# A Generic Approach to Identity-based Sequential Aggregate Signatures: New constructions from 2-level HIBE Schemes

Yanqing Yao[a,**], Hua Guo[a,c], Zhoujun Li[a,b,*]

[a] *State Key Laboratory of Software Development Environment, Beihang University, Beijing 100191, China*

[b] *Beijing Key Laboratory of Network Technology, Beihang University, Beijing, China*

[c] *State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093, China*

## Abstract

Identity-based sequential aggregate signature (IBSAS) schemes are usually applied to secure network routing and sensor networks, since they allow multiple signers to sequentially produce a short signature of different messages to reduce bandwidth overhead and storage space for signatures, and allow signers to attest to these messages as well as the order in which they signed using their identities. In CCS'07, Boldyreva et al. introduced this concept and constructed the first IBSAS scheme in the random oracle model. After that, a couple of IBSAS schemes are proposed and proved. Unfortunately, none of them is constructed based on a standard computational problem and secure in the standard model (i.e., without random oracles). How to construct this kind of scheme is still an open problem. In this paper, we propose a generic construction of IBSAS schemes by employing 2-level Hierarchical Identity-based Encryption Schemes, and then prove its security in the security model proposed by Boldyreva et al. in CCS' 07. Afterwards, we instantiate the generic construction to obtain a concrete IBSAS scheme secure under the Computational Diffie-Hellman (CDH) assumption in the standard model, thus solving the above open problem. An extra fruit of our generic construction is that it can be used to construct the first lattice-based

[*] Corresponding authors.
*Email addresses:* yaoyanqing1984@buaa.edu.cn, lizj@buaa.edu.cn.

[**] Most of the work done when the author visited New York University.

IBSAS scheme, which is secure in the random oracle model. Finally, we show the performance comparisons between our schemes and previous ones.

*Keywords:*
Identity-based Signatures; Sequential Aggregate Signatures; the CDH Assumption; the Learning with Errors Problem; Network Security; Sensor Networks

## 1. Introduction

Aggregate signature schemes, introduced by Boneh et al. [10] in 2003, are digital signatures that allow multiple signers to sign different messages while keeping the total signature size constant. One of the main concerns is to find an effective method of compressing a list of signatures to reduce bandwidth overhead and storage space for signatures. Such schemes can be applied to the secure border gateway protocol (BGP) in network routing and sensor networks [4, 19, 35]. In BGP, routers generate and forward route attestations to other routers to advertise the routes which should be used to reach their networks. Secure BGP solves the problem of attestation forgery by having each router add its signature to a valid attestation before forwarding it to its neighbors. Because of the size of route attestations is limited, aggregate signatures are useful in reducing the overhead of multiple signatures along a path. Nicol et al. [28] gave a detailed analysis of the application of aggregate signatures to the secure BGP routing protocol [22].

For applications such as compressing certificate chains [10], the ability to combine preexisting individual signatures into an aggregate is unnecessary. Each user, when producing a signature, is aware of the signatures above his in the chain. Thus aggregation for certificate chains should be performed incrementally and sequentially. The formal definition of sequential aggregate signature was introduced by Lysyanskaya et al. [25] in 2004, which is constructed sequentially, with each signer modifying the aggregate-so-far signature in turn. Roughly speaking, the sequential aggregate signing algorithm takes as input a private key $SK_i$, a message $M_i$ and a sequential aggregate $\sigma'$ to sign, where $\sigma'$ is the signature of the ordered messages $M_1$, $M_2$, …, $M_{i-1}$ under the ordered public keys $PK_1$, $PK_2$, …, $PK_{i-1}$. All of $M_1$, $M_2$, …, $M_{i-1}$ and $PK_1$, $PK_2$, …, $PK_{i-1}$ must also be provided as input of the sequential aggregate signing algorithm. Then the algorithm outputs a sequential aggregate signature $\sigma$ on all $i$ messages $M_1$, $M_2$, …, $M_i$.

Recently, sequential aggregate signature schemes have aroused great interest. Lysyanskaya et al. [25] constructed a sequential aggregate signature scheme based on RSA in the random oracle model. The first sequential aggregate signature scheme in the standard model, namely without random oracles, was proposed by Lu et al. [24]. They converted the Waters' signature scheme [34] to the sequential aggregate signature scheme, and proved its security under the well known Computational Diffie-Hellman (CDH) assumption (see Definition 1). In 2011, Schr$\ddot{o}$der [32] constructed a sequential aggregate signature with short public keys using the Camenisch-Lysyanskaya signature scheme, but the security is only proven under the interactive LRSW assumption [26](see Appendix B), which can be considered as a relaxed notion of [12]. Recently, Lee et al. [23] proposed the first sequential aggregate signature scheme with short public keys (i.e., a constant number of group elements) in prime order (asymmetric) bilinear groups, which is secure under static assumptions in the standard model. One common limitation of the above schemes is that they are constructed based on concrete hardness assumptions.

Unfortunately, using public-key-based sequential aggregate signature schemes that necessitate a public-key infrastructure (PKI) dramatically increases setup and storage or bandwidth overhead of secure networking protocols. Namely, in routing-based network applications, such schemes will either (a) incur the setup and storage overhead of distributing the public keys and corresponding certificates of all users out-of-band, and participating routers storing them indefinitely; or (b) defeat the aim of minimizing bandwidth if public keys (which cannot be aggregated) and certificates of the signer in each signature would always have to be sent along with the latter for verification.

If the public key of a user is self-evidently associated with the user's identity information (e.g., name, IP address, email address, phone number), then in essence there is no need to certificate the public key. Identity-based cryptosystems were introduced by Shamir [33] in 1984 to achieve it. In an identity-based cryptosystem, the key generation process includes two algorithms: (1) Setup algorithm, which generates the system public parameters and a master secret key from a security parameter, (2) Key Generation algorithm, which generates the user's private key from the master secret key and user's identity information (more formally, public key). As users' private keys are generated by the master secret key, they have to trust the master absolutely. Therefore, identity-based cryptosystems are only suitable for the scenario where an unconditional trust is acceptable.

Boldyreva et al. [4] treated sequential aggregate signatures in the identity-based setting and thus introduced the concept of identity-based sequential aggregate signature (IBSAS) schemes (see Section 2.2). Further, under an interactive pairings-based assumption, they constructed an IBSAS scheme in the random oracle model. Meanwhile, they raised an open problem, i.e., how to construct a secure IBSAS scheme under standard computational hardness assumption in the standard model. However, Hwang et al. [21] pointed out that the pairing assumption used in [4] is not intractable, and they showed a forgery attack on the corresponding IBSAS scheme of [4]. After that, Boldyreva et al. [5] proposed a new IBSAS scheme by modifying their previous construction and proved its security in the generic group model. Recently, Gerbush et al. [16] proved the security of the modified IBSAS scheme of Boldyreva et al. [4] under static assumptions via dual form signatures framework. In 2012, Dou et al. [14] presented an IBSAS scheme, which is based on RSA instead of pairings. Although a few of elegant schemes have been proposed, none of them is based on a standard computational problem and secure in the standard model.

As an interesting observation, Moni Naor observed that an Identity-Based Encryption (IBE) scheme can be immediately converted into a public key signature scheme [9]. The intuition is below. The public key in the signature scheme is the system public parameters for the IBE scheme. The signer's private key is the master key in the IBE scheme. The signature on a message $M$ is the IBE decryption key for $id = M$. To verify a signature, choose a random message $M_r$, encrypt $M_r$ using the public key $id = M$, and then attempt to decrypt using the given signature on $M$ as the decryption key. If the ciphertext decrypts correctly, the signature is considered valid. Gentry et al. [18] noted that an Identity-Based Signature (IBS) scheme can be constructed in a very similar way via replacing the IBE scheme with a Hierarchical Identity-Based Encryption (HIBE) scheme. This technique was used by Boneh et al. [7, 11] to construct short signatures, and likewise by Paterson et al. [29] to construct specific identity-based signatures in the standard model. Unfortunately, it seems impossible to borrow the idea of Gentry et al. [18] directly to construct IBSAS schemes, since IBSAS schemes require signatures be aggregated one-by-one while keeping total signature size constant. In spite of this bottleneck, it's very meaningful to give a generic method via converting an HIBE scheme, as there have been many fruits about IBE schemes.

Note that Galindo et al. [15] studied whether there is a generic construction of "identity-based signature schemes with additional properties" (such as aggregate signatures, identity-based blind signatures, ...) from standard signature schemes with the same properties. In particular, they concluded that a secure identity-based aggregate signature scheme can be constructed from a secure aggregate signature scheme, where the length of an identity-based aggregate signature depends on the number of different signers. However, compared with IBE schemes, known aggregate signature schemes are much less. One main goal of our paper is to find a general construction of IBSAS schemes from HIBE schemes.

### *1.1. Our Contribution and Techniques*

Motivated by the open problem of Boldyreva et al. [4, 5], i.e., constructing an IBSAS scheme which is based on a standard computational problem and secure in the standard model, we propose a new generic construction of IBSAS schemes from 2-level HIBE schemes, which is different from the counterpart of [15].

We start with borrowing some ideas of constructing an IBS scheme from a 2-level HIBE scheme as follows [18]. The public key is the system parameters of the 2-level HIBE scheme. The private key of identity $id$ is the counterpart in the 2-level HIBE scheme. The signature on message $M$ under identity $id$ is then the private key of the identity tuple $(id, M)$ in the 2-level HIBE scheme. And verification is performed by selecting a random message $M_r$, encrypting $M_r$ with the identity tuple $(id, M)$, and verifying that $M_r$ is the decryption of the ciphertext via using the given signature as the decryption key. From this construction, of course we can get a signature $\sigma_1$ on the message $M_1$ under identity $id_1$. The difficulty is how to obtain the signature given the identity-message pairs $((id_1, M_1), (id_2, M_2))$ and an aggregate-so-far signature $\sigma_1$ without increasing the resulting signature size. To achieve this purpose, we consider the string $M_2||M_1||id_1$ as a new message, then we can sign this message under identity $id_2$ after verifying that the signature $\sigma_1$ on message $M_1$ under identity $id_1$ is valid. Similarly, we can gain the signature given the identity-message pairs $((id_1, M_1), (id_2, M_2), \ldots, (id_i, M_i))$ and an aggregate-so-far signature $\sigma_{i-1}$. Furthermore, we obtain the following general result, which shows that the security of the IBSAS scheme relies on the security of the corresponding 2-level HIBE scheme.

**General Result.** *The IBSAS scheme is existentially unforgeable against adaptively chosen identity-message pairs attack if the corresponding 2-level HIBE scheme is IND-ID-CPA secure where the messages are chosen from* $\{0,1\}^*$ *and the identities are chosen from* $\{0,1\}^* \cup (\{0,1\}^* \times \{0,1\}^*)$.

Based on this result, we illustrate it with two concrete examples below.

- We present a concrete IBSAS scheme, which is secure under the CDH assumption (see Definition 1) in the standard model. It's constructed from a 2-level HIBE scheme, which is a natural extension of Waters' IBE scheme [34]. Since the CDH assumption is a standard cryptographic assumption [30], we proposes a solution to the open problem pointed out by Boldyreva et al. [5].
- We construct a lattice-based IBSAS scheme, which is secure under the hardness of Learning with Errors (LWE) problem (see Definition 2) in the random oracle model. It's constructed from the HIBE scheme of Agrawal et al. [2]. Note that this is the first lattice-based IBSAS scheme. It's well known that lattice-based cryptosystems have recently acquired much importance because of their probable security against quantum computing attacks, average-case to worst-case equivalence as well as simplicity and potential efficiency.

### *1.2. Organization*

The rest of the paper is organized as follows. In the following Section, we recall some notation and concepts to be used in the paper. In Section 3, we present how to construct an IBSAS scheme via converting a 2-level HIBE scheme and prove its existential unforgeability against adaptively chosen identity-message pairs attack. In Section 4, we design two concrete IBSAS schemes based on the CDH assumption and LWE assumption, and prove their security. Then in Section 5, we show the performance comparisons between our schemes and other existing ones. Section 6 concludes the paper.

## 2. Preliminaries

### *2.1. Notation and Hardness Assumptions*

For a positive integer $n$, $[n]$ denotes the set $\{1, 2, \ldots, n\}$. For $x \in \mathbb{R}$, $\lfloor x \rceil$ denotes the closest integer to $x$. For any ordered set of linearly independent (column) vectors $\mathbf{S} = \{\mathbf{s}_1, \ldots, \mathbf{s}_k\}$ in $\mathbb{R}^m$, the norm of $\mathbf{S}$ is defined by $\|\mathbf{S}\| \overset{def}{=} \max_{j\in[k]} \|\mathbf{s}_j\|$, and denote $\tilde{\mathbf{S}}$ as the Gram-Schmidt orthogonalization of $\mathbf{S}$. For any $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ and $\mathbf{u} \in \mathbb{Z}_q^n$, denote $\Lambda_q^{\perp}(\mathbf{A}) \overset{def}{=} \{\mathbf{x} \in \mathbb{Z}^m : \mathbf{A}\cdot\mathbf{x} = \mathbf{0}\ mod\ q\}$ and $\Lambda_q^{\mathbf{u}}(\mathbf{A}) \overset{def}{=} \{\mathbf{x} \in \mathbb{Z}^m : \mathbf{A}\cdot\mathbf{x} = \mathbf{u}\ mod\ q\}$. If $D$ is a set, then let $d \overset{\$}{\leftarrow} D$ represent sampling $d$ according to the uniform distribution over the set $D$.

For an $\alpha \in (0,1)$ and a prime $q$, let $\overline{\Psi}_\alpha$ be the distribution over $\mathbb{Z}_q$ of the random variable $\lfloor qX \rceil\ mod\ q$ where $X$ is a normal random variable with mean 0 and standard deviation $\alpha/\sqrt{2\pi}$. Denote $s_R \overset{def}{=} \sqrt{n\log q}\cdot\omega(\sqrt{\log m})$. Let $\mathcal{D}_{\mathbb{Z}^m,s_R}$ be the discrete Gaussian distribution over $\mathbb{Z}^m$ with center $\mathbf{0}$ and parameter $s_R$, then we have $\mathcal{D}_{\mathbb{Z}^m,s_R}(\mathbf{y}) \overset{def}{=} \frac{\exp(-\pi\|\mathbf{y}\|^2/s_R^2)}{\sum\limits_{\mathbf{x}\in\mathbb{Z}^m}\exp(-\pi\|\mathbf{x}\|^2/s_R^2)}$ for all $\mathbf{y} \in \mathbb{Z}^m$. We say that a matrix $R \in \mathbb{Z}^{m\times m}$ is $\mathbb{Z}_q$-invertible if $R\ mod\ q$ is invertible as a matrix in $\mathbb{Z}_q^{m\times m}$. Denote $\mathcal{D}_{m\times m}$ as the distribution on matrices in $\mathbb{Z}^{m\times m}$ defined as $(\mathcal{D}_{\mathbb{Z}^m,s_R})^m$ conditioned on the resulting matrix being $\mathbb{Z}_q$-invertible.

**Definition 1.** (see [27]) Consider a cyclic group $\mathbb{G}$ of order $q$. The Computational Diffie-Hellman (CDH) assumption states that, given $(g, g^a, g^b)$ for a randomly chosen generator $g$ and $a, b \overset{\$}{\leftarrow} \{0, 1, \ldots, q-1\}$, it is computationally intractable to compute the value $g^{ab}$.

**Definition 2.** (see [31]) Consider a prime $q$, a positive integer $n$, and a distribution $\lambda$ over $\mathbb{Z}_q$. For a vector $\mathbf{s} \in \mathbb{Z}_q^n$, denote $A_{\mathbf{s},\lambda}$ as the distribution over $\mathbb{Z}_q^n \times \mathbb{Z}_q$ obtained by choosing a uniformly random vector $\mathbf{a}$ from $\mathbb{Z}_q^n$ and outputting $(\mathbf{a}, \mathbf{a}^T\mathbf{s} + b)$ where $b$ is a sample drawn from the distribution $\lambda$. The $(\mathbb{Z}_q, n, \lambda)-LWE$ (Learning with Errors) problem is to distinguish, given oracle access to any desired $m = poly(n)$ samples, between the distribution $A_{\mathbf{s},\lambda}$ (for constant random secret key $\mathbf{s} \in \mathbb{Z}_q^n$) and the uniform distribution over $\mathbb{Z}_q^n \times \mathbb{Z}_q$.

Regev [31] showed that if there exists an efficient, possibly quantum, algorithm for solving the $(\mathbb{Z}_q, n, \overline{\Psi}_\alpha)-$LWE problem where $\alpha \in (0,1)$ and $q > 2\sqrt{n}/\alpha$, then there exists an efficient quantum algorithm for approximating the shortest independent vectors problem (SIVP) and the decision version of the shortest vector problem (GAPSVP), to within $\tilde{O}(n/\alpha)$ factors in the $l_2$ norm.

### *2.2. The Formal Definition of IBSAS Schemes and Their Security*

Now we recall the formal definition of IBSAS schemes and their security.

An identity-based sequential aggregate signature (IBSAS) scheme [4] consists of the following algorithms:

- **Setup**: This probabilistic algorithm initially runs by the trusted private-key generator (PKG) that takes as input a security parameter $n$ and generates the public parameters $PP$ and master secret key $msk$.
- **Extract**: This probabilistic algorithm runs by the PKG that takes as input $PP$, $msk$, and a user's identity $id \in \{0,1\}^*$, then outputs the private key $sk_{id}$ for user $id$ (we identify user with his $id$).
- **Sign**: This probabilistic algorithm runs by a user $id$ that takes as input its private key $sk_{id}$, a message $M \in \{0,1\}^*$, a list $((id_1, M_1), \ldots, (id_{i-1}, M_{i-1}))$ of identity-message pairs, and an aggregate-so-far $\sigma$, and returns a new aggregate signature $\sigma'$, or $\perp$ to indicate that the input was invalid.
- **Vf**: This deterministic algorithm takes as input the master public key $mpk$, a list $((id_1, M_1), \ldots, (id_j, M_j))$ of identity-message pairs and an IBSAS $\sigma$, and outputs 1 if $\sigma$ is a valid identity-based aggregate signature on identity-message pairs $((id_1, M_1), \ldots, (id_j, M_j))$, or 0 otherwise.

Correctness of the scheme demands that $\Pr(\mathbf{Vf}(L_j, \sigma_j) = 1) = 1 - negl(n)$, for all $j \in \mathbb{N}$ and all $\{(id_i, M_i) | 1 \leq i \leq j, id_i \in \{0,1\}^*, M_i \in \{0,1\}^*\}$, where the probability is over the following experiment.

Experiment:

```
$(mpk, msk) \xleftarrow{\$} \mathbf{Setup}$
For all $i = 1, \ldots, j$ do
    $sk_{id_i} \xleftarrow{\$} \mathbf{Extract}(msk, id_i)$
$\sigma_0, L_0 \leftarrow \emptyset$
For all $i = 1, \ldots, j$ do
    $\sigma_i \xleftarrow{\$} \mathbf{Sign}(sk_{id_i}, M_i, L_{i-1}, \sigma_{i-1})$
    $L_i \leftarrow ((id_1, M_1), \ldots, (id_i, M_i))$.
```

In the following, we present the security model of IBSAS schemes introduced by Boldyreva et al. [4].

**Definition 3.** Let $\mathbf{AS} = (\mathbf{Setup}, \mathbf{Extract}, \mathbf{Sign}, \mathbf{Vf})$ be an IBSAS scheme. We consider an experiment with a forger (i.e. adversary) $\mathcal{F}$ assumed to be a probabilistic Turing machine taking as input a security parameter $n$ and running in three stages.

- **Setup**: The experiment first takes as input a security parameter $n$ and generates the public parameters $PP$ and master secret key $msk$.
- **Attack**: $\mathcal{F}$ runs on input $PP$ with access to private key extraction oracle $\mathcal{O}_{\mathbf{Extract}}(msk, \cdot)$ and signing oracle $\mathcal{O}_{\mathbf{Sign}}(\cdot,\cdot,\cdot,\cdot)$. The first operates according to the above definition of the private-key derivation algorithm for IBSAS. The second on input an identity $id_i$, a message $M_i$, a list of identity-message pairs $L_{i-1} = ((id_1, M_1), (id_2, M_2), \dots, (id_{i-1}, M_{i-1}))$, and an aggregate-so-far $\sigma_{i-1}$, sets $sk_{id_i} \leftarrow \mathcal{O}_{\mathsf{Extract}}(msk, id_i)$ and returns $\mathbf{Sign}(sk_{id_i}, M_i, L_{i-1}, \sigma_{i-1})$.
- **Forgery**: Eventually, $\mathcal{F}$ halts with outputting a list of identity-message pairs $L^* = ((id_1^*, M_1^*), (id_2^*, M_2^*), \dots, (id_j^*, M_j^*))$ and a purported aggregate signature $\sigma^*$. This output is considered to be a forgery if (1) all of $id_1^*, \dots, id_j^*$ are distinct, (2) $\mathbf{Vf}(L^*, \sigma^*) = 1$ and (3) There exists some $k^* \in \{1, \dots, j\}$ such that $id_{k^*}^*$ was not queried by $\mathcal{F}$ to its private key extraction oracle and $\mathcal{F}$ did not query $(id_{k^*}^*, M_{k^*}^*, ((id_1^*, M_1^*), (id_2^*, M_2^*), \dots, (id_l^*, M_l^*)), \sigma')$ to oracle $\mathcal{O}_{\mathbf{Sign}}(\cdot,\cdot,\cdot,\cdot)$ for any $\sigma' \in \{0,1\}^*$ and any $l \in \mathbb{N}$.

We define $\mathcal{F}'s$ success probability by

$$IBSAS - Adv^{UF}(\mathcal{F}) = \Pr(\mathbf{Vf}(((id_1^*, M_1^*), \dots, (id_j^*, M_j^*)), \sigma^*) = 1).$$

The IBSAS scheme is said to be existentially unforgeable against adaptively chosen identity-message pairs attack if $IBSAS - Adv^{UF}(\mathcal{F})$ is negligible in the security parameter $n$.

### *2.3. 2-level HIBE Schemes and Their Security*

Our new generic construction of IBSAS schemes is based on 2-level HIBE schemes. Recall that a 2-level hierarchical identity-based encryption (HIBE) scheme $\prod = $ (**Setup**, $\{\mathbf{KeyGen}_i\}_{i=1,2}$, **Encrypt**, **Decrypt**) [20] is a tuple of polynomial-time algorithms as follows.

- **Setup**: Input: a security parameter $n$. Output: the system parameters $PP$ and master secret key $msk$.

- **KeyGen$_1$**: Input: $PP$, $msk$ and the identity $id \in \{0,1\}^*$. Output: the private key $sk_{id}$ of identity $id$. (It's called the first level private key generation oracle.)

- **KeyGen$_2$**: Input: $PP$, $sk_{id}$ and the identity $(id, id')$ where $id, id' \in \{0,1\}^*$. Output: the private key $sk_{(id,id')}$ of identity $(id, id')$. (It's called the second level private key generation oracle.)

- **Encrypt**: Input: $PP$, identity $id$ (or $(id, id')$) and a message $M \in \mathcal{M}$. Output: the corresponding ciphertext.

- **Decrypt**: Input: $PP$, identity $id$ (or $(id, id')$), a ciphertext, and a private key $sk_{id}$ (or $sk_{(id,id')}$). Output: the corresponding plaintext.

**Definition 4.** The 2-level HIBE security (or indistinguishability from random) under adaptive chosen-identity and chosen-plaintext attack (IND-ID-CPA) (see [8]) is defined using the following game between a challenger and an adversary:

Let $\prod = (\textbf{Setup}, \{\textbf{KeyGen}_i\}_{i=1,2}, \textbf{Encrypt}, \textbf{Decrypt})$ be a 2-level HIBE scheme. Consider the following experiment:

- **Setup**: The challenger runs **Setup** and gives the adversary the resulting public system parameters $PP$. It keeps the master secret key $msk$ to itself.

- **Phase 1**: The adversary issues any number of private key generation queries adaptively. For each first level private key generation query, the adversary submits an identity $id$ and is told $sk_{id}$. For each second level private key generation query, the adversary submits an identity $id'$ and the "child identity" $(id', id'')$ and obtains $sk_{(id',id'')}$.

- **Challenge**: Once the adversary decides that Phase 1 is over, it chooses an arbitrary target identity $id^*$. The restriction is that it did not issue a private-key query for $id^*$ or a prefix of $id^*$ during phase 1. It outputs two equal length plaintexts $M_0, M_1 \in \mathcal{M}$ on which it wishes to be challenged. The challenger picks a random bit $b \in \{0,1\}$ and computes the challenge ciphertext $C = \textbf{Encrypt}(PP, id^*, M_b)$. It sends $C$ as the challenge to the adversary.

- **Phase 2**: The adversary issues additional private key generation queries for identities adaptively, except that it now may not ask for identity $id^*$ or the identity as a prefix of $id^*$. The challenger responds as in Phase 1.

- **Guess**: Eventually, the adversary outputs a guess $b' \in \{0, 1\}$. The adversary wins if $b = b'$.

We refer to such an adversary $\mathcal{A}$ as an IND-ID-CPA adversary and define its advantage in attacking the scheme $\mathcal{S} = ($**Setup**, $\{$**KeyGen**$_i\}_{i=1,2}$, **Encrypt**, **Decrypt**$)$ as $Adv_{\mathcal{S}}(\mathcal{A}) = |\Pr(b = b') - \frac{1}{2}|$.

Since there exist much more HIBE schemes than IBSAS schemes, new IBSAS schemes can be constructed from exiting HIBE schemes.

## 3. The Generic Construction of IBSAS Schemes

In this section, we firstly present a generic construction of IBSAS schemes via converting 2-level HIBE schemes, and then give a rigorous proof of its security. Since HIBE schemes have been widely studied, we'll use HIBE schemes to construct IBSAS Schemes. Though an IBE scheme can be immediately converted into a public key signature scheme [9] and an IBS scheme can be constructed in a very similar way via replacing the IBE scheme with an HIBE scheme [18], it seems impossible to employ the above method to construct an IBSAS scheme from an HIBE scheme, since IBSAS schemes require signatures be aggregated one-by-one while keeping total signature size constant.

### *3.1. The Generic Construction of IBSAS Schemes*

For simplicity, for all positive integer $i > 1$, $L_{i-1} \overset{def}{=} ((id_1, M_1), (id_2, M_2), \ldots, (id_{i-1}, M_{i-1}))$, and $\widetilde{L_{i-1}} \overset{def}{=} M_1||id_1||M_2||id_2||\ldots M_{i-1}||id_{i-1}$ as a concatenation of identity-message pairs in the remainder of the paper.

*Given a 2-level HIBE scheme, we build an Identity-Based Sequential Aggregate Signature (IBSAS) scheme $\mathcal{S}$=(***Setup***,* ***Extract***,* ***Sign***,* ***Vf****) as follows. The system parameters $PP$ and master secret key $msk$ are the same as the counterparts of the 2-level HIBE scheme. The private key of an identity $id$ is just that of the first level identity in the 2-level HIBE scheme. To reduce the signature size, the signature of the current message $M_i$, given previous* *identity-message pairs* $((id_1, M_1), (id_2, M_2), \ldots, (id_{i-1}, M_{i-1}))$ *and current identity* $id_i$*, is the private key of the second level identity* $(id_i, M_i||\widetilde{L_{i-1}})$ *in the 2-level HIBE scheme. The verification runs via selecting a random message* $m_r$*, encrypting* $m_r$ *with the identity* $(id_i, M_i||\widetilde{L_{i-1}})$*, and verifying that* $m_r$ *is the decryption of the ciphertext via using the given signature as the decryption key. The detailed description is shown as follows.*

- **Setup**$(1^n)$: The step **Setup** is the same as **Setup** of the 2-level HIBE scheme in Section 2.3.
- **Extract**$(PP, msk, id)$: The step **Extract** is the same as $\mathbf{KeyGen}_1$ of the 2-level HIBE scheme in Section 2.3.
- **Sign**$(sk_{id_i},\ M_i,\ L_{i-1},\ \sigma_{i-1})$: The algorithm first checks that $\sigma_{i-1}$ is a valid aggregate signature (according to the verification algorithm below) and returns $\perp$ if not. (This step is skipped for a first signer.) If so, let $id'_i = (id_i, M_i||\widetilde{L_{i-1}})$ and compute $\sigma_i = sk_{id'_i} = \mathbf{KeyGen}_2(sk_{id_i}, id'_i)$.
- **Vf**$(PP,\ L_i,\ \sigma_i)$: The algorithm first checks that all of $id_1, id_2, \ldots, id_i$ are distinct and outputs 0 if not. Let $id'_i = (id_i, M_i||\widetilde{L_{i-1}})$, $M' \xleftarrow{\$} \mathcal{M}$. Compute $C' = \mathbf{Encrypt}(PP, id'_i, M')$. If $\mathbf{Decrypt}(PP, sk_{id'_i} = \sigma_i, C') = M'$, then the algorithm returns 1; else it returns 0.

It can be easily checked that the above scheme satisfies the correctness.

### *3.2. The Security of the Generic Construction*

Now we prove existential unforgeability of our IBSAS scheme under the assumption that the original 2-level HIBE scheme is IND-ID-CPA secure. Given an adversary that breaks the IBSAS scheme, we construct an adversary that simulates the IBSAS scheme. By Definition 4, this adversary can in turn successfully break the 2-level HIBE scheme. An algorithm is introduced to overcome the difficulty of obtaining the target identity in the **Challenge** step of the IND-ID-CPA game. We now give the theorem as follows.

**Theorem 1.** *The IBSAS scheme is existentially unforgeable against adaptively chosen identity-message pairs attack if the corresponding 2-level HIBE scheme is IND-ID-CPA secure where the messages are chosen from* $\{0,1\}^*$ *and the identities are chosen from* $\{0,1\}^* \cup (\{0,1\}^* \times \{0,1\}^*)$.

*Proof.* Let $\mathcal{F}$ be an adversary as in Definition 3. We construct simulator $\mathcal{F}'$ as in Definition 4 against the 2-level HIBE scheme. Assume $\mathcal{F}'$ maintains three lists in its local storage, called $E-list$, $V-list$, and $S-list$. They are set to empty initially. Simulator $\mathcal{F}'$ interacts with $\mathcal{F}$ as follows.

**Setup**: Simulator $\mathcal{F}'$ runs $\mathcal{F}$ supplying it with the public parameters.

**Private Key Extraction Queries**: Considering a query for the private key of an identity $id$, simulator $\mathcal{F}'$ asks its own first level private key generation oracle, returns the result $sk_{id}$ to $\mathcal{F}$, and puts $(id, sk_{id})$ into the $E-list$.

**Signing Queries**: Algorithm $\mathcal{F}$ requests a sequential aggregate signature by supplying an identity $id_i$, a message $M_i$, a list of identity-message pairs $L_{i-1} = ((id_1, M_1), (id_2, M_2), \ldots, (id_{i-1}, M_{i-1}))$, and an aggregate-so-far $\sigma_{i-1}$. $\mathcal{F}$ first checks whether $\sigma_{i-1}$ is a valid signature. If it's invalid, $\mathcal{F}$ outputs $\perp$. Otherwise, $\mathcal{F}'$ puts $(L_{i-1}, \sigma_{i-1})$ into the $V-list$, queries its own second level private key generation oracle with input $PP$ and $(id_i, M_i||\widetilde{L_{i-1}})$, sends the resulting private key $sk_{(id_i, M_i||\widetilde{L_{i-1}})}$ of identity $(id_i, M_i||\widetilde{L_{i-1}})$ to $\mathcal{F}$, and puts $(L_i, sk_{(id_i, M_i||\widetilde{L_{i-1}})})$ into the $S-list$.

**Output**: Finally, $\mathcal{F}$ halts, outputting a list of identity-message pairs $L_j^* \overset{def}{=} ((id_1^*, M_1^*), (id_2^*, M_2^*), \ldots, (id_j^*, M_j^*))$ and a purported aggregate signature $\sigma_j^*$.

Assume that the output of $\mathcal{F}$ is a valid forgery, then there exists some $k^* \in \{1, 2, \ldots, j\}$ such that $id_{k^*}^*$ was not queried by $\mathcal{F}$ to its private key extraction oracle and $\mathcal{F}$ did not query $(id_{k^*}^*, M_{k^*}^*, ((id_1^*, M_1^*), (id_2^*, M_2^*), \ldots, (id_{k^*}^*, M_{k^*}^*)), \sigma')$ to oracle $\mathcal{O}_{\mathbf{Sign}}(\cdot, \cdot, \cdot, \cdot)$ for any $\sigma' \in \{0, 1\}^*$.

Algorithm 1:

```
$i \leftarrow j + 1$
repeat
    $i \leftarrow i - 1$
    if $(L_i^*, \sigma_i^*)$ is not in the $S-list$
        if $(id_i^*, sk_{id_i^*})$ is not in the $E-list$
            then  $k^* \leftarrow i$, output $k^*$ and $(L_{k^*}^*, \sigma_{k^*}^*)$, and halt
        else  we infer that $(L_{i-1}^*, \sigma_{i-1}^*)$ is in the $S-list$, thus $(L_{i-2}^*, \sigma_{i-2}^*)$
              is in the $V-list$. $i \leftarrow i - 1$
    else we infer that $(L_{i-1}^*, \sigma_{i-1}^*)$ is in the $V-list$.
```

For simplicity, denote $L_i^* \overset{def}{=} ((id_1^*, M_1^*), (id_2^*, M_2^*), \ldots, (id_i^*, M_i^*))$ for all $i \in \{1, 2, \ldots, j\}$. We propose an algorithm (Algorithm 1) to show how to find the corresponding $k^*$.

Since $(L_j^*, \sigma_j^*)$ is a valid forgery, the above algorithm must halt at a certain step. From the above simulation, we have that (1) $\sigma_{k^*}^*$ can be considered to be the private key of the "child" identity $id^* \overset{def}{=} (id_{k^*}^*, M_{k^*}^* || M_1^* || id_1^* || M_2^* || id_2^* || \ldots || M_{k^*-1}^* || id_{k^*-1}^*)$ in the 2-level HIBE scheme; (2) $id_{k^*}^*$ was not queried by $\mathcal{F}'$ to its own first level private key generation oracle; (3) $\mathcal{F}'$ did not query its own second level private key generation oracle with input $PP$, identity $id_{k^*}^*$, and the "child identity" $id^*$.

During the **Challenge** step of the IND-ID-CPA game, adversary $\mathcal{F}'$ chooses the target identity as $id^*$ and outputs two equal length plaintexts $M_0, M_1 \in \mathcal{M}$ on which it wishes to be challenged. The challenger picks a random bit $b \in \{0, 1\}$ and computes the challenge ciphertext $C = \mathsf{Encrypt}(PP, id^*, M_b)$. It sends $C$ as the challenge to the adversary $\mathcal{F}'$. Adversary $\mathcal{F}'$ skips the **Phase 2** step of the IND-ID-CPA game without doing any private key queries. Then adversary $\mathcal{F}'$ computes $\mathsf{Decrypt}(PP, sk_{id^*}, C) = \mathsf{Decrypt}(PP, \sigma_{k^*}^*, C) = M_b$. Thus adversary $\mathcal{F}'$ can successfully get $b' = b$.

Consequently, $\mathcal{F}'$ is successful whenever $\mathcal{F}$ is. Algorithm $\mathcal{F}'$ makes as many first level private key generation queries as $\mathcal{F}$ makes private key extraction queries. And $\mathcal{F}'$ makes as many second level private key generation queries as $\mathcal{F}$ makes signing queries.

## 4. Two Concrete IBSAS Schemes from the generic construction

In this section, two concrete IBSAS schemes are proposed by instantiating the generic contraction of IBSAS, the first one is secure in the standard model based on the hardness of the CDH problem, and the second one is the first lattice-based IBSAS scheme.

### *4.1. The First Concrete IBSAS Scheme Based on the CDH Assumption*

Now we present the first concrete IBSAS scheme, then prove its security in the standard model based on the hardness of the CDH problem, thus solving the open problem proposed by Boldyreva et al. [4, 5].

To construct an IBSAS scheme which allows identities and messages of arbitrary lengths, like [4, 5, 29], collision-resistant hash functions $H_1 : \{0, 1\}^* \rightarrow$

$\{0,1\}^l$ and $H_2 : \{0,1\}^* \rightarrow \{0,1\}^n$ can be defined and used to create identities and the concatenation of identities and messages with desired length respectively. The scheme is constructed as follows:

Scheme 1:

Denote $\mathcal{U}_{id} \subseteq \{1, 2, \ldots, l\}$ to be the set of indices such that for all $i \in \mathcal{U}_{id}$, $H_1(id)[i] = 1$, where $H_1(id)[i]$ is the $i$th bit of $H_1(id)$. Similarly, denote $\mathcal{V}_V \subseteq \{1, 2, \ldots, n\}$ to be the set of indices such that for all $k \in \mathcal{V}_V$, $H_2(V)[k] = 1$.

- **Setup**: Let $\mathbb{G}$ and $\mathbb{G}_T$ be two groups with prime order $p$ respectively, for which there exists an efficiently computable bilinear pairing: $\mathbf{e} : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$ (see Appendix A). Choose a random generator $g$ of $\mathbb{G}$, pick a secret $\alpha \xleftarrow{\$} \mathbb{Z}_p$, and compute $g_1 = g^{\alpha}$. Choose $g_2$ randomly from $\mathbb{G}$. Moreover, it chooses random values $u'$, $v'$ from $\mathbb{G}$. Let $\mathbf{U} = (u_1\, u_2 \ldots u_l)$ and $\mathbf{V} = (v_1\ v_2 \ldots v_n)$, whose elements are all chosen at random from $\mathbb{G}$. The public parameters are $PP = (\mathbb{G}, \mathbb{G}_T, g, g_1, g_2, u', v', \mathbf{U}, \mathbf{V})$. The master secret key is $msk = g_2^{\alpha}$.

- **Extract**: To construct the private key $sk_{id}$ of identity $id$, choose $r_{id} \xleftarrow{\$} \mathbb{Z}_p$ and compute $sk_{id} = (g_2^{\alpha}(u' \prod_{i \in \mathcal{U}_{id}} u_i)^{r_{id}}, g^{r_{id}})$.

- **Sign**: The input is $sk_{id_i}$, $M_i$, $L_{i-1} = ((id_1, M_1), (id_2, M_2), \ldots, (id_{i-1}, M_{i-1}))$, $\sigma_{i-1}$. For $i \geq 2$, the algorithm checks that $\sigma_{i-1} = (\sigma_{i-1}^{(1)}, \sigma_{i-1}^{(2)}, \sigma_{i-1}^{(3)})$ is a valid aggregate signature (according to the verification algorithm below) and returns $\perp$ if not. If so, denote $V_i = M_i || \widetilde{L_{i-1}}$ and choose $r_{V_i} \xleftarrow{\$} \mathbb{Z}_p$. $\sigma_i = (g_2^{\alpha} \cdot (u' \prod_{i \in \mathcal{U}_{id_i}} u_i)^{r_{id}} \cdot (v' \prod_{j \in \mathcal{V}_{V_i}} v_j)^{r_{V_i}}, g^{r_{id}}, g^{r_{V_i}})$.

- **Vf**: On input $PP$, $L_i$, $\sigma_i = (\sigma_i^{(1)}, \sigma_i^{(2)}, \sigma_i^{(3)})$. The algorithm first checks that all of $id_1, id_2, \ldots, id_i$ are distinct and outputs 0 if not. It then checks if

$$\mathbf{e}(\sigma_i^{(1)}, g) = \mathbf{e}(g_2, g_1)\mathbf{e}(u' \prod_{i \in \mathcal{U}_{id_i}} u_i, \sigma_i^{(2)})\mathbf{e}(v' \prod_{j \in \mathcal{V}_{V_i}} v_j, \sigma_i^{(3)}).$$

  If so, the algorithm returns 1; else it returns 0.

Now we prove that existential unforgeability of the identity-based signature scheme in [29] implies existential unforgeability of Scheme 1. Given an adversary that breaks the IBSAS scheme, we construct an adversary that simulates the IBSAS scheme and successfully break the identity-based signature scheme in [29]. Similar to the proof of Theorem 1, we use an algorithm to obtain a valid forgery in the existential unforgeability game of the identity-based signature scheme in [29]. We give the following theorem.

**Theorem 2.** *Scheme 1 is existentially unforgeable against adaptively chosen identity-message pairs attack in the standard model if the identity-based signature scheme in [29] is existentially unforgeable against adaptively chosen identity and message attack in the standard model.*

*In particular, let $\mathcal{F}$ be a polynomial-time adversary as in Definition 3 that makes $q_k$ private key extraction queries and $q_s$ signing queries. Then there exists a polynomial-time adversary $\mathcal{F}'$ in the existential unforgeability game of the identity-based signature scheme (see Section 2.1 of [29]) that makes $q_k$ private key extraction queries and $q_s$ signing queries, such that*

$$IBSAS - Adv^{UF}(\mathcal{F}) \leq IBS - Adv^{UF}(\mathcal{F}').$$

*Proof.* Let $\mathcal{F}$ be an adversary as in Definition 3 that makes at most $q_k$ queries to its private key extraction oracle, and at most $q_s$ queries to its signing oracle, and succeeds with advantage $\epsilon$. We construct simulator $\mathcal{F}'$ to play the existential unforgeability game of [29]. Simulator $\mathcal{F}'$ interacts with $\mathcal{F}$ as follows.

Assume $\mathcal{F}'$ maintains three lists in its local storage, called $E - list$, $V - list$, and $S - list$. They are set to empty initially.

**Setup**: Simulator $\mathcal{F}'$ runs $\mathcal{F}$ supplying it with the public parameters.

**Hash Queries**: When $\mathcal{F}$ queries the value of $H_1(id)$, the simulator $\mathcal{F}'$ makes its own identity hash oracle and returns $H_{ID}(id)$ to $\mathcal{F}$. When $\mathcal{F}$ queries the value of $H_2(M_i||\widetilde{L_{i-1}})$, the simulator $\mathcal{F}'$ makes its message hash oracle and returns $H_M(M_i||\widetilde{L_{i-1}})$ to $\mathcal{F}$.

**Private Key Extraction Queries**: Considering a query for the private key of an identity $id$, simulator $\mathcal{F}'$ asks its own private key extraction oracle and returns the result to $\mathcal{F}$, and puts $(id, sk_{id})$ into the $E - list$.

**Signing Queries**: Algorithm $\mathcal{F}$ requests a sequential aggregate signature by supplying an identity $id_i$, a message $M_i$, a list of identity-message pairs $L_{i-1} = ((id_1, M_1), (id_2, M_2), \ldots, (id_{i-1}, M_{i-1}))$, and an aggregate-so-far $\sigma_{i-1}$. Simulator $\mathcal{F}'$ first checks whether $\sigma_{i-1}$ is a valid signature of the identity $id_i$ on message $M_i||\widetilde{L_{i-1}}$. If it's invalid, simulator $\mathcal{F}'$ transmits reject to $\mathcal{F}$, and $\mathcal{F}$ outputs $\perp$. Otherwise, $\mathcal{F}'$ puts $(L_{i-1}, \sigma_{i-1})$ into the $V - list$, queries its own signing oracle for identity $id_i$ and message $M_i||\widetilde{L_{i-1}}$, and sends the resulting signature $\sigma_i$ to $\mathcal{F}$, and puts $(L_i, \sigma_i)$ into the $S - list$.

**Output**: Finally, $\mathcal{F}$ halts, outputting a list of identity-message pairs $L_j^* \overset{def}{=} ((id_1^*, M_1^*), (id_2^*, M_2^*), \ldots, (id_j^*, M_j^*))$ and a purported aggregate signature $\sigma_j^*$.

Assume that the output of $\mathcal{F}$ is a valid forgery, then there exists some $k^* \in \{1, 2, \ldots, j\}$ such that $id_{k^*}^*$ was not queried by $\mathcal{F}$ to its private key extraction oracle and $\mathcal{F}$ did not query $(id_{k^*}^*, M_{k^*}^*, ((id_1^*, M_1^*), (id_2^*, M_2^*), \ldots, (id_{k^*-1}^*, M_{k^*-1}^*)), \sigma')$ to oracle $\mathcal{O}_{\mathbf{Sign}}(\cdot, \cdot, \cdot, \cdot)$ for any $\sigma' \in \{0, 1\}^*$.

For simplicity, denote $L_i^* \overset{def}{=} ((id_1^*, M_1^*), (id_2^*, M_2^*), \ldots, (id_i^*, M_i^*))$ for all $i \in \{1, 2, \ldots, j\}$. We employ Algorithm 1 in the proof of Theorem 1 to get the corresponding $k^*$ and $(L_{k^*}^*, \sigma_{k^*}^*)$. Algorithm $\mathcal{F}'$ outputs the signature $\sigma_{k^*}^*$ of identity $id_{k^*}^*$ on message $M_A^* \overset{def}{=} M_{k^*}^*||M_1^*||id_1^*||M_2^*||id_2^*|| \ldots ||M_{k^*-1}^*||id_{k^*-1}^*$.

Without loss of generality, we assume that $id_{k^*}^*$ and $M_A^*$ have been requested in the **Hash Queries** step. Parse $\sigma_{k^*}^*$ as $(\sigma_1^*, \sigma_2^*, \sigma_3^*)$. Then we have

$$\mathbf{e}(\sigma_1^*, g) = \mathbf{e}(g_2, g_1)\mathbf{e}(u' \prod_{i \in \mathcal{U}_{id_{k^*}^*}} u_i, \sigma_2^*)\mathbf{e}(v' \prod_{l \in \mathcal{V}_{M_A^*}} v_l, \sigma_3^*).$$

Moreover, from the above simulation, we have that (1) $id_{k^*}^*$ was not queried by $\mathcal{F}'$ to its own private key extraction oracle; (2) $\mathcal{F}'$ did not make a signing query of identity $id_{k^*}^*$ on message $M_A^*$. Therefore, $\sigma_{k^*}^*$ is a valid forgery of identity $id_{k^*}^*$ on message $M_A^*$ in the existential unforgeability game against the scheme of [29].

Consequently, $\mathcal{F}'$ is successful whenever $\mathcal{F}$ is. Algorithm $\mathcal{F}'$ makes as many private key extraction queries (resp. signing queries) as $\mathcal{F}$ does.

□

**Corollary 1.** *Scheme 1 is existentially unforgeable against adaptively chosen* *identity-message pairs attack in the standard model based on the hardness of the CDH problem.*

*Proof.* Since the identity-based signature scheme in [29] is existentially unforgeable against adaptively chosen identity and message attack in the standard model based on the hardness of the CDH problem, we obtain this corollary from Theorem 2. □

Therefore, we make a solution to the open problem of Boldyreva et al. [4, 5].

**Remark 1.** Essentially, Scheme 1 is constructed via transforming the 2-level HIBE scheme as an extension of Waters' IBE scheme [34]. From [34], we have that the 2-level HIBE scheme is secure in the standard model assuming the decisional Bilinear Diffie-Hellman (BDH) assumption holds. Therefore, from Theorem 1, we immediately get that Scheme 1 is secure in the standard model based on the hardness of the decisional BDH problem. However, Paterson et al. [29] proved that the identity-based signature scheme, obtained from a modification of Waters' proposed IBE scheme [34], is secure in the standard model under the CDH assumption. The CDH assumption seems more natural than many of the hardness assumptions recently introduced to pairing based cryptography. Thus, in this section, we reduce the security of Scheme 1 to the security of the identity-based signature scheme of Paterson et al. [29].

### *4.2. The Second Concrete IBSAS Scheme Based on the LWE Assumption*

In this section, we propose the first lattice-based IBSAS scheme. Compared with other existing IBSAS schemes, the lattice-based scheme has several advantages: its probable security against quantum computing attacks, average-case to worst-case equivalence as well as simplicity and potential efficiency. We employ the HIBE scheme of Agrawal et al. [2] rather than Cash et al. [13] in order to keep the size of private keys and signatures in the corresponding IBSAS scheme short (Please see Table 1 of [2] for the comparison.). The size of the signature is kept constant regardless of how many signers and messages.

**Lemma 1.** (see [2, 17]) *Let $\mathbf{e}$ be some vector in $\mathbb{Z}^m$ and let $\mathbf{y} \overset{\$}{\leftarrow} \overline{\Psi}_\alpha^m$. Then the quantity $|\mathbf{e}^T\mathbf{y}|$ treated as an integer in $[0, q-1]$ satisfies $|\mathbf{e}^T\mathbf{y}| \leq \|\mathbf{e}\| q\alpha \cdot \omega(\sqrt{\log m}) + \|\mathbf{e}\|\sqrt{m}/2$ with all but negligible probability in $m$.*

**Lemma 2.** (see [2]) *Let $q$ be an integer. There is a fixed constant $C > 1$ and a probabilistic polynomial-time algorithm* $\mathbf{GenBasis}(1^n; 1^m; q)$ *that, for $poly(n)-$bounded $m \geq Cnlgq$, outputs $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ and $\mathbf{T} \in \mathbb{Z}^{m\times m}$ such that:*

- *the distribution of $\mathbf{A}$ is within negl(n) statistical distance of uniform,*
- *$\mathbf{T}$ is a basis of $\Lambda_q^{\perp}(\mathbf{A})$, and*
- *$\|\widetilde{\mathbf{T}}\| \leq \widetilde{L} = O(\sqrt{n \log q})$.*

*Furthermore, suppose $\mathbf{R} \in \mathbb{Z}^{m\times m}$ is sampled from $\mathcal{D}_{m\times m}$ and $s$ satisfies $s > \|\widetilde{\mathbf{T}}\| \cdot s_R \sqrt{m} \cdot \omega(\log^{3/2} m)$. Then there is an algorithm* $\mathbf{BasisDel}(\mathbf{A}, \mathbf{R}, \mathbf{T}, s)$ *that outputs a basis $\mathbf{T}'$ of $\Lambda_q^{\perp}(\mathbf{AR}^{-1})$. Let $\mathbf{T}_{ar}$ be an arbitrary basis of $\Lambda_q^{\perp}(\mathbf{AR}^{-1})$ satisfying $\|\widetilde{\mathbf{T}_{ar}}\| < s/\omega(\sqrt{\log m})$. Then $\mathbf{T}'$ is distributed statistically close to the distribution* $\mathbf{RandBasis}(\mathbf{T}_{ar}, s)$, *where* $\mathbf{RandBasis}(\mathbf{T}_{ar}, s)$ *is a probabilistic polynomial-time algorithm that outputs a basis $\mathbf{T}''$ satisfying $\|\widetilde{\mathbf{T}''}\| \leq s\sqrt{m}$ with overwhelming probability. Moreover, if $\mathbf{R}$ is a product of $l$ matrices sampled from $\mathcal{D}_{m\times m}$, then $s > \|\widetilde{\mathbf{T}}\| \cdot (s_R\sqrt{m} \cdot \omega(\log^{1/2} m))^l \cdot \omega(\log m)$.*

**Lemma 3.** (see [17]) *Let $q \geq 2$ and $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ where $m > n$. Let $\mathbf{T}$ be a basis of $\Lambda_q^{\perp}(\mathbf{A})$ and $\tau \geq \|\widetilde{\mathbf{T}}\| \cdot \omega(\sqrt{\log m})$. Then for $\mathbf{u} \in \mathbb{Z}_q^n$, there is a PPT algorithm* $\mathbf{SamplePre}(\mathbf{A}, \mathbf{T}, \mathbf{u}, \tau)$ *that returns $\mathbf{d} \in \Lambda_q^{\mathbf{u}}(\mathbf{A})$ sampled from a distribution statistically close to $\mathcal{D}_{\Lambda_q^{\mathbf{u}}(\mathbf{A}),\tau}$, whenever $\Lambda_q^{\mathbf{u}}(\mathbf{A})$ is not empty.*

Now we present the IBSAS scheme based on the LWE problem via converting the 2-level HIBE scheme of Agrawal et al. [2].

Scheme 2:

Assuming there exists a hash function $H : (\{0,1\}^*)^{\leq 2} \to \mathbb{Z}^{m\times m}$ satisfying that over the choice of the random oracle $H$, the output $H(id)$ is distributed as $\mathcal{D}_{m\times m}$.

- $\mathbf{Setup}(1^n)$: On input a security parameter $n$, get the public parameters $\mathbf{params} = (q, m, n, \widetilde{L}, s_R, s, s', \alpha, D)$ where prime $q = poly(n)$, dimension $m \geq 2n \lg q$, $\widetilde{L} = O(\sqrt{n \log q})$, $s_R = \sqrt{n \log q} \cdot \omega(\sqrt{\log m})$, $s > \|\widetilde{\mathbf{T}}\| \cdot s_R\sqrt{m} \cdot \omega(\log^{3/2} m)$, $s' > \|\widetilde{\mathbf{T}}\| \cdot (s_R\sqrt{m} \cdot \omega(\log^{1/2} m))^2 \cdot \omega(\log m)$, where $\mathbf{T}$ is the output of $\mathbf{GenBasis}(1^n; 1^m; q)$, $D = \{\mathbf{T}' \in \mathbb{Z}^{m\times m} : \|\widetilde{\mathbf{T}'}\| < s'\sqrt{m}\}$.

  To keep the correctness of the scheme, the parameters should be as follows (see Section 4.2 of [2] for the detailed analysis). $n^{\delta} > \lceil \log q \rceil = O(2 \log n)$, $m = \lceil 6n^{1+\delta} \rceil = O(2n \log n)$, $q = m^5 \cdot \omega(\log^5 n)$, $s = m^2 \cdot \omega(\log^2 n)$, $s' = m^{7/2} \cdot \omega(\log^4 n)$, $\alpha = [s'm \cdot \omega(\log n)]^{-1}$.

Invoke $\mathbf{GenBasis}(1^n; 1^m; q)$ to generate a uniformly random matrix $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ together with a short basis $\mathbf{T} \in \mathbb{Z}^{m\times m}$ for $\Lambda_q^{\perp}(\mathbf{A})$. Generate a uniformly random vector $\mathbf{u_0} \in \mathbb{Z}_q^n$. It returns $(\mathbf{A}, \mathbf{u_0})$ as the public parameters $PP$ and $\mathbf{T}$ as the master secret key $msk$.

- $\mathbf{Extract}(msk, id)$: Compute $sk_{id} \leftarrow \mathbf{BasisDel}(\mathbf{A}, H(id), \mathbf{T}, s)$. Return $sk_{id}$.

- $\mathbf{Sign}(L_{i-1}, sk_{id_i}, \sigma_{i-1},\ M_i)$: For $i \geq 2$, the algorithm checks that $\sigma_{i-1}$ is a valid aggregate signature (according to the verification algorithm below) and returns $\perp$ if not. If so, denote $id'_i \leftarrow (id_i, M_i||\widetilde{L_{i-1}})$, and compute

$$\sigma_i \leftarrow \mathbf{BasisDel}(\mathbf{A}H(id_i)^{-1}, H(id_i, M_i||\widetilde{L_{i-1}}), sk_{id_i}, s').$$

  Return $\sigma_i$.

- $\mathbf{Vf}(PP,\ L_i,\ \sigma_i)$: The algorithm first checks that all of $id_1, id_2, \ldots, id_i$ are distinct and outputs 0 if not. Let $id'_i = (id_i, M_i||\widetilde{L_{i-1}})$. Choose noise vectors $\mathbf{x} \overset{\overline{\Psi}_\alpha}{\leftarrow} \mathbb{Z}_q$ and $\mathbf{y} \overset{\overline{\Psi}_\alpha^m}{\leftarrow} \mathbb{Z}_q^m$. Let $\tau = s'\sqrt{m} \cdot \omega(\sqrt{\log m})(\geq \|\widetilde{sk_{id'_i}}\| \cdot \omega(\sqrt{\log m}))$. Set $\mathbf{d}_{id'_i} \leftarrow \mathbf{SamplePre}(\mathbf{A}H(id_i)^{-1}H(id_i, M_i||\widetilde{L_{i-1}})^{-1}, \sigma_i, \mathbf{u_0}, \tau)$.

  Check if $\sigma_i \in D$ and $|\mathbf{x} - \mathbf{d}_{id'_i}^T\mathbf{y}| < \frac{q}{5}$. If so, the algorithm returns 1; else it returns 0.

It should be noted that the above verification step is equivalent to the counterpart in Section 3.1 via replacing the "**Encrypt**" and "**Decrypt**" algorithms with the concrete counterparts in [2]. Without confusion, we denote the letter as $\mathbf{Vf'}(PP,\ L_i,\ \sigma_i)$. In the following, we first show the concrete form of $\mathbf{Vf'}(PP,\ L_i,\ \sigma_i)$, and then prove the equivalence.

$\mathbf{Vf'}(PP,\ L_i,\ \sigma_i)$: The algorithm first checks that all of $id_1, id_2, \ldots, id_i$ are distinct and outputs 0 if not. Let $id'_i = (id_i, M_i||\widetilde{L_{i-1}})$ and $M' \overset{\$}{\leftarrow} \{0, 1\}$.

1. Compute $C' = \mathbf{Encrypt}(PP, id'_i, M')$:

Pick uniformly random vector $\mathbf{s} \overset{\$}{\leftarrow} \mathbb{Z}_q^n$ and noise vectors $x \overset{\overline{\Psi}_\alpha}{\leftarrow} \mathbb{Z}_q$ and $\mathbf{y} \overset{\overline{\Psi}_\alpha^m}{\leftarrow} \mathbb{Z}_q^m$. Using Regev's dual public key encryption (as defined in [17]), we get the ciphertext $(c_0 = \mathbf{u_0}^T\mathbf{s} + x + M'\lfloor \frac{q}{2} \rfloor, \mathbf{c_1} = (\mathbf{A}H(id_i)^{-1}H(id_i, M_i||\widetilde{L_{i-1}})^{-1})^T\mathbf{s} + \mathbf{y})$.

2. Compute $\mathbf{Decrypt}(PP, sk_{id'_i} = \sigma_i, C')$:

(a) Let $\tau = s'\sqrt{m} \cdot \omega(\sqrt{\log m})$ $(\geq \|\widetilde{sk_{id'_i}}\| \cdot \omega(\sqrt{\log m}))$.

(b) Set $\mathbf{d}_{id'_i} \leftarrow \mathbf{SamplePre}(\mathbf{A}H(id_i)^{-1}H(id_i, M_i||\widetilde{L_{i-1}})^{-1}, \sigma_i, \mathbf{u_0}, \tau)$.

(c) Compute $w = c_0 - \mathbf{d}^T_{id'_i}\mathbf{c_1} \in \mathbb{Z}_q$. Compare $w$ and $\lfloor \frac{q}{2} \rfloor$. If they are close, i.e., if $|w - \lfloor \frac{q}{2} \rfloor| < \lfloor \frac{q}{4} \rfloor$ in $\mathbb{Z}$, then the algorithm returns 1; else it returns 0.

3. Check whether $\mathbf{Decrypt}(PP, sk_{id'_i} = \sigma_i, C') = M'$. If so, the algorithm returns 1; else it returns 0.

**Corollary 2.** $\mathbf{Vf}'$*(PP, $L_i$, $\sigma_i$) and* $\mathbf{Vf}$*(PP, $L_i$, $\sigma_i$) are equivalent.*

*Proof.* If $\mathbf{Vf}(PP,\ L_i,\ \sigma_i)$ outputs 1, then $|\mathbf{x} - \mathbf{d}^T_{id'_i}\mathbf{y}| < \frac{q}{5}$ and $|w - \lfloor \frac{q}{2} \rfloor| = |\mathbf{x} + M'\lfloor \frac{q}{2} \rfloor - \mathbf{d}^T_{id'_i}\mathbf{y} - \lfloor \frac{q}{2} \rfloor|$. Hence, $M' = 1$ iff $\mathbf{Decrypt}(PP, sk_{id'_i} = \sigma_i, C') = 1$, that is, $\mathbf{Vf}'(PP,\ L_i,\ \sigma_i)$ outputs 1. On the other hand, if $\mathbf{Vf}'(PP,\ L_i,\ \sigma_i)$ outputs 1, from the analysis of [2], we have $|\mathbf{x} - \mathbf{d}^T_{id'_i}\mathbf{y}| < \frac{q}{5}$, and $\sigma_i \in D$. Thus, $\mathbf{Vf}(PP,\ L_i,\ \sigma_i)$ outputs 1.

**Theorem 3.** *Scheme 2 is existentially unforgeable against adaptively chosen identity-message pairs attack in the random oracle model under the hardness of the LWE problem.*

*Proof.* From Theorem 5 of [2], we have that the 2-level HIBE scheme is IND-ID-CPA secure in the random oracle model under the hardness of the LWE problem. Combining it with Theorem 1, we obtain this result. □

## 5. Performance Comparisons of IBSAS Schemes

Table 1 shows the details of the performance comparisons between our schemes and previous IBSAS schemes [5, 14]. We assume there are $Q$ cosigners involved. The efficiency is considered to include the private key extraction cost, signing cost, verification cost, aggregate signature size, hardness assumption, resistance to quantum computing based attacks and security model. For simplicity, we have the following notations:

$M$: multiplication in $\mathbb{G}$;
$E$: exponentiation in $\mathbb{G}$;
$P$: bilinear pairing in $\mathbb{G}$;

$M'$: multiplication in $\mathbb{G}_T$;
$|\mathbb{G}|$: the bit-length of an element in the group $\mathbb{G}$;
$T_{BasisDel}$: the time of the algorithm $\mathbf{BasisDel}(\cdot,\cdot,\cdot,\cdot)$;
$T_s$: the time of choosing a sample from $\bar{\Psi}_\alpha$;
$T_{mv}$: the time of matrix-vector multiplication in $\mathbb{Z}_q^{m\times m}\times\mathbb{Z}_q^m$;
$E_N$: exponentiation in $\mathbb{Z}_N^*$;
$M_N$: multiplication in $\mathbb{Z}_N^*$.

Moreover, we let RQA denote resistance to quantum-computer-based attacks, ROM denote random oracle model, and SM denote standard model, respectively. For simplicity, we employ the upper bound of the Gram-Schmidt norm of the aggregate signature to represent the size of aggregate signature in Scheme 2.

| Schemes | Cost of Extract | Cost of Sign | Cost of Verify | Size of Aggregate Signature | Hardness Assumption | RQA | ROM/ SM |
|---|---|---|---|---|---|---|---|
| Scheme 1 | $lM+2E$ | $nM+2E$ | $4P+2M'+$<br>$(n+l)M$ | $3\|\mathbb{G}\|$ | CDH | No | SM |
| Scheme 2 | $T_{BasisDel}$ | $T_{BasisDel}$ | $(m+1)T_s+T_{mv}$ | $m^{7/2}\cdot\omega(\log^4 n)\sqrt{m}$ | LWE | Yes | ROM |
| [5] | $2E$ | $5E+6M$ | $4P+2M'+$<br>$2(Q-1)M+QE$ | $3\|\mathbb{G}\|$ | IBSAS-CDH | No | ROM |
| [14] | $E_N$ | $2E_N+2M_N$ | $(Q+1)E_N$ | $(Q+1)\log N$ | RSA | No | ROM |

Table 1: Performance Comparisons of IBSAS Schemes.

Compared to previous constructions, our schemes have several advantages. Firstly, Scheme 1 is the first secure IBSAS scheme in the standard model, as others are secure in the random oracle model. Secondly, Scheme 1 is based on the CDH assumption, which is more standard than the IBSAS-CDH problem adopted in [5]. Thirdly, Scheme 2 is based on the lattice problem (i.e., the LWE problem), for which there are currently no known quantum algorithms, while all previous constructions are based on the IBSAS-CDH problem and RSA problem, both of which can be solved by efficient quantum algorithms [30]. Although large-scale quantum computers are not expected to exist in the near future, it's meaningful to construct cryptosystems which are resistant to quantum-computer-based attacks. Finally, unlike Scheme 1 and the IBSAS scheme of [5], no bilinear pairing operation is needed during the verification step of Scheme 2, the operations of Scheme 2 are simpler.

## 6. Conclusion

In this paper, we firstly presented a generic construction of IBSAS schemes, and proved that an IBSAS scheme is existential unforgeable against adaptively chosen identity-message pairs attack under the assumption that the original 2-level HIBE scheme is IND-ID-CPA secure. Afterwards, we initiated the generic construction to obtain two concrete IBSAS schemes. Note that the first concrete IBSAS scheme is a solution to the open problem pointed out by Boldyreva et al. in CCS' 2007, since the security of this scheme can be reduced to the hardness of CDH problem in the standard model. Additionally, the second concrete scheme is the first lattice-based IBSAS scheme. We also compared our schemes with previous ones. We believe that our constructions may be applied to secure network routing and sensor networks.

## Acknowledgements

The authors would like to thank Prof. Yevgeniy Dodis and Prof. Victor Shoup for their help about this paper. This work is supported by the National Natural Science Foundation of China (Grant Nos. 61602025, 61170189, 61370126 and 61300172), the Fund for the Doctoral Program of Higher Education of China(Grant No. 20121102120017), the Fund of the State Key Laboratory of Software Development Environment(Grant No. SKLSDE-2016ZX-22), and CSC grant 201206020063.

[1] Agrawal S, Boneh D, Boyen X. Efficient Lattice (H)IBE in the Standard Model. *EUROCRYPT* 2010; 553-572.

[2] Agrawal S, Boneh D, Boyen X. Lattice Basis Delegation in Fixed Dimension and Shorter-Ciphertext Hierarchical IBE. *CRYPTO* 2010; 98-115.

[3] Bellare M, Namprempre C, Neven G. Unrestricted Aggregate Signatures. *ICALP* 2007; 4596: 411-422.

[4] Boldyreva A, Gentry C, O'Neill A, Yum DH. Ordered Multisignatures and Identity-Based Sequential Aggregate Signatures, with Applications to Secure Routing. *CCS* 2007; 276-285.

[5] Boldyreva A, Gentry C, O'Neill A, Yum DH. Ordered multisignatures and identity-based sequential aggregate signatures, with applications to secure routing. Cryptology ePrint Archive, Report 2007/438, 2010. http://eprint.iacr.org/2007/438.

[6] Boneh D, Boyen X. Efficient Selective Identity-Based Encryption Without Random Oracles. *Journal of Cryptology* 2011; 24(4): 659-693.

[7] Boneh D, Boyen X. Short signatures without random oracles. *EUROCRYPT* 2004; 56-73.

[8] Boneh D, Boyen X, Goh E. Hierarchical Identity Based Encryption with Constant Size Ciphertext. *EUROCRYPT* 2005; 440-456.

[9] Boneh D, Franklin MK. Identity-based encryption from the weil pairing. *SIAM J. Comput.* 2003; 32(3): 586-615.

[10] Boneh D, Gentry C, Lynn B, Shacham H. Aggregate and verifiably encrypted signatures from bilinear maps. *EUROCRYPT* 2003; 2656: 416-432.

[11] Boneh D, Lynn B, Shacham H. Short signatures from the Weil pairing. *Journal of Cryptology* 2004; 17(4): 297-319.

[12] Camenisch J, Lysyanskaya A. Signature Schemes and Anonymous Credentials from Bilinear Maps. *CRYPTO* 2004; 3152: 56-72.

[13] Cash D, Hofheinz D, Kiltz E, Peikert C. Bonsai Trees, or How to Delegate a Lattice Basis. *EUROCRYPT* 2010; 523-552.

[14] Dou B, Chen C, Zhang H, Xu C. Identity-based Sequential Aggregate Signature scheme Based on RSA. *International Journal of Innovative Computing, Information and Control* 2012; 8(9): 6401-6413.

[15] Galindo D, Herranz J, Kiltz E. On the Generic Construction of Identity-Based Signatures with Additional Properties. *ASIACRYPT* 2006; 4284: 178-193.

[16] Gerbush M, Lewko A, O'Neill A, Waters B. Dual Form Signatures: An Approach for Proving Security from Static Assumptions. *ASIACRYPT* 2012; 7658: 25-42.

[17] Gentry C, Peikert C, Vaikuntanathan V. Trapdoors for hard lattices and new cryptographic constructions. *STOC* 2008; 197-206.

[18] Gentry C, Silverberg A. Hierarchical ID-based cryptography. *ASIACRYPT* 2002; 2501: 548-566.

[19] Graells-Garrido E, Peredo O, Garcia J. Sensing Urban Patterns with Antenna Mappings: The Case of Santiago, Chile. *Sensors* 2016; 16(7): 1098. DOI: 10.3390/s16071098

[20] Horwitz J, Lynn B. Toward hierarchical identity-based encryption. *EUROCRYPT* 2002; 466-481.

[21] Hwang J, Lee D, Yung M. Universal forgery of the identity-based sequential aggregate signature scheme. *ASIACCS* 2009; 157-160.

[22] Kent S, Lynn C, Seo K. Secure border gateway protocol (Secure-BGP). *IEEE J. Selected Areas in Comm.* 2000; 18(4): 582-592.

[23] Lee K, Lee DH, Yung M. Sequential Aggregate Signatures with Short Public Keys: Design, Analysis and Implementation Studies. *Public Key Cryptography* 2013; 423-442.

[24] Lu S, Ostrovsky R, Sahai A, Shacham H, Waters B. Sequential aggregate signatures and multisignatures without random oracles. *EUROCRYPT* 2006; 4004: 465-485.

[25] Lysyanskaya A, Micali S, Reyzin L, Shacham H. Sequential Aggregate Signatures from Trapdoor Permutations. *EUROCRYPT* 2004; 3027: 74-90.

[26] Lysyanskaya A, Rivest R, Sahai A, Wolf S. Pseudonym systems. *Selected Areas in Cryptography* 1999; 1758: 184-199.

[27] Mao WB. *Modern Cryptography: Theory and Practice.* Prentice Hall PTR: New Jersey, 2003.

[28] Nicol D, Smith S, Zhao M. Evaluation of efficient security for BGP route announcements using parallel simulation. *Simulation Modelling Practice and Theory* 2004; 12: 187-216.

[29] Paterson KG, Schuldt JCN. Efficient identity-based signatures secure in the standard model. *ACISP* 2006; 4058: 207-222.

[30] Regev O. Lattice-based cryptography. *CRYPTO* 2006; 131-141.

[31] Regev O. On lattices, learning with errors, random linear codes, and cryptography. *J. ACM* 2009; 56(6). Extended abstract in *STOC* 2005; 84-93.

[32] Schr*ö*der D. How to Aggregate the CL Signature Scheme. *ESORICS* 2011; 298-314.

[33] Shmair A. Identity-based cryptosystems and signature schemes. *CRYPTO* 1984; 196: 47-53.

[34] Waters B. Efficient identity-based encryption without random oracles. *EUROCRYPT* 2005; 3494: 114-127.

[35] Zoha A, Gluhak A, Imran MA, Rajasegarar S. Non-Intrusive Load Monitoring Approaches for Disaggregated Energy Sensing: A Survey. *Sensors* 2012; 12(12): 16838-16866.

## Appendix A.

**Definition 5** *Suppose that $\mathbb{G}$ and $\mathbb{G}_T$ are two multiplicative groups of prime order $p$ and $g$ is a generator of $\mathbb{G}$, where all group operations can be computed efficiently. An efficiently computational map $\mathbf{e} : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ is called an efficiently computational bilinear pairing, if it satisfies the following two properties:*

- *Non-degeneracy: $\mathbf{e}(g, g) \neq 1$ and is thus a generator of $\mathbb{G}_T$;*
- *Bilinearity: $\forall u, v \in \mathbb{G}$ and $\forall a, b \in \mathbb{Z}$: $\mathbf{e}(u^a, v^b) = \mathbf{e}(u, v)^{ab}$ holds.*

## Appendix B.

**Definition 6** *(see [26, 32]) Let $\mathbb{G}$ and $\mathbb{G}_T$ be two multiplicative groups of prime order $p$ and $g$ be a generator of $\mathbb{G}$. Let $X, Y \in \mathbb{G}$ such that $X = g^x$ and $Y = g^y$ for some $x, y \in \mathbb{Z}_p$. Denote $\rho := (p, \mathbb{G}, \mathbb{G}_T, g, \mathbf{e}, X, Y)$ and let $O_{X,Y}$ be an oracle that on input a value $M \in \mathbb{Z}_p$ outputs a triplet $(a, a^y, a^{x+Mxy})$ for a randomly chosen $a \in \mathbb{G}$. The LRWS assumption states that for all efficient algorithms $\mathcal{A}^{O_{X,Y}}$, the following holds:*

$$\Pr[x \leftarrow \mathbb{Z}_p; y \leftarrow \mathbb{Z}_p; X \leftarrow g^x; Y \leftarrow g^y; (Q, M, a, b, c) \leftarrow \mathcal{A}^{O_{X,Y}}(\rho) : \\ M \notin Q \wedge a \in \mathbb{G} \wedge b = a^y \wedge c = a^{x+Mxy}] = negl(1^n),$$

*where $Q$ is the set of oracle queries.*