{A}, \mathbf{G} \in \mathbb{Z}_q^{n\times m}$*, a low-norm matrix* $\mathbf{S} \in \mathbb{Z}^{m\times m}$*, a short basis* $\mathbf{T}_{\mathbf{G}}$ *for lattice* $\Lambda_q^{\perp}(\mathbf{G})$ *and* $s \ge \sqrt{5} \cdot (\|\mathbf{S}\|_2 + 1) \cdot \omega(\sqrt{\log m})$*, outputs a basis* $\mathbf{T}_{(\mathbf{A}|\mathbf{AS}+\mathbf{G})}$ for $\Lambda_q^{\perp}(\mathbf{A}|\mathbf{AS} + \mathbf{G})$ *distributed statistically close to* $(\mathcal{D}_{s,\Lambda_q^{\perp}(\mathbf{A}|\mathbf{AS}+\mathbf{G})})^m$.
- *There exists a PPT algorithm* ***SampleLeft***$(\mathbf{A}, \mathbf{B}, \mathbf{T}_{\mathbf{A}}, \mathbf{D}, \tau)$ *that, given matrices* $\mathbf{A}, \mathbf{D} \in \mathbb{Z}_q^{n\times m}$, $\mathbf{B} \in \mathbb{Z}_q^{n\times \bar{m}}$*, a short basis* $\mathbf{T}_{\mathbf{A}}$ *for lattice* $\Lambda_q^{\perp}(\mathbf{A})$ *and* $\tau \ge \|\mathbf{T}_{\mathbf{A}}\|_{\mathrm{GS}} \cdot \omega(\sqrt{\log(m + \bar{m})})$*, outputs a matrix* $\mathbf{R} \in \mathbb{Z}_q^{(m+\bar{m})\times m}$ *distributed statistically close to* $\mathcal{D}_{\tau,\Lambda_q^{\mathbf{D}}(\mathbf{A}|\mathbf{B})}$*. Furthermore, for any random matrices* $\mathbf{B}', \mathbf{D}' \in \mathbb{Z}_q^{n\times m'}$ *and* $\tau \ge \|\mathbf{T}_{\mathbf{A}}\|_{\mathrm{GS}} \cdot \omega(\sqrt{\log(m + m')})$*. Let* $\widetilde{\mathbf{R}} \leftarrow \mathcal{D}_{s,\mathbb{Z}^{(m+m')\times m'}}$ *and compute* $\widetilde{\mathbf{D}} = (\mathbf{A}|\mathbf{B}') \cdot \widetilde{\mathbf{R}}$*. Then, the distribution* $(\mathbf{A}, \widetilde{\mathbf{D}}, \widetilde{\mathbf{R}})$ *is statistically close to the distribution* $(\mathbf{A}, \mathbf{D}', \mathbf{R}')$*, where* $\mathbf{R}' \leftarrow$ ***SampleLeft***$(\mathbf{A}, \mathbf{B}', \mathbf{T}_{\mathbf{A}}, \mathbf{D}', \tau)$.
- *There exists a PPT algorithm* ***SampleRight***$(\mathbf{A}, \mathbf{G}, \mathbf{R}, \mathbf{T}_{\mathbf{G}}, \mathbf{D}, \tau)$ *that, on input matrices* $\mathbf{A}, \mathbf{G} \in \mathbb{Z}_q^{n\times m}$*, a low-norm matrix* $\mathbf{R} \in \mathbb{Z}^{m\times m}$*, a basis* $\mathbf{T}_{\mathbf{G}}$ *for lattice* $\Lambda_q^{\perp}(\mathbf{G})$*, a random matrix* $\mathbf{D} \in \mathbb{Z}_q^{n\times \tilde{m}}$*, and a parameter* $\tau \ge \sqrt{5} \cdot (\|\mathbf{R}\|_2 + 1) \cdot \omega(\sqrt{\log m})$*, outputs a matrix* $\mathbf{E} \in \mathbb{Z}_q^{2m\times \tilde{m}}$ *distributed statistically close to* $\mathcal{D}_{\tau,\Lambda_q^{\mathbf{D}}(\mathbf{F})}$ *where* $\mathbf{F} := (\mathbf{A}|\mathbf{AR} + \mathbf{G})$.
- *There is a publicly known basis* $\mathbf{T}_{\mathbf{G}}$ *for* $\Lambda_q^{\perp}(\mathbf{G})$ *where* $\mathbf{G} \in \mathbb{Z}_q^{n\times m}$ *is a gadget matrix and* $\|\mathbf{T}_{\mathbf{G}}\|_{\mathrm{GS}} \le \sqrt{5}$.

In fact, the above **SampleBasisRight** algorithm can be obtained from Lemma 2.7, and **SampleRight** algorithm can be obtained by combining Lemma 2.7 with **SamplePre** algorithm of Lemma 2.2. We will show how to obtain a variant of **SampleRight** algorithm which we call **ExtSampleRight**. Looking ahead, the security proof of our scheme relies on the following **ExtSampleRight** algorithm.

Lemma 2.9. *Given* $n \ge 1$, $q \ge 2$, $m' \ge n$, *and* $m = \Theta(n \log q)$*. Let matrices* $\mathbf{A}, \mathbf{G} \in \mathbb{Z}_q^{n\times m}$*, two low-norm matrices* $\mathbf{R}_1, \mathbf{R}_2 \in \mathbb{Z}^{m\times m}$*, a basis* $\mathbf{T}_{\mathbf{G}}$ *for lattice* $\Lambda_q^{\perp}(\mathbf{G})$ *with* $\|\mathbf{T}_{\mathbf{G}}\|_{\mathrm{GS}} \le \sqrt{5}$*, a random matrix* $\mathbf{D} \in \mathbb{Z}_q^{n\times m'}$*, and a parameter* $\sigma \ge \sqrt{5} \cdot (\|\mathbf{R}_1 + \mathbf{R}_2\|_2 + 1) \cdot \omega(\sqrt{\log 3m})$*. Then there exists a PPT algorithm* ***ExtSampleRight***$(\mathbf{A}, \mathbf{G}, \mathbf{R}_1 + \mathbf{R}_2, \mathbf{T}_{\mathbf{G}}, \mathbf{D}, \sigma)$ *that outputs a matrix* $\mathbf{E} \in \mathbb{Z}_q^{3m\times m'}$ *distributed statistically close to* $\mathcal{D}_{\sigma,\Lambda_q^{\mathbf{D}}(\mathbf{F})}$*, where* $\mathbf{F} := (\mathbf{A}|\mathbf{AR}_1 + \mathbf{G}|\mathbf{AR}_2 + \mathbf{G})$, $\mathbf{F} := (\mathbf{A}|\mathbf{AR}_1|\mathbf{AR}_2 + \mathbf{G})$ or $\mathbf{F} := (\mathbf{A}|\mathbf{AR}_1 + \mathbf{G}|\mathbf{AR}_2)$.

Proof. It suffices to prove the case when $\mathbf{F} \coloneqq (\mathbf{A}|\mathbf{AR}_1 + \mathbf{G}|\mathbf{AR}_2 + \mathbf{G})$. Since

$$\mathbf{F}\begin{bmatrix} -(\mathbf{R}_1 + \mathbf{R}_2) \\ \mathbf{I}_m \\ \mathbf{I}_m \end{bmatrix} = 2\mathbf{I}_n\mathbf{G},$$

the matrix $-(\mathbf{R}_1 + \mathbf{R}_2)$ is a $\mathbf{G}$-trapdoor for $\mathbf{F}$ and $2\mathbf{I}_n$ is the tag. By Lemma 2.7, we can generate a basis $\mathbf{S}_\mathbf{F} \in \mathbb{Z}_q^{3m\times 3m}$ for $\Lambda_q^\perp(\mathbf{F})$ with $\|\mathbf{S}_\mathbf{F}\|_{\mathrm{GS}} \le \sqrt{5}\cdot(\|\mathbf{R}_1 + \mathbf{R}_2\|_2 + 1)$. With the basis $\mathbf{S}_\mathbf{F}$, sample $\mathbf{E} \in \mathbb{Z}_q^{3m\times m'} \leftarrow \mathbf{SamplePre}(\mathbf{F}, \mathbf{S}_\mathbf{F}, \mathbf{D}, \sigma)$, which is distributed statistically close to $\mathcal{D}_{\sigma,\Lambda_q^{\mathbf{D}}(\mathbf{F})}$, as required. □

Remark 2.10. It is not hard to see that the aforementioned lemma can be extended to the case when $\mathbf{F} \coloneqq (\mathbf{A}|\mathbf{AR}_1 + v_1\mathbf{G}|\cdots|\mathbf{AR}_t + v_t\mathbf{G})$ for any $v_i \in \mathbb{Z}$, $i \in [t]$ and $t \ge 1$ as long as $v_1 + \cdots + v_t \ne 0$ holds.

Matrix embeddings. Boneh et al. [8] proposed an ABE scheme for arithmetic circuits by introducing an approach of embedding circuits into LWE matrices. This method has subsequently been used for a number of other LWE-based constructions such as PE [20], constrained PRFs [10], private puncturable PRFs [9], and watermarking for PRFs [22]. Below we summarize the properties of the matrix embeddings.

Lemma 2.11. [8,20] *Given parameters (λ, n, m, q, χ), where λ is a security parameter and χ is a B-bounded distribution. For any matrices $\mathbf{B}_1, \ldots, \mathbf{B}_\ell \in \mathbb{Z}_q^{n\times m}$, any Boolean circuit $f : \{0,1\}^\ell \to \{0,1\}$ of depth $\le d$, and any $\mathbf{x} \in \{0,1\}^\ell$, if*

$$\mathbf{c}_i = (\mathbf{B}_i + \mathbf{x}_i\mathbf{G})^T\mathbf{s} + \mathbf{e}_i \quad \forall i \in [\ell]$$

for some vector $\mathbf{s} \in \mathbb{Z}_q^n$ and $\mathbf{e}_i \leftarrow \chi^m$ for $i \in [\ell]$, then there exist algorithms $(\mathbf{Eval}_{pk}, \mathbf{Eval}_{ct}, \mathbf{Eval}_{sim})$.

- $\mathbf{Eval}_{pk}(f, (\mathbf{B}_1, \ldots, \mathbf{B}_\ell)) \to \mathbf{B}_f$: *On input a circuit f and ℓ matrices $(\mathbf{B}_1, \ldots, \mathbf{B}_\ell)$, output a matrix $\mathbf{B}_f$.*
- $\mathbf{Eval}_{ct}(f, \{(\mathbf{B}_i, \mathbf{x}_i, \mathbf{c}_i)\}_{i\in[\ell]}) \to \mathbf{c}_f$: *On input a circuit f, ℓ matrices $(\mathbf{B}_1, \ldots, \mathbf{B}_\ell)$, length ℓ string $\mathbf{x}$, and ℓ vectors $(\mathbf{c}_1, \ldots, \mathbf{c}_\ell)$, output a vector $\mathbf{c}_f$, satisfying*

  $$\mathbf{c}_f = (\mathbf{B}_f + f(\mathbf{x})\mathbf{G})^T\mathbf{s} + \mathbf{e}_f,$$

  where $\mathbf{B}_f = \mathbf{Eval}_{pk}(f, (\mathbf{B}_1, \ldots, \mathbf{B}_\ell))$ and $\|\mathbf{e}_f\| \le B\sqrt{m}\cdot(m+1)^d$ with all but negligible probability.
- $\mathbf{Eval}_{sim}(f, \{(\mathbf{S}_i^*, \mathbf{x}_i^*)\}_{i\in[\ell]}, \mathbf{A}) \to \mathbf{S}_f^*$: *On input a circuit f, ℓ matrices $\mathbf{S}_1^*, \ldots, \mathbf{S}_\ell^* \in \mathbb{Z}_q^{m\times m}$, a matrix $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$, and length ℓ string $\mathbf{x}^*$, output a matrix $\mathbf{S}_f^* \in \mathbb{Z}_q^{m\times m}$, satisfying*

  $$\mathbf{AS}_f^* - f(\mathbf{x}^*)\mathbf{G} = \mathbf{B}_f,$$

  where $\mathbf{B}_f = \mathbf{Eval}_{pk}(f, (\mathbf{AS}_1^* - \mathbf{x}_1^*\mathbf{G}, \ldots, \mathbf{AS}_\ell^* - \mathbf{x}_\ell^*\mathbf{G}))$. Moreover, if $\mathbf{S}_1^*, \ldots, \mathbf{S}_\ell^* \in \{-1,1\}^{m\times m}$, then $\|\mathbf{S}_f^*\|_2 \le 20\sqrt{m}\cdot(m+1)^d$ with all but negligible probability.

2.3 The CS method

The CS method, introduced by Naor et al. [32], has been extensively used to realize user revocation. The CS algorithm first builds a complete binary tree BT and employs the following notation: If y is a non-leaf node, then y_L and y_R denote the left and right children of y, respectively; if y' is a leaf node, the set $\mathsf{Path}(y')$ denotes all nodes on the path from y' to the root (including y' and the root). The CS algorithm runs a node selection algorithm called the $\mathsf{KUNodes}$ algorithm as described in Algorithm 1 that, taking as input BT and a revocation list RL, outputs a set of nodes Y. As shown in ref. [32], the set Y generated by $\mathsf{KUNodes}(\mathsf{BT},\mathsf{RL})$ has a size at most $r\log\frac{N}{r}$, where $r = |\mathsf{RL}|$.

Algorithm 1: KUNodes algorithm

Input: BT, RL.
Output: Y.
1. $X, Y \leftarrow \varnothing$; $\forall y \in \mathsf{RL}$, set $X \leftarrow X \cup \mathsf{Path}(y)$.
2. $\forall y \in X$:
   If $y_{\mathrm{L}} \notin X$, set $Y \leftarrow Y \cup \{y_{\mathrm{L}}\}$;
   If $y_{\mathrm{R}} \notin X$, set $Y \leftarrow Y \cup \{y_{\mathrm{R}}\}$.
3. If $Y = \varnothing$, set $Y \leftarrow Y \cup \{\mathsf{root}\}$.
4. Return Y.

3 Revocable key-policy ABPRE

We introduce the notion of revocable KP-ABPRE and its game-based definition of security. Let $\mathcal{M}$ be a message space, $\mathcal{X}$ be an attribute space, and $\mathcal{I}$ be an index space. A revocable KP-ABPRE scheme for a family of functions $\mathcal{F} = \{f : \mathcal{X} \to \{0,1\}\}$ contains polynomial-time algorithms (**Setup**, **KeyGen**, **Enc**, **Dec**, **ReKeyGen**, **ReEnc**, and **ReDec**), which is defined as follows:

- **Setup**(1^λ). Take as input a security parameter λ, output a state information ST, a public key pk, and a master secret key msk.
- **KeyGen**(msk, f). Take as input msk and a function $f \in \mathcal{F}$, output a secret key sk_f.
- **Enc**$(mpk, \mu, x, \mathsf{RL})$. Take as input mpk, a message $\mu \in \mathcal{M}$, an attribute $x \in \mathcal{X}$, and a revocation list $\mathsf{RL} \subseteq \mathcal{I}$, output a ciphertext c. We call it "fresh" ciphertext.
- **Dec**(sk_f, c, x). Take as input sk_f, c, and $x \in \{0,1\}^\ell$, output a message $\mu \in \mathcal{M}$ if $f(x) = 0$, and $\perp$ otherwise.
- **ReKeyGen** $(sk_f, \mathsf{ST}, g, I)$. Take as input sk_f, a state ST, a function $g \in \mathcal{F}$, and an index $I \in \mathcal{I}$, output a re-encryption key $rk_{f\to(g,I)}$, and an updated state ST.
- **ReEnc** $(rk_{f\to(g,I)}, c, x)$. Take as input a re-encryption key $rk_{f\to(g,I)}$, a ciphertext c, and $x \in \mathcal{X}$, output a re-encrypted ciphertext $c_{f\to(g,I)}$ if $f(x) = 0$.[3] Otherwise, output $\perp$. Note that the ciphertext c with respect to x is invalid for the re-encryption key $rk_{f\to(g,I)}$ if $f(x) = 1$.
- **ReDec** $(sk_g, c_{f\to(g,I)})$. Take sk_g (which, like sk_f, was generated by the **KeyGen** algorithm) and a re-encrypted ciphertext $c_{f\to(g,I)}$ as input, output a message $\mu' \in \mathcal{M}$ or $\perp$ otherwise.

Correctness. There are two cases for the correctness of the revocable KP-ABPRE scheme: one case for fresh ciphertext, and the other case for re-encrypted ciphertext. We say the correctness of the revocable KP-ABPRE scheme is guaranteed if the following holds:

- For all $(\mathsf{ST}, pk, msk) \leftarrow$ **Setup**(1^λ), all $sk_f \leftarrow$ **KeyGen**(msk, f) for $f \in \mathcal{F}$, all message $\mu \in \mathcal{M}$, all $\mathsf{RL} \subseteq \mathcal{I}$, and all attribute $x \in \mathcal{X}$, we have

$$\Pr[\mathbf{Dec}(sk_f, \mathbf{Enc}(pk, \mu, x, \mathsf{RL}), x) = \mu] = 1 - \mathrm{negl}(\lambda)$$

  if $f(x) = 0$.
- For all $(\mathsf{ST}, pk, msk) \leftarrow$ **Setup**(1^λ), all $sk_f \leftarrow$ **KeyGen**(msk, f), $sk_g \leftarrow$ **KeyGen**(msk, g) for $f, g \in \mathcal{F}$, all message $\mu \in \mathcal{M}$, all $\mathsf{RL} \subseteq \mathcal{I}$, and all attribute $x \in \mathcal{X}$, we have

$$\Pr[\mathbf{ReDec}(sk_g, \mathbf{ReEnc}(rk_{f\to(g,I)}, c, x)) = \mu] = 1 - \mathrm{negl}(\lambda)$$

3 Note that we can check if $f(x) = 0$.

if $f(x) = 0$ and $I \notin \mathsf{RL}$, where $c = \mathbf{Enc}(pk, \mu, x, \mathsf{RL})$ and $rk_{f\to(g,I)} \leftarrow \mathbf{ReKeyGen}(sk_f, \mathsf{ST}, g, I)$. Otherwise, we have

$$\Pr[\mathbf{ReDec}(sk_g, \mathbf{ReEnc}(rk_{f\to(g,I)}, c, x)) = \perp] = 1 - \mathrm{negl}(\lambda).$$

Definition 3.1. [Multi/Single-hop]. A revocable KP-ABPRE scheme is multi-hop if a semi-trusted proxy can perform further re-encryption procedures on any re-encrypted ciphertext. Otherwise, it is single-hop.

3.1 Security definition

Since revocable ABPRE is ABPRE with revocation mechanism, we adapt the CPA security of ABPRE given in ref. [26] to obtain the CPA security for revocable KP-ABPRE. We consider the CPA security in the selective model, where the adversary $\mathcal{A}$ is required to declare the challenge attribute x^* and revocation list RL^* beforehand. We define the following game $\mathbf{Expt}^{\mathrm{CPA}}_{\mathcal{A}}(1^\lambda)$ that describes the interaction between a challenger and a PPT adversary $\mathcal{A}$.

1. **Setup**: $\mathcal{A}$ announces an attribute $x^* \in \mathcal{X}$ and a revocation list $\mathsf{RL}^* \subseteq \mathcal{I}$. The challenger computes $(\mathsf{ST}, pk, msk) \leftarrow \mathbf{Setup}(1^\lambda)$ and returns pk to $\mathcal{A}$.
2. **Query phase 1**: The challenger and $\mathcal{A}$ proceed as follows:
   – **Key query** $O_{\mathbf{KeyGen}}$: $\mathcal{A}$ sends a function $f \in \mathcal{F}$ to the challenger, the challenger replies with sk_f by running $sk_f \leftarrow \mathbf{KeyGen}(msk, f)$.
   – **Re-encryption key query** $O_{\mathbf{ReKeyGen}}$: $\mathcal{A}$ sends a pair $(f, g, I) \in \mathcal{F} \times \mathcal{F} \times \mathcal{I}$ to the challenger, the challenger returns $\perp$ if $I \in \mathsf{ST}$. Otherwise, the challenger returns a re-encryption key $rk_{f\to(g,I)}$ by running $rk_{f\to(g,I)} \leftarrow \mathbf{ReKeyGen}(sk_f, \mathsf{ST}, g, I)$ and updates the state $\mathsf{ST} \leftarrow \mathsf{ST} \cup \{I\}$.
   – **Re-encryption query** $O_{\mathbf{ReEnc}}$: $\mathcal{A}$ sends (c, x, f, g, I) to the challenger, where $f(x) = 0$, the challenger returns a re-encrypted ciphertext $c_{f\to(g,I)}$ by running $c_{f\to(g,I)} \leftarrow \mathbf{ReEnc}(rk_{f\to(g,I)}, c, x)$ if $I \in \mathsf{ST}$. Otherwise, the challenger computes a re-encryption key $rk_{f\to(g,I)}$ as in O_{ReKeyGen}, updates the state $\mathsf{ST} \leftarrow \mathsf{ST} \cup \{I\}$, and returns a re-encrypted ciphertext $c_{f\to(g,I)}$ by running $c_{f\to(g,I)} \leftarrow \mathbf{ReEnc}(rk_{f\to(g,I)}, c, x)$.
3. **Challenge query**: $\mathcal{A}$ submits a pair of messages (μ_0, μ_1), the challenger chooses a uniformly random bit $b \leftarrow \{0, 1\}$ and returns $c^* \leftarrow \mathbf{Enc}(pk, \mu, x^*, \mathsf{RL}^*)$.
4. **Query phase 2**: The same as Query phase 1.
5. For simplicity, we use **legal** to denote the event, where $O_{\mathbf{KeyGen}}$ is subject to the condition: $f(x^*) = 1$, $O_{\mathbf{ReKeyGen}}$ is subject to the condition: $f(x^*) = 1$ or $(f(x^*) = 0, I \in \mathsf{RL}^*)$, and $O_{\mathbf{ReEnc}}$ is subject to the condition: $(x \neq x^*, f(x^*) = 1)$ or $(f(x^*) = 0, I \in \mathsf{RL}^*)$.
6. **Output**: $\mathcal{A}$ returns a bit $\tilde{b} \in \{0, 1\}$, C outputs $b' = \tilde{b}$ if **legal**, and a uniformly random bit b' otherwise. $\mathcal{A}$'s advantage in winning the experiment $\mathbf{Expt}^{\mathrm{CPA}}_{\mathcal{A}}(1^\lambda)$ is defined

$$\mathrm{Adv}^{\mathrm{CPA}}_{\mathcal{A}}(1^\lambda) = |\Pr[b' = b] - 1/2|.$$

In the above game, the restrictions prevent the adversary to trivially win the game by decrypting the challenge ciphertext c^*. It is not hard to prove that for $O_{\mathbf{KeyGen}}$ and $O_{\mathbf{ReKeyGen}}$. Recall that the ciphertext c with respect to x is invalid for the re-encryption key $rk_{f\to(g,I)}$ if $f(x) = 1$. Thus, given any query (c, x, f, g, I), the challenger can check if $f(x) = 0$, so it is reasonable to assume that $f(x) = 0$, because the challenger can reject the query if $f(x) = 1$. Now that $f(x) = 0$, we have $x \neq x^*$ or $(x = x^*, I \in \mathsf{RL}^*)$. This is because if $x = x^*$, then $f(x^*) = 0$ and hence the adversary can obtain a re-encrypted ciphertext $c^*_{f\to(g,I)}$ from the challenge ciphertext c^* with respect to x^*, which means that she can trivially win the game by decrypting $c^*_{f\to(g,I)}$ using the secret key sk_g if $I \notin \mathsf{RL}^*$, where $g(x^*) = 1$. Since $f(x) = 0$, the condition $(x = x^*, I \in \mathsf{RL}^*)$ is equivalent to

$(f(x^*) = 0, I \in \mathsf{RL}^*)$. We remark that the condition $(x \neq x^*, f(x^*) = 1)$ is a little strong, since we do not have $f(x^*) = 1$ from $x \neq x^*$ and $f(x) = 0$.

A stronger notion is the adaptive security model, where $\mathcal{A}$ announces the challenge attribute x^* and revocation list RL^* after she sees the public key.

Definition 3.2. (Selectively CPA). We say a revocable key-policy ABPRE scheme is selectively CPA secure in the standard model, if $\mathcal{A}$ wins the experiment $\mathbf{Expt}_{\mathcal{A}}^{\mathrm{CPA}}(1^\lambda)$ only with negligible advantage.

4 Revocable KP-ABPRE from LWE

In this section, we construct a revocable KP-ABPRE scheme based on the LWE problem, building upon the selectively secure LWE-based ABE [8]. To satisfy the correctness requirement that the decryption algorithm outputs $\perp$ with all but negligible probability when $I \in \mathsf{RL}$, like ref. [27], we define the encoding function $\mathsf{encode} : \{0,1\} \to \{0,1\}^k$ for $k = \omega(\log \lambda)$, such that for each $\mu \in \{0,1\}$, we have $\mathsf{encode}(\mu) = (\mu, 0, \ldots, 0) \in \{0,1\}^k$. Given a family of functions $\mathcal{F} = \{f : \{0,1\}^\ell \to \{0,1\}\}$ of depth $\leq d$ (represented as Boolean circuits), an attribute space $\mathcal{X} = \{0,1\}^\ell$, a message space $\mathcal{M} = \{0,1\}$, and an index space $\mathcal{I} = [N]$, our revocable KP-ABPRE construction that works for any ℓ, $d = \mathrm{poly}(\lambda)$ is described as follows:

- **Setup**$(1^\lambda, \ell, N, L)$. Take as input the security parameter λ, the maximum length of the attributes ℓ, and the maximum expected number of users N, the setup algorithm proceeds as follows:
  1. Generate $\mathbf{GenTrap}(1^n, m, q) \to (\mathbf{A}, \mathbf{T}_\mathbf{A})$.
  2. Sample a uniformly random matrix $\mathbf{D} \in \mathbb{Z}_q^{n \times k}$.
  3. Sample ℓ uniformly random matrices $\mathbf{B}_1, \ldots, \mathbf{B}_\ell \in \mathbb{Z}_q^{n \times m}$.
  4. Build a complete binary tree BT with N leaf nodes, and choose uniformly random matrix $\mathbf{U}_y \in \mathbb{Z}_q^{n \times m}$ as the "identifier" for each node $y \in \mathsf{BT}$.
  5. Initialize the state $\mathsf{ST} = \varnothing$, which records the assigned indices so far.

  Output a state ST, a public key $pk := (\mathbf{A}, \mathbf{D}, \mathbf{B}_1, \ldots, \mathbf{B}_\ell, \mathsf{BT})$, and a master secret key $msk := (\mathbf{T}_\mathbf{A})$.
- **KeyGen** $(msk, f \in \mathcal{F})$. Take as input msk and a function $f \in \mathcal{F}$, compute $\mathbf{B}_f = \mathbf{Eval}_{pk}(f, (\mathbf{B}_1, \ldots, \mathbf{B}_\ell))$ and generate an extended trapdoor $\mathbf{T}_{(\mathbf{A}|\mathbf{B}_f)} \in \mathbb{Z}^{2m \times 2m}$ by running

$$\mathbf{T}_{(\mathbf{A}|\mathbf{B}_f)} \leftarrow \mathbf{SampleBasisLeft}(\mathbf{A}, \mathbf{B}_f, \mathbf{T}_\mathbf{A}, s).$$

  Output a secret key $sk_f := \mathbf{T}_{(\mathbf{A}|\mathbf{B}_f)}$.
- **Enc** $(pk, \mu, \mathbf{x}, \mathsf{RL})$. Take as input pk, a message $\mu \in \{0,1\}$, an attribute $\mathbf{x} \in \{0,1\}^\ell$, and a revocation list $\mathsf{RL} \subseteq [N]$, the encryption algorithm does:
  1. Choose uniformly at random a vector $\mathbf{s} \in \mathbb{Z}_q^n$, two error vectors $\mathbf{e}_0 \in \chi^m$, $\mathbf{e}_1 \in \chi^k$, and matrices $\mathbf{S}_{\hat{y}}, \mathbf{S}_i \in \{\pm 1\}^{m \times m}$ for each $\hat{y} \in \mathsf{KUNodes}(\mathsf{BT},\mathsf{RL})$ and $i \in [\ell]$.
  2. Set

$$\mathbf{H} = (\mathbf{A}|\mathbf{B}_1 + \mathbf{x}_1\mathbf{G}|\cdots|\mathbf{B}_\ell + \mathbf{x}_\ell\mathbf{G}) \in \mathbb{Z}_q^{n \times (\ell+1)m},$$
$$\mathbf{e} = (\mathbf{I}_m|\mathbf{S}_1|\cdots|\mathbf{S}_\ell)^T \cdot \mathbf{e}_0 \in \mathbb{Z}_q^{(\ell+1)m},$$

     and compute $\mathbf{c}_0 = \mathbf{H}^T\mathbf{s} + \mathbf{e} \in \mathbb{Z}_q^{(\ell+1)m}$.
  3. For each node $\hat{y} \in \mathsf{KUNodes}(\mathsf{BT},\mathsf{RL})$, compute $\hat{\mathbf{c}}_{\hat{y}} = \mathbf{U}_{\hat{y}}^T\mathbf{s} + \mathbf{S}_{\hat{y}}^T\mathbf{e}_0 \in \mathbb{Z}_q^m$. Set $\mathbf{c}_1 = \{\hat{\mathbf{c}}_{\hat{y}}\}_{\hat{y} \in \mathsf{KUNodes}(\mathsf{BT},\mathsf{RL})}$.
  4. Compute $\mathbf{c}_2 = \mathbf{D}^T\mathbf{s} + \mathbf{e}_1 + \lfloor q/2 \rfloor \cdot \mathsf{encode}(\mu) \in \mathbb{Z}_q^k$.

  Output a ciphertext $\mathbf{c} = (\mathbf{c}_0, \mathbf{c}_1, \mathbf{c}_2)$.

- **Dec**$(sk_f, \mathbf{c}, \mathbf{x})$. Parse $\mathbf{c} = (\mathbf{c}_0, \mathbf{c}_1, \mathbf{c}_2)$ and $\mathbf{c}_0 = (\mathbf{c}_{in}, \mathbf{c}_0^1, \ldots, \mathbf{c}_0^\ell)$. If $f(\mathbf{x}) = 1$, output $\perp$. Otherwise, the decryption algorithm does:
  1. Run **SamplePre**$((\mathbf{A}|\mathbf{B}_f), \mathbf{T}_{(\mathbf{A}|\mathbf{B}_f)}, \mathbf{D}, \sigma_1)$ to generate a low-norm matrix $\mathbf{R}_f \in \mathbb{Z}^{2m\times k}$ such that $(\mathbf{A}|\mathbf{B}_f)\cdot \mathbf{R}_f = \mathbf{D}$.
  2. Compute $\mathbf{c}_f = \mathbf{Eval}_{ct}(f, \{(\mathbf{B}_i, \mathbf{x}_i, \mathbf{c}_0^i)\}_{i\in[\ell]}) \in \mathbb{Z}_q^m$.
  3. Compute $\mathbf{w} = \mathbf{c}_2 - \mathbf{R}_f^T(\mathbf{c}_{in}|\mathbf{c}_f)$. Output $\mu' = 1$ if $|\lfloor q/2\rfloor - \mathbf{w}_1| < q/4$. Otherwise, output $\mu' = 0$.
- **ReKeyGen**$(sk_f, \mathsf{ST}, g, I)$. Take as input a secret key sk_f, a state ST, a function $g \in \mathcal{F}$, and an index $I \in [N]$, the re-encryption key generation algorithm does:
  1. If $I \in \mathsf{ST}$, output $\perp$. Otherwise, update $\mathsf{ST} \leftarrow \mathsf{ST} \cup \{I\}$.
  2. Compute $\mathbf{B}_g = \mathbf{Eval}_{pk}(g, (\mathbf{B}_1, \ldots, \mathbf{B}_\ell))$ and $\mathbf{B}_f = \mathbf{Eval}_{pk}(f, (\mathbf{B}_1, \ldots, \mathbf{B}_\ell))$.
  3. For each node $\gamma \in \mathsf{Path}(I)$, generate a low-norm matrix $\mathbf{R}_\gamma \in \mathbb{Z}^{3m\times 2m}$ such that $(\mathbf{A}|\mathbf{B}_f|\mathbf{U}_\gamma)\cdot \mathbf{R}_\gamma = (\mathbf{A}|\mathbf{B}_g)$ by running $\mathbf{R}_\gamma \leftarrow$ **SampleLeft**$((\mathbf{A}|\mathbf{B}_f), \mathbf{U}_\gamma, \mathbf{T}_{(\mathbf{A}|\mathbf{B}_f)}, (\mathbf{A}|\mathbf{B}_g), \sigma_2)$.

Output a re-encryption key $rk_{f\to(g,I)} \coloneqq (f, g, I, \{\mathbf{R}_\gamma\}_{\gamma\in\mathsf{Path}(I)})$ and an updated state ST.

- **ReEnc**$(rk_{f\to(g,I)}, \mathbf{c}, \mathbf{x})$. Parse $rk_{f\to(g,I)} = (f, g, I, \{\mathbf{R}_\gamma\}_{\gamma\in\mathsf{Path}(I)})$, $\mathbf{c} = (\mathbf{c}_0, \mathbf{c}_1, \mathbf{c}_2)$, $\mathbf{c}_0 = (\mathbf{c}_{in}, \mathbf{c}_0^1, \ldots, \mathbf{c}_0^\ell)$, and $\mathbf{c}_1 = \{\hat{\mathbf{c}}_{\hat{\gamma}}\}_{\hat{\gamma}\in\mathsf{KUNodes(BT,RL)}}$. The re-encryption algorithm proceeds as follows:
  1. Compute $\mathbf{c}_f = \mathbf{Eval}_{ct}(f, \{(\mathbf{B}_i, \mathbf{x}_i, \mathbf{c}_0^i)\}_{i\in[\ell]}) \in \mathbb{Z}_q^m$.
  2. For all $\gamma \in \mathsf{Path}(I)$, $\hat{\gamma} \in \mathsf{KUNodes(BT,RL)}$, compute $\mathbf{c}_{\gamma,\hat{\gamma}} = \mathbf{R}_\gamma^T \cdot (\mathbf{c}_{in}|\mathbf{c}_f|\hat{\mathbf{c}}_{\hat{\gamma}}) \in \mathbb{Z}_q^{2m}$.

Output a re-encrypted ciphertext $\mathbf{c}_{f\to(g,I)} = (\tilde{\mathbf{c}}, \mathbf{c}_2)$, where $\tilde{\mathbf{c}} = \{\mathbf{c}_{\gamma,\hat{\gamma}}\}_{\gamma\in\mathsf{Path}(I),\hat{\gamma}\in\mathsf{KUNodes(BT,RL)}}$.

- **ReDec**$(sk_g, \mathbf{c}_{f\to(g,I)})$. Take as input $sk_g = \mathbf{T}_{(\mathbf{A}|\mathbf{B}_g)}$ and $\mathbf{c}_{f\to(g,I)} = (\tilde{\mathbf{c}}, \mathbf{c}_2)$, where $\tilde{\mathbf{c}} = \{\mathbf{c}_{\gamma,\hat{\gamma}}\}_{\gamma\in\mathsf{Path}(I),\hat{\gamma}\in\mathsf{KUNodes(BT,RL)}}$. The re-decryption algorithm does:
  1. Compute $\mathbf{B}_g = \mathbf{Eval}_{pk}(g, (\mathbf{B}_1, \ldots, \mathbf{B}_\ell))$ and run **SamplePre**$((\mathbf{A}|\mathbf{B}_g), \mathbf{T}_{(\mathbf{A}|\mathbf{B}_g)}, \mathbf{D}, \sigma_1)$ to generate a low-norm matrix $\mathbf{R}_g \in \mathbb{Z}^{2m\times k}$ such that $(\mathbf{A}|\mathbf{B}_g)\cdot \mathbf{R}_g = \mathbf{D}$.
  2. For all pairs $(\gamma, \hat{\gamma})$, compute $\mathbf{w}_{\gamma,\hat{\gamma}} = \mathbf{c}_2 - \mathbf{R}_g^T\mathbf{c}_{\gamma,\hat{\gamma}} \in \mathbb{Z}_q^k$.
  3. If there exists a pair $(\gamma, \hat{\gamma})$ such that $\left\lfloor \frac{2}{q}\cdot \mathbf{w}_{\gamma,\hat{\gamma}} \right\rceil = \mathsf{encode}(\mu')$ for some $\mu' \in \{0, 1\}$, output μ'. Otherwise, output $\perp$.

Remark 4.1. We remark that the above construction is single-hop, as the further re-encryption procedure cannot establish a new revocation mechanism on the original re-encrypted ciphertext. Indeed, the re-encryption procedure needs to establish a complete binary tree BT which associates the re-encryption key with the ciphertext and the ciphertext contains information that can only be disclosed by non-revoked users, so further re-encryption of any re-encrypted ciphertext implies that we should establish a new complete binary tree BT' to associate the re-encryption key with the original re-encrypted ciphertext. Since the original re-encrypted ciphertext contains a private vector $\mathbf{s} \in \mathbb{Z}_q^n$ and information of the non-revoked node in BT and the re-encryption key, a new revocation mechanism cannot be linked to the original re-encrypted ciphertext.

4.1 Parameters and correctness

Parameters. We set the parameters to meet the correctness and security requirements as follows: $\lambda = n$, $N, \ell, d = \mathrm{poly}(n)$, $k = \omega(\log n)$, $m = 2n\log q$, and $q/B > 4(m+1)^{3d+11/2}$; to apply Lemma 2.8 (item 3) in the security proof, we set $s = \omega((m+1)^{d+1})$; we set $\sigma_1 = \omega((m+1)^{d+1}\cdot\sqrt{\log(2m)})$ to satisfy the requirement of **SamplePre** algorithm; and to apply Lemma 2.9 in the security proof, we set $\sigma_2 = \omega((m+1)^{d+3/2}\cdot\sqrt{\log(3m)})$.

Correctness. Given an honestly generated ciphertext $\mathbf{c} = (\mathbf{c}_0, \mathbf{c}_1, \mathbf{c}_2)$ of message $\mu \in \{0,1\}$, with respect to some attribute $\mathbf{x} \in \{0,1\}^\ell$ and $\mathsf{RL} \subseteq [N]$, where $\mathbf{c}_0 = \{\mathbf{c}_{in}, \mathbf{c}_0^1, \ldots, \mathbf{c}_0^\ell\}$, and $\mathbf{c}_1 = \{\hat{\mathbf{c}}_{\hat{y}}\}_{\hat{y}\in\mathsf{KUNodes(BT,RL)}}$. Then, we consider the following two cases.

- Let $sk_f := \mathbf{T}_{(\mathbf{A}|\mathbf{B}_f)}$ be a correctly generated secret key. When $f(\mathbf{x}) = 0$, we have $\mathbf{c}_f = \mathbf{B}_f^T\mathbf{s} + \mathbf{e}_f$ by $\mathbf{c}_f = \mathbf{Eval}_{ct}(f, \{(\mathbf{B}_i, \mathbf{x}_i, \mathbf{c}_0^i)\}_{i\in[\ell]})$ (see Lemma 2.11), where $\|\mathbf{e}_f\| \le 20Bm\cdot(m+1)^d$. Consequently,

$$(\mathbf{c}_{in}|\mathbf{c}_f) = (\mathbf{A}|\mathbf{B}_f)^T\mathbf{s} + \mathbf{e}'_f, \text{ where} \|\mathbf{e}'_f\| \le 20Bm\cdot(m+1)^d + B\sqrt{m}.$$

  Since $(\mathbf{A}|\mathbf{B}_f)\cdot\mathbf{R}_f = \mathbf{D}$ where $\|\mathbf{R}_f\|_2 \le \sigma_1\cdot\sqrt{2mk}$ with overwhelming probability. Therefore, we have

$$\mathbf{c}_2 - \mathbf{R}_f^T\cdot(\mathbf{c}_{in}|\mathbf{c}_f) = (\mathbf{D}^T\mathbf{s} + \mathbf{e}_1 + \lfloor q/2\rfloor\cdot\mathsf{encode}(\mu)) - (\mathbf{D}^T\mathbf{s} + \mathbf{R}_f^T\mathbf{e}'_f) = \mathbf{e}_1 - \mathbf{R}_f^T\mathbf{e}'_f + \lfloor q/2\rfloor\cdot\mathsf{encode}(\mu),$$

  where $\|\mathbf{e}_1 - \mathbf{R}_f^T\mathbf{e}'_f\| \le B\sqrt{m} + \sqrt{2mk}\sigma_1\cdot(20Bm(m+1)^d + B\sqrt{m}) \le B\cdot(m+1)^{2d+3} < q/4$ with overwhelming probability, which thereby ensures correct decryption of $\mu \in \{0,1\}$.
- Let $sk_g := \mathbf{T}_{(\mathbf{A}|\mathbf{B}_g)}$ be a correctly generated secret key. Given a re-encryption key $rk_{f\to(g,I)} := (f, g, I, \{\mathbf{R}_\gamma\}_{\gamma\in\mathsf{Path}(I)})$ and an updated state ST such that $(\mathbf{A}|\mathbf{B}_f|\mathbf{U}_\gamma)\cdot\mathbf{R}_\gamma = (\mathbf{A}|\mathbf{B}_g)$ for each node $\gamma \in \mathsf{Path}(I)$, where $\|\mathbf{R}_\gamma\|_2 \le \sqrt{6}m\sigma_2$ with overwhelming probability, and a re-encrypted ciphertext $\mathbf{c}_{f\to(g,I)} = (\tilde{\mathbf{c}}, \mathbf{c}_2)$, where $\tilde{\mathbf{c}} = \{\mathbf{c}_{\gamma,\hat{\gamma}}\}$ and $\mathbf{c}_{\gamma,\hat{\gamma}} = \mathbf{R}_\gamma^T\cdot(\mathbf{c}_{in}|\mathbf{c}_f|\hat{\mathbf{c}}_{\hat{\gamma}})$ for all $\gamma \in \mathsf{Path}(I)$, $\hat{\gamma} \in \mathsf{KUNodes(BT,RL)}$. Again, when $f(\mathbf{x}) = 0$, we have $\mathbf{c}_f = \mathbf{B}_f^T\mathbf{s} + \mathbf{e}_f$ by the correctness of algorithm $\mathbf{Eval}_{ct}$ of Lemma 2.11, where $\|\mathbf{e}_f\| \le 20Bm\cdot(m+1)^d$. Since RL is kept hidden, we cannot check whether it holds that $I \notin \mathsf{RL}$ directly. Therefore, we consider two cases:
  1. When $I \notin \mathsf{RL}$, there exists $(\gamma, \hat{\gamma})$ for $\gamma \in \mathsf{Path}(I)$, $\hat{\gamma} \in \mathsf{KUNodes(BT,RL)}$ corresponding to the same node in BT which satisfies

$$\left(\mathbf{A}|\mathbf{B}_f|\mathbf{U}_{\hat{\gamma}}\right)\cdot\mathbf{R}_\gamma = (\mathbf{A}|\mathbf{B}_g).$$

     Therefore, for such a pair $(\gamma, \hat{\gamma})$, we have

$$\mathbf{c}_{\gamma,\hat{\gamma}} = \mathbf{R}_\gamma^T\cdot\left(\mathbf{c}_{in}|\mathbf{c}_f|\hat{\mathbf{c}}_{\hat{\gamma}}\right) = \mathbf{R}_\gamma^T\cdot\left(\mathbf{A}|\mathbf{B}_f|\mathbf{U}_{\hat{\gamma}}\right)^T s + \mathbf{R}_\gamma^T\cdot\left(\mathbf{e}_0|\mathbf{e}_f|\mathbf{S}_{\hat{\gamma}}^T\mathbf{e}_0\right) = (\mathbf{A}|\mathbf{B}_g)^T s + \mathbf{R}_\gamma^T\cdot\left(\mathbf{e}_0|\mathbf{e}_f|\mathbf{S}_{\hat{\gamma}}^T\mathbf{e}_0\right), \quad (1)$$

     where $\|\mathbf{R}_\gamma^T\cdot(\mathbf{e}_0|\mathbf{e}_f|\mathbf{S}_{\hat{\gamma}}^T\mathbf{e}_0)\| \le \sqrt{6}m\sigma_2\cdot(B\sqrt{m} + 20Bm\cdot(m+1)^d + 20Bm)$. Then, since $(\mathbf{A}|\mathbf{B}_g)\cdot\mathbf{R}_g = \mathbf{D}$ where $\|\mathbf{R}_g\|_2 \le \sigma_1\cdot\sqrt{2mk}$ with overwhelming probability, we have

$$\begin{aligned}\mathbf{w}_{\gamma,\hat{\gamma}} &= \mathbf{c}_2 - \mathbf{R}_g^T\mathbf{c}_{\gamma,\hat{\gamma}}\\ &= \mathbf{c}_2 - \mathbf{R}_g^T\cdot\left((\mathbf{A}|\mathbf{B}_g)^T s + \mathbf{R}_\gamma^T\cdot\left(\mathbf{e}_0|\mathbf{e}_f|\mathbf{S}_{\hat{\gamma}}^T\mathbf{e}_0\right)\right)\\ &= \mathbf{D}^T\mathbf{s} + \mathbf{e}_1 + \lfloor q/2\rfloor\cdot\mathsf{encode}(\mu) - \mathbf{D}^T\mathbf{s} - \mathbf{R}_g^T\mathbf{R}_\gamma^T\cdot\left(\mathbf{e}_0|\mathbf{e}_f|\mathbf{S}_{\hat{\gamma}}^T\mathbf{e}_0\right)\\ &= \lfloor q/2\rfloor\cdot\mathsf{encode}(\mu) + \mathbf{e}_1 - \mathbf{R}_g^T\mathbf{R}_\gamma^T\cdot\left(\mathbf{e}_0|\mathbf{e}_f|\mathbf{S}_{\hat{\gamma}}^T\mathbf{e}_0\right),\end{aligned}$$

     where $\|\mathbf{e}_1 - \mathbf{R}_g^T\mathbf{R}_\gamma^T\cdot(\mathbf{e}_0|\mathbf{e}_f|\mathbf{S}_{\hat{\gamma}}^T\mathbf{e}_0)\| \le B\sqrt{k} + 2\sqrt{3k}m^{3/2}\sigma_1\sigma_2\cdot(B\sqrt{m} + 20Bm\cdot(m+1)^d + 20Bm)$ with

$$\le B\cdot(m+1)^{3d+11/2} < q/4$$

     overwhelming probability, which thereby ensures correct decryption of $\mu \in \{0,1\}$.

  2. When $I \in \mathsf{RL}$, there does not exist such $(\gamma, \hat{\gamma})$ for $\gamma \in \mathsf{Path}(I)$, $\hat{\gamma} \in \mathsf{KUNodes(BT,RL)}$ corresponding to the same node in BT which satisfies

$$\left(\mathbf{A}|\mathbf{B}_f|\mathbf{U}_{\hat{\gamma}}\right)\cdot\mathbf{R}_\gamma = (\mathbf{A}|\mathbf{B}_g).$$

     In other words, the re-decryption algorithm taking as input the secret key sk_g cannot obtain the above equation (1). This implies that $\mathbf{w}_{\gamma,\hat{\gamma}}$ for each pair $(\gamma, \hat{\gamma})$ is indistinguishable from uniform due to the security of our scheme (which we will show in the next section). Therefore, the probability that the last $k-1$ coordinates of $\left\lfloor\frac{2}{q}\cdot\mathbf{w}_{\gamma,\hat{\gamma}}\right\rceil$ are all 0 is at most $2^{-(k-1)} = 2^{-\omega(\log\lambda)}$, which is negligible in λ. Therefore, the re-decryption algorithm outputs $\perp$ with all but negligible probability.

Therefore, for any $d = \text{poly}(n)$, we have $2^{n^\varepsilon} > 4 \cdot (m+1)^{3d+11/2}$ by setting $n = \tilde{O}(d)^{1/\varepsilon}$ for $0 < \varepsilon < 1$, and hence we have to rely on sub-exponential LWE with $q = B \cdot 2^{n^\varepsilon}$, which is at least as hard as SIV P_γ and GapSV P_γ for $\gamma = 2^{\Omega(n^\varepsilon)}$ by Lemma 2.5.

4.2 Security proof

We show that our revocable KP-ABPRE scheme is selectively CPA secure in the standard model.

Theorem 4.2. *Given the three algorithms* $(\mathbf{Eval}_{pk}, \mathbf{Eval}_{ct}, \mathbf{Eval}_{sim})$ *for* $\mathcal{F}$, *the revocable KP-ABPRE scheme above is selectively CPA secure in the standard model as defined in* Definition 3.2, *assuming the hardness of the* $\mathrm{LWE}_{n,m,q,\chi}$ *problem.*

Proof. In this proof, we adopt a game-based approach, where a number of sequential games are evaluated. The first game is the real security game as defined in Definition 3.2, and $\mathcal{A}$ has advantage zero in the last game. The LWE problem will be used to show the indistinguishability between Games 2 and 3. In the following, we build the games to prove that $\mathcal{A}$ wins the selective security game with negligible advantage.

- **Game 0**. This is the real selective security game between the challenger and the adversary $\mathcal{A}$.
- **Game 1**. This is the same as Game 0 except that we change how the public matrices $\mathbf{B}_i$, $\mathbf{U}_\gamma$ for each $i \in [\ell]$ and each $\gamma \in \mathsf{BT}$ are generated. In this game, upon receiving the challenge attribute $\mathbf{x}^* \in \{0,1\}^\ell$ and revocation list $\mathsf{RL}^* \subseteq [N]$, the challenger does:
  1. Choose uniformly at random ℓ matrices $\mathbf{S}_1^*, \ldots, \mathbf{S}_\ell^* \in \{-1,1\}^{m \times m}$ and set $\mathbf{B}_i = \mathbf{A}\mathbf{S}_i^* - \mathbf{x}_i^* \mathbf{G}$ for $i \in [\ell]$.
  2. Build a complete binary tree BT, choose uniformly at random $\mathbf{S}_\gamma \in \{-1,1\}^{m \times m}$ for each $\gamma \in \mathsf{BT}$, and set the identifier

$$\mathbf{U}_\gamma = \begin{cases} \mathbf{A}\mathbf{S}_\gamma, & \text{if } \gamma \in \mathsf{KUNodes}(\mathsf{BT}, \mathsf{RL}^*), \\ \mathbf{A}\mathbf{S}_\gamma - \mathbf{G}, & \text{otherwise.} \end{cases}$$

  In addition, at the challenge query, the challenger computes $\mathbf{S}_i^{*T}\mathbf{e}_0$ for $i \in [\ell]$ to generate the challenge ciphertext $\mathbf{c}_0^*$ and $\mathbf{S}_\gamma^T \mathbf{e}_0$ for $\gamma \in \mathsf{BT}$ to generate the challenge ciphertext $\mathbf{c}_1^*$, for some $\mathbf{e}_0 \in \chi^m$.
- **Game 2**. This is the same as Game 1 except that we change how $\mathbf{A}$ is generated. In this game, the challenger samples a uniformly random matrix $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$. The challenger has no trapdoor of $\Lambda_q^\perp(\mathbf{A})$, but she can answer all $O_{\mathbf{KeyGen}}$, $O_{\mathbf{ReKeyGen}}$, and $O_{\mathbf{ReEnc}}$, as follows.

  Note that (by Definition 3.2) only functions satisfying $f(\mathbf{x}^*) = 1$ are allowed for $O_{\mathbf{KeyGen}}$. To produce a secret key for such functions f, the challenger does:
- Compute $\mathbf{B}_f = \mathbf{Eval}_{pk}(f, (\mathbf{B}_1, \ldots, \mathbf{B}_\ell))$.
- Run $\mathbf{S}_f^* \leftarrow \mathbf{Eval}_{sim}(f, \{(\mathbf{S}_i^*, \mathbf{x}_i^*)\}_{i \in [\ell]}, \mathbf{A})$ (see Lemma 2.11) such that $\mathbf{A}\mathbf{S}_f^* - f(\mathbf{x}^*)\mathbf{G} = \mathbf{B}_f$. By definition of $\mathbf{Eval}_{sim}$, we have $\|\mathbf{S}_f^*\|_2 \le 20\sqrt{m} \cdot (m+1)^d$.
- Generate a secret key $sk_f := \mathbf{T}_{(\mathbf{A}|\mathbf{B}_f)} \leftarrow \mathbf{SampleBasisRight}(\mathbf{A}, \mathbf{G}, \mathbf{S}_f^*, \mathbf{T}_\mathbf{G}, s)$. By definition of **SampleBasisRight** of Lemma 2.8, item 3, we have that $\mathbf{T}_{(\mathbf{A}|\mathbf{B}_f)}$ is distributed as required. Indeed, since $\|\mathbf{S}_f^*\|_2 \le 20\sqrt{m} \cdot (m+1)^d$, we have that $s \ge \sqrt{5} \cdot (\|\mathbf{S}_f^*\|_2 + 1) \cdot \omega(\sqrt{\log m})$ as needed for **SampleBasisRight**.

Moreover, note that (by Definition 3.2) only pairs $(f, g, I) \in \mathcal{F} \times \mathcal{F} \times [N]$ satisfying $f(\mathbf{x}^*) = 1$ or $(f(\mathbf{x}^*) = 0, I \in \mathsf{RL}^*)$ are allowed for $O_{\mathbf{ReKeyGen}}$. To generate a re-encryption key for such (f, g, I), the challenger takes as input $(\{\mathbf{S}_i^*\}_{i \in [\ell]}, \{\mathbf{S}_\gamma\}_{\gamma \in \mathsf{BT}})$, a state ST, functions $f, g \in \mathcal{F}$, an index $I \in [N]$, the challenge attribute $\mathbf{x}^* \in \{0,1\}^\ell$ and revocation list $\mathsf{RL}^* \subseteq [N]$, and returns $\perp$ if $I \in \mathsf{ST}$. Otherwise, the challenger outputs the updated state $\mathsf{ST} \leftarrow \mathsf{ST} \cup \{I\}$ and computes a re-encryption key $rk_{f \to (g,I)} := (f, g, I, \{\mathbf{R}_\gamma\}_{\gamma \in \mathsf{Path}(I)})$, as follows:

- **Case 1**: When $f(\mathbf{x}^*) = 1$. In this case, from the secret key generation procedure above, we have $\mathbf{B}_f = \mathbf{AS}_f^* - f(\mathbf{x}^*)\mathbf{G} = \mathbf{AS}_f^* - \mathbf{G}$, where $\mathbf{B}_f = \mathbf{Eval}_{pk}(f, (\mathbf{B}_1, \ldots, \mathbf{B}_\ell))$ and $\mathbf{S}_f^* = \mathbf{Eval}_{sim}(f, \{(\mathbf{S}_i^*, \mathbf{x}_i^*)\}_{i\in[\ell]}, \mathbf{A})$ with $\|\mathbf{S}_f^*\|_2 \le 20\sqrt{m}\cdot(m+1)^d$. Whenever $I \in \mathsf{RL}^*$ or $I \notin \mathsf{RL}^*$, sample $\mathbf{R}_\gamma \in \mathbb{Z}_q^{2m\times 3m} \leftarrow \mathbf{ExtSampleRight}(\mathbf{A}, \mathbf{G}, \mathbf{S}_f^* + \mathbf{S}_\gamma, \mathbf{T}_\mathbf{G}, (\mathbf{A}|\mathbf{B}_g), \sigma_2)$ such that $(\mathbf{A}|\mathbf{B}_f|\mathbf{U}_\gamma)\cdot\mathbf{R}_\gamma = (\mathbf{A}|\mathbf{B}_g)$ for each $\gamma \in \mathsf{Path}(I)$, where $\sigma \ge \sqrt{5}\cdot(\|\mathbf{S}_f^* + \mathbf{S}_\gamma\|_2 + 1)\cdot\omega(\sqrt{\log 3m})$ by Lemma 2.9, as required. Thus, the challenger obtains a re-encryption key $rk_{f\to(g,I)} \coloneqq (f, g, I, \{\mathbf{R}_\gamma\}_{\gamma\in\mathsf{Path}(I)})$.
- **Case 2**: When $(f(\mathbf{x}^*) = 0, I \in \mathsf{RL}^*)$. In this case, we have $\mathbf{B}_f = \mathbf{AS}_f^* - f(\mathbf{x}^*)\mathbf{G} = \mathbf{AS}_f^*$, so we do not have a trapdoor for lattice $\Lambda_q^\perp(\mathbf{G})$. Instead, since $I \in \mathsf{RL}^*$, which implies that $\mathsf{Path}(I) \cap \mathsf{KUNodes}(\mathsf{BT}, \mathsf{RL}^*) = \varnothing$ and hence we have $\mathbf{U}_\gamma = \mathbf{AS}_\gamma - \mathbf{G}$ for each $\gamma \in \mathsf{Path}(I)$. Then, similar to Case 1, the challenger samples $\mathbf{R}_\gamma \in \mathbb{Z}_q^{2m\times 3m} \leftarrow \mathbf{ExtSampleRight}(\mathbf{A}, \mathbf{G}, \mathbf{S}_f^* + \mathbf{S}_\gamma, \mathbf{T}_\mathbf{G}, (\mathbf{A}|\mathbf{B}_g), \sigma_2)$ such that $(\mathbf{A}|\mathbf{B}_f|\mathbf{U}_\gamma)\cdot\mathbf{R}_\gamma = (\mathbf{A}|\mathbf{B}_g)$ for each $\gamma \in \mathsf{Path}(I)$. Hence, the challenger obtains a re-encryption key $rk_{f\to(g,I)} \coloneqq (f, g, I, \{\mathbf{R}_\gamma\}_{\gamma\in\mathsf{Path}(I)})$.

With the above ability of generating sk_f for functions f that satisfies $f(\mathbf{x}^*) = 1$ and the ability of generating $rk_{f\to(g,I)}$ for pairs $(f, g, I) \in \mathcal{F}\times\mathcal{F}\times[N]$ that satisfies $f(\mathbf{x}^*) = 1$ or $(f(\mathbf{x}^*) = 0, I \in \mathsf{RL}^*)$, the challenger can answer all queries raised by the adversary $\mathcal{A}$ as follows:

- **Key generation query** $O_{\mathbf{KeyGen}}$: $\mathcal{A}$ sends a function $f \in \mathcal{F}$ to the challenger, the challenger generates a secret key sk_f for the function f as described above and returns it to $\mathcal{A}$.
- **Re-encryption key generation query** $O_{\mathbf{ReKeyGen}}$: $\mathcal{A}$ sends a pair $(f, g, I) \in \mathcal{F}\times\mathcal{F}\times[N]$ to the challenger. The challenger generates a re-encryption key $rk_{f\to(g,I)}$ for the pair (f, g, I) as described above and returns it to $\mathcal{A}$.
- **Re-encryption query** O_{ReEnc}: $\mathcal{A}$ sends $((f, g, I), \mathbf{c}', \mathbf{x})$ to the challenger where $f(\mathbf{x}) = 0$, subject to the condition: $(\mathbf{x} \ne \mathbf{x}^*, f(\mathbf{x}^*) = 1)$ or $(f(\mathbf{x}^*) = 0, I \in \mathsf{RL}^*)$, the challenger computes a re-encryption key $rk_{f\to(g,I)}$ as in $O_{\mathbf{ReKeyGen}}$ and returns a re-encrypted ciphertext $\mathbf{c}'_{f\to(g,I)}$ by running $\mathbf{c}'_{f\to(g,I)} \leftarrow \mathbf{ReEnc}(rk_{f\to(g,I)}, \mathbf{c}', x)$.
- **Game 3**. This is the same as Game 2 except that we choose a uniformly random vector $\mathbf{c}^*$ from $\mathbb{Z}_q^{(\ell+2)m}$ as the challenge ciphertext. In this case, since the challenge ciphertext $\mathbf{c}^*$ is independent of the bit $b \in \{0, 1\}$, $\mathcal{A}$'s advantage is zero. □

To prove Theorem 4.2, we will first prove the following lemmas, which show the statistical indistinguishability or computational indistinguishability under the LWE assumption between any two consecutive games.

Lemma 4.3. Game 0 *is statistically indistinguishable from* Game 1 *in the view of* $\mathcal{A}$.

Proof. Recall that in Game 0, the public matrices $\mathbf{B}_i$, $\mathbf{U}_\gamma$ for all $i \in [\ell]$, $\gamma \in \mathsf{BT}$ are uniformly random matrices in $\mathbb{Z}_q^{n\times m}$, whereas in Game 1, for each $i \in [\ell]$, $\gamma \in \mathsf{BT}$, we have $\mathbf{B}_i = \mathbf{AS}_i^* - \mathbf{x}_i^*\mathbf{G}$, $\mathbf{U}_\gamma = \mathbf{AS}_\gamma - \rho_\gamma\mathbf{G}$, where $\mathbf{S}_i^*, \mathbf{S}_\gamma \in \{-1, 1\}^{m\times m}$ and $\rho_\gamma \in \{0, 1\}$. By Lemma 2.3, given uniformly random matrices $\{\mathbf{B}_i\}_{i\in[\ell]}$ and $\{\mathbf{U}_\gamma\}_{\gamma\in\mathsf{BT}}$ in $\mathbb{Z}_q^{n\times m}$, for each $i \in [\ell]$ and each $\gamma \in \mathsf{BT}$, the distribution of $(\mathbf{A}, \mathbf{AS}_i^* - \mathbf{x}_i^*\mathbf{G}, \mathbf{S}_i^{*T}\mathbf{e}_0)$ is statistically indistinguishable from the distribution of $(\mathbf{A}, \mathbf{B}_i, \mathbf{S}_i^{*T}\mathbf{e}_0)$ and the distribution of $(\mathbf{A}, \mathbf{AS}_\gamma + \rho_\gamma\mathbf{G}, \mathbf{S}_\gamma^T\mathbf{e}_0)$ is statistically indistinguishable from the distribution of $(\mathbf{A}, \{\mathbf{U}_\gamma\}_{\gamma\in\mathsf{BT}}, \mathbf{S}_\gamma^T\mathbf{e}_0)$. Hence, the public parameters $(\mathbf{A}, \mathbf{B}_1, \ldots, \mathbf{B}_\ell, \mathsf{BT})$ in Games 0 and 1 are statistically indistinguishable. In other words, Game 0 is statistically indistinguishable from Game 1. □

Lemma 4.4. Game 1 *is statistically indistinguishable from* Game 2 *in the view of* $\mathcal{A}$.

Proof. Recall that in Game 1, the matrix $\mathbf{A}$ is generated via $\mathbf{GenTrap}(1^n, m, q)$, whereas in Game 2, it is chosen uniformly at random from $\mathbb{Z}_q^{n\times m}$. By Lemma 2.8 (item 1), the matrix $\mathbf{A}$ in Games 1 and 2 are statistically indistinguishable. Since there is a trapdoor $\mathbf{T}_\mathbf{A}$ for lattice $\Lambda_q^\perp(\mathbf{A})$ in Game 1, so the challenger generates the matrix $\mathbf{T}_{(\mathbf{A}|\mathbf{B}_f)}$ as the secret key $sk_f = \mathbf{T}_{(\mathbf{A}|\mathbf{B}_f)}$ by running **SampleBasisLeft** algorithm, and generates the

matrices $\{\mathbf{R}_\gamma\}_{\gamma\in\mathsf{Path}(I)}$ to obtain the re-encryption key $rk_{f\to(g,I)} = (f, g, I, \{\mathbf{R}_\gamma\}_{\gamma\in\mathsf{Path}(I)})$ by running **SampleLeft** algorithm. However, in Game 2, the challenger does not have such a trapdoor, but instead she used a publicly known trapdoor $\mathbf{T_G}$ for lattice $\Lambda_q^\perp(\mathbf{G})$ to generate these private matrices by running **SampleBasisRight** and **ExtSampleRight** algorithms. The properties of these sampling algorithms (see Lemmas 2.8 and 2.9) guarantee that the distributions of these private matrices in Games 1 and 2 are statistically indistinguishable.

In summary, the public parameters and answers to all queries in Game 1 are statistically indistinguishable from those in Game 2, so we conclude that Game 1 is statistically indistinguishable from Game 2. □

Lemma 4.5. Game 2 *is computationally indistinguishable from* Game 3 *in the view of* $\mathcal{A}$, *assuming the hardness of the* $\mathrm{LWE}_{n,m,q,\chi}$ *problem.*

Proof. To prove this, we build an algorithm $\mathcal{B}$ to solve the LWE problem if $\mathcal{A}$ can distinguish Games 2 and 3 with non-negligible advantage.

LWE instance. $\mathcal{B}$ obtains an LWE instance: $(\mathbf{A}, \mathbf{D}) \in \mathbb{Z}_q^{n\times m} \times \mathbb{Z}_q^{n\times k}$ and $(\mathbf{w}_0, \mathbf{w}_1) \in \mathbb{Z}_q^m \times \mathbb{Z}_q^k$. We have that $(\mathbf{w}_0, \mathbf{w}_1) \in \mathbb{Z}_q^m \times \mathbb{Z}_q^k$ are either random or

$$\mathbf{w}_0 = \mathbf{A}^T\mathbf{s} + \mathbf{e}_0 \quad \text{and} \quad \mathbf{w}_1 = \mathbf{D}^T\mathbf{s} + \mathbf{e}_1 \tag{2}$$

for some random vector $\mathbf{s} \in \mathbb{Z}_q^n$ and $\mathbf{e}_0 \leftarrow \chi^m$, $\mathbf{e}_1 \leftarrow \chi^k$.

Public parameters. $\mathcal{B}$ sets pk as in Game 2, that is, sample uniformly at random matrices $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$, $\mathbf{D} \in \mathbb{Z}_q^{n\times k}$ and generate public matrices $\mathbf{B}_i = \mathbf{A}\mathbf{S}_i^* - \mathbf{x}_i^*\mathbf{G}$, $\mathbf{U}_\gamma = \mathbf{A}\mathbf{S}_\gamma - \rho_\gamma\mathbf{G}$ for each $i \in [\ell]$, $\gamma \in \mathsf{BT}$, where $\mathbf{S}_i^*, \mathbf{S}_\gamma \in \{-1, 1\}^{m\times m}$ and $\rho_\gamma \in \{0, 1\}$.

Query 1. $\mathcal{B}$ answers $\mathcal{A}$'s all queries ($O_{\mathbf{KeyGen}}$, $O_{\mathbf{ReKeyGen}}$, and $O_{\mathbf{ReEnc}}$) as in Game 2.

Challenge ciphertext. Upon receiving $\mu_0, \mu_1 \in \{0, 1\}$, $\mathcal{B}$ samples a random bit $b \leftarrow \{0, 1\}$ and creates the challenge ciphertext $\mathbf{c}^* = (\mathbf{c}_0^*, \mathbf{c}_1^*, \mathbf{c}_2^*)$ by letting

$$\begin{aligned}
\mathbf{c}_0^* &= (\mathbf{I}_m|\mathbf{S}_1^*|\cdots|\mathbf{S}_\ell^*)^T\mathbf{w}_0 \in \mathbb{Z}_q^{(\ell+1)m},\\
\mathbf{c}_1^* &= \left\{\hat{\mathbf{c}}_{\hat{\gamma}}\right\}_{\hat{\gamma}\in\mathsf{KUNodes}(\mathsf{BT},\mathsf{RL}^*)}, \quad \text{where } \hat{\mathbf{c}}_{\hat{\gamma}} = \mathbf{S}_{\hat{\gamma}}^T\mathbf{w}_0 \in \mathbb{Z}_q^m,\\
\mathbf{c}_2^* &= \mathbf{w}_1 + \lfloor q/2\rfloor\cdot \mathsf{encode}(\mu_b) \in \mathbb{Z}_q^k.
\end{aligned} \tag{3}$$

Then, $\mathcal{B}$ returns $\mathbf{c}^*$ to $\mathcal{A}$. Next, we show two cases: the first is the case where LWE instance is pseudorandom (i.e., equation (2) holds), and the second is where the LWE challenge is random.

1. We show that $\mathbf{c}^*$ is distributed as in Game 2 if the LWE instance is pseudorandom. First, we have $\mathbf{c}_0^* = \mathbf{A}^T\mathbf{s} + \mathbf{e}_0 \in \mathbb{Z}_q^m$, which is distributed exactly as in Game 2. Letting

$$\mathbf{H} = (\mathbf{A}|\mathbf{B}_1 + \mathbf{x}_1^*\mathbf{G}|\cdots|\mathbf{B}_\ell + \mathbf{x}_\ell^*\mathbf{G}) = (\mathbf{A}|\mathbf{A}\mathbf{S}_1^* - \mathbf{x}_1^*\mathbf{G} + \mathbf{x}_1^*\mathbf{G}|\cdots|\mathbf{A}\mathbf{S}_\ell^* - \mathbf{x}_\ell^*\mathbf{G} + \mathbf{x}_\ell^*\mathbf{G}) = (\mathbf{A}|\mathbf{A}\mathbf{S}_1^*|\cdots|\mathbf{A}\mathbf{S}_\ell^*)$$

then, $\mathbf{c}_0^*$ given in equation (3) satisfies:

$$\begin{aligned}
\mathbf{c}_1^* &= (\mathbf{I}_m|\mathbf{S}_1^*|\cdots|\mathbf{S}_\ell^*)^T\mathbf{w}_0\\
&= (\mathbf{I}_m|\mathbf{S}_1^*|\cdots|\mathbf{S}_\ell^*)^T\cdot(\mathbf{A}^T\mathbf{s} + \mathbf{e}_0)\\
&= (\mathbf{A}|\mathbf{A}\mathbf{S}_1^*|\cdots|\mathbf{A}\mathbf{S}_\ell^*)^T\cdot\mathbf{s} + (\mathbf{I}_m|\mathbf{S}_1^*|\cdots|\mathbf{S}_\ell^*)^T\cdot\mathbf{e}_0\\
&= \mathbf{H}^T\mathbf{s} + \mathbf{e} \in \mathbb{Z}_q^{\ell m},
\end{aligned}$$

where $(\mathbf{I}_m|\mathbf{S}_1^*|\cdots|\mathbf{S}_\ell^*)^T\cdot\mathbf{e}_0 \in \mathbb{Z}_q^{(\ell+1)m}$, and hence we conclude that $\mathbf{c}_1^*$ is distributed exactly as in Game 2. Moreover, for each $\hat{\gamma} \in \mathsf{KUNodes}(\mathsf{BT}, \mathsf{RL}^*)$ we have

$$\hat{\mathbf{c}}_{\hat{\gamma}} = \mathbf{S}_{\hat{\gamma}}^T\mathbf{w}_0 = \mathbf{S}_{\hat{\gamma}}^T\cdot(\mathbf{A}^T\mathbf{s} + \mathbf{e}_0) = \mathbf{U}_{\hat{\gamma}}^T\mathbf{s} + \mathbf{U}_{\hat{\gamma}}^T\mathbf{e}_0 \in \mathbb{Z}_q^m,$$

and hence for $\mathbf{c}_2^* = \{\hat{\mathbf{c}}_{\hat{y}}\}_{\hat{y}\in \mathsf{KUNodes}(\mathsf{BT},\mathsf{RL}^*)}$ we conclude that $\mathbf{c}_2^*$ is distributed exactly as in Game 2. Finally, we have that $\mathbf{c}_3 = \mathbf{D}^T\mathbf{s} + \mathbf{e}_1 + \lfloor q/2 \rfloor \cdot \mathsf{encode}(\mu) \in \mathbb{Z}_q^k$ which is distributed exactly as in Game 2. In summary, we conclude that $\mathbf{c}^*$ is distributed as in Game 2.

2. We show that $\mathbf{c}^*$ is distributed as in Game 3 if the LWE instance is random. First, since $(\mathbf{w}_0, \mathbf{w}_1) \in \mathbb{Z}_q^m \times \mathbb{Z}_q^k$ are uniformly random matrices, $\mathbf{c}_2^* = \mathbf{w}_1 + \lfloor q/2 \rfloor \cdot \mathsf{encode}(\mu_b)$ is uniform random over $\mathbb{Z}_q^k$, and is therefore distributed exactly as in Game 3. Moreover, by applying the leftover hash lemma [37], we conclude that $\mathbf{c}_0^*$ and $\hat{\mathbf{c}}_{\hat{y}}$ for each $\hat{y} \in \mathsf{KUNodes}(\mathsf{BT}, \mathsf{RL}^*)$ defined in equation (3) are uniform random over $\mathbb{Z}_q^{(\ell+1)m}$ and $\mathbb{Z}_q^m$, respectively. Therefore, we conclude that $\mathbf{c}^*$ is distributed as in Game 3.

Query 2. The same as Query 1.

Output. $\mathcal{A}$ gives a guess as to whether it interacts with Game 2 or with Game 3, and $\mathcal{B}$ outputs $\mathcal{A}$'s guess.

As stated above, $\mathcal{A}$'s view is as in Game 2 if the LWE instance is pseudorandom, and $\mathcal{A}$'s view is as in Game 3 if the LWE instance is random. Therefore, $\mathcal{B}$'s advantage in solving LWE problem is identical to $\mathcal{A}$'s advantage in distinguishing Games 2 and 3. Under the $\mathrm{LWE}_{n,m,q,\chi}$ assumption, we conclude that Game 2 is computationally indistinguishable from Game 3. □

Overall, since $\mathcal{A}$'s advantage is zero in Game 3, the theorem holds. This completes the proof.

5 Conclusion

We introduced the notion of revocable KP-ABPRE, which supports an efficient revocation mechanism while maintaining the functionality of KP-ABPRE. We instantiated this notion from lattices by proposing a lattice-based revocable KP-ABPRE scheme. Our scheme is the first revocable KP-ABPRE scheme that supports polynomial-depth Boolean circuits and has short private keys that are solely dependent on the depth of the supported policy circuits. In addition, our scheme would yield the first lattice-based KP-ABPRE scheme by letting the revocation list be an empty set. However, our scheme is single-hop and can only be proven CPA secure in a selective manner. Therefore, one of the possible immediate extensions to this work is to construct a multi-hop lattice-based revocable KP-ABPRE construction. Another possible extension is to investigate how we can provide adaptively CPA security in the standard model from LWE with a polynomial-time reduction.

Conflict of interest: Authors state no conflict of interest.

References

[1] Agrawal S, Boneh D, Boyen X. Efficient lattice (H)IBE in the standard model. In: Advances in Cryptology - EUROCRYPT 2010, Proceedings of the 29th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Monaco/French Riviera, May 30–June 3, 2010; 2010. p. 553–72.

[2] Agrawal S, Freeman DM, Vaikuntanathan V. Functional encryption for inner product predicates from learning with errors. In: Advances in Cryptology - ASIACRYPT 2011, Proceedings of the 17th International Conference on the Theory and Application of Cryptology and Information Security, Seoul, South Korea, December 4–8, 2011; 2011. p. 21–40.

[3] Ateniese G, Fu K, Green M, Hohenberger S. Improved proxy re-encryption schemes with applications to secure distributed storage. In: Proceedings of the Network and Distributed System Security Symposium, NDSS 2005, San Diego, California, USA; 2005.

[4] Attrapadung N, Imai H. Attribute-based encryption supporting direct/indirect revocation modes. In: Parker MG, editor. Cryptography and Coding, Cryptography and Coding, Proceedings of the 12th IMA International Conference, Cryptography

and Coding 2009, Cirencester, UK, December 15–17, 2009. Lecture Notes in Computer Science, vol. 5921. Berlin, German: Springer; 2009. p. 278–300.
[5] Bendlin R, Damgård I. Threshold decryption and zero-knowledge proofs for lattice-based cryptosystems. In: Micciancio D, editor. Theory of Cryptography, Proceedings of the 7th Theory of Cryptography Conference, TCC 2010, Zurich, Switzerland, February 9–11, 2010. Lecture Notes in Computer Science, vol 5978. Berlin, German: Springer; 2010. p. 201–18.
[6] Blaze M, Bleumer G, Strauss M. Divertible protocols and atomic proxy cryptography. In: Nyberg K, editor. Advances in Cryptology - EUROCRYPT '98, Proceedings of the International Conference on the Theory and Application of Cryptographic Techniques, Espoo, Finland, May 31–June 4, 1998. Lecture Notes in Computer Science, vol. 1403. Berlin, German: Springer; 1998. p. 127–44.
[7] Boldyreva A, Goyal V, Kumar V. Identity-based encryption with efficient revocation. In: Ning P, Syverson PF, Jha S, editors. Proceedings of the 2008 ACM Conference on Computer and Communications Security, CCS 2008, Alexandria, Virginia, USA, October 27–31, 2008. New York, NY: ACM; 2008. p. 417–26.
[8] Boneh D, Gentry C, Gorbunov S, Halevi S, Nikolaenko V, Segev G, et al. Fully key-homomorphic encryption, arithmetic circuit ABE and compact garbled circuits. In: Nguyen PQ, Oswald E, editors. Advances in Cryptology – EUROCRYPT 2014, Proceedings of the 33rd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Copenhagen, Denmark, May 11–15, 2014. Lecture Notes in Computer Science, vol. 8441. Berlin, German: Springer; 2014. p. 533–56.
[9] Boneh D, Kim S, Montgomery HW. Private puncturable prfs from standard lattice assumptions. In: Advances in Cryptology – EUROCRYPT 2017, Proceedings of the 36th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Paris, France, April 30–May 4, 2017, Part I; 2017. p. 415–45.
[10] Brakerski Z, Vaikuntanathan V. Constrained key-homomorphic prfs from standard lattice assumptions – or: How to secretly embed a circuit in your PRF. In: Theory of Cryptography, Proceedings of the 12th Theory of Cryptography Conference, TCC 2015, Warsaw, Poland, March 23–25, 2015, Part II; 2015. p. 1–30.
[11] Brakerski Z, Vaikuntanathan V. Circuit-abe from LWE: unbounded attributes and semi-adaptive security. In: Robshaw M, Katz J, editors. Advances in Cryptology, Proceedings of the CRYPTO 2016 - 36th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2016, Part III. Lecture Notes in Computer Science, vol. 9816. Berlin, German: Springer; 2016. p. 363–84.
[12] Canard S, Devigne J. Highly privacy-protecting data sharing in a tree structure. Future Gener Comput Syst. 2016;62:119–27.
[13] Cash D, Hofheinz D, Kiltz E, Peikert C. Bonsai trees, or how to delegate a lattice basis. In: Gilbert H, editor. Advances in Cryptology – EUROCRYPT 2010, Proceedings of the 29th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Monaco/French Riviera, May 30–June 3, 2010. Lecture Notes in Computer Science, vol. 6110. Berlin, German: Springer; 2010; p. 523–52.
[14] Chen J, Lim HW, Ling S, Wang H, Nguyen K. Revocable identity-based encryption from lattices. In: Susilo W, Mu Y, Seberry J, editors Information Security and Privacy – Proceedings of the 17th Australasian Conference, ACISP 2012, Wollongong, NSW, Australia, July 9–11, 2012. Lecture Notes in Computer Science, vol. 7372. Berlin, German: Springer; 2012. p. 390–403.
[15] Chu C, Weng J, Chow SSM, Zhou J, Deng RH. Conditional proxy broadcast re-encryption. In: Information Security and Privacy, Proceedings of the 14th Australasian Conference, ACISP 2009, Brisbane, Australia, July 1–3, 2009; 2009. p. 327–42.
[16] Chunpeng Ge, Liu Z, Xia J, Liming F. Revocable identity-based broadcast proxy re-encryption for data sharing in clouds. IEEE Trans Dependable Secure Comput. 2019;20(3):618–30.
[17] Ge C, Susilo W, Fang L, Wang J, Shi Y. A cca-secure key-policy attribute-based proxy re-encryption in the adaptive corruption model for dropbox data sharing system. Design Code Cryptogr. 2018;86(11):2587–603.
[18] Ge C, Susilo W, Wang J, Huang Z, Fang L, Ren Y. A key-policy attribute-based proxy re-encryption without random oracles. Comput J. 2016;59(7):970–82.
[19] Gentry C, Peikert C, Vaikuntanathan V. 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; 2008. p. 197–206.
[20] Gorbunov S, Vaikuntanathan V, Wee H. Predicate encryption for circuits from LWE. In: Advances in Cryptology - CRYPTO 2015, Proceedings of the 35th Annual Cryptology Conference, Santa Barbara, CA, USA, August 16–20, 2015, Part II; 2015. p. 503–23.
[21] Katsumata S, Matsuda T, Takayasu A. Lattice-based revocable (hierarchical) IBE with decryption key exposure resistance. In: Lin D, Sako K, editors. Public-Key Cryptography - PKC 2019, Proceedings of the 22nd IACR International Conference on Practice and Theory of Public-Key Cryptography, Beijing, China, April 14-17, 2019, Part II, Lecture Notes in Computer Science, vol. 11443. Berlin, German: Springer; 2019. p. 441–71
[22] Kim S, Wu DJ. Watermarking prfs from lattices: Stronger security via extractable prfs. In: Advances in Cryptology - CRYPTO 2019, Proceedings of the 39th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 18–22, 2019, Part III; 2019. p. 335–66.

[23] Lee K, Park S. Revocable hierarchical identity-based encryption with shorter private keys and update keys. Des Codes Cryptogr. 2018;86(10):2407–40.
[24] Li J, Ma C, Zhang K. A novel lattice-based CP-ABPRE scheme for cloud sharing. Symmetry. 2019;11(10):1262.
[25] Li K, Zhang Y, Ma H. Key policy attribute-based proxy re-encryption with matrix access structure. In: 2013 5th International Conference on Intelligent Networking and Collaborative Systems, Xi'an city, Shaanxi province, China, September 9–11, 2013, Piscataway, NJ: IEEE; 2013. p. 46–50.
[26] Liang X, Cao Z, Lin H, Shao J. Attribute based proxy re-encryption with delegating capabilities. In: Li W, Susilo W, Tupakula UK, Safavi-Naini R, Varadharajan V, editors. Proceedings of the 2009 ACM Symposium on Information, Computer and Communications Security, ASIACCS 2009, Sydney, Australia, March 10–12, 2009. New York, NY: ACM; 2009. p. 276–86.
[27] Ling S, Nguyen K, Wang H, Zhang J. Revocable predicate encryption from lattices. In: Okamoto T, Yu Y, Au MH, Li Y, editors. Provable Security, Proceedings of the 11th International Conference, ProvSec 2017, Xi'an, China, October 23–25, 2017. Lecture Notes in Computer Science, vol. 10592. Berlin, German: Springer; 2017. p. 305–26.
[28] Ling S, Nguyen K, Wang H, Zhang J. Server-aided revocable predicate encryption: Formalization and lattice-based instantiation. Comput J. 2019;62(12):1849–62.
[29] Meng F. Directly revocable ciphertext-policy attribute-based encryption from lattices. IACR Cryptol. ePrint Arch. 2020;940:1–23.
[30] Micciancio D, Peikert C. Trapdoors for lattices: Simpler, tighter, faster, smaller. In: Advances in Cryptology - EUROCRYPT 2012, Proceedings of the 31st Annual International Conference on the Theory and Applications of Cryptographic Techniques, Cambridge, UK, April 15–19, 2012; 2012. p. 700–18.
[31] Micciancio D, Regev O. Worst-case to average-case reductions based on gaussian measures. In: Proceedingsof the 45th Symposium on Foundations of Computer Science (FOCS 2004), 17–19 October 2004, Rome, Italy. Los Alamitos, CA: IEEE Computer Society; 2004. p. 372–81.
[32] Naor D, Naor M, Lotspiech J. Revocation and tracing schemes for stateless receivers. In: Kilian J, editor. Advances in Cryptology - CRYPTO 2001, Proceedings of the 21st Annual International Cryptology Conference, Santa Barbara, California, USA, August 19–23, 2001. Lecture Notes in Computer Science, vol. 2139. Berlin, German: Springer; 2001. p. 41–62.
[33] Nieto JMG, Manulis M, Sun D. Fully private revocable predicate encryption. In: Susilo W, Mu Y, Seberry J, editors. Information Security and Privacy, Proceedings of the 17th Australasian Conference, ACISP 2012, Wollongong, NSW, Australia, July 9–11, 2012. Lecture Notes in Computer Science, vol. 7372. Berlin, German: Springer; 2012. p. 350–63.
[34] Park S, Lee K, Lee DH. New constructions of revocable identity-based encryption from multilinear maps. IEEE Trans Inf Forensics Secur. 2015;10(8):1564–77.
[35] Peikert C. Public-key cryptosystems from the worst-case shortest vector problem: extended abstract. In: Mitzenmacher M, editor. Proceedings of the 41st Annual ACM Symposium on Theory of Computing, STOC 2009, Bethesda, MD, USA, May 31–June 2, 2009. New York, NY: ACM; 2009. p. 333–42.
[36] Polyakov Y, Rohloff K, Sahu G, Vaikuntanathan V. Fast proxy re-encryption for publish/subscribe systems. ACM Trans Priv Secur. 2017;20(4):14:1–14:31.
[37] Shoup V. A computational introduction to number theory and algebra. Cambridge, UK: Cambridge University Press; 2006.
[38] Susilo W, Chen R, Guo F, Yang G, Mu Y, Chow Y. Recipient revocable identity-based broadcast encryption: How to revoke some recipients in IBBE without knowledge of the plaintext. In: Proceedings of the 11th ACM on Asia Conference on Computer and Communications Security, AsiaCCS 2016, Xi'an, China, May 30–June 3, 2016; 2016. p. 201–10.
[39] Takayasu A, Watanabe Y. Lattice-based revocable identity-based encryption with bounded decryption key exposure resistance. In: Pieprzyk J, Suriadi S, editors. Information Security and Privacy, Proceedings of the 22nd Australasian Conference, ACISP 2017, Auckland, New Zealand, July 3–5, 2017, Part I. Lecture Notes in Computer Science, vol. 10342. Berlin, German: Springer; 2017. p. 184–204.
[40] Wang S, Zhang X, Zhang Y. Efficient revocable and grantable attribute-based encryption from lattices with fine-grained access control. IET Inf Secur. 2018;12(2):141–9.
[41] Wang Y. Lattice ciphertext policy attribute-based encryption in the standard model. Int J Netw Secur. 2014;16(6):444–51.
[42] Watanabe Y, Emura K, Seo JH. New revocable IBE in prime-order groups: Adaptively secure, decryption key exposure resistant, and with short public parameters. In: Handschuh H, editor. Topics in Cryptology - CT-RSA 2017, Proceedings of the Cryptographers' Track at the RSA Conference 2017, San Francisco, CA, USA, February 14–17, 2017. Lecture Notes in Computer Science, vol. 10159. Berlin, German: Springer; 2017. p. 432–49.
[43] Weng J, Deng RH, Ding X, Chu C, Lai J. Conditional proxy re-encryption secure against chosen-ciphertext attack. In: Li W, Susilo W, Tupakula UK, Safavi-Naini R, Varadharajan V, editors. Proceedings of the 2009 ACM Symposium on Information, Computer and Communications Security, ASIACCS 2009, Sydney, Australia, March 10–12, 2009. New York, NY: ACM; 2009. p. 322–32.
[44] Yang K, Wu G, Dong C, Fu X, Li F, Wu T. Attribute based encryption with efficient revocation from lattices. Int J Netw Secur. 2020;22(1):161–70.