Identity-based Matchmaking Encryption with Stronger Security and Instantiation on Lattices

Yuejun Wang[1,2], Baocang Wang[1], Qiqi Lai[3,2], and Yu Zhan[1]

[1] State Key Laboratory of Integrated Service Networks, Xidian University, Xi'an, China
yuejun.w@stu.xidian.edu.cn bcwang@xidian.edu.cn zhanyu@xidian.edu.cn
[2] Henan Key Laboratory of Network Cryptography Technology, Henan, China
[3] School of Computer Science, Shaanxi Normal University, Xi'an, China
laiqq@snnu.edu.cn

Abstract. An Identity-based matchmaking encryption (IB-ME) scheme proposed at CRYPTO 2019 supports anonymous but authenticated communications in a way that communication parties can both specify the senders or receivers on the fly. IB-ME is easy to be used in several network applications requiring privacy-preserving for its efficient implementation and special syntax. In the literature, IB-ME schemes are built from the variants of Diffie-Hellman assumption and all fail to retain security for quantum attackers. Despite the rigorous security proofs in previous security models, the existing schemes are still possibly vulnerable to some potential neglected attacks. Aiming at the above problems, we provide stronger security definitions considering new attacks to fit real-world scenarios and then propose a generic construction of IB-ME satisfying the new model. Inspired by the prior IB-ME construction of Chen *et al.*, the proposed scheme is constructed by combining 2-level anonymous hierarchical IBE (HIBE) and identity-based signature (IBS) schemes. In order to upgrade lattice-based IB-ME with better efficiency, we additionally improve a lattice IBS, as an independent technical contribution, to shorten its signature and thus reduce the final IB-ME ciphertext size. By combining the improved IBS and any 2-level adaptively-secure lattice-based HIBE with anonymity, we finally obtain the *first* IB-ME from lattices.

Keywords: Matchmaking Encryption, Identity-based Matchmaking Encryption, Lattice, Identity-based Signature, Security Model

1 Introduction

Matchmaking Encryption (ME) [AFNV19] is a quite useful primitive which allows any sender and receiver to predesignate policies that the other party should satisfy to reveal message. In an ME scheme, a sender uses the secret encryption key ek_σ associated with his attribute σ to generate ciphertexts with additional specifying policy $\mathbb{R}$. Each receiver obtains different decryption keys from the authority with just one key dk_ρ associated with his attribute ρ and others $\mathsf{dk}_\mathbb{S}$ related to his chosen policy $\mathbb{S}$. When decrypting a ciphertext linking (σ, $\mathbb{R}$) using dk_ρ and $\mathsf{dk}_\mathbb{S}$, the receiver recovers the plaintext by matching the attributes and policies of both participants. The entire procedure of policy matchmaking is privacy-preserved. In other words, nothing is leaked beyond the fact that a match occurred/did not occur. Furthermore, malicious attacks fail to forge ciphertexts embedding fake attributes which were not certificated by the authority.

ME naturally supports several network applications requiring secret communication, such as the scenarios that the both communicating parties need to specify access polices to encrypted plaintext. Ateniese *et al.* [AFNV19] proposed two generic constructions of ME relying on 2-input Functional Encryption (FE) scheme [GGG+14] or FE scheme for randomized functionalities (rFE) [GJKS15, AW17]. However, the ME scheme based on rFE only achieves security against bounded collusions. Another approach requiring 2-input FE for general circuits retains full security, but this construction can only be instantiated on sub-exponentially secure indistinguishable obfuscation (iO), which is a non-standard strong assumption. In fact, this notion of 2-input FE only requires identity function over plaintexts with access control over attributes. Hence, Francati *et al.* [FFMV22] recently presented multi-key predicate encryption (PE) bulit from learning with errors (LWE) directly and then construct ME with unbounded collusions using 2-key PE.

Concept of IB-ME. By contrast, the ME scheme in the restricted identity-based setting can be more efficient than ME for general functions and be easy to implement. Identity-based matchmaking encryption (IB-ME) can also be used to construct an anonymous but authentic communication environment.

Being a special case of ME under the identities equality policy, IB-ME features that each sender is given a secret encryption key related to his identity σ, and each receiver has a secret decryption key for his identity ρ. Similarly, senders can select target receiver rcv and encrypt secretly. The receiver takes dk_ρ and arbitrary identity snd as input to decryption algorithm, without an additional decryption key for snd, and obtains messages if and only if identities equality policies both match ($\rho = \mathsf{rcv} \ \wedge \ \sigma = \mathsf{snd}$). The security requirements of IB-ME are privacy and authenticity. When mismatch happens ($\rho \neq \mathsf{rcv} \ \vee \ \sigma \neq \mathsf{snd}$), privacy not only protects plaintext from illegal leaking, but also prevents decryptors from learning any extra information about sender's identity. Another property, namely authenticity, promises that the ciphertexts associated with σ could only be generated by encryption key ek_σ.

Existing work for IB-ME. The existing constructions for IB-ME are all based on the variants of discrete log problems. By amplifying a secure identity-based encryption (IBE) under chosen plaintext attack [BF01], Ateniese *et al.* [AFNV19] provided the first IB-ME from bilinear Diffie-Hellman (BDH) assumption. The follow-up work [FGRV21] proposed an instantiation without random oracle based on non-standard augumented bilinear Diffie-Hellman exponent assumption (q-ABDHE), and its privacy relies on the underlying anonymous IBE [Gen06] in the standard model. The scheme also requires non-interactive zero knowledge proof (NIZK) to achieve authenticity.

Very recently, Chen *et al.* [CLWW22] presented an IB-ME from symmetric external Diffie-Hellman (SXDH) assumption in the standard model. The scheme is built from a variant of anonymous IBE [CLL+13] by absorbing the idea of 2-level Hierarchical Predicate Encryption (HPE) [OT09]. In the scheme, receivers obtain 1-level decryption key dk_ρ from the authority, and each sender obtains an encryption key ek_σ which is the signature for message σ (encoded in 2-level) signed by the authority using master secret key. Sender is allowed to use ek_σ to generate ciphertexts making decryption works correctly if the counterpart matches the corresponding equality policy. The privacy relies on the anonymity of 1-level IBE, while the authenticity can be reduced to the unforgeability of the underlying signature scheme.

More general application scenarios. In previous security models, the authenticity is solely identity authentication, especially less relevant to encrypted message. In other words, the existing security models fail to prevent tampering with plaintexts, even forging. When considering some real-world application scenarios, there exists some classes of forgery of message might affect authenticity, meaning that it is necessary to model new security models capturing such attacks. The first potential forgery is *"forging-to-itself"*. In such scenarios, adversaries might get access to obtain ciphertexts from chosen sources, even decrypt partially. Suppose the dean of a faculty usually authorize the associate dean to act him while he is busy. IB-ME enables the dean to delegate part of powers to the associate dean, with the guarantee that the third party fails to learn information about the encrypted content. However, once the associate dean can construct a fake authorization on his own, he might have the possession of arbitrary authority and claim that he was authorized by the dean.

Another potential forgery is *tamper forgery*. Suppose IB-ME was applied to a bulletin board hidden service, everyone gets access to upload ciphertexts to open server and download ciphertexts from it. Then hackers might obtain and manipulate ciphertexts then upload them without needing any keys, receivers will thus get wrong information.

Nevertheless, the previous schemes [AFNV19, CLWW22] is proven secure under the original security model, these schemes still requires some additional adjustments to recognize the aforementioned forgeries due to the reason that these constructions mainly consider security under chosen-plaintext attacks. To some extent, tamper forgeries could be identified simply relying on pre-existing coding rules of plaintext, as tampering will lead to decryption results that do not meet the rules with high probability. While using NIZK is another relatively direct way to solve both potential forgeries in the same time. Intuitively, ciphertexts would contain additional witness to prove the certification of the sender's identity and the plaintext, the decryption algorithm also need to check the validity of the proof. The soundness of NIZK guarantees that the witness generating by malicious adversary fails to pass the verification. On the other hand, nothing secret leaks from the proof itself due to the zero-knowledge property.

Motivation. Although the prior schemes [AFNV19,FGRV21,CLWW22] could supply more practical applications using some tweaks, their security will be entirely broken down for attacks using quantum computers. Quantum algorithms can efficiently solve the mathematical problems such as factoring and discrete log problem, while the previous schemes rely on the latter to protect both privacy and authenticity. This motivates us to think about the following question:

Can we build a post-quantum secure IB-ME supporting more general applications?

1.1 This Work

The work gives several contributions for an IB-ME with stronger security.

- *Improved security definitions.* We modified the security definitions to match more general application requirements. Specifically, we allow the adversary to forge to itself and obtain ciphertexts encrypted with its chosen source and destination by querying encryption oracle. Also the "forge-to-itself" ciphertexts are admissible.

- *Generic construction.* Inspired by the previous schemes, we propose a generic construction to satisfy modified stronger security definitions. Our construction is based on adaptively secure 2-level anonymous HIBE and Identity-based Signature scheme (IBS). The privacy of IB-ME is implied by the anonymity of underlying HIBE. The unforgeability of IBS can guarantee the authenticity.

- *Instantiation on lattices.* To further improve the efficiency of lattice-based IB-ME, we additionally modify an existing IBS based on SIS problem to achieve shorter signature (reduce by $n \cdot (\lceil \log q \rceil)^2$ bits) with better efficiency. Finally, by combining our improved IBS and any existing 2-level Hierarchical IBE (HIBE) with adaptive security and anonymity (e.g., [ABB10b]), we obtain the *first* IB-ME from lattices in the random oracle model.

1.2 Technical Overview

Here we present an overview of our technical approach for IB-ME construction. We focus on showing our new and easy-understanding construction method to satisfy stronger security.

IB-ME from 2-level HIBE and IBS. Intuitively, our generic construction approach can be separated into two steps. Here we show a brief overview, the complete construction is given in section 4.

HIBE implies an imperfect IB-ME. Inspired by Chen *et al.* [CLWW22], we observe that a 2-level hierarchical identity-based encrypion (HIBE) scheme can directly be used to construct an IB-ME if taking authenticity aside for a while. Note that the main distinction between syntaxes of HIBE and IB-ME is that a receiver is allowed to assign sender in an IB-ME scheme. Thus, we allow decryptors to delegate key associated with $\mathsf{rcv} \mid \mathsf{snd}$ for any snd.

In more details, we construct the IB-ME scheme as follows:

- Setup: Run $\mathsf{HIBE.Setup}$ to obtain $\mathsf{mpk} := \mathsf{HIBE.mpk}$ and $\mathsf{msk} := \mathsf{HIBE.msk}$.
- $\mathsf{SKGen}(\mathsf{msk}, \sigma)$: $\mathsf{ek}_\sigma := \sigma$.
- $\mathsf{RKGen}(\mathsf{msk}, \rho)$: $\mathsf{sk}_\rho \leftarrow \mathsf{HIBE.Keygen}(\mathsf{HIBE.msk}, \rho)$, set $\mathsf{dk}_\rho := \mathsf{sk}_\rho$.
- $\mathsf{Enc}(\mathsf{mpk}, \mathsf{ek}_\sigma, \mathsf{rcv}, m)$: $\mathsf{ct} := \mathsf{HIBE.Enc}(\mathsf{rcv} \mid \sigma, m)$.
- $\mathsf{Dec}(\mathsf{mpk}, \mathsf{dk}_\rho, \mathsf{snd}, C)$:
  1. $\mathsf{sk}_{\rho|\mathsf{snd}} \leftarrow \mathsf{HIBE.Derive}(\mathsf{dk}_\rho, \rho \mid \mathsf{snd})$.
  2. m *or* $\perp \leftarrow \mathsf{HIBE.Dec}(\mathsf{sk}_{\rho|\mathsf{snd}}, \mathsf{ct})$.

It is clear that when both match conditions satisfy, receiver can generate the correct 2-level key to help to recover message. Although authenticity has been ignored for the moment, privacy issues have been already fixed out. As long as one single condition doesn't hold ($\mathsf{rcv} \neq \rho$ or $\mathsf{snd} \neq \sigma$), receiver(malicious or not) cannot learn anything but decryption failure itself.

So far, our construction approach looks quite similar to the idea of the variant construction of 2-level IBE in [CLWW22]. However, the authenticity in their construction is related to the unforgeability of underlying signature scheme, while we choose a different way.

Guarantee authenticity by sign-then-encrypt. To further overcome authenticity problem, sender has to deliver a witness to persuade legal receivers that the received ciphertext came from claimed source exactly. In our setting, the witness is just an identity-based signature for message rcv generated by sender. Unlike Chen *et al.* [CLWW22], authority gives out the identity-based signing capability to each sender rather than the signature for message id. Thus, authenticity naturally guaranteed by the unforgeability of IBS. However, merely adding identity-based signature as part of ciphertexts would break privacy of IB-ME, because owners of non-target $\mathsf{dk}_{\mathsf{id}}$ could also verify and learn identity information of sender. The solution is let sender also encrypt witness under 2-level public key $\mathsf{rcv} \mid \mathsf{snd}$, which implies that only target receiver can check authenticity by verifying the validity of signature. For those illegal receivers whom $\mathsf{id} \neq \mathsf{rcv}$, nothing leaks. Furthermore, in order to avoid the case that the receiver reuses the witness to forge ciphertexts, we tweak the above signature to contain encrypted plaintext additionally.

Notice also that the generic constructions for ME and Arranged ME (A-ME) [AFNV19] all require NIZK as essential tools to achieve authenticity. To replace NIZK in prior schemes, we can choose a similar approach that accomplishing the goal of correctness and privacy primarily then guaranteeing authenticity by attribute-based signature (ABS). Precisely, using attribute-based signature key from the authority, a sender additionally encrypts attribute-based signatures for related messages and attributes. Hence, the authenticity can be naturally captured by the unforgeability of underlying ABS.

Instantiated with shorter ciphertext. The proposed generic construction can be instantiated from 2-level anonymous with adaptively security HIBE (e.g., based on lattice assumptions [ABB10a, CHKP12, ABB10b, BL16] or SXDH [LP20b, LP19, LP20a]) and adaptively secure IBS from various assumptions. As the final ciphertext contains the encryption of signature, we improve an existing IBS [PW21] based on short interger solution (SIS) [Ajt96] to reduce the signature size, thus obtain a shorter ciphertext.

In an IBS scheme [Sha84], any authorized user can generate signature using secret signing key, and everyone can verify whether the signature is valid or not by public parameters and user's identity. The previous works for IBS with tight adaptive security are mainly constructed by the following approaches. The standard signature scheme which is tightly secure in the multi-user setting with adaptive corruption can be used to obtain tightly secure IBS [LPLL20, DKXY03, BNN04]. However, the related existing works [BHJ+15, GJ18, DGJL21] are all based on Diffie-Hellman assumption. Even though there is generic construction [BHJ+15], it still requires a non-interactive witness-indistinguishable proof of knowledge (NIWIPoK) system which has no efficient instantiation in the post-quantum setting like lattices. Another approach [GS02] is to transform any 2-level HIBE into a tightly secure IBS. But the existing technical approaches [BL16,LLW20] that use Katz-Wang random-bit technique [GJKW07] to achieve (almost) tightly secure IBE fail to trivially construct tightly secure 2-level HIBE, informally due to the reason that the hidden random bit $b_{\mathsf{id}_1^*}$ of level-1 challenge identity id_1^* could be learned by asking secret key for $(\mathsf{id}_1^*, \mathsf{id}_2)$, where $\mathsf{id}_2 \neq \mathsf{id}_2^*$, and then the security proof cannot work.

Pan and Wagner [PW21] recently proposed a new approach to construct tightly adaptive secure identity-based signature with signature size independent of message length from lattices. Informally, the first step is constructing an IBS with unforgeability under non-adaptive chosen message attacks (UF-naCMA), and the second step is to upgrade UF-naCMA construction into adaptively secure (UF-CMA) one by generic transformations using tools like chameleon hash functions. In our work, we will show a simpler signing algorithm and the length of signature can reduce by $n \cdot (\lceil \log q \rceil)^2$ bits. The resulting IBS will be further used to construct the IB-ME scheme.

1.3 Related Work

Authenticated identity-based encryption and identity-based signcryption. Authenticated encryption (AE) [Zhe97] in the identity-based setting, i.e. identity-based signcryption [Mal02, Boy03, BLMQ05], enables the intended receiver has the sole ability to decrypt and can authenticate that the message is indeed from the specified sender. However, the receivers in the identity-based signcryption scheme usually need to firstly recover the purported identity (also signature and message) and then check the validity. In an IB-ME scheme, a receiver just take ciphertexts, its decryption key and chosen sender's identity as input, and finally get the message only when matches happen. The whole

decryption procedure implicitly contains the authentication to the message source and the message itself. Also the key generation mechanisms are different. There is only one KeyGen algorithm for users to encrypt or decrypt, while authorities in IB-ME schemes will generate encryption keys and decryption keys, respectively.

2 Preliminaries

2.1 Notations

We denote the natural numbers by $\mathbb{N}$, the integers by $\mathbb{Z}$ and the real numbers by $\mathbb{R}$. Let $\mathbb{Z}_q$ be $\mathbb{Z}/(q\mathbb{Z})$. For a non-negative integer n, we let $[n] = \{1, \cdots, n\}$. Vectors are written in bold lower-case letters (e.g., $\mathbf{x}$) and are always assumed to be in column form. The ith component of vector $\mathbf{x}$ is denoted by x_i. Matrices are written using bold capital letters(e.g., $\mathbf{X}$) and we denote the i-th column vector of a matrix $\mathbf{X}$ by $\mathbf{x}_i$. The Euclidean norm, or l_2 norm, of a vector is denoted by $\|\mathbf{x}\|$. We denote the Euclidean norm and spectral norm of a matrix $\mathbf{X}$ by $\|\mathbf{X}\|$ and $s_1(\mathbf{X})$, respectively. For any set $\mathbf{S} = \{\mathbf{s}_1, \cdots, \mathbf{s}_n\} \subset \mathbb{R}^n$ of linearly independent vectors, we use $\widetilde{\mathbf{S}}$ to denote its Gram-Schmidt orthogonalization.

The security parameter is $\lambda \in \mathbb{N}$ throughout this paper, and every algorithm will implicitly get it as an input. We use standard asymptotic notations to classify the growth of functions, and say that $f(n) = O(g(n))$ if f is bounded above by g asymptotically up to constant factor, $f(n) = o(g(n))$ if f is dominated by g asymptotically, and $f(n) = \omega(g(n))$ if f dominates g asymptotically. We say that a function $f : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ is *negligible* if for any $c \in \mathbb{N}$, $f(\lambda) = o(1/\lambda^c)$, and denote such function by $\mathsf{negl}(\lambda)$. A probability is *overwhelming* if it is $1 - \mathsf{negl}(\lambda)$.

We write $x \xleftarrow{\$} D$ to define that element x is sampled uniformly random from set D. Suppose X and Y are probability distributions on a countable domain D, then their statistical distance is defined as $\Delta(X, Y) = \frac{1}{2} \sum_{d \in D} \mid X(d) - Y(d) \mid$. We say that the distributions X and Y are *statistically close* if $\Delta(X, Y)$ is negligible in n, denoted by $X \overset{s}{\approx} Y$. If for every probabilistic poly-time algorithm $\mathcal{A}$, $\mid \Pr[\mathcal{A}(1^n, \mathrm{X}) = 1] - \Pr[\mathcal{A}(1^n, \mathrm{Y}) = 1] \mid$ is negligible in n, then the two distributions X and Y are *computationally indistinguishable*, denoted by $X \overset{c}{\approx} Y$.

2.2 Lattices Background

A n-dimensional lattice Λ, being a discrete additive subgroup of $\mathbb{R}^n$, is the set $\mathcal{L}(\mathbf{b_1}, \cdots, \mathbf{b_n}) = \{\mathbf{Bz} = \sum_{i \in [n]} z_i \cdot \mathbf{b}_i \mid z_i \in \mathbb{Z}\}$ of all integral combinations of some n linearly independent vectors $\{\mathbf{b}_1, \cdots, \mathbf{b}_n\} \subset \mathbb{R}^n$. The sequence of vectors $\{\mathbf{b}_1, \cdots, \mathbf{b}_n\}$ is called a lattice basis, and it is conveniently represented as a matrix $\mathbf{B} = \{\mathbf{b}_1, \cdots, \mathbf{b}_n\}$.

Many lattice-based cryptosystems use q-ary integer lattices defined by a matrix over $\mathbb{Z}_q$. Formally, let $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ be arbitrary matrix for some positive integers n, m, q, define the full-rank m-dimensional q-ary lattices as follows

$$\Lambda(\mathbf{A}) = \{\mathbf{z} \in \mathbb{Z}^m \mid \exists\ \mathbf{s} \in \mathbb{Z}_q^n,\ s.t.\ \mathbf{A}^t\mathbf{s} = \mathbf{z} \mod q\}$$
$$\Lambda^{\perp}(\mathbf{A}) = \{\mathbf{z} \in \mathbb{Z}^m \mid \mathbf{Az} = \mathbf{0} \mod q\}.$$

For any fixed $\mathbf{u} \in \mathbb{Z}_q^n$, define a coset of $\Lambda^{\perp}$ as:

$$\Lambda_{\mathbf{u}}^{\perp}(\mathbf{A}) = \{\mathbf{z} \in \mathbb{Z}^m \mid \mathbf{Az} = \mathbf{u} \mod q\}.$$

Gaussian on Lattices. For any positive s, the Gaussian distribution $\mathcal{D}_{s,\mathbf{c}}$ centered at $\mathbf{c} \in \mathbb{R}^n$ with parameter s is defined by the following probability distribution function

$$\forall \mathbf{x} \in \mathbb{Z}^n,\ \rho_{s,\mathbf{c}}(\mathbf{x}) = \exp(-\pi \|\mathbf{x} - \mathbf{c}\|^2 / s^2).$$

The subscripts $\mathbf{c}$ is taken to be $\mathbf{0}$ when omitted.

Let $s > 0$, $\mathbf{c} \in \mathbb{R}^n$, for any n-dimensional lattice Λ, define the *discrete Gaussian distribution over* Λ as

$$\forall \mathbf{x} \in \Lambda, \mathcal{D}_{\Lambda,s,\mathbf{c}} = \frac{\rho_{s,\mathbf{c}}(\mathbf{x})}{\rho_{s,\mathbf{c}}(\Lambda)}.$$

The following is a bound on the smoothing parameter for random lattices.

Lemma 1 ([GPV08]). *Let Λ be an n-dimensional lattice with basis $\mathbf{B}$, and let real $\epsilon > 0$, we have*

$$\eta_\epsilon(\Lambda) \leq \|\widetilde{\mathbf{B}}\| \cdot \sqrt{\log(2n(1+1/\epsilon))/\pi}.$$

For any $\omega(\sqrt{\log n})$ function, there is a negligible function $\epsilon(n)$ for which $\eta_\epsilon(\Lambda) \leq \|\widetilde{B}\| \cdot \omega(\sqrt{\log n})$.

Lemma 2 ([GPV08]). *Let n, m, q be positive integers, and let $m \geq 2n \log q$. Then for all but an at most q^{-m} fraction of $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$, we have $\lambda_1^\infty(\Lambda) \geq q/4$.*

In particular, for such $\mathbf{A}$ and any $\omega(\sqrt{\log m})$ function, there is a negligible function $\epsilon(m)$ such that $\eta_\epsilon(\Lambda^\perp(\mathbf{A})) \leq \omega(\sqrt{\log m})$.

Lemma 3 ([MR04, GPV08]). *For any m-dimensional lattice Λ, let $\mathbf{B}$ be a basis for Λ. Suppose $s \geq \|\widetilde{\mathbf{B}}\| \cdot \omega(\sqrt{\log m})$, then $\Pr[\|\mathbf{x}\| \geq s\sqrt{m} : \ \mathbf{x} \leftarrow \mathcal{D}_{\Lambda,s}] \leq \mathsf{negl}(m)$.*

Lemma 4 ([GPV08]). *Let n and q be positive integers with q prime, and let $m \geq 2n \lg q$. Then for all but a $2q^{-n}$ fraction of all $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ and for any $s \geq \omega(\sqrt{\log m})$, the distribution of the syndrome $\mathbf{u} = \mathbf{Ae} \mod q$ is statistically close to uniform over $\mathbb{Z}_q^n$, where $\mathbf{e} \leftarrow \mathcal{D}_{\mathbb{Z}^m,s}$.*

Furthermore, fix $\mathbf{u} \leftarrow \mathbb{Z}_q^n$ and let $\mathbf{x} \leftarrow \mathbb{Z}^m$ be an arbitrary solution to $\mathbf{Ax} = \mathbf{u} \mod q$. Then the conditional distribution of $\mathbf{e} \leftarrow \mathcal{D}_{\mathbb{Z}^m,s}$ given $\mathbf{Ae} = \mathbf{u} \mod q$ is exactly $\mathcal{D}_{\Lambda_\mathbf{u}^\perp(\mathbf{A}),s}$.

The following lemma is the bound for spectral norm of random matrices from the non-asymptotic theory.

Lemma 5 ([MP12]). *Let $\mathbf{X} \in \mathbb{R}^{n\times m}$ be a δ-subgaussian random matrix with parameter s. There exists a universal constant C, which is very close to $1/\sqrt{2\pi}$ such that for any $t \geq 0$, we have $s_1(\mathbf{X}) \leq C \cdot s \cdot (\sqrt{m} + \sqrt{n} + t)$ except with probability at most $2\exp(\delta)\exp(-\pi t^2)$.*

Trapdoors and Sampling Algorithms. Here we recall some lattices trapdoors and gaussian sampling algorithms.

The gadget matrix $\mathbf{G} \in \mathbb{Z}_q^{n\times m}$ is a primitive matrix defined by gadget vector $\mathbf{g}$ as $\mathbf{G} := \mathbf{I}_n \otimes \mathbf{g}^t \in \mathbb{Z}_q^{n\times nk}$. We usually consider gadget vector $\mathbf{g}^t := [1\ 2\ 4\ \cdots\ 2^{k-1}] \in \mathbb{Z}_q^{1\times k}$, where $k = \lceil \log_2 q \rceil$.

Lemma 6 ([MP12, Theorem 4.1]). *Let n, m, k, q be any integers with $q \geq 2$, $n \geq 1$, $m = nk$ and $k = \lceil \log_2 q \rceil$, then there is a primitive matrix $\mathbf{G} \in \mathbb{Z}_q^{n\times m}$ such that the lattice $\Lambda^\perp(\mathbf{G})$ has a known basis $\mathbf{S} \in \mathbb{Z}^{m\times m}$ with $\|\widetilde{\mathbf{S}}\| \leq \sqrt{5}$ and $\|\mathbf{S}\| \leq \max\{\sqrt{5}, \sqrt{k}\}$.*

Lemma 7 (TrapGen [MP12]). *There is a probabilistic polynomial-time algorithm $\mathsf{TrapGen}(1^n, 1^m, s, q)$ that, given any integers $n \geq 1$, $q \geq 2$, $s > 0$ and sufficiently large $m = O(n \log q)$, outputs a parity-check matrix $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ and a trapdoor $\mathbf{R} \in \mathbb{Z}^{(m-w)\times w}$ where $w := n\lceil \log_2 q \rceil$, such that the distribution of $\mathbf{A}$ is statistically close to uniform, and the entries of $\mathbf{R}$ are sampled from $\mathcal{D}_{\mathbb{Z},s}$ such that $s_1(\mathbf{R}) = s \cdot O(\sqrt{m-w} + \sqrt{w})$.*

Lemma 8 (SamplePre [MP12, PPS21]). *Let $q \geq 2$, let $\mathbf{R}$ be a trapdoor for matrix $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$, for any $\mathbf{u} \in \mathbb{Z}_q^n$ and $s \geq (s_1(\mathbf{R})^2 + 1) \cdot \|\widetilde{\mathbf{S}}\| \cdot \omega(\sqrt{\log n})$, there exists a p.p.t. algorithm $\mathsf{SamplePre}(\mathbf{A}, \mathbf{R}, \mathbf{u}, s)$ that samples preimages from a distribution which is statistically close to $\mathcal{D}_{\Lambda_\mathbf{u}^\perp(\mathbf{A}),s}$.*

In particular, the output distribution of the following two experiments are with $\mathsf{negl}(n)$ statistical distance:

- *choose $\mathbf{z} \leftarrow \mathcal{D}_{\mathbb{Z},s}^m$, and output $(\mathbf{z}, \mathbf{u} = \mathbf{A} \cdot \mathbf{z} \in \mathbb{Z}_q^n)$.*
- *choose uniformly random $\mathbf{u} \xleftarrow{\$} \mathbb{Z}_q^n$ and $\mathbf{z} \leftarrow \mathsf{SamplePre}(\mathbf{A}, \mathbf{R}, \mathbf{u}, s)$, and output $(\mathbf{e}, \mathbf{u})$.*

Lemma 9 (DelTrap [MP12]). *Let $q \geq 2$, for any pair of public matrix and its trapdoor $(\mathbf{A} \in \mathbb{Z}^{n\times m}, \mathbf{R})$ generated from $\mathsf{TrapGen}$ algorithm in Lemma 7, any extension matrix $\mathbf{A}_1 \in \mathbb{Z}^{n\times w}$, and $m' \geq m + w$, $s' \geq \omega(\sqrt{\log m})$, there exists a p.p.t. algorithm $\mathsf{DelTrap}(\mathbf{A}' = [\mathbf{A} \mid \mathbf{A}_1], \mathbf{R}, s')$ that outputs a trapdoor $\mathbf{R}'$ for $\mathbf{A}'$ and $s_1(\mathbf{R}') \leq s' \cdot O(\sqrt{m} + \sqrt{w})$ with overwhelming probability. Usually, s' is required to be sufficiently large relative to $s_1(\mathbf{R})$ when implementing algorithm.*

Hardness Assumption.

Definition 10 (SIS [Ajt96,MR04]). *The short integer solution problem $SIS_{n,m,q,\beta}$ (in the l_2 norm) is defined as follows: Given an integer q, m uniformly random vectors $\mathbf{a}_i \in \mathbb{Z}_q^n$, forming the columns of a matrix $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$, find a nonzero integer vector $\mathbf{z} \in \mathbb{Z}^m$ of norm $\|\mathbf{z}\| \leq \beta$ such that $\mathbf{Az} = \sum_i \mathbf{a}_i \cdot z_i = \mathbf{0} \mod q$.*

2.3 Hierarchical Identity-based Encryption

We now recall the definition of (anonymous) 2-level hierarchical identity-based encryption (HIBE) as we require it as building block to constructing IB-ME. Since we focus on 2-level case, we consider the encryption and decryption algorithms only for level-2 users. Such modifications apply to all existing HIBE schemes.

Definition 11 (2-level Hierarchical Identity-based Encryption [Gen06,ABB10b,KMT19]). *A 2-level HIBE scheme over a message space $\mathcal{M}$, an identity space $\mathcal{ID}$ is a tuple of algorithms $\Pi_{HIBE} = (Setup, Extract, Derive, Enc, Dec)$ with the following properties:*

- $Setup(1^\lambda) \to (mpk, msk)$*: On input the security parameter λ, the setup algorithm outputs the master public key mpk and the master secret key msk.*
- $Extract(msk, id_1) \to sk_{id_1}$*: On input the master secret key msk and a first-level identity $id_1 \in \mathcal{ID}$, the key-extract algorithm outputs a secret key sk_{id_1}.*
- $Derive(sk_{id_1}, id_2) \to sk_{id_1|id_2}$*: On input a secret key sk_{id_1} and a second-level identity $id_2 \in \mathcal{ID}$, outputs a secret key $sk_{id_1|id_2}$.*
- $Enc(mpk, id = (id_1 \mid id_2), m) \to ct$*: On input the master public key mpk, a level-2 user's identity $id = (id_1 \mid id_2) \in (\mathcal{ID})^2$ and a message $m \in \mathcal{M}$, outputs a ciphertext ct.*
- $Dec(sk_{id_1|id_2}, ct) \to m/\perp$*: On input a secret decryption key $sk_{id_1|id_2}$ (for a level-2 user with $id = (id_1 \mid id_2)$) and a ciphertext ct, outputs either a message $m \in \mathcal{M}$ or a special symbol $\perp$.*

A 2-level HIBE scheme should satisfy the following properties:

Correctness. For all identities $\mathsf{id} = (\mathsf{id}_1 \mid \mathsf{id}_2) \in (\mathcal{ID})^2$, and all messages $m \in \mathcal{M}$, if we set $(\mathsf{mpk}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda)$, $\mathsf{sk}_{\mathsf{id}_1} \leftarrow \mathsf{Extract}(\mathsf{msk}, \mathsf{id}_1)$, $\mathsf{sk}_{\mathsf{id}_1|\mathsf{id}_2} \leftarrow \mathsf{Derive}(\mathsf{sk}_{\mathsf{id}_1}, \mathsf{id}_2)$, $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{mpk}, \mathsf{id} = (\mathsf{id}_1 \mid \mathsf{id}_2), m)$, it holds that

$$\Pr[\mathsf{Dec}(\mathsf{sk}_{\mathsf{id}_1|\mathsf{id}_2}, \mathsf{ct}) = \mathrm{m}] \geq 1 - \mathsf{negl}(\lambda)$$

Security. We define chosen-plaintext security for HIBE systems under chosen identity attacks via the following game $\mathbf{IND\text{-}ID\text{-}CPA}^{\mathcal{A}}_{\mathsf{HIBE}}(\lambda)$.

- **Setup**: The challenger runs the Setup algorithm given it the security parameter λ as input. It gives the adversary the resulting master public key mpk. It keeps the master secret key msk to itself.
- **Pre-challenge querying Phase**:
  1. Level-1 secret key query
     The Adversary issues queries on identities $\mathsf{id}_1^1, \mathsf{id}_1^2, \cdots$ where each $\mathsf{id}_1^i \in \mathcal{ID}$. For each query the challenger executes $\mathsf{sk}_{\mathsf{id}_1^i} \leftarrow \mathsf{Extract}(\mathsf{msk}, \mathsf{id}_1^i)$ and returns $\mathsf{sk}_{\mathsf{id}_1^i}$ to the adversary.
  2. Level-2 secret key query
     The Adversary issues queries on identities $(\mathsf{id}_1^1 \mid \mathsf{id}_2^1), (\mathsf{id}_1^2 \mid \mathsf{id}_2^2), \cdots$ where each $(\mathsf{id}_1^j \mid \mathsf{id}_2^j) \in (\mathcal{ID})^2$. For each query the challenger executes $\mathsf{sk}_{\mathsf{id}_1^j} \leftarrow \mathsf{Extract}(\mathsf{msk}, \mathsf{id}_1^j)$ and $\mathsf{sk}_{\mathsf{id}_1^j|\mathsf{id}_2^j} \leftarrow \mathsf{Derive}(\mathsf{sk}_{\mathsf{id}_1^j}, \mathsf{id}_2^j)$. Then the challenger returns $\mathsf{sk}_{\mathsf{id}_1^j|\mathsf{id}_2^j}$ to the adversary.
- **Challenge Phase**:
  Once the adversary decides that pre-challenge querying phase is over, it submits an identity $\mathsf{id}^* = (\mathsf{id}_1^* \mid \mathsf{id}_2^*) \in (\mathcal{ID})^2$ and a message $m \in \mathcal{M}$. The challenge identity id^* and its prefix must not have appeared in any secret key query (both level-1 and level-2) in pre-challenge querying phase. The challenger picks a random bit $b \in \{0, 1\}$ and a random ciphertext ct from ciphertext space. If $b = 0$, it sets the challenge ciphertext to $\mathsf{ct}^* := \mathsf{Enc}(\mathsf{mpk}, \mathsf{id}^* = (\mathsf{id}_1^* \mid \mathsf{id}_2^*), m)$. If $b = 1$, it sets $\mathsf{ct}^* := \mathsf{ct}$. The challenger then sends ct^* to the adversary.

- **Post-challenge querying phase**:
  The adversary issues additional level-1 and level-2 key queries as in the pre-challenge querying phase, and the challenger responds as before, except that the adversary should not request a secret key for id^* or the prefix of id^*.
- **Guess**: Finally, the adversary submits a guess $b' \in \{0,1\}$ and wins if $b = b'$.

We refer to such an adversary as an IND-ID-CPA adversary.

Definition 12 (IND-ID-CPA secure 2-level HIBE). *A 2-level HIBE is IND-ID-CPA secure if for all p.p.t. adversaries $\mathcal{A}$,*

$$Adv^{IND\text{-}ID\text{-}CPA}_{\Pi,\mathcal{A}}(\lambda) := |\Pr[b = b'] - \tfrac{1}{2}| \leq negl(\lambda).$$

We define the anonymous property to the above adaptive-identity secure HIBE by slightly adjusting the challenge phase and post-challenge phase. Concretely, in the game $\mathbf{ANON\text{-}IND\text{-}ID\text{-}CPA}^{\mathcal{A}}_{\mathsf{HIBE}}(\lambda)$, the adversary submits two identities $\mathsf{id}^* = (\mathsf{id}_1^* \mid \mathsf{id}_2^*)$ and $\mathsf{id}^\dagger = (\mathsf{id}_1^\dagger \mid \mathsf{id}_2^\dagger)$ and two message m^* and $m^\dagger$. The challenge identities and corresponding prefixes may not been queried in the pre-challenge phase. After received the challenge tuple, the challenger picks a random bit $b \in \{0,1\}$. If $b = 0$, the challenger computes $\mathsf{ct} := \mathsf{Enc}(\mathsf{mpk}, \mathsf{id}^* = (\mathsf{id}_1^* \mid \mathsf{id}_2^*), m^*)$, otherwise computes $\mathsf{ct} := \mathsf{Enc}(\mathsf{mpk}, \mathsf{id}^\dagger = (\mathsf{id}_1^\dagger \mid \mathsf{id}_2^\dagger), m^\dagger)$, and then sends the resulting ciphertext to adversary. Post-challenge phase is the same as pre-challenge, except that the adversary cannot request a secret key for id^* or $\mathsf{id}^\dagger$, or their prefixes. At the end of the game, the adversary outputs its guess $b' \in \{0,1\}$ and wins if $b = b'$. It is clear that the ANON-IND-ID-CPA security implies the IND-ID-CPA security.

Definition 13 (ANON-IND-ID-CPA secure 2-level HIBE). *A 2-level HIBE is ANON-IND-ID-CPA secure if for all p.p.t. adversaries $\mathcal{A}$,*

$$Adv^{ANON\text{-}IND\text{-}ID\text{-}CPA}_{\Pi,\mathcal{A}}(\lambda) := |\Pr[b = b'] - \tfrac{1}{2}| \leq negl(\lambda).$$

2.4 Identity-based Signature

Definition 14 (Identity-based Signature [Sha84, PW21]). *An IBS scheme is specified by four algorithms IBS = (Setup, KeyExt, Sign, Ver) with running time polynomial in the security parameter. The first three may be randomized while the last is deterministic.*

- $Setup(1^\lambda) \to (mpk, msk)$*: The trusted authority takes security parameters as input and run the setup algorithm to obtain a master public key mpk and a master secret key msk. We assume that mpk implicitly specified a message space $\mathcal{M}$ and identity space $\mathcal{ID}$.*
- $KeyExt(msk, id) \to sk_{id}$*: To generate secret signing key for user with identity $id \in \mathcal{ID}$, the authority revokes the key-extract algorithm on input a master secret key msk and an identity id and output secret signing key sk_{id}.*
- $Sign(sk_{id}, m) \to \sigma$*: On input secret signing key sk_{id} and message $m \in \mathcal{M}$, user with identity id will obtain a signature σ, which is the output of signing algorithm.*
- $Ver(mpk, id, m, \sigma) \to 0/1$*: On input a master public key mpk, a user identity id, a message m and a signature σ, verifying algorithm returns 1 if signature is valid for id and m, otherwise returns 0.*

Correctness. For every $(\mathsf{mpk}, \mathsf{msk})$ generated as above, $m \in \mathcal{M}$, $\mathsf{id} \in \mathcal{ID}$, we have:

$$\Pr[\mathsf{Ver}(\mathsf{mpk}, \mathsf{id}, \mathrm{m}, \sigma) = 1 \mid \mathsf{sk}_{\mathsf{id}} \leftarrow \mathsf{KeyExt}(\mathsf{msk}, \mathsf{id}), \sigma \leftarrow \mathsf{Sign}(\mathsf{sk}_{\mathsf{id}}, m)] = 1 - \mathsf{negl}(\lambda)$$

Security. We define the unforgeability for IBS systems under chosen message attacks via the following game $\mathbf{UF\text{-}CMA}^{\mathcal{A}}_{\mathsf{IBS}}(\lambda)$. Oracles Key, Sig are implemented by $\mathsf{KeyExt}(\cdot)$, $\mathsf{Sign}(\cdot)$, respectively. Lists $\mathcal{L}_{id}$ and $\mathcal{L}_m$ are updated after each query.

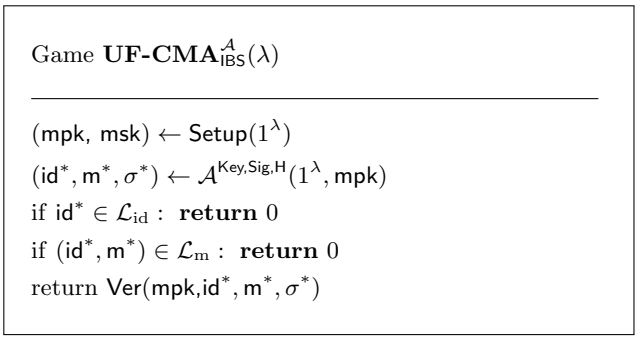

Fig. 1. Game $\mathbf{UF\text{-}CMA}^{\mathcal{A}}_{\mathsf{IBS}}(\lambda)$

Definition 15 (UF-(na)CMA). *Let IBS = (Setup, KeyExt, Sign, Ver) be an IBS scheme. We say that the IBS scheme is UF-CMA secure, if for every p.p.t algorithm $\mathcal{A}$, the following advantage is negligible in λ:*

$$Adv^{UF\text{-}CMA}_{\mathcal{A},IBS} := \Pr[\boldsymbol{UF\text{-}CMA}^{\mathcal{A}}_{IBS}(\lambda) \Rightarrow 1]$$

UF-naCMA security is defined similarly, but with additional restriction that the adversary should submit all the signing key queries $\mathcal{L}_{id}$ and $\mathcal{L}_m$ signature queries before setup phase.

We say that the IBS scheme is UF-naCMA secure, if for every p.p.t algorithm $\mathcal{A}$, the following advantage is negligible in λ:

$$Adv^{UF\text{-}naCMA}_{\mathcal{A},IBS} := \Pr[\boldsymbol{UF\text{-}naCMA}^{\mathcal{A}}_{IBS}(\lambda) \Rightarrow 1]$$

3 Improved Formal Definitions for Identity-based Matchmaking Encryption

In this section, we will firstly recall the syntax and the formal definition of correctness for IB-ME. Then we will propose the improved security definitions to match general and practical security requirements. As we will discuss in detail later, our security definition is able to capture more real-world attacks than previous ones [AFNV19, FGRV21, CLWW22].

Definition 16 (Identity-based Matchmaking Encryption [AFNV19]). *A IB-ME scheme for a set of identities $\mathcal{ID}$ and a message space $\mathcal{M}$ is a tuple of polynomial-time algorithms IB-ME = (Setup, SKGen, RKGen, Enc, Dec) defined as follows:*

- $\mathsf{Setup}(1^\lambda) \to (\mathsf{mpk}, \mathsf{msk})$*: Upon input the security parameter 1^λ, the setup algorithm outputs the master public key* mpk *and the master secret key* msk*.*
- $\mathsf{SKGen}(\mathsf{msk}, \phi) \to \mathsf{ek}_\phi$*: The sender-key generator takes as input the master secret key* msk*, and identity ϕ. The algorithm outputs a secret encryption key ek_ϕ for identity ϕ.*
- $\mathsf{RKGen}(\mathsf{msk}, \rho) \to \mathsf{dk}_\rho$*: The receiver-key generator takes as input the master secret key* msk*, and an identity ρ. The algorithm outputs a secret decryption key dk_ρ for identity ρ.*
- $\mathsf{Enc}(\mathsf{ek}_\phi, \mathsf{rcv}, m) \to c$*: The encryption algorithm takes as input a secret encryption key ek_ϕ for identity ϕ, a target receiver's identity* rcv*, and a message $m \in \mathcal{M}$. The algorithm produces a ciphertext c linked to both ϕ and* rcv*.*
- $\mathsf{Dec}(\mathsf{dk}_\rho, \mathsf{snd}, c) \to m/\perp$*: On input a secret decryption key dk_ρ for identity ρ, a target sender's identity* snd *and a ciphertext c, the decryption algorithm outputs either a message m or $\perp$.*

Correctness. Intuitively, the output of decryption algorithm for the ciphertext encrypted under encryption key for ϕ and target identity rcv using decryption key for ρ and target identity snd will be the original plaintext if and only if the receiver's identity matches the identity rcv chosen by the encryptor, and the sender's identity matches the identity snd selected by the decryptor in the meantime.

Definition 17 (Correctness of IB-ME). *For all messages $m \in \mathcal{M}$, all identities $\phi, \rho, \mathit{rcv}, \mathit{snd} \in \mathcal{ID}$ such that $\rho = \mathit{rcv} \wedge \phi = \mathit{snd}$, if we set $(\mathit{mpk}, \mathit{msk}) \leftarrow \mathit{Setup}(1^\lambda)$, $\mathit{ek}_\phi \leftarrow \mathit{SKGen}(\mathit{msk}, \phi)$, $\mathit{dk}_\rho \leftarrow \mathit{RKGen}(\mathit{msk}, \rho)$, then*

$$\Pr[\mathit{Dec}(\mathit{dk}_\rho, \mathit{snd}, \mathit{Enc}(\mathit{ek}_\phi, \mathit{rcv}, \mathrm{m})) = \mathrm{m}] \geq 1 - \mathit{negl}(\lambda)$$

Security Definitions. Here, we give an improved formal security definition for IB-ME. The security of an IB-ME scheme can be viewed as two properties, called *privacy* and *authenticity*.

Like the previous privacy of IB-ME, we focus on the privacy in the case of mismatch, which means that as long as the malicious receiver does not own the decryption key associated with the right identity, he cannot learn anything about message and the information about the sender's identity. The reason why we do not consider the condition of match is that match cases obviously imply $\rho = \mathsf{rcv} \wedge \phi = \mathsf{snd}$. We modified the original privacy game to allow additional ciphertext queries and the modified privacy game $\mathrm{G}^{\mathsf{new\text{-}priv}}_{\Pi,\mathcal{A}}$ is showed in Fig. 2.

Definition 18. *(Stronger Privacy of IB-ME). We say that an IB-ME Π satisfies stronger privacy if for all admissible p.p.t algorithms $\mathcal{A}$,*

$$|\Pr[\mathrm{G}^{\mathit{new\text{-}priv}}_{\Pi,\mathcal{A}} = 1] - \tfrac{1}{2}| \leq \mathit{negl}(\lambda).$$

Oracles O_S, O_R, O_E are implemented by $\mathit{SKGen}(\cdot)$, $\mathit{RKGen}(\cdot)$, $\mathit{Enc}(\cdot)$, respectively. And an adversary $\mathcal{A}$ is admissible if for all identities ρ the adversary submitted to the decryption key oracle, it holds that $\rho \neq \mathit{rcv}_0 \wedge \rho \neq \mathit{rcv}_1$.

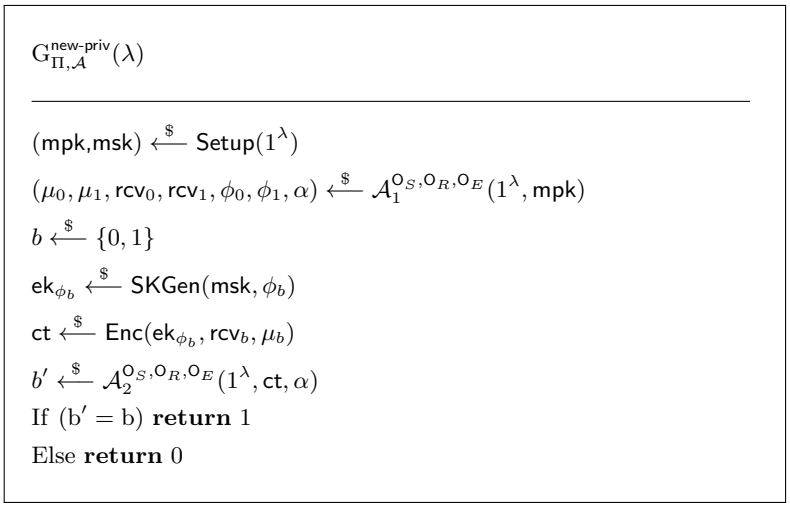

Fig. 2. $\mathrm{Game}^{\mathsf{new\text{-}priv}}_{\Pi,\mathcal{A}}(\lambda)$

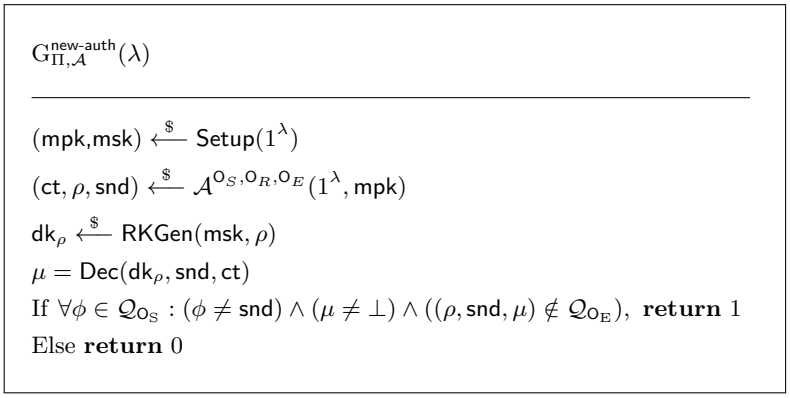

Fig. 3. $\mathrm{Game}^{\mathsf{new\text{-}auth}}_{\Pi,\mathcal{A}}(\lambda)$

Authenticity guarantees that adversary, without corresponding encryption keys, cannot produce ciphertexts embedding fake identities. We observe that the previous security definitions [AFNV19, FGRV21, CLWW22] fail to capture the possible forgeries like "forging-to-itself" or tamper forgeries. Hence, we modify the authenticity game to cancel the restrictions on the challenge receivers' identities and give adversaries access to encryption oracle. In other words, the improved game enables attackers to obtain ciphertexts with known plaintexts from chosen sources, also it is admissible for adversaries to submit forgeries that send to corrupted receivers. The modified authenticity game is presented in Fig. 3.

Definition 19. *(Stronger Authenticity of IB-ME). We say that an IB-ME* Π *satisfies stronger authenticity if for all p.p.t algorithms* $\mathcal{A}$*,*

$$| \Pr[\mathrm{G}^{\mathit{new\text{-}auth}}_{\Pi,\mathcal{A}} = 1] | \leq \mathit{negl}(\lambda).$$

4 Generic Construction of IB-ME

In this section, we provide the details about our generic construction of IB-ME satisfying stronger security definitions. Namely, we show how to construct an IB-ME scheme by combining a 2-level HIBE scheme and an IBS scheme. We require that both underlying schemes to be adaptively secure, so that IB-ME achieving adaptive security.

4.1 Constructing IB-ME from 2-level HIBE and IBS

Construction 1 (Identity-based Matchmaking Encryption)**.** Let HIBE and IBS be respectively an anonymous 2-level HIBE and a identity-based signature scheme, we construct our IB-ME with identity space $\mathcal{ID}$ and message space $\mathcal{M}$ as follows:

- $\mathsf{Setup}(1^\lambda)$: On input the security parameter 1^λ, the setup algorithm runs $(\mathsf{HIBE.mpk}, \mathsf{HIBE.msk}) \leftarrow \mathsf{HIBE.Setup}(1^\lambda)$, $(\mathsf{IBS.mpk}, \mathsf{IBS.msk}) \leftarrow \mathsf{IBS.Setup}\ (1^\lambda)$, and outputs
$$\mathsf{mpk} = (\mathsf{HIBE.mpk}, \mathsf{IBS.mpk}) \text{ and } \mathsf{msk} = (\mathsf{HIBE.msk}, \mathsf{IBS.msk})$$
- $\mathsf{SKGen}(\mathsf{msk}, \phi)$: On input the master secret key $\mathsf{msk} = (\mathsf{HIBE.msk}, \mathsf{IBS.msk})$ and an identity ϕ, the sender key-generation algorithm computes signing key $\mathsf{IBS.sk}_\phi \leftarrow \mathsf{IBS.KeyExt}(\mathsf{IBS.msk}, \phi)$. It outputs $\mathsf{ek}_\phi \leftarrow \mathsf{IBS.sk}_\phi$.
- $\mathsf{RKGen}(\mathsf{msk}, \rho)$: On input the master secret key $\mathsf{msk} = (\mathsf{HIBE.msk}, \mathsf{IBS.msk})$ and an identity ρ, the receiver key-generation algorithm computes $\mathsf{HIBE.sk}_\rho \leftarrow \mathsf{HIBE.Extract}(\mathsf{H\text{-}IBE.msk}, \rho)$. Then, it outputs $\mathsf{dk}_\rho \leftarrow \mathsf{HIBE.sk}_\rho$.
- $\mathsf{Enc}(\mathsf{mpk}, \mathsf{ek}_\phi, \mathsf{rcv}, \mu)$: On input the master public key $\mathsf{mpk} = (\mathsf{HIBE.mpk}, \mathsf{IBS.mpk})$, secret encryption key ek_ϕ, target identity rcv and message $\mu \in \mathcal{M}$, the encryption algorithm first generate identity-based signature $\mathbf{t} \leftarrow \mathsf{IBS.Sign}(\mathsf{IBS.sk}_\phi, \mathsf{rcv} \mid \mu)$. It then computes ciphertexts under public key $\mathsf{rcv} \mid \phi$ (for HIBE) to obtain
$$\mathsf{ct}^\mu \leftarrow \mathsf{HIBE.Enc}(\mathsf{HIBE.mpk}, \mathsf{rcv} \mid \phi, \mu), \ \ \mathsf{ct}^{\mathbf{t}} \leftarrow \mathsf{HIBE.Enc}(\mathsf{HIBE.mpk}, \mathsf{rcv} \mid \phi, \mathbf{t})$$
Finally, it outputs ciphertext $\mathsf{ct} = (\mathsf{ct}^\mu, \mathsf{ct}^{\mathbf{t}})$.
- $\mathsf{Dec}(\mathsf{dk}_\rho, \mathsf{snd}, \mathsf{ct})$: On input a secret decryption key dk_ρ, a ciphertext ct and a selected sender's identity snd, the decryption algorithm first delegates key $\mathsf{dk}_{\rho|\mathsf{snd}} \leftarrow \mathsf{H\text{-}IBE.Derive}(\mathsf{dk}_\rho, \rho \mid \mathsf{snd})$, and recovers $(\mathbf{t}, \mu) \leftarrow \mathsf{H\text{-}IBE.Dec}(\mathsf{dk}_{\rho|\mathsf{snd}}, \mathsf{ct})$. Then it verifies the validity of signature $(0/1) \leftarrow \mathsf{IBS.Verify}(\mathsf{IBS.pk}, \mathbf{t}, \rho \mid \mu)$. Finally, it outputs μ if signature is valid. Otherwise, it returns $\perp$.

Theorem 20 (Correctness). *If underlying HIBE and IBS are both correct, then IB-ME from Construction 1 is correct.*

Proof. Take a message $m \in \mathcal{M}$, an sender's identity $\phi \in \mathcal{ID}$, and a target receiver's identity $\mathsf{rcv} \in \mathcal{ID}$. Take a receiver's identity $\rho \in \mathcal{ID}$, and a target sender's identity $\mathsf{snd} \in \mathcal{ID}$. Take $(\mathsf{mpk}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda)$, $\mathsf{ek}_\phi \leftarrow \mathsf{SKGen}(\mathsf{msk}, \phi)$, $\mathsf{dk}_\rho \leftarrow \mathsf{RKGen}(\mathsf{msk}, \rho)$, $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{mpk}, \mathsf{ek}_\phi, \mathsf{rcv}, \mu)$.

In this case, $\mathsf{mpk} = (\mathsf{HIBE.mpk}, \mathsf{IBS.mpk})$, $\mathsf{msk} = (\mathsf{HIBE.msk}, \mathsf{IBS.msk})$, ek_ϕ is the signing key output by $\mathsf{IBS.KeyExt}(\mathsf{IBS.msk}, \phi)$, dk_ρ is the decryption key obtained by $\mathsf{HIBE.Extract}(\mathsf{H\text{-}IBE.msk}, \rho)$, and

$\mathsf{ct} = (\mathsf{ct}^{\mu}, \mathsf{ct}^{t})$ is output by $\mathsf{ct}^{\mu} \leftarrow \mathsf{HIBE.Enc}(\mathsf{HIBE.mpk}, \mathsf{rcv} \mid \phi, \mu)$ and $\mathsf{ct}^{\mathbf{t}} \leftarrow \mathsf{HIBE.Enc}(\mathsf{HIBE.mpk}, \mathsf{rcv} \mid \phi, \mathbf{t})$, respectively, where $\mathbf{t}$ is the identity-based signature corresponding to ϕ, rcv and μ. It is obvious that when $\rho = \mathsf{rcv} \wedge \phi = \mathsf{snd}$, $\mathsf{rcv} \mid \phi = \rho \mid \mathsf{snd}$, the receiver with identity ρ can generate decryption key for $\rho \mid \mathsf{snd}$ using its key dk_{ρ} and then recover message μ and signature $\mathbf{t}$. Moreover, signature $\mathbf{t}$ is signed by sender with identity ϕ for $\mathsf{rcv} \mid \mu$, we have $1 \leftarrow \mathsf{IBS.Verify}(\mathsf{IBS.pk}, \mathbf{t}, \rho \mid \mu)$ with overwhelming probability. Thus, the claim follows by correctness of HIBE and IBS. □

4.2 The Security Analysis

We provide the formal security analysis of the identity-based matchmaking encryption scheme from Construction 1.

Theorem 21 (Security). *Let HIBE, IBS be as above. Suppose that Π_{HIBE} is ANON-IND-ID-CPA secure (Def. 13), IBS is UF-CMA secure (Def. 15), then $\Pi_{\mathsf{IB\text{-}ME}}$ from Construction 1 is secure.*

Proof. We prove privacy and authenticity separately.

Lemma 22. *If Π_{HIBE} is ANON-IND-ID-CPA secure (Def. 13), then $\Pi_{\mathsf{IB\text{-}ME}}$ from Construction 1 satisfies stronger privacy(Def. 18). In particular, for every p.p.t. algorithm $\mathcal{A}$ there is a p.p.t. algorithm $\mathcal{B}$ such that*

$$\mathsf{Adv}^{\mathsf{new\text{-}priv}}_{\mathcal{A},\mathsf{IB\text{-}ME}}(\lambda) \leq \ 2\mathsf{Adv}^{\mathsf{IND\text{-}ID\text{-}CPA}}_{\mathcal{B}_1,\mathsf{HIBE}}(\lambda) + \mathsf{Adv}^{\mathsf{ANON\text{-}IND\text{-}ID\text{-}CPA}}_{\mathcal{B}_2,\mathsf{HIBE}}(\lambda) + \mathsf{negl}(\lambda).$$

Proof. We proceed with a hybrid argument:

- Hyb_0: This is experiment $\mathrm{G}^{\mathsf{new\text{-}priv}}_{\Pi,\mathcal{A}}(\lambda)$ for case that $b = 0$. Namely, the challenger responds to the challenge query $(\mu_0, \mu_1, \mathsf{rcv}_0, \mathsf{rcv}_1, \phi_0, \phi_1)$ with the ciphertext $\mathsf{ct} \xleftarrow{\$} \mathsf{Enc}(\mathsf{ek}_{\phi_0}, \mathsf{rcv}_0, \mu_0)$, where $\mathsf{ek}_{\phi_0} \xleftarrow{\$} \mathsf{SKGen}(\mathsf{msk}, \phi_0)$.
- Hyb_1: Same as Hyb_0, except the challenger constructs the challenger ciphertext as $\mathsf{ct} \leftarrow (\mathsf{ct}^{\mu}, \mathsf{ct}^{*})$, where the first part ct^{μ} is same as in the Hyb_0, and the second part ct^{*} is set as a random ciphertext from ciphertext space.
- Hyb_2: Same as Hyb_1, except the challenger constructs the first part of challenge ciphertext to be $\mathsf{ct}^{\mu} \leftarrow \mathsf{HIBE.Enc}(\mathsf{HIBE.mpk}, \mathsf{rcv}_1 \mid \phi_1, \mu_1)$, and sets second part the same as in the Hyb_1. Then, the challenger responds with $\mathsf{ct} \leftarrow (\mathsf{ct}^{\mu}, \mathsf{ct}^{*})$.
- Hyb_3: Same as Hyb_2, except the challenger constructs the challenger ciphertext as $\mathsf{ct} \xleftarrow{\$} \mathsf{Enc}(\mathsf{ek}_{\phi_1}, \mathsf{rcv}_1, \mu_1)$, where $\mathsf{ek}_{\phi_1} \xleftarrow{\$} \mathsf{SKGen}(\mathsf{msk}, \phi_1)$.

We denote the advantage of adversary $\mathcal{A}$ in each game as $\mathsf{Adv}_i(\mathcal{A}) := \Pr[\mathsf{Hyb}^{\mathcal{A}}_i \Rightarrow 1]$. Note that the Hyb_3 is the experiment $\mathrm{G}^{\mathsf{new\text{-}priv}}_{\Pi,\mathcal{A}}(\lambda)$ for case that $b = 1$. We can obtain the advantage of $\mathcal{A}$ by $\mathsf{Adv}^{\mathsf{new\text{-}priv}}_{\mathcal{A},\mathsf{IB\text{-}ME}}(\lambda) = \mid \mathsf{Adv}_0(\mathcal{A}) - \mathsf{Adv}_3(\mathcal{A}) \mid$.

We now show that each consecutive pair of hybrid experiments are computationally indistinguishable.

- Hybrids Hyb_0 and Hyb_1 are computationally indistinguishable by IND-ID-CPA security (Def. 12) of HIBE. Specifically, suppose that there exists an efficient adversary $\mathcal{A}$ that can distinguish Hyb_0 from Hyb_1. We then uses $\mathcal{A}$ to construct an adversary $\mathcal{B}_1$ for the IND-ID-CPA security game:
  1. At the beginning of the IND-ID-CPA security game, adversary $\mathcal{B}_1$ receives the public parameters $\mathsf{HIBE.mpk}$ from the IND-ID-CPA security challenger. In addition, it runs $\mathsf{IBS.Setup}$ to obtain $\mathsf{IBS.mpk}$ and $\mathsf{IBS.msk}$. Then adversary $\mathcal{B}_1$ sets $\mathsf{mpk} \leftarrow (\mathsf{HIBE.mpk}, \mathsf{IBS.mpk})$ and sends it to the adversary $\mathcal{A}$.
  2. Whenever $\mathcal{A}$ makes a decryption key query on an identity $\rho \in \mathcal{ID}$, algorithm $\mathcal{B}_1$ makes a key-generation query to its challenger on level-1 identity ρ to obtain a key $\mathsf{HIBE.sk}_{\rho}$, which it forwards to $\mathcal{A}$. Whenever $\mathcal{A}$ makes a encryption key query on an identity $\phi \in \mathcal{ID}$, algorithm $\mathcal{B}_1$ computes $\mathsf{IBS.sk}_{\phi} \leftarrow \mathsf{IBS.KeyExt}(\mathsf{IBS.msk}, \phi)$. It sets $\mathsf{ek}_{\phi} \leftarrow \mathsf{IBS.sk}_{\phi}$ and sends it to $\mathcal{A}$. Whenever $\mathcal{A}$ makes a ciphertext query on a tuple of $(\phi, \mathsf{rcv}, \mu)$, algorithm $\mathcal{B}$ first generates $\mathsf{IBS.sk}_{\phi} \leftarrow \mathsf{IBS.KeyExt}(\mathsf{IBS.msk}, \phi)$ and computes $\mathbf{t} \leftarrow \mathsf{IBS.Sign}(\mathsf{IBS.sk}_{\phi}, \mathsf{rcv} \mid \mu)$, then computes ct^{μ} and $\mathsf{ct}^{\mathbf{t}}$ as in the real Enc. Finally, it sets $\mathsf{ct} \leftarrow (\mathsf{ct}^{\mu}, \mathsf{ct}^{\mathbf{t}})$ and sends it to $\mathcal{A}$.

3. Whenever $\mathcal{A}$ makes a challenge query on input $(\mu_0, \mu_1, \mathsf{rcv}_0, \mathsf{rcv}_1, \phi_0, \phi_1)$, algorithm $\mathcal{B}_1$ first selects random bit $b \in \{0,1\}$, computes $\mathsf{IBS.sk}_{\phi_b} \leftarrow \mathsf{IBS.KeyExt}(\mathsf{IBS.msk}, \phi_b)$ and $\mathbf{t} \leftarrow \mathsf{IBS.Sign}(\mathsf{IBS.sk}_{\phi_b}, \mathsf{rcv}_b \mid \mu_b)$, and then obtains $\mathsf{ct}^{\mu_b} \leftarrow \mathsf{HIBE.Enc}(\mathsf{HIBE.mpk}, \mathsf{rcv}_b \mid \phi_b, \mu_b)$. The adversary $\mathcal{B}_1$ submits the pair $(\mathsf{rcv}_b \mid \phi_b, \mathbf{t})$ to the challenger as challenge query. The challenger replies to $\mathcal{B}_1$ with ct^*. Then, the algorithm $\mathcal{B}_1$ sets the challenge ciphertext as $\mathsf{ct} \leftarrow (\mathsf{ct}^{\mu_b}, \mathsf{ct}^*)$ and sends it to adversary $\mathcal{A}$.
4. At the end of the game, algorithm $\mathcal{B}_1$ outputs what $\mathcal{A}$ outputs.

We first argue that $\mathcal{B}_1$ is admissible for the IND-ID-CPA game. Since $\mathcal{A}$ is admissible for the privacy game, this means that for all decryption key generation queries $\rho \in \mathcal{ID}$ that $\mathcal{A}$ makes, it must satisfy that $\rho \neq \mathsf{rcv}_0 \wedge \rho \neq \mathsf{rcv}_1$. Due to the reason that the challenge query submitted by $\mathcal{B}_1$ forms of $(\mathsf{rcv}_b \mid \phi_b, \mathbf{t})$ and all key generation queries it issued are exactly identities ρ which queried by $\mathcal{A}$, this means that $\mathcal{B}_1$ never asked secret key for challenge identity or its prefix. Thus, $\mathcal{B}_1$ is admissible for the IND-ID-CPA game.

By construction, if $\mathsf{ct}^* \leftarrow \mathsf{HIBE.Enc}(\mathsf{HIBE.mpk}, \mathsf{rcv}_b \mid \phi_b, \mathbf{t})$, then $\mathcal{B}_1$ perfectly simulated Hyb_0 for $\mathcal{A}$, and if ct^* is the random ciphertext chosen by challenger, then $\mathcal{B}_1$ perfectly simulated Hyb_1 for $\mathcal{A}$. Thus,

$$| \mathsf{Adv}_0(\mathcal{A}) - \mathsf{Adv}_1(\mathcal{A}) | \leq \mathsf{Adv}^{\mathsf{IND\text{-}ID\text{-}CPA}}_{\mathcal{B}_1,\mathsf{HIBE}}(\lambda)$$

- Hybrids Hyb_1 and Hyb_2 are computationally indistinguishable by ANON-IND-ID-CPA security (Def. 13) of HIBE. Specifically, suppose that there exists an efficient adversary $\mathcal{A}$ that can distinguish Hyb_1 from Hyb_2. We then uses $\mathcal{A}$ to construct an adversary $\mathcal{B}_2$ for the ANON-IND-ID-CPA security game:
  1. At the beginning of the ANON-IND-ID-CPA security game, adversary $\mathcal{B}_2$ receives the public parameters $\mathsf{HIBE.mpk}$ from the ANON-IND-ID-CPA security challenger. In addition, it runs $\mathsf{IBS.Setup}$ to obtain $\mathsf{IBS.mpk}$ and $\mathsf{IBS.msk}$. Then adversary $\mathcal{B}_2$ sets $\mathsf{mpk} \leftarrow \mathsf{HIBE.mpk}, \mathsf{IBS.mpk}$ and sends it to the adversary $\mathcal{A}$.
  2. Whenever $\mathcal{A}$ makes a decryption key query on an identity $\rho \in \mathcal{ID}$, algorithm $\mathcal{B}_2$ then makes a key-generation query to its challenger on level-1 identity ρ to obtain a key $\mathsf{HIBE.sk}_\rho$, which it forwards to $\mathcal{A}$. Whenever $\mathcal{A}$ makes a encryption key query on an identity $\phi \in \mathcal{ID}$, algorithm $\mathcal{B}_2$ computes $\mathsf{IBS.sk}_\phi \leftarrow \mathsf{IBS.KeyExt}(\mathsf{IBS.msk}, \phi)$. It then sets $\mathsf{ek}_\phi \leftarrow \mathsf{IBS.sk}_\phi$ and sends it to $\mathcal{A}$. Whenever $\mathcal{A}$ makes a ciphertext query on a tuple of $(\phi, \mathsf{rcv}, \mu)$, algorithm $\mathcal{B}_2$ first generates $\mathsf{IBS.sk}_\phi \leftarrow \mathsf{IBS.KeyExt}(\mathsf{IBS.msk}, \phi)$ and computes $\mathbf{t} \leftarrow \mathsf{IBS.Sign}(\mathsf{IBS.sk}_\phi, \mathsf{rcv} \mid \mu)$ It then computes ct^μ and $\mathsf{ct}^{\mathbf{t}}$ as in the real Enc algorithm. Finally, it sets $\mathsf{ct} \leftarrow (\mathsf{ct}^\mu, \mathsf{ct}^{\mathbf{t}})$ and sends back to $\mathcal{A}$.
  3. Whenever $\mathcal{A}$ makes a challenge query on input $(\mu_0, \mu_1, \mathsf{rcv}_0, \mathsf{rcv}_1, \phi_0, \phi_1)$, algorithm $\mathcal{B}_2$ first selects random ct^* from ciphertext space. Then it submits the challenge query $(\mathsf{rcv}_0 \mid, \phi_0, \mathsf{rcv}_1 \mid \phi_1)$ and $(\mu_0, \mu_1,)$ to the challenger. The challenger replies to $\mathcal{B}_2$ with $\mathsf{ct}^{\mu_{b'}}$. Then, the algorithm $\mathcal{B}_2$ sets the challenge ciphertext as $\mathsf{ct} \leftarrow (\mathsf{ct}^{\mu_{b'}}, \mathsf{ct}^*)$ and sends it to $\mathcal{A}$.
  4. At the end of the game, algorithm $\mathcal{B}_2$ outputs what $\mathcal{A}$ outputs.

  Similarly, $\mathcal{B}_2$ is admissible for the ANON-IND-ID-CPA game. By construction, if $\mathsf{ct}^{\mu_{b'}} \leftarrow \mathsf{HIBE.Enc}(\mathsf{HIBE.mpk}, \mathsf{rcv}_0 \mid \phi_b, \mu_1)$, then $\mathcal{B}_2$ perfectly simulated Hyb_1 for $\mathcal{A}$, and if $\mathsf{ct}^{\mu_{b'}} \leftarrow \mathsf{HIBE.Enc}(\mathsf{HIBE.mpk}, \mathsf{rcv}_1 \mid \phi_1, \mu_1)$, then $\mathcal{B}_2$ perfectly simulated Hyb_2 for $\mathcal{A}$. Thus,

$$| \mathsf{Adv}_1(\mathcal{A}) - \mathsf{Adv}_2(\mathcal{A}) | \leq \mathsf{Adv}^{\mathsf{ANON\text{-}IND\text{-}ID\text{-}CPA}}_{\mathcal{B}_2,\mathsf{HIBE}}(\lambda)$$

- Hybrids Hyb_2 and Hyb_3 are computationally indistinguishable by IND-ID-CPA security (Def. 12) of HIBE via the same argument used to show indistinguishability of hybrids Hyb_0 and Hyb_1. Thus,

$$| \mathsf{Adv}_2(\mathcal{A}) - \mathsf{Adv}_3(\mathcal{A}) | \leq \mathsf{Adv}^{\mathsf{IND\text{-}ID\text{-}CPA}}_{\mathcal{B}_1,\mathsf{HIBE}}(\lambda)$$

Thus, we can bound that

$$\mathsf{Adv}^{\mathsf{new\text{-}priv}}_{\mathcal{A},\mathsf{IB\text{-}ME}}(\lambda) \leq \ 2\mathsf{Adv}^{\mathsf{IND\text{-}ID\text{-}CPA}}_{\mathcal{B}_1,\mathsf{HIBE}}(\lambda) + \mathsf{Adv}^{\mathsf{ANON\text{-}IND\text{-}ID\text{-}CPA}}_{\mathcal{B}_2,\mathsf{HIBE}}(\lambda) + \mathsf{negl}(\lambda).$$

□

Lemma 23. *If Π_{IBS} is UF-CMA secure (Def. 15), then $\Pi_{IB\text{-}ME}$ from Construction 1 satisfies stronger authenticity(Def. 19). In particular, for every p.p.t. algorithm $\mathcal{A}$ there is a p.p.t. algorithm $\mathcal{B}$ such that*

$$Adv^{new\text{-}auth}_{\mathcal{A},IB\text{-}ME}(\lambda) \leq Adv^{UF\text{-}CMA}_{\mathcal{B},IBS}(\lambda) + negl(\lambda).$$

Proof. The proof strategy of Lemma 23 is based on a contradiction, i.e. we assume that there exists an adversary $\mathcal{A}'$ which can break the authenticity of Construction 1 with non-negligible advantage, then we could build an attacker $\mathcal{B}'$ that breaks UF-CMA of IBS. And the reduction procedure is in the following way:

1. At the beginning, algorithm $\mathcal{B}'$ receives IBS.mpk from the challenger. Then, it executes (HIBE.mpk, HIBE.msk) $\leftarrow$ HIBE.Setup(1^λ), and sends mpk = (HIBE.mpk,IBS.mpk) to adversary $\mathcal{A}'$.
2. For the queries issued by $\mathcal{A}'$, $\mathcal{B}'$ proceeds as follows:
   - When $\mathcal{A}'$ issues encryption key queries for ϕ, $\mathcal{B}'$ queries its challenger for secret signing key on input identity ϕ. $\mathcal{B}'$ sets the ek_ϕ as the signing key sk_ϕ received from the challenger, and sends it back to $\mathcal{A}'$.
   - When $\mathcal{A}'$ issues decryption key queries for ρ, $\mathcal{B}'$ revokes $\mathsf{HIBE.sk}_\rho \leftarrow \mathsf{HIBE.Extract}(\mathsf{H\text{-}IBE.msk}, \rho)$, then sets $\mathsf{dk}_\rho \leftarrow \mathsf{HIBE.sk}_\rho$ and returns it to $\mathcal{A}'$.
   - When $\mathcal{A}'$ issues ciphertext queries for $(\phi, \mathsf{rcv}, \mu)$, $\mathcal{B}'$ first queries its challenger for signature on input $(\phi, \mathsf{rcv} \mid \mu)$ and receivers $\mathbf{t}$, then runs the encryption algorithm to obtain ct^μ and $\mathsf{ct}^{\mathbf{t}}$. Finally, it sends $\mathsf{ct} = (\mathsf{ct}^\mu, \mathsf{ct}^{\mathbf{t}})$ to $\mathcal{A}'$.
3. Once the algorithm $\mathcal{B}'$ receives the forgery output $(\mathsf{ct}, \rho, \mathsf{snd})$ from adversary $\mathcal{A}'$, $\mathcal{B}'$ executes in the following way:
   - If $\mathcal{A}'$ ever asked encryption key for snd, returns 0.
   - Else, $\mathcal{B}'$ firstly generate $\mathsf{dk}_\rho \leftarrow \mathsf{HIBE.Extract}(\mathsf{H\text{-}IBE.msk}, \rho)$ and delegates the level-2 decryption key $\mathsf{dk}_{\rho|\mathsf{snd}} \leftarrow \mathsf{H\text{-}IBE.Derive}(\mathsf{dk}_\rho, \rho \mid \mathsf{snd})$. Then, it recovers $(\mathbf{t}, \mu) \leftarrow \mathsf{H\text{-}IBE.Dec}(\mathsf{dk}_{\rho|\mathsf{snd}}, \mathsf{ct})$. It is clear that the decryption algorithm will output μ only when the signature $\mathbf{t}$ is valid corresponding to $(\mathsf{snd}, \rho \mid \mu)$.
   - If either $\mu = \perp$ or $\mathcal{A}'$ ever asked ciphertext for the same identities and message pair $(\mathsf{snd}, \rho \mid \mu)$, returns 0.
   - Else, $\mathcal{B}'$ returns $(\mathsf{snd}, \rho \mid \mu, \mathbf{t})$ as forgery signature to its challenger.

All the oracle queries of $\mathcal{A}'$ are perfectly simulated by $\mathcal{B}'$ helped by the secret key oracle and signing oracle of IBS challenger. The validity for forge signature is also obvious, because the valid conditions of authenticity forgery output already contains the checking conditions for IBS. Concretely, $\mathcal{A}'$ is not allowed to ask secret key for snd or signature for $(\mathsf{snd}, \rho \mid \mu)$, so as $\mathcal{B}'$. Thus, we extract $(\mathsf{snd}, \rho \mid \mu, \mathbf{t})$ as valid forgery breaking the UF-CMA of IBS. □

By combining Lemma 22 and Lemma 23, we can conclude that Construction 1 is secure. □

5 Adaptively Secure Identity-Based Signature

To improve the efficiency of IB-ME instantiation based on lattices assumptions, we shorten the signature size of an existing lattice-based IBS scheme to reduce the final IB-ME ciphertext sizes. Note that Pan and Wagner [PW21] proposed two generic transformations from non-adaptive IBS to adaptive one. We follow the same approach to obtain adaptive IBS. In other words, we pay attention to improving non-adaptive IBS from SIS assumptions respectively, and proving non-adaptive security. Using the lattice-based 2-level HIBE scheme of Agrawal *et al.* [ABB10b] and our improved IBS with adaptive security, our result implies the first lattice-based IB-ME.

5.1 Improved Non-adaptive IBS from SIS

We provide our improved SIS-based IBS scheme in Fig.4. The intuition of improvement is reduce the signature size by slightly changing the "hash-and-sign" approach in the signing algorithm.

Correctness. For correctness, we check that verification algorithm will accept valid signatures generated by user id with overwhelming probability.

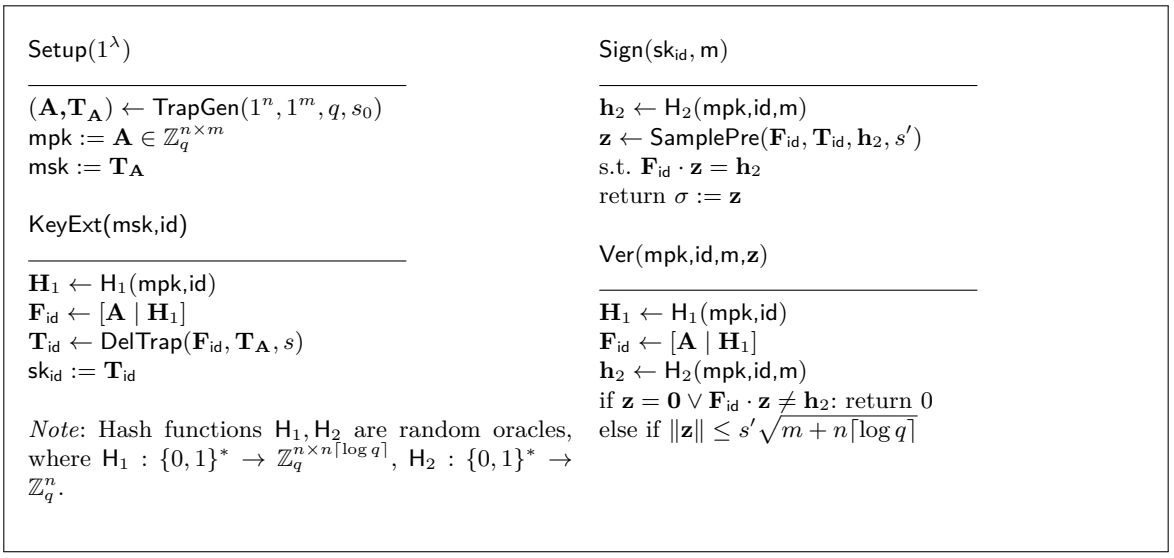

Fig. 4. Improved na-IBS$_{\mathsf{SIS}}$ = (Setup, KeyExt, Sign, Ver).

Lemma 24. *The identity-based signature scheme IBS$_{SIS}$ in Fig.4 is correct with overwhelming probability.*

Proof. For master public key and master secret key $(\mathbf{A}, \mathbf{T_A}) \leftarrow \mathsf{Setup}(1^\lambda)$, an identity $\mathsf{id} \in \mathcal{ID}$ and message $m \in \mathcal{M}$. Let $\mathsf{sk}_{\mathsf{id}} \leftarrow \mathsf{KeyExt}(\mathsf{msk},\mathsf{id})$ and $\mathbf{z} \leftarrow \mathsf{Sign}(\mathsf{sk}_{\mathsf{id}}, \mathsf{m})$. According to the definition of key generation algorithm, signing key $\mathsf{sk}_{\mathsf{id}}$ is the trapdoor for matrix $\mathbf{F}_{\mathsf{id}} := [\mathbf{A} \mid \mathbf{H}_1]$, which is an extension of public matrix $\mathbf{A}$. Thus, the trapdoor $\mathbf{T}_{\mathsf{id}}$ can be used to sample preimages in the signing algorithm. And the output vector $\mathbf{z}$ is sampled from the distribution statistically close to $\mathcal{D}_{\Lambda^\perp_{\mathbf{h}_2}(\mathbf{A}),s'}$ by Lemma 8. In other word, $\mathbf{z}$ satisfies $\mathbf{h}_2 = \mathbf{F}_{\mathsf{id}} \cdot \mathbf{z}$ and $\|\mathbf{z}\| \leq s'\sqrt{m + n\lceil\log q\rceil}$. □

Parameter Selection. We provide the parameters to satisfy following restrictions.

- Let $m \geq 2n\lceil\log q\rceil$ and $s_0 > 0$ so that algorithm TrapGen of Lemma 7 works as specified.
- Let $s \gg s_1(\mathbf{T_A})$ so that algorithm DelTrap of Lemma 9 works as specified.
- Let $s' \geq (s_1(\mathbf{T}_{\mathsf{id}})^2 + 1) \cdot \|\widetilde{\mathbf{S}}\| \cdot \omega(\sqrt{\log n})$ so that algorithm SamplePre of Lemma 8 works as specified.
- Let the modulus q be sufficiently large relative to β, so that the hardness assumption of related SIS problem applies.

An appropriate choice of parameters is as follows:

$$n = \mathsf{poly}(\lambda),\ m = O(n\lceil\log q\rceil),\ \beta = s \cdot s' \cdot O(m + n\lceil\log q\rceil)$$
$$s_0 = \omega(\sqrt{\log m}),\ s = s_0 \cdot O(\sqrt{m - n\lceil\log q\rceil} + \sqrt{n\lceil\log q\rceil})$$
$$s' = s^2 \cdot O(m + n\lceil\log q\rceil + \sqrt{m \cdot n\lceil\log q\rceil}) \cdot \omega(\sqrt{\log n})$$

We obtain the following keys and signature sizes:

- Master public key mpk is in $\mathbb{Z}_q^{n\times m}$ and hence has size $m \cdot n\lceil\log q\rceil$ bits.
- Master secret key msk is in $\mathbb{Z}^{(m-n\lceil\log q\rceil)\times(n\lceil\log q\rceil)}$ and hence has size $(m - n\lceil\log q\rceil) \times n \cdot \lceil\log q\rceil^2$ bits.
- Signing keys $\mathsf{sk}_{\mathsf{id}} = \mathbf{T}_{\mathsf{id}}$ are in $\mathbb{Z}^{m\times(n\lceil\log q\rceil)}$ and hence have size $m \cdot n \cdot \lceil\log q\rceil^2$ bits.
- Signatures $\mathbf{z}$ are in $\mathbb{Z}_q^{m+n\lceil\log q\rceil}$ and hence have size $(m+n\lceil\log q\rceil)\cdot\lceil\log q\rceil$ bits, which are $n\cdot\lceil\log q\rceil^2$ bits shorter than signatures in [PW21].

Remark 25. Due to space limit, we only focus on the improved IBS construction bases on SIS hardness. As to RSIS-based instantiation construction, the high-level construction idea and proving method are quite similar. And we can also reduce the signature size following the similar improvement approach. Concretely, the size of signatures can be $h \cdot n \cdot \lceil\log q\rceil$ bits shorter than that in [PW21], where h refers to the dimension of gadget vector $\mathbf{g}$.

5.2 The Security Proof

Here we prove the UF-naCMA security of $\mathsf{na\text{-}IBS_{SIS}}$ scheme showed in Fig.4. Using the generic transformations presented in [PW21], the final scheme will achieve UF-CMA security.

Theorem 26. *Assuming the hardness of* $\mathrm{SIS}_{n,m,q,\beta}$*, then the identity-based signature scheme described in Fig. 4 achieves UF-naCMA security (Def.15). In particular, for every p.p.t. algorithm* $\mathcal{A}$ *there is a p.p.t. algorithm* $\mathcal{B}$ *such that*

$$Adv^{UF\text{-}naCMA}_{\mathcal{A},na\text{-}IBS}(\lambda) \leq Adv^{SIS_{n,m,q,\beta}}_{\mathcal{B}}(\lambda) + negl(\lambda).$$

Proof. We prove the UF-naCMA security of $\mathsf{na\text{-}IBS_{SIS}}$ by constructing an algorithm $\mathcal{B}$, presented in Fig. 5, which can solve SIS problem by interacting with adversary $\mathcal{A}$. The details of reduction process are as follows:

At the beginning of security reduction, algorithm $\mathcal{B}$ was given random SIS problem instance $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ as input. $\mathcal{B}$ then sets $\mathbf{A}$ as master public key for IBS scheme and sends it to adversary $\mathcal{A}$. After receiving mpk, $\mathbf{A}$ sends lists $\mathcal{L}_{id}$ and $\mathcal{L}_m$ to $\mathcal{B}$.

Algorithm $\mathcal{B}$ (Given $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$)

1. $(\mathcal{L}_{id}, \mathcal{L}_m, St) \leftarrow \mathcal{A}(1^\lambda)$
2. $\mathsf{mpk} := \mathbf{A}$
3. **for** $\mathsf{id} \in \mathcal{L}_{id}$:
   - $\hat{\mathbf{R}}_{\mathsf{id}} \leftarrow \mathcal{D}^{m\times n\lceil\log q\rceil}_{\mathbb{Z},s}$
   - $h[1,\mathsf{mpk},\mathsf{id}] := \mathbf{A}\hat{\mathbf{R}}_{\mathsf{id}} + \mathbf{G}$
   - $\mathsf{sk}_{\mathsf{id}} := \hat{\mathbf{R}}_{\mathsf{id}}$
   - $\mathcal{L}_{sk} := \mathcal{L}_{sk} \cup \{\mathsf{sk}_{\mathsf{id}}\}$
4. **for** $(\mathsf{id}, m) \in \mathcal{L}_m$:
   - $\mathbf{H}_1 \leftarrow \mathsf{H}_1(\mathsf{mpk},\mathsf{id})$
   - $\mathbf{z}_{\mathsf{id},\mathsf{m}} \leftarrow \mathcal{D}^{m+n\lceil\log q\rceil}_{\mathbb{Z},s'}$
   - $h[2,\mathsf{mpk},\mathsf{id},\mathsf{m}] := [\mathbf{A} \mid \mathbf{H}_1] \cdot \mathbf{z}_{\mathsf{id},\mathsf{m}}$
   - $\mathcal{L}_{sig} := \mathcal{L}_{sig} \cup \{\mathbf{z}_{\mathsf{id},\mathsf{m}}\}$

Oracle $\mathsf{H}_1(\mathsf{mpk},\mathsf{id})$

if $h[1,\mathsf{mpk},\mathsf{id}] = \perp$:

- $\hat{\mathbf{R}}_{\mathsf{id}} \leftarrow \mathcal{D}^{m\times n\lceil\log q\rceil}_{\mathbb{Z},s}$
- $h[1,\mathsf{mpk},\mathsf{id}] := \mathbf{A}\hat{\mathbf{R}}_{\mathsf{id}}$

return $h[1,\mathsf{mpk},\mathsf{id}]$

5. $(\mathsf{id}^*, \mathsf{m}^*, \mathbf{z}^*) \leftarrow \mathcal{A}^{\mathsf{H}_1,\mathsf{H}_2}(St, \mathsf{mpk}, \mathcal{L}_{sk}, \mathcal{L}_{sig})$
6. **if** $\mathsf{id}^* \in \mathcal{L}_{id} \vee (\mathsf{id}^*, \mathsf{m}^*) \in \mathcal{L}_m$:
   return $\perp$
   if $\|\mathbf{z}^*\| > s'\sqrt{m + n\lceil\log q\rceil} \vee$
   $\mathbf{z}^* = \mathbf{0}$: **return** $\perp$
7. $\mathbf{F}_{\mathsf{id}^*} \leftarrow [\mathbf{A} \mid \mathbf{A}\hat{\mathbf{R}}_{\mathsf{id}^*}]$
   $\mathbf{h}_2 \leftarrow \mathsf{H}_2(\mathsf{mpk}, \mathsf{id}^*, \mathsf{m}^*)$
   if $\mathbf{F}_{\mathsf{id}^*}\mathbf{z}^* \neq \mathbf{h}_2$: **return** $\perp$
8. $\mathbf{z} := [\mathbf{I}_m \mid \hat{\mathbf{R}}_{\mathsf{id}^*}](\mathbf{z}^* - \mathbf{z}'_{\mathsf{id}^*,\mathsf{m}^*})$
   return $\mathbf{z}$

Oracle $\mathsf{H}_2(\mathsf{mpk},\mathsf{id},\mathsf{m})$

if $h[2,\mathsf{mpk},\mathsf{id},\mathsf{m}] = \perp$:

- $\mathbf{H}_1 \leftarrow \mathsf{H}_1(\mathsf{mpk},\mathsf{id})$
- $\mathbf{z}'_{\mathsf{id},\mathsf{m}} \leftarrow \mathcal{D}^{m+n\lceil\log q\rceil}_{\mathbb{Z},s'}$
- $h[2,\mathsf{mpk},\mathsf{id},\mathsf{m}] := [\mathbf{A} \mid \mathbf{H}_1] \cdot \mathbf{z}'_{\mathsf{id},\mathsf{m}}$
- $\mathcal{L}'_{sig} := \mathcal{L}'_{sig} \cup \{\mathbf{z}'_{\mathsf{id},\mathsf{m}}\}$

return $h[2,\mathsf{mpk},\mathsf{id},\mathsf{m}]$

Fig. 5. Algorithm $\mathcal{B}$ for solving $\mathrm{SIS}_{n,m,q,\beta}$ problem, using an adversary $\mathcal{A}$ against UF-naCMA security of $\mathsf{na\text{-}IBS_{SIS}}$

- For every identity $\mathsf{id} \in \mathcal{L}_{id}$, $\mathcal{B}$ firstly selects matrix $\hat{\mathbf{R}}_{\mathsf{id}}$ from $\mathcal{D}^{m\times n\lceil\log q\rceil}_{\mathbb{Z},s}$, and program the random oracle H_1 as $h[1,\mathsf{mpk},\mathsf{id}] := \mathbf{A}\hat{\mathbf{R}}_{\mathsf{id}} + \mathbf{G}$. Thus, $\hat{\mathbf{R}}_{\mathsf{id}}$ is a trapdoor for matrix $\mathbf{F}_{\mathsf{id}} := [\mathbf{A} \mid \mathsf{H}_1(\mathsf{mpk},\mathsf{id})]$, also a secret signing key $\mathsf{sk}_{\mathsf{id}}$ for identity id, as it supports a connection with gadget matrix $\mathbf{G}$ for further operation like Gaussian sampling. Moreover, according to the definition of UF-naCMA security, adversary has not queried random oracles until now, which means programming

is available. The distribution of the secret keys $\mathsf{sk}_{\mathsf{id}} := \hat{\mathbf{R}}_{\mathsf{id}}$ is statistically close to the real secret keys.

- For every identity and message pair $(\mathsf{id}, \mathsf{m}) \in \mathcal{L}_m$, $\mathcal{B}$ samples vector $\mathbf{z}_{\mathsf{id},\mathsf{m}}$ from distribution $\mathcal{D}_{\mathbb{Z},s'}^{m+n\lceil \log q \rceil}$, then program the random oracle as $h[2, \mathsf{mpk}, \mathsf{id}, \mathsf{m}] := [\mathbf{A} \mid \mathbf{H}_1] \cdot \mathbf{z}_{\mathsf{id},\mathsf{m}}$, and set vector $\mathbf{z}_{\mathsf{id},\mathsf{m}}$ to be the signature for $(\mathsf{id}, \mathsf{m})$. The programming is also available as the hash values have not been asked for. The signature generated by algorithm $\mathcal{B}$ is statistically close to the honest signatures.

After receiving the list of secret signing keys $\mathcal{L}_{sk}$ and signatures $\mathcal{L}_{sig}$, $\mathcal{A}$ queries random oracles H_1 and H_2 adaptively.

- For every identity id, for which adversary $\mathcal{A}$ queries for $\mathsf{H}_1(\mathsf{mpk}, \mathsf{id})$, algorithm $\mathcal{B}$ first checks whether the hash value $\mathsf{H}_1(\mathsf{mpk}, \mathsf{id})$ has been defined or not. If it has not been defined yet, then $\mathcal{B}$ draws matrix $\hat{\mathbf{R}}_{\mathsf{id}} \leftarrow \mathcal{D}_{\mathbb{Z},s}^{m \times n\lceil \log q \rceil}$, and programs $h[1, \mathsf{mpk}, \mathsf{id}] := \mathbf{A}\hat{\mathbf{R}}_{\mathsf{id}}$.
- For every identity and message pair $(\mathsf{id}, \mathsf{m})$, for which adversary $\mathcal{A}$ queries for $\mathsf{H}_2(\mathsf{mpk}, \mathsf{id}, \mathsf{m})$, algorithm $\mathcal{B}$ first checks whether the hash value $\mathsf{H}_2(\mathsf{mpk}, \mathsf{id}, \mathsf{m})$ has been defined or not. If it has not been defined yet, then $\mathcal{B}$ samples vector $\mathbf{z}'_{\mathsf{id},\mathsf{m}} \leftarrow \mathcal{D}_{\mathbb{Z},s'}^{m+n\lceil \log q \rceil}$, and programs $h[2, \mathsf{mpk}, \mathsf{id}, \mathsf{m}] := [\mathbf{A} \mid \mathbf{H}_1] \cdot \mathbf{z}'_{\mathsf{id},\mathsf{m}}$. Note that $\mathcal{B}$ only returns $\mathsf{H}_2(\mathsf{mpk}, \mathsf{id}, \mathsf{m})$ back to $\mathcal{A}$, and keeps $\mathbf{z}'_{\mathsf{id},\mathsf{m}}$ secret to its own.

At the end of UF-naCMA security game, adversary $\mathcal{A}$ outputs forgery signature $(\mathsf{id}^*, \mathsf{m}^*, \mathbf{z}^*)$. If the forgery is valid, in other words $\mathcal{A}$ wins the security game successfully, then by definition of UF-naCMA security $\mathsf{id}^* \notin \mathcal{L}_{id} \wedge (\mathsf{id}^*, \mathsf{m}^*) \notin \mathcal{L}_m$, and $[\mathbf{A} \mid \mathbf{A}\hat{\mathbf{R}}_{\mathsf{id}^*}]\mathbf{z}^* = \mathbf{h}_2$. Recall that when answering the query for random oracle H_2, challenger $\mathcal{B}$ additionally sampled a vector $\mathbf{z}'_{\mathsf{id}^*,\mathsf{m}^*}$ from gaussian distribution with parameter s', therefore the following equation holds: $[\mathbf{A} \mid \mathbf{A}\hat{\mathbf{R}}_{\mathsf{id}^*}]\mathbf{z}'_{\mathsf{id},\mathsf{m}} = \mathbf{h}_2$. It implies that

$$[\mathbf{A} \mid \mathbf{A}\hat{\mathbf{R}}_{\mathsf{id}^*}]\mathbf{z}^* = \mathbf{h}_2 = [\mathbf{A} \mid \mathbf{A}\hat{\mathbf{R}}_{\mathsf{id}^*}]\mathbf{z}'_{\mathsf{id},\mathsf{m}},$$

and then $\mathcal{B}$ can set the solution to SIS problem as

$$\mathbf{z} := [\mathbf{I}_m \mid \hat{\mathbf{R}}_{\mathsf{id}^*}](\mathbf{z}^* - \mathbf{z}'_{\mathsf{id},\mathsf{m}}).$$

It remains to show that $\mathbf{z}$ is a valid solution for $\mathrm{SIS}_{n,m,q,\beta}$ problem, i.e. $\mathbf{z} \neq \mathbf{0}$ and $\|\mathbf{z}\| \leq \beta$.

We prove that $\mathbf{z}$ is non-zero firstly. Note that $\mathbf{z}'_{\mathsf{id},\mathsf{m}}$ is a random vector sampled from gaussian distribution $\mathcal{D}_{\mathbb{Z},s'}^{m+n\lceil \log q \rceil}$, thus $(\mathbf{z}^* - \mathbf{z}'_{\mathsf{id},\mathsf{m}}) \neq \mathbf{0}$ with high probability. Then set $\overline{\mathbf{z}} := (\mathbf{z}^* - \mathbf{z}'_{\mathsf{id},\mathsf{m}})$, write $\overline{\mathbf{z}} = [\overline{\mathbf{z}}_1 \in \mathbb{Z}_q^m \mid \ \overline{\mathbf{z}}_2 \in \mathbb{Z}_q^{n\lceil \log q \rceil}]^t$, and $\mathbf{z}$ can be represented as

$$\mathbf{z} = \overline{\mathbf{z}}_1 + \hat{\mathbf{R}}_{\mathsf{id}^*} \cdot \overline{\mathbf{z}}_2.$$

If $\mathbf{z} = \mathbf{0}$, then it cannot be the case that $\overline{\mathbf{z}}_2 = \mathbf{0}$, because it implies that $\overline{\mathbf{z}}_1 = \mathbf{0}$ which makes $\overline{\mathbf{z}} = \mathbf{0}$. Thus, assume that $\overline{\mathbf{z}}_2$ has a non-zero component $\overline{z}_{2,j}$ where $j \in [n\lceil \log q \rceil]$. Denote each column of $\hat{\mathbf{R}}_{\mathsf{id}^*}$ as $\hat{\mathbf{r}}_i$, where $i \in [n\lceil \log q \rceil]$. Assume that $\mathbf{z} = \mathbf{0}$, it implies that

$$\hat{\mathbf{r}}_j = -\tfrac{1}{\overline{z}_{2,j}}(\overline{\mathbf{z}}_1 + \sum_{i \neq j} \overline{z}_{2,i}\hat{\mathbf{r}}_i).$$

Note that $\hat{\mathbf{R}}_{\mathsf{id}^*}$ is drawn from distribution $\mathcal{D}_{\mathbb{Z},s}^{m \times n\lceil \log q \rceil}$, which has a large min-entropy and it implies that the above equation holds with negligible probability.

At last, we check the norm of $\mathbf{z}$. As in Lemma 5, $s_1(\hat{\mathbf{R}}_{\mathsf{id}^*}) \leq s \cdot O(\sqrt{m} + \sqrt{n\lceil \log q \rceil})$ with overwhelming probability. Then

$$\begin{aligned}
\|\mathbf{z}\| &\leq \|\overline{\mathbf{z}}_1\| + \|\hat{\mathbf{R}}_{\mathsf{id}^*}\| \cdot \|\overline{\mathbf{z}}_2\| \\
&\leq (1 + s \cdot O(\sqrt{m} + \sqrt{n\lceil \log q \rceil}))s' \cdot O(\sqrt{m + n\lceil \log q \rceil}) \\
&= s' \cdot O(\sqrt{m + n\lceil \log q \rceil}) + s \cdot s' \cdot O(\sqrt{m} + \sqrt{n\lceil \log q \rceil})O(\sqrt{m + n\lceil \log q \rceil}) \\
&\leq s' \cdot O(\sqrt{m + n\lceil \log q \rceil}) + s \cdot s' \cdot O(m + n\lceil \log q \rceil) \\
&\leq s \cdot s' \cdot O(m + n\lceil \log q \rceil) \leq \beta,
\end{aligned}$$

which means that $\mathbf{z}$ is a valid solution for $\mathrm{SIS}_{n,m,q,\beta}$ problem. Hence, we conclude that $\mathsf{na\text{-}IBS}_{\mathsf{SIS}}$ scheme is UF-naCMA secure. □

6 Conclusion

In this paper, we provide an IB-ME satisfying stronger security. We first proposed the improved security definitions considering practical application requirements. Then we present a generic construction of IB-ME, which is proven secure under stronger security definitions, from 2-level HIBE and IBS. To improve the efficiency of IB-ME instantiated on lattices, we further modify an existing SIS-based IBS to obtain shorter signatures along with simpler signature algorithm. By combining the improved IBS and any 2-level adaptively-secure lattice-based HIBE with anonymity, we finally obtain the first IB-ME from lattices.

References

ABB10a. S. Agrawal, D. Boneh, and X. Boyen. Efficient lattice (H)IBE in the standard model. In *Advances in Cryptology – EUROCRYPT 2010*, *Lecture Notes in Computer Science* 6110, pages 553–572, French Riviera, May 30 – June 3, 2010. Springer, Heidelberg, Germany.

ABB10b. S. Agrawal, D. Boneh, and X. Boyen. Lattice basis delegation in fixed dimension and shorter-ciphertext hierarchical IBE. In *Advances in Cryptology – CRYPTO 2010*, *Lecture Notes in Computer Science* 6223, pages 98–115, Santa Barbara, CA, USA, August 15–19, 2010. Springer, Heidelberg, Germany.

AFNV19. G. Ateniese, D. Francati, D. Nuñez, and D. Venturi. Match me if you can: Matchmaking encryption and its applications. In *Advances in Cryptology – CRYPTO 2019, Part II*, pages 701–731, Santa Barbara, CA, USA, 2019. Springer, Heidelberg, Germany.

Ajt96. M. Ajtai. Generating hard instances of lattice problems (extended abstract). In *28th Annual ACM Symposium on Theory of Computing*, pages 99–108, Philadephia, PA, USA, May 22–24, 1996. ACM Press.

AW17. S. Agrawal and D. J. Wu. Functional encryption: Deterministic to randomized functions from simple assumptions. In *Advances in Cryptology – EUROCRYPT 2017, Part II*, *Lecture Notes in Computer Science* 10211, pages 30–61, Paris, France, April 30 – May 4, 2017. Springer, Heidelberg, Germany.

BF01. D. Boneh and M. K. Franklin. Identity-based encryption from the Weil pairing. In *Advances in Cryptology – CRYPTO 2001*, Lecture Notes in Computer Science, pages 213–229, Santa Barbara, CA, USA, August 19–23, 2001. Springer, Heidelberg, Germany.

BHJ+15. C. Bader, D. Hofheinz, T. Jager, E. Kiltz, and Y. Li. Tightly-secure authenticated key exchange. In *TCC 2015: 12th Theory of Cryptography Conference, Part I*, Lecture Notes in Computer Science, pages 629–658, Warsaw, Poland, March 23–25, 2015. Springer, Heidelberg, Germany.

BL16. X. Boyen and Q. Li. Towards tightly secure lattice short signature and id-based encryption. In *Advances in Cryptology – ASIACRYPT 2016, Part II*, Lecture Notes in Computer Science, pages 404–434, Hanoi, Vietnam, December 4–8, 2016. Springer, Heidelberg, Germany.

BLMQ05. P. S. L. M. Barreto, B. Libert, N. McCullagh, and J.-J. Quisquater. Efficient and provably-secure identity-based signatures and signcryption from bilinear maps. In *Advances in Cryptology – ASIACRYPT 2005*, *Lecture Notes in Computer Science* 3788, pages 515–532, Chennai, India, December 4–8, 2005. Springer, Heidelberg, Germany.

BNN04. M. Bellare, C. Namprempre, and G. Neven. Security proofs for identity-based identification and signature schemes. In *Advances in Cryptology – EUROCRYPT 2004*, Lecture Notes in Computer Science, pages 268–286, Interlaken, Switzerland, May 2–6, 2004. Springer, Heidelberg, Germany.

Boy03. X. Boyen. Multipurpose identity-based signcryption (a swiss army knife for identity-based cryptography). In *Advances in Cryptology – CRYPTO 2003*, *Lecture Notes in Computer Science* 2729, pages 383–399, Santa Barbara, CA, USA, August 17–21, 2003. Springer, Heidelberg, Germany.

CHKP12. D. Cash, D. Hofheinz, E. Kiltz, and C. Peikert. Bonsai trees, or how to delegate a lattice basis. *Journal of Cryptology*, 25(4):601–639, October 2012.

CLL+13. J. Chen, H. W. Lim, S. Ling, H. Wang, and H. Wee. Shorter IBE and signatures via asymmetric pairings. In *PAIRING 2012: 5th International Conference on Pairing-based Cryptography*, *Lecture Notes in Computer Science* 7708, pages 122–140, Cologne, Germany, May 16–18, 2013. Springer, Heidelberg, Germany.

CLWW22. J. Chen, Y. Li, J. Wen, and J. Weng. Identity-based matchmaking encryption from standard assumptions. Cryptology ePrint Archive, Report 2022/1246, 2022. To be appeared in ASIACRYPT 2022. `https://eprint.iacr.org/2022/1246`.

DGJL21. D. Diemert, K. Gellert, T. Jager, and L. Lyu. More efficient digital signatures with tight multi-user security. In *PKC 2021: 24th International Conference on Theory and Practice of Public Key Cryptography, Part II*, Lecture Notes in Computer Science, pages 1–31, Virtual Event, May 10–13, 2021. Springer, Heidelberg, Germany.

DKXY03. Y. Dodis, J. Katz, S. Xu, and M. Yung. Strong key-insulated signature schemes. In *PKC 2003: 6th International Workshop on Theory and Practice in Public Key Cryptography*, Lecture Notes in Computer Science, pages 130–144, Miami, FL, USA, January 6–8, 2003. Springer, Heidelberg, Germany.

FFMV22. D. Francati, D. Friolo, G. Malavolta, and D. Venturi. Multi-key and multi-input predicate encryption from learning with errors. Cryptology ePrint Archive, Report 2022/806, 2022. `https://eprint.iacr.org/2022/806`.

FGRV21. D. Francati, A. Guidi, L. Russo, and D. Venturi. Identity-based matchmaking encryption without random oracles. In *Progress in Cryptology – INDOCRYPT 2021*, pages 415–435, Cham, 2021. Springer International Publishing.

Gen06. C. Gentry. Practical identity-based encryption without random oracles. In *Advances in Cryptology – EUROCRYPT 2006*, Lecture Notes in Computer Science, pages 445–464, St. Petersburg, Russia, May 28 – June 1, 2006. Springer, Heidelberg, Germany.

GGG+14. S. Goldwasser, S. D. Gordon, V. Goyal, A. Jain, J. Katz, F.-H. Liu, A. Sahai, E. Shi, and H.-S. Zhou. Multi-input functional encryption. In *Advances in Cryptology – EUROCRYPT 2014*, Lecture Notes in Computer Science, pages 578–602, Copenhagen, Denmark, May 11–15, 2014. Springer, Heidelberg, Germany.

GJ18. K. Gjøsteen and T. Jager. Practical and tightly-secure digital signatures and authenticated key exchange. In *Advances in Cryptology – CRYPTO 2018, Part II*, Lecture Notes in Computer Science, pages 95–125, Santa Barbara, CA, USA, August 19–23, 2018. Springer, Heidelberg, Germany.

GJKS15. V. Goyal, A. Jain, V. Koppula, and A. Sahai. Functional encryption for randomized functionalities. In *TCC 2015: 12th Theory of Cryptography Conference, Part II*, pages 325–351, Warsaw, Poland, 2015. Springer, Heidelberg, Germany.

GJKW07. E.-J. Goh, S. Jarecki, J. Katz, and N. Wang. Efficient signature schemes with tight reductions to the Diffie-Hellman problems. *Journal of Cryptology*, 20(4):493–514, October 2007.

GPV08. C. Gentry, C. Peikert, and V. Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In *40th Annual ACM Symposium on Theory of Computing*, pages 197–206, Victoria, BC, Canada, May 17–20, 2008. ACM Press.

GS02. C. Gentry and A. Silverberg. Hierarchical ID-based cryptography. In *Advances in Cryptology – ASIACRYPT 2002*, Lecture Notes in Computer Science, pages 548–566, Queenstown, New Zealand, December 1–5, 2002. Springer, Heidelberg, Germany.

KMT19. S. Katsumata, T. Matsuda, and A. Takayasu. Lattice-based revocable (hierarchical) IBE with decryption key exposure resistance. In *PKC 2019: 22nd International Conference on Theory and Practice of Public Key Cryptography, Part II*, Lecture Notes in Computer Science, pages 441–471, Beijing, China, April 14–17, 2019. Springer, Heidelberg, Germany.

LLW20. Q. Lai, F.-H. Liu, and Z. Wang. Almost tight security in lattices with polynomial moduli - PRF, IBE, all-but-many LTF, and more. In *PKC 2020: 23rd International Conference on Theory and Practice of Public Key Cryptography, Part I*, Lecture Notes in Computer Science, pages 652–681, Edinburgh, UK, May 4–7, 2020. Springer, Heidelberg, Germany.

LP19. R. Langrehr and J. Pan. Tightly secure hierarchical identity-based encryption. In *PKC 2019: 22nd International Conference on Theory and Practice of Public Key Cryptography, Part I*, *Lecture Notes in Computer Science* 11442, pages 436–465, Beijing, China, April 14–17, 2019. Springer, Heidelberg, Germany.

LP20a. R. Langrehr and J. Pan. Hierarchical identity-based encryption with tight multi-challenge security. In *PKC 2020: 23rd International Conference on Theory and Practice of Public Key Cryptography, Part I*, *Lecture Notes in Computer Science* 12110, pages 153–183, Edinburgh, UK, May 4–7, 2020. Springer, Heidelberg, Germany.

LP20b. R. Langrehr and J. Pan. Unbounded HIBE with tight security. In *Advances in Cryptology – ASIACRYPT 2020, Part II*, *Lecture Notes in Computer Science* 12492, pages 129–159, Daejeon, South Korea, December 7–11, 2020. Springer, Heidelberg, Germany.

LPLL20. Y. Lee, J. H. Park, K. Lee, and D. H. Lee. Tight security for the generic construction of identity-based signature (in the multi-instance setting). *Theoretical Computer Science*, 847:122–133, 2020.

Mal02. J. Malone-Lee. Identity-based signcryption. Cryptology ePrint Archive, Report 2002/098, 2002. `https://eprint.iacr.org/2002/098`.

MP12. D. Micciancio and C. Peikert. Trapdoors for lattices: Simpler, tighter, faster, smaller. In *Advances in Cryptology – EUROCRYPT 2012*, Lecture Notes in Computer Science, pages 700–718, Cambridge, UK, April 15–19, 2012. Springer, Heidelberg, Germany.

MR04. D. Micciancio and O. Regev. Worst-case to average-case reductions based on Gaussian measures. In *45th Annual Symposium on Foundations of Computer Science*, pages 372–381, Rome, Italy, October 17–19, 2004. IEEE Computer Society Press.

OT09. T. Okamoto and K. Takashima. Hierarchical predicate encryption for inner-products. In *Advances in Cryptology – ASIACRYPT 2009*, Lecture Notes in Computer Science, pages 214–231, Tokyo, Japan, December 6–10, 2009. Springer, Heidelberg, Germany.

PPS21. C. Peikert, Z. Pepin, and C. Sharp. Vector and functional commitments from lattices. In *TCC 2021: 19th Theory of Cryptography Conference, Part III*, Lecture Notes in Computer Science, pages 480–511, Raleigh, NC, USA, November 8–11, 2021. Springer, Heidelberg, Germany.

PW21. J. Pan and B. Wagner. Short identity-based signatures with tight security from lattices. In *Post-Quantum Cryptography - 12th International Workshop, PQCrypto 2021*, pages 360–379, Daejeon, South Korea, July 20–22 2021. Springer, Heidelberg, Germany.

Sha84. A. Shamir. Identity-based cryptosystems and signature schemes. In *Advances in Cryptology – CRYPTO'84*, Lecture Notes in Computer Science, pages 47–53, Santa Barbara, CA, USA, August 19–23, 1984. Springer, Heidelberg, Germany.

Zhe97. Y. Zheng. Digital signcryption or how to achieve cost(signature & encryption) $\ll$ cost(signature) + cost(encryption). In *Advances in Cryptology – CRYPTO'97*, *Lecture Notes in Computer Science* 1294, pages 165–179, Santa Barbara, CA, USA, August 17–21, 1997. Springer, Heidelberg, Germany.