SCIS 2016 2016 Symposium on
Cryptography and Information Security
Kumamoto, Japan, Jan. 19 - 22, 2016
The Institute of Electronics,
Information and Communication Engineers

Lattice-based Multi-signature with Linear Homomorphism

Rakyong Choi * Kwangjo Kim *

Abstract: This paper extends the lattice-based linearly homomorphic signature to have multiple signers with the security proof. In our construction, we assume that there are one trusted dealer and either single signer or multiple signers for a message. The dealer pre-shares the message vector $\mathbf{v}$ during the set-up phase and issues a pre-shared vector $\mathbf{v}_i$ to each signer. Then, from partial signatures σ_i of $\mathbf{v}_i$ signed by each signer, one obtains a valid signature σ of $\mathbf{v}$ by combining all partial signatures σ_i of $\mathbf{v}_i$. We use well-known lattice-based algorithms like trapdoor generation algorithm and extracting basis algorithm to distribute different secret keys to each signer. Our signature holds multi-unforgeability and weakly context hiding property and is shown to be provably secure in the random oracle model under k-Small Integer Solution problem assuming the soundness of Boneh and Freeman's signature.

Keywords: lattice, linearly homomorphic signature, k-SIS, multi-signature

1 Introduction

1.1 Background and Motivation

In the improvement of cloud systems, one of uprising security challenges is how the cloud server computes a function of encrypted messages without decryption. Despite of Gentry's seminal work on fully homomorphic encryption on ideal lattices [1] to enable the server to calculate any function of encrypted message without decryption, there is another security issue in cloud system; how the cloud server gives authenticity for the function of encrypted message. A digital signature is a well-known cryptographic primitive that gives authenticity to the server and non-repudiation. This primitive guarantees that the information is not modified during its transmission, processing and storage.

Separately, lattices are a fascinating tool in modern cryptography. Lattice-based cryptography has a lot of advantages that their security is based on the average-case hardness problems and such problems remain secure against quantum computing attacks. With these advantages, there have been many cryptographic primitives based on lattices such as fully homomorphic encryption [2, 3], multilinear maps [4] and fully homomorphic signatures [5–7].

For authenticity of cloud systems, the cloud server should generate the proper signature for a computation of messages without permission from the signer of each message. If the signature satisfies this condition, we say that the signature has the *homomorphic property*. Especially, a signature is called *linearly homomorphic* when it supports constructing the proper signature for the linear combination of messages [5, 6] and *fully homomorphic* when it supports constructing the proper signature for any function of messages [7].

* School of Computing, KAIST. 291, Daehak-ro, Yuseong-gu, Daejeon, South Korea 34141. {thepride, kkj}@kaist.ac.kr

1.2 Our Contribution

Considering the real world, some information on cloud system is signed by an organization instead of an individual. There should be at least two people to authenticate the message where each person has his/her secret key. In this situation, we need multiple signers with different secret keys for a single message and the corresponding signature is valid only if all users are trustworthy.

There are some progress on homomorphic signatures with multiple secret keys for each message like homomorphic aggregate signatures which support multiple secret keys for different messages [8]. But we have a lack of knowledge on homomorphic signatures with multiple secret keys for a single message. In this paper, we suggest a new linearly homomorphic signature with multiple signers.

By adopting the lattice-based algorithm called extracting basis algorithm, we achieve the novel lattice-based signature that is proper for the aforementioned situation. In particular, we give a security proof on our signature under k-Small Integer Solution (k-SIS) problem assuming the soundness of Boneh and Freeman's signature [5]; our signature satisfies multi-unforgeability and weakly context hiding property. With all functionalities provided by linearly homomorphic signature, our signature provides another functionality that signature derived from multiple signers also satisfies the linearly homomorphic property.

1.3 Related Work

In 2011, Boneh and Freeman [5] published their seminal work on linearly homomorphic signature over binary fields based on lattices with new lattice-based hard problems called k-SIS problem. After that, lattices have become a main tool to make linearly and fully homomorphic signatures.

After that, Boneh and Freeman [6] suggested that building some bounded homomorphic signature is possible using ideal lattices from Gentry's fully homomorphic encryption [1]. Zhang *et al.* [8] and Jing [9] separately suggested the homomorphic aggregate signature with linear homomorphism which doesn't need to have the same secret key to combine multiple messages. Recently, fully homomorphic signature becomes available by Gorbunov, Vaikuntanathan, and Wichs [7].

On the other hand, there are a lot of lattice-based digital signatures with multiple signers. Gordon *et al.* [10] introduced the first construction of a lattice-based group signature with a new algorithm for sampling a basis for an orthogonal lattice and its trapdoor. Feng *et al.* [11] proposed a threshold signature whose signing algorithm proceeds sequentially through each member of the group to be highly interactive. Cayrel *et al.* [12] proposed a lattice-based threshold ring signature, in which at least t members are required to create an anonymous signature. In this system, each member has its own public key, and verification time grows linearly with the number of members. Also, Bendlin *et al.* [13] suggested a threshold variant of Gentry *et al.*'s signature [14].

But, there is no linearly homomorphic signature with multiple signers for a single message in the open literature to the best of our knowledge.

1.4 Outline of the Paper

Section 2 describes a background on lattices including Small Integer Solution (SIS) problem, some lattice-based algorithms used in our signature, and the formal definition of linearly homomorphic signature with detailed construction in Appendix A. We give a formal definition of the linearly homomorphic signature with multiple signers and present our construction in Section 3. In Section 4, we discuss the security requirements of the proposed signature with rigorous proof in Appendix B. And we give a concluding remark with future work in Section 5.

2 Preliminaries

2.1 Notation

We denote vectors using small bold letters (*e.g.*, $\mathbf{x}$, $\mathbf{y}$) and denote matrices as big bold letters (*e.g.*, $\mathbf{A}$, $\mathbf{B}$).

Let $\mathbb{R}$ and $\mathbb{Z}$ express the set of real numbers and the set of integers, respectively and small alphabet letters mean real numbers (*e.g.*, a, b, c).

For any integer $q \geq 2$, $\mathbb{Z}_q$ denotes the ring of integers modulo q and $\mathbb{Z}_q^{n\times m}$ denotes the set of $n \times m$ matrices with entries in $\mathbb{Z}_q$. When $\mathbf{A} \in \mathbb{Z}_q^{n\times m_1}$, $\mathbf{B} \in \mathbb{Z}_q^{n\times m_2}$, we write the concatenation of $\mathbf{A}$ and $\mathbf{B}$ as $\mathbf{A}\|\mathbf{B} \in \mathbb{Z}_q^{n\times(m_1+m_2)}$.

Let $f(a, b)$ be a function f on a and b. We say a function $f : \mathbb{Z} \to \mathbb{R}^+$ is *negligible* when $f = O(n^{-c})$ for all $c > 0$ and denoted by $\mathrm{negl}(n)$. A function $g(m) = \lceil m \rceil$ is the ceiling function from $\mathbb{R}$ to $\mathbb{Z}$ such that $g(m)$ is the smallest integer which is greater than or equal to m.

$\|\mathbf{x}\|$ represents *the Euclidean norm* of $\mathbf{x}$ and $\|\mathbf{B}\|$ represents the maximum of Euclidean norms of the columns of $\mathbf{B}$. For instance, when $\mathbf{B} = \{\mathbf{b}_1|\mathbf{b}_2|\cdots|\mathbf{b}_m\}$, $\|\mathbf{B}\| = \max_i \|\mathbf{b}_i\|$. Then, we denote $\widetilde{\mathbf{B}} = (\widetilde{\mathbf{b}}_1|\widetilde{\mathbf{b}}_2|\cdots|\widetilde{\mathbf{b}}_m)$ for the Gram-Schmidt orthogonalization of columns of $\mathbf{B}$ and denote $\|\widetilde{\mathbf{B}}\| = \max_i \|\widetilde{\mathbf{b}}_i\|$ for *Gram-Schmidt norm* of $\mathbf{B}$.

2.2 Lattices and How to Delegate a Basis

Briefly, lattices Λ can be defined as a discrete subgroup of $\mathbb{R}^m$ with its basis $\mathcal{S}$. A basis $\mathcal{S}$ of Λ is a set of linearly independent vectors $\mathcal{S} = \{\mathbf{b}_1, \mathbf{b}_2, \cdots, \mathbf{b}_m\}$ which spans the lattice Λ and $\mathbf{S} = (\mathbf{b}_1|\mathbf{b}_2|\cdots|\mathbf{b}_m)$ is a basis matrix of lattice Λ.

Integer lattices are defined as a subgroup of $\mathbb{Z}^m$ instead of $\mathbb{R}^m$. For a matrix $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$, we can denote lattices as a set $\Lambda_q^{\mathbf{u}}(\mathbf{A}) = \{\mathbf{e} \in \mathbb{Z}^m | \mathbf{A}\cdot\mathbf{e} = \mathbf{u} \bmod q\}$ and as a set $\Lambda_q^{\perp}(\mathbf{A}) = \{\mathbf{e} \in \mathbb{Z}^m | \mathbf{A}\cdot\mathbf{e} = \mathbf{0} \bmod q\}$ when $\mathbf{u} = \mathbf{0}$.

Alwen and Peikert [15] introduced the trapdoor generation algorithm $\mathbf{TrapGen}(n, m, q)$ which generates a matrix $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ with its "trapdoor" matrix $\mathbf{T} \in \mathbb{Z}^{m\times m}$ satisfying the following functionality:

TrapGen(n, m, q) :
For the security parameter n, $m = \lceil 6n \log q \rceil$ and an integer q, this algorithm outputs a matrix $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ and its trapdoor matrix $\mathbf{T}$ such that $\mathbf{T}$ is a basis of $\Lambda_q^{\perp}(\mathbf{A})$ with low Gram-Schmidt norm $\|\widetilde{\mathbf{T}}\| \leq 30\sqrt{n \log q}$.

Without loss of generality, we assume that a matrix $\mathbf{A}$ extracted from $\mathbf{TrapGen}(n, m, q)$ has a full rank. In our signature, we extract the public keys $\{\mathbf{A}_i\}_{i=1}^g$ and secret keys $\{\mathbf{T}_i\}_{i=1}^g$ from trapdoor generation algorithm $\mathbf{TrapGen}(n, m, q)$ and extracting basis algorithm $\mathbf{ExtBasis}(\mathbf{T}, \mathbf{B})$ like Cash *et al.*'s work [16]. Then, we extract a new basis $\mathbf{S}_i$ from the new matrix $\mathbf{B} = \mathbf{A}\|\mathbf{A}'$, where $\mathbf{A}' \in \mathbb{Z}_q^{n\times m'}$, using the $\mathbf{ExtBasis}(\mathbf{T}, \mathbf{B})$ and generate the signature σ_i using Gaussian sampling algorithm $\mathbf{SamplePre}(\mathbf{A}, \mathbf{T}, \gamma, \mathbf{u})$ from Gentry *et al.*'s work [14]. We leave the discussion on Gaussian sampling algorithm in the next section and focus ourselves on how extracting basis algorithm operates.

ExtBasis(T, B) :
For the trapdoor matrix $\mathbf{T}$ of $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ and the matrix $\mathbf{B} = \mathbf{A}\|\mathbf{A}' \in \mathbb{Z}_q^{n\times(m+m')}$, this algorithm outputs a basis $\mathbf{S}$ for $\Lambda_q^{\perp}(\mathbf{B})$ with $\|\widetilde{\mathbf{S}}\| = \|\widetilde{\mathbf{T}}\|$, *i.e.*, Gram-Schmidt norm of $\mathbf{S}$ is equal to Gram-Schmidt norm of $\mathbf{T}$.

2.3 Discrete Gaussian Distribution and Sampling Algorithm

Given L be any subset of $\mathbb{Z}^m$, a Gaussian function on $\mathbb{R}^m$ with center $\mathbf{c}$ and parameter γ can be defined

as $\rho_{\gamma,\mathbf{c}}(\mathbf{x}) = \exp(-\pi\|\mathbf{x}\text{-}\mathbf{c}\|^2/\gamma^2)$ for any vector $\mathbf{c} \in \mathbb{R}^m$ and any positive parameter $\gamma > 0$,

For a subset $L \subset \mathbb{Z}^m$, we can define *discrete Gaussian distribution*, which is the m-dimensional Gaussian distribution whose support is restricted to the subset L and its density function is defined as

$$\mathcal{D}_{L,\gamma,\mathbf{c}}(\mathbf{x}) = \frac{\rho_{\gamma,\mathbf{c}}(\mathbf{x})}{\sum_{y\in L}\rho_{\gamma,\mathbf{c}}(\mathbf{y})}$$

and for the sake of simplicity, we denote $\rho_\gamma(\mathbf{x})$ and $\mathcal{D}_{L,\gamma}(\mathbf{x})$ when center $\mathbf{c} = \mathbf{0}$.

Gentry *et al.* [14] proved that this distribution can be sampled efficiently for $\gamma \geq \|\widetilde{\mathbf{T}}\| \cdot \omega(\sqrt{\log n})$ where $\mathbf{T}$ is a trapdoor matrix of an n-dimensional lattice Λ as follows:

SamplePre(A, T, γ, u) :
For the matrix $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$, its trapdoor matrix $\mathbf{T} \in \mathbb{Z}_q^{m\times m}$, a real number $\gamma > 0$, and a vector $\mathbf{u} \in \mathbb{Z}^n$, this algorithm outputs a sample σ from a distribution that is statistically close to $\mathcal{D}_{\Lambda_q^{\mathbf{u}}(\mathbf{A}),\gamma}$.

The smoothing parameter $\eta_\epsilon(\Lambda)$ of Λ enables every coset of Λ to get roughly equal mass in the following **Lemmas 1** and **2**.

Lemma 1. *[14] Let q be a prime and n, m be integers with $m > 2n\log q$. Let f be some $\omega(\sqrt{\log m})$ function. Then, there is a negligible function $\epsilon(m)$ such that for all but at most q^{-n} fraction of matrix $\boldsymbol{A} \in \mathbb{Z}_q^{n\times m}$, we have $\eta_{\epsilon(m)}(\Lambda_q^\perp(\boldsymbol{A})) < f(m)$.*

Lemma 2. *[5] Let $\Lambda \subset \mathbb{R}^n$ be a lattice. Suppose $\rho \geq \eta_\epsilon(\Lambda)$ for some negligible ϵ. Then, we have*

$$\Pr\left[0 \leq \|\mathbf{v}\| \leq 2\rho\sqrt{\frac{n}{2\pi}} : \mathbf{v} \leftarrow \mathcal{D}_{\Lambda_q^{\mathbf{u}}(\mathbf{A}),\gamma}\right] \geq 1-\mathrm{negl}(n).$$

Lemma 1 declares that we extract a sample vector almost uniformly by sampling algorithm **SamplePre** $(\mathbf{A}, \mathbf{T}, \gamma, \mathbf{u})$ with proper parameters and **Lemma 2** determines the upper bound on the length $\|\mathbf{v}\|$ of a sample vector $\mathbf{v}$ from the Gaussian distribution $\mathcal{D}_{\Lambda_q^{\mathbf{u}}(\mathbf{A}),\gamma}$.

2.4 Small Integer Solution Problem and k-SIS Problem

Small Integer Solution (SIS) problem on a lattice $\Lambda_q^\perp(\mathbf{A}) = \{\mathbf{e} \in \mathbb{Z}^m | \mathbf{A}\cdot\mathbf{e} = \mathbf{0} \bmod q, \}$ where $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ is to find a small vector $\mathbf{e} \in \Lambda_q^\perp(\mathbf{A})$ whose coefficients are either $-1, 0,$ or 1. In this paper, we focus ourselves on a modified SIS problem called k-SIS problem and prove the security of our signature based on this problem.

Problem. (k-SIS problem) Given a matrix $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ and k short vectors $\mathbf{e}_1, \mathbf{e}_2, \cdots, \mathbf{e}_k \in \Lambda_q^\perp(\mathbf{A})$ satisfying $\mathbf{A}\cdot\mathbf{e}_i = 0 \pmod q$ for all positive integer i with $i \leq k$, find a short vector $\mathbf{e} \in \mathbb{Z}^m$ satisfying $\|\mathbf{e}\| \leq \beta$ and $\mathbf{A}\cdot\mathbf{e} = 0 \pmod q$, such that $\mathbf{e}$ is not in $\mathbb{Q}$-span$(\{\mathbf{e}_1, \mathbf{e}_2, \cdots, \mathbf{e}_k\})$.

This problem is called k-SIS$_{q,m,\beta,\gamma}$ problem when each $\mathbf{e}_i$ is drawn from the distribution $\mathcal{D}_{\Lambda_q^\perp(\mathbf{A}),\gamma}$.

k-SIS problem is a natural generalization of a normal SIS problem since this is a normal SIS problem when $k = 0$, *i.e.*, we have no information of the short vectors on the lattice. Boneh and Freeman proved that this k-SIS problem can be reduced to a normal SIS problem [5].

2.5 Linearly Homomorphic Signature (LHS)

We let the public parameters **params**$=(N, k, L, m, q, \gamma)$ where $N = n$ is the dimension of vectors to be signed, k is the dimension of the subspace to be signed $(k < n)$, L is the maximum number of signatures in linear combinations, $m(n, L) > n$ is an integer, $q(n, L)$ is an odd prime, and $\gamma(n, L)$ is a real number.

With those parameters, Boneh and Freeman [5] presented the linearly homomorphic signature over binary fields with a tuple of PPT algorithms $\mathcal{LHS} =$ (**Setup**, **Sign**, **Combine**, **Verify**) which does the following functionality:

Setup(n, params) :
Given a security parameter n and public parameters **params**$=(N, k, L, m, q, \gamma)$,

1. $(\mathbf{A}, \mathbf{T}) \leftarrow$ **TrapGen**$(n, m, 2q)$ where a matrix $\mathbf{A} \in \mathbb{Z}_{2q}^{n\times m}$ and its trapdoor basis $\mathbf{T}$ of $\Lambda_{2q}^\perp(\mathbf{A})$ satisfies that $\|\widetilde{\mathbf{T}}\| \leq 30\sqrt{n\log 2q}$.
2. Let $H : \{0,1\}^* \to \mathbb{Z}_{2q}^{n\times m}$ be a hash function, viewed as a random oracle.
3. Output the public key $pk \leftarrow (\mathbf{A}, H)$ and the secret key $sk \leftarrow (\mathbf{A}, H, \mathbf{T})$.

Sign(sk, id, v) :
Given a secret key $sk \leftarrow (\mathbf{A}, H, \mathbf{T})$, a tag $id \in \{0,1\}^n$ and a vector $\mathbf{v} \in \mathbb{F}_2^n$,

1. Set $\mathbf{B} \leftarrow \mathbf{A}\|H(id) \in \mathbb{Z}_{2q}^{n\times 2m}$.
2. Let $\mathbf{S} \leftarrow$ **ExtBasis**$(\mathbf{T}, \mathbf{B})$ be a basis for $\Lambda_{2q}^\perp(B)$ with $\|\widetilde{\mathbf{S}}\| = \|\widetilde{\mathbf{T}}\|$.
3. Output $\sigma \leftarrow$ **SamplePre**$(\mathbf{B}, \mathbf{S}, \gamma, q\cdot\mathbf{v})$.

Combine$(pk, id, \{(\alpha_i, \sigma_i)\}_{i=1}^l)$:
Given a public key $pk = (\mathbf{A}, H)$, a tag $id \in \{0,1\}^n$ and pairs $\{(\alpha_i, \sigma_i)\}_{i=1}^l$ where $\alpha_i \in \mathbb{F}_2 = \{0,1\}$ and σ_i is a signature of the i-th vector $\mathbf{v}_i$, output $\sigma \leftarrow \Sigma_{i=1}^l \alpha_i\sigma_i \in \mathbb{Z}^{2m}$.

Verify$(pk, id$, **y**, $\sigma)$:
Given a public key $pk = (\mathbf{A}, H)$, a tag $id \in 0, 1^n$, a vector $\mathbf{y} \in \mathbb{F}_2^n$ and a signature $\sigma \in \mathbb{Z}^{2m}$,

1. Set $\mathbf{B} \leftarrow \mathbf{A}\|H(id) \in \mathbb{Z}_{2q}^{n\times 2m}$.
2. If $\|\sigma\| \leq L\cdot\gamma\sqrt{2m}$ and $\mathbf{B}\cdot\sigma = q\cdot\mathbf{y} \bmod 2q$, output 1 (accept). Otherwise, output 0 (reject).

To verify the correctness of this signature, for each (pk, sk), we should have

a. For all tags id and every vector $\mathbf{y}$, the verification algorithm $\mathbf{Verify}(pk, id, \mathbf{y}, \sigma)$ outputs 1 for every valid signature $\sigma \leftarrow \mathbf{Sign}(sk, id, \mathbf{y})$.

b. Whenever we operate linear combination of the vector, we can output the valid signature for such combination.

The security requirements of linearly homomorphic signature are stated in *unforgeability* and *weakly context hiding* property as below:

Definition 1. (unforgeability). A linearly homomorphic signature is *unforgeable* if the advantage of any PPT adversary $\mathcal{A}$, in the following security game is negligible in the security parameter n.

Setup :
The challenger $\mathcal{C}$ sets $(pk, sk) \leftarrow \mathbf{Setup}(n, \mathbf{params})$, then sends the public key pk to $\mathcal{A}$.

Queries :
Proceeding adaptively, the adversary $\mathcal{A}$ specifies a sequence of k-dimensional subspaces V_i with basis vectors $\{\mathbf{v}_j^{(i)}\}_{j=1}^{k}$. For each i, the challenger $\mathcal{C}$ chooses a tag $id_i \leftarrow \{0,1\}^n$ uniformly and gives id_i with j signatures $\sigma_{ij} \leftarrow \mathbf{Sign}(sk, id_i, \mathbf{v}_j^{(i)})$ for $j = 1, 2, \cdots, k$.

Output :
The adversary $\mathcal{A}$ outputs a tag $id^* \in \{0,1\}^n$, a non-zero vector $\mathbf{y}^*$, and a signature σ^*.

The adversary $\mathcal{A}$ wins the game if the signature σ is valid and either (1) $id^* \neq id_i$ for all i, or (2) $id^* = id_i$ for some i but $\mathbf{y}^* \notin V_i$.

Definition 2. (weakly context hiding). A linearly homomorphic signature is *weakly context hiding* if the advantage of any PPT adversary $\mathcal{A}$, in the following security game is negligible in the security parameter n.

Setup :
The challenger $\mathcal{C}$ sets $(pk, sk) \leftarrow \mathbf{Setup}(n, \mathbf{params})$, then sends both public key pk and secret key sk to $\mathcal{A}$.

Challenge :
The adversary $\mathcal{A}$ outputs two k-dimensional vector spaces V_0, V_1 with basis vectors $\{\mathbf{v}_i^{(0)}\}_{i=1}^{k}$ and $\{\mathbf{v}_i^{(1)}\}_{i=1}^{k}$, respectively and linear functions on both $\{\mathbf{v}_i^{(0)}\})_{i=1}^{k}$ and $\{\mathbf{v}_i^{(1)}\}_{i=1}^{k}$ which satisfies

$$f_j\left(\mathbf{v}_1^{(0)}, \mathbf{v}_2^{(0)}, \cdots, \mathbf{v}_k^{(0)}\right) = f_j\left(\mathbf{v}_1^{(1)}, \mathbf{v}_2^{(1)}, \cdots, \mathbf{v}_k^{(1)}\right)$$

for all $j = 1, 2, \cdots, s$.

The challenger $\mathcal{C}$ chooses $b \in \{0,1\}$ and a tag $id \in \{0,1\}^n$ and signs the vector space V_b with a tag id. Then, $\mathcal{C}$ uses $\mathbf{Combine}(pk, id, \{(\alpha_i, \sigma_i)\}_{i=1}^{k})$ algorithm to derive signatures σ_j of the function $f_j\left(\mathbf{v}_1^{(b)}, \mathbf{v}_2^{(b)}, \cdots, \mathbf{v}_k^{(b)}\right)$ for all $j = 1, 2, \cdots, s$. The adversary $\mathcal{A}$ gets signatures σ_j. The function can be out adaptively after choosing V_0 and V_1.

Output :
The adversary $\mathcal{A}$ outputs a bit b'.

The adversary $\mathcal{A}$ wins the game if $b = b'$.

Lemma 3. *Let $\mathcal{LHS}$ be the linearly homomorphic signature over $\mathbb{F}_2$ as above. Suppose q be a prime, n, m be integers with $m > 2n \log q$, and $\gamma > 30\sqrt{n \log 2q} \cdot \omega(\sqrt{\log n})$. Then $\|\sigma\| \leq L \cdot \gamma\sqrt{2m}$ and $\boldsymbol{B} \cdot \sigma = q \cdot \boldsymbol{y}$ mod $2q$ for all valid signatures $\sigma \leftarrow$ **Combine**$(pk, id, \{(\alpha_i, \sigma_i)\}_{i=1}^{l})$*

Moreover, **Lemmas 4** and **5** from Boneh and Freeman's work show that this signature is unforgeable in the random oracle model and it holds the weakly context hiding property [5].

Lemma 4. *Let $\mathcal{LHS}$ be the linearly homomorphic signature over $\mathbb{F}_2$ as above. Suppose that $m = \lceil 6n \log 2q \rceil$ and $\gamma = 30\sqrt{n \log 2q} \log n$. Let $\beta = L \cdot \gamma\sqrt{2m}$. Then $\mathcal{LHS}$ is unforgeable in the random oracle model assuming that k-$SIS_{q,2m,\beta,\gamma}$ problem is infeasible.*

Lemma 5. *Let $\mathcal{LHS}$ be the linearly homomorphic signature over $\mathbb{F}_2$ as above. Suppose that $k < \dfrac{\log n}{2 \log \log n}$, $m = \lceil 6n \log 2q \rceil$ and $\gamma = 30\sqrt{n \log 2q} \log n$. Then $\mathcal{LHS}$ is weakly context hiding.*

Proof. (sketch) In $\mathcal{LHS}$, one obtains a signature σ on a linear combination $\mathbf{v}$ of vectors by calculating a linear combination of the signatures $\sigma_1, \sigma_2, \cdots, \sigma_g$ of the original vectors $\mathbf{v}_1, \mathbf{v}_2, \cdots, \mathbf{v}_g$. The obtained signature σ on $\mathbf{v}$ is a linear combination of relatively short vectors in cosets of some lattice.

The signature σ does not leak information on the original signatures since a linear combination of k signatures from **Sign** algorithm is indeed a short vector sampled from a distribution that depends only on the computed function and the vector $\mathbf{v}$, *i.e.*, σ does not have any information on the original vectors $\mathbf{v}_1, \mathbf{v}_2, \cdots, \mathbf{v}_g$ and $\mathcal{LHS}$ holds the weakly context hiding property. □

3 Multi-signature with Linear Homomorphism

We give a formal definition of multi-signature with linear homomorphism over binary fields, having multiple signers. Then, we propose our construction on multi-signature with linear homomorphism.

3.1 Formal Definition

A multi-signature is a sort of group-based cryptography which involves two or more participants to authenticate a single message. In multi-signature, all group members are involved to sign the message with multiple

secret keys. To achieve multi-signature with linear homomorphism, there is a trusted dealer who pre-shares the original message and sends them to group members. With this process, we can achieve a multi-signature with linear homomorphism. The formal definition of our signature is as follows:

Definition 3. (multi-signature with linear homomorphism) A multi-signature with linear homomorphism is a tuple of PPT algorithms $\mathcal{MLH}$ = (**Setup**, **PreShare**, **Sign**, **Combine**, **LinCom**, **Verify**) with the following functionality:

Setup(n, g, **params**) :
For the security parameter n, a number of group members g, and public parameters **params**, this algorithm outputs a common public key pk and a pair of public keys and secret keys $\{pk_i, sk_i\}_{i=1}^{g}$.

PreShare(g, **v**) :
For a number of group members g and a vector $\mathbf{v}$, there is a dealer who pre-shares $\mathbf{v}$ into $\mathbf{v}_i$ such that $\mathbf{v} = \sum_{i=1}^{g} \mathbf{v}_i$.

Sign($\{pk_i, sk_i\}, id, \mathbf{v}_i$) :
For a pair of public keys and secret keys $\{pk_i, sk_i\}$, a tag id, and a vector $\mathbf{v}_i$, this algorithm outputs a signature ρ_i.

Combine($pk, id, g, \{\sigma_i\}_{i=1}^{g}$) :
For a public key pk, a tag id, a number of group members g, and set of signatures $\{\sigma_i\}_{i=1}^{g}$, this algorithm outputs a signature σ.

LinCom($pk, id, \{(g_j, \sigma_j)\}_{j=1}^{l}$) :
For a public key pk, a tag id, and a set of pairs $\{(g_j, \sigma_j)\}_{j=1}^{l}$, this algorithm outputs a signature σ_{lin}.

Verify(pk, id, **y**, σ) :
For a public key pk, a tag id, a vector $\mathbf{y}$ and a signature σ, this algorithm outputs either 0 (reject) or 1 (accept).

We can check from which subspace a vector $\mathbf{v}$ is taken for a given tag id. To verify the correctness of **Definition 3**, for each $(pk, \{sk_i\}_{i=1}^{g})$, we should have:

a. For every tag id and set of vectors $\{\mathbf{v}_i\}_{i=1}^{g}$, if a signature $\sigma_i \leftarrow$ **Sign**$(\{pk_i, sk_i\}, id, \mathbf{v}_i)$, then **Verify**$(pk, id, \mathbf{v}_i, \sigma_i) = 1$.

b. For every tag id and every vector $\mathbf{v}$, if a signature $\sigma \leftarrow$ **Combine**$(pk, id, g, \{\sigma_i\}_{i=1}^{g})$ where $\sigma_i \leftarrow$ **Sign**$(\{pk_i, sk_i\}, id, \mathbf{v}_i)$ and $\mathbf{v} = \sum_{i=1}^{g} \mathbf{v}_i$, then **Verify**$(pk, id, \mathbf{v}, \sigma) = 1$.

c. (linear homomorphism) Whenever we operate linear combination of the message with the same tag, we output the valid signature for such combination. *i.e.*, **Verify**$(pk, id, \mathbf{v}_{lin}, \sigma_{lin}) = 1$ where $\sigma_{lin} \leftarrow$ **LinCom**$(pk, id, \{(g_j, \sigma_j)\}_{j=1}^{l})$ and $\mathbf{v}_{lin} = \sum_{j=1}^{l} \mathbf{v}_j$.

3.2 Our Construction

We let the public parameters **params**=(N, k, L, m, q, γ) as $N = n$ is the dimension of vectors to be signed, k is the dimension of the subspace to be signed (thus, $k < n$), L is the maximum number of signatures in linear combinations, $m(n, L) > n$ is an integer, $q(n, L)$ is an odd prime, and $\gamma(n, L)$ is a real number.

There is a trusted dealer who calculates noise vectors and distributes the message with the noise to group members. The main idea of our signature is that the sum of such message is indeed the original message to sign. With the above public parameters, we instantiate our signature $\mathcal{MLH}$ with the following functionality:

Setup(n, g, **params**) :
Given a security parameter n and public parameters **params**=(N, k, L, m, q, γ), do the following:

1. Run **TrapGen**$(n, m, 2q)$ to generate a matrix $\{\mathbf{A}_i\}_{i=1}^{L} \in \mathbb{Z}_{2q}^{n \times m}$ and its corresponding trapdoor basis $\{\mathbf{T}_i\}_{i=1}^{L}$ of $\Lambda_{2q}^{\perp}(\mathbf{A}_i)$ such that $\|\widetilde{\mathbf{T}}_i\| \leq 30\sqrt{n \log 2q}$.
2. Define $\mathbf{A} = \mathbf{A}_1 \| \mathbf{A}_2 \| \cdots \| \mathbf{A}_L$
3. Let $H : \{0,1\}^* \rightarrow \mathbb{Z}_{2q}^{n \times m}$ be a hash function, viewed as a random oracle.
4. Output the common public key $pk \leftarrow (\mathbf{A}, H)$ and a pair of public keys and secret keys $(pk_i, sk_i) \leftarrow (\mathbf{A}_i, \mathbf{T}_i)$ for each i.

PreShare(g, **v**) :
For a number of group members $g \leq L$ and a vector $\mathbf{v}$, a dealer runs this algorithm to

1. Output the noise vector $\mathbf{u}_1, \mathbf{u}_2, \cdots, \mathbf{u}_{g-1}$ from a discrete Gaussian distribution $\mathcal{D}_{\Lambda_{2q}^{\perp}(\mathbf{A}), \gamma}$ using **SamplePre**$(\mathbf{A}, \mathbf{T}, \gamma, \mathbf{0})$ where $\mathbf{T} \leftarrow$ **ExtBasis**$(\mathbf{T}_1, \mathbf{A})$.
2. Get $\mathbf{u}_g = \sum_{i=1}^{g-1} \mathbf{u}_i$.
3. Output the set of pre-shared vectors $\{\mathbf{v}_1, \mathbf{v}_2, \cdots, \mathbf{v}_g\} = \{\mathbf{v}+\mathbf{u}_1, \mathbf{v}+\mathbf{u}_2, \cdots, \mathbf{v}+\mathbf{u}_g\}$ when g is odd and output $\{\mathbf{v}_1, \mathbf{v}_2, \cdots, \mathbf{v}_g\} = \{\mathbf{v}+\mathbf{u}_1, \mathbf{v}+\mathbf{u}_2, \cdots, \mathbf{u}_g\}$ when g is even.

Sign($\{pk_i, sk_i\}, id, \mathbf{v}_i$) :
For a pair of public keys and secret keys $(pk_i, sk_i) \leftarrow (\mathbf{A}_i, \mathbf{T}_i)$, a tag $id \in \{0,1\}^n$, and a vector $\mathbf{v}_i \in \mathbb{F}_2^n$, this algorithm is used to

1. Set $\mathbf{B} \leftarrow \mathbf{A} \| H(id) \in \mathbb{Z}_{2q}^{n \times 2m}$.
2. Let $\mathbf{S}_i \leftarrow$ **ExtBasis**$(\mathbf{T}_i, \mathbf{B})$ be a short basis for $\Lambda_{2q}^{\perp}(B)$ with $\|\widetilde{\mathbf{S}}_i\| = \|\widetilde{\mathbf{T}}_i\|$.
3. Output a partial signature $\sigma_i \leftarrow$ **SamplePre** $(\mathbf{B}, \mathbf{S}_i, \gamma, q \cdot \mathbf{v}_i)$.

Combine($pk, id, g, \{\sigma_i\}_{i=1}^{g}$) :
Given a public key $pk = (\mathbf{A}, H)$, a tag $id \in \{0,1\}^n$, a number of group members $g \leq L$, and set of signatures $\{\sigma_i\}_{i=1}^{g}$, output $\sigma = \sum_{i=1}^{g} \sigma_i \in \mathbb{Z}^{2m}$.

LinCom$(pk, id, \{(g_j, \sigma_j)\}_{j=1}^{l})$:
Given a public key $pk = (\mathbf{A}, H)$, a tag $id \in \{0,1\}^n$, and a set of pairs $\{(g_j, \sigma_j)\}_{j=1}^{l}$, output $\sigma_{lin} = \sum_{j=1}^{l} \sigma_j \in \mathbb{Z}^{2m}$ only if $\sum_{j=1}^{l} g_j \leq L$.

Verify$(pk, id, \mathbf{y}, \sigma)$:
Given a public key $pk = (\mathbf{A}, H)$, a tag $id \in \{0,1\}^n$, a vector $\mathbf{y} \in \mathbb{F}_2^n$ and a signature $\sigma \in \mathbb{Z}^{2m}$, do the following:

1. Set $\mathbf{B} \leftarrow \mathbf{A} \| H(id) \in \mathbb{Z}_{2q}^{n \times 2m}$.
2. If $\|\sigma\| \leq L \cdot \gamma\sqrt{2m}$ and $\mathbf{B} \cdot \sigma = q \cdot \mathbf{y} \bmod 2q$, output 1 (accept). Otherwise, output 0 (reject).

In the proposed signature, each $\mathbf{u}_i$ is uniformly random from **SamplePre**$(\mathbf{A}, \mathbf{T}_1, \gamma, \mathbf{0})$ with high probability and so is $\mathbf{v}_i$.

3.3 Correctness and Linear Homomorphism

To verify the correctness of the proposed signature, we must show that the correctness condition in **Definition 3** holds for any public key pk, secret keys $\{sk_i\}_{i=1}^{g}$ where $g \leq L$.

Theorem 1. *Suppose q be a prime, n, m be integers with $m > 2n \log q$, and $\gamma > 30\sqrt{n \log 2q} \cdot \omega(\sqrt{\log n})$. Then, the proposed multi-signature is correct for a single signature.*

Proof. Each partial signature σ_i from **Sign** algorithm is valid since **SamplePre**$(\mathbf{B}, \mathbf{S}_i, \gamma, q \cdot \mathbf{v}_i)$ outputs a signature σ_i such that $\mathbf{B} \cdot \sigma_i = q \cdot \mathbf{v}_i \bmod 2q$ and $\|\sigma_i\| \leq 2\rho\sqrt{\frac{n}{2\pi}} \leq \gamma\sqrt{2m}$ with extremely high probability by **Lemma 2**, *i.e.*, **Verify**$(pk, id, \mathbf{v}_i, \sigma_i) = 1$ for all pairs $\{(\sigma_i, \mathbf{v}_i)\}_{i=1}^{g}$.

Likewise, for a signature $\sigma = \sum_{i=1}^{g} \sigma_i \leftarrow$ **Combine** $(pk, id, g, \{\sigma_i\}_{i=1}^{g})$, we have $\|\sigma\| \leq \Sigma_{i=1}^{g} \|\sigma_i\| \leq g \cdot \gamma\sqrt{2m} \leq L \cdot \gamma\sqrt{2m}$ if $g \leq L$.

Since σ_i is a valid signature of $\mathbf{v}_i$, we have $\mathbf{B} \cdot \sigma_i = q \cdot \mathbf{v}_i \bmod 2q$ and since q is odd, this implies that $\mathbf{B} \cdot \sigma_i = 0 \bmod q$ and $\mathbf{B} \cdot \sigma_i = \mathbf{v}_i \bmod 2$. By simple addition, $\mathbf{B} \cdot \sigma = 0 \bmod q$ and $\mathbf{B} \cdot \sigma = \sum_{i=1}^{g} \mathbf{v}_i = \mathbf{v} \bmod 2$ since $\sigma = \sum_{i=1}^{g} \sigma_i$.

We get $\mathbf{B} \cdot \sigma = q \cdot \mathbf{v} \bmod 2q$ by Chinese Remainder Theorem and **Verify**$(pk, id, \mathbf{v}, \sigma) = 1$.

Thus, the proposed multi-signature is correct. □

Corollary 1. *Suppose q be a prime, n, m be integers with $m > 2n \log q$, and $\gamma > 30\sqrt{n \log 2q} \cdot \omega(\sqrt{\log n})$. Then, the proposed multi-signature is linearly homomorphic.*

Proof. Assume that we combine l messages $\mathbf{m}_1, \mathbf{m}_2, \cdots, \mathbf{m}_l$ into $\mathbf{m}_{lin} = \sum_{j=1}^{l} \mathbf{m}_j$. From the proof of **Theorem 1**, for all $\sigma_j \leftarrow$ **Combine**$(pk, id, g_j, \{\sigma_{j_k}\}_{k=1}^{g_j})$, $\|\sigma_j\| \leq g_j \cdot \gamma\sqrt{2m} \leq L \cdot \gamma\sqrt{2m}$ if $g_j \leq L$ and $\mathbf{B} \cdot \sigma_j = q \cdot \mathbf{m}_j \bmod 2q$.

Thus, for a signature $\sigma_{lin} = \sum_{j=1}^{l} \sigma_j \leftarrow$ **LinCom** $(pk, id, \{(g_j, \sigma_j)\}_{j=1}^{l})$, $\|\sigma_{lin}\| \leq \sum_{j=1}^{l} g_j \cdot \gamma\sqrt{2m} \leq L \cdot \gamma\sqrt{2m}$ if $\sum_{j=1}^{l} g_j \leq L$ and $\mathbf{B} \cdot \sigma = q \cdot \mathbf{m} \bmod 2q$, *i.e.*, **Verify**$(pk, id, \mathbf{m}_{lin}, \sigma_{lin}) = 1$ and the proposed signature satisfies the linearly homomorphic property if $\sum_{j=1}^{l} g_j \leq L$. □

4 Security of the Proposed Signature

4.1 Security Model

In the proposed signature, we assume that there are some corrupted group members to forge a signature but not all group members are corrupted.

Our multi-signature with linear homomorphism should satisfy *multi-unforgeability* in **Definition 4**.

Definition 4. (multi-unforgeability). A multi-signature with linear homomorphism is *multi-unforgeable* if the advantage of any PPT adversary $\mathcal{A}$, in the following security game is negligible in the security parameter n.

Setup :
The challenger $\mathcal{C}$ sets $(pk, \{sk_\tau\}_{\tau=1}^{g}) \leftarrow$ **Setup** $(n, g, \mathbf{params})$, then sends the public key pk and g_c secret keys sk_τ to the adversary $\mathcal{A}$ with $g_c < g$.

Queries :
Proceeding adaptively, the adversary $\mathcal{A}$ specifies a sequence of k-dimensional subspaces V_i with basis vectors $\{\mathbf{v}_j^{(i)}\}_{j=1}^{k}$. For each i, the challenger $\mathcal{C}$ chooses a tag $id_i \leftarrow \{0,1\}^n$ uniformly and

1. Run **PreShare**$(g, \mathbf{v}_j^{(i)})$ to pre-share the vector $\mathbf{v}_j^{(i)}$ into $\{\mathbf{v}_\tau^{(ij)}\}_{\tau=1}^{g}$.
2. Run **Sign**$(sk_\tau, id_i, \mathbf{v}_\tau^{(ij)})$ to get $\sigma_\tau^{(ij)}$ for $\tau = 1, 2, \cdots, g$.
3. Gives id_i with j signatures $\sigma_{ij} \leftarrow$ **Combine** $(pk, id_i, \{\sigma_\tau^{(ij)}\}_{\tau=1}^{g})$ for $j = 1, 2, \cdots, k$.

Output :
The adversary $\mathcal{A}$ outputs a tag $id^* \in \{0,1\}^n$, a non-zero vector $\mathbf{y}^*$, and a signature σ^*.

The adversary $\mathcal{A}$ wins the game if the signature σ is valid and either (1) $id^* \neq id_i$ for all i, or (2) $id^* = id_i$ for some i but $\mathbf{y}^* \notin V_i$.

Also, our signture should satisfy *weakly context hiding* property in **Definition 5** as Boneh and Freeman's signature.

Definition 5. (weakly context hiding property). A multi-signature with linear homomorphism is *weakly context hiding* if the advantage of any PPT adversary $\mathcal{A}$, in the following security game is negligible in the security parameter n.

Setup :

The challenger $\mathcal{C}$ sets $(pk, \{sk_\tau\}_{\tau=1}^{g}) \leftarrow$ **Setup** $(n, g, \textbf{params})$ for security parameter n and other public parameters **params**, then sends both public key pk and all secret keys $\{sk_\tau\}_{\tau=1}^{g}$ to the adversary $\mathcal{A}$.

Challenge :

The adversary $\mathcal{A}$ outputs two k-dimensional vector spaces V_0, V_1 with basis vectors $\{\mathbf{v}_i^{(0)}\}_{i=1}^{k}$ and $\{\mathbf{v}_i^{(1)}\}_{i=1}^{k}$, respectively and linear functions on both $\{\mathbf{v}_i^{(0)}\})_{i=1}^{k}$ and $\{\mathbf{v}_i^{(1)}\}_{i=1}^{k}$ which satisfies

$$f_j\left(\mathbf{v}_1^{(0)}, \mathbf{v}_2^{(0)}, \cdots, \mathbf{v}_k^{(0)}\right) = f_j\left(\mathbf{v}_1^{(1)}, \mathbf{v}_2^{(1)}, \cdots, \mathbf{v}_k^{(1)}\right)$$

for all $j = 1, 2, \cdots, s$.

The challenger $\mathcal{C}$ chooses $b \in \{0, 1\}$ and a tag $id \in \{0, 1\}^n$ and signs the vector space V_b with a tag id. Then, $\mathcal{C}$ uses $\textbf{LinCom}(pk, id, \{(g_j, \sigma_j)\}_{j=1}^{l})$ algorithm to derive signatures σ_j on functions $f_j\left(\mathbf{v}_1^{(b)}, \mathbf{v}_2^{(b)}, \cdots, \mathbf{v}_k^{(b)}\right)$ for all $j = 1, 2, \cdots, s$ and gives them to $\mathcal{A}$. The function can be out adaptively after choosing V_0 and V_1.

Output :

$\mathcal{A}$ outputs a bit b'.

The adversary $\mathcal{A}$ wins the game if $b = b'$.

4.2 Security Analysis

To prove the security requirements of our signature, we show *multi-unforgeability* and *weakly context hiding* property of our construction in **Theorems 2** and **3**. We prove multi-unforgeability of our multi-signature with linear homomorphism over $\mathbb{F}_2$ in the random oracle model. From an adversary that forges the signature of the proposed signature over $\mathbb{Z}_{2q}$, we make an adversary that solves the k-SIS problem over $\mathbb{Z}_q$.

Theorem 2. *For the proposed signature $\mathcal{MLH}$, suppose that $m = \lceil 6n \log 2q \rceil$ and $\gamma = 30\sqrt{n \log 2q} \log n$. Let $\beta = L \cdot \gamma\sqrt{2m}$. Then the proposed signature is multi-unforgeable in the random oracle model assuming that k-SIS$_{q,2m,\beta,\gamma}$ problem is infeasible and weakly context hiding property holds for our signature.*

Proof. Let $\mathcal{A}$ be an adversary that has the advantage ϵ_0 of the challenge-response game in **Definition 4**. We show the advantage of this game played by $\mathcal{A}$ is negligible on n assuming that k-SIS$_{q,2m,\beta,\gamma}$ problem is infeasible.

Game 0. Game 0 is equal to the challenge-response game in **Definition 4** for some g_c.

Game 1. Let Game 1 is equal to the challenge-response game in **Definition 1**. From the weakly context hiding property of our signature, given signatures $\sigma_1, \sigma_2, \cdots, \sigma_g$ for original vectors $\mathbf{v}_1, \mathbf{v}_2, \cdots, \mathbf{v}_g$, signatures do not reveal any information on partial signatures ρ_{ij} for each $\mathbf{v}_i$. They only depends on the vector $\mathbf{v} = \sum_{i=1}^{g} \mathbf{v}_i$ and the participants of signatures. Thus, finding a signature σ^* for a vector $\mathbf{v}^*$ in Game 0 is more difficult than finding a signature σ^* for a vector $\mathbf{v}^*$ in Game 1. The advantage ϵ_1 of Game 1 is greater than or equal to the advantage ϵ_0 of Game 0, *i.e.*, $\epsilon_1 \geq \epsilon_0$.

Game 2. Let Game 2 is a game to break k-SIS$_{q,2m,\beta,\gamma}$ problem. The probability ϵ_2 of Game 2 is greater than or equal to the advantage ϵ_1 of Game 1 by **Lemma 4**, *i.e.*, $\epsilon_2 \geq \epsilon_1$.

Thus, the adversary that wins Game 0 with non-negligible probability breaks k-SIS$_{q,2m,\beta,\gamma}$ problem with non-negligible probability. So, our signature is multi-unforgeable in the random oracle model assuming that k-SIS$_{q,2m,\beta,\gamma}$ problem is infeasible. □

Theorem 3. *For the proposed signature $\mathcal{MLH}$, suppose that $k < \dfrac{\log n}{2 \log \log n}$, $m = \lceil 6n \log 2q \rceil$ and $\gamma = 30\sqrt{n \log 2q} \log n$. Then, the proposed signature is weakly context hiding.*

Proof. In our signature, one obtains a signature σ on a linear combination $\mathbf{v}$ of vectors by calculating a linear combination of the signatures $\sigma_1, \sigma_2, \cdots, \sigma_g$ of the original vectors $\mathbf{v}_1, \mathbf{v}_2, \cdots, \mathbf{v}_g$ like Boneh and Freeman's signature. Thus, from **Lemma 5**, a signature σ_{lin} on a linear combination vector $\mathbf{v}_{lin}$ of vectors by summing up signatures $\sigma_1, \sigma_2, \cdots, \sigma_g$ of vectors $\mathbf{v}_1, \mathbf{v}_2, \cdots, \mathbf{v}_g$ does not leak information on the original signatures and our signature is weakly context hiding. □

5 Concluding Remark

We construct a novel lattice-based multi-signature with linear homomorphism over binary fields by generalizing linearly homomorphic signature by Boneh and Freeman [5].

In Table 1, we compare our signature with previous lattice-based signatures with linear homomorphism. Although one needs to deal with multiple signers on a single message in the real world, Boneh and Freeman's original signature (BF11) does not satisfy this issue. Zhang *et al.* (ZYW12) [8] and Jing (J13) [9] suggested how to aggregate multiple signatures on multiple messages with each single signer having different secret key. Still, they do not suggest how to manage multiple signers on a single message. And we suggest how to manage multiple signatures on a single message with multiple signers with different secret key.

Our signature is based on k-SIS problem like Boneh and Freeman's signature and is proved to exhibit very similar security requirements and homomorphic property of Boneh and Freeman's signature.

One can implement our signature in the real world cloud systems which have both individual signers and

Table 1: Comparison of **LHS**

Scheme	BF11 [5]	ZYW12 [8]	J13 [9]	Ours
Hard Problem	k-SIS	**SIS**	**SIS**	k-SIS
Known Attack	**X**	O	**X**	**X**
Number of Secret Key	unique	**multiple**	**multiple**	**multiple**
Number of Signers	single	single	single	**multiple**

multiple signers. By merging our signature with any homomorphic encryption, one can obtain a more authentic cloud system by giving integrity. For example, we encrypt a set of messages $\{\mathbf{pt}_1, \mathbf{pt}_2, \cdots, \mathbf{pt}_g\}$ into $\{\mathbf{ct}_1, \mathbf{ct}_2, \cdots, \mathbf{ct}_g\}$ using homomorphic encryption and make a signature σ_i for each $\mathbf{ct}_i$ using our signature. Then, a signature $\sigma = \sum_{i=1}^{g} \sigma_i$ for a message $\mathbf{ct} = \sum_{i=1}^{g} \mathbf{ct}_i$ is valid and we check the integrity of the message before decrypting it.

Till now, our scheme is restricted because of its bothersome reset process when there are another member joining. Thus, as future work, it is mandatory to make our signature to be more adaptive and efficient so that the signature is applicable to the real world scenario. Also, finding the concrete value of g_c in **Definition 4** is still an open problem. It is challenging to make other group-oriented signature with linear homomorphism such as linearly homomorphic group signatures and linearly homomorphic ring signatures or to extend the group-oriented signature with fully homomorphic property.

Acknowledgement

This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIP) (No. NRF-2015R1A2A2A01006812).

References

[1] C. Gentry, "Fully homomorphic encryption using ideal lattices," in *Proceedings of the Forty-First Annual ACM on Symposium on Theory of Computing*, pp. 169–178, ACM, 2009.

[2] Z. Brakerski and V. Vaikuntanathan, "Efficient fully homomorphic encryption from (standard) lwe," *SIAM Journal on Computing*, vol. 43, no. 2, pp. 831–871, 2014.

[3] C. Gentry, A. Sahai, and B. Waters, "Homomorphic encryption from learning with errors: Conceptually-simpler, asymptotically-faster, attribute-based," in *Advances in Cryptology–CRYPTO 2013*, pp. 75–92, Springer, 2013.

[4] S. Garg, C. Gentry, and S. Halevi, "Candidate multilinear maps from ideal lattices," in *Advances in Cryptology–EUROCRYPT 2013*, vol. 7881, pp. 1–17, Springer, 2013.

[5] D. Boneh and D. M. Freeman, "Linearly homomorphic signatures over binary fields and new tools for lattice-based signatures," in *Public Key Cryptography–PKC 2011*, pp. 1–16, Springer, 2011.

[6] D. Boneh and D. M. Freeman, "Homomorphic signatures for polynomial functions," in *Advances in Cryptology–EUROCRYPT 2011*, pp. 149–168, Springer, 2011.

[7] S. Gorbunov, V. Vaikuntanathan, and D. Wichs, "Leveled fully homomorphic signatures from standard lattices," in *Proceedings of the Forty-Seventh Annual ACM on Symposium on Theory of Computing*, pp. 469–477, 2015.

[8] P. Zhang, J. Yu, and T. Wang, "A homomorphic aggregate signature scheme based on lattice," *Chinese Journal of Electronics*, vol. 21, no. 4, pp. 701–704, 2012.

[9] Z. Jing, "An efficient homomorphic aggregate signature scheme based on lattice," *Mathematical Problems in Engineering*, 2014. Avaliable online at `http://dx.doi.org/10.1155/2014/536527`.

[10] S. D. Gordon, J. Katz, and V. Vaikuntanathan, "A group signature scheme from lattice assumptions," in *Advances in Cryptology–ASIACRYPT 2010*, pp. 395–412, Springer, 2010.

[11] T. Feng, Y. Gao, and J. Ma, "Changeable threshold signature scheme based on lattice theory," in *E-Business and E-Government, 2010 International Conference on*, pp. 1311–1315, IEEE, 2010.

[12] P.-L. Cayrel, R. Lindner, M. Rückert, and R. Silva, "A lattice-based threshold ring signature scheme," in *Progress in Cryptology–LATINCRYPT 2010*, pp. 255–272, Springer, 2010.

[13] R. Bendlin, S. Krehbiel, and C. Peikert, "How to share a lattice trapdoor: Threshold protocols for signatures and (h) ibe," in *Applied Cryptography and Network Security*, pp. 218–236, Springer, 2013.

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

[15] J. Alwen and C. Peikert, "Generating shorter bases for hard random lattices," *Theory of Computing Systems*, vol. 48, no. 3, pp. 535–553, 2011.

[16] D. Cash, D. Hofheinz, E. Kiltz, and C. Peikert, "Bonsai trees, or how to delegate a lattice basis," *Journal of Cryptology*, vol. 25, no. 4, pp. 601–639, 2012.