Hindawi
Security and Communication Networks
Volume 2020, Article ID 1743421, 11 pages
https://doi.org/10.1155/2020/1743421

Research Article

Scalable Revocable Identity-Based Signature Scheme with Signing Key Exposure Resistance from Lattices

Congge Xie, Jian Weng, and Jinming Wen

College of Information Science and Technology, Jinan University, Guangzhou, China

Correspondence should be addressed to Jian Weng; cryptjweng@gmail.com

Received 23 August 2019; Accepted 12 December 2019; Published 14 January 2020

Academic Editor: Bruce M. Kapron

In 2014, a new security definition of a revocable identity-based signature (RIBS) with signing key exposure resistance was introduced. Based on this new definition, many scalable RIBS schemes with signing key exposure resistance were proposed. However, the security of these schemes is based on traditional complexity assumption, which is not secure against attacks in the quantum era. Lattice-based cryptography has many attractive features, and it is believed to be secure against quantum computing attacks. We reviewed existing lattice-based RIBS schemes and found that all these schemes are vulnerable to signing key exposure. Hence, in this paper, we propose the first lattice-based RIBS scheme with signing key exposure resistance by using the left-right lattices and delegation technology. In addition, we employ a complete subtree revocation method to ensure our construction meeting scalability. Finally, we prove that our RIBS scheme is selective-ID existentially unforgeable against chosen message attacks (EUF-sID-CMA) under the standard short integer solutions (SIS) assumption in the random oracle model.

1. Introduction

Nowadays, dynamic management systems are becoming increasingly popular in enterprises and schools. A common operation in these systems is deleting users. Although many works with efficient revocation mechanisms have been proposed for public key cryptography, and only a few studies have been conducted on identity-based cryptography, which uses a user's identity as his/her public key. Boneh and Franklin [1] proposed an approach to revoke users in an identity-based setting, but this results in a heavy computation overhead at the key generation center (KGC). In order to reduce the KGC's overhead, Boldyreva et al. [2] employed a binary tree to propose the first scalable revocable identity-based encryption (RIBE) scheme, in which the overhead of the KGC is logarithmically increased based on the number of users. All subsequent scalable RIBE schemes [3–5] followed Boldyreva et al.'s basic framework and security model. However, in 2013, Seo and Emura [6] found that all existing Boldyreva-based scalable RIBE schemes are vulnerable to decryption key exposure. They then improved the definition and security model of RIBE and proposed the first scalable RIBE with decryption key exposure resistance. Meanwhile, they also introduced a new security definition of an RIBS with signing key exposure resistance and proposed the first scalable RIBS scheme. In 2013, Tsai et al. studied the revocation problem in identity-based signatures and proposed the first revocable identity-based signature (RIBS) scheme from bilinear pairings. All subsequent RIBS schemes [7–9] were constructed based on Tsai's definition and security model. Unfortunately, these RIBS schemes are also vulnerable to signing key exposure attacks. It is worth noting that the security of all abovementioned RIBS schemes are based on the traditional complexity problem, namely, the Diffie–Hellman problem. Hence, these schemes would not be secure against attacks from quantum computers. Lattice-based cryptography has many attractive features, and it is believed to be secure against attacks using quantum computers. In 2015, Xinyin [10] utilized a complete subtree structure to propose the first lattice-based RIBS scheme, which adopts secret channels to send periodic time update keys. Since the update keys are transmitted in a secret way, they can avoid signing key exposure. Recently, Hung et al. [11] employed the rejection sampling technique [12] to propose a new latticed-based RIBS scheme. Their scheme has a better performance in terms of signature size, signing key

size, and the revocation mechanism with public channels. However, the KGC's overhead increases linearly with the number of users, which results in issues with scalability. Moreover, their scheme cannot avoid signing key exposure. As far as we know, there is no lattice-based scalable RIBS scheme with signing key exposure resistance and the update keys are regularly sent over public channels.

1.1. Our Contributions. The main contributions of our paper are summarized as follows:

(1) We study the approach used to achieve signing key exposure resistance in Seo et al.'s RIBS scheme and adopt Agrawal et al.'s left-right lattices and delegation technology to propose the first lattice-based RIBS scheme with signing key resistance

(2) We then set all the parameters to ensure our construction's correctness and prove that our construction is selective-ID existentially unforgeable against chosen message attacks (EUF-sID-CMA) under the standard short integer solutions (SIS) assumption

(3) Our revocation mechanism employs Boldyreva et al.'s binary tree structure to meet scalability needs and ensures that the KGC does not need to send periodic time update keys over secret channels

1.2. Related Work

1.2.1. Revocable Identity-Based Encryption Schemes. In 1984, Shamir [13] first proposed the concept of an identity-based cryptosystem, which uses a user's identity as his/her public key. In 2001, Boneh and Franklin [1] proposed the first practical identity-based encryption (IBE) scheme. They also considered the revocation problem in an identity-based setting and presented an approach to revoke users, but this imposes a huge overhead on the KGC. In 2008, Boldyreva et al. [2] introduced a basic framework and security model for RIBE and proposed the first scalable RIBE scheme, which greatly reduces the KGC's overhead. However, their scheme is only selective-ID secure. Later, Libert and Vergnaud [3] proposed an adaptive-ID secure RIBE scheme. All of the above RIBE schemes are constructed based on parings. In 2012, Chen et al. [4] proposed the first lattice-based scalable RIBE scheme based on Boldyreva's definition and security model. In 2013, Seo and Emura [6] revisited all existing scalable RIBE schemes and found that all of them are vulnerable to decryption key exposure. They then introduced a new security definition for RIBE and proposed the first scalable RIBE scheme with decryption key exposure resistance. Since then, several RIBE schemes with decryption key exposure resistance [14–17] have been proposed that are built on bilinear groups. In 2017, Takayasu and Watanabe [18] proposed the first lattice-based RIBE scheme with bounded decryption key exposure resistance.

1.2.2. Revocable Identity-Based Signature Schemes. In 2013, Tsai et al. [7] proposed the first RIBS scheme based on bilinear pairings, which is EUF-CMA in a random oracle. Also relying on bilinear pairings, Hung et al. [9] proposed a strongly EUF-CMA secure RIBS scheme without a random oracle. Since the pairing computation is considered a rather expensive operation, Sun et al. [8] proposed an RIBS scheme without pairings. Aforementioned all RIBS schemes are not scalable or satisfying signing key exposure resistance. In 2013, Seo and Emura [6] introduced a new security definition of revocable identity-based signature (RIBS) with signing key exposure resistance and proposed the first scalable RIBS scheme based on the Paterson–Schuldt IBS. In order to improve the efficiency of key revocation, Liu et al. [19] proposed a strongly unforgeable RIBS scheme in the standard model, which can efficiently revoke users and resist key exposure attacks. Subsequently, Wei et al. [20] proposed an RIBS scheme with forward secure. Yang et al. [21] improved Hung et al.'s RIBS scheme [9] and also constructed an RIBS scheme in the standard model with strong unforgeability and signing key exposure resistance. And Zhao et al. [22] also proposed a communication-efficient RIBS scheme from multilinear maps. However, all of them cannot resist quantum attacks. In 2015, Xinyin [10] proposed an RIBS scheme based on lattices, but the revocable functionality channel is secret. Then, Hung et al. [11] proposed an RIBS scheme based on NTRU lattices, in which the KGC sends regular update keys in public channels. One limitation of their scheme is that it is vulnerable to decryption key exposure. Therefore, no lattice-based scalable RIBS schemes with signing key exposure resistance have been proposed so far.

The rest of the paper is organized as follows: in Section 2, we present some important preliminaries. In Section 3, we present the definition and security model of the RIBS. In Section 4, based on Agrawal et al. 's left-right lattices and delegation technology, we construct a concrete scalable RIBS scheme and prove that our construction meets the EUF-sID-CMA security standards. In Section 5, we give some comparisons with previous works in terms of functionalities. Section 6 concludes the paper.

2. Preliminaries

2.1. Notation. $\mathbb{Z}_q$ is a ring of integers modulo q and represents integers in $(-q/2, q/2]$, where $q \geq 2$. $\mathbb{Z}_q^{n\times m}$ is a set of $n \times m$ matrices, in which the elements are in $\mathbb{Z}_q$. $\|\mathbf{x}\|$ denotes the Euclidean norm of a vector $\mathbf{x}$. Matrix $\mathbf{A}^T$ is the transpose of $\mathbf{A}$, and $\|\mathbf{A}\|$ represents the norm of a matrix $\mathbf{A}$ which is defined as the largest norm of its columns. Given matrix $\mathbf{B}$, $\widetilde{B}$ and $\|\widetilde{B}\|$ are $\mathbf{B}$'s Gram–Schmidt orthogonalization and Gram–Schmidt norm.

2.2. Lattices

Definition 1. Given prime q, $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$, and $\mathbf{u} \in \mathbb{Z}_q^n$, define following lattices:

$$\begin{aligned}\Lambda_q^{\perp}(\mathbf{A}) &= \{\mathbf{y} \in \mathbb{Z}^m \colon \mathbf{A}\mathbf{y} = 0 \bmod q\},\\ \Lambda_q^{\mathbf{u}}(\mathbf{A}) &= \{\mathbf{y} \in \mathbb{Z}^m \colon \mathbf{A}\mathbf{y} = \mathbf{u} \bmod q\}.\end{aligned} \tag{1}$$

Let $\rho_{\sigma,\mathbf{c}}(\mathbf{x}) \coloneqq \exp(-\pi\|\mathbf{x}-\mathbf{c}\|^2/\sigma^2)$ be the standard Gaussian function with center $\mathbf{c}$ and variance σ. For a lattice Λ and all the Λ's points $\mathbf{x}$, define the discrete Gaussian distribution over Λ as $\mathcal{D}_{\Lambda,\sigma,\mathbf{c}}(\mathbf{x}) \coloneqq \rho_{\sigma,\mathbf{c}}(\mathbf{x})/\rho_{\sigma,\mathbf{c}}(\Lambda)$. For ease of notation, we use $\mathcal{D}_{\Lambda,\sigma}$ as $\mathcal{D}_{\Lambda,\sigma,0}$'s abbreviate.

The following lemma from [23, 24] is very useful for our construction.

Lemma 1. *Given $q \geq 2$ and $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$, $\Lambda_q^{\perp}(\mathbf{A})$'s basis $\mathbf{T}_{\mathbf{A}}$, $s \geq \|\widetilde{\mathbf{T}_A}\|\omega(\sqrt{\log m})$, Then, for $\mathbf{c} \in \mathbb{R}^m$ and $\mathbf{u} \in \mathbb{Z}_q^n$, algorithm SamplePre$(\mathbf{A}, \mathbf{T}_A, \mathbf{u}, s)$ outputs $\mathbf{x} \in \Lambda_q^{\mathbf{u}}(\mathbf{A})$ and its distribution is statistically close to $\mathcal{D}_{\Lambda_q^{\mathbf{u}}(\mathbf{A}),s,\mathbf{c}}$, where $m > n$ and $\Lambda_q^{\mathbf{u}}(\mathbf{A})$ is not empty.*

The following fact from [23] will be used in our security proof.

Lemma 2. *Given n, positive prime q, $m \geq 2n\log q$, and $s \geq \omega(\sqrt{\log m})$, $\mathbf{u} \coloneqq \mathbf{Ae}$ mod q's distribution is statistically close to $\mathbb{Z}_q^n$'s uniform except with a $2q^{-n}$ fraction of all $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$, where $\mathbf{e}$ is from $\mathcal{D}_{\mathbb{Z}^m,s}$.*

2.3. Some Algorithms. We first recall an algorithm investigated by [24–26], which generates a random lattice $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ and $\Lambda_q^{\perp}(\mathbf{A})$'s short basis $\mathbf{T}_{\mathbf{A}}$.

Lemma 3. *Given $q \geq 2$ and $m \geq 6n\log q$, with overwhelming probability, TrapGen$(1^n, 1^m, q)$ outputs ($\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ and its basis $\mathbf{T}_{\mathbf{A}} \in \mathbb{Z}^{m\times m}$), $\|\widetilde{\mathbf{T}_{\mathbf{A}}}\| \leq O(\sqrt{n\log q})$, where $\mathbf{A}$'s distribution is statistically close to $\mathbb{Z}_q^{n\times m}$'s uniform.*

The following lemma derived from [23] shows that the algorithm can convert any full-rank set of vectors in a lattice to a basis efficiently and meanwhile cannot increase the norm of the Gram–Schmidt vectors.

Lemma 4. *Given a l-dimensional lattice Λ's arbitrary basis $\mathbf{B}$ and a full-rank set $\mathbf{S} \in \Lambda$, there exist a deterministic algorithm that can convert $\mathbf{S}$ to $\mathbf{T}$ and for all $i \in [l]$, $\|\widetilde{t_i}\| \leq \|\widetilde{s_i}\|$, where $\mathbf{T}$ is a basis of Λ.*

Another two algorithms in [27] will be used to construct and prove our following scheme, respectively.

Lemma 5. *Given $q \geq 2$, $m \geq n$, and $s \geq \|\widetilde{\mathbf{T}_{\mathbf{A}}}\| \cdot \omega(\sqrt{\log(m+m_1)})$, algorithm SampleLeft $(\mathbf{A}, \mathbf{B}, \mathbf{T}_{\mathbf{A}}, \mathbf{u}, s)$ outputs a vector $\mathbf{e} \in \mathbb{Z}^{m+m_1}$ distributed statistically close to $\mathcal{D}_{\Lambda_q^{\mathbf{u}}(\mathbf{G}),s}$ with full-rank matrix $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$, $\Lambda_q^{\perp}(\mathbf{A})$'s basis $\mathbf{T}_{\mathbf{A}}$, matrix $\mathbf{B} \in \mathbb{Z}_q^{n\times m_1}$, and vector $\mathbf{u} \in \mathbb{Z}_q^n$, where $\mathbf{G} \coloneqq (\mathbf{A} \mid \mathbf{B})$.*

Lemma 6. *Given $q > 2$, $m > n$, and $s > \|\widetilde{\mathbf{T}_{\mathbf{B}}}\| \cdot l \cdot s_{\mathbf{R}} \cdot \omega(\sqrt{\log m})$, algorithm SampleRight $(\mathbf{A}, \mathbf{B}, \mathbf{R}_1, \mathbf{T}_{\mathbf{B}}, \mathbf{u}, s)$ outputs a vector $\mathbf{e} \in \mathbb{Z}^{m+k}$ distributed statistically close to $\mathcal{D}_{\Lambda_q^{\mathbf{u}}(\mathbf{G}),s}$ with $\mathbf{A} \in \mathbb{Z}_q^{n\times k}$, $\mathbf{B} \in \mathbb{Z}_q^{n\times m}$, $\mathbf{R} \in \mathbb{Z}_q^{k\times m}$, $\Lambda_q^{\perp}(\mathbf{B})$'s basis $\mathbf{T}_{\mathbf{B}}$, vector $\mathbf{u} \in \mathbb{Z}_q^n$, and parameter $s > \|\widetilde{\mathbf{T}_{\mathbf{B}}}\| \cdot s_{\mathbf{R}} \cdot \omega(\sqrt{\log m})$, where $s_{\mathbf{R}} = \sup_{\|y\|=1}\|\mathbf{R}y\|$, $\mathbf{G} = (\mathbf{A} \mid \mathbf{AR} + \mathbf{B})$.*

The following two algorithms from [28] will be also used to construct and prove our following scheme respectively.

Lemma 7. *Given a rank n matrix $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$, a $\mathbb{Z}_q$-invertible matrix $\mathbf{R} \in \mathbb{Z}^{m\times m}$ sampled from $\mathcal{D}_{m\times m}$, a basis $\mathbf{T}_{\mathbf{A}}$ of $\Lambda_q^{\perp}(\mathbf{A})$, and a parameter $s \in \mathbb{R}_{>0}$. Algorithm BasisDel$(\mathbf{A}, \mathbf{R}, \mathbf{T}_{\mathbf{A}}, s)$ outputs a basis $\mathbf{T}_{\mathbf{B}}$ of $\Lambda_q^{\perp}(\mathbf{B})$, where $\mathbf{B} = \mathbf{AR}^{-1}$.*

Lemma 8. *Let $m > 2n\log q$ and $q > 2$ be a prime. For all but at most a q^{-n} fraction of rank n matrix $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$, algorithm SampleRwithBasis$(\mathbf{A})$ outputs a matrix $\mathbf{R} \in \mathbb{Z}^{m\times m}$ sampled from a distribution statistically close to $\mathcal{D}_{m\times m}$. The generated basis $\mathbf{T}_{\mathbf{B}}$ of $\Lambda_q^{\perp}(\mathbf{AR}^{-1})$ satisfies $\|\widetilde{\mathbf{T}_{\mathbf{B}}}\| \leq \sqrt{n\log q}$ with overwhelming probability.*

2.4. Norm of Random Matrix. The following definition from [28] defines a distribution on matrices with low norm and defines an algorithm to sample these matrices.

Definition 2. We say that a matrix $\mathbf{R}$ in $\mathbb{Z}^{m\times m}$ is $\mathbb{Z}_q$-invertible if $\mathbf{R}$ mod q is invertible as a matrix in $\mathbb{Z}_q^{m\times m}$. Define $\alpha_{\mathbf{R}} \coloneqq \sqrt{n\log q} \cdot \omega(\sqrt{\log m})$. We let $\mathcal{D}_{m\times m}$ denote the distribution on matrices in $\mathbb{Z}^{m\times m}$ defined as $(\mathcal{D}_{\mathbb{Z}^m,\alpha_{\mathbf{R}}})^m$ with the condition that $\mathbf{R} \in \mathbb{Z}_q$-invertible and meanwhile $\|\mathbf{R}\| \leq \alpha_{\mathbf{R}} \cdot \sqrt{m}$. Algorithm SampleR$(1^m)$ samples matrices in $\mathbb{Z}^{m\times m}$ and their distributions are statistically close to $\mathcal{D}_{m\times m}$.

2.5. Small Integer Solution Problem

Definition 3 (SIS). For any $n \in \mathbb{Z}$ and any function $m \coloneqq m(n), q \coloneqq q(n)$, and $\beta \coloneqq \beta(n)$, given an integer q, a uniform and random matrix $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ and a real number $\beta \in \mathbb{R}$, the average-case $\mathrm{SIS}_{q,n,m,\beta}$ is the problem of finding a nonzero integer vector $\mathbf{s} \in \mathbb{Z}^m\backslash\{0\}$ satisfying $\mathbf{As} = 0$ mod q and $\|\mathbf{s}\| \leq \beta$.

It has been proved by Micciancio and Regev [29] that for certain parameters, the average-case $\mathrm{SIS}_{q,n,m,\beta}$ problem is as hard as approximating to the worst-case shortest independent vector problem within certain $\gamma = \beta \cdot \widetilde{O}(\sqrt{n})$ factors.

2.6. Encoding Identities or Times as Matrices. In our following scheme, we use an injective encoding function $H\colon \mathbb{Z}_q^n \longrightarrow \mathbb{Z}_q^{n\times n}$ to map identities or times in $\mathbb{Z}_q^n$ to matrices in $\mathbb{Z}_q^{n\times n}$, whose concrete construction can be found in [27].

Definition 4. Given a prime q and a positive integer n, $H\colon \mathbb{Z}_q^n \longrightarrow \mathbb{Z}_q^{n\times n}$ which is an encoding with full-rank differences (FRD) that satisfies the following:

(1) $H(\mathbf{u}) - H(\mathbf{v}) \in \mathbb{Z}_q^{n\times n}$ is full rank for all distinct $\mathbf{u}, \mathbf{v} \in \mathbb{Z}_q^n$

(2) *H is a polynomial computable function in $n\log q$*

3. Definition of RIBS Scheme

In this section, we give the formal definition of the syntax and the security model of RIBS derived in [6]. First, we give the syntax of the RIBS scheme. An RIBS scheme consists of

seven algorithms (Setup, PriKeyGen, KeyUpd, SignKeyGen, Sign, Verify, and KeyRev). Let $\mathcal{U}$, $\mathcal{I}$, and $\mathcal{T}$ be a message space, an identity space, and a time space, respectively.

Definition 5 (syntax of RIBS).

Setup $(1^\lambda, N)$ takes a security parameter λ and a maximal number of users N as input and outputs the public parameters PP which includes the message space $\mathcal{U}$, a master secret key MSK, a revocation list RL (initially empty), and a state ST. This is a stateful algorithm and run by the key authority.

PriKeyGen (PP, MSK, id) takes the public parameters PP, the master secret key MSK, and an identity id $\in \mathcal{I}$ as input and outputs a private key SK_{id} and an updated state ST. This is stateful and also run by the key authority.

KeyUpd (PP, MSK, t, RL, and ST) takes the public parameters PP, the master secret key MSK, a time $t \in \mathcal{T}$, the current revocation list RL, and the state ST as input and outputs a update key KU_t. This is run by the key authority.

SignKeyGen (SK_{id}, KU_t) takes a private key SK_{id} and an update key KU_t as input and outputs a signing key $SK_{id,t}$ or a special symbol $\perp$ indicating that id was revoked. This is a probabilistic algorithm and run by the signer.

Sign (PP, id, t, $SK_{id,t}$, and μ) takes the public parameters PP, an identity id $\in \mathcal{I}$, a time $t \in \mathcal{T}$, a signing key $SK_{id,t}$, and a message $\mu \in \mathcal{U}$ as input and outputs a signature σ.

Verify (PP, id, t, μ, and σ) takes the public parameters PP and an identity id $\in \mathcal{I}$, a time $t \in \mathcal{T}$, and a message/signature pair (μ, σ) as input and outputs 1 (accept) or 0 (reject).

KeyRev (id, t, RL, and ST) takes a revoked identity id $\in \mathcal{I}$, a time $t \in \mathcal{T}$, a revocation list RL, and a state ST as input and outputs an updated revocation list RL. This is stateful and run by the key authority.

3.1. Correctness. The correctness requires that for all $\lambda \in \mathbb{N}$, polynomials N, all (PP, MSK) output by Setup, all $\mu \in \mathcal{U}$, id $\in \mathcal{I}$, $t \in \mathcal{T}$, and all possible valid states ST and revocation lists RL, if identity id is not revoked in time t, then for $(SK_{id}, ST) \longleftarrow$ PriKeyGen (PP, MSK, id), $KU_t \longleftarrow$ KeyUpd (PP, MSK, t, RL, ST), and $SK_{id,t} \longleftarrow$ SignKeyGen (SK_{id}, KU_t), it has Verify $(PP, id, t, \mu, \sigma \longleftarrow$ Sign (PP, id, t, $SK_{id,t}$, $\mu)) = 1$.

Next, we provide a security definition of the RIBS scheme that captures realistic threats including signing key exposure.

Definition 6 (EUF-sID-CMA). The selective-ID essential unforgeable against chosen message attacks (EUF-sID-CMA) security model of the RIBS scheme is defined via the following game between an adversary $\mathcal{A}$ and a challenger $\mathcal{C}$.

Initial: the adversary $\mathcal{A}$ first outputs the challenge identity id^*, time t^*, and also some information state it wants to preserve.

Setup: the challenger runs **Setup** to generate public parameters PP, a master secret key MSK, a revocation list RL (initial empty), and a state ST. It gives PP to the adversary and keeps MSK secret.

Query: A may make some queries as follows:

(i) ***PriKeyGen* Query:** on receiving a private key query, the challenger runs PriKeyGen to return a private key to $\mathcal{A}$
(ii) ***Key Update Query:*** on ***receiving*** an update key query, the challenger runs KeyUpd to return an update key KU_t to $\mathcal{A}$
(iii) ***Revocation Query:*** on receiving a revocation query, the challenger runs KeyRev to return an update list RL to $\mathcal{A}$
(iv) ***SignKeyGen Query:*** on ***receiving*** a signing key query for an identity and a time, the challenger runs PriKeyGen and KeyUpd to return a signing key to $\mathcal{A}$
(v) ***Signature Query:*** on receiving a signature query for a chosen message, an identity, and a time, the challenger runs Sign to return a signature to $\mathcal{A}$

Forgery: at the end of the game, $\mathcal{A}$ outputs a forgery of a message/signature pair on the behalf of the identity id^* in the time period t^*.

If $\mathcal{A}$ satisfies the following four conditions, then it wins the above game:

(i) Verify $(PP, id^*, t^*, \mu^*, \sigma^*) = 1$
(ii) Key Update Query and ***Revocation*** Query can be queried on time which is greater than or equal to the time of all previous queries
(iii) Revocation Query cannot be queried on time t if Key Update Query was queried on t
(iv) If PriKeyGen Query was ***queried*** for identity id^*, then the Revocation Query must be queried for identity id^* on time $t \leq t^*$
(v) SignKeyGen Query cannot be queried on time t before Key Update Query was queried on t
(vi) SignKeyGen Query cannot be queried for identity id^* on time t^*
(vii) Signature Query cannot be ***queried*** for challenge message μ^* under identity id^* on time t^*

The probability of winning the above game is the advantage of $\mathcal{A}$. If there does not exist any PPT adversary who has nonnegligible advantage in the above game, then the RIBS scheme is EUF-sID-CMA secure.

4. Our Scalable RIBS Scheme

In our RIBS scheme, a user's signing key is generated from a private key and periodical update key. The PKG periodically generates update keys and sends them to the nonrevoked users. After obtaining the update keys, the nonrevoked users can generate a valid signing key with their private key. In the

process of constructing our RIBS, it is a challenging technical difficulty that a valid signing key should be guaranteed as a certain lattice space's short basis which is used as a trapdoor to sample vectors. In order to solve this difficulty, we improve the preimage sampling algorithm and redefine it as the SampleSubInv algorithm. Therefore, in the following, we first introduce the SampleSubInv algorithm and then we give the concrete construction of RIBS and provide the process of proving its security. Throughout the section, the function H_0 refers to the FRD map H_0: $\mathbb{Z}_q^n \longrightarrow \mathbb{Z}_q^{n\times n}$ (if $\mathrm{id} \in \{0,1\}^*$, it can be hashed into $\mathbb{Z}_q^n$ by using a collision resistant hash). Hash function H_1 is a random oracle that outputs matrices in $\mathbb{Z}^{3m\times 3m}$, namely, H_1: $\{0,1\}^* \longrightarrow \mathbb{Z}^{3m\times 3m}$: $t \longrightarrow H_1(t) \sim \mathcal{D}_{3m\times 3m}$.

4.1. SampleSubInv Algorithm. **Algorithm SampleSubInv** $(\mathbf{A}, \mathbf{M}, \mathbf{T_A}, \mathbf{U}, \text{and } s)$: the algorithm takes a rank n matrix $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$, a matrix $\mathbf{M} \in \mathbb{Z}^{m\times m}$, a basis $\mathbf{T_A}$ of $\Lambda_q^{\perp}(\mathbf{A})$, a matrix $\mathbf{U} = (\mathbf{u}_1, \ldots, \mathbf{u}_m) \in \mathbb{Z}_q^{n\times m}$, and a parameter $s \in \mathbb{R}_{>0}$ as input and works as follows:

(1) Let $\mathbf{M} = (\mathbf{e}_1, \ldots, \mathbf{e}_m)$, for $i = 1, \ldots, m$ do

(1a) Sample $\mathbf{t}_i \in \mathbb{Z}^m$ as the output of SamplePre $(\mathbf{A}, \mathbf{T_A}, \mathbf{u}_i, s)$, then $\mathbf{A}\mathbf{t}_i = \mathbf{u}_i \bmod q$ and $\mathbf{t}_i$ is sampled from a distribution statistically close to $\mathcal{D}_{\Lambda_q^{\mathbf{u}_i}(\mathbf{A}),s}$

(1b) If $\mathbf{t}_i - \mathbf{e}_i$ is linearly independent of $\{\mathbf{t}_1 - \mathbf{e}_1, \ldots, \mathbf{t}_{i-1} - \mathbf{e}_{i-1}\}$, accept $\mathbf{t}_i$. Otherwise, repeat step (1a)

(2) Let $\mathbf{T} \in \mathbb{Z}^{m\times m}$ be the matrix whose columns are $\mathbf{t}_1, \ldots, \mathbf{t}_m$. Output $\mathbf{T}$

Note that $\mathbf{AT} = \mathbf{U}$, and matrix $\mathbf{T} - \mathbf{M}$ is invertible. The analysis of algorithm SampleSubInv uses Corollary 3.16 of literature [30] which shows that a linearly independent set is produced in step (1a) w.h.p. after m^2 samples from SamplePre $(\mathbf{A}, \mathbf{T_A}, \mathbf{u}_i, s)$. It is not difficult to show that only $2m$ samples are needed in expectation.

4.2. Construction. Our concrete scalable RIBS scheme with signing key resistance works as follows:

Setup $(1^\lambda, N)$: the setup algorithm takes a security parameter λ and a maximal number of users N as input and proceeds as follows:

(1) Pick Gaussian parameters s_1, s_2, and s_3 and set the parameters $n = n(\lambda), q = q(n)$, and $m = m(n)$. These parameters are implicitly known to all of the algorithms below.

(2) Sample $l + 3$ random matrices: $\mathbf{A}_1, \mathbf{A}_2, \mathbf{B}, \{\mathbf{D}_i\}_{i\in[l]}$ in $\mathbb{Z}_q^{n\times m}$, where l is the length of message.

(3) Sample matrix with associated trapdoor: $(\mathbf{A}, \mathbf{T_A}) \longleftarrow \mathrm{TrapGen}(1^n, 1^m, q)$.

(4) Output public parameters $\mathrm{PP} = (\mathbf{A}, \mathbf{A}_1, \mathbf{A}_2, \mathbf{B}, \{\mathbf{D}_i\}_{i\in[l]})$, master secret key $\mathrm{MSK} = \mathbf{T_A}$, a revocation list $\mathrm{RL} = \varnothing$, and a state $\mathrm{ST} = \mathrm{BT}$.

PriKeyGen (PP, MSK, id, and ST): this algorithm takes as input the public parameters PP, a master secret key MSK, an identity $\mathrm{id} \in \{0,1\}^*$, and a state ST. Then it picks an unassigned leaf node ν from BT, stores id in that node, and performs the following steps:

(1) For each node $\theta \in \mathrm{Path}(\nu)$, pick random matrix $\mathbf{U}_\theta \in \mathbb{Z}_q^{n\times m}$ and store it in node θ

(2) Sample $\mathbf{R}_1 \longleftarrow \mathcal{D}_{m\times m}$, $\mathbf{E}_\nu \longleftarrow \mathrm{SamplePre}(\mathbf{A}, \mathbf{T_A}, -(\mathbf{A}_1 + H_0(id)\mathbf{B})\mathbf{R}_1, s_1)$

(3) Sample $\begin{pmatrix}\mathbf{E}_{\theta,1}\\ \mathbf{E}_{\theta,2}\end{pmatrix} \longleftarrow \mathrm{SampleLeft}(\mathbf{A}, \mathbf{A}_1 + \mathbf{H}_0(\mathrm{id})\mathbf{B}, \mathbf{T_A}, \mathbf{U}_\theta, s_1)$

(4) Store $\mathbf{H}_\theta := \mathbf{E}_{\theta,1} - \mathbf{E}_\nu \mathbf{R}_1^{-1}\mathbf{E}_{\theta,2}$ in node θ

(5) Output private key $\mathrm{SK_{id}} = \left(\left\{\theta, \begin{pmatrix}\mathbf{E}_{\theta,1}\\ \mathbf{E}_{\theta,2}\end{pmatrix}\right\}_{\theta\in\mathrm{Path}(\nu)}, \mathbf{E}_\nu, \mathbf{R}_1, \mathbf{ST}\right)$

KeyUpd (PP, MSK, t, RL, and ST): this algorithm takes the public parameters PP, a master secret key MSK, a time $t \in \{0,1\}^*$, a revocation list RL, and a state ST as input and proceeds as follows:

(1) Sample $\mathbf{R}_2 \longleftarrow \mathcal{D}_{m\times m}$ and $\mathbf{E}_t \longleftarrow \mathrm{SamplePre}(\mathbf{A}, \mathbf{T_A}, -(\mathbf{A}_2 + H_0(t)\mathbf{B})\mathbf{R}_2, s_1)$

(2) $\forall\theta \in \mathrm{KUNodes}(\mathrm{BT}, \mathrm{RL}, \text{and } t)$ and sample $\mathbf{E}'_{\theta,2} \longleftarrow (\mathcal{D}_{\mathbb{Z}^m,s_1})^m$ and compute $\mathbf{H}'_\theta = \mathbf{E}_t\mathbf{R}_2^{-1}\mathbf{E}'_{\theta,2}$, where $\mathrm{KUNodes}(\mathrm{BT}, \mathrm{RL}, \text{and } t)$ is an algorithm KUNodes from [4]

(3) Sample $\mathbf{E}'_{\theta,1} \longleftarrow \mathrm{SampleSubInv}(\mathbf{A}, \mathbf{H}'_\theta - \mathbf{H}_\theta, \mathbf{T}_A, -\mathbf{U}_\theta - (\mathbf{A}_2 + H_0(t)\mathbf{B})\mathbf{E}'_{\theta,2}, s_1)$

(4) Output update key $\mathrm{KU}_t = \left(\left\{\theta, \begin{pmatrix}E'_{\theta,1}\\ E'_{\theta,2}\end{pmatrix}\right\}_{\theta\in\mathrm{KUNodes}(\mathrm{BT},\mathrm{RL},t)}, \mathbf{E}_t, \mathbf{R}_2\right)$

SignKeyGen ($\mathrm{SK_{id}}$ and KU_t): this algorithm takes a private key $\mathrm{SK_{id}}$ and an update key KU_t as input and runs the following steps:

(1) Set $\mathbf{A}_{\mathrm{id},t} := (\mathbf{A} \mid \mathbf{A}_1 + H_0(\mathrm{id})\mathbf{B} \mid \mathbf{A}_2 + H_0(t)\mathbf{B})$ and compute $\mathbf{R}_{\mathrm{id},t} = H_1(\mathrm{id} \mid t)$ and $\mathbf{F}_{\mathrm{id},t} := \mathbf{A}_{\mathrm{id},t} \cdot \mathbf{R}_{\mathrm{id},t}^{-1}$

(2) $\forall\left(i, \begin{pmatrix}\mathbf{E}_{i,1}\\ \mathbf{E}_{i,2}\end{pmatrix}\right) \in \mathrm{SK_{id}}$ and $\left(j, \begin{pmatrix}\mathbf{E}'_{j,1}\\ \mathbf{E}'_{j,2}\end{pmatrix}\right) \in \mathrm{KU}_t$, if $\exists i = j$, then set

$$\mathbf{S}_{\mathrm{id},t} = \begin{pmatrix}\mathbf{E}_{i,1} + \mathbf{E}'_{j,1} & \mathbf{E}_\nu & \mathbf{E}_t\\ \mathbf{E}_{i,2} & \mathbf{R}_1 & 0\\ \mathbf{E}'_{j,2} & 0 & \mathbf{R}_2\end{pmatrix}. \tag{2}$$

(3) Generate $\mathrm{SK}'_{\mathrm{id},t}$ by running Lemma 4 which takes $\mathbf{S}_{\mathrm{id},t}$ as input ($\mathbf{S}_{\mathrm{id},t}$ is an invertible matrix with some norm which will be proved in the Theorem 1). Then, run the algorithm $\mathrm{BasisDel}(\mathbf{A}_{\mathrm{id},t}, \mathbf{R}_{\mathrm{id},t}, \mathrm{SK}'_{\mathrm{id},t}, s_2)$ to generate $\mathrm{SK}_{\mathrm{id},t}$ and output $\mathrm{SK}_{\mathrm{id},t}$.

Sign (PP, id, t, $\mathrm{SK}_{\mathrm{id},t}$, and μ): the signing algorithm takes the public parameters PP, an identity id, a time t, a signing key $\mathrm{SK}_{\mathrm{id},t}$, and a message $\mu = (\mu(1), \ldots, \mu(l)) \in \{0,1\}^l$ as input and outputs signature σ as follows:

(1) Sample the algorithm $\sigma \longleftarrow \text{SampleLeft}(\mathbf{F}_{\text{id},t}, \mathbf{B} + \sum_{i=1}^{l}(-1)^{\mu(i)}\mathbf{D}_i, \text{SK}_{\text{id},t}, 0, s_3)$ such that $(\mathbf{F}_{\text{id},t} \mid \mathbf{B} + \sum_{i=1}^{l}(-1)^{\mu(i)}\mathbf{D}_i) \cdot \sigma = 0$, where $s_3 \geq \|\widetilde{\text{SK}_{\text{id},t}}\| \cdot \omega(\sqrt{\log 4m})$

(2) Output signature σ

Verify (PP, id, t, μ, and σ) : the verify algorithm takes the public parameters PP, an identity $\text{id} \in \{0,1\}^*$, a time $t \in \{0,1\}^*$, and a pair of message/signature (μ, σ) as input and accepts only if the following conditions are satisfied:

(i) $\sigma \in \mathbb{Z}^{4m}$, $\|\sigma\| \leq s_3\sqrt{4m}$

(ii) $[\mathbf{F}_{\text{id},t} \mid \mathbf{B} + \sum_{i=1}^{l}(-1)^{\mu(i)}\mathbf{D}_i] \cdot \sigma = 0$

KeyRev (id, t, RL, and ST): this algorithm takes a revoked identity id, a time t, a revocation list RL, and a state ST as input and adds (id, t) to RL for all nodes ν associated with identity id and returns RL.

Theorem 1. $\mathbf{S}_{id,t}$ *is an invertible matrix with small norm.*

Proof. First, we show that $\mathbf{S}_{\text{id},t}$ is a small norm matrix. From the construction, we know that

$$\begin{aligned} \|\mathbf{E}_{i,1}\| \leq s_1 \cdot \sqrt{m}, \|\mathbf{E}'_{j,1}\| \leq s_1 \cdot \sqrt{m}, \\ \|\mathbf{E}_{i,2}\| \leq s_1 \cdot \sqrt{m}, \|\mathbf{E}'_{j,2}\| \leq s_1 \cdot \sqrt{m}, \\ \|\mathbf{E}_{\nu}\| \leq s_1 \cdot \sqrt{m}, \|\mathbf{E}_t\| \leq s_1 \cdot \sqrt{m}, \\ \|\mathbf{R}_1\| \leq \alpha_R \cdot \sqrt{m}, \|\mathbf{R}_2\| \leq \alpha_R \cdot \sqrt{m}, \end{aligned} \tag{3}$$

which are small norm matrices. Then, we can get $||\mathbf{S}_{\text{id},t}|| \leq \max\{4s_1\sqrt{m}, \sqrt{m}(s_1 + \alpha_R)\}$. Second, we will show $\mathbf{S}_{\text{id},t}$ is an invertible matrix. For matrix $\mathbf{S}_{\text{id},t}$, we make an elementary transformation and finally obtain

$$\mathbf{C} \coloneqq \begin{pmatrix} \mathbf{E}_{i,1} + \mathbf{E}'_{j,1} - \mathbf{E}_{\nu}\mathbf{R}_1^{-1}\mathbf{E}_{i,2} - \mathbf{E}_t\mathbf{R}_2^{-1}\mathbf{E}'_{j,2} & 0 & 0 \\ 0 & \mathbf{R}_1 & 0 \\ 0 & 0 & \mathbf{R}_2 \end{pmatrix}. \tag{4}$$

Since $\mathbf{E}_{i,1} + \mathbf{E}'_{j,1} - \mathbf{E}_{\nu}\mathbf{R}_1^{-1}\mathbf{E}_{i,2} - \mathbf{E}_t\mathbf{R}_2^{-1}\mathbf{E}'_{j,2} = \mathbf{E}'_{j,1} - (\mathbf{H}'_j - \mathbf{H}_i)$, and by the SampleSubInv algorithm in the KeyGen phase, we can see $\mathbf{E}'_{j,1} - (\mathbf{H}'_j - \mathbf{H}_i)$ is an invertible matrix. Because $\mathbf{R}_1$ and $\mathbf{R}_2$ are $\mathbb{Z}_q$-invertible matrices, then $\mathbf{C}$ is invertible and then $\mathbf{S}_{\text{id},t}$ is an invertible matrix. In summary, $\mathbf{S}_{\text{id},t}$ is an invertible matrix with small norm.

The correctness equality in Verify follows from the SampleLeft algorithm. In order to ensure the correctness and security of the above construction, we set the following: $n = \lambda$, $m = O(n\log q)$, $\eta = \omega(\sqrt{\log m})$, $s_1 = O(\sqrt{n\log q})\,\omega(\sqrt{\log 2m})$, $s_2 = O(\sqrt{n\log q})\omega(\sqrt{\log n})$, $s_3 = \sqrt{l}\,m^{3/2}\omega(\sqrt{\log m})^4$, $\beta = (\sqrt{l}\,m^2 + \sqrt{3}\,l\eta m^{5/2})2\;\omega(\sqrt{\log m})^4$, $q \geq \max\{2lm^3\omega(\sqrt{\log m})^6, 2Q\}$, where λ is the security parameter.

4.3. Security. The following theorem shows that our scalable RIBS scheme is secure.

Theorem 2. *The construction of the scalable RIBS scheme is EUF-sID-CMA secure provided that the* $SIS_{q,n,m,\beta}$ *assumption holds.*

Proof. Assume $\mathcal{A}$ wins the game in Section 3 and Q is the signing queries maximum number. We can construct a solver $\mathcal{B}$ that provides a valid solution to the SIS problem. $\mathcal{A}$ first gives the target identity and time (id^*, t^*). $\mathcal{B}$ is given a random instance of the $(q, n, m,$ and $\beta)$-SIS problem and is asked to return an admissible solution. That is,

(i) Provided: a matrix $\mathbf{A}_0$ from $\mathbb{Z}_q^{n\times 3m}$'s uniform distribution

(ii) Returned: any vector $\mathbf{e}_0 \in \mathbb{Z}^{3m}$ such that $\mathbf{A}_0\mathbf{e}_0 = 0 \bmod q$ and $0 \neq \|\mathbf{e}_0\| \leq \beta$

$\mathcal{B}$ performs as follows:

Setup* $(1^\lambda, N, l)$: $\mathcal{B}$ provides the adversary $\mathcal{A}$ with simulated parameters as follows:

(1) Run algorithm TrapGen to obtain $(\mathbf{B}, \mathbf{T}_\mathbf{B}) \longleftarrow \text{TrapGen}(1^n, 1^m, q)$

(2) Choose $l+3$ random matrices $\mathbf{U}_0, \mathbf{U}_1, \mathbf{U}_2 \sim \mathcal{D}_{m\times m}$ and $\mathbf{R}_i^* \sim \mathcal{D}_{\mathbb{Z}^m,\eta}$ and also choose l random scalars $h_i \in \mathbb{Z}_q$ for $i \in [l]$. Let $\mathbf{A} = \mathbf{A}_0\begin{pmatrix}\mathbf{U}_0^{-1}\\ \mathbf{U}_1^{-1}\\ \mathbf{U}_2^{-1}\end{pmatrix}$, $\mathbf{A}_1 = \mathbf{A}\mathbf{U}_1 - H_0(\text{id}^*)\mathbf{B} \bmod q$, $\mathbf{A}_2 = \mathbf{A}\mathbf{U}_2 - H(t^*)\mathbf{B} \bmod q$, and $\mathbf{D}_i = \mathbf{A}_0\begin{pmatrix}\mathbf{R}_i^*\\ \mathbf{R}_i^*\\ \mathbf{R}_i^*\end{pmatrix} + h_i \cdot \mathbf{B} \in \mathbb{Z}_q^{n\times m}$

(3) Output the parameters PP = $(\mathbf{A}, \mathbf{A}_1, \mathbf{A}_2, \mathbf{B},$ and $\{\mathbf{D}_i\}_{i\in[l]})$

PriKeyGen Query and Key Update Query: $\mathcal{B}$ first picks a node $\nu^* \in \text{BT}$ and allots it to id^*. It is worth noting that there will be two types of adversaries:

(i) Type (a) adversaries are revoked users and can challenge the target identity id^* before or on time t^*

(ii) Type (b) adversaries cannot challenge the target identity id^* at any time

$\mathcal{B}$ picks a **bit** $\text{rev} \overset{\$}{\longleftarrow} \{0,1\}$ randomly as a guess for the type of adversarial behavior. Clearly, the probability of $\mathcal{B}$'s correct guesses is 1/2. Type (a) adversaries (rev = 0): $\mathcal{B}$ simulates each node θ's matrix $\mathbf{U}_\theta$ as follows:

(i) if $\theta \in \text{Path}(\nu^*)$, sample $\mathbf{E}_{\theta,1}, \mathbf{E}_{\theta,2} \longleftarrow (\mathcal{D}_{\mathbb{Z}^m,s_1})^m$, and $\mathbf{R}_1 \longleftarrow \mathcal{D}_{m\times m}$, set $\mathbf{U}_\theta \longleftarrow [\mathbf{A} \mid \mathbf{A}_1 + H_0(\mathbf{id}^*)\mathbf{B}] \cdot \begin{pmatrix}\mathbf{E}_{\theta,1}\\ \mathbf{E}_{\theta,2}\end{pmatrix}$, $\mathbf{E}_{\nu^*} = -\mathbf{U}_1 \cdot \mathbf{R}_1$. Compute $\mathbf{H}_\theta \coloneqq \mathbf{E}_{\theta,1} - \mathbf{E}_{\nu^*}\mathbf{R}_1^{-1}\mathbf{E}_{\theta,2}$, and store $\mathbf{H}_\theta, \mathbf{U}_\theta$ in the node θ.

(ii) if $\theta \notin \text{Path}(\nu^*)$, sample $\mathbf{E}'_{\theta,1}, \mathbf{E}'_{\theta,2} \longleftarrow (\mathcal{D}_{\mathbb{Z}^m,s_1})^m$, and $\mathbf{R}_2 \longleftarrow \mathcal{D}_{m\times m}$, set $-\mathbf{U}_\theta \longleftarrow [\mathbf{A} \mid \mathbf{A}_2 + H_0(t^*)\mathbf{B}] \cdot \begin{pmatrix}\mathbf{E}'_{\theta,1}\\ \mathbf{E}'_{\theta,2}\end{pmatrix}$, $\mathbf{E}_{t^*} = -\mathbf{U}_2 \cdot \mathbf{R}_2$. Store $\mathbf{U}_\theta$ in the node θ.

Since id^* has been revoked before time t^*, it has $\text{KUNodes}(\text{BT}, t^*) \cap \text{Path}(\nu^*) = \varnothing$ and $\mathcal{B}$ returns id^*'s

private key $\left(\left\{\theta, \begin{pmatrix}\mathbf{E}_{\theta,1}\\ \mathbf{E}_{\theta,2}\end{pmatrix}\right\}_{\theta\in \mathrm{Path}(\nu^*)}, \mathbf{E}_{\nu^*}, \mathbf{R}_1\right)$ and t^*'s update key $\left(\left\{\theta, \begin{pmatrix}\mathbf{E}'_{\theta,1}\\ \mathbf{E}'_{\theta,2}\end{pmatrix}\right\}_{\theta\in \mathrm{KUNodes}(\mathrm{BT},t^*)}, \mathbf{E}_{t^*}, \mathbf{R}_2\right)$.

Type (b) adversaries (rev = 1): $\mathscr{B}$ simulates each node θ's matrix $\mathbf{U}_\theta$ as follows: Since id^* is never queried, $\mathscr{B}$ does not need to simulate id^*'s private key. It samples $\mathbf{E}'_{\theta,1}, \mathbf{E}'_{\theta,2} \longleftarrow (\mathscr{D}_{\mathbb{Z}^m,s_1})^m$ and $\mathbf{R}_2 \longleftarrow \mathscr{D}_{m\times m}$ and sets $-\mathbf{U}_\theta \longleftarrow [\mathbf{A}\,|\,\mathbf{A}_2 + H_0(t^*)\mathbf{B}]\cdot\begin{pmatrix}\mathbf{E}'_{\theta,1}\\ \mathbf{E}'_{\theta,2}\end{pmatrix}$, $\mathbf{E}_{t^*} = -\mathbf{U}_2\cdot\mathbf{R}_2$. It stores $\mathbf{U}_\theta$ in the node θ. If $t = t^*$, $\mathscr{B}$ returns t^*'s update key $\left(\left\{\theta, \begin{pmatrix}\mathbf{E}'_{\theta,1}\\ \mathbf{E}'_{\theta,2}\end{pmatrix}\right\}_{\theta\in \mathrm{KUNodes}(\mathrm{BT},t^*)}, \mathbf{E}_{t^*}, \mathbf{R}_2\right)$.Note that, when rev = 0, for each node θ, $\mathbf{U}_\theta$ is generated either through $\mathbf{U}_\theta \longleftarrow [\mathbf{A}\,|\,\mathbf{A}_1 + H_0(\mathrm{id}^*)\mathbf{B}]\cdot\begin{pmatrix}\mathbf{E}_{\theta,1}\\ \mathbf{E}_{\theta,2}\end{pmatrix}$ or though $-\mathbf{U}_\theta \longleftarrow [\mathbf{A}\,|\,\mathbf{A}_2 + H_0(t^*)\mathbf{B}]\cdot\begin{pmatrix}\mathbf{E}'_{\theta,1}\\ \mathbf{E}'_{\theta,2}\end{pmatrix}$ where $\mathbf{E}_{\theta,1}, \mathbf{E}_{\theta,2} \longleftarrow (\mathscr{D}_{\mathbb{Z}^m,s_1})^m$ and $\mathbf{E}'_{\theta,1}, \mathbf{E}'_{\theta,2} \longleftarrow (\mathscr{D}_{\mathbb{Z}^m,s_1})^m$. Since the distribution of $[\mathbf{A}\,|\,\mathbf{A}_1 + H_0(\mathrm{id}^*)\mathbf{B}] = [\mathbf{A}\,|\,\mathbf{A}\mathbf{U}_1]$ and $[\mathbf{A}\,|\,\mathbf{A}_2 + H_0(t^*)\mathbf{B}] = [\mathbf{A}\,|\,\mathbf{A}\mathbf{U}_2]$ are statistically close to $\mathbb{Z}_q^{n\times 2m}$'s uniform distribution by leftover hash lemma, then all the above $\mathbf{U}_\theta$ are statistically close to uniform over $\mathbb{Z}_q^{n\times m}$; when rev = 1, for each node θ, $\mathbf{U}_\theta$ is generated either through or through uniform distribution in $\mathbb{Z}_q^{n\times m}$, where $\mathbf{E}'_{\theta,1}, \mathbf{E}'_{\theta,2} \longleftarrow (\mathscr{D}_{\mathbb{Z}^m,s_1})^m$. No matter which case, we can have $\mathbf{R}_1 \sim \mathscr{D}_{m\times m}$ and $\mathbf{R}_2 \sim \mathscr{D}_{m\times m}$ and $\mathbf{E}_{\nu^*} = -\mathbf{U}_1\cdot\mathbf{R}_1$ and $\mathbf{E}_{t^*} = -\mathbf{U}_2\cdot\mathbf{R}_2$. Since the distribution of $\mathbf{R}_1$ and $\mathbf{R}_2$ are Gaussian distribution, then the distribution of $\mathbf{E}_{\nu^*}$ and $\mathbf{E}_{t^*}$ is statistically close to Gaussian distribution. In conclusion, the distribution of the simulated system is statistically close to the real system.

For $\mathrm{id} \neq \mathrm{id}^*$'s private key queries and $t \neq t^*$'s update key queries, $\mathscr{B}$ uses the trapdoor $\mathbf{T}_\mathbf{B}$ instead of $\mathbf{T}_\mathbf{A}$. Let $[\mathbf{A}\,|\,\mathbf{A}_1 + H_0(\mathrm{id})\mathbf{B}] = [\mathbf{A}\,|\,\mathbf{A}\mathbf{U}_1 + (H_0(\mathrm{id}) - H_0(\mathrm{id}^*))\mathbf{B}]$ and $[\mathbf{A}\,|\,\mathbf{A}_2 + H_0(t)\mathbf{B}] = [\mathbf{A}\,|\,\mathbf{A}\mathbf{U}_2 + (H_0(t) - H_0(t^*))\mathbf{B}]$. From the definition of FRD, $H_0(\mathrm{id}) - H_0(\mathrm{id}^*)$ and $H_0(t) - H_0(t^*)$ are nonsingular. Then, $\mathscr{B}$ runs the sampleRightExt algorithm and generates $\mathrm{id} \neq \mathrm{id}^*$'s private key

$$\begin{pmatrix}\mathbf{E}_{\theta,1}\\ \mathbf{E}_{\theta,2}\end{pmatrix} \longleftarrow \mathrm{SampleRight}\big(\mathbf{A}, (H_0(\mathrm{id}) - H_0(\mathrm{id}^*))\mathbf{B}, \mathbf{U}_1, \mathbf{T}_\mathbf{B}, \mathbf{U}_\theta, s'_2\big) \in \mathbb{Z}_q^{2m\times m}, \tag{5}$$

where $s'_2 > \|\widetilde{\mathbf{T}_\mathbf{B}}\|\cdot s_{\mathbf{U}_1}\cdot\omega(\sqrt{\log m})$. Moreover, $\mathscr{B}$ samples $\mathbf{R}'_1 \longleftarrow \mathscr{D}_{m\times m}$ and calls algorithm $\mathbf{RandBasis}(\mathbf{T}_\mathbf{B}, s_4)$ from [28]. It can obtain $\mathbf{T}'_\mathbf{B}$ satisfying $[\mathbf{A}\,|\,\mathbf{A}_1 + H_0(\mathrm{id})\mathbf{B}]\cdot\begin{pmatrix}-\mathbf{U}_1\\ I\end{pmatrix}\cdot\mathbf{T}'_\mathbf{B}\mathbf{R}'_1 = 0$ where $s_4 \geq \|\widetilde{\mathbf{T}_\mathbf{B}}\|\cdot\omega(\sqrt{\log n})$, that is, $\mathbf{A}\cdot(-\mathbf{U}_1\mathbf{T}'_\mathbf{B}\mathbf{R}'_1) = -(\mathbf{A}_1 + H_0(\mathrm{id})\mathbf{B})\mathbf{T}'_\mathbf{B}\mathbf{R}'_1$. Then, it sets $\mathbf{E}_\nu = -\mathbf{U}_1\mathbf{T}'_\mathbf{B}\mathbf{R}'_1$ and $\mathbf{R}_1 = \mathbf{T}'_\mathbf{B}\mathbf{R}'_1$ and returns $\left(\left\{\theta, \begin{pmatrix}\mathbf{E}_{\theta,1}\\ \mathbf{E}_{\theta,2}\end{pmatrix}\right\}_{\theta\in \mathrm{Path}(\nu)}, \mathbf{E}_\nu, \mathbf{R}_1\right)$. For all $t \neq t^*$'s update key queries, $\mathscr{B}$ runs

$$\begin{pmatrix}\mathbf{E}'_{\theta,1}\\ \mathbf{E}'_{\theta,2}\end{pmatrix} \longleftarrow \mathrm{SampleRight}\big(\mathbf{A}, (H_0(t) - H_0(t^*))\cdot\mathbf{B}, \mathbf{U}_2, \mathbf{T}_\mathbf{B}, -\mathbf{U}_\theta, s'_2\big) \in \mathbb{Z}_q^{2m\times m}, \tag{6}$$

where $s'_2 > \|\widetilde{\mathbf{T}_\mathbf{B}}\|\cdot s_{\mathbf{U}_2}\cdot\omega(\sqrt{\log m})$. Moreover, it samples $\mathbf{R}'_2 \longleftarrow \mathscr{D}_{m\times m}$ and then calls algorithm $\mathbf{RandBasis}(\mathbf{T}_\mathbf{B}, s_4)$ to get a basis $\mathbf{T}''_\mathbf{B}$ satisfying $[\mathbf{A}\,|\,\mathbf{A}_2 + H_0(t)\mathbf{B}]\cdot\begin{pmatrix}-\mathbf{U}_2\\ I\end{pmatrix}\cdot\mathbf{T}''_\mathbf{B}\mathbf{R}'_2 = 0$, where $s_4 \geq \|\widetilde{\mathbf{T}_\mathbf{B}}\|\cdot\omega(\sqrt{\log n})$, that is, $A\cdot(-\mathbf{U}_2\mathbf{T}'_\mathbf{B}\mathbf{R}'_2) = -(\mathbf{A}_2 + H_0(t)\mathbf{B})\mathbf{T}''_\mathbf{B}\mathbf{R}'_2$. It sets $\mathbf{E}_t = -\mathbf{U}_2\mathbf{T}''_\mathbf{B}\mathbf{R}'_2$ and $\mathbf{R}_2 = \mathbf{T}''_\mathbf{B}\mathbf{R}'_2$ and then generates the update key $\left(\left\{\theta, \begin{pmatrix}\mathbf{E}'_{\theta,1}\\ \mathbf{E}'_{\theta,2}\end{pmatrix}\right\}_{\theta\in \mathrm{KUNodes}(\mathrm{BT},\mathrm{RL},t)}, \mathbf{E}_t, \mathbf{R}_2\right)$.

Since the parameter s'_2 is sufficiently large, $\mathbf{E}_{\theta,1}$'s distribution is statistically close to $\mathscr{D}^{\Lambda_q^\perp}_{\mathbf{U}_\theta-(\mathbf{A}\,|\,\mathbf{A}\mathbf{U}_1)\mathbf{E}_{\theta,2}}$ and $\mathbf{E}'_{\theta,1}$'s distribution is statistically close to $\mathscr{D}^{\Lambda_q^\perp}_{-\mathbf{U}_\theta-(\mathbf{A}\,|\,\mathbf{A}\mathbf{U}_2)\mathbf{E}_{\theta,2}}$. $\mathbf{E}_{\theta,2}$ and $\mathbf{E}'_{\theta,2}$'s distributions are distributed close to $(\mathscr{D}_{\mathbb{Z}^m,s_2})^m$, respectively. Since $\mathbf{T}'_\mathbf{B}$ and $\mathbf{T}''_\mathbf{B}$ are the random bases of $\Lambda_q^\perp(\mathbf{B})$, then $\mathbf{E}_\nu = -\mathbf{U}_1\mathbf{T}'_\mathbf{B}\mathbf{R}'_1$ and $\mathbf{E}_T = -\mathbf{U}_2\mathbf{T}''_\mathbf{B}\mathbf{R}'_2$'s distributions are the same as those in the real system. $\mathbf{R}_1 = \mathbf{T}'_\mathbf{B}\mathbf{R}'_1$ and $\mathbf{R}_2 = \mathbf{T}''_\mathbf{B}\mathbf{R}'_2$ are $\mathbb{Z}_q$-invertible, that is, $\mathbf{R}_1$ and $\mathbf{R}_2$ are distributed as in the real system.

4.3.1. Signing Key Query. $\mathscr{B}$ first samples a random matrix $\mathbf{R}^* \sim \mathscr{D}_{m\times m}$ and defines $H_1(\mathrm{id}^*\,|\,t^*) \coloneqq \mathbf{R}^*$. Then, it sets $\mathbf{G} = \mathbf{A}_0\cdot\begin{pmatrix}\mathbf{R}^* & 0 & 0\\ 0 & \mathbf{R}^* & 0\\ 0 & 0 & \mathbf{R}^*\end{pmatrix}$ and picks an empty list L. If the adversary $\mathscr{A}$ asks the signing key on identity id^* and time $t \neq t^*$, $\mathscr{B}$ runs $\mathrm{SampleRwithBasis}(\mathbf{G})$ to obtain a random $\mathbf{R}_{\mathrm{id}^*,t} \sim \mathscr{D}_{3m\times 3m}$ and a short basis $\mathbf{T}_{\mathbf{F}_{\mathrm{id}^*,t}}$ for $\mathbf{F}_{\mathrm{id}^*,t} = \mathbf{G}\mathbf{R}_{\mathrm{id}^*,t} \bmod q$. Then, it saves the tuple $(\mathrm{id}^*\,|\,t, \mathbf{R}_{\mathrm{id}^*,t}, \mathbf{F}_{\mathrm{id}^*,t}, \mathbf{T}_{\mathbf{F}_{\mathrm{id}^*,t}})$ in list L for future use and returns $\mathbf{T}_{\mathbf{F}_{\mathrm{id}^*,t}}$ to the adversary $\mathscr{A}$. If the adversary $\mathscr{A}$ asks the signing key on identity $\mathrm{id} \neq \mathrm{id}^*$ and time t, $\mathscr{B}$ retrieves the list L and checks whether $i\,d|t$ in L. If $(\mathrm{id}\,|\,t, \mathbf{R}_{\mathrm{id},t}, \mathbf{F}_{\mathrm{id},t}, \mathbf{T}_{\mathbf{F}_{\mathrm{id},t}})$ is in L, $\mathscr{B}$ returns $\mathbf{T}_{\mathbf{F}_{\mathrm{id},t}}$ to $\mathscr{A}$. Otherwise, $\mathscr{B}$ also runs $\mathrm{SampleRwithBasis}(\mathbf{G})$ to obtain a random $\mathbf{R}_{\mathrm{id},t} \sim \mathscr{D}_{3m\times 3m}$ and a short basis $\mathbf{T}_{\mathbf{F}_{\mathrm{id},t}}$ for $\mathbf{F}_{\mathrm{id},t} = \mathbf{G}\mathbf{R}_{\mathrm{id},t} \bmod q$. Then, it saves the tuple $(\mathrm{id}\,|\,t, \mathbf{R}_{\mathrm{id},t}, \mathbf{F}_{\mathrm{id},t}, \mathbf{T}_{\mathbf{F}_{\mathrm{id},t}})$ in list L for future use and returns $\mathbf{T}_{\mathbf{F}_{\mathrm{id},t}}$ to the adversary $\mathscr{A}$. In the real SignKeyGen phase, the signing keys are generated by using algorithm BasisDel. In the Simulated signing key queries, the signing keys are generated from algorithm SampleRwithBasis. By Lemma 7 and 8, the responses to H_1 oracle queries and signing key queries are as in the real system.

4.3.2. Signature Queries. $\mathscr{B}$ simulates signatures of messages μ for $\mathscr{A}$'s adaptive signature queries as follows:

(i) If the queried messages μ are under target identity id^* in target time t^*, $\mathscr{B}$ answers as follows:

(1) Compute $\mathbf{R}_\mu = \sum_{i=1}^{l}(-1)^{\mu(i)}\begin{pmatrix}\mathbf{R}_i^*\\ \mathbf{R}_i^*\\ \mathbf{R}_i^*\end{pmatrix}$ and $h_\mu = 1 + \sum_{i=1}^{l}(-1)^{\mu(i)}h_i$

(2) If $h_\mu = 0 \pmod q$, abort the simulation
(3) Otherwise sample vector:

$$\sigma \longleftarrow \text{SampleRight}\left(\mathbf{A}_0, h_\mu \mathbf{B}, \mathbf{R}_\mu, \mathbf{T}_{\mathbf{B}}, 0, s_2'\right). \quad (7)$$

(4) Output $\sigma \in \mathbb{Z}^{4m}$

(ii) If the queried messages μ are under identity $\text{id} \neq \text{id}^*$ or in time $t \neq t^*$, $\mathcal{B}$ first retrieves the list L and checks whether it has tuple $(\text{id} \mid t, \mathbf{R}_{\text{id},t}, \mathbf{F}_{\text{id},t}, \mathbf{T}_{\mathbf{F}_{\text{id},t}})$. If yes, $\mathcal{B}$ gets $\mathbf{T}_{\mathbf{F}_{\text{id},t}}$ and samples vector $\sigma \longleftarrow \text{SampleLeft}(\mathbf{F}_{\text{id},t}, \mathbf{A}_0\mathbf{R}_\mu + h_\mu\mathbf{B}, \mathbf{T}_{\mathbf{F}_{\text{id},t}}, 0, s_2')$. Otherwise, $\mathcal{B}$ also runs $\text{SampleRwithBasis}(\mathbf{G})$ to obtain a random $\mathbf{R}_{\text{id},t} \sim \mathcal{D}_{3m\times 3m}$ and a short basis $\mathbf{T}_{\mathbf{F}_{\text{id},t}}$ for $\mathbf{F}_{\text{id},t} = \mathbf{G}\mathbf{R}_{\text{id},t} \bmod q$. Then, it saves the tuple $(\text{id} \mid t, \mathbf{R}_{\text{id},t}, \mathbf{F}_{\text{id},t}, \mathbf{T}_{\mathbf{F}_{\text{id},t}})$ in list L and uses $\mathbf{T}_{\mathbf{F}_{\text{id},t}}$ to sample algorithm $\sigma \longleftarrow \text{SampleLeft}(\mathbf{F}_{\text{id},t}, \mathbf{A}_0\mathbf{R}_\mu + h_\mu\mathbf{B}, \mathbf{T}_{\mathbf{F}_{\text{id},t}}, 0, s_2')$. At last, $\mathcal{B}$ returns σ to the adversary $\mathcal{A}$.

Forgery: $\mathcal{B}$ obtains a forged signature σ^* on an unqueried message μ^* and performs the following:

(1) Compute $\mathbf{R}_{\mu^*} = \sum_{i=1}^{l} (-1)^{\mu^*(i)} \begin{pmatrix} \mathbf{R}_i^* \\ \mathbf{R}_i^* \\ \mathbf{R}_i^* \end{pmatrix}$ and $h_{\mu^*} = 1 + \sum_{i=1}^{l} (-1)^{\mu^*(i)} h_i$
(2) If $h_{\mu^*} \neq 0 \pmod q$, abort the simulation
(3) Otherwise, separate $\sigma^* = (\sigma_1^{*T} \mid \sigma_2^{*T})^T$
(4) Output $\mathbf{e}_0 = \sigma_1^* + \mathbf{R}_{\mu^*}\sigma_2^* \in \mathbb{Z}^{3m}$

From Lemma 9, $\mathbf{e}_0$ is small and nonzero with good probability and is a valid (q, n, m, β)-SIS solution with good probability.

Lemma 9. *If $\mathcal{A}$ provides a valid forgery $(\sigma_1^{*T} \mid \sigma_2^{*T})^T$ for message μ^* with $h_{\mu^*} = 0$ (mod q), then for polynomial function $\beta = poly(l, n, m) = poly(\lambda)$, $\mathbf{e}_0 = \sigma_1^* + \mathbf{R}_{\mu^*}\sigma_2^* \in \mathbb{Z}^{3m}$ satisfying $\mathbf{e}_0 \in \Lambda^\perp(\mathbf{A}_0)$ and $0 \neq \|\mathbf{e}_0\| \leq \beta$.*

Proof. First, when $h_{\mu^*} = 0$, we have $\mathbf{A}_0\mathbf{R}_{\mu^*} + h_{\mu^*}\mathbf{B} = \mathbf{A}_0\mathbf{R}_{\mu^*}$. Since σ^* is message μ^*'s valid signature under identity id^* in time t^*, then we can obtain

$$\mathbf{A}_0\left(\sigma_1^* + \mathbf{R}_{\mu^*}\sigma_2^*\right) = \left[\mathbf{A}_0 \mid \mathbf{A}_0\mathbf{R}_{\mu^*}\right]\begin{pmatrix} \sigma_1^* \\ \sigma_2^* \end{pmatrix} = 0 \bmod q. \quad (8)$$

Next, we will show that $\mathbf{e}_0$ is appropriately short. Since $\mathbf{R}_{\mu^*}$ is the sum of l low-norm matrices $\begin{pmatrix} \mathbf{R}_i^* \\ \mathbf{R}_i^* \\ \mathbf{R}_i^* \end{pmatrix}$ and $\mathbf{R}_i^*$ are nearly independent with the same variance $V\{\mathbf{R}_1^*\}$ and their coefficients are in $\{0, 1\}$, then we will have

$$\begin{aligned} V\left\{\mathbf{R}_{\mu^*}\right\} &= V\left\{\sum_{i=1}^{l} \mu^*(i) \begin{pmatrix} \mathbf{R}_i^* \\ \mathbf{R}_i^* \\ \mathbf{R}_i^* \end{pmatrix}\right\} \approx \sum_{i=1}^{l} V\left\{\begin{pmatrix} \mathbf{R}_i^* \\ \mathbf{R}_i^* \\ \mathbf{R}_i^* \end{pmatrix}\right\} \\ &= l \cdot V\left\{\begin{pmatrix} \mathbf{R}_1^* \\ \mathbf{R}_1^* \\ \mathbf{R}_1^* \end{pmatrix}\right\}. \end{aligned} \quad (9)$$

In particular, since they are almost independent discrete Gaussian, $\mathbf{E}\{\mathbf{R}_{\mu^*}\} \approx \mathbf{E}\left\{\begin{pmatrix} \mathbf{R}_1^* \\ \mathbf{R}_1^* \\ \mathbf{R}_1^* \end{pmatrix}\right\} = 0$, $\Pr\left\{\left\|\begin{pmatrix} \mathbf{R}_i^* \\ \mathbf{R}_i^* \\ \mathbf{R}_i^* \end{pmatrix}\right\| > \eta\sqrt{3m}\right\} = \text{negl}(m)$ and $\Pr\{\|\mathbf{R}_{\mu^*}\| > \sqrt{l}\,\eta\sqrt{3m}\} = \text{negl}(m)$. With overwhelming probability $\|\mathbf{e}_0\| \leq \beta$ for $\beta = \text{poly}(l, n, m) = \text{poly}(\lambda)$, we set $\beta = (1 + \eta\sqrt{l}\,\sqrt{3m})\sqrt{4m} \cdot s_3$.

Finally, it remains to show that $\mathbf{e}_0 \neq 0$. Suppose an easy case that $\sigma_2^* = 0$, then for a valid forgery, it must have $\sigma_1^* \neq 0$, thus $\mathbf{e}_0 \neq 0$. On the contrary, $\sigma_2^* \neq 0$, since $\|\sigma_2^*\| < \sqrt{4m} \cdot s_2 \ll q$; thus, there must be at least one σ_2^*'s coordinate that is nonzero modulo q. We set z be σ_4^*'s last one coordinate. Let $\mathbf{r}^*$ be $\mathbf{R}_{\mu^*}$'s last column and $\mathbf{r}_i^*$ be $\mathbf{R}_i^*$'s last column for each $i \in [l]$. Then, we have $\mathbf{r}^* = \sum_{i=1}^{l} (-1)^{\mu^*(i)}\mathbf{r}_i^*$, where the coefficients depend on the message bits. For $\mathbf{r}_1^*$, let $\mathbf{v} = (-1)^{\mu^*(1)} z\mathbf{r}_1^*$, then $\mathbf{e}_0 = z\mathbf{r}^* + \mathbf{e}_0' = \mathbf{v} + \mathbf{e}_0''$, where $\mathbf{v}$ depends on $\mathbf{r}_1^*$ and $\mathbf{e}_0''$ does not. The only information about $\mathbf{r}_1^*$ available to $\mathcal{A}$ is contained in the last column of $\mathbf{D}_1$'s last column. From a simple pigeonhole principle, there exists many admissible and equally likely vector $\mathbf{r}_1^*$ that are the same with $\mathcal{A}$'s view. $\mathcal{A}$ cannot know the value of $z\mathbf{r}_1^*$ with probability exceeding once third, then every other $z\mathbf{r}_1^*$ would fail to do so. Thus, $\Pr\{\mathbf{e}_0 \neq 0\} \geq 1/2^{m(m-1)}$. If $\mathcal{A}$ forges a signature with probability ε with $Q \leq q/2$, then the algorithm $\mathcal{B}$ can solve the $\text{SIS}(n, m, q, \beta)$ problem with the advantage of $1/2^{m(m-1)}(1/2) \cdot \varepsilon(1 - q^{-1}Q)q^{-1} \geq \varepsilon/3q$.

We will show that the simulation (without aborting) is statistically indistinguishable from the real system.

The differences of the algorithms are summarized as follows:

(i) In real Setup, matrix $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ is sampled from the algorithm TrapGen. Matrices $(\mathbf{A}_1, \mathbf{A}_2, \mathbf{B}$, and $\{\mathbf{D}_i\}_{i\in[l]})$ are chosen uniformly at random. In simulated Setup^*, $\mathbf{A} = \mathbf{A}_0\begin{pmatrix} \mathbf{U}_0^{-1} \\ \mathbf{U}_1^{-1} \\ \mathbf{U}_2^{-1} \end{pmatrix}$, $\mathbf{A}_1 = \mathbf{A}\mathbf{U}_1 - H_0(\text{id}^*)\mathbf{B} \bmod q$, $\mathbf{A}_2 = \mathbf{A}\mathbf{U}_2 - H(t^*)\mathbf{B} \bmod q$, $\mathbf{D}_i = \mathbf{A}_0\begin{pmatrix} \mathbf{R}_i^* \\ \mathbf{R}_i^* \\ \mathbf{R}_i^* \end{pmatrix} + h_i \cdot \mathbf{B} \in \mathbb{Z}_q^{n\times m}$ for random $\mathbf{U}_0, \mathbf{U}_1, \mathbf{U}_2 \sim \mathcal{D}_{m\times m}$ and $\mathbf{R}_i^* \sim \mathcal{D}_{\mathbb{Z}^m,\eta}$, and $(\mathbf{B}, \mathbf{T}_{\mathbf{B}} \longleftarrow \text{TrapGen}(q, n))$

(ii) In the real Sign phase, algorithm SampleLeft generates signatures σ, whereas in the simulated Sign phase, signatures are generated by algorithm SampleLeft or algorithm SampleRight

We now argue that the distribution $(\mathbf{A}, \mathbf{A}_1, \mathbf{A}_2, \mathbf{B}, \{\mathbf{D}_i\}_{i\in[l]})$ is statistically indistinguishable in the two experiments. Let

$$\mathbf{A}' = \left[\mathbf{A}_1 \mid \mathbf{A}_2 \mid \mathbf{D}_1 \mid \cdots \mid \mathbf{D}_l\right] \in \mathbb{Z}_q^{n\times(l+2)m}. \quad (10)$$

Then, from Lemma 2, we have

Table 1: Efficiency comparisons with previous works.

	[20]	[21]	[22]	[10]	[11]	Ours
Signing key size	$4\|G\|$	$3\|G\|$	$3\|G_2\|$	$2Nm \cdot (\log r) \cdot \log(\bar{s}\sqrt{2m})$	$4N \cdot \log(s\sqrt{N})$	$s_1\sqrt{m} + s_{\mathbf{R}}\sqrt{m}$
Signature size	$4\|G\|$	$4\|G\|$	$4\|G_2\|$	$3Nm \cdot (\log r) \cdot \log(\bar{s}\sqrt{3m})$	$2N\log\sigma + \lambda\log N + \lambda$	$s_3\sqrt{4m}$
Computation cost of signing	$O(1)e$	$3e$	$6e$	$2NmT_m + T_{\text{sp}}$	$7N^2T_m + 14NT_m$	$O(9m^2n + lmn) \cdot T_+ + O(m^2n)T_m + T_{\text{inv}} + T_{\text{sl}}$
Computation cost of verifying	$O(1)p$	$e + 4p$	$4p$	$3NmT_m$	$N^2T_m + NT_m$	$O(mn)T_m + O(lmn)T_+$
Authority's workload	$O(\log r)$	$O(r)$	$\varnothing$	$O(\log r)$	$O(r)$	$O(\log r)$

N: security parameter; λ: positive integer; r: number of nonrevoked users; $m > 6N\log q$; $\bar{s} = m\log r \cdot \omega(\sqrt{\log N})$; $s = N^{5/2} \cdot \sqrt{2q}\omega(\sqrt{\log N})$; $\sigma = 12\lambda sN$; $s_1 = O(\sqrt{m}) \cdot \omega(\sqrt{\log m})$; $s_3 = \sqrt{m}(s_1 + s_{\mathbf{R}}) \cdot \omega(\sqrt{\log 4m})$; $|G|$ or $|G_2|$ is the size of an element in the group G or G_2; p and e denote a pairing operation and an exponentiation operation, respectively; T_+ is the cost of executing an addition operation in $\mathbb{Z}_q$; T_m is the cost of executing a multiplication operation in $\mathbb{Z}_q$; T_{sp} is the cost of executing the Samplepre algorithm in [23]; T_{inv} is the cost of executing a matrix inversion operation in $\mathbb{Z}^{3m\times 3m}$; T_{sl} is the cost of executing the Sampleleft algorithm in [27]; $\varnothing$ denotes that the workload of the authority performing the update procedure is independent of the number of nonrevoked users.

Table 2: Functionality comparisons with previous works.

Schemes	Key exposure resistance	Quantum attacks resistance	Model	Transport channel	Scalability	Assumption
[20]	Yes	No	SD	Public	Yes	q-DHE
[21]	Yes	No	SD	Public	Yes	CDH
[22]	Yes	No	SD	Public	Yes	$(3,n)$-MDHE
[10]	No	Yes	SD	Secure	No	SIS
[11]	No	Yes	RO	Public	No	RSIS
Ours	Yes	Yes	RO	Public	Yes	SIS

$$(\mathbf{A}, \mathbf{A}')^s \approx \left(\mathbf{A}, \left[\mathbf{A}\mathbf{U}_1 - H_0(id^*)\mathbf{B} \,\middle|\, \mathbf{A}\mathbf{U}_2 - H_0(t^*)\mathbf{B} \,\middle|\, \mathbf{A}\mathbf{R}_1^* + h_1\mathbf{B} \,\middle|\, \cdots \,\middle|\, \mathbf{A}\mathbf{R}_l^* + h_l\mathbf{B}\right]\right), \tag{11}$$

for random matrices $\mathbf{A}, \mathbf{A}'$ from $\mathbb{Z}_q^{n\times m}, \mathbb{Z}_q^{n\times(l+2)m}$ and

$$\left[\mathbf{U}_1 \,\middle|\, \mathbf{U}_2 \,\middle|\, \mathbf{R}_0^* \,\middle|\, \cdots \,\middle|\, \mathbf{R}_l^*\right] \sim \mathcal{D}_{m\times 2m} \times \left(\mathcal{D}_{\mathbb{Z}^m,\eta}\right)^l, \tag{12}$$

chosen at random. Here, $\mathbf{B}$'s distribution is statistically close to $\mathbb{Z}_q^{n\times m}$'s uniform distribution. Therefore, the distribution of PP in the simulation is statistically indistinguishable to them in the real system.

For each μ's signature query, if Gaussian parameters s_3 and s_3' are sufficiently large, then the outputs' distributions of SampleLeft and SampleRight are statistically close.

At last, since the SIS problem is as hard as approximating to the worst-case shortest independent vector problem from [29], the adversary $\mathcal{A}$ cannot succeed in attacking our scalable RIBS scheme.

5. Comparisons

This section will be divided into two parts to evaluate the performance of our RIBS scheme: first efficiency and second functionality.

5.1. Efficiency Comparisons. Here, we emphasis that since our scheme is built upon lattices, which is different from other listed schemes ([20–22]) based on bilinear maps or 3-linear maps, the running times in our scheme and in other listed map-based schemes may be different. Thus, when we discuss the computation cost, we only focus on the number of the corresponding operations and give comparisons with the lattice-based schemes ([10, 11]). First, since in scheme [11], the authors adopted the ideal lattice (NTRU lattice) to generate a user's signing key, it is obvious that both the signing key and signature sizes of their scheme are less than those of both Xiang's RIBS [10] and ours RIBS schemes. Next, for the computation cost of signing, in Xiang's RIBS scheme, it requires $2NmT_m + T_{\text{sp}}$ to obtain a signature, where T_{sp} is the cost of executing the SamplePre algorithm. In scheme [11], it requires $7(N^2 + 2N)T_m$ to obtain a signature, where T_m is the cost of executing a multiplication operation in $\mathbb{Z}_q$. In our RIBS scheme, it needs $O(9m^2n + lmn)T_+ + O(m^2n)T_m + T_{\text{inv}} + T_{\text{sl}}$ level time to obtain a signature, where T_+ is the cost of executing an addition operation in $\mathbb{Z}_q$, T_{inv} is the cost of executing a matrix inversion operation in $\mathbb{Z}^{3m\times 3m}$, and T_{sl} is the cost of executing the SampleLeft algorithm. For the computation cost of verifying, three schemes require $3NmT_m$, $(N^2 + N)T_m$ and $O(mn)T_m + O(lmn)T_+$, respectively, to verify a signature. Since $m > 6N\log q$, it is obvious that the ideal lattice-based scheme [11] has higher efficiency in terms of the computation costs of signing and verifying. Finally, since our RIBS scheme employs the binary tree to revoke users, the authority's workload performing the update procedure is the logarithm of nonrevoked users. In Xiang's RIBS scheme, it also employs the binary tree to revoke users. But it then uses secure channels to send periodic signing keys to nonrevoked users, and it will make the authority having linearity level workload. In scheme [11], the authority's workload is linearity of nonrevoked users (Table 1).

5.2. Functionality Comparisons. Firstly, as indicated in Table 2, Wei et al.'s [20] RIBS, Yang et al.'s [21] RIBS, and Zhao et al.'s [22] RIBS schemes can resist key exposure attacks in the standard (SD) model with scalability under different Diffie–Hellman assumptions (namely, q-DHE, CDH, and (3,n)-MDHE assumptions, respectively). However, these schemes cannot resist quantum computer attacks. Our RIBS, Xinyin's [10] RIBS, and Hung et al.'s [11] RIBS schemes from lattices can resist quantum computer attacks. Secondly, both our RIBS and Hung et al.'s [11] RIBS schemes broadcast the update keys regularly through public channels. But Xinyin's [10] RIBS scheme uses secure channels to send periodic update keys. Thirdly, both Xinyin's [10] RIBS and Hung's [11] RIBS schemes are not scalable RIBS schemes. Although Xiang's RIBS scheme uses the binary tree structure, it requires a secure channel between a user and PKG and does not support scalability. Our RIBS is a scalable RIBS scheme by using the binary structure and public channel. Finally, both our RIBS and Hung et al.'s RIBS schemes have a limitation that the schemes are only proved secure in the random oracle (RO) model under SIS assumption and RSIS assumption, respectively.

6. Conclusion

In this paper, we have proposed the first scalable RIBS scheme with signing key exposure resistance over lattices, in which the update keys are broadcast regularly over public channels. Then, we proved its EUF-sID-CMA security under the SIS assumption. In the future, we intend to improve the efficiency of the lattice-based RIBS with signing key exposure resistance.

Data Availability

All data used to support the findings of this study are included within the article.

Disclosure

Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this article.

Acknowledgments

Jian Weng was supported by National Natural Science Foundation of China (Grant Nos. 61825203, U1736203 and No. 61732021), the Major Program of Guangdong Basic and Applied Research (Grant No. 2019B030302008), and Science and Technology Program of Guangzhou of China (Grant No. 201802010061). Congge Xie was supported by National Natural Science Foundation of China (Grant No. 61872153), Science and Technology Planning Project of Guangdong Province of China (Grant No. 2017B010111005), National Key R&D Program of China (Grant No. 2018YFB1402600). Jinming Wen was supported by National Natural Science Foundation of China (Grant No. 11871248).

References

[1] D. Boneh and M. K. Franklin, "Identity-based encryption from the weil pairing," in *Proceedings of the 21st Annual International Cryptology Conference-Advances in Cryptology (CRYPTO 2001)*, pp. 213–229, Santa Barbara, CA, USA, August 2001.

[2] A. Boldyreva, V. Goyal, and V. Kumar, "Identity-based encryption with efficient revocation," in *Proceedings of the 2008 ACM Conference on Computer and Communications Security (CCS 2008)*, pp. 417–426, Alexandria, VA, USA, October 2008.

[3] B. Libert and D. Vergnaud, "Adaptive-ID secure revocable identity-based encryption," in *Proceedings of the Cryptographers' Track at the RSA Conference 2009-Topics in Cryptology (CT-RSA 2009)*, pp. 1–15, San Francisco, CA, USA, April 2009.

[4] J. Chen, H. W. Lim, S. Ling, H. Wang, and K. Nguyen, "Revocable identity-based encryption from lattices," in *Proceedings of the 17th Australasian Conference-Information Security and Privacy (ACISP)*, pp. 390–403, Wollongong, Australia, July 2012.

[5] S. Cheng and J. Zhang, "Adaptive-ID secure revocable identity-based encryption from lattices via subset difference method," in *Proceedings of the 11th International Conference-Information Security Practice and Experience (ISPEC 2015)*, pp. 283–297, Beijing, China, May 2015.

[6] J. H. Seo and K. Emura, "Revocable identity-based encryption revisited: security model and construction," in *Proceedings of the 16th International Conference on Practice and Theory in Public-Key Cryptography, PKC 2013*, pp. 216–234, Nara, Japan, January 2013.

[7] T. Tsai, Y. Tseng, and T. Wu, "Provably secure revocable ID-based signature in the standard model," *Security and Communication Networks*, vol. 6, no. 10, pp. 1250–1260, 2013.

[8] Y. Sun, F. Zhang, L. Shen, and R. H. Deng, "Revocable identity-based signature without pairing," in *Proceedings of the 5th International Conference on Intelligent Networking and Collaborative Systems 2013*, pp. 363–365, Xi'an, China, September 2013.

[9] Y. Hung, T. Tsai, Y. Tseng, and S. Huang, "Strongly secure revocable ID-based signature without random oracles," *ITC*, vol. 43, no. 3, pp. 264–276, 2014.

[10] X. Xinyin, "Adaptive secure revocable identity-based signature scheme over lattices," *Computer Engineering*, vol. 10, p. 25, 2015.

[11] Y.-H. Hung, Y.-M. Tseng, and S.-S. Huang, "Revocable ID-based signature with short size over lattices," *Security and Communication Networks*, vol. 2017, Article ID 7571201, 9 pages, 2017.

[12] V. Lyubashevsky, "Lattice signatures without trapdoors," in *Proceedings of the 31st Annual International Conference on the Theory and Applications of Cryptographic Techniques-Advances in Cryptology (EUROCRYPT 2012)*, pp. 738–755, Cambridge, UK, April 2012.

[13] A. Shamir, "Identity-based cryptosystems and signature schemes," in *Proceedings of the CRYPTO'84-Advances in Cryptology (CRYPTO 1984)*, pp. 47–53, Santa Barbara, CA, USA, April 1984.

[14] Y. Ishida, Y. Watanabe, and J. Shikata, "Constructions of CCA-secure revocable identity-based encryption," in *Proceedings of the 20th Australasian Conference on Information Security and Privacy, ACISP 2015*, pp. 174–191, Brisbane, Australia, June-July 2015.

[15] K. Emura, J. H. Seo, and T. Youn, "Semi-generic transformation of revocable hierarchical identity-based encryption and its DBDH instantiation," *IEICE Transactions*, vol. E99.A, no. 1, pp. 83–91, 2016.

[16] J. H. Seo and K. Emura, "Revocable hierarchical identity-based encryption via history-free approach," *Theoretical Computer Science*, vol. 615, pp. 45–60, 2016.

[17] K. Lee and S. Park, "Revocable hierarchical identity-based encryption with shorter private keys and update keys," *Designs, Codes and Cryptography*, vol. 86, no. 10, pp. 2407–2440, 2018.

[18] A. Takayasu and Y. Watanabe, "Lattice-based revocable identity-based encryption with bounded decryption key exposure resistance," in *Proceedings of the 22nd Australasian Conference on Information Security and Privacy, ACISP 2017*, pp. 184–204, Auckland, New Zealand, July 2017.

[19] Z. Liu, X. Zhang, Y. Hu, and T. Takagi, "Revocable and strongly unforgeable identity-based signature scheme in the standard model," *Security and Communication Networks*, vol. 9, no. 14, pp. 2422–2433, 2016.

[20] J. Wei, W. Liu, and X. Hu, "Forward-secure identity-based signature with efficient revocation," *International Journal of Computer Mathematics*, vol. 94, no. 7, pp. 1390–1411, 2017.

[21] X. Yang, T. Ma, P. Yang, F. An, and C. Wang, "Security analysis of a revocable and strongly unforgeable identity-based signature scheme," *ITC*, vol. 47, no. 3, pp. 575–587, 2018.

[22] J. Zhao, B. Wei, and Y. Su, "Communication-efficient revocable identity-based signature from multilinear maps," *Journal of Ambient Intelligence and Humanized Computing*, vol. 10, no. 1, pp. 187–198, 2019.

[23] C. Gentry, C. Peikert, and V. Vaikuntanathan, "Trapdoors for hard lattices and new cryptographic constructions," in *Proceedings of the 40th Annual ACM Symposium on Theory of Computing*, pp. 197–206, Victoria, Canada, May 2008.

[24] D. Micciancio and C. Peikert, "Trapdoors for lattices: simpler, tighter, faster, smaller," in *Proceedings of the 31st Annual International Conference on the Theory and Applications of Cryptographic Techniques-Advances in Cryptology (EUROCRYPT 2012)*, pp. 700–718, Cambridge, UK, April 2012.

[25] M. Ajtai, "Generating hard instances of the short basis problem," in *Proceedings of the 26th International Colloquium-Automata, Languages and Programming (ICALP 1999)*, pp. 1–9, Prague, Czech Republic, July 1999.

[26] J. Alwen and C. Peikert, "Generating shorter bases for hard random lattices," in *Proceedings of the 26th International Symposium on Theoretical Aspects of Computer Science (STACS 2009)*, pp. 75–86, Freiburg, Germany, February 2009.

[27] S. Agrawal, D. Boneh, and X. Boyen, "Efficient lattice (H)IBE in the standard model," in *Proceedings of the 29th Annual International Conference on the Theory and Applications of Cryptographic Techniques—Advances in Cryptology (EUROCRYPT 2010)*, pp. 553–572, Riviera, French, May-June 2010.

[28] S. Agrawal, D. Boneh, and X. Boyen, "Lattice basis delegation in fixed dimension and shorter-ciphertext hierarchical IBE," in *Proceedings of the 30th Annual Cryptology Conference-Advances in Cryptology (CRYPTO 2010)*, pp. 98–115, Santa Barbara, CA, USA, August 2010.

[29] D. Micciancio and O. Regev, "Worst-case to average-case reductions based on Gaussian measures," *SIAM Journal on Computing*, vol. 37, no. 1, pp. 267–302, 2007.

[30] O. Regev, "On lattices, learning with errors, random linear codes, and cryptography," in *Proceedings of the 37th Annual ACM Symposium on Theory of Computing*, pp. 84–93, Baltimore, MD, USA, May 2005.