Identity-Based Matchmaking Encryption, Revisited

Strong Security and Practical Constructions from Standard Classical and Post-Quantum Assumptions

Sohto Chiku[1], Keitaro Hashimoto[2], Keisuke Hara[1,2], and Junji Shikata[1]

[1] Yokohama National University, Kanagawa, Japan
chiku-sohto-tw@ynu.jp
{hara-keisuke-kj,shikata-junji-rb}@ynu.ac.jp
[2] National Institute of Advanced Industrial Science and Technology (AIST), Tokyo, Japan
keitaro.hashimoto@aist.go.jp

September 21, 2023

Abstract. Identity-based matchmaking encryption (IB-ME), proposed by Ateniese et al. at Crypto 2019, allows users to communicate privately in an anonymous and authenticated manner. In this work, we revisit the security definitions and construction of IB-ME. First, we re-formalize the existing security notions for IB-ME. We reorganize privacy and authenticity notions into respective three and four definitions, which allows us to compare IB-ME schemes accurately. Second, we propose a highly efficient and strongly secure IB-ME scheme from the bilinear Diffie-Hellman assumption in the random oracle model. This scheme is based on the IB-ME scheme proposed by Ateniese et al., but we introduce several techniques to improve its security and efficiency. Third, we propose a new generic construction of IB-ME from anonymous identity-based encryption and identity-based signature. This is the first generic construction that does not rely on hierarchical identity-based encryption. Through this construction, we obtain various IB-ME schemes from both classical and post-quantum assumptions. For example, we obtain a more efficient scheme from the symmetric external Diffie-Hellman assumption in the standard model, and a practical scheme from lattices in the quantum random oracle model whose secret keys and ciphertexts are less than 10 Kilobytes. Moreover, our generic construction produces the first pairing-free IB-ME scheme in the standard model and the first tightly secure lattice-based IB-ME scheme in the quantum random oracle model.

Keywords: Identity-Based Matchmaking Encryption · Security Model · Pairing-Based Cryptography · Generic Construction · Post-Quantum.

Table of Contents

1 Introduction

1.1 Background

Identity-based matchmaking encryption (IB-ME), proposed by Ateniese et al. [2], is a new identity-based cryptographic primitive designed to ensure confidential and authenticated message delivery. Similar to conventional identity-based encryption (IBE) [5], a key generation center generates secret keys of users corresponding to their identity, and in the IB-ME setting, both a sender and receiver possess their secret keys. When a sender with an identity σ sends a message, it encrypts the message with its (secret) encryption key ek_σ and a target receiver's identity rcv. Then, a receiver with an identity ρ decrypts the ciphertext with its secret decryption key dk_ρ and a target sender's identity snd. The decryption process is successful only if the identities match, meaning that $\sigma = \mathsf{snd}$ and $\mathsf{rcv} = \rho$ hold. In case the identities do not match, nothing is leaked except for the fact that the identities are mismatched. IB-ME has many practical applications such as secret handshake protocols [3], privacy-preserving bulletin boards [2], etc.

Security notions for IB-ME. Ateniese et al. [2] defined the security requirements for IB-ME which they call *privacy* and *authenticity*. In essence, privacy guarantees the confidentiality of messages against unintended receivers who do not match the sender's policy. Also, authenticity guarantees the legitimacy of senders, preventing impersonation without knowledge of their encryption key. We can see that privacy (resp., authenticity) is similar to the semantic security of encryption schemes (resp., unforgeability of signature schemes.[3]) Although the definitions capture basic threats, they do not satisfy some desired properties. For example, their definition of authenticity does not take into account the case where an adversary might compromise the secret key of a target receiver (not the secret key of a target sender[4]).

Following the pioneering work by Ateniese et al., a lot of works have explored more desirable security notions. Regarding authenticity, Francati et al. [18] and Chen et al. [11] defined a new authenticity that allows an adversary to compromise the receivers' secret keys freely in contrast to Ateniese et al.'s definition[5]. Wang et al. [35] proposed an extended version of authenticity notions, which they call "strong authenticity", allowing the adversary to access the encryption oracle that computes a ciphertext of adversarially chosen messages[6]. Also, for stronger privacy guarantees, Chiku et al. [13] considered privacy against chosen-ciphertext attacks (CCA). Their privacy game allows an adversary to access the decryption oracle that computes plaintexts of adversarially chosen ciphertexts. Francati et al. [18] highlighted a deficiency in the original privacy security definition by Ateniese et al. They pointed out that it does not account for privacy in the case where the target identity snd chosen by a receiver mismatches with the actual sender's identity σ. That is, the original definition does not guarantee the confidentiality of messages when the case $\mathsf{rcv} = \rho$ but $\mathsf{snd} \neq \sigma$ occurs during decryption[7]. This gap led them to introduce a new privacy notion called "enhanced privacy", which captures privacy in cases involving mismatched sender identities used during decryption.

As explained, many security definitions for IB-ME have been considered, but they are not well-organized. In particular, existing works compared the efficiency of each scheme ignoring the differences in the security properties, which makes the evaluation inaccurate. From such a situation, we realize the first question:

Q1: What are proper security definitions of IB-ME for accurate comparison?

[3] As explained later, we call them privacy against chosen plaintext attacks (CPA) and authenticity against no message attacks (NMA), respectively.

[4] Note that the authenticity does not hold inherently if a sender's secret key is compromised since an adversary can forge any ciphertext associated with the sender.

[5] The difference was not explained explicitly in [11, 18], which confuses the comparison. Especially, Francati et al. cited the original paper despite this difference.

[6] The attack scenario can be seen as ordinary chosen message attacks (CMA), but they did not explain it as such.

[7] As mentioned in [18], Ateniese et al. noticed this gap, and informally argued that their IB-ME scheme ensures the confidentiality of messages in such a case.

Table 1: Comparison between our schemes and the existing IB-ME schemes.

Schemes	Security properties			Assumptions	Model
	Privacy	Authenticity	Mismatch		
Ateniese et al. [2]	CPA	oNMA		BDH	ROM
Francati et al. [18]	CPA	iNMA	√	q-ABDHE+NIZK +Reusable extractors	StdM
Chen et al. [11]	CPA	iNMA		SXDH	StdM
Wang et al. [35]	CPA	iCMA		Anon HIBE+IBS	StdM
Ours (§ 4)	CCA	oCMA	√	BDH	ROM
Ours (§ 5)	CCA	iCMA	√	Anon IBE+IBS +Reusable extractors	StdM

Constructions of IB-ME. Ateniese et al. introduced the initial IB-ME scheme based on the bilinear Diffie-Hellman (BDH) assumption in the random oracle model (ROM) [2]. Their scheme seems a combination of the Boneh-Franklin IBE scheme [5] and the Sakai-Ohgishi-Kasahara identity-based non-interactive key exchange (IB-NIKE) scheme [33]. However, this combination does not appear to be intuitive. A straightforward fusion of IBE and IB-NIKE would typically result in a receiver possessing two group elements, but in their scheme, it involves three elements. Furthermore, ciphertexts include two random group elements, but one of them does not appear to contribute to security. Additionally, their scheme does not achieve the stronger security proposed after their work. This raises the second question about whether it is possible to construct a more efficient and strongly secure scheme from the BDH assumption:

Q2: Can we construct a more efficient and strongly secure IB-ME scheme from the BDH assumption in the ROM?

Following the initial work by Ateniese et al., several works have made efforts to develop improved IB-ME schemes, with a particular focus on the standard model (StdM) [11, 18, 35]. Francati et al. [18] proposed an IB-ME scheme in the StdM by building upon Gentry's IBE scheme [21]. While their scheme is secure in the StdM, it relies on a non-standard q-augumented bilinear Diffie-Hellman exponent (q-ABDHE) assumption. To remove the reliance on non-standard assumptions, Chen et al. [11] constructed an IB-ME scheme based on anonymous IBE scheme by Chen et al. [12], whose security relies on the symmetric external Diffie-Hellman (SXDH) assumption in the StdM. Recently, Wang et al. [35] proposed a generic construction of IB-ME from anonymous 2-level hierarchical IBE (HIBE) and identity-based signature (IBS) to realize IB-ME schemes from lattices. Moreover, Chiku et al. [13] proposed a new variant of IB-ME, called "hierarchical" IB-ME, and its generic construction based on anonymous HIBE and hierarchical IBS (HIBS).

We notice that the existing schemes relying on specific computational assumptions [2, 11, 18] are based on IBE, but the existing generic constructions [13, 35] are based on HIBE. Moreover, these generic constructions do not satisfy "enhanced privacy" which captures a vital security property of IB-ME. This fact gives us the third question:

Q3: Can we generic construct a strongly secure IB-ME from IBE, not HIBE?

1.2 Our Contributions

We revisit the concept of IB-ME and answer the above three research questions. We first re-formalize security notions for IB-ME, and then, present an efficient and strongly secure IB-ME scheme from the BDH assumption in the ROM and a new generic construction from IBE, IBS, and reusable extractors in the StdM. The comparison of our schemes and the existing ones is summarized in Table 1. See Section 6 for the detailed comparison, especially the efficiency of them.

A1: Re-formalizing security notions. We sort out the differences in security notions for IB-ME. At first, we reorganize the authenticity notions in the previous works. We notice that the existing definitions can be classified along two axes: one is whether an adversary has access to the encryption oracle and the other is whether it can compromise the target receiver's secret key. For the former axes, we name the respective attacks as chosen message attacks (CMA) and no message attacks (NMA) according to the presence or absence of access to the encryption oracle. For the latter axes, we call the adversary who compromises the target receiver *insiders* and otherwise *outsiders* since we can regard the adversary, who knows the receiver's key, is inside the communication.[8] As a result, we define four authenticity notions oNMA, iNMA, oCMA, and iCMA (Table 1 shows the correspondence of them and the previous works).

For privacy, we rename the original definition by Ateniese et al. as CPA security since the adversary cannot access the decryption oracle, and define CCA security [13]. Then, we redefine the security game for "enhanced privacy" which captures privacy in mismatch cases during decryption. Francati et al. [18] defined a single definition that includes both privacy originally considered and privacy in mismatch cases, which complicates the understanding of the definition and security proofs. Thus, we extract the essence from it and give a new simple security definition, called Priv-MisMatch security, which only considers privacy in mismatch cases. Roughly, it captures the confidentiality of messages in the case the adversary knows the target receiver's secret key but does not know the sender's identity. As a result, we can separate security proofs for standard privacy and privacy in mismatch cases. See Section 3 for the details.

A2: An efficient and strongly secure IB-ME scheme from BDH in the ROM. We construct a new IB-ME scheme from the BDH assumption in the ROM. Similar to the work by Ateniese et al., our basic idea is combining the Boneh-Franklin IBE scheme [5] and the Sakai-Ohgishi-Kasahara IB-NIKE scheme [33]. At a high-level, a sender with an identity σ holds an IB-NIKE key $\mathsf{H}(\sigma)^{\mathsf{msk}}$ as its encryption key and a receiver with an identity ρ holds an IB-NIKE key $\mathsf{H}(\rho)^{\mathsf{msk}}$ and IBE key $\mathsf{H}(\rho)^{\mathsf{msk}'}$ as its decryption key, where H is an (appropriate) hash function and msk (resp., msk') is a master secret key of the IB-NIKE scheme (resp., the IBE scheme). When the sender σ encrypts a message m for targeting a receiver rcv, it computes a ciphertext as $(g^r, \mathsf{m} \oplus \hat{\mathsf{H}}(e(X^r, \mathsf{H}(\mathsf{rcv})), e(\mathsf{H}(\sigma)^{\mathsf{msk}}, \mathsf{H}(\mathsf{rcv}))))$, where g is a generator (of the underlying group), $X = g^{\mathsf{msk}'}$ is a public parameter of the IBE scheme, and e is a symmetric pairing. To reduce the key size, we reuse the same master secret key for the IBE part and the IB-NIKE part. That is, we use the key $\mathsf{H}(\mathsf{id})^{\mathsf{msk}}$ for both the IBE scheme and the IB-NIKE scheme, where id is an identity for either sender or receiver. This reduces the size of a user's secret key but weakens the security level since the compromise of a user leaks both encryption and decryption keys. To overcome this problem, we separate the domains of senders' and receivers' keys by employing asymmetric pairings. By using different hash functions H_1 and H_2, we compute the key of a sender σ as $\mathsf{H}_1(\sigma)^{\mathsf{msk}} \in \mathbb{G}_1$ and the key of a receiver ρ as $\mathsf{H}_2(\rho)^{\mathsf{msk}} \in \mathbb{G}_2$. This allows us to reduce the key size without weakening the security. Intuitively, privacy is followed by the security of the IBE scheme, and authenticity is followed by the security of the IB-NIKE scheme[9]. To achieve the stronger CCA security, we employ the Fujisaki-Okamoto (FO) transformation [19, 20]. Somewhat surprisingly, the FO transformation allows us to achieve the oCMA security for free. Moreover, we formally prove that our scheme also achieves Priv-MisMatch security. As a result, we get a more efficient and strongly secure IB-ME scheme from the BDH assumption in the ROM. A user's key contains only one group element and the ciphertext contains one group element and a λ-bits string, both of which are smaller than that of Ateniese et al.'s scheme. See Section 4 for the details.

A3: Efficient and strongly secure IB-ME schemes in the StdM. We propose a new generic construction of IB-ME from anonymous IBE, IBS, and reusable extractors. In our construction, a sender σ holds an IBS's user key ek_σ and a receiver ρ holds an IBE's user key. The sender σ encrypts a message m to a receiver rcv as $\mathsf{ct} \leftarrow \mathsf{IBE.Enc}(\mathsf{mpk}_{\mathsf{IBE}}, \mathsf{rcv}, \mathsf{m}||\mathsf{sig})$, where $\mathsf{mpk}_{\mathsf{IBE}}$ (resp., $\mathsf{mpk}_{\mathsf{IBS}}$) is a public parameter of the IBE (resp., IBS) scheme and $\mathsf{sig} \leftarrow \mathsf{IBS.Sign}(\mathsf{mpk}_{\mathsf{IBS}}, \mathsf{ek}_\sigma, \mathsf{m})$. We can show that this simple construction achieves the

[8] Here, we employ the naming in a similar situation in signcryption [28]

[9] Due to the symmetry of keys in the IB-NIKE part, the authenticity only holds when both sender and receiver are not compromised, i.e., only holds against outsiders. This is also the case in the work by Ateniese et al.

CCA security and the iCMA security from the CCA security of the IBE scheme and the CMA security of the IBS scheme, respectively. However, it is not Priv-MisMatch secure. The main reason is that an adversary who knows the receivers' keys can decrypt IBE ciphertexts and thus get the encrypted messages without knowing the designated sender's identity. To hide messages in mismatch cases, we use reusable extractors similar to the work by Francati et al. [18]. Roughly, $\mathsf{m}||\mathsf{sig}$ is masked by the extractor's output $Z := \mathsf{Ext}(s, \sigma)$ before encryption. That is, $(\mathsf{m}||\mathsf{sig}) \oplus Z$ is encrypted by IBE. This mask prevents an adversary from recovering messages without knowing the sender's identity, which leads to the Priv-MisMatch security. It is worth noting that this result makes it clear that HIBE is not necessary for constructing IB-ME schemes. Through our generic construction, we can obtain a lot of IB-ME schemes from both classical and post-quantum assumptions in both (quantum) ROM ((Q)ROM) and StdM.[10] For example, we obtain a more efficient and strongly secure IB-ME scheme from the SXDH assumption in the StdM, and a practical post-quantum IB-ME scheme from lattices in the QROM. The latter scheme offers a small user's key and ciphertext of less than 10 Kilobytes. Moreover, as feasibility results, we get the first pairing-free IB-ME scheme in the StdM from a pairing-free anonymous IBE scheme [8][11] and an IBS scheme [25], and the first tightly secure IB-ME scheme from lattices in the QROM from lattice-based tightly secure anonymous IBE scheme [24] and IBS scheme [16]. See Section 5 for the details.

1.3 Related Work

Identity-based encryption. Identity-based encryption, proposed by Shamir [34], is an encryption scheme that allows users to use arbitrary strings (e.g., e-mail addresses) as their public keys. After quite a long time, Boneh and Franklin constructed the first IBE scheme [5] using bilinear pairings, and then a lot of IBE schemes have been proposed from various assumptions [1, 15, 21, 22, 24, 36, 37]. In IBE, the sender specifies only the receiver's identity, but in IB-ME, the sender specifies not only the receiver's identity but also the sender's identity.

Identity-based signcryption. Signcryption [39] is a cryptographic primitive that offers both private and authenticated delivery of messages. The motivation for signcryption is to provide equivalent functionality more efficiently than a simple combination of encryption and signature schemes. The notion of identity-based signcryption (IB-SC) was proposed by Malone-Lee [27]. The difference between IB-ME and IB-SC is that the former ensures the anonymity of communicating users and the confidentiality of messages when ciphertexts are decrypted with mismatched sender identities. Therefore, IB-ME provides better security properties than IB-SC.

(General) Matchmaking encryption. Ateniese et al. proposed matchmaking encryption [2]. In ME setting, the sender and the receiver have their own attribute, and they can specify access policies the other party must satisfy. Ateniese et al. also gave generic constructions of ME based on functional encryption, signature scheme, and non-interactive zero-knowledge. Recently, Francati et al. [17] proposed a simple ME scheme based on two-key predicate encryption. Note that IB-ME is an ME supporting the policy of identity equivalence.

1.4 Organization of This Paper

The remaining part of this paper is organized as follows. In Section 2, we introduce notations and definitions of the cryptographic primitives that will be used in this paper. Then, in Section 3, we give the relevant definitions including syntax and security definitions of IB-ME. Section 4 shows an efficient and strongly secure IB-ME scheme based on BDH assumption in the ROM. In Section 5, we provide a new generic construction of IB-ME based on IBE, IBS, and reusable extractor in the StdM. Finally, Section 6 presents a comparison between our IB-ME schemes and the existing schemes.

[10] Reusable extractors can be instantiated from RO [6] or a variant of the decisional Diffie-Hellman (DDH) assumption or of the learning party with noise (LPN) assumption in the StdM [9, 14].

[11] We can convert [8] to CCA secure one by using Naor-Yung transformation [29] with a pairing-free NIZK from the sub-exponential DDH assumption [23]. Note that the Naor-Yung transformation preserves the anonymity of the underlying IBE scheme.

2 Preliminaries

In this section, we first define some notations used in this work. Then we recall asymmetric bilinear groups, identity-based encryption, identity-based signature, and reusable computational extractors.

2.1 Notation

$\mathbb{N}$ denotes the set of positive integers. $\emptyset$ denotes the empty set. $\hat{e}$ denotes the base of the natural logarithm. PPT stands for probabilistic polynomial time. For $n \in \mathbb{N}$, we denote $[n] \coloneqq \{1, 2, \ldots, n\}$. $x \coloneqq y$ denotes that x is defined by y. $y \leftarrow \mathcal{A}(x; r)$ denotes that a PPT algorithm $\mathcal{A}$ outputs y on input x and a randomness r. We simply denote $y \leftarrow \mathcal{A}(x)$ when $\mathcal{A}$ uses uniform randomness. $\mathcal{A}^{\mathcal{O}}$ means $\mathcal{A}$ has oracle access to a function $\mathcal{O}(\cdot)$. We say a function $f(\lambda)$ is negligible in λ if $f(\lambda) = o(1/\lambda^c)$ for every $c \in \mathbb{Z}$, and we write $\mathsf{negl}(\lambda)$ to denote a negligible function in λ. $x \leftarrow\!\!\$\ \mathcal{X}$ denotes an element x is sampled uniformly at random from a finite set $\mathcal{X}$. Let X be a distribution over $\mathcal{X}$. The min-entropy of X is define as $\mathrm{H}_\infty(X) \coloneqq -\log \max_{x \in \mathcal{X}} \Pr[X = x]$. We call a distribution with min-entropy k k-distribution. $x \leftarrow\!\!\$\ X$ denotes an element $x \in \mathcal{X}$ is sampled following the distribution X.

2.2 Asymmetric Bilinear Groups

We recall (asymmetric) bilinear groups[12] and the bilinear Diffie-Hellman (BDH) assumption from [4,7]. Let $\mathbb{G}_1$, $\mathbb{G}_2$ and $\mathbb{G}_T$ be groups of prime order p. Let $g_1 \in \mathbb{G}_1$ and $g_2 \in \mathbb{G}_2$ be respective generators of $\mathbb{G}_1$ and $\mathbb{G}_2$. Let $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ be an efficiently computable function that satisfies (1) for any $u \in \mathbb{G}_1$, $v \in \mathbb{G}_2$ and $\alpha, \beta \in \mathbb{Z}_p$, $e(u^\alpha, v^\beta) = e(u, v)^{\alpha\beta}$ (i.e., bilinearity) and (2) $e(g_1, g_2) \neq 1$, where 1 is the unit element in $\mathbb{G}_T$ (i.e., non-degeneracy). Such function e is called a *bilinear map* or *pairing*. We call $G \coloneqq (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, g_1, g_2, e)$ bilinear group. We define bilinear group generators that generate a bilinear group corresponding to the input security parameter.

Definition 1 (Bilinear Group Generator). *A* bilinear group generator $\mathcal{G}$ *is a* PPT *algorithm that, on input* 1^λ*, outputs the description of a bilinear group* $G = (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, g_1, g_2, e)$.

We define the BDH assumption for $\mathcal{G}$.

Definition 2 (Bilinear Diffie-Hellman (BDH) Assumption [4, 7]). *Let* $\mathcal{G}$ *be a bilinear group generator. We say that* BDH assumption holds for $\mathcal{G}$ *if for all* PPT *adversaries* $\mathcal{A}$*, it holds that*

$$\mathsf{Adv}^{\mathsf{bdh}}_{\mathcal{A},\mathcal{G}}(\lambda) \coloneqq \Pr\left[D = e(g_1, g_2)^{\alpha\beta\gamma} \;\middle|\; \begin{array}{c} G \coloneqq (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, g_1, g_2, e) \leftarrow \mathcal{G}(1^\lambda), \\ \alpha, \beta, \gamma \leftarrow\!\!\$\ \mathbb{Z}_p, \\ D \leftarrow \mathcal{A}(G, g_1^\alpha, g_2^\alpha, g_2^\beta, g_1^\gamma) \end{array} \right] = \mathsf{negl}(\lambda).$$

2.3 Identity-Based Encryption

Syntax. An IBE scheme IBE consists of the following four algorithms.

$\mathsf{Setup}(1^\lambda) \to (\mathsf{mpk}, \mathsf{msk})$**:** The setup algorithm takes the security parameter 1^λ, and outputs a public parameter mpk and a master secret key msk. mpk defines the identity space $\mathcal{ID}$, message space $\mathcal{M}$ and ciphertext space $\mathcal{CT}$.

$\mathsf{KGen}(\mathsf{mpk}, \mathsf{msk}, \mathsf{id}) \to \mathsf{sk}_{\mathsf{id}}$**:** The key generation algorithm takes mpk, msk, and am identity $\mathsf{id} \in \mathcal{ID}$ as input, and outputs a secret key $\mathsf{sk}_{\mathsf{id}}$.

$\mathsf{Enc}(\mathsf{mpk}, \mathsf{id}, \mathsf{m}) \to \mathsf{ct}$**:** The encryption algorithm takes mpk, $\mathsf{id} \in \mathcal{ID}$, and a plaintext $\mathsf{m} \in \mathcal{M}$ as input, and outputs a ciphertext $\mathsf{ct} \in \mathcal{CT}$.

$\mathsf{Dec}(\mathsf{mpk}, \mathsf{sk}_{\mathsf{id}}, \mathsf{ct}) \to \mathsf{m}$ **or** $\perp$**:** The decryption algorithm takes mpk, $\mathsf{sk}_{\mathsf{id}}$, and ct as input, and outputs $\mathsf{m} \in \mathcal{M}$ or a special symbol $\perp \notin \mathcal{M}$.

[12] In this work, we only work on asymmetric bilinear groups. So, we omit the term "asymmetric".

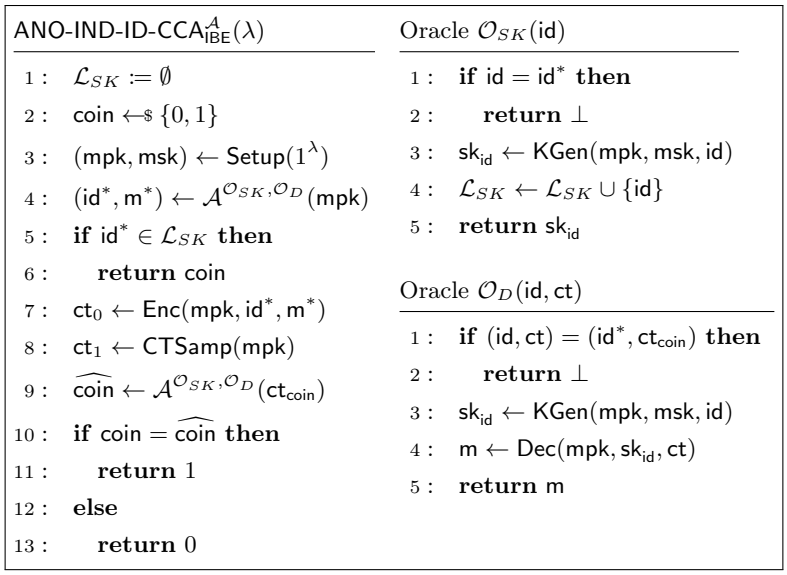

Fig. 1: The security game for IBE.

Correctness. We say that an IBE scheme IBE is *correct* if for all $\lambda \in \mathbb{N}$, $\mathsf{id} \in \mathcal{ID}$ and $\mathsf{m} \in \mathcal{M}$, it holds that

$$\Pr\left[\mathsf{Dec}(\mathsf{mpk}, \mathsf{sk}_{\mathsf{id}}, \mathsf{ct}) = \mathsf{m} \;\middle|\; \begin{array}{l}(\mathsf{mpk}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda),\\ \mathsf{sk}_{\mathsf{id}} \leftarrow \mathsf{KGen}(\mathsf{mpk}, \mathsf{msk}, \mathsf{id}),\\ \mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{mpk}, \mathsf{id}, \mathsf{m})\end{array}\right] = 1 - \mathsf{negl}(\lambda).$$

Security. We recall adaptive-identity anonymity against chosen-ciphertext attacks (ANO-IND-ID-CCA security) for IBE (used in e.g., [24]). Let $\mathsf{CTSamp}(\cdot)$ be a PPT algorithm that takes as input a master public key and outputs an element in the ciphertext space.

Definition 3 (ANO-IND-ID-CCA Security of IBE). *We say that an IBE scheme* IBE *is* ANO-IND-ID-CCA secure *if for all* PPT *adversaries* $\mathcal{A}$*,*

$$\mathsf{Adv}^{\mathsf{ano\text{-}ind\text{-}id\text{-}cca}}_{\mathcal{A},\mathsf{IBE}}(\lambda) := \left|\Pr\left[\mathsf{ANO\text{-}IND\text{-}ID\text{-}CCA}^{\mathcal{A}}_{\mathsf{IBE}}(\lambda) \Rightarrow 1\right] - \frac{1}{2}\right| = \mathsf{negl}(\lambda),$$

where the security game $\mathsf{ANO\text{-}IND\text{-}ID\text{-}CCA}^{\mathcal{A}}_{\mathsf{IBE}}(\lambda)$ *is depicted in Fig. 1.*

2.4 Identity-Based Signature

Syntax. An IBS scheme IBS consists of the following four algorithms.

$\mathsf{Setup}(1^\lambda) \to (\mathsf{mpk}, \mathsf{msk})$**:** The setup algorithm takes the security parameter 1^λ, and outputs a public parameter mpk and master secret key msk. mpk defines the identity space $\mathcal{ID}$, message space $\mathcal{M}$ and signature's bit-length siglen.

$\mathsf{KGen}(\mathsf{mpk}, \mathsf{msk}, \mathsf{id}) \to \mathsf{sk}_{\mathsf{id}}$**:** The key generation algorithm takes mpk, msk, and an identity $\mathsf{id} \in \mathcal{ID}$ as input, and outputs a signing key $\mathsf{sk}_{\mathsf{id}}$.

$\mathsf{Sign}(\mathsf{mpk}, \mathsf{sk}_{\mathsf{id}}, \mathsf{m}) \to \mathsf{sig}$**:** The signing algorithm takes mpk, $\mathsf{sk}_{\mathsf{id}}$, and a message $\mathsf{m} \in \mathcal{M}$ as input, and outputs a signature sig.

$\mathsf{Ver}(\mathsf{mpk}, \mathsf{id}, \mathsf{m}, \mathsf{sig}) \to 0$ **or** 1**:** The verification algorithm takes mpk, $\mathsf{id} \in \mathcal{ID}$, m and sig as input, and outputs a bit $b \in \{0,1\}$.

$\mathsf{EUF\text{-}ID\text{-}CMA}^{\mathcal{A}}_{\mathsf{IBS}}(\lambda)$

1 : $\mathcal{L}_{SK}, \mathcal{L}_{SIG} := \emptyset$
2 : $(\mathsf{mpk}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda)$
3 : $(\mathsf{id}^*, \mathsf{m}^*, \mathsf{sig}^*) \leftarrow \mathcal{A}^{\mathcal{O}_{SK}, \mathcal{O}_{SIG}}(\mathsf{mpk})$
4 : **if** $\mathsf{id}^* \in \mathcal{L}_{SK} \vee (\mathsf{id}^*, \mathsf{m}^*) \in \mathcal{L}_{SIG}$ **then**
5 : **return** 0
6 : **if** $\mathsf{Ver}(\mathsf{mpk}, \mathsf{id}^*, \mathsf{m}^*, \mathsf{sig}^*) = 1$ **then**
7 : **return** 1
8 : **else**
9 : **return** 0

Oracle $\mathcal{O}_{SK}(\mathsf{id})$

1 : $\mathsf{sk}_{\mathsf{id}} \leftarrow \mathsf{KGen}(\mathsf{mpk}, \mathsf{msk}, \mathsf{id})$
2 : $\mathcal{L}_{SK} \leftarrow \mathcal{L}_{SK} \cup \{\mathsf{id}\}$
3 : **return** $\mathsf{sk}_{\mathsf{id}}$

Oracle $\mathcal{O}_{SIG}(\mathsf{id}, \mathsf{m})$

1 : $\mathsf{sk}_{\mathsf{id}} \leftarrow \mathsf{KGen}(\mathsf{mpk}, \mathsf{msk}, \mathsf{id})$
2 : $\mathsf{sig} \leftarrow \mathsf{Sign}(\mathsf{mpk}, \mathsf{sk}_{\mathsf{id}}, \mathsf{m})$
3 : $\mathcal{L}_{SIG} \leftarrow \mathcal{L}_{SIG} \cup \{(\mathsf{id}, \mathsf{m})\}$
4 : **return** sig

Fig. 2: The security game for IBS.

Correctness. We say that an IBS scheme IBS is *correct* if for all $\lambda \in \mathbb{N}$, $\mathsf{id} \in \mathcal{ID}$ and $\mathsf{m} \in \mathcal{M}$, it holds that

$$\Pr\left[\mathsf{Ver}(\mathsf{mpk}, \mathsf{id}, \mathsf{m}, \mathsf{sig}) = 1 \;\middle|\; \begin{array}{l}(\mathsf{mpk}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda),\\ \mathsf{sk}_{\mathsf{id}} \leftarrow \mathsf{KGen}(\mathsf{mpk}, \mathsf{msk}, \mathsf{id}),\\ \mathsf{sig} \leftarrow \mathsf{Sign}(\mathsf{mpk}, \mathsf{sk}_{\mathsf{id}}, \mathsf{m})\end{array}\right] = 1 - \mathsf{negl}(\lambda).$$

Security. We recall adaptive-identity unforgeability against chosen-message attacks (EUF-ID-CMA security) [25].

Definition 4 (EUF-ID-CMA Security of IBS). *We say that an IBS scheme* IBS *is* EUF-ID-CMA secure *if for all* PPT *adversaries* $\mathcal{A}$*, it holds that*

$$\mathsf{Adv}^{\mathsf{euf\text{-}id\text{-}cma}}_{\mathcal{A},\mathsf{IBS}}(\lambda) := \Pr\left[\mathsf{EUF\text{-}ID\text{-}CMA}^{\mathcal{A}}_{\mathsf{IBS}}(\lambda) \Rightarrow 1\right] = \mathsf{negl}(\lambda),$$

where the security game $\mathsf{EUF\text{-}ID\text{-}CMA}^{\mathcal{A}}_{\mathsf{IBS}}(\lambda)$ *is depicted in Fig. 2.*

2.5 Reusable Computational Extractors

Let $\mathsf{Ext} : \mathcal{S} \times \mathcal{X} \to \mathcal{Y}$ be an efficiently computable function that on input a seed $s \in \mathcal{S}$ and a value $x \in \mathcal{X}$ outputs $y \in \mathcal{Y}$. Intuitively, we say that Ext is an extractor if $y = \mathsf{Ext}(s, x)$ is pseudorandom when s is sampled uniformly at random from $\mathcal{S}$ and x is sampled from a k-distribution X (defined over $\mathcal{X}$) for appropriate k, even if the seed s is made public. An extractor is *reusable* [14] if it produces pseudorandom outputs even if the same input is evaluated multiple times with different seeds. The formal definition is provided below.

Definition 5 (Reusable Computational Extractors). *We say that* $\mathsf{Ext} : \mathcal{S} \times \mathcal{X} \to \mathcal{Y}$ *is a* (k, n)-reusable computational extractor *if for any* k*-distribution* X *over* $\mathcal{X}$ *and for all* PPT *adversaries* $\mathcal{A}$*, it holds that*

$$\begin{aligned}\mathsf{Adv}^{\mathsf{ext}}_{\mathcal{A},\mathsf{Ext}}(\lambda) := \;& \left|\Pr\left[1 \leftarrow \mathcal{A}\left(\{(s_i, \mathsf{Ext}(s_i, x)\}_{i\in[n]}\right) \;\middle|\; s_i \leftarrow\!\$\, \mathcal{S}, x \leftarrow\!\$\, X\right]\right.\\ & \left. - \Pr\left[1 \leftarrow \mathcal{A}\left(\{(s_i, y_i)\}_{i\in[n]}\right) \;\middle|\; s_i \leftarrow\!\$\, \mathcal{S}, y_i \leftarrow\!\$\, \mathcal{Y}\right]\right|\\ = \;& \mathsf{negl}(\lambda).\end{aligned}$$

Reusable computational extractors can be constructed from random oracle [6], a strong variant of the DDH assumption [9], or the auxiliary-input LPN assumption [14].

3 Identity-Based Matchmaking Encryption

In this section, we first recall the syntax and security definition of identity-based matchmaking encryption (IB-ME) defined by Ateniese et al. [2]. Then, we introduce stronger security notions of them and reformulate privacy in cases of mismatch during decryption introduced by Francati et al. [18].

3.1 Syntax

An IB-ME scheme $\mathsf{IB\text{-}ME}$ consists of the following five algorithms.

$\mathsf{Setup}(1^\lambda) \to (\mathsf{mpk}, \mathsf{msk})$**:** The setup algorithm takes the security parameter 1^λ, and outputs a public parameter mpk and master secret key msk. mpk defines the identity space $\mathcal{ID}$, message space $\mathcal{M}$ and ciphertext space $\mathcal{CT}$.

$\mathsf{SKGen}(\mathsf{mpk}, \mathsf{msk}, \sigma) \to \mathsf{ek}_\sigma$**:** The sender key generation algorithm takes mpk, msk, and a sender's identity $\sigma \in \mathcal{ID}$ as input, and outputs an encryption key ek_σ.

$\mathsf{RKGen}(\mathsf{mpk}, \mathsf{msk}, \rho) \to \mathsf{dk}_\rho$**:** The receiver key generation algorithm takes mpk, msk, and a receiver's identity $\rho \in \mathcal{ID}$ as input, and outputs a decryption key dk_ρ.

$\mathsf{Enc}(\mathsf{mpk}, \mathsf{ek}_\sigma, \mathsf{rcv}, \mathsf{m}) \to \mathsf{ct}$**:** The encryption algorithm takes mpk, ek_σ, a receiver identity rcv, and a plaintext $\mathsf{m} \in \mathcal{M}$ as input, and outputs a ciphertext $\mathsf{ct} \in \mathcal{CT}$.

$\mathsf{Dec}(\mathsf{mpk}, \mathsf{dk}_\rho, \mathsf{snd}, \mathsf{ct}) \to \mathsf{m}$ **or** $\perp$**:** The decryption algorithm takes mpk, dk_ρ, a sender identity snd, and ct as input, and outputs $\mathsf{m} \in \mathcal{M}$ or a special symbol $\perp \notin \mathcal{M}$.

Correctness. We say that an IB-ME scheme $\mathsf{IB\text{-}ME}$ is *correct* if for all $\lambda \in \mathbb{N}, \sigma, \rho, \mathsf{snd}, \mathsf{rcv} \in \mathcal{ID}$ such that $\mathsf{snd} = \sigma$ and $\mathsf{rcv} = \rho$, and $\mathsf{m} \in \mathcal{M}$, it holds that

$$\Pr\left[\mathsf{Dec}(\mathsf{mpk}, \mathsf{dk}_\rho, \mathsf{snd}, \mathsf{ct}) = \mathsf{m} \;\middle|\; \begin{array}{l} (\mathsf{mpk}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda), \\ \mathsf{ek}_\sigma \leftarrow \mathsf{SKGen}(\mathsf{mpk}, \mathsf{msk}, \sigma), \\ \mathsf{dk}_\rho \leftarrow \mathsf{RKGen}(\mathsf{mpk}, \mathsf{msk}, \rho), \\ \mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{mpk}, \mathsf{ek}_\sigma, \mathsf{rcv}, \mathsf{m}) \end{array}\right] = 1 - \mathsf{negl}(\lambda).$$

We say that an IB-ME scheme is *perfectly correct* if the above probability is equal to 1 (i.e., no error occurs).

3.2 Security Notions

Standard security notions. IB-ME schemes must satisfy two primary security properties: *privacy* and *authenticity*. In essence, privacy ensures that nothing is disclosed to unintended recipients who do not adhere to the sender's policy, while authenticity guarantees that it is impossible to impersonate the sender without possessing the sender's secret key. We revisit the definitions of privacy and authenticity outlined by Ateniese et al. [2]. To clarify, we rename their definitions *privacy against chosen plaintext attacks* ($\mathsf{Priv\text{-}CPA}$), and *authenticity against no-message attacks from outsiders* ($\mathsf{Auth\text{-}oNMA}$). The term "outsiders" indicates neither the target sender nor the target receiver are compromised. Subsequently, an authenticity notion where adversaries can compromise the target receiver is considered [11, 18]. Since the adversary knows the target reciever's key, we call such adversary insiders, and call the corresponding authenticity notion *authenticity against no-message attacks from insiders* ($\mathsf{Auth\text{-}iNMA}$). It is worth noting that this distinction between insider and outsider adversaries is a well-established concept in the context of Signcryption [28].

The security games are depicted in Fig. 3. We remark that we employ a "real-or-random" style $\mathsf{Priv\text{-}CPA}$ game instead of the "left-or-right" style game of Ateniese et al. In greater detail, to account for sender and receiver anonymity, Ateniese et al. designed the security game where the adversary outputs $\{(\mathsf{snd}_i, \mathsf{rcv}_i, \mathsf{m}_i)\}_{i\in\{0,1\}}$ and presents a challenge ciphertext generated with one of them depending on the challenge bit $\mathsf{coin} \in \{0, 1\}$. In contrast, we define the game in a way that the adversary outputs $(\mathsf{snd}, \mathsf{rcv}, \mathsf{m})$ and is provided with either a real ciphertext generated using this information or a random ciphertext generated by a sampling algorithm $\mathsf{CTSamp}(\cdot)$ similar to the anonymity in IBE (cf. Section 2.3). In essence, our definition asserts that ciphertexts convey no information beyond what is derived from the master public keys. Although we do not furnish formal proof, our security definition encompasses Ateniese et al.'s security definition immediately.

Definition 6 (Priv-CPA **Security of IB-ME).** *We say that an IB-ME scheme* IB-ME *is* Priv-CPA secure *if for all* PPT *adversaries* $\mathcal{A}$*, it holds that*

$$\mathsf{Adv}^{\mathsf{priv\text{-}cpa}}_{\mathcal{A},\mathsf{IB\text{-}ME}}(\lambda) := \left|\Pr\left[\mathsf{Priv\text{-}CPA}^{\mathcal{A}}_{\mathsf{IB\text{-}ME}}(\lambda) \Rightarrow 1\right] - \frac{1}{2}\right| = \mathsf{negl}(\lambda),$$

where the security game $\mathsf{Priv\text{-}CPA}^{\mathcal{A}}_{\mathsf{IB\text{-}ME}}(\lambda)$ *is depicted in Fig. 3.*

Definition 7 (Auth-{o, i}NMA **Security of IB-ME).** *Let* $\mathsf{x} \in \{\mathsf{o}, \mathsf{i}\}$*. We say that an IB-ME scheme* IB-ME *is* Auth-xNMA secure *if for all* PPT *adversaries* $\mathcal{A}$*, it holds that*

$$\mathsf{Adv}^{\mathsf{auth\text{-}xnma}}_{\mathcal{A},\mathsf{IB\text{-}ME}}(\lambda) := \Pr\left[\mathsf{Auth\text{-}xNMA}^{\mathcal{A}}_{\mathsf{IB\text{-}ME}}(\lambda) \Rightarrow 1\right] = \mathsf{negl}(\lambda),$$

where the security game $\mathsf{Auth\text{-}xNMA}^{\mathcal{A}}_{\mathsf{IB\text{-}ME}}(\lambda)$ *is depicted in Fig. 3.*

Stronger security notions. In this work, we define stronger security notions for IB-ME. We consider *privacy against chosen-ciphertext attacks* (Priv-CCA) and *authenticity against chosen-message attacks from outsiders or insiders* (Auth-oCMA or Auth-iCMA). In the Priv-CCA game, the adversary can access the decryption oracle, similar to the standard CCA attack scenario. In the Auth-xCMA game, the adversary can access the encryption oracle and receive a ciphertext for a message of its choice, as with the signing oracle in the standard digital signature security game. These notions Priv-CCA and Auth-xCMA are the desired security properties in practice. We note that Priv-CCA security was first defined in [13], and Auth-iCMA is the same as the "strong authenticity" by Wang et al. [35].

Definition 8 (Priv-CCA **Security of IB-ME).** *We say that an IB-ME scheme* IB-ME *is* Priv-CCA secure *if for all* PPT *adversaries* $\mathcal{A}$*, it holds that*

$$\mathsf{Adv}^{\mathsf{priv\text{-}cca}}_{\mathcal{A},\mathsf{IB\text{-}ME}}(\lambda) := \left|\Pr\left[\mathsf{Priv\text{-}CCA}^{\mathcal{A}}_{\mathsf{IB\text{-}ME}}(\lambda) \Rightarrow 1\right] - \frac{1}{2}\right| = \mathsf{negl}(\lambda),$$

where the security game $\mathsf{Priv\text{-}CCA}^{\mathcal{A}}_{\mathsf{IB\text{-}ME}}(\lambda)$ *is depicted in Fig. 3.*

Definition 9 (Auth-{o, i}CMA **Security of IB-ME).** *Let* $\mathsf{x} \in \{\mathsf{o}, \mathsf{i}\}$*. We say that an IB-ME scheme* IB-ME *is* Auth-xCMA secure *if for all* PPT *adversaries* $\mathcal{A}$*, it holds that*

$$\mathsf{Adv}^{\mathsf{auth\text{-}xcma}}_{\mathcal{A},\mathsf{IB\text{-}ME}}(\lambda) := \Pr\left[\mathsf{Auth\text{-}xCMA}^{\mathcal{A}}_{\mathsf{IB\text{-}ME}}(\lambda) \Rightarrow 1\right] = \mathsf{negl}(\lambda),$$

where the security game $\mathsf{Auth\text{-}xCMA}^{\mathcal{A}}_{\mathsf{IB\text{-}ME}}(\lambda)$ *is depicted in Fig. 3.*

Privacy in the case of mismatch during decryption. We additionally consider the case where ciphertexts are decrypted with the valid receiver's key but mismatched sender's identities. Intuitively, IB-ME must ensure the privacy of messages in this case from the design concept of IB-ME. This guarantees that the adversary who compromises a receiver but has no knowledge about the sender cannot decrypt ciphertexts. This is a crucial security property of IB-ME, which is different from IB-SC, but the original work did not consider it explicitly[13]. Subsequently, Francati et al. [18] defined the security notion that captures privacy in cases of mismatch (called enhanced privacy). To model that the adversary does not know who the sender is, Francati et al. assumed that the target sender's identities are chosen from a high min-entropy distribution. Their definition effectively captures this intuition, but they used a single game that includes both conventional privacy and privacy in mismatch cases, complicating the understanding of the definition and security proofs.

[13] Ateniese et al. informally argued that their IB-ME scheme hides the message and the sender's identity in the case of mismatch, but they did not provide a formal model or a formal proof.

$\mathsf{Priv\text{-}CPA}^{\mathcal{A}}_{\mathsf{IB\text{-}ME}}(\lambda)$ $\boxed{\mathsf{Priv\text{-}CCA}^{\mathcal{A}}_{\mathsf{IB\text{-}ME}}(\lambda)}$

$\mathcal{L}_S, \mathcal{L}_R \coloneqq \emptyset$
$\mathsf{coin} \leftarrow\!\!\$\ \{0,1\}$
$(\mathsf{mpk}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda)$
$(\sigma^*, \mathsf{rcv}^*, \mathsf{m}^*) \leftarrow \mathcal{A}^{\mathcal{O}}(\mathsf{mpk})$
if $\mathsf{rcv}^* \in \mathcal{L}_R$ **then**
  return coin
$\mathsf{ek}_{\sigma^*} \leftarrow \mathsf{SKGen}(\mathsf{mpk}, \mathsf{msk}, \sigma^*)$
$\mathsf{ct}_0 \leftarrow \mathsf{Enc}(\mathsf{mpk}, \mathsf{ek}_{\sigma^*}, \mathsf{rcv}^*, \mathsf{m}^*)$
$\mathsf{ct}_1 \leftarrow \mathsf{CTSamp}(\mathsf{mpk})$
$\widehat{\mathsf{coin}} \leftarrow \mathcal{A}^{\mathcal{O}}(\mathsf{ct}_{\mathsf{coin}})$
if $\mathsf{coin} = \widehat{\mathsf{coin}}$ **then**
  return 1
else
  return 0

$\mathsf{Auth\text{-}xYYY}^{\mathcal{A}}_{\mathsf{IB\text{-}ME}}(\lambda)$

$/\!/\ \mathsf{x} \in \{\mathsf{o}, \mathsf{i}\}, \mathsf{YYY} \in \{\mathsf{NMA}, \mathsf{CMA}\}$
$\mathcal{L}_S, \mathcal{L}_R, \mathcal{L}_E \coloneqq \emptyset$
$(\mathsf{mpk}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda)$
$(\mathsf{snd}^*, \rho^*, \mathsf{ct}^*) \leftarrow \mathcal{A}^{\mathcal{O}}(\mathsf{mpk})$
$\mathsf{dk}_{\rho^*} \leftarrow \mathsf{RKGen}(\mathsf{mpk}, \mathsf{msk}, \rho^*)$;
$\mathsf{m}^* \leftarrow \mathsf{Dec}(\mathsf{mpk}, \mathsf{dk}_{\rho^*}, \mathsf{snd}^*, \mathsf{ct}^*)$
if $\mathsf{x} = \mathsf{o} \wedge \rho^* \in \mathcal{L}_R$ **then**
  return 0
if $\mathsf{YYY} = \mathsf{CMA}$
    $\wedge\ (\mathsf{snd}^*, \rho^*, \mathsf{m}^*) \in \mathcal{L}_E$ **then**
  return 0
if $\mathsf{m}^* \neq \bot \wedge \mathsf{snd}^* \notin \mathcal{L}_S$ **then**
  return 1
else
  return 0

Available Oracles

$\mathsf{Priv\text{-}CCA} : \mathcal{O} = \{\mathcal{O}_S, \mathcal{O}_R, \mathcal{O}_D\}$
$\mathsf{Auth\text{-}xCMA} : \mathcal{O} = \{\mathcal{O}_S, \mathcal{O}_R, \mathcal{O}_E\}$
$\text{Others} : \mathcal{O} = \{\mathcal{O}_S, \mathcal{O}_R\}$

Oracle $\mathcal{O}_S(\sigma)$

$\mathsf{ek}_\sigma \leftarrow \mathsf{SKGen}(\mathsf{mpk}, \mathsf{msk}, \sigma)$
$\mathcal{L}_S \leftarrow \mathcal{L}_S \cup \{\sigma\}$
return ek_σ

Oracle $\mathcal{O}_R(\rho)$

$\boxed{\textbf{if } \rho = \mathsf{rcv}^* \textbf{ then}}$
  $\boxed{\textbf{return } \bot}$
$\mathsf{dk}_\rho \leftarrow \mathsf{RKGen}(\mathsf{mpk}, \mathsf{msk}, \rho)$
$\mathcal{L}_R \leftarrow \mathcal{L}_R \cup \{\rho\}$
return dk_ρ

Oracle $\mathcal{O}_E(\sigma, \mathsf{rcv}, \mathsf{m})$

$\mathsf{ek}_\sigma \leftarrow \mathsf{SKGen}(\mathsf{mpk}, \mathsf{msk}, \sigma)$
$\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{mpk}, \mathsf{ek}_\sigma, \mathsf{rcv}, \mathsf{m})$
$\mathcal{L}_E \leftarrow \mathcal{L}_E \cup \{(\sigma, \mathsf{rcv}, \mathsf{m})\}$
return ct

Oracle $\mathcal{O}_D(\mathsf{snd}, \rho, \mathsf{ct})$

$\boxed{\textbf{if } (\mathsf{snd}, \rho, \mathsf{ct}) = (\sigma^*, \mathsf{rcv}^*, \mathsf{ct}_{\mathsf{coin}}) \textbf{ then}}$
  $\boxed{\textbf{return } \bot}$
$\mathsf{dk}_\rho \leftarrow \mathsf{RKGen}(\mathsf{mpk}, \mathsf{msk}, \rho)$
$\mathsf{m} \leftarrow \mathsf{Dec}(\mathsf{mpk}, \mathsf{snd}, \mathsf{dk}_\rho, \mathsf{ct})$
return m

Fig. 3: The security games for IB-ME schemes. The boxed lines are only for the Priv-CCA game.

$\mathsf{Priv\text{-}MisMatch}^{\mathcal{A}}_{\mathsf{IB\text{-}ME}}(\lambda)$

$\mathcal{L}_S, \mathcal{L}_R := \emptyset$
$\mathsf{coin} \leftarrow\!\!\$\ \{0,1\}$
$(\mathsf{mpk}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda)$
$(\Sigma_0, \Sigma_1, \mathsf{rcv}^*, \mathsf{m}_0, \mathsf{m}_1) \leftarrow \mathcal{A}^{\mathcal{O}_S, \mathcal{O}_R}(\mathsf{mpk})$
$\mathsf{dk}_{\mathsf{rcv}^*} \leftarrow \mathsf{RKGen}(\mathsf{mpk}, \mathsf{msk}, \mathsf{rcv}^*)$
for $i \in \{0,1\}$ **do**
  $\sigma_i^* \leftarrow\!\!\$\ \Sigma_i$ // Sample from the distribution.
  $\mathsf{ek}_{\sigma_i^*} \leftarrow \mathsf{SKGen}(\mathsf{mpk}, \mathsf{msk}, \sigma_i^*)$
  $\mathsf{ct}_i \leftarrow \mathsf{Enc}(\mathsf{mpk}, \mathsf{ek}_{\sigma_i^*}, \mathsf{rcv}^*, \mathsf{m}_i^*)$
$\widehat{\mathsf{coin}} \leftarrow \mathcal{A}^{\mathcal{O}_S, \mathcal{O}_R, \mathcal{O}_{E^*}}(\mathsf{dk}_{\mathsf{rcv}^*}, \mathsf{ct}_{\mathsf{coin}})$
if $\mathsf{coin} = \widehat{\mathsf{coin}}$ **then return** 1
else return 0

Oracle $\mathcal{O}_{E^*}(i \in \{0,1\}, \mathsf{rcv}, \mathsf{m})$

$\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{mpk}, \mathsf{ek}_{\sigma_i^*}, \mathsf{rcv}, \mathsf{m})$
return ct

Fig. 4: The privacy game in the case of mismatch for IB-ME schemes. The oracles $\mathcal{O}_S$ and $\mathcal{O}_R$ are defined in Fig. 3.

Therefore, in this work, we redefine the above intuition as another simple security game, which we call Priv-MisMatch security.

The new security game Priv-MisMatch is depicted in Fig. 3. The difference from Francati et al. is the adversary specifies one target receiver and is given the secret key of the target receiver explicitly. This represents the intuition that, even if the adversary knows the key of the target receiver, if it is difficult for the adversary to guess the sender's identity, the privacy of messages is guaranteed.

Definition 10 (Priv-MisMatch Security of IB-ME). *We say that an IB-ME scheme* IB-ME *is* Priv-MisMatch secure *if for all* PPT *adversaries* $\mathcal{A}$ *that output* $\omega(\log \lambda)$*-distributions* Σ_0 *and* Σ_1*, it holds that*

$$\mathsf{Adv}^{\mathsf{priv\text{-}mismatch}}_{\mathcal{A},\mathsf{IB\text{-}ME}}(\lambda) := \left| \Pr\left[\mathsf{Priv\text{-}MisMatch}^{\mathcal{A}}_{\mathsf{IB\text{-}ME}}(\lambda) \Rightarrow 1\right] - \frac{1}{2} \right| = \mathsf{negl}(\lambda),$$

where the security game $\mathsf{Priv\text{-}MisMatch}^{\mathcal{A}}_{\mathsf{IB\text{-}ME}}(\lambda)$ *is depicted in Fig. 4.*

4 Practical IB-ME from BDH in the ROM

In this section, we propose a practical IB-ME scheme from BDH assumption in the ROM. The idea of our scheme is to combine Boneh-Franklin IBE scheme [5] and Sakai-Ohgishi-Kasahara IB-NIKE scheme [33], and introduce several optimization to reduce key and ciphertext sizes. To achieve stronger security, we employ Fujisaki-Okamoto (FO) transformation [19,20]. Interestingly, the FO transformation allows us to achieve not only Priv-CCA security at minimum costs but also Auth-oCMA security for free. We also provide a formal proof of its Priv-MisMatch security. As a result, we obtain a highly efficient and strongly secure IB-ME scheme from the BDH assumption in the ROM compared with the BDH-based scheme by Ateniese et al. [2].

4.1 Construction

The proposed IB-ME scheme $\mathsf{IB\text{-}ME}^{\mathsf{BDH}}$ is as follows. Its identity and message spaces are $\mathcal{ID} = \{0,1\}^*$ and $\mathcal{M} = \{0,1\}^{\mathsf{mlen}}$, respectively.

$\mathsf{Setup}(1^\lambda)$**:** It first generates a bilinear group $G \coloneqq (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, g_1, g_2, e) \leftarrow \mathcal{G}(1^\lambda)$ and selects hash functions $\mathsf{H}_1 : \{0,1\}^* \to \mathbb{G}_1$, $\mathsf{H}_2 : \{0,1\}^* \to \mathbb{G}_2$, $\hat{\mathsf{H}} : \{0,1\}^* \times \{0,1\}^* \times \mathbb{G}_1 \times \mathbb{G}_T \times \mathbb{G}_T \to \{0,1\}^{\mathsf{mlen}+\lambda}$, and $\mathsf{G} : \{0,1\}^* \times \{0,1\}^* \times \{0,1\}^{\mathsf{mlen}} \times \{0,1\}^\lambda \to \mathbb{Z}_p$. Then, it samples $x \leftarrow\!\!\$\ \mathbb{Z}_p$ and sets $X \coloneqq g_1^x$. Finally, it outputs $\mathsf{mpk} \coloneqq (G, \mathsf{H}_1, \mathsf{H}_2, \hat{\mathsf{H}}, \mathsf{G}, X)$ and $\mathsf{msk} \coloneqq x$.
$\mathsf{SKGen}(\mathsf{mpk}, \mathsf{msk}, \sigma)$**:** It computes $\mathsf{u}_\sigma \coloneqq \mathsf{H}_1(\sigma)$ and outputs $\mathsf{ek}_\sigma \coloneqq \mathsf{u}_\sigma^x$.
$\mathsf{RKGen}(\mathsf{mpk}, \mathsf{msk}, \rho)$**:** It computes $\mathsf{u}_\rho \coloneqq \mathsf{H}_2(\rho)$ and outputs $\mathsf{dk}_\rho \coloneqq \mathsf{u}_\rho^x$.
$\mathsf{Enc}(\mathsf{mpk}, \mathsf{ek}_\sigma, \mathsf{rcv}, \mathsf{m})$**:** It picks $\mathsf{k} \leftarrow\!\!\$\ \{0,1\}^\lambda$ and computes $r \coloneqq \mathsf{G}(\sigma, \mathsf{rcv}, \mathsf{m}, \mathsf{k})$. Then, it computes $\mathsf{u}_{\mathsf{rcv}} \coloneqq \mathsf{H}_2(\mathsf{rcv})$ and

$$R \coloneqq g_1^r, \qquad \mathsf{ctxt} \coloneqq (\mathsf{m}||\mathsf{k}) \oplus \hat{\mathsf{H}}(\sigma, \mathsf{rcv}, R, e(X^r, \mathsf{u}_{\mathsf{rcv}}), e(\mathsf{ek}_\sigma, \mathsf{u}_{\mathsf{rcv}})).$$

Finally, it outputs $\mathsf{ct} \coloneqq (R, \mathsf{ctxt})$.
$\mathsf{Dec}(\mathsf{mpk}, \mathsf{dk}_\rho, \mathsf{snd}, \mathsf{ct} = (R, \mathsf{ctxt}))$**:** It computes $\mathsf{u}_{\mathsf{snd}} \coloneqq \mathsf{H}_1(\mathsf{snd})$ and

$$\mathsf{m}||\mathsf{k} \coloneqq \mathsf{ctxt} \oplus \hat{\mathsf{H}}(\mathsf{snd}, \rho, R, e(R, \mathsf{dk}_\rho), e(\mathsf{u}_{\mathsf{snd}}, \mathsf{dk}_\rho)).$$

It then computes $r \coloneqq \mathsf{G}(\mathsf{snd}, \rho, \mathsf{m}, \mathsf{k})$ and checks if $R = g_1^r$. If so, it outputs m. Otherwise, it outputs $\perp$.

Correctness. We can verify that $\mathsf{IB\text{-}ME}^{\mathsf{BDH}}$ is perfectly correct. For any $\lambda \in \mathbb{N}$, $(\mathsf{mpk}, \mathsf{msk}) \in \mathsf{Setup}(1^\lambda)$ and any $\sigma, \rho, \mathsf{snd}, \mathsf{rcv} \in \{0,1\}^*$ such that $\sigma = \mathsf{snd}$ and $\rho = \mathsf{rcv}$, we have

$$\begin{aligned} e(X^r, \mathsf{u}_{\mathsf{rcv}}) &= e((g_1^x)^r, \mathsf{H}_2(\mathsf{rcv})) = e(g_1^r, \mathsf{H}_2(\rho)^x) = e(R, \mathsf{dk}_\rho), \\ e(\mathsf{ek}_\sigma, \mathsf{u}_{\mathsf{rcv}}) &= e(\mathsf{H}_1(\sigma)^x, \mathsf{H}_2(\mathsf{rcv})) = e(\mathsf{H}_1(\mathsf{snd}), \mathsf{H}_2(\rho)^x) = e(\mathsf{u}_{\mathsf{snd}}, \mathsf{dk}_\rho). \end{aligned}$$

That is, it holds that

$$\hat{\mathsf{H}}(\sigma, \mathsf{rcv}, R, e(X, \mathsf{u}_{\mathsf{rcv}})^r, e(\mathsf{ek}_\sigma, \mathsf{u}_{\mathsf{rcv}})) = \hat{\mathsf{H}}(\mathsf{snd}, \rho, R, e(R, \mathsf{dk}_\rho), e(\mathsf{u}_{\mathsf{snd}}, \mathsf{dk}_\rho)),$$

and the receiver recovers $\mathsf{m}||\mathsf{k}$ that the sender $\sigma = \mathsf{snd}$ encrypts. Thus, the receiver can recompute $r \coloneqq \mathsf{G}(\mathsf{snd}, \rho, \mathsf{m}, \mathsf{k})$ that satisfies $R = g_1^r$.

4.2 Security Proof

We show that $\mathsf{IB\text{-}ME}^{\mathsf{BDH}}$ is Priv-CCA, Priv-MisMatch and Auth-oCMA secure in the ROM. First, we prove its Priv-CCA security. To do so, we use the intermediate scheme $\mathsf{IB\text{-}ME}^{\mathsf{Basic}}$, which is a simplified version of $\mathsf{IB\text{-}ME}^{\mathsf{BDH}}$. We prove that $\mathsf{IB\text{-}ME}^{\mathsf{Basic}}$ is Priv-CPA secure under the BDH assumption, and then prove the Priv-CCA security of $\mathsf{IB\text{-}ME}^{\mathsf{BDH}}$ assuming the Priv-CPA security of $\mathsf{IB\text{-}ME}^{\mathsf{Basic}}$.

Basic IB-ME scheme. The IB-ME scheme $\mathsf{IB\text{-}ME}^{\mathsf{Basic}}$ is as follows. The differences between $\mathsf{IB\text{-}ME}^{\mathsf{Basic}}$ and $\mathsf{IB\text{-}ME}^{\mathsf{BDH}}$ are that $\mathsf{IB\text{-}ME}^{\mathsf{Basic}}.\mathsf{Enc}$ samples uniform randomness r instead of generating it with a hash function G, and $\mathsf{IB\text{-}ME}^{\mathsf{Basic}}.\mathsf{Dec}$ does not perform the ciphertext validity check (i.e., do not check if $R = g_1^r$ holds). Its identity and message spaces are $\mathcal{ID} = \{0,1\}^*$ and $\mathcal{M} = \{0,1\}^{\mathsf{mlen}+\lambda}$, respectively. We can easily verify that $\mathsf{IB\text{-}ME}^{\mathsf{Basic}}$ is correct.

$\mathsf{Setup}(1^\lambda)$**:** It is identical to $\mathsf{IB\text{-}ME}^{\mathsf{BDH}}.\mathsf{Setup}$ except that G is not chosen.
$\mathsf{SKGen}(\mathsf{mpk}, \mathsf{msk}, \sigma)$**:** It is identical to $\mathsf{IB\text{-}ME}^{\mathsf{BDH}}.\mathsf{SKGen}$.
$\mathsf{RKGen}(\mathsf{mpk}, \mathsf{msk}, \rho)$**:** It is identical to $\mathsf{IB\text{-}ME}^{\mathsf{BDH}}.\mathsf{RKGen}$.
$\mathsf{Enc}(\mathsf{mpk}, \mathsf{ek}_\sigma, \mathsf{rcv}, \mathsf{m})$**:** It chooses $r \leftarrow\!\!\$\ \mathbb{Z}_p$ and computes $\mathsf{u}_{\mathsf{rcv}} \coloneqq \mathsf{H}_2(\mathsf{rcv})$ and

$$R \coloneqq g_1^r, \qquad \mathsf{ctxt} \coloneqq \mathsf{m} \oplus \hat{\mathsf{H}}(\sigma, \mathsf{rcv}, R, e(X^r, \mathsf{u}_{\mathsf{rcv}}), e(\mathsf{ek}_\sigma, \mathsf{u}_{\mathsf{rcv}})).$$

Finally, it outputs $\mathsf{ct} \coloneqq (R, \mathsf{ctxt})$.

$\mathsf{Dec}(\mathsf{mpk}, \mathsf{dk}_\rho, \mathsf{snd}, \mathsf{ct} = (R, \mathsf{ctxt}))$**:** It computes $\mathsf{u}_{\mathsf{snd}} \coloneqq \mathsf{H}_1(\mathsf{snd})$ and

$$\mathsf{m} \coloneqq \mathsf{ctxt} \oplus \hat{\mathsf{H}}(\mathsf{snd}, \rho, R, e(R, \mathsf{dk}_\rho), e(\mathsf{u}_{\mathsf{snd}}, \mathsf{dk}_\rho)).$$

Finally, it outputs m.

We show that $\mathsf{IB\text{-}ME}^{\mathsf{Basic}}$ is Priv-CPA secure.

Theorem 1. *Suppose the hash functions $\mathsf{H}_1, \mathsf{H}_2, \hat{\mathsf{H}}$ are random oracles. Under the BDH assumption for $\mathcal{G}$, $\mathsf{IB\text{-}ME}^{\mathsf{Basic}}$ is Priv-CPA secure in the ROM. Formally, if there exists an adversary $\mathcal{A}$ that breaks the Priv-CPA security of $\mathsf{IB\text{-}ME}^{\mathsf{Basic}}$, there exists an adversary $\mathcal{B}$ that breaks the BDH assumption for $\mathcal{G}$ such that*

$$\mathsf{Adv}^{\mathsf{priv\text{-}cpa}}_{\mathcal{A},\mathsf{IB\text{-}ME}^{\mathsf{Basic}}}(\lambda) \leq \hat{e}(1 + q_R) q_{\hat{\mathsf{H}}} \cdot \mathsf{Adv}^{\mathsf{bdh}}_{\mathcal{B},\mathcal{G}}(\lambda),$$

where q_R and $q_{\hat{\mathsf{H}}}$ are the maximum number of queries $\mathcal{A}$ sends to $\mathcal{O}_R$ and $\hat{\mathsf{H}}$ oracles, respectively. The running time of $\mathcal{B}$ is about that of $\mathcal{A}$.

Proof. Let $\mathsf{CTSamp}(\mathsf{mpk})$ be an algorithm that outputs a random element in $\mathbb{G}_1 \times \{0,1\}^{\mathsf{mlen}+\lambda}$. To prove the theorem, we consider the following sequence of games Game_i for $i \in \{0, 1, 2\}$. We define the advantage of $\mathcal{A}$ in Game_i as

$$\epsilon_i \coloneqq \left| \Pr\left[\mathsf{Game}_i^{\mathcal{A}}(\lambda) \Rightarrow 1\right] - \frac{1}{2} \right|.$$

Game_0 : This is the original security game. By definition, we have

$$\epsilon_0 = \mathsf{Adv}^{\mathsf{priv\text{-}cpa}}_{\mathcal{A},\mathsf{IB\text{-}ME}^{\mathsf{Basic}}}(\lambda).$$

Game_1 : In this game, we add abort conditions. We guess the challenge identity ρ^* that is not sent to $\mathcal{O}_R$ oracle. If the guess fails, the game aborts and sets a random coin as $\mathcal{A}$'s output. To do so, we change the challenger's procedures as follows. (The other procedures are worked as in the previous game.)

- When $\mathcal{A}$ sends ρ to H_2 oracle, it flips a coin d which yields 0 with probability $1 - \delta$. Then, it samples $b \leftarrow\!\!\$\, \mathbb{Z}_p$, computes $\mathsf{u}_\rho \coloneqq g_2^b$ and updates $\mathcal{L}_{\mathsf{H}_2} \leftarrow \mathcal{L}_{\mathsf{H}_2} \cup \{(\rho, \mathsf{u}_\rho, b, d)\}$. Then it returns u_ρ to $\mathcal{A}$.
- When $\mathcal{A}$ sends ρ to $\mathcal{O}_R$ oracle, it searches an entry $(\rho, \mathsf{u}_\rho, b, d) \in \mathcal{L}_{\mathsf{H}_2}$[14]. If $d = 0$, the game aborts. Otherwise (i.e., $d = 1$), it computes $\mathsf{dk}_\rho \coloneqq (g_2^x)^b$ and returns it to $\mathcal{A}$.
- When $\mathcal{A}$ outouts $(\sigma^*, \mathsf{rcv}^*, \mathsf{m}^*)$ to request a challenge ciphertext, it searches $(\mathsf{rcv}^*, \mathsf{u}_{\mathsf{rcv}^*}, b, d)$ from $\mathcal{L}_{\mathsf{H}_2}$. If $d = 1$, the game aborts. Otherwise (i.e., $d = 0$), it works as in Game_0.

The advantage of $\mathcal{A}$ in Game_1 is equal to the advantage of $\mathcal{A}$ in Game_0 conditioning on the game does not abort. Therefore, we have

$$\epsilon_1 = \epsilon_0 \cdot \Pr[\neg\mathsf{abort}].$$

Let us estimate the probability $\Pr[\neg\mathsf{abort}]$. The probability that the game does not abort in $\mathcal{O}_R$ oracle is δ^{q_R}. The probability the game does not abort when $\mathcal{A}$ request a challenge ciphertext is $1 - \delta$. Hence, the overall non-aborting probability is $\delta^{q_R}(1 - \delta)$. This value is maximum when $\hat{\delta} = \frac{q_R}{1+q_R}$, and thus we have $\Pr[\neg\mathsf{abort}] \leq \frac{1}{\hat{e}(1+q_R)}$ for large q_R. Therefore, we have

$$\epsilon_0 \leq \hat{e}(1 + q_R) \cdot \epsilon_1.$$

[14] If no entry exists, $\mathsf{H}_2(\rho)$ is internally queried and flips a coin d. (In the rest of this paper, when we have a similar situation, we also deal with it in the same manner.)

Game_2 : In this game, the challenge $\mathsf{ct}_0 \coloneqq (R^*, \mathsf{ctxt}^*)$ is computed as

$$r^* \leftarrow\!\$\, \mathbb{Z}_p, \quad Z \leftarrow\!\$\, \{0,1\}^{\mathsf{mlen}+\lambda}, \quad R^* \coloneqq g_1^{r^*}, \quad \mathsf{ctxt}^* \leftarrow \mathsf{m}^* \oplus Z.$$

Let BadQ be the event that $\mathcal{A}$ queries $(\cdot, \mathsf{rcv}^*, R^*, U^*, \cdot)$ to the oracle $\hat{\mathsf{H}}$ where $U^* \coloneqq e(R^*, \mathsf{dk}_{\mathsf{rcv}^*})$. Since Z is chosen independently at random from random oracles, $\mathcal{A}$ can distinguish the two games if BadQ occurs and otherwise they proceed identically. Thus, we have

$$|\epsilon_2 - \epsilon_1| \leq \Pr[\mathsf{BadQ}].$$

To estimate $\Pr[\mathsf{BadQ}]$, we show that if $\mathcal{A}$ triggers BadQ, we can construct an adversary $\mathcal{B}$ that solves the BDH problem.The construction of $\mathcal{B}$ is as follows.

1. Upon receiving $(G = (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, g_1, g_2, e), g_1^\alpha, g_2^\alpha, g_2^\beta, g_1^\gamma)$, $\mathcal{B}$ sets $X \coloneqq g_1^\alpha$ (i.e., msk is implicitly set α) and prepares three random oracles $\mathsf{H}_1, \mathsf{H}_2, \hat{\mathsf{H}}$ (i.e., initialize the lists $\mathcal{L}_{\mathsf{H}_1}, \mathcal{L}_{\mathsf{H}_2}, \mathcal{L}_{\hat{\mathsf{H}}}$). Also, $\mathcal{B}$ flip a coin $\mathsf{coin} \leftarrow\!\$\, \{0,1\}$. Then, $\mathcal{B}$ executes $\mathcal{A}$ on input $\mathsf{mpk} \coloneqq (G, \mathsf{H}_1, \mathsf{H}_2, \hat{\mathsf{H}}, X)$.
2. When $\mathcal{A}$ makes oracle queries, $\mathcal{B}$ answers them as follows:
   (a) When $\mathcal{A}$ sends σ to H_1 oracle, $\mathcal{B}$ samples $b \leftarrow\!\$\, \mathbb{Z}_p$ and computes $\mathsf{u}_\sigma \coloneqq g_1^b$. Then, $\mathcal{B}$ updates $\mathcal{L}_{\mathsf{H}_1} \leftarrow \mathcal{L}_{\mathsf{H}_1} \cup \{(\sigma, \mathsf{u}_\sigma, b)\}$ and returns u_σ to $\mathcal{A}$.
   (b) When $\mathcal{A}$ sends ρ to H_2 oracle, $\mathcal{B}$ samples $b \leftarrow\!\$\, \mathbb{Z}_p$. With probability $1-\delta$, $\mathcal{B}$ computes $\mathsf{u}_\rho \coloneqq (g_2^\beta)^b$ and updates $\mathcal{L}_{\mathsf{H}_2} \leftarrow \mathcal{L}_{\mathsf{H}_2} \cup \{(\rho, \mathsf{u}_\rho, b, 0)\}$. Otherwise, $\mathcal{B}$ computes $\mathsf{u}_\rho \coloneqq g_2^b$ and updates $\mathcal{L}_{\mathsf{H}_2} \leftarrow \mathcal{L}_{\mathsf{H}_2} \cup \{(\rho, \mathsf{u}_\rho, b, 1)\}$. Then, $\mathcal{B}$ returns u_ρ to $\mathcal{A}$.
   (c) When $\mathcal{B}$ sends (σ, ρ, R, U, V) to $\hat{\mathsf{H}}$ oracle, $\mathcal{B}$ samples $Z \leftarrow\!\$\, \{0,1\}^{\mathsf{mlen}}$ and updates $\mathcal{L}_{\hat{\mathsf{H}}} \leftarrow \mathcal{L}_{\hat{\mathsf{H}}} \cup \{(\sigma, \rho, R, U, V, Z)\}$. Then, $\mathcal{B}$ returns Z to $\mathcal{A}$.
   (d) When $\mathcal{A}$ sends σ to $\mathcal{O}_S$ oracle, $\mathcal{B}$ searches $(\sigma, \mathsf{u}_\sigma, b) \in \mathcal{L}_{\mathsf{H}_1}$ and computes $\mathsf{ek}_\sigma \coloneqq (g_1^\alpha)^b$. Then, $\mathcal{B}$ returns ek_σ to $\mathcal{A}$.
   (e) When $\mathcal{A}$ sends ρ to $\mathcal{O}_R$ oracle, $\mathcal{B}$ searches $(\rho, \mathsf{u}_\rho, b, d) \in \mathcal{L}_{\mathsf{H}_2}$. If $d = 0$, $\mathcal{B}$ aborts the game. Otherwise (i.e., $d = 1$), $\mathcal{B}$ computes $\mathsf{dk}_\rho \coloneqq (g_2^\alpha)^b$. Then, $\mathcal{B}$ returns dk_ρ to $\mathcal{A}$.
   (f) When $\mathcal{A}$ outputs $(\sigma^*, \mathsf{rcv}^*, \mathsf{m}^*)$ to request a challenge ciphertext, $\mathcal{B}$ searches $(\mathsf{rcv}^*, \mathsf{u}_{\mathsf{rcv}^*}, b^*, d^*) \in \mathcal{L}_{\mathsf{H}_2}$. If $d^* = 1$, $\mathcal{B}$ aborts the game. Otherwise, $\mathcal{B}$ sets $R^* \coloneqq g_1^\gamma$ and computes $\mathsf{ctxt}^* \coloneqq \mathsf{m}^* \oplus Z$ where $Z \leftarrow\!\$\, \{0,1\}^{\mathsf{mlen}+\lambda}$. Then $\mathcal{B}$ sets $\mathsf{ct}_0 \coloneqq (R^*, \mathsf{ctxt}^*)$ and $\mathsf{ct}_1 \leftarrow\!\$\, \mathcal{CT}$, and returns $\mathsf{ct}_{\mathsf{coin}}$ to $\mathcal{A}$.
3. Finally, $\mathcal{A}$ outputs a guess $\widehat{\mathsf{coin}}$. Then, $\mathcal{B}$ picks an entry $(\cdot, \mathsf{rcv}^*, R^*, U^*, \cdot) \in \mathcal{L}_{\hat{\mathsf{H}}}$ at random and outputs $D \coloneqq (U^*)^{\frac{1}{b^*}}$ as the solution of the BDH problem.

We can see that $\mathcal{B}$ perfectly simulates the $\mathsf{Priv\text{-}CPA}$ game against $\mathcal{A}$ if $\mathcal{B}$ does not abort. Moreover, we know that $\mathsf{dk}_{\mathsf{rcv}^*} = (\mathsf{u}_{\mathsf{rcv}^*})^\alpha = (g_2^{\alpha\beta})^{b^*}$ and $R^* = g_1^\gamma$, and thus

$$U^* = e(R^*, \mathsf{dk}_{\mathsf{rcv}^*}) = e(g_1^\gamma, g_2^{\alpha\beta b^*}) = (e(g_1, g_2)^{\alpha\beta\gamma})^{b^*}.$$

If $\mathcal{A}$ distinguish the two games, $\mathcal{A}$ has queried $\hat{\mathsf{H}}(\cdot, \mathsf{rcv}^*, R^*, U^*, \cdot)$, and thus with probability at least $\frac{1}{q_{\hat{\mathsf{H}}}}$, $\mathcal{B}$ can solve the BDH problem correctly. Thus we have

$$|\epsilon_2 - \epsilon_1| \leq \Pr[\mathsf{BadQ}] \leq q_{\hat{\mathsf{H}}} \cdot \mathsf{Adv}^{\mathsf{bdh}}_{\mathcal{G},\mathcal{B}}(\lambda).$$

In Game_2, both ct_0 and ct_1 are chosen at random from the ciphertext space. Since coin is information-theoretically hidden from $\mathcal{A}$, we have $\epsilon_2 = 0$.

Putting everything together, we obtain

$$\mathsf{Adv}^{\mathsf{priv\text{-}cpa}}_{\mathcal{A},\mathsf{IB\text{-}ME}^{\mathsf{Basic}}}(\lambda) \leq \hat{e}(1 + q_R) q_{\hat{\mathsf{H}}} \cdot \mathsf{Adv}^{\mathsf{bdh}}_{\mathcal{B},\mathcal{G}}(\lambda).$$

□

We now prove the $\mathsf{Priv\text{-}CCA}$ security of $\mathsf{IB\text{-}ME}^{\mathsf{BDH}}$ assuming the $\mathsf{Priv\text{-}CPA}$ security of $\mathsf{IB\text{-}ME}^{\mathsf{Basic}}$. The proof is similar to the proof of the FO transformation for PKE/IBE schemes [19, 20].

Theorem 2. *Suppose the hash function* G *is a random oracle. If* $\mathsf{IB\text{-}ME}^{\mathsf{Basic}}$ *is* $\mathsf{Priv\text{-}CPA}$ *secure, then* $\mathsf{IB\text{-}ME}^{\mathsf{BDH}}$ *is* $\mathsf{Priv\text{-}CCA}$ *secure in the ROM. Formally, if there exists an adversary* $\mathcal{A}$ *that breaks the* $\mathsf{Priv\text{-}CCA}$ *security of* $\mathsf{IB\text{-}ME}^{\mathsf{BDH}}$*, there exists an adversary* $\mathcal{B}$ *that breaks the* $\mathsf{Priv\text{-}CPA}$ *security of* $\mathsf{IB\text{-}ME}^{\mathsf{Basic}}$ *such that*

$$\mathsf{Adv}^{\mathsf{priv\text{-}cca}}_{\mathcal{A},\mathsf{IB\text{-}ME}^{\mathsf{BDH}}}(\lambda) \leq 3\mathsf{Adv}^{\mathsf{priv\text{-}cpa}}_{\mathcal{B},\mathsf{IB\text{-}ME}^{\mathsf{Basic}}}(\lambda) + \frac{q_{\mathsf{Dec}}}{p} + \frac{3q_{\mathsf{G}}}{2^{\lambda}}.$$

where p *is the order of the underlying bilinear group, and* q_D *and* q_{G} *are the maximum number of queries* $\mathcal{A}$ *makes to* $\mathcal{O}_D$ *and* G *oracles, respectively. The running time of* $\mathcal{B}$ *is about that of* $\mathcal{A}$*.*

Proof. To prove the theorem, we consider the following sequence of games Game_i for $i \in \{0, \cdots, 5\}$. Define the advantage of $\mathcal{A}$ in Game_i as

$$\epsilon_i := \left| \Pr\left[\mathsf{Game}_i^{\mathcal{A}}(\lambda) \Rightarrow 1\right] - \frac{1}{2} \right|.$$

Game_0 : This is the original security game. By definition, we have

$$\epsilon_0 = \mathsf{Adv}^{\mathsf{priv\text{-}cca}}_{\mathcal{A},\mathsf{IB\text{-}ME}^{\mathsf{BDH}}}(\lambda).$$

Game_1 : In this game, the randomness $\mathsf{k}^* \in \{0,1\}^{\lambda}$ (used to generate the challenge ciphertext) is chosen in the setup phase instead of the challenge phase. Since there is no difference in $\mathcal{A}$'s view, we have

$$\epsilon_1 = \epsilon_0.$$

Game_2 : In this game, we change the behavior of G oracle. When $\mathcal{A}$ sends a tuple $(\sigma, \rho, \mathsf{m}, \mathsf{k})$ to G, the challenger picks $r \leftarrow\!\!\$\, \mathbb{Z}_p$, and computes

$$\mathsf{ek}_\sigma := \mathsf{H}_1(\sigma)^x, \qquad \mathsf{ct} \leftarrow \mathsf{IB\text{-}ME}^{\mathsf{Basic}}.\mathsf{Enc}(\mathsf{mpk}^{15}, \mathsf{ek}_\sigma, \rho, \mathsf{m}||\mathsf{k}; r).$$

Then, it updates $\mathcal{L}_{\mathsf{G}} \leftarrow \mathcal{L}_{\mathsf{G}} \cup \{((\sigma, \rho, \mathsf{m}, \mathsf{k}), r, \mathsf{ct})\}$ and returns r to $\mathcal{A}$. Since there is no difference in the behaviors of oracles from $\mathcal{A}$'s viewpoint, we have

$$\epsilon_2 = \epsilon_1.$$

We remark that ek_σ is unique for each identity σ, and thus the ciphertext computed as above can be uniquely determined by $(\sigma, \rho, \mathsf{m}, \mathsf{k})$.

Game_3 : In this game, we change the behavior of $\mathcal{O}_D$ oracle. When $\mathcal{A}$ sends $(\mathsf{snd}, \rho, \mathsf{ct})$ to $\mathcal{O}_D$, it finds an entry $((\mathsf{snd}, \rho, \mathsf{m}, \mathsf{k}), r, \mathsf{ct}) \in \mathcal{L}_{\mathsf{G}}$. If such a tuple exists, $\mathsf{m}||\mathsf{k}$ is returned to $\mathcal{A}$. Otherwise, $\perp$ is returned to $\mathcal{A}$.

Let BadD be the event that $\mathcal{A}$ submits a decryption query on $(\mathsf{snd}, \rho, \mathsf{ct})$ such that $((\mathsf{snd}, \rho, \mathsf{m}, \mathsf{k}), r, \mathsf{ct}) \notin \mathcal{L}_{\mathsf{G}}$ but it is not rejected in the previous game. Due to the perfect correctness of the scheme, the two games proceed identically unless BadD occurs. Thus, we have

$$|\epsilon_3 - \epsilon_2| \leq \Pr[\mathsf{BadD}].$$

We now estimate $\Pr[\mathsf{BadD}]$. In the previous game, if $((\mathsf{snd}, \rho, \mathsf{m}, \mathsf{k}), r, \mathsf{ct}) \notin \mathcal{L}_{\mathsf{G}}$ when $(\mathsf{snd}, \rho, \mathsf{ct})$ is sent to $\mathcal{O}_D$, $\mathsf{G}(\mathsf{snd}, \rho, \mathsf{m}, \mathsf{k})$ is queried internally and $r \leftarrow\!\!\$\, \mathbb{Z}_q$ is sampled. Then, $\mathcal{O}_D$ checks whether $R = g_1^r$ holds. For any $R \in \mathbb{G}_1$, the probability that $R = g_1^r$ holds for randomly chosen $r \in \mathbb{Z}_p$ is $1/p$. Since $\mathcal{A}$ queries $\mathcal{O}_D$ at most q_D, we have

$$|\epsilon_3 - \epsilon_2| \leq \Pr[\mathsf{BadD}] \leq \frac{q_D}{p}.$$

After this game, the decryption oracle is simulated without any decryption keys.

[15] For simplicity, we use the same symbol mpk for $\mathsf{IB\text{-}ME}^{\mathsf{Basic}}$ and $\mathsf{IB\text{-}ME}^{\mathsf{BDH}}$ since mpk of $\mathsf{IB\text{-}ME}^{\mathsf{BDH}}$ covers that of $\mathsf{IB\text{-}ME}^{\mathsf{Basic}}$.

Game_4 : In this game, we add an abort condition into G oracle. If $\mathcal{A}$ sends a tuple $(\cdot,\cdot,\cdot,\mathsf{k})$ such that $\mathsf{k}=\mathsf{k}^*$ before the challenge phase, the game aborts. Since $\mathsf{k}^* \in \{0,1\}^\lambda$ is chosen at random and information-theoretically hidden from $\mathcal{A}$ before the challenge phase, we have

$$|\epsilon_4 - \epsilon_3| \leq \frac{q_\mathsf{G}}{2^\lambda}.$$

Game_5 : In this game, we change how to generate the challenge ciphertext ct_0. To generate ct_0, the challenger chooses $r^* \leftarrow\!\!\$\ \mathbb{Z}_p$ and computes

$$\mathsf{ek}_{\sigma^*} := \mathsf{H}_1(\sigma^*)^x, \mathsf{ct}_0 \leftarrow \mathsf{IB\text{-}ME}^{\mathsf{Basic}}.\mathsf{Enc}(\mathsf{mpk}, \mathsf{ek}_{\sigma^*}, \mathsf{rcv}^*, \mathsf{m}^*||\mathsf{k}^*; r^*).$$

Now, the randomness r^* is chosen independently from G. Let BadQ be the event that $\mathcal{A}$ sends $(\cdot,\cdot,\cdot,\mathsf{k}^*)$ to G oracle after it requests the challenge ciphertext. Since $\mathcal{A}$'s view is identical unless BadQ occurs, we have

$$|\epsilon_5 - \epsilon_4| \leq \Pr[\mathsf{BadQ}].$$

To estimate $\Pr[\mathsf{BadQ}]$, we show that if $\mathcal{A}$ can trigger the event BadQ, there exists an adversary $\mathcal{B}_1$ that breaks the Priv-CPA security of $\mathsf{IB\text{-}ME}^{\mathsf{Basic}}$.

The construction of $\mathcal{B}_1$ is as follows. Upon receiving mpk (of $\mathsf{IB\text{-}ME}^{\mathsf{Basic}}$), $\mathcal{B}_1$ samples $\mathsf{k}^* \leftarrow\!\!\$\ \{0,1\}^\lambda$, prepares mpk of $\mathsf{IB\text{-}ME}^{\mathsf{BDH}}$, and executes $\mathcal{A}$ on input it. Then, $\mathcal{B}_1$ simulates the Priv-CCA game against $\mathcal{A}$ as in Game_5. When a query is sent to $\mathcal{O}_S$ or $\mathcal{O}_R$ oracle, $\mathcal{B}_1$ uses its oracles to generate encryption or decryption keys. When $\mathcal{A}$ requests a challenge ciphertext on $(\sigma^*, \mathsf{rcv}^*, \mathsf{m}^*)$, $\mathcal{B}_1$ sends $(\sigma^*, \mathsf{rcv}^*, \mathsf{m}^*||\mathsf{k}^*)$ to its challenger, receiving the challenge ciphertext ct^*. $\mathcal{B}_1$ forwards it to $\mathcal{A}$. When $\mathcal{A}$ triggers the event BadQ, $\mathcal{B}_1$ outputs $\widehat{\mathsf{coin}} := 0$ to its challenger as its guess of coin. If $\mathcal{A}$ does not trigger the event BadQ, $\mathcal{B}_1$ outputs a randomly chosen $\widehat{\mathsf{coin}} \leftarrow\!\!\$\ \{0,1\}$ to its challenger.

Now, we evaluate the $\mathcal{B}_1$'s advantage. Let Fail be the event that BadQ occurs when $\widehat{\mathsf{coin}} = 1$ (i.e., ct^* is sampled by CTSamp). Since k^* is uniformly distributed and independent from $\mathcal{B}_1$'s view when ct^* is sampled by CTSamp, $\Pr[\mathsf{Fail}] \leq q_\mathsf{G}/2^\lambda$. Assume Fail did not happen, i.e., BadQ occurs only when $\widehat{\mathsf{coin}} = 0$. Since $\mathcal{B}_1$ always outputs 0 when BadQ occurs, $\Pr\left[\mathsf{coin} = \widehat{\mathsf{coin}}\right] = 1$. If BadQ did not occur, $\mathcal{B}_1$ outputs a random coin and thus $\Pr\left[\mathsf{coin} = \widehat{\mathsf{coin}}\right] = 1/2$. Thus, we have

$$\begin{aligned}\mathsf{Adv}^{\mathsf{priv\text{-}cpa}}_{\mathcal{B}_1,\mathsf{IB\text{-}ME}^{\mathsf{Basic}}}(\lambda) + \frac{q_\mathsf{G}}{2^\lambda} &\geq \left|\Pr\left[\mathsf{coin} = \widehat{\mathsf{coin}}\right] - \frac{1}{2}\right| \\ &= \left|\Pr[\mathsf{BadQ}] + \frac{1}{2}\Pr[\neg\mathsf{BadQ}] - \frac{1}{2}\right| = \frac{1}{2}\Pr[\mathsf{BadQ}].\end{aligned}$$

Therefore, we have

$$|\epsilon_5 - \epsilon_4| \leq \Pr[\mathsf{BadQ}] \leq 2\mathsf{Adv}^{\mathsf{priv\text{-}cpa}}_{\mathcal{B}_1,\mathsf{IB\text{-}ME}^{\mathsf{Basic}}}(\lambda) + \frac{2q_\mathsf{G}}{2^\lambda}.$$

We finally bound ϵ_5. If $\mathcal{A}$ can breaks the Priv-CCA security in Game_5, there exisits an adversary $\mathcal{B}_2$ that breaks the Priv-CPA security of $\mathsf{IB\text{-}ME}^{\mathsf{Basic}}$ such that

$$\epsilon_5 = \mathsf{Adv}^{\mathsf{priv\text{-}cpa}}_{\mathcal{B}_2,\mathsf{IB\text{-}ME}^{\mathsf{Basic}}}(\lambda).$$

The proof is straightforward because $\mathcal{B}_2$ can simulate $\mathcal{O}_D$ without any decryption keys and the challenge ciphertext is generated with independent randomness r^*.

Putting everything together and folding both adversaries $\mathcal{B}_1$ and $\mathcal{B}_2$ into one adversary $\mathcal{B}$, we obtain

$$\mathsf{Adv}^{\mathsf{priv\text{-}cca}}_{\mathcal{A},\mathsf{IB\text{-}ME}^{\mathsf{BDH}}}(\lambda) \leq 3\mathsf{Adv}^{\mathsf{priv\text{-}cpa}}_{\mathcal{B},\mathsf{IB\text{-}ME}^{\mathsf{Basic}}}(\lambda) + \frac{q_D}{p} + \frac{3q_\mathsf{G}}{2^\lambda}.$$

□

We then prove that IB-ME$^{\mathsf{BDH}}$ is Priv-MisMatch secure in the ROM.

Theorem 3. IB-ME$^{\mathsf{BDH}}$ *is* Priv-MisMatch *secure in the ROM. Formally, an adversary* $\mathcal{A}$ *attacking the* Priv-MisMatch *security of* IB-ME$^{\mathsf{BDH}}$ *has advantage*

$$\mathsf{Adv}^{\mathsf{priv\text{-}mismatch}}_{\mathcal{A},\mathsf{IB\text{-}ME}^{\mathsf{BDH}}}(\lambda) \leq \frac{q_{\hat{\mathsf{H}}} + q_{\mathsf{G}}}{2^{\omega(\log\lambda)-1}}.$$

where $q_{\hat{\mathsf{H}}}$ *and* q_{G} *are the maximum number of queries* $\mathcal{A}$ *makes to* $\hat{\mathsf{H}}$ *and* G *oracles, respectively. The running time of* $\mathcal{B}$ *is about that of* $\mathcal{A}$*.*

Proof. To prove the theorem, we consider the following sequence of games Game_i for $i \in \{0, \cdots, 3\}$. Define the advantage of $\mathcal{A}$ in Game_i as

$$\epsilon_i := \left| \Pr\left[\mathsf{Game}_i^{\mathcal{A}}(\lambda) \Rightarrow 1\right] - \frac{1}{2} \right|.$$

Game_0 : This is the original security game. By definition, we have

$$\epsilon_0 = \mathsf{Adv}^{\mathsf{priv\text{-}mismatch}}_{\mathcal{A},\mathsf{IB\text{-}ME}^{\mathsf{BDH}}}(\lambda).$$

Game_1 : In this game, the challenger aborts the game if σ_0^* or σ_1^* are sent to $\hat{\mathsf{H}}$ or G oracle before $\mathcal{A}$ requests the challenge ciphertext. Since both of them are chosen independently at random and from $\omega(\log\lambda)$-distribution, we have

$$|\epsilon_1 - \epsilon_0| \leq \frac{q_{\hat{\mathsf{H}}} + q_{\mathsf{G}}}{2^{\omega(\log\lambda)}}.$$

Game_2 : In this game, the challenge ciphertext ct_i $(i \in \{0,1\})$ is computed as $\mathsf{ct}_i \leftarrow (g_1^{r_i}, (\mathsf{m}_0||\mathsf{k}_0) \oplus Z_i)$ for random $r_i \leftarrow\!\!\$\ \mathbb{Z}_p$ and $Z_i \leftarrow\!\!\$\ \{0,1\}^{\mathsf{mlen}+\lambda}$. $\mathcal{A}$ may notice this change when it sends σ_0^* or σ_1^* to $\hat{\mathsf{H}}$ or G oracle. Since σ_0^* and σ_1^* are chosen independently at random from $\omega(\log\lambda)$-distribution, we have

$$|\epsilon_2 - \epsilon_1| \leq \frac{q_{\hat{\mathsf{H}}} + q_{\mathsf{G}}}{2^{\omega(\log\lambda)}}.$$

In Game_2, both ct_0 and ct_1 are distributed uniformly at random. Since coin is information-theoretically hidden from $\mathcal{A}$, we have

$$\epsilon_2 = 0.$$

Putting everything together, we obtain

$$\mathsf{Adv}^{\mathsf{priv\text{-}mismatch}}_{\mathcal{A},\mathsf{IB\text{-}ME}^{\mathsf{BDH}}}(\lambda) \leq \frac{q_{\hat{\mathsf{H}}} + q_{\mathsf{G}}}{2^{\omega(\log\lambda)-1}}.$$

□

We finally show that the Auth-oCMA security of IB-ME$^{\mathsf{BDH}}$ under the BDH assumption. To prove it, we need to simulate the encryption oracle $\mathcal{O}_E$ without knowing the senders' encryption key while the adversary adaptively compromises senders. To do so, we employ the programmability of RO, similar to the proof technique for non-committing encryption in the ROM [30].

Theorem 4. *Suppose the hash functions* H_1, H_2, $\hat{\mathsf{H}}$, *and* G *are random oracles. Under the BDH assumption,* IB-ME$^{\mathsf{BDH}}$ *is* Auth-oCMA *secure in the ROM. Formally, if there exists an adversary* $\mathcal{A}$ *that breaks the* Auth-oCMA *security of* IB-ME$^{\mathsf{BDH}}$*, there exists an adversary* $\mathcal{B}$ *that breaks the BDH assumption such that*

$$\mathsf{Adv}^{\mathsf{auth\text{-}ocma}}_{\mathcal{A},\mathsf{IB\text{-}ME}^{\mathsf{BDH}}}(\lambda) \leq \frac{\hat{e}^2(q_S + q_R)^2 q_{\hat{\mathsf{H}}}}{2} \cdot \mathsf{Adv}^{\mathsf{bdh}}_{\mathcal{B},\mathcal{G}}(\lambda) + \frac{q_{\mathsf{G}}}{2^{\mathsf{mlen}+\lambda}} + \frac{1}{p},$$

where p *is the order of the underlying bilinear group and* q_S, q_R, $q_{\hat{\mathsf{H}}}$, *and* q_{G} *are the maximum number of queries* $\mathcal{A}$ *makes to* $\mathcal{O}_S$, $\mathcal{O}_R$, $\hat{\mathsf{H}}$, *and* G *oracles, respectively. The running time of* $\mathcal{B}$ *is about that of* $\mathcal{A}$*.*

Proof. To prove the theorem, we consider the following sequence of games Game_i for $i \in \{0, \cdots, 3\}$. Define the advantage of $\mathcal{A}$ in Game_i as

$$\epsilon_i := \Pr\left[\mathsf{Game}_i^{\mathcal{A}}(\lambda) \Rightarrow 1\right].$$

Game_0 : This is the original Auth-oCMA game. By definition, we have

$$\epsilon_0 = \mathsf{Adv}^{\mathsf{auth\text{-}ocma}}_{\mathcal{A},\mathsf{IB\text{-}ME}^{\mathsf{BDH}}}(\lambda).$$

Game_1 : In this game, we change the behavior of $\mathcal{O}_S$, $\mathcal{O}_R$, and $\mathcal{O}_E$ as follows.

- When $\mathcal{A}$ sends σ to $\mathcal{O}_S$ oracle, it computes $\mathsf{ek}_\sigma \coloneqq \mathsf{H}_1(\sigma)^x$. Then, it searches entries $(\mathsf{snd}, \mathsf{rcv}, \mathsf{m}||\mathsf{k}, \mathsf{ctxt}) \in \mathcal{L}_E$ such that $\mathsf{snd} = \sigma$. If such entries exist, it works as follows for each such entry. Let $\mathsf{u}_{\mathsf{rcv}} \coloneqq \mathsf{H}_2(\mathsf{rcv})$ and $r \coloneqq \mathsf{G}(\sigma, \mathsf{rcv}, \mathsf{m}, \mathsf{k})$.
  - If there exists an entry $(\mathsf{snd}, \mathsf{rcv}, g_1^r, e(X^r, \mathsf{u}_{\mathsf{rcv}}), e(\mathsf{ek}_\sigma, \mathsf{u}_{\mathsf{rcv}}), *) \in \mathcal{L}_{\hat{\mathsf{H}}}$, it aborts the game. (In this case, it cannot program the random oracle.)
  - Else, it updates

$$\mathcal{L}_{\hat{\mathsf{H}}} \leftarrow \mathcal{L}_{\hat{\mathsf{H}}} \cup \{(\mathsf{snd}, \mathsf{rcv}, g_1^r, e(X^r, \mathsf{u}_{\mathsf{rcv}}), e(\mathsf{ek}_\sigma, \mathsf{u}_{\mathsf{rcv}}), \mathsf{ctxt} \oplus (\mathsf{m}||\mathsf{k}))\}.$$

  After that, it removes the programmed entries from $\mathcal{L}_E$.

  Finally, it returns ek_σ to $\mathcal{A}$.
- When $\mathcal{A}$ sends ρ to $\mathcal{O}_R$ oracle, it computes $\mathsf{dk}_\rho \coloneqq \mathsf{H}_2(\rho)^x$. Then, it searches entries $(\mathsf{snd}, \mathsf{rcv}, \mathsf{m}||\mathsf{k}, \mathsf{ctxt}) \in \mathcal{L}_E$ such that $\mathsf{rcv} = \rho$. If such entries exist, it works as follows for each such entry. Let $\mathsf{u}_{\mathsf{snd}} \coloneqq \mathsf{H}_1(\mathsf{snd})$ and $r \coloneqq \mathsf{G}(\mathsf{snd}, \rho, \mathsf{m}, \mathsf{k})$.
  - If there exists an entry $(\mathsf{snd}, \mathsf{rcv}, g_1^r, e(g_1^r, \mathsf{dk}_\rho), e(\mathsf{H}_1(\mathsf{snd}), \mathsf{dk}_\rho), *) \in \mathcal{L}_{\hat{\mathsf{H}}}$, it aborts the game.
  - Else, for each entry, it updates

$$\mathcal{L}_{\hat{\mathsf{H}}} \leftarrow \mathcal{L}_{\hat{\mathsf{H}}} \cup \{(\mathsf{snd}, \mathsf{rcv}, g_1^r, e(g_1^r, \mathsf{dk}_\rho), e(\mathsf{u}_{\mathsf{snd}}, \mathsf{dk}_\rho), \mathsf{ctxt} \oplus (\mathsf{m}||\mathsf{k}))\}.$$

  Finally, it returns dk_ρ to $\mathcal{A}$.
- When $\mathcal{A}$ sends a tuple $(\sigma, \mathsf{rcv}, \mathsf{m})$ to $\mathcal{O}_E$ oracle, it samples $\mathsf{k} \leftarrow\!\!\$\ \{0,1\}^\lambda$ and computes $r \coloneqq \mathsf{G}(\sigma, \mathsf{rcv}, \mathsf{m}, \mathsf{k})$ and $R \coloneqq g_1^r$. Then, it computes ctxt as follows.
  1. If $\sigma \in \mathcal{L}_S$, it retrieves ek_σ[16] and computes $\mathsf{u}_{\mathsf{rcv}} \coloneqq \mathsf{H}_2(\mathsf{rcv})$ and

$$\mathsf{ctxt} \coloneqq (\mathsf{m}||\mathsf{k}) \oplus \hat{\mathsf{H}}(\sigma, \mathsf{rcv}, R, e(X^r, \mathsf{u}_{\mathsf{rcv}}), e(\mathsf{ek}_\sigma, \mathsf{u}_{\mathsf{rcv}})).$$

  2. If $\sigma \notin \mathcal{L}_S$ and $\mathsf{rcv} \in \mathcal{L}_R$, it retrieves $\mathsf{dk}_{\mathsf{rcv}}$[17] and computes $\mathsf{u}_{\mathsf{snd}} \coloneqq \mathsf{H}_1(\mathsf{snd})$ and

$$\mathsf{ctxt} \coloneqq (\mathsf{m}||\mathsf{k}) \oplus \hat{\mathsf{H}}(\sigma, \mathsf{rcv}, R, e(R, \mathsf{dk}_{\mathsf{rcv}}), e(\mathsf{u}_{\mathsf{snd}}, \mathsf{dk}_{\mathsf{rcv}}))$$

  3. If $\sigma \notin \mathcal{L}_S$ and $\mathsf{rcv} \notin \mathcal{L}_R$, it samples $\mathsf{ctxt} \leftarrow\!\!\$\ \{0,1\}^{\mathsf{mlen}+\lambda}$ and updates $\mathcal{L}_E \leftarrow \mathcal{L}_E \cup \{(\sigma, \mathsf{rcv}, \mathsf{m}||\mathsf{k}, \mathsf{ctxt})\}$.

Let Fail be the event that Game_1 aborts if $(\mathsf{snd}, \mathsf{rcv}, g_1^r, e(X^r, \mathsf{u}_{\mathsf{rcv}}), e(\mathsf{ek}_\sigma, \mathsf{u}_{\mathsf{rcv}}), *) \in \mathcal{L}_{\hat{\mathsf{H}}}$ exists. Game_0 and Game_1 are identical unless Fail occurs. Therefore, we have

$$|\epsilon_1 - \epsilon_0| \leq \Pr[\mathsf{Fail}].$$

To estimate $\Pr[\mathsf{Fail}]$, we show that if $\mathcal{A}$ can triggers Fail, we can construct an adversary $\mathcal{B}_1$ that solves the BDH problem. The construction of $\mathcal{B}_1$ is as follows.

1. Upon receiving $(G = (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, g_1, g_2, e), g_1^\alpha, g_2^\alpha, g_2^\beta, g_1^\gamma)$, $\mathcal{B}_1$ sets $X \coloneqq g_1^\alpha$ (i.e., msk is implicitly set α) and prepares the random oracles H_1, H_2, $\hat{\mathsf{H}}$, and G (i.e., initialize the lists $\mathcal{L}_{\mathsf{H}_1}$, $\mathcal{L}_{\mathsf{H}_2}$, $\mathcal{L}_{\hat{\mathsf{H}}}$, and $\mathcal{L}_{\mathsf{G}}$). Then, $\mathcal{B}_1$ samples $I \leftarrow\!\!\$\ [q_{\hat{\mathsf{H}}}]$ and executes $\mathcal{A}$ on input $\mathsf{mpk} \coloneqq (G, \mathsf{H}_1, \mathsf{H}_2, \hat{\mathsf{H}}, \mathsf{G}, X)$.
2. When $\mathcal{A}$ makes oracle queries, $\mathcal{B}_1$ answers them as follows:
   (a) When $\mathcal{A}$ sends σ to H_1 oracle, $\mathcal{B}_1$ samples $b \leftarrow\!\!\$\ \mathbb{Z}_p$. With probability $1-\delta$, $\mathcal{B}_1$ computes $\mathsf{u}_\sigma = (g_1^\gamma)^b$, and updates $\mathcal{L}_{\mathsf{H}_1} \leftarrow \mathcal{L}_{\mathsf{H}_1} \cup \{(\sigma, \mathsf{u}_\sigma, b, 0)\}$. Otherwise, $\mathcal{B}_1$ computes $\mathsf{u}_\sigma \coloneqq g_1^b$ and updates $\mathcal{L}_{\mathsf{H}_1} \leftarrow \mathcal{L}_{\mathsf{H}_1} \cup \{(\sigma, \mathsf{u}_\sigma, b, 1)\}$. Then, $\mathcal{B}_1$ returns u_σ to $\mathcal{A}$.

[16] Since $\sigma \in \mathcal{L}_S$, the challenger already has computed ek_σ.
[17] Since $\mathsf{rcv} \in \mathcal{L}_R$, the challenger already has computed $\mathsf{dk}_{\mathsf{rcv}}$.

(b) When $\mathcal{A}$ sends ρ to H_2 oracle, $\mathcal{B}_1$ samples $\hat{b} \leftarrow\!\!\$\ \mathbb{Z}_p$. With probability $1-\delta$, $\mathcal{B}_1$ computes $\mathsf{u}_\rho := (g_2^\beta)^{\hat{b}}$, and updates $\mathcal{L}_{\mathsf{H}_2} \leftarrow \mathcal{L}_{\mathsf{H}_2} \cup \{(\rho, \mathsf{u}_\rho, \hat{b}, 0)\}$. Otherwise, $\mathcal{B}_1$ computes $\mathsf{u}_\rho = g_2^{\hat{b}}$ and updates $\mathcal{L}_{\mathsf{H}_2} \leftarrow \mathcal{L}_{\mathsf{H}_2} \cup \{(\rho, \mathsf{u}_\rho, \hat{b}, 1)\}$. Then, $\mathcal{B}_1$ returns u_ρ to $\mathcal{A}$.

(c) When $\mathcal{A}$ sends (σ, ρ, R, U, V) to $\hat{\mathsf{H}}$ oracle, if this is the I-th query to $\hat{\mathsf{H}}$, $\mathcal{B}_1$ checks if both $(\sigma, \mathsf{u}_\sigma, b, d) \in \mathcal{L}_{\mathsf{H}_1}$ and $(\rho, \mathsf{u}_\rho, \hat{b}, \hat{d}) \in \mathcal{L}_{\mathsf{H}_2}$ has coin $d = 0$ and $\hat{d} = 0$. If not, $\mathcal{B}_1$ aborts the game. Otherwise $(d = \hat{d} = 0)$, $\mathcal{B}_1$ outputs $D := V^{\frac{1}{b\hat{b}}}$ as the solution of the BDH problem. If this is not the I-th query, $\mathcal{B}_1$ samples $Z \leftarrow\!\!\$\ \{0,1\}^{\mathsf{mlen}}$ and updates $\mathcal{L}_{\hat{\mathsf{H}}} \leftarrow \mathcal{L}_{\hat{\mathsf{H}}} \cup \{(\sigma, \rho, R, U, V, Z)\}$. $\mathcal{B}_1$ returns Z to $\mathcal{A}$.

(d) When $\mathcal{A}$ sends $(\sigma, \rho, \mathsf{m}, \mathsf{k})$ to G oracle, $\mathcal{B}_1$ samples $r \leftarrow\!\!\$\ \mathbb{Z}_p$ and updates $\mathcal{L}_{\mathsf{G}} \leftarrow \mathcal{L}_{\mathsf{G}} \cup \{(\sigma, \rho, \mathsf{m}, \mathsf{k}, r)\}$. Then, $\mathcal{B}_1$ returns r to $\mathcal{A}$.

(e) When $\mathcal{A}$ sends $(\sigma, \mathsf{rcv}, \mathsf{m})$ to $\mathcal{O}_E$ oracle, it answers as in Game_1.

(f) When $\mathcal{A}$ sends σ to $\mathcal{O}_S$ oracle, $\mathcal{B}_1$ extracts $(\sigma, \mathsf{u}_\sigma, b, d)$ from $\mathcal{L}_{\mathsf{H}_1}$. If $d = 0$, $\mathcal{B}_1$ aborts the game. Otherwise, if $d = 1$, $\mathcal{B}_1$ computes $\mathsf{ek}_\sigma = (g_1^\alpha)^b$ and works as in Game_1.

(g) When $\mathcal{A}$ sends ρ to $\mathcal{O}_R$ oracle, $\mathcal{B}_1$ extracts $(\rho, \mathsf{u}_\rho, \hat{b}, d)$ from $\mathcal{L}_{\mathsf{H}_2}$. If $d = 0$, $\mathcal{B}_1$ aborts the game. Otherwise, if $d = 1$, $\mathcal{B}_1$ computes $\mathsf{dk}_\rho = (g_2^\alpha)^{\hat{b}}$ and works as in Game_1.

Roughly, $\mathcal{B}_1$ guesses the identities and the $\hat{\mathsf{H}}$ query that causes the event Fail, and if $\mathcal{B}_1$ succeeds to guess, it perfectly simulates the $\mathsf{Auth\text{-}oCMA}$ game against $\mathcal{A}$. Let us estimate the probability that $\mathcal{B}_1$ succeeds to guess. The probability Fail occurs at the I-th $\hat{\mathsf{H}}$ query is $\frac{1}{q_{\hat{\mathsf{H}}}}$. The probability $\mathcal{O}_S$ and $\mathcal{O}_R$ do not abort is $\delta^{q_S+q_R}$. The probability the game does not abort when $\mathcal{A}$ sends the I-th $\hat{\mathsf{H}}$ query is $(1-\delta)^2$. Hence, the overall probability that $\mathcal{B}_1$ succeeds to guess is $\frac{1}{q_{\hat{\mathsf{H}}}} \cdot \delta^{q_S+q_R}(1-\delta)^2$. This value is maximum when $\hat{\delta} = 1 - \frac{2}{q_S+q_R+2}$, and thus the probability is at most $\frac{4}{\hat{e}^2(q_S+q_R)^2 q_{\hat{\mathsf{H}}}}$ for large $q_S + q_R$. Moreover, if $\mathcal{B}_1$ succeeds to guess, we know that $\mathsf{u}_\sigma = (g_1^\gamma)^b$ and $\mathsf{u}_\rho = (g_2^\beta)^{\hat{b}}$ if $\sigma \notin \mathcal{L}_S$ and $\rho \notin \mathcal{L}_R$, and thus

$$V = e(\mathsf{u}_\sigma, \mathsf{u}_\rho)^\alpha = e(g_1^{\gamma b}, g_2^{\beta\hat{b}})^\alpha = (e(g_1, g_2)^{\alpha\beta\gamma})^{b\hat{b}}.$$

$\mathcal{B}_1$ can solve the BDH problem correctly when it does not abort. Thus, we have

$$|\epsilon_1 - \epsilon_0| \leq \Pr[\mathsf{Fail}] \leq \frac{\hat{e}^2(q_S+q_R)^2 q_{\hat{\mathsf{H}}}}{4} \cdot \mathsf{Adv}^{\mathsf{bdh}}_{\mathcal{B}_1,\mathcal{G}}(\lambda).$$

Game_2 : In this game, the challenger decrypts ctxt^* with a random $Z^* \leftarrow\!\!\$\ \{0,1\}^{\mathsf{mlen}+\lambda}$ instead of $Z^* := \hat{\mathsf{H}}(\mathsf{snd}^*, \rho^*, R^*, e(R^*, \mathsf{dk}_{\rho^*}), e(\mathsf{H}_1(\mathsf{snd}^*), \mathsf{dk}_{\rho^*}))$.

Let BadQ be the event that $\mathcal{A}$ makes a query $(\sigma^*, \rho^*, \cdot, \cdot, V^*)$ to the oracle $\hat{\mathsf{H}}$ where $V^* := e(\mathsf{u}_{\sigma^*}, \mathsf{u}_{\rho^*})^x$. Since Z^* is now chosen independently from random oracles, $\mathcal{A}$ notices the difference between the two games if BadQ occurs and otherwise the two games proceed identically. Thus, we have

$$|\epsilon_2 - \epsilon_1| \leq \Pr[\mathsf{BadQ}].$$

To estimate $\Pr[\mathsf{BadQ}]$, we show that if $\mathcal{A}$ triggers BadQ, we can construct an adversary $\mathcal{B}_2$ that solves the BDH problem. The construction of $\mathcal{B}_2$ is as follows.

1. Upon receiving $(G = (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, g_1, g_2, e), g_1^\alpha, g_2^\alpha, g_2^\beta, g_1^\gamma)$, $\mathcal{B}_2$ sets $X := g_1^\alpha$ (i.e., msk is implicitly set α) and prepares three random oracles H_1, H_2, $\hat{\mathsf{H}}$, and G (i.e., initialize the lists $\mathcal{L}_{\mathsf{H}_1}$, $\mathcal{L}_{\mathsf{H}_2}$, $\mathcal{L}_{\hat{\mathsf{H}}}$, and $\mathcal{L}_{\mathsf{G}}$). Then, $\mathcal{B}_2$ executes $\mathcal{A}$ on input $\mathsf{mpk} := (G, \mathsf{H}_1, \mathsf{H}_2, \hat{\mathsf{H}}, \mathsf{G}, X)$.
2. When $\mathcal{A}$ makes oracle queries, $\mathcal{B}_2$ answers them as follows:
   (a) When $\mathcal{A}$ sends σ to H_1 oracle, $\mathcal{B}_2$ samples $b \leftarrow\!\!\$\ \mathbb{Z}_p$. With probability $1-\delta$, $\mathcal{B}_2$ computes $\mathsf{u}_\sigma = (g_1^\gamma)^b$ and updates $\mathcal{L}_{\mathsf{H}_1} \leftarrow \mathcal{L}_{\mathsf{H}_1} \cup \{(\sigma, \mathsf{u}_\sigma, b, 0)\}$. Otherwise, $\mathcal{B}_2$ computes $\mathsf{u}_\sigma := g_1^b$ and updates $\mathcal{L}_{\mathsf{H}_1} \leftarrow \mathcal{L}_{\mathsf{H}_1} \cup \{(\sigma, \mathsf{u}_\sigma, b, 1)\}$. Then, $\mathcal{B}_2$ returns u_σ to $\mathcal{A}$.
   (b) When $\mathcal{A}$ sends ρ to H_2 oracle, $\mathcal{B}_2$ samples $\hat{b} \leftarrow\!\!\$\ \mathbb{Z}_p$. With probability $1-\delta$, $\mathcal{B}_2$ computes $\mathsf{u}_\rho := (g_2^\beta)^{\hat{b}}$ and updates $\mathcal{L}_{\mathsf{H}_2} \leftarrow \mathcal{L}_{\mathsf{H}_2} \cup \{(\rho, \mathsf{u}_\rho, \hat{b}, 0)\}$. Otherwise, $\mathcal{B}_2$ computes $\mathsf{u}_\rho = g_2^{\hat{b}}$ and updates $\mathcal{L}_{\mathsf{H}_2} \leftarrow \mathcal{L}_{\mathsf{H}_2} \cup \{(\rho, \mathsf{u}_\rho, \hat{b}, 1)\}$. Then, $\mathcal{B}_2$ returns u_ρ to $\mathcal{A}$.

(c) When $\mathcal{A}$ sends (σ, ρ, R, U, V) to $\hat{\mathsf{H}}$ oracle, $\mathcal{B}_2$ samples $Z \leftarrow\!\!\$\ \{0,1\}^{\mathsf{mlen}}$ and updates $\mathcal{L}_{\hat{\mathsf{H}}} \leftarrow \mathcal{L}_{\hat{\mathsf{H}}} \cup \{(\sigma, \rho, R, U, V, Z)\}$. Then, $\mathcal{B}_2$ returns Z to $\mathcal{A}$.
(d) When $\mathcal{A}$ sends $(\sigma, \rho, \mathsf{m}, \mathsf{k})$ to G oracle, $\mathcal{B}_2$ samples $r \leftarrow\!\!\$\ \mathbb{Z}_p$ and updates $\mathcal{L}_{\mathsf{G}} \leftarrow \mathcal{L}_{\mathsf{G}} \cup \{(\sigma, \rho, \mathsf{m}, \mathsf{k}, r)\}$. Then, $\mathcal{B}_2$ returns r to $\mathcal{A}$.
(e) When $\mathcal{A}$ sends $(\sigma, \mathsf{rcv}, \mathsf{m})$ to $\mathcal{O}_E$ oracle, it answers as in the previous game.
(f) When $\mathcal{A}$ sends σ to $\mathcal{O}_S$ oracle, $\mathcal{B}_2$ extracts $(\sigma, \mathsf{u}_\sigma, b, d)$ from $\mathcal{L}_{\mathsf{H}_1}$. If $d = 0$, $\mathcal{B}_2$ aborts the game. Otherwise (that is, $d = 1$), $\mathcal{B}_2$ computes $\mathsf{ek}_\sigma = (g_1^\alpha)^b$ and returns it to $\mathcal{A}$.
(g) When $\mathcal{A}$ sends ρ to $\mathcal{O}_R$ oracle, $\mathcal{B}_2$ extracts $(\rho, \mathsf{u}_\rho, \hat{b}, d)$ from $\mathcal{L}_{\mathsf{H}_2}$. If $d = 0$, $\mathcal{B}_2$ aborts the game. Otherwise (that is, $d = 1$), $\mathcal{B}_2$ computes $\mathsf{dk}_\rho = (g_2^\alpha)^{\hat{b}}$ and return it to $\mathcal{A}$.

3. $\mathcal{A}$ outputs $(\mathsf{snd}^*, \rho^*, \mathsf{ct}^* := (R^*, \mathsf{ctxt}^*))$. $\mathcal{B}_2$ sets $\sigma^* := \mathsf{snd}^*$. If both $(\sigma^*, \mathsf{u}_{\sigma^*}, b^*, d^*) \in \mathcal{L}_{\mathsf{H}_1}$ and $(\rho^*, \mathsf{u}_{\rho^*}, \hat{b}^*, \hat{d}^*) \in \mathcal{L}_{\mathsf{H}_2}$ do not have coins $d^* = 0$ and $\hat{d}^* = 0$, $\mathcal{B}$ aborts the game. Otherwise, $\mathcal{B}_2$ picks an entry $(\sigma^*, \rho^*, R^*, U, V, \hat{h}) \in \mathcal{L}_{\hat{\mathsf{H}}}$ at random, and outputs $D := V^{\frac{1}{b^*\hat{b}^*}}$ as the solution of the BDH problem.

We can see that $\mathcal{B}_2$ perfectly simulates the Auth-oCMA game if $\mathcal{B}_2$ does not abort. Let us estimate the probability $\Pr[\neg\mathsf{abort}]$. The probability $\mathcal{O}_S$ and $\mathcal{O}_R$ do not abort is $\delta^{q_S+q_R}$. The probability the game does not abort when $\mathcal{A}$ outputs a forgery is $(1-\delta)^2$. Hence, the overall non-aborting probability is $\delta^{q_S+q_R}(1-\delta)^2$. This value is maximum when $\hat{\delta} = \frac{q_S+q_R}{q_S+q_R+2}$, and thus $\Pr[\neg\mathsf{abort}] \leq \frac{4}{\hat{e}^2(q_S+q_R)^2}$ for large $q_S + q_R$. Moreover, we know that $\mathsf{u}_{\sigma^*} = (g_1^\gamma)^{b^*}$, $\mathsf{u}_{\rho^*} = (g_2^\beta)^{\hat{b}^*}$, and thus

$$V^* = e(\mathsf{u}_{\sigma^*}, \mathsf{u}_{\rho^*})^\alpha = e(g_1^{\gamma b^*}, g_2^{\beta\hat{b}^*})^\alpha = (e(g_1, g_2)^{\alpha\beta\gamma})^{b^*\hat{b}^*}.$$

If $\mathcal{A}$ can distinguish the two games, $\mathcal{A}$ has queried $\hat{\mathsf{H}}(\sigma^*, \mathsf{rcv}^*, \cdot, \cdot, V^*)$, and thus $\mathcal{B}_2$ can solve the BDH problem correctly with probability at least $\frac{1}{q_{\hat{\mathsf{H}}}}$. Therefore,

$$|\epsilon_2 - \epsilon_1| \leq \Pr[\mathsf{BadQ}] \leq \frac{\hat{e}^2(q_S+q_R)^2 q_{\hat{\mathsf{H}}}}{4} \cdot \mathsf{Adv}^{\mathsf{bdh}}_{\mathcal{B}_2,\mathcal{G}}(\lambda).$$

Game_3 : In this game, the challenger checks if $\mathsf{G}(\mathsf{m}^*, \mathsf{k}^*, \mathsf{snd}^*, \rho^*)$ has been queried, and if so, it aborts the game. Otherwise, it samples $r^* \leftarrow\!\!\$\ \mathbb{Z}_p$ at random instead of generating it with G. Since $\mathsf{m}^*||\mathsf{k}^*$ is chosen independently at random, the probability $\mathsf{G}(\mathsf{m}^*, \mathsf{k}^*, \mathsf{snd}^*, \rho^*)$ was queried is $\frac{q_\mathsf{G}}{2^{\mathsf{mlen}+\lambda}}$, and thus we have

$$|\epsilon_3 - \epsilon_2| \leq \frac{q_\mathsf{G}}{2^{\mathsf{mlen}+\lambda}}.$$

We finally evaluate ϵ_3. In Game_3, $\mathcal{A}$ breaks the Auth-oCMA security if $R^* = g_1^{r^*}$ holds for randomly chosen $r^* \in \mathbb{Z}_p$. Since for any $R \in \mathbb{G}_1$ the probability that $R^* = g_1^{r^*}$ holds for a randomly chosen $r^* \in \mathbb{Z}_p$ is $\frac{1}{p}$, we have

$$\epsilon_3 = \frac{1}{p}.$$

Putting everything together and folding both adversaries $\mathcal{B}_1$ and $\mathcal{B}_2$ into one adversary $\mathcal{B}$, we obtain

$$\mathsf{Adv}^{\mathsf{auth\text{-}ocma}}_{\mathcal{A},\mathsf{IB\text{-}ME}^{\mathsf{BDH}}}(\lambda) \leq \frac{\hat{e}^2(q_S+q_R)^2 q_{\hat{\mathsf{H}}}}{2} \cdot \mathsf{Adv}^{\mathsf{bdh}}_{\mathcal{B},\mathcal{G}}(\lambda) + \frac{q_\mathsf{G}}{2^{\mathsf{mlen}+\lambda}} + \frac{1}{p}.$$

□

5 IB-ME from IBE and IBS in the Standard Model

In this section, we propose a new generic construction of IB-ME based on IBE, IBS, and reusable extractors. To achieve Priv-MisMatch security, we hide messages with reusable extractors similarly to Francati et al. [18]. We formally show that the resulting scheme, which we call $\mathsf{IB\text{-}ME}^{\mathsf{IBE}+\mathsf{IBS}}$, satisfies Priv-CCA, Priv-MisMatch and Auth-iCMA security in the StdM. Our result shows that HIBE is not necessary for constructing IB-ME in contrast to the conventional approaches [13, 35].

5.1 Construction

To construct an IB-ME scheme with identity space $\mathcal{ID} = \{0,1\}^*$ and message space $\mathcal{M} = \{0,1\}^{\mathsf{mlen}}$, we use the following building blocks.

- An IBE scheme $\mathsf{IBE} = (\mathsf{IBE.Setup}, \mathsf{IBE.KGen}, \mathsf{IBE.Enc}, \mathsf{IBE.Dec})$ with $\mathcal{ID}_{\mathsf{IBE}} = \{0,1\}^*$ and $\mathcal{M}_{\mathsf{IBE}} = \{0,1\}^{\mathsf{mlen}+\mathsf{siglen}}$.
- An IBS scheme $\mathsf{IBS} = (\mathsf{IBS.Setup}, \mathsf{IBS.KGen}, \mathsf{IBS.Sign}, \mathsf{IBS.Ver})$ with $\mathcal{ID}_{\mathsf{IBS}} = \{0,1\}^*$ and siglen bits signatures.
- A reusable computational extractor $\mathsf{Ext} : \mathcal{S} \times \mathcal{ID} \to \{0,1\}^{\mathsf{mlen}+\mathsf{siglen}}$.

The proposed IB-ME scheme $\mathsf{IB\text{-}ME}^{\mathsf{IBE}+\mathsf{IBS}}$ is as follows. The correctness of $\mathsf{IB\text{-}ME}^{\mathsf{IBE}+\mathsf{IBS}}$ immediately follows from the correctness of IBE and IBS.

$\mathsf{Setup}(1^\lambda)$**:** It computes $(\mathsf{mpk}_{\mathsf{IBE}}, \mathsf{msk}_{\mathsf{IBE}}) \leftarrow \mathsf{IBE.Setup}(1^\lambda)$ and $(\mathsf{mpk}_{\mathsf{IBS}}, \mathsf{msk}_{\mathsf{IBS}}) \leftarrow \mathsf{IBS.Setup}(1^\lambda)$, and outputs $\mathsf{mpk} := (\mathsf{mpk}_{\mathsf{IBE}}, \mathsf{mpk}_{\mathsf{IBS}})$ and $\mathsf{msk} := (\mathsf{msk}_{\mathsf{IBE}}, \mathsf{msk}_{\mathsf{IBS}})$.

$\mathsf{SKGen}(\mathsf{mpk}, \mathsf{msk}, \sigma)$**:** It computes $\mathsf{ek}_\sigma \leftarrow \mathsf{IBS.KGen}(\mathsf{mpk}_{\mathsf{IBS}}, \mathsf{msk}_{\mathsf{IBS}}, \sigma)$ and outputs ek_σ.

$\mathsf{RKGen}(\mathsf{mpk}, \mathsf{msk}, \rho)$**:** It computes $\mathsf{dk}_\rho \leftarrow \mathsf{IBE.KGen}(\mathsf{mpk}_{\mathsf{IBE}}, \mathsf{msk}_{\mathsf{IBE}}, \rho)$ and outputs dk_ρ.

$\mathsf{Enc}(\mathsf{mpk}, \mathsf{ek}_\sigma, \mathsf{rcv}, \mathsf{m})$**:** It samples $s \leftarrow\!\!\$\ \mathcal{S}$ and computes $\mathsf{sig} \leftarrow \mathsf{IBS.Sign}(\mathsf{mpk}_{\mathsf{IBS}}, \mathsf{ek}_\sigma, \mathsf{m})$, $Z := \mathsf{Ext}(s, \sigma)$, $\hat{\mathsf{m}} \leftarrow (\mathsf{m}||\mathsf{sig}) \oplus Z$, and $\hat{\mathsf{ct}} \leftarrow \mathsf{IBE.Enc}(\mathsf{mpk}_{\mathsf{IBE}}, \mathsf{rcv}, \hat{\mathsf{m}})$. It outputs $\mathsf{ct} := (\hat{\mathsf{ct}}, s)$.

$\mathsf{Dec}(\mathsf{mpk}, \mathsf{dk}_\rho, \mathsf{snd}, \mathsf{ct} = (\hat{\mathsf{ct}}, s))$**:** It computes $\hat{\mathsf{m}} \leftarrow \mathsf{IBE.Dec}(\mathsf{mpk}_{\mathsf{IBE}}, \mathsf{dk}_\rho, \hat{\mathsf{ct}})$ and $\mathsf{m}||\mathsf{sig} \leftarrow \hat{\mathsf{m}} \oplus \mathsf{Ext}(s, \mathsf{snd})$. Then, it computes $b \leftarrow \mathsf{IBS.Ver}(\mathsf{mpk}_{\mathsf{IBS}}, \mathsf{snd}, \mathsf{m}, \sigma)$. If $b = 1$, it outputs m, else outputs $\perp$.

5.2 Security Proof

We prove $\mathsf{IB\text{-}ME}^{\mathsf{IBE}+\mathsf{IBS}}$ is Priv-CCA, Priv-MisMatch and Auth-iCMA security.

Theorem 5. *If* IBE *is* ANO-IND-ID-CCA *secure,* $\mathsf{IB\text{-}ME}^{\mathsf{IBE}+\mathsf{IBS}}$ *is* Priv-CCA *secure. Formally, if there exists an adversary* $\mathcal{A}$ *that breaks the* Priv-CCA *security of* $\mathsf{IB\text{-}ME}^{\mathsf{IBE}+\mathsf{IBS}}$*, there exists an adversary* $\mathcal{B}$ *that breaks the* ANO-IND-ID-CCA *security of* IBE *such that*

$$\mathsf{Adv}^{\mathsf{priv\text{-}cca}}_{\mathcal{A},\mathsf{IB\text{-}ME}}(\lambda) = \mathsf{Adv}^{\mathsf{ano\text{-}ind\text{-}id\text{-}cca}}_{\mathcal{B},\mathsf{IBE}}(\lambda).$$

The running time of $\mathcal{B}$ *is about that of* $\mathcal{A}$*.*

Proof. Let $\mathsf{CTSamp}(\mathsf{mpk})$ be an algorithm that outputs a pair of a random element in $\mathcal{S}$ and an output of $\mathsf{CTSamp}_{\mathsf{IBE}}(\mathsf{mpk})$, which is a sampling algorithm used for the ANO-IND-ID-CCA security. Let $\mathcal{A}$ be an adversary that breaks the Priv-CCA security of $\mathsf{IB\text{-}ME}^{\mathsf{IBE}+\mathsf{IBS}}$. We show an adversary $\mathcal{B}$ that breaks the ANO-IND-ID-CCA security of IBE by using $\mathcal{A}$. The description of $\mathcal{B}$ is as follows.

1. Upon receiving the master public key $\mathsf{mpk}_{\mathsf{IBE}}$, $\mathcal{B}$ generates $(\mathsf{mpk}_{\mathsf{IBS}}, \mathsf{msk}_{\mathsf{IBS}}) \leftarrow \mathsf{IBS.Setup}(\lambda)$ and executes $\mathcal{A}$ on input $\mathsf{mpk} := (\mathsf{mpk}_{\mathsf{IBE}}, \mathsf{mpk}_{\mathsf{IBS}})$.
2. $\mathcal{B}$ answers queries from $\mathcal{A}$ as follows.
   - When $\mathcal{A}$ sends σ to $\mathcal{O}_S$ oracle, $\mathcal{B}$ computes $\mathsf{ek}_\sigma \leftarrow \mathsf{IBS.KGen}(\mathsf{mpk}_{\mathsf{IBS}}, \mathsf{msk}_{\mathsf{IBS}}, \sigma)$ and returns it to $\mathcal{A}$.
   - When $\mathcal{A}$ sends ρ to $\mathcal{O}_R$ oracle, $\mathcal{B}$ sends ρ to $\mathcal{O}_{SK}$ oracle and receives dk_ρ. Then $\mathcal{B}$ returns it to $\mathcal{A}$.
   - When $\mathcal{A}$ sends $(\mathsf{snd}, \rho, \mathsf{ct} = (\hat{\mathsf{ct}}, s))$ to $\mathcal{O}_D$ oracle, $\mathcal{B}$ sends $(\rho, \hat{\mathsf{ct}})$ to its decryption oracle $\mathcal{O}_D$ and receives $\hat{\mathsf{m}}$. Then, it computes $\mathsf{m}||\mathsf{sig} \leftarrow \hat{\mathsf{m}} \oplus \mathsf{Ext}(s, \mathsf{snd})$ and $b \leftarrow \mathsf{IBS.Ver}(\mathsf{mpk}_{\mathsf{IBS}}, \mathsf{snd}, \mathsf{m}, \sigma)$. If $b = 1$, it returns m; else returns $\perp$.
3. When $\mathcal{A}$ sends $(\sigma^*, \mathsf{rcv}^*, \mathsf{m}^*)$ to request a challenge ciphertext, $\mathcal{B}$ first samples $s^* \leftarrow\!\!\$\ \mathcal{S}$ and computes $\mathsf{sig}^* \leftarrow \mathsf{IBS.Sign}(\mathsf{mpk}_{\mathsf{IBS}}, \mathsf{ek}_{\sigma^*}, \mathsf{m}^*)$, $\hat{\mathsf{m}}^* \leftarrow (\mathsf{m}^*||\mathsf{sig}^*) \oplus \mathsf{Ext}(s^*, \sigma^*)$. Then, it sends $(\mathsf{rcv}^*, \hat{\mathsf{m}}^*)$ to its challenger and receives the challenge ciphertext $\hat{\mathsf{ct}}^*$. Then, $\mathcal{B}$ returns $(\hat{\mathsf{ct}}^*, s^*)$ to $\mathcal{A}$.
4. Finally, when $\mathcal{A}$ outputs $\widehat{\mathsf{coin}}$, $\mathcal{B}$ sends it to the challenger as its guess.

We can verify that $\mathcal{B}$ perfectly simulates the Priv-CCA game against $\mathcal{A}$. Moreover, $\mathsf{rcv}^* \notin \mathcal{L}_{\mathcal{O}_R}$ implies $\mathsf{rcv}^* \notin \mathcal{L}_{SK}$. Therefore, if $\mathcal{A}$ breaks the Priv-CCA security, $\mathcal{B}$ also breaks the ANO-IND-ID-CCA security, that is,

$$\mathsf{Adv}^{\mathsf{priv\text{-}cca}}_{\mathcal{A},\mathsf{IB\text{-}ME}}(\lambda) = \mathsf{Adv}^{\mathsf{ano\text{-}ind\text{-}id\text{-}cca}}_{\mathcal{B},\mathsf{IBE}}(\lambda).$$

□

Theorem 6. *If* Ext *is a* $(\omega(\log(\lambda)), q_E+1)$*-reusable computational extractor,* $\mathsf{IB\text{-}ME}^{\mathsf{IBE}+\mathsf{IBS}}$ *is* Priv-MisMatch *secure. Formally, if there exists an adversary* $\mathcal{A}$ *that breaks the* Priv-MisMatch *security of* $\mathsf{IB\text{-}ME}^{\mathsf{IBE}+\mathsf{IBS}}$*, there exists an adversary* $\mathcal{B}$ *that breaks the security of* Ext *such that*

$$\mathsf{Adv}^{\mathsf{priv\text{-}mismatch}}_{\mathcal{A},\mathsf{IB\text{-}ME}}(\lambda) \leq 2\mathsf{Adv}^{\mathsf{ext}}_{\mathcal{B},\mathsf{Ext}}(\lambda).$$

The running time of $\mathcal{B}$ *is about that of* $\mathcal{A}$*.*

Proof. To prove the theorem, we consider the following sequence of games Game_i for $i \in \{0,1,2\}$. Define the advantage of $\mathcal{A}$ in Game_i as

$$\epsilon_i := \left|\Pr\left[\mathsf{Game}_i^{\mathcal{A}}(\lambda) \Rightarrow 1\right] - \frac{1}{2}\right|.$$

Game_0 : This is the original security game. By definition, we have

$$\epsilon_0 = \mathsf{Adv}^{\mathsf{priv\text{-}mismatch}}_{\mathcal{A},\mathsf{IB\text{-}ME}^{\mathsf{IBE}+\mathsf{IBS}}}(\lambda).$$

Game_1 : In this game, when $\mathcal{A}$ sends $(0, \mathsf{rcv}, \mathsf{m})$ to $\mathcal{O}_{E^*}$ or the challenge ciphertext ct_0 is generated, the challenger uses a random $Z \leftarrow\!\!\$\ \{0,1\}^{\mathsf{mlen}+\mathsf{siglen}}$ instead of $Z := \mathsf{Ext}(s, \sigma_0^*)$. Since σ_0^* is sampled from $\omega(\log(\lambda))$-distribution and $\mathcal{A}$ requests ciphertexts on σ_0^* at most q_E+1 times, $(\omega(\log(\lambda)), q_E+1)$-reusable computational extractor Ext ensures that Game_0 and Game_1 are indistinguishable for $\mathcal{A}$. Formally, we can construct an adversary $\mathcal{B}_1$ such that

$$|\epsilon_1 - \epsilon_0| \leq \mathsf{Adv}^{\mathsf{ext}}_{\mathcal{B}_1,\mathsf{Ext}}(\lambda).$$

Game_2 : In this game, when $\mathcal{A}$ sends $(1, \mathsf{rcv}, \mathsf{m})$ to $\mathcal{O}_{E^*}$ or the challenge ciphertext ct_1 is generated, the challenger uses $Z \leftarrow\!\!\$\ \{0,1\}^{\mathsf{mlen}+\mathsf{siglen}}$ instead of $Z := \mathsf{Ext}(s, \sigma_1^*)$. From the same argument above, we can construct $\mathcal{B}_2$ such that

$$|\epsilon_2 - \epsilon_1| \leq \mathsf{Adv}^{\mathsf{ext}}_{\mathcal{B}_2,\mathsf{Ext}}(\lambda).$$

In Game_2, the ciphertexts $\hat{\mathsf{ct}}$ generated via $\mathcal{O}_{E^*}$ and the challenge ciphertexts ct_0 and ct_1 encrypt a random message. Thus, they do not have information about encrypted messages and the senders σ_0^* and σ_1^*. This means that the challenge bit coin is information-theoretically hidden from $\mathcal{A}$. Therefore, we have

$$\epsilon_2 = 0.$$

Putting everything together and folding $\mathcal{B}_1$ and $\mathcal{B}_2$ into $\mathcal{B}$, we obtain

$$\mathsf{Adv}^{\mathsf{priv\text{-}mismatch}}_{\mathcal{A},\mathsf{IB\text{-}ME}}(\lambda) \leq 2\mathsf{Adv}^{\mathsf{ext}}_{\mathcal{B},\mathsf{Ext}}(\lambda).$$

□

Theorem 7. *If* IBS *is* EUF-ID-CMA *secure,* $\mathsf{IB\text{-}ME}^{\mathsf{IBE}+\mathsf{IBS}}$ *is* Auth-iCMA *secure. Formally, if there exists an adversary* $\mathcal{A}$ *that breaks the* Auth-iCMA *security of* $\mathsf{IB\text{-}ME}^{\mathsf{IBE}+\mathsf{IBS}}$*, there exists an adversary* $\mathcal{B}$ *that breaks the* EUF-ID-CMA *security of* IBS *such that*

$$\mathsf{Adv}^{\mathsf{auth\text{-}icma}}_{\mathcal{A},\mathsf{IB\text{-}ME}}(\lambda) = \mathsf{Adv}^{\mathsf{euf\text{-}id\text{-}cma}}_{\mathcal{B},\mathsf{IBS}}(\lambda).$$

The running time of $\mathcal{B}$ *is about that of* $\mathcal{A}$*.*

Proof. Let $\mathcal{A}$ be an adversary that breaks the Auth-iCMA security of IB-ME$^{\mathsf{IBE+IBS}}$. We show an adversary $\mathcal{B}$ that breaks the EUF-ID-CMA security of IBS by using $\mathcal{A}$. The description of $\mathcal{B}$ is as follows.

1. Upon receiving the master public key $\mathsf{mpk}_{\mathsf{IBS}}$, $\mathcal{B}$ generates $(\mathsf{mpk}_{\mathsf{IBE}}, \mathsf{msk}_{\mathsf{IBE}}) \leftarrow \mathsf{IBE.Setup}(\lambda)$ and executes $\mathcal{A}$ on input $\mathsf{mpk} := (\mathsf{mpk}_{\mathsf{IBE}}, \mathsf{mpk}_{\mathsf{IBS}})$.
2. $\mathcal{B}$ answers queries from $\mathcal{A}$ as follows.
   - When $\mathcal{A}$ sends σ to $\mathcal{O}_S$ oracle, $\mathcal{B}$ sends σ to its key generation oracle $\mathcal{O}_{SK}$ oracle and receives ek_σ. Then $\mathcal{B}$ returns it to $\mathcal{A}$.
   - When $\mathcal{A}$ sends ρ to $\mathcal{O}_R$ oracle, $\mathcal{B}$ computes $\mathsf{dk}_\rho \leftarrow \mathsf{IBE.KGen}(\mathsf{mpk}_{\mathsf{IBE}}, \mathsf{msk}_{\mathsf{IBE}}, \rho)$ and returns it to $\mathcal{A}$.
   - When $\mathcal{A}$ sends $(\sigma, \mathsf{rcv}, \mathsf{m})$ to $\mathcal{O}_E$ oracle, $\mathcal{B}$ first sends (σ, m) to its signing oracle and receives sig. Then, it samples $s \leftarrow\!\!\$\ \mathcal{S}$ and computes $\hat{\mathsf{m}} \leftarrow (\mathsf{m}||\mathsf{sig}) \oplus \mathsf{Ext}(s, \sigma)$ and $\hat{\mathsf{ct}} \leftarrow \mathsf{IBE.Enc}(\mathsf{mpk}_{\mathsf{IBE}}, \mathsf{rcv}, \sigma, \hat{\mathsf{m}})$. It returns $\mathsf{ct} := (\hat{\mathsf{ct}}, s)$ to $\mathcal{A}$.
3. When $\mathcal{A}$ outputs $(\mathsf{snd}^*, \rho^*, \mathsf{ct}^* = (\hat{\mathsf{ct}}^*, s^*))$ as a forgery, $\mathcal{B}$ computes $\hat{\mathsf{m}} \leftarrow \mathsf{IBE.Dec}(\mathsf{mpk}_{\mathsf{IBE}}, \mathsf{dk}_\rho, \mathsf{snd}^*, \hat{\mathsf{ct}}^*)$ and $\mathsf{m}^*||\mathsf{sig}^* \leftarrow \hat{\mathsf{m}} \oplus \mathsf{Ext}(s^*, \mathsf{snd}^*)$ and outputs $(\mathsf{m}^*, \mathsf{sig}^*)$ as its forgery.

We can verify that $\mathcal{B}$ perfectly simulates the Auth-iCMA game. If $\mathcal{A}$ makes a valid forgery, we have $\mathsf{snd}^* \notin \mathcal{L}_S$, $(\mathsf{snd}^*, \rho^*, \mathsf{m}^*) \notin \mathcal{L}_E$, and $\mathsf{IBS.Ver}(\mathsf{mpk}_{\mathsf{IBS}}, \mathsf{snd}^*, \mathsf{m}^*, \sigma^*) = 1$. Due to the construction of IB-ME$^{\mathsf{IBE+IBS}}$, $\mathsf{IBS.Ver}(\mathsf{mpk}_{\mathsf{IBS}}, \mathsf{snd}^*, \mathsf{m}^*, \sigma^*) =$ implies $\mathsf{m}^* \neq \perp$. Also, $\mathsf{snd}^* \notin \mathcal{L}_S$ implies $\mathsf{snd}^* \notin \mathcal{L}_{SK}$, and $(\mathsf{snd}^*, \rho^*, \mathsf{m}^*) \notin \mathcal{L}_E$ implies $(\mathsf{snd}^*, \mathsf{m}^*) \notin \mathcal{L}_{SIG}$. Therefore, if $\mathcal{A}$ breaks the Auth-iCMA security, $\mathcal{B}$ also breaks the EUF-ID-CMA security. Thus, we have

$$\mathsf{Adv}^{\mathsf{auth\text{-}icma}}_{\mathcal{A},\mathsf{IB\text{-}ME}}(\lambda) = \mathsf{Adv}^{\mathsf{euf\text{-}id\text{-}cma}}_{\mathcal{B},\mathsf{IBS}}(\lambda).$$

□

6 Comparison

In this section, we compare our IB-ME schemes, IB-ME$^{\mathsf{BDH}}$ and IB-ME$^{\mathsf{IBE+IBS}}$, with the existing IB-ME schemes by Ateniese et al. [2], Chen et al. [11], and Wang et al. [35], which are based on standard assumptions [18]. Their security and the size of secret keys and ciphertexts are summarized in Tables 2 and 3.

IB-ME from the BDH assumption in the ROM. We compare IB-ME$^{\mathsf{BDH}}$, IB-ME$^{\mathsf{IBE+IBS}}$ and Ateniese et al. scheme [2]. We instantiate IB-ME$^{\mathsf{IBE+IBS}}$ with the Bobeh-Franklin IBE scheme and the Cha-Cheon IBS scheme [10].[19] Table 2a summaries their security property and space complexity. IB-ME$^{\mathsf{BDH}}$ is the best scheme in terms of key and ciphertext sizes since they contain only one group elements. Moreover, it achieves stronger Priv-CCA and Auth-oCMA security. IB-ME$^{\mathsf{IBE+IBS}}$ has about two times larger ciphertext than IB-ME$^{\mathsf{BDH}}$, but it achieves Auth-iCMA security (i.e., secure even if the receiver's key is compromised), which is stronger than Auth-oCMA. Thus, IB-ME$^{\mathsf{BDH}}$ and IB-ME$^{\mathsf{IBE+IBS}}$ offer a trade-off between the security level (i.e., outsider security vs. insider security) and efficiency.

IB-ME from the SXDH assumption in the StdM. We compare IB-ME$^{\mathsf{IBE+IBS}}$, Chen et al. scheme [11], and Wang et al. scheme [35]. To instantiate IB-ME$^{\mathsf{IBE+IBS}}$ from the SXDH assumption in the StdM, we use the CCA-secure anonymous IBE scheme [22] (and the IBS scheme [31] based on CDH assumption). Table 2b summaries the comparison results. Among them, our scheme achieves the smallest secret key size while achieving stronger Priv-CCA and Auth-iCMA security. In particular, the size of the decryption key is 4 or more times shorter than the existing schemes. Regarding the ciphertext size, ours is smaller than Wang et al. scheme but slightly larger than Chen et al. scheme. This difference between ours and Chen et al. scheme can be interpreted as the cost our scheme pays for stronger security. We conclude that our scheme achieves stronger security with reasonable space complexity.

[18] We do not consider Francati et al. scheme [18] here since its security relies on a non-standard q-type assumption.
[19] RO is used as a reusable extractor.

Table 2: Comparison of the IB-ME schemes based on bilinear groups. The column "Ciphertext" indicates the difference between the length of ciphertext and that of plaintext. $|\mathbb{G}_1|$, $|\mathbb{G}_2|$ and $|\mathbb{G}_T|$ denotes the size of group element in $\mathbb{G}_1$, $\mathbb{G}_2$ and $\mathbb{G}_T$, respectively.

(a) IB-ME schemes from the BDH assumption in the ROM.

Schemes	Security			Space complexity		
	Priv	Auth	Mismatch	Enc. key	Dec. key	Ciphertext
Ateniese et al. [2]	CPA	oNMA		$\lvert\mathbb{G}_1\rvert$	$3\lvert\mathbb{G}_2\rvert$	$2\lvert\mathbb{G}_1\rvert + \lambda$
IB-ME$^{\mathsf{BDH}}$(§ 4.1)	CCA	oCMA	$\surd$	$\lvert\mathbb{G}_1\rvert$	$\lvert\mathbb{G}_2\rvert$	$\lvert\mathbb{G}_1\rvert + \lambda$
IB-ME$^{\mathsf{IBE+IBS}}$(§ 5.1) (IBE [5]+IBS [10])	CCA	iCMA	$\surd$	$\lvert\mathbb{G}_1\rvert$	$\lvert\mathbb{G}_2\rvert$	$3\lvert\mathbb{G}_1\rvert + \lambda$

(b) IB-ME schemes from the SXDH assumption in the StdM.

Schemes	Security			Space complexity		
	Priv	Auth	Mismatch	Enc. key	Dec. key	Ciphertext
Chen et al. [11]	CPA	iNMA		$8\lvert\mathbb{G}_1\rvert$	$16\lvert\mathbb{G}_2\rvert + \lvert\mathbb{G}_T\rvert$	$8\lvert\mathbb{G}_1\rvert$
Wang et al. [35] (HIBE [26]+IBS [31])	CPA	iCMA		$2\lvert\mathbb{G}_1\rvert$	$52\lvert\mathbb{G}_2\rvert$	$13\lvert\mathbb{G}_1\rvert$
IB-ME$^{\mathsf{IBE+IBS}}$(§ 5.1) (IBE [22]+IBS [31])	CCA	iCMA	$\surd$	$2\lvert\mathbb{G}_1\rvert$	$4\lvert\mathbb{G}_2\rvert$	$10\lvert\mathbb{G}_1\rvert + \lambda$

Table 3: Comparison of IB-ME schemes from lattices in the QROM. The data sizes are provided in bytes. The column "Ciphertext" indicates the difference between the length of ciphertext and that of plaintext. Note that LATTE-1 and LATTE-3 offer different security levels, 128-bit and 80-bit respectively.

Schemes	Security			Space complexity		
	Priv	Auth	Mismatch	Enc. key	Dec. key	Ciphertext
Wang et al. [35] (LATTE-3 [38]+Falcon-IBS†)	CPA	iCMA		1579	92160	29941
IB-ME$^{\mathsf{IBE+IBS}}$(§ 5.1) (LATTE-1 [38]+Falcon-IBS†)	CCA	iCMA	$\surd$	1579	3072	8533

†: IBE scheme derived from Falcon [32] via the signature-to-IBS conversion [25].

IB-ME from lattices in the QROM. We finally compare post-quantum lattice-based IB-ME schemes in the QROM derived from our IB-ME$^{\mathsf{IBE+IBS}}$ and the Wang et al. scheme [35]. We instantiate them with a lattice-based anonymous (H)IBE scheme LATTE [38] and a lattice-based IBS scheme derived from Falcon [32] via the signature-to-IBS conversion [25]. Table 3 summarizes their security and space complexity. Our scheme is significantly space efficient than Wang et al. scheme. Especially, our decryption key and ciphertext are respectively 30 and 3.5 times shorter than that of Wang et al. scheme, and all of them are less than 10 Kilobytes. This is due to the fact that our scheme is simply based on IBE, not HIBE. Moreover, our scheme achives stronger security. Our result makes post-quantum IB-ME schemes practical.

Feasibility results by IB-ME$^{\mathsf{IBE+IBS}}$. It is worth noting that IB-ME$^{\mathsf{IBE+IBS}}$ provides IB-ME schemes that have not been realized to date. We obtain the first pairing-free IB-ME scheme in the StdM from a pairing-free anonymous IBE scheme [8][11] and IBS scheme [25]. We also obtain the first tightly-secure IB-ME scheme from lattices in the QROM from a tightly-secure lattice-based anonymous IBE scheme [24] and IBS scheme [16].

Acknowledgements This research was in part conducted under a contract of "Research and development on new generation cryptography for secure wireless communication services" among "Research and Development

for Expansion of Radio Wave Resources (JPJ000254)," which was supported by the Ministry of Internal Affairs and Communications, Japan. Keitaro Hashimoto and Keisuke Hara were partially supported by JST CREST JPMJCR22M1, Japan. Also, Keisuke Hara was partially supported by JST-AIP JPMJCR22U5, Japan. Junji Shikata was partially supported by JSPS KAKENHI Grant Numbers JP22H03590, Japan.

References

1. Agrawal, S., Boneh, D., Boyen, X.: Efficient lattice (H)IBE in the standard model. In: Gilbert, H. (ed.) EUROCRYPT 2010. LNCS, vol. 6110, pp. 553–572. Springer, Heidelberg (May / Jun 2010). https://doi.org/10.1007/978-3-642-13190-5_28 6
2. Ateniese, G., Francati, D., Nuñez, D., Venturi, D.: Match me if you can: Matchmaking encryption and its applications. In: Boldyreva, A., Micciancio, D. (eds.) CRYPTO 2019, Part II. LNCS, vol. 11693, pp. 701–731. Springer, Heidelberg (Aug 2019). https://doi.org/10.1007/978-3-030-26951-7_24 3, 4, 6, 10, 13, 25, 26
3. Balfanz, D., Durfee, G., Shankar, N., Smetters, D., Staddon, J., Wong, H.C.: Secret handshakes from pairing-based key agreements. In: Proceedings of 2003 Symposium on Security and Privacy. pp. 180–196 (May 2003). https://doi.org/10.1109/SECPRI.2003.1199336 3
4. Boneh, D., Boyen, X.: Efficient selective identity-based encryption without random oracles. Journal of Cryptology **24**(4), 659–693 (Oct 2011). https://doi.org/10.1007/s00145-010-9078-6 7
5. Boneh, D., Franklin, M.K.: Identity-based encryption from the Weil pairing. In: Kilian, J. (ed.) CRYPTO 2001. LNCS, vol. 2139, pp. 213–229. Springer, Heidelberg (Aug 2001). https://doi.org/10.1007/3-540-44647-8_13 3, 4, 5, 6, 13, 26
6. Boyen, X.: Reusable cryptographic fuzzy extractors. In: Atluri, V., Pfitzmann, B., McDaniel, P. (eds.) ACM CCS 2004. pp. 82–91. ACM Press (Oct 2004). https://doi.org/10.1145/1030083.1030096 6, 9
7. Boyen, X., Mei, Q., Waters, B.: Direct chosen ciphertext security from identity-based techniques. In: Atluri, V., Meadows, C., Juels, A. (eds.) ACM CCS 2005. pp. 320–329. ACM Press (Nov 2005). https://doi.org/10.1145/1102120.1102162 7
8. Brakerski, Z., Lombardi, A., Segev, G., Vaikuntanathan, V.: Anonymous IBE, leakage resilience and circular security from new assumptions. In: Nielsen, J.B., Rijmen, V. (eds.) EUROCRYPT 2018, Part I. LNCS, vol. 10820, pp. 535–564. Springer, Heidelberg (Apr / May 2018). https://doi.org/10.1007/978-3-319-78381-9_20 6, 26
9. Canetti, R.: Towards realizing random oracles: Hash functions that hide all partial information. In: Kaliski Jr., B.S. (ed.) CRYPTO'97. LNCS, vol. 1294, pp. 455–469. Springer, Heidelberg (Aug 1997). https://doi.org/10.1007/BFb0052255 6, 9
10. Cha, J.C., Cheon, J.H.: An identity-based signature from gap Diffie-Hellman groups. In: Desmedt, Y. (ed.) PKC 2003. LNCS, vol. 2567, pp. 18–30. Springer, Heidelberg (Jan 2003). https://doi.org/10.1007/3-540-36288-6_2 25, 26
11. Chen, J., Li, Y., Wen, J., Weng, J.: Identity-based matchmaking encryption from standard assumptions. In: Agrawal, S., Lin, D. (eds.) ASIACRYPT 2022, Part III. LNCS, vol. 13793, pp. 394–422. Springer, Heidelberg (Dec 2022). https://doi.org/10.1007/978-3-031-22969-5_14 3, 4, 10, 25, 26
12. Chen, J., Lim, H.W., Ling, S., Wang, H., Wee, H.: Shorter identity-based encryption via asymmetric pairings. Designs, Codes and Cryptography **73**(3), 911–947 (Dec 2014). https://doi.org/10.1007/s10623-013-9834-3 4
13. Chiku, S., Hara, K., Shikata, J.: Hierarchical identity-based matchmaking encryption. In: IEICE Technical Report. vol. 123, pp. 60–67. The Institute of Electronics, Information and Communication Engineers (July 2023) 3, 4, 5, 11, 22
14. Dodis, Y., Kalai, Y.T., Lovett, S.: On cryptography with auxiliary input. In: Mitzenmacher, M. (ed.) 41st ACM STOC. pp. 621–630. ACM Press (May / Jun 2009). https://doi.org/10.1145/1536414.1536498 6, 9
15. Ducas, L., Lyubashevsky, V., Prest, T.: Efficient identity-based encryption over NTRU lattices. In: Sarkar, P., Iwata, T. (eds.) ASIACRYPT 2014, Part II. LNCS, vol. 8874, pp. 22–41. Springer, Heidelberg (Dec 2014). https://doi.org/10.1007/978-3-662-45608-8_2 6
16. Foo, E., Li, Q.: Tightly secure lattice identity-based signature in the quantum random oracle model. In: Simpson, L., Rezazadeh Baee, M.A. (eds.) ACISP 2023. LNCS, vol. 13915, pp. 381–402. Springer, Heidelberg (July 2023). https://doi.org/10.1007/978-3-031-35486-1_17 6, 26
17. Francati, D., Friolo, D., Malavolta, G., Venturi, D.: Multi-key and multi-input predicate encryption from learning with errors. Cryptology ePrint Archive, Report 2022/806 (2022), https://eprint.iacr.org/2022/806 6

18. Francati, D., Guidi, A., Russo, L., Venturi, D.: Identity-based matchmaking encryption without random oracles. In: Adhikari, A., Küsters, R., Preneel, B. (eds.) INDOCRYPT 2021. LNCS, vol. 13143, pp. 415–435. Springer, Heidelberg (December 2021). https://doi.org/10.1007/978-3-030-92518-5_19 3, 4, 5, 6, 10, 11, 22, 25
19. Fujisaki, E., Okamoto, T.: How to enhance the security of public-key encryption at minimum cost. In: Imai, H., Zheng, Y. (eds.) PKC'99. LNCS, vol. 1560, pp. 53–68. Springer, Heidelberg (Mar 1999). https://doi.org/10.1007/3-540-49162-7_5 5, 13, 16
20. Galindo, D.: Boneh-Franklin identity based encryption revisited. In: Caires, L., Italiano, G.F., Monteiro, L., Palamidessi, C., Yung, M. (eds.) ICALP 2005. LNCS, vol. 3580, pp. 791–802. Springer, Heidelberg (Jul 2005). https://doi.org/10.1007/11523468_64 5, 13, 16
21. Gentry, C.: Practical identity-based encryption without random oracles. In: Vaudenay, S. (ed.) EUROCRYPT 2006. LNCS, vol. 4004, pp. 445–464. Springer, Heidelberg (May / Jun 2006). https://doi.org/10.1007/11761679_27 4, 6
22. Hofheinz, D., Jia, D., Pan, J.: Identity-based encryption tightly secure under chosen-ciphertext attacks. In: Peyrin, T., Galbraith, S. (eds.) ASIACRYPT 2018, Part II. LNCS, vol. 11273, pp. 190–220. Springer, Heidelberg (Dec 2018). https://doi.org/10.1007/978-3-030-03329-3_7 6, 25, 26
23. Jain, A., Jin, Z.: Non-interactive zero knowledge from sub-exponential DDH. In: Canteaut, A., Standaert, F.X. (eds.) EUROCRYPT 2021, Part I. LNCS, vol. 12696, pp. 3–32. Springer, Heidelberg (Oct 2021). https://doi.org/10.1007/978-3-030-77870-5_1 6
24. Katsumata, S., Yamada, S., Yamakawa, T.: Tighter security proofs for GPV-IBE in the quantum random oracle model. In: Peyrin, T., Galbraith, S. (eds.) ASIACRYPT 2018, Part II. LNCS, vol. 11273, pp. 253–282. Springer, Heidelberg (Dec 2018). https://doi.org/10.1007/978-3-030-03329-3_9 6, 8, 26
25. Kiltz, E., Neven, G.: Identity-based signatures. In: Joye, M., Neven, G. (eds.) Identity-Based Cryptography, Cryptology and Information Security Series, vol. 2, pp. 31–44. IOS Press (2009). https://doi.org/10.3233/978-1-58603-947-9-31 6, 9, 26
26. Langrehr, R., Pan, J.: Tightly secure hierarchical identity-based encryption. In: Lin, D., Sako, K. (eds.) PKC 2019, Part I. LNCS, vol. 11442, pp. 436–465. Springer, Heidelberg (Apr 2019). https://doi.org/10.1007/978-3-030-17253-4_15 26
27. Malone-Lee, J.: Identity-based signcryption. Cryptology ePrint Archive, Report 2002/098 (2002), https://eprint.iacr.org/2002/098 6
28. Matsuda, T., Matsuura, K., Schuldt, J.C.N.: Efficient constructions of signcryption schemes and signcryption composability. In: Roy, B.K., Sendrier, N. (eds.) INDOCRYPT 2009. LNCS, vol. 5922, pp. 321–342. Springer, Heidelberg (Dec 2009) 5, 10
29. Naor, M., Yung, M.: Public-key cryptosystems provably secure against chosen ciphertext attacks. In: 22nd ACM STOC. pp. 427–437. ACM Press (May 1990). https://doi.org/10.1145/100216.100273 6
30. Nielsen, J.B.: Non-committing encryption is too easy in the random oracle model. BRICS Report Series **8**(47) (Dec 2001). https://doi.org/10.7146/brics.v8i47.21707 19
31. Paterson, K.G., Schuldt, J.C.N.: Efficient identity-based signatures secure in the standard model. In: Batten, L.M., Safavi-Naini, R. (eds.) ACISP 06. LNCS, vol. 4058, pp. 207–222. Springer, Heidelberg (Jul 2006) 25, 26
32. Prest, T., Fouque, P.A., Hoffstein, J., Kirchner, P., Lyubashevsky, V., Pornin, T., Ricosset, T., Seiler, G., Whyte, W., Zhang, Z.: FALCON. Tech. rep., National Institute of Standards and Technology (2022), available at https://csrc.nist.gov/Projects/post-quantum-cryptography/selected-algorithms-2022 26
33. Sakai, R., Ohgishi, K., Kasahara, M.: Cryptosystems based on pairing. In: The 2000 Symposium on Cryptography and Information Security (January 2000) 4, 5, 13
34. Shamir, A.: Identity-based cryptosystems and signature schemes. In: Blakley, G.R., Chaum, D. (eds.) CRYPTO'84. LNCS, vol. 196, pp. 47–53. Springer, Heidelberg (Aug 1984) 6
35. Wang, Y., Wang, B., Lai, Q., Zhan, Y.: Identity-based matchmaking encryption with stronger security and instantiation on lattices. Cryptology ePrint Archive, Report 2022/1718 (2022), https://eprint.iacr.org/2022/1718 3, 4, 11, 22, 25, 26
36. Waters, B.: Dual system encryption: Realizing fully secure IBE and HIBE under simple assumptions. In: Halevi, S. (ed.) CRYPTO 2009. LNCS, vol. 5677, pp. 619–636. Springer, Heidelberg (Aug 2009). https://doi.org/10.1007/978-3-642-03356-8_36 6
37. Waters, B.R.: Efficient identity-based encryption without random oracles. In: Cramer, R. (ed.) EUROCRYPT 2005. LNCS, vol. 3494, pp. 114–127. Springer, Heidelberg (May 2005). https://doi.org/10.1007/11426639_7 6
38. Zhao, R.K., McCarthy, S., Steinfeld, R., Sakzad, A., O'Neill, M.: Quantum-safe HIBE: does it cost a latte? Cryptology ePrint Archive, Report 2021/222 (2021), https://eprint.iacr.org/2021/222 26

39. Zheng, Y.: Digital signcryption or how to achieve cost(signature & encryption) ≪ cost(signature) + cost(encryption). In: Kaliski Jr., B.S. (ed.) CRYPTO'97. LNCS, vol. 1294, pp. 165–179. Springer, Heidelberg (Aug 1997). https://doi.org/10.1007/BFb0052234 6