Generic Methods to Achieve Tighter Security Reductions for a Category of IBE Schemes

Yu Chen[1] and Liqun Chen[2] and Zhong Chen[1]

[1] Information Security Lab, School of EECS, Peking University, Beijing, China
{chenyu,chen}@infosec.pku.edu.cn
[2] Hewlett-Packard Laboratories, Bristol, United Kingdom.
liqun.chen@hp.com

Abstract. We show that Katz-Wang's duplicating key and ciphertext technique can be extended to a generic method that can be used in a certain category of Identity-Based Encryption (IBE) schemes for the purposes of improving their security reductions. We further develop two refined approaches by adapting the randomness reuse technique in the Katz-Wang technique: one is public key duplication, and the other is master key duplication. Compared to the Katz-Wang technique, our two refined approaches do not only improve the performances of the resulting IBE schemes but also enable a reduction algorithm to deal with decryption queries correctly and therefore can achieve chosen ciphertext security. As case studies, we apply these two approaches to modify the Boneh-Franklin IBE scheme and the Boneh-Boyen IBE scheme, respectively. Both of the modifications improve the tightness of security reductions, compared to the original schemes, with a reasonably low cost.

Key words: identity based encryption, provable security, tight reduction, generic method

1 Introduction

It is well known that a tight security reduction is crucial to cryptographic schemes, not only from theoretical aspects, but also from practical aspects [2, 13].

A common method of reducing the security of an IBE scheme $\mathcal{E}$ to the hardness of some underlying problem $\mathcal{P}$ is following the partitioning strategy [13, 19]. In the random oracle model [3, 11], it is done by modeling the identity hash function $H(\cdot)$ as a random oracle, then creating a reduction algorithm $\mathcal{B}$ who utilizes the programmability of $H(\cdot)$ to partition the identity space V into two orthogonal subspaces V_1 and V_2, (1) V_1 - identities for which $\mathcal{B}$ can create private keys with some trapdoor information; and (2) V_2 - identities for which $\mathcal{B}$ can generate the challenge ciphertext embedded with the instance of $\mathcal{P}$. The reduction algorithm $\mathcal{B}$ expects that the identities of private key queries comes from V_1 and the challenge (target) identity I^* comes from V_2. The two subspaces are always orthogonal, i.e. $V = V_1 \cup V_2$ and $V_1 \cap V_2 = \emptyset$.

During the reduction, $\mathcal{B}$ may abort for two reasons: (1) $\mathcal{B}$ can not generate the private keys for some identity I_i ($I_i \notin V_1$) when answering the private key queries; (2) $\mathcal{B}$ can not embed the challenge instance of $\mathcal{P}$ into the challenge ciphertext under I^* ($I^* \notin V_2$). Let p_1 be the probability that $\mathcal{B}$ does not abort due to Reason 1, p_2 be the probability that $\mathcal{B}$ does not abort due to Reason 2. Thereby the probability that $\mathcal{B}$ does not abort throughout the simulation is $\Pr[\overline{\texttt{abort}}] = p_1 p_2$. For example, denote the maximum number of private key queries and random oracle queries by Q_e and Q_h, for the Boneh-Franklin IBE scheme (BF-IBE for short) [6] that is $\Pr[\overline{\texttt{abort}}] = (1-\delta)^{Q_e} \cdot \delta$ for some $0 < \delta < 1$; for the Sakai-Kasahara IBE scheme (SK-IBE for short) [10, 17] that is $\Pr[\overline{\texttt{abort}}] = (1 - Q_e/Q_h) \cdot 1/(Q_h - Q_e)$. We stress that the overall looseness of security reduction comes from two aspects, one is $\Pr[\overline{\texttt{abort}}]$, while the other is the probability p_3 that $\mathcal{B}$ works out the right solution of $\mathcal{P}$ based on the outputs of the adversary or the associated random oracle queries logs. Note that the notation introduced in this section will be used throughout the paper.

Two Types of Reduction Technique. In the realm of the random oracle model, the security reduction technique of for IBE schemes can be classified into two types. The first type is that the reduction algorithm outputs its solution to $\mathcal{P}$ based on the adversary's output in $\{0, 1\}$. Most of the schemes based on some decisional number-theoretic problem $\mathcal{P}$ belong to this type. Informally speaking, with the goal to determine whether the input value T is the right solution of $\mathcal{P}$, the reduction algorithm embeds the challenge instance of $\mathcal{P}$ into the challenge ciphertext C^* of M_β. If the value T is the right solution, then the adversary has the advantage ϵ to guess the right bit β. Otherwise, the adversary's advantage is negligible since M_β is information-theoretically hidden from the adversary. The second type is that the reduction algorithm outputs its solution to $\mathcal{P}$ based on the lists used in the simulation for random oracles, i.e., extracting the desired answer from the entries in the associated random oracle query lists. Interestingly, Type-1 proofs can be easily transformed to Type-2 proofs by wrapping the wanted term in a hash function which will be treated as a random oracle. When the simulation is finished, the reduction algorithm can determine whether T is the right solution of $\mathcal{P}$ by checking if T appears with some form in the corresponding random query list. For this reason, we assume that all the IBE schemes proven secure in the random oracle model follow the Type-2 style proof.

1.1 Related Work and Motivation

Katz and Wang [15] proposed a FDH-/PFDH-like signature scheme achieving a tight security reduction without using a random salt. They also pointed out that their technique can be extended to allow a tighter security proof for Boneh-Franklin IBE scheme (BF-IBE for short) [6]. In their modified BF-IBE, for any ID there are two "public keys" $PK_{\mathsf{ID},0} = H(\mathsf{ID}||0)$ and $PK_{\mathsf{ID},1} = H(\mathsf{ID}||1)$ (for hash function H modeled as a random oracle); to encrypt a message M under identity ID, a sender now encrypts the same message with respect to both of the two public keys, the resulting cipheretext is $C = \langle C_0, C_1 \rangle$, where C_0 is the encrypted with $PK_{\mathsf{ID},0}$ and C_1 is the encrypted with $PK_{\mathsf{ID},1}$. The PKG (Private Key Generator), however, only gives ID one of the corresponding private keys (either $SK_{\mathsf{ID},0}$ or $SK_{\mathsf{ID},1}$ but not both). Note that a single private key is sufficient to enable correct decryption. A simulation can be bulit in which through the control of the random oracle H the reduction algorithm knows exactly one private key for every ID. This allows all the private key extraction queries can be answered by the reduction algorithm, while ensuring that encryption to any non exposed ID remains secret. The successful adversary partially decrypts the challenge ciphertext with the "wrong" private key with probability $1/2$, thus giving the reduction algorithm useful information. A disadvantage of their modified scheme is its cost: the ciphertext of their modified scheme is twice as much as that of BF-IBE, and the efficiency of encryption is reduced by a factor of two. Besides, Katz and Wang did not explain how to achieve chosen ciphertext security with their scheme.

Attrapadung et al. [1] enhanced Katz-Wang's idea with some sophisticated techniques to propose a scheme named $\mathsf{TightIBE}$ based on BF-IBE, which has a tight reduction of chosen ciphertext security. However, they did not discuss how their idea can be used in other IBE schemes. They also concluded that combining the Katz-Wang technique and the Fujisaki-Okamoto transformation [12] straightforwardly cannot provide achieve chosen ciphertext security and tighter reduction for BF-IBE simultaneously. To see this, consider the following attack: when the adversary gets a challenge ciphertext $C^* = \langle C_0, C_1 \rangle$ (of the message M_β that it is asked to distinguish), it picks a random message $M' \in \{0, 1\}^n$ and creates two ciphertexts $\langle C'_0, C_1 \rangle$ and $\langle C_0, C'_1 \rangle$, where C'_0 is the encryption of M' under $PK_{\mathsf{ID},0}$ and C'_1 is the encryption of M' under $PK_{\mathsf{ID},1}$. Then by querying the decryption oracle with $\langle C'_0, C_1 \rangle$ and $\langle C_0, C'_1 \rangle$, it would learn the message M_β with probability 1, since one of C_0 and C_1 would be decrypted when the challenger answers the decryption queries $\langle C'_0, C_1 \rangle$ and $\langle C_0, C'_1 \rangle$. So the scheme directly derived from the Katz-Wang technique and Fujisaki-Okamoto transformation is not immune to chosen ciphertext attack. The underlying reason is that the two parts of the ciphertext are mutual independent, thus the ill-formed decryption queries $\langle C_0, C_1 \rangle$ (where C_0 and C_1 are the ciphertext under same identity but of different messages) would not be detected and get rejected. Clearly, to achieve $\mathsf{IND\text{-}ID\text{-}CCA}$ security, it is necessary that the equality of the underlying messages in the two parts of the ciphertext could be tested.

As far as we know, the application of Katz-Wang's technique in IBE is confined to only BF-IBE [6], and the costs to achieve tighter security reduction is a bit expensive. Thus it is natural to ask if the Katz-Wang technique can be extended to other IBE schemes? What kind of IBE scheme can benefit from it? Can the Katz-Wang technique be improved? Does there exist an approach to make the Katz-Wang technique and the Fujisaki-Okamoto transformation work together to provide a tighter reduction of chosen ciphertext security?

1.2 Our Contributions

Our first contribution is showing that the Katz-Wang technique can be extended to a generic method of improving the tightness of security reductions (minimize $\Pr[\mathtt{abort}]$) for a category of IBE schemes, which satisfy the following conditions:

- An IBE scheme $\mathcal{E}$ is provably secure in the random oracle model, and in its security reduction, for any identity the reduction algorithm can generate the corresponding private key with probability 1/2. In other words, the reduction algorithm can partition the whole identity space in some way to make $|V_1| = |V_2|$. That will maximize the probability p_3 that the reduction algorithm can solve the underlying problem $\mathcal{P}$, as well as ensure the responses to the private key queries are indistinguishable from the adversary's view.

Then we can double $\mathcal{E}$ to obtain $\mathcal{E}^2$ using the Katz-Wang technique. In Section 3 we prove that the reduction for $\mathcal{E}^2$ is $1/(2\Pr[\mathtt{abort}])$ times tighter compared to the reduction for the original scheme $\mathcal{E}$.

Among pairing-based IBE schemes [7], we observe that in the random oracle model the full domain hash IBE family [6] and the commutative blinding IBE family [4] meet the above conditions. Thus the IBE schemes from these two families can benefit from our generic method. However, directly using the Katz-Wang technique for the transformation has two drawbacks. First, the resulting scheme $\mathcal{E}^2$ only has chosen plaintext security. Second, the ciphertext size and the computation cost of $\mathcal{E}^2$ are twice as much as them the original scheme $\mathcal{E}$.

Be aware that in the above transformation, one message is encrypted twice under two public keys using two independent randomnesses r_0 and r_1. Thus in the ciphertext $C = \langle C_0, C_1 \rangle$ of $\mathcal{E}^2$, two components C_0 and C_1 are mutual independent. This happens to be the reason that directly combining Katz-Wang's technique with any existing CPA-to-CCA transformation, e.g., the Fujisaki-Okamoto transformation, cannot lead to CCA security. We observe that the randomness r_0 and r_1 are not necessarily to be independent in $\mathcal{E}^2$. More surprisingly, the randomness reuse when doubling the encryption will not only enable us to shrink the ciphertext size, reduce the computation cost, but also can bind C_0 and C_1 together. Therefore adapting the randomness reuse technique with such a CPA-to-CCA transformation (in the paper we use the Fujisaki-Okamoto transformation as an example) to $\mathcal{E}$, the resulting scheme $\mathcal{E}^2_{hy}$ would be CCA secure with a tighter security reduction. Intuitively, the randomness reuse enables the decryption oracle to delect/reject the ill-formed ciphertexts.

To further explain our ideas we recall the encryption algorithm of an IBE scheme as follows: (1) choose a randomness r and then encapsulate it using algorithm Encaps, the results consist of a value $U = \mathsf{OW}(mpk, PK_{\mathsf{ID}}, r)$ (U is part of the final ciphertext) and a session key $k = \mathsf{KDF}(mpk, PK_{\mathsf{ID}}, r)$, where mpk is the public parameters, OW is a one-way function and KDF

is a key derivation function; (2) use the session key k to mask the message M. Obviously, the technique obstacle arising from the randomness reuse when doubling encryption is that the reduction algorithm must be able to generate the challenge ciphertext without knowing the real randomness, i.e., the ciphertext U of r of the added encryption must be the same as what of the original encryption. Our second contribution is further proposing two refined approaches according to the different constructions of the one-way function.

- Refined Approach I if the one-way function is of the form $\mathsf{OW}(mpk, r)$.
When the one-way function of an IBE scheme $\mathcal{E}$ takes only the public parameters mpk and a randomness r, then the value U is unrelated to the public key of ID, Refined Approach I which extends the Katz-Wang technique with randomness reuse and the Fujisaki-Okamoto transformation can transform $\mathcal{E}$ to an CCA secure IBE scheme $\mathcal{E}^2_{hy}$ with a tighter reduction. The crux is that in the reduction for $\mathcal{E}^2_{hy}$, the reduction algorithm can create the value $U = \mathsf{OW}(mpk, r)$ (which is a part of the challenge ciphertext) the same way as it does in the security reduction of $\mathcal{E}$. We note that the one-way function in the full domain hash IBE family is exactly of this construction, thus the IBE schemes from this family can benefited from Refined Approach I.
- Refined Approach II if the one-way function is of the form $\mathsf{OW}(mpk, PK_{\mathsf{ID}}, r)$.
Notice that in the Katz-Wang IBE system, one identity has two public keys $PK_{\mathsf{ID},0}$ and $PK_{\mathsf{ID},1}$. If we apply the Katz-Wang technique to $\mathcal{E}$, in the reduction for the resulting scheme, the reduction algorithm is only able to generate $\mathsf{OW}(mpk, PK_{\mathsf{ID},0}, r)$ or $\mathsf{OW}(mpk, PK_{\mathsf{ID},1}, r)$ as it does in the reduction for $\mathcal{E}$. The added ciphertext can not be generated since the randomness r is unknown to it. To overcome this obstacle, we expect that one identity still has one public key and at the same time the reduction algorithm can generate a private key for any identity. We propose Refined Approach II which manage to this by doubling the master secret key. In the resulting scheme $\mathcal{E}^2_{hy}$, the PKG generates two different master secret keys named msk_0 and msk_1 and the corresponding master public parameters mpk_0 and mpk_1, while one identity still has one public key as the usual IBE schemes but has two private keys with respect to the two master secret keys. However, the PKG only generates one private key for identity ID with a randomly picked master secret key (either msk_0 or msk_1). A message is encrypted under one identity and two sets of public parameters using one randomness, while a single private key is sufficient to guarantee decrypting correctly. In contrast to the public key duplication of the Katz-Wang technique, we call the trick used in Refined Approach II the master secret key duplication. A simulation for $\mathcal{E}^2_{hy}$ can then be set up in which one master secret key is known and the other one is unknown to the reduction algorithm. For any identity ID, the reduction algorithm programs $H(\mathsf{ID})$ into V_1 with probability $1/2$; for a private key query $\langle \mathsf{ID} \rangle$, if $H(\mathsf{ID}) \in V_1$ it extracts the private key as it does in the security reduction for $\mathcal{E}$, otherwise it extracts the private key using the master secret key known to itself. This trick allows the reduction algorithm can answer all the private key queries. At the same time, since one identity only has one private key, the probability that the adversary can embed the challenge instance of $\mathcal{P}$ into the challenge ciphertext is $1/2$ since $H(\mathsf{ID}^*)$ falls into V_1 with probability $1/2$. Therefore we have $p_2 = 1/2$. The successful adversary partially decrypts the challenge ciphertext with "another" private key with probability $1/2$, giving the reduction algorithm useful information.

1.3 Outline

In Section 3, we present a generic method of improving the security reductions for a certain category of IBE schemes. Section 4 and Section 6 describe two refined approaches with respect to the different constructions of one-way function. These sections start with the descriptions of each approach and conclude with the proofs of security. Section 5 applies Refined Approach I to BF-IBE, and Section 7 applies Refined Approach II to BB_1-IBE. We provide some further discussions in Section 8 and conclude in Section 9.

2 Preliminaries

2.1 Bilinear Maps

We briefly review the facts about groups with efficiently computable bilinear map. Let $\mathbb{G}$ and $\mathbb{G}_T$ be two groups of large prime order p, and $e : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$ be the bilinear map between these two groups. A bilinear map satisfying the following three properties is said to be an admissible bilinear map.

1. Bilinearity. The map $e : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$ is bilinear if $e(u^a, v^b) = e(u, v)^{ab}$ for all $u, v \in \mathbb{G}$ and arbitrary $a, b \in \mathbb{Z}_p$.
2. Non-degeneracy. The map does not send all pairs in $\mathbb{G} \times \mathbb{G}$ to the identity in $\mathbb{G}_T$.
3. Computability. There is an efficient algorithm to compute $e(u, v)$ for any $u, v \in \mathbb{G}$.

Bilinear Map Parameter Generator. We say that a randomized algorithm $\mathcal{G}$ is a Bilinear Map parameter generator if (1) $\mathcal{G}$ takes a security parameter $\kappa \in \mathbb{Z}^+$, (2) $\mathcal{G}$ runs in polynomial time in κ, and (3) $\mathcal{G}$ outputs a κ bits prime number p, the description of two groups $\mathbb{G}$, $\mathbb{G}_T$ of order p, and the description of an admissible bilinear map $e : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$. We denote the output of $\mathcal{G}$ by $\mathcal{G}(1^\kappa) = \langle \mathbb{G}, \mathbb{G}_T, p, e \rangle$.

2.2 Bilinear Diffie-Hellman Assumption

The BDH problem [5, 14, 18] in $\mathbb{G}$ is as follows: given a tuple $g, g^x, g^y, g^z \in \mathbb{G}$ as input, output $e(g, g)^{xyz} \in \mathbb{G}_T$. An algorithm $\mathcal{A}$ has advantage ϵ in solving BDH in $\mathbb{G}$ if

$$\Pr[\mathcal{A}(g, g^x, g^y, g^z) = e(g, g)^{xyz}] \geq \epsilon$$

where the probability is over the random choice of generator g in $\mathbb{G}^*$, the random choice of $x, y, z \in \mathbb{Z}_p$.

Definition 2.1 *The (t, ϵ)-BDH assumption holds if no t-time adversary has advantage at least ϵ in solving the BDH problem in $\mathbb{G}$.*

Without loss of generality, for a number-theoretic assumption $\mathcal{P}$ we say (t, ϵ)-$\mathcal{P}$ assumption holds if no t-time adversary has advantage at least ϵ in solving the problem $\mathcal{P}$.

2.3 Basic Definitions

2.4 Basic Definitions

Extending the usual syntax of IBE [6], we describe an IBE scheme $\mathcal{E}$ with chosen plaintext security in the random oracle model by the following four fine-grained algorithms:
Setup. Takes a security parameter κ and returns the master public key *mpk* and the master secret key *msk*. Let H_1 be the identity map function which maps an identity $\mathsf{ID} \in \{0, 1\}^n$ to the underlying public key PK_{ID}, H_2 be a cryptographic hash function which maps a session key $k \in \mathcal{K}$ to a one-time-pad in $\{0, 1\}^n$. Denote the randomness space by $\mathcal{R}$. Wlog assume that the message space is $\mathcal{M} \in \{0, 1\}^n$ for some integer n.

Extract. Takes as input mpk, msk, and PK_{ID}, returns a corresponding private key SK_{ID}; we write $SK_{\mathsf{ID}} \leftarrow \mathsf{Extract}(mpk, msk, PK_{\mathsf{ID}})$, where $PK_{\mathsf{ID}} = H_1(\mathsf{ID})$.

Encrypt. Takes as input mpk, PK_{ID}, a plaintext M and a randomness r, returns a ciphertext; we write $C \leftarrow \mathsf{Encrypt}(mpk, PK_{\mathsf{ID}}, M, r)$ (in our context it is important to make explicit the randomness used in the algorithms). The algorithm $\mathsf{Encrypt}$ can be decomposed as: (1) compute $(U, k) \leftarrow \mathsf{Encaps}(r, mpk, PK_{\mathsf{ID}})$, where Encaps is an encapsulation algorithm, U is a part of the final ciphertext and k is a random session key k; (for the ease of future analysis, we further decompose algorithm Encaps into an one-way function OW and a key derivation function KDF, where $U \leftarrow \mathsf{OW}(mpk, PK_{\mathsf{ID}}, r)$ and $k \leftarrow \mathsf{KDF}(mpk, PK_{\mathsf{ID}}, r)$.) (2) set $V = M \oplus H_2(k)$. Thus the ciphertext C is of the form $\langle U, V \rangle$.

Decrypt. Takes as input mpk, private key SK_{ID}, and a ciphertext C, returns the corresponding plaintext M; we write $M \leftarrow \mathsf{Decrypt}(mpk, SK_{\mathsf{ID}}, C)$. The algorithm $\mathsf{Decrypt}$ can be decomposed as: (1) get back the session key via computing $k \leftarrow \mathsf{Decaps}(mpk, SK_{\mathsf{ID}}, U)$, where Decaps is the corresponding decapsulation algorithm; (2) return $M = V \oplus H_2(k)$.

For consistency, we require that for all $(PK_{\mathsf{ID}}, SK_{\mathsf{ID}})$, and all randomness $r \in \mathcal{R}$,

$$\Pr[\mathsf{Decaps}(mpk, SK_{\mathsf{ID}}, U) = k \mid (U, k) \leftarrow \mathsf{Encaps}(mpk, PK_{\mathsf{ID}}, r)] = 1$$

2.5 Security Notions

Chosen Ciphertext Security for IBE. An IBE scheme $\mathcal{E}$ is said to be secure against adaptively chosen ciphertext attack ($\mathsf{IND\text{-}ID\text{-}CCA}$) if no probabilistic polynomial time (PPT) adversary has a non-negligible advantage against the challenger in the following game [6]:

Setup. The challenger takes the security parameter and runs the Setup algorithm. It gives the adversary the resulting system parameters mpk and keeps the master secret msk to itself.

Phase 1. The adversary issues queries $q_1, \ldots, q_m$ where query q_i is one of:

- Extraction query $\langle \mathsf{ID}_i \rangle$. The challenger responds by running algorithm $\mathsf{Extract}$ to generate a private key d_i corresponding to ID_i. It sends d_i to the adversary.
- Decryption query $\langle \mathsf{ID}_i, C_i \rangle$. The challenger responds by running algorithm $\mathsf{Extract}$ to generate the private key d_i corresponding to ID_i. It then runs algorithm $\mathsf{Decrypt}$ to decrypt the ciphertext C_i using the private key d_i. It sends the resulting plaintext to the adversary.

These queries may be asked adaptively, that is, each query q_i may depend on the replies to $q_1, \ldots, q_{i-1}$.

Challenge. Once the adversary decides that Phase 1 is over it outputs two equal length plaintexts $M_0, M_1 \in \mathcal{M}$ and an identity ID on which it wishes to be challenged. The only constraint is that ID did not appear in any private key extraction query in Phase 1. The challenger picks a random bit $\beta \in \{0, 1\}$ and sets $C = \mathsf{Encrypt}(mpk, \mathsf{ID}, M_\beta)$. It sends C as the challenge to the adversary.

Phase 2. The adversary issues more queries $q_{m+1}, \ldots, q_r$ where q_i is one of:

- Extraction query $\langle \mathsf{ID}_i \rangle \neq \mathsf{ID}$. The challenger responds as in Phase 1.
- Decryption query $\langle \mathsf{ID}_i, C_i \rangle \neq \langle \mathsf{ID}, C \rangle$. The challenger responds as in Phase 1.

These queries may be asked adaptively as in Phase 1.

Guess. Finally, the adversary outputs a guess $\beta' \in \{0, 1\}$ and wins the game if $\beta = \beta'$.

We refer to such an adversary $\mathcal{A}$ as an $\mathsf{IND\text{-}ID\text{-}CCA}$ adversary. We define adversary $\mathcal{A}$'s advantage over the scheme $\mathcal{E}$ by $\mathrm{Adv}^{\mathsf{CCA}}_{\mathcal{A},\mathcal{E}}(\kappa) = \left|\Pr[\beta = \beta'] - \frac{1}{2}\right|$, where κ is the security parameter. The probability is over the random bits used by the challenger and the adversary. Similarly, the $\mathsf{IND\text{-}ID\text{-}CPA}$ security notion can be defined by using a similar game as the one above but disallowing decryption queries. The corresponding advantage of an adversary $\mathcal{A}$ is defined by $\mathrm{Adv}^{\mathsf{CPA}}_{\mathcal{A},\mathcal{E}}(\kappa) = \left|\Pr[\beta = \beta'] - \frac{1}{2}\right|$.

Definition 2.2 *We say that an IBE scheme $\mathcal{E}$ is (t, Q_e, Q_d, ϵ) chosen ciphertext secure if for any t-time* $\mathsf{IND\text{-}ID\text{-}CCA}$ *adversary $\mathcal{A}$ that makes at most Q_e chosen private key queries and at most Q_d chosen decryption queries we have that $\mathrm{Adv}^{\mathsf{CCA}}_{\mathcal{A},\mathcal{E}} < \epsilon$. As shorthand, we say that $\mathcal{E}$ is (t, Q_e, Q_d, ϵ)* $\mathsf{IND\text{-}ID\text{-}CCA}$ *secure.*

Definition 2.3 *We say that an IBE scheme $\mathcal{E}$ is (t, Q_e, ϵ) chosen plaintext secure if for any t-time* $\mathsf{IND\text{-}ID\text{-}CPA}$ *adversary $\mathcal{A}$ that makes at most Q_e chosen private key queries we have that $\mathrm{Adv}^{\mathsf{CPA}}_{\mathcal{A},\mathcal{E}} < \epsilon$. As shorthand, we say that $\mathcal{E}$ is (t, Q_e, ϵ)* $\mathsf{IND\text{-}ID\text{-}CPA}$ *secure.*

3 Generic Method to Achieve Tighter CPA Security

Rather than proceed in an ad hoc manner, in this section we present a generic method directly from the Katz-Wang technique which improves IBE schemes in terms of the (CPA) security reduction. The generic method transforms a CPA secure IBE scheme $\mathcal{E}$ to an IBE scheme $\mathcal{E}^2$ as follows:

Setup^2. The same as $\mathcal{E}$.

Extract^2. For a given identity ID, picks a random bit $b \in \{0, 1\}$, returns a private key $SK^2_{\mathsf{ID}} = (b, SK_{\mathsf{ID},b}) = (b, \mathsf{Extract}(mpk, msk, PK_{\mathsf{ID},b}))$, where $PK_{\mathsf{ID},b} = H_1(\mathsf{ID}||b)$.

Encrypt^2. To encrypt a message M under ID, computes $C_0 = \mathsf{Encrypt}(mpk, PK_{\mathsf{ID},0}, M, r_0)$ and $C_1 = \mathsf{Encrypt}(mpk, PK_{\mathsf{ID},1}, M, r_1)$, where $C_0 = \langle U_0, V_0\rangle$, $C_1 = \langle U_1, V_1\rangle$, where r_0 and r_1 are two independent random values used by the algorithm. The ciphertext is $C = \langle U_0, V_0, U_1, V_1\rangle$.

Decrypt^2. Computes $M = \mathsf{Decrypt}(mpk, SK_{\mathsf{ID},b}, C_b)$ using $SK^2_{\mathsf{ID}} = (b, SK_{\mathsf{ID},b})$.

If $\mathcal{E}$ satisfies two constraints as addressed before: (1) provably secure in the random oracle model; (2) the δ in the original security reduction for $\mathcal{E}$ could be $1/2$, then we have the following theorem about the security of $\mathcal{E}^2$.

Theorem 3.1 *If $\mathcal{E}$ is (t, Q_e, ϵ)* $\mathsf{IND\text{-}ID\text{-}CPA}$ *secure assuming $(t', p_1p_2p_3\epsilon)$-$\mathcal{P}$ holds, then $\mathcal{E}^2$ is (t, Q_e, ϵ)* $\mathsf{IND\text{-}ID\text{-}CPA}$ *secure assuming $(t', \frac{1}{2}p_3\epsilon)$-$\mathcal{P}$ holds.*

Proof. Suppose $\mathcal{A}_1$ is a (t, Q_e, ϵ) $\mathsf{IND\text{-}ID\text{-}CPA}$ adversary against $\mathcal{E}$. According to the assumption in the theorem, there exists a $(t', p_1p_2p_3\epsilon)$ adversary $\mathcal{B}_1$ against $\mathcal{P}$, who interacts with $\mathcal{A}_1$ in an $\mathsf{IND\text{-}ID\text{-}CPA}$ game (Game 1) as follows:

Setup. $\mathcal{B}_1$ builds the mpk of $\mathcal{E}$ from the given challenge instance of $\mathcal{P}$, while the corresponding msk is unknown to $\mathcal{B}_1$. $\mathcal{B}_1$ starts by initializing one empty list L_i for each random oracle H_i.

H_1-**queries**. When $\mathcal{A}_1$ queries random oracle H_1 at point ID, $\mathcal{B}_1$ programs $H_1(\mathsf{ID})$ to be an element in V_1 with probability δ, in V_2 with probability $1 - \delta$.

H_2-**queries**. $\mathcal{B}_1$ handles the queries to H_2 in an obvious way, by producing a randomly sampled element from the appropriate codomain, and adding both query and answer to the L_2 list.

Phase 1 - Private key queries. When $\mathcal{A}_1$ queries the private key for ID, if $H_1(\mathsf{ID})$ belongs to V_1, $\mathcal{B}_1$ is able to generate the corresponding private key, otherwise $\mathcal{B}_1$ aborts.

Challenge. $\mathcal{A}_1$ submits two messages M_1, M_2 and an identity ID^* which it wishes to be challenged on. If $H_1(\mathsf{ID}^*)$ belongs to V_1, $\mathcal{B}_1$ aborts. Otherwise, $\mathcal{B}_1$ picks a random bit β and generates the challenge ciphertext $C^* = \langle U^*, V^*\rangle$ of M_β embedded with the challenge instance of $\mathcal{P}$.

Phase 2 - Private key queries. The same as Phase 1.

Guess. $\mathcal{A}_1$ outputs $\beta' \in \{0,1\}$. $\mathcal{B}_1$ outputs its answer based on the entries on the L_2 list.

This finishes the description of Game 1. ■

Let $\mathcal{A}_2$ be a (t, Q_e, ϵ) IND-ID-CPA adversary against $\mathcal{E}^2$, and we build an adversary $\mathcal{B}_2$ against $\mathcal{P}$. Concretely speaking, $\mathcal{B}_2$ interacts with $\mathcal{A}_2$ in an IND-ID-CPA game (Game 2) as follows:

Setup. The same as Game 1.

H_1**-queries**. When $\mathcal{A}_2$ queries random oracle H_1 at point $\mathsf{ID}|*$ ($*$ denotes 0 or 1), $\mathcal{B}_2$ picks a random bit $b \in \{0,1\}$ and programs $H_1(\mathsf{ID}||b)$ to be an element in V_1, $H_1(\mathsf{ID}||\bar{b})$ to be an element in V_2. It is easy to see that $H_1(\mathsf{ID}||0)$ and $H_1(\mathsf{ID}||1)$ are uniform in V and are independent of $\mathcal{A}_2$'s current view.

Phase 1 - Private key queries. When $\mathcal{A}_2$ queries the private key of ID,

- If $H_1(\mathsf{ID}||0) \in V_1$, $\mathcal{B}_2$ generates $SK_{\mathsf{ID},0}$ for $H_1(\mathsf{ID}||0)$ with the trapdoor information as $\mathcal{B}_1$ does in Game 1, then responds with $SK^2_{\mathsf{ID}} = (0, SK_{\mathsf{ID},0})$.
- Otherwise $\mathcal{B}_2$ generates $SK_{\mathsf{ID},1}$ for to $H_1(\mathsf{ID}||1)$ with the trapdoor information as $\mathcal{B}_1$ does in Game 1, then responds with $SK^2_{\mathsf{ID}} = (1, SK_{\mathsf{ID},1})$.

Note that $\mathcal{B}_2$ can answer all the private key queries.

Challenge. $\mathcal{A}_2$ submits two messages M_1, M_2, and an identity ID^* which it wishes to be challenged on. Suppose $H_1(\mathsf{ID}^*||b) \in V_2$, $\mathcal{B}_2$ picks a random bit β and generates C^*_b of M_β as $\mathcal{B}_1$ does in Game 1 (C^*_b is embedded with the challenge instance of $\mathcal{P}$). Additionally, $\mathcal{B}_2$ picks a random value $r^*_{\bar{b}}$ and computes $C^*_{\bar{b}} = \mathsf{Encrypt}(mpk, PK_{\mathsf{ID}^*||\bar{b}}, M_\beta, r^*_{\bar{b}})$. The challenge ciphertext of M_β is $\langle C^*_b, C^*_{\bar{b}} \rangle$.

Phase 2 - Private key queries. The same as Phase 1.

Guess. $\mathcal{A}_2$ outputs its guess $\beta' \in \{0,1\}$. $\mathcal{B}_2$ outputs its answer to $\mathcal{P}$ based on the entries in the L_2 list.

This finishes the description of Game 2. ■

Claim. $\mathcal{B}_2$ outputs the correct solution of $\mathcal{P}$ with probability $\frac{1}{2}p_3\epsilon$.
Proof of claim. For adversary $\mathcal{B}_2$, we have:

- $p'_1 = 1$. $\mathcal{B}_2$ can answer all the private key queries.
- $p'_2 = 1$. For any challenge identity ID^*, $\mathcal{B}_2$ can generate the challenge ciphertext embedded with the challenge instance of $\mathcal{P}$.
- $p'_3 = p_3/2$. Because the bit b is information-theoretically hidden from $\mathcal{A}_2$, the probability that $\mathcal{A}_2$ decrypts M_β using the part of C^* embedded with the challenge instance of $\mathcal{P}$ is at least 1/2.

This finishes the proof of Theorem 3.1. □

4 Refined Approach I

When the one-way function in the encapsulation algorithm Encaps of $\mathcal{E}$ takes on the form of $\mathsf{OW}(mpk, r)$, that is, the encryption result of the randomness r is unrelated of the public key PK_{ID}, we can improve the generic approach proposed in Section 3. We name the improved approach Refined Approach I, which transforms a CPA-secure IBE scheme $\mathcal{E}$ to a CCA-secure IBE scheme $\mathcal{E}^2_{hy}$ with a tighter security reduction. It works as follows:

$\mathbf{Setup}^2_{hy}$. As in $\mathcal{E}$. In addition, picks a cryptographic hash function $H_3 : \{0,1\}^n \times \{0,1\}^n \to \mathcal{R}$, and a cryptographic hash function $H_4 : \{0,1\}^n \to \{0,1\}^n$.

Extract$^2_{hy}$. The same as $\mathcal{E}^2$ in Section 3.

Encrypt$^2_{hy}$. To encrypt a message M under ID do the following:

1. Compute $PK_{\mathsf{ID},0} = H_1(\mathsf{ID}||0)$ and $PK_{\mathsf{ID},1} = H_1(\mathsf{ID}||1)$;
2. Pick a random $\sigma \in \{0,1\}^n$, then compute $r = H_3(\sigma, M)$ and $U = \mathsf{OW}(mpk, r)$;
3. Compute $k_0 = \mathsf{KDF}(mpk, PK_{\mathsf{ID},0}, r)$ and $k_1 = \mathsf{KDF}(mpk, PK_{\mathsf{ID},1}, r)$;
4. Set $V_0 = \sigma \oplus H_2(k_0)$ and $V_1 = \sigma \oplus H_2(k_1)$;
5. Compute $W = M \oplus H_4(\sigma)$. The ciphertext C is $\langle U, V_0, V_1, W\rangle$.

Decrypt$^2_{hy}$. To decrypt C using the private key $SK^2_{\mathsf{ID}} = (b, SK_{\mathsf{ID},b})$, the algorithm does the following steps:

1. Compute $k_b = \mathsf{Decaps}(mpk, SK_{\mathsf{ID},b}, U)$ and return $\sigma = V_b \oplus H_2(k_b)$;
2. Compute $M = W \oplus H_4(\sigma)$ and set $r = H_3(\sigma, M)$;
3. Test that if $U = \mathsf{OW}(mpk, r)$. If not, reject the ciphertext.
4. Compute $k_{\bar{b}} = \mathsf{KDF}(mpk, PK_{\mathsf{ID},\bar{b}}, r)$, set $\sigma' = V_{\bar{b}} \oplus H_2(k_{\bar{b}})$. Test that if $\sigma = \sigma'$. If so, output M as the decryption of C. If not, reject the ciphertext.

We have the following theorem regarding to the security of $\mathcal{E}^2_{hy}$:

Theorem 4.1 *If $\mathcal{E}$ is (t, Q_e, ϵ) $\mathsf{IND\text{-}ID\text{-}CPA}$ secure assuming $(t', p_1p_2p_3\epsilon)$-$\mathcal{P}$ holds, then $\mathcal{E}^2_{hy}$ is (t, Q_e, Q_d, ϵ) $\mathsf{IND\text{-}ID\text{-}CCA}$ secure assuming $(t', \frac{1}{2}p_3\epsilon)$-$\mathcal{P}$ holds.*

Proof. Suppose $\mathcal{A}_1$ is a (t, Q_e, ϵ) $\mathsf{IND\text{-}ID\text{-}CPA}$ adversary against $\mathcal{E}$. According to the assumption the theorem, there exists a $(t', p_1p_2\epsilon)$ adversary $\mathcal{B}_1$ against $\mathcal{P}$. $\mathcal{B}_1$ interacts with $\mathcal{A}_1$ in an $\mathsf{IND\text{-}ID\text{-}CPA}$ game (Game 1). Game 1 is exactly the same as that described in the proof of the general approach in Section 3. Let $\mathcal{A}_2$ be a (t, Q_e, Q_d, ϵ) adversary against $\mathcal{E}^2_{hy}$, we build an adversary $\mathcal{B}_2$ against $\mathcal{P}$, who interacts with $\mathcal{A}_2$ in an $\mathsf{IND\text{-}ID\text{-}CCA}$ game (Game 3) as follows:

Setup. The same as the Game 2 presented in Section 3.

H_1-**queries**. The same as the Game 2 described in Section 3.

H_2-**queries**. The same as the Game 2 described in Section 3.

Phase 1 - Private key queries. The same as the Game 2 presented in Section 3.

Phase 1 - Decryption queries. Upon receiving the decryption query $\langle C, \mathsf{ID}\rangle$, $\mathcal{B}_2$ decrypts C using the private key SK^2_{ID} normally. Note that $\mathcal{B}_2$ can generate the private key for any identity, thus it can answer all the decryption queries.

Challenge. $\mathcal{A}_2$ submits two messages M_1, M_2, and the identity ID^* which it wishes to be challenged on. Suppose $H_1(\mathsf{ID}^*||b) \in V_2$, $\mathcal{B}_2$ picks a random bit β, random $\sigma^* \in \{0,1\}^n$. $\mathcal{B}_2$ generates U^* and V^*_b the same way as $\mathcal{B}_1$ generates U^* and V^* in Game 1, while the only difference is replacing M_β with σ^*. $\mathcal{B}_2$ computes $V^*_{\bar{b}} = M_\beta \oplus H_2(k^*_{\bar{b}})$ using the private key $SK^2_{\mathsf{ID}^*} = (\bar{b}, SK_{\mathsf{ID}^*,\bar{b}})$, where $k^*_{\bar{b}} = \mathsf{Decaps}(mpk, SK_{\mathsf{ID}^*,\bar{b}}, U^*)$. Finally, $\mathcal{B}_2$ sets $W^* = M_\beta \oplus H_4(\sigma^*)$. The challenge ciphertext $C^* = \langle U^*, V^*_b, V^*_{\bar{b}}, W^*\rangle$.

Phase 2 - Private key queries. Handled the same way as Phase 1.

Phase 2 - Decryption queries. Handled the same way as Phase 1.

Guess. $\mathcal{A}_2$ outputs its guess $\beta' \in \{0,1\}$. $\mathcal{B}_2$ outputs its answer to $\mathcal{P}$ based on the entries in the L_2 list.

This finishes the description of Game 3. ■

Claim. $\mathcal{B}_2$ outputs the correct solution of $\mathcal{P}$ with advantage $\frac{1}{2}p_3\epsilon$.

Proof of claim. For adversary $\mathcal{B}_2$, we have:

- $p'_1 = 1$. Since $\mathcal{B}_2$ can answer all the private key queries.
- $p'_2 = 1$. For any challenge identity ID^*, $\mathcal{B}_2$ can always generate the challenge ciphertext embedded with the challenge instance of $\mathcal{P}$.
- $p'_3 = p_3/2$. Since the bit b is information-theoretically hidden from $\mathcal{A}_2$.

This finishes the proof of Theorem 4.1. □

5 A Variant of BF-IBE with Tight Security Reduction

In this section, we apply the Refined Approach I to the BF-IBE ($\mathsf{BasicIdent}$) [6]. The resulting scheme is as follows:

Setup. To generate system parameters, picks a random generator $g \in \mathbb{G}^*$, a random integer $s \in \mathbb{Z}_p^*$ and sets $X = g^s$. Chooses four cryptographic hash functions: $H_1 : \{0,1\}^* \to \mathbb{G}^*$, $H_2 : \mathbb{G}_T \to \{0,1\}^n$ for some integer n, $H_3 : \{0,1\}^n \times \{0,1\}^n \to \mathbb{Z}_p^*$, and $H_4 : \{0,1\}^n \to \{0,1\}^n$. The mpk is (g, X), while the msk is s. The message space is $\mathcal{M} = \{0,1\}^n$. The ciphertext space is $\mathcal{C} = \mathbb{G}^* \times \{0,1\}^n \times \{0,1\}^n \times \{0,1\}^n$.

KeyGen. To generate the private key SK_{ID} for an identity $\mathsf{ID} \in \{0,1\}^*$, the algorithm does the following steps: (1) pick a random bit $b \in \{0,1\}$ and compute $Q_b = H_1(\mathsf{ID}||b) \in \mathbb{G}^*$, (2) set the private key SK_{ID} to be (b, Q_b^s) where s is the master secret key.

Encrypt. To encrypt a message $M \in \{0,1\}^n$ under an identity ID, the algorithm does the following steps:

1. Compute $Q_0 = H_1(\mathsf{ID}||0)$ and $Q_1 = H_1(\mathsf{ID}||1)$;
2. Choose a random $\sigma \in \{0,1\}^n$ and compute $r = H_3(\sigma, M)$;
3. Set the ciphertext to be $C = \langle U, V_0, V_1, W \rangle = \langle g^r, \sigma \oplus H_2(e(Q_0, X)^r), \sigma \oplus H_2(e(Q_1, X)^r), M \oplus H_4(\sigma) \rangle$.

Decrypt. To decrypt a given ciphertext $C = \langle U, V_0, V_1, W \rangle$ under ID using the private key $SK_{\mathsf{ID}} = (b, Q_b^s)$, the algorithm does the following steps:

1. Compute $V_b \oplus H_2(e(Q_b^s, U)) = \sigma$.
2. Compute $W \oplus H_4(\sigma) = M$.
3. Set $r = H_3(\sigma, M)$. Test that $U = g^r$. If not, reject the ciphertext.
4. Compute $V_{\bar{b}} \oplus H_2(e(Q_{\bar{b}}, X)^r) = \sigma'$. Test that if $\sigma = \sigma'$. If so, output the plaintext M. If not, reject the ciphertext.

Theorem 5.1 *Our variant of BF-IBE is* $\mathsf{IND\text{-}ID\text{-}CCA}$ *secure provided that H_1 and H_2 are two random oracles and the CBDH assumption holds in $\mathbb{G}$. Concretely, if there is an* $\mathsf{IND\text{-}ID\text{-}CCA}$ *adversary $\mathcal{A}$ that has advantage ϵ against the scheme. Suppose $\mathcal{A}$ makes at most $Q_{h_2} > 0$ hash queries to H_2. Then there is an algorithm $\mathcal{B}$ that solves the CBDH problem in $\mathbb{G}$ with advantage at least:* $\mathrm{Adv}_{\mathcal{B}} \geq \epsilon / Q_{h_2}$.

Proof. Suppose $\mathcal{A}$ has advantage ϵ in attacking the system. We build an algorithm $\mathcal{B}$ that solves the BDH problem. Algorithm $\mathcal{B}$ is given as input a random 4-tuple $(g, g^x, g^y, g^z) = (g, g_1, g_2, g_3)$ with the goal to output $T = e(g,g)^{xyz}$. Algorithm $\mathcal{B}$ works by interacting with $\mathcal{A}$ in an $\mathsf{IND\text{-}ID\text{-}CCA}$ game as follows.

Setup. $\mathcal{B}$ set the mpk to be (g, X), where $X = g^x$. This operation implicitly sets $msk = x$. From the perspective of the adversary $\mathcal{A}$ the distribution of the public parameters are identical to the real construction.

H_1**-queries**. At any time $\mathcal{A}$ can query the random oracle H_1. To respond to these queries $\mathcal{B}$ maintains a list of tuples $\langle \mathsf{ID}, c, m_c, Q_c, m_{\bar{c}}, Q_{\bar{c}} \rangle$ as explained below. We refer to this list as the L_1 list, which is initially empty. When $\mathcal{A}$ queries the oracle H_1 at a point $\mathsf{ID}||b$ (b could be 0 or 1) algorithm $\mathcal{B}$ responds as follows:

1. If ID already appears on the L_1 list in a tuple $(\mathsf{ID}, c, m_c, Q_c, m_{\bar{c}}, Q_{\bar{c}})$, then algorithm $\mathcal{B}$ responds with $H_1(\mathsf{ID}||b) = Q_b$.
2. Otherwise, $\mathcal{B}$ picks a random bit $c \in \{0, 1\}$, $m_c, m_{\bar{c}} \in \mathbb{Z}_p^*$ and computes $Q_c = g^{m_c}$ and $Q_{\bar{c}} = g_2^{m_{\bar{c}}}$. $\mathcal{B}$ inserts $\langle \mathsf{ID}, c, m_c, Q_c, m_{\bar{c}}, Q_{\bar{c}} \rangle$ into the L_1 list, and responds to $\mathcal{A}$ with $H_1(\mathsf{ID}||b) = Q_b$.

Note that $H_1(\mathsf{ID}||b)$ is uniformly distributed over $\mathbb{G}^*$ and independent of $\mathcal{A}$'s view.

H_2**-queries**. $\mathcal{B}$ handles the queries to H_2 the obvious way, by producing a randomly sampled element from the appropriate codomain, and adding both query and answer to the L_2 list.

Phase 1 - Private key queries. Upon receiving the private key extraction query for an identity ID, let $\langle \mathsf{ID}, c, m_c, Q_c, m_{\bar{c}}, Q_{\bar{c}} \rangle$ be the corresponding tuple in the L_1 list. $\mathcal{B}$ responds the private key $SK_{\mathsf{ID}} = (c, Q_c^x) = (c, X^{m_c})$. Note that SK_{ID} is a valid private key of ID since $Q_c = g^{m_c}$.

Phase 1 - Decryption queries. Upon receiving the decryption query $\langle C, \mathsf{ID} \rangle$, $\mathcal{B}$ decrypts M using the private key SK_{ID} normally. Note that $\mathcal{B}$ can generate the private key for any identity, thus it can answer all the decryption queries.

Challenge. The adversary $\mathcal{A}$ submits two messages $M_0, M_1 \in \{0, 1\}^n$ and an identity ID^* where it wishes to be challenged. Suppose $\langle \mathsf{ID}^*, c, m_c, Q_c, m_{\bar{c}}, Q_{\bar{c}} \rangle$ is the corresponding entry on the L_1 list. $\mathcal{B}$ flips a fair coin $\beta \in \{0, 1\}$, picks random bits $\sigma \in \{0, 1\}^n$, a random string $R \in \{0, 1\}^n$, set the challenge ciphertext C^* to be

$$U^* = g_3, V_c^* = \sigma \oplus H_2(e(Q_c, X)^z), V_{\bar{c}}^* = \sigma \oplus R, W = M \oplus \sigma.$$

This operation implicitly sets $r = z$, $R = H_2(e(Q_{\bar{c}}, X)^r) = H_2(e(g_2^{m_{\bar{c}}}, g_1)^z) = H_2(T^{m_{\bar{c}}})$. $\mathcal{B}$ obtains $e(Q_c, X)^z$ by computing $e(Q_c^x, U^*)$ using the private key $SK_{\mathsf{ID}^*} = (c, Q_c^x)$ of ID^*.

Phase 2 - Private key queries. Handled the same way as Phase 1.

Phase 2 - Decryption queries. Handled the same way as Phase 1.

Guess. Finally, the adversary $\mathcal{A}$ outputs its guess β' for β. At this point $\mathcal{B}$ picks a random tuple $\langle \hat{T}, \hat{R} \rangle$ from the L_2 list and outputs $\hat{T}^{m_{\bar{c}}^{-1}}$ as the solution to the given instance of BDH problem.

Claim. The responses to H_1-queries and H_2 queries are as in the real attack. All responses to private key extraction queries and decryption queries are valid. Since the bit c is information-theoretically hidden from the view of adversary $\mathcal{A}$. The desired $\hat{T}$ will appear on some entry in the L_2 list with probability at least ϵ.

This shows that $\mathcal{B}$'s advantage is at least ϵ/Q_{h_2} as required. □

6 Refined Approach II

In IBE schemes, we can further split mpk into two parts, one part we denote by mpk^* which is independent of msk, the other part we denote by mpk' which is related to msk. We have $mpk = (mpk^*, mpk')$. When the one-way function of the encapsulation algorithm in $\mathcal{E}$ takes on the form of $\mathsf{OW}(mpk^*, PK_{\mathsf{ID}}, r)$, that is, the value U is unrelated to mpk', we can improve the generic method proposed in Section 3 using master secret key duplication. We name the refined approach Refined Approach II, which transforms a CPA-secure IBE scheme $\mathcal{E}$ to a CCA-secure IBE scheme $\mathcal{E}_{hy}^2$ with a tighter reduction. It works as follows:

Setup$^2_{hy}$. Doubles msk to obtain msk_0 and msk_1, generates the corresponding mpk'_0 and mpk'_1, keep mpk^* unaltered. The resulting public parameter is $mpk = mpk_0 \cap mpk_1$, where $mpk_0 = (mpk^*, mpk'_0)$ and $mpk_1 = (mpk^*, mpk'_1)$. In addition, picks a cryptographic hash function $H_3 : \{0,1\}^n \times \{0,1\}^n \rightarrow \mathcal{R}$, and a cryptographic hash function $H_4 : \{0,1\}^n \rightarrow \{0,1\}^n$.

Extract$^2_{hy}$. For a given an identity ID, picks a random bit $b \in \{0,1\}$, returns a private key $SK^2_{\mathsf{ID}} = (b, \mathsf{Extract}(mpk_b, msk_b, PK_{\mathsf{ID}}))$, where $PK_{\mathsf{ID}} = H_1(\mathsf{ID})$.

Encrypt$^2_{hy}$. To encrypt a message M under ID do the following:

1. Compute $PK_{\mathsf{ID}} = H_1(\mathsf{ID})$;
2. Pick a random $\sigma \in \{0,1\}^n$, compute $r = H_3(\sigma, M)$, $U = \mathsf{OW}(mpk^*, PK_{\mathsf{ID}}, r)$;
3. Compute $k_0 = \mathsf{KDF}(mpk_0, PK_{\mathsf{ID}}, r)$ and $k_1 = \mathsf{KDF}(mpk_1, PK_{\mathsf{ID}}, r)$;
4. Set $V_0 = \sigma \oplus H_2(k_0)$ and $V_1 = \sigma \oplus H_2(k_1)$;
5. Compute $W = M \oplus H_4(\sigma)$. The ciphertext C is $\langle U, V_0, V_1, W \rangle$.

Decrypt$^2_{hy}$. To decrypt C using the private key $SK^2_{\mathsf{ID}} = (b, SK_{\mathsf{ID},b})$, the algorithm does the following steps:

1. Compute the session key $k_b = \mathsf{Decaps}(mpk_b, SK_{\mathsf{ID},b}, U)$ and return $\sigma = V_b \oplus H_2(k_b)$;
2. Compute $M = W \oplus H_4(\sigma)$ and $r = H_3(\sigma, M)$.
3. Test that if $U = \mathsf{OW}(mpk^*, PK_{\mathsf{ID}}, r)$. If not, reject the ciphertext.
4. Compute $k_{\bar{b}} = \mathsf{KDF}(mpk_{\bar{b}}, PK_{\mathsf{ID}}, r)$, set $\sigma' = V_{\bar{b}} \oplus k_{\bar{b}}$. Test that if $\sigma = \sigma'$. If so, output the plaintext M. If not, reject the ciphertext.

We have the following theorem regarding to the security of $\mathcal{E}^2_{hy}$:

Theorem 6.1 *If $\mathcal{E}$ is (t, Q_e, ϵ)* $\mathsf{IND\text{-}ID\text{-}CPA}$ *secure assuming $(t', p_1 p_2 p_3 \epsilon)$-$\mathcal{P}$ holds, then $\mathcal{E}^2_{hy}$ is (t, Q_e, Q_d, ϵ) secure* $\mathsf{IND\text{-}ID\text{-}CCA}$ *assuming $(t', \frac{1}{4} p_3 \epsilon)$-$\mathcal{P}$ holds.*

Proof. Suppose $\mathcal{A}_1$ is a (t, Q_e, ϵ) $\mathsf{IND\text{-}ID\text{-}CPA}$ adversary against $\mathcal{E}$. According to the assumption in the theorem, there exists a $(t', p_1 p_2 p_3 \epsilon)$ adversary $\mathcal{B}_1$ against $\mathcal{P}$. $\mathcal{B}_1$ interacts with $\mathcal{A}_1$ in an $\mathsf{IND\text{-}ID\text{-}CPA}$ game (Game 1) as described in Section 3. Let $\mathcal{A}_2$ be a (t, Q_e, Q_d, ϵ) $\mathsf{IND\text{-}ID\text{-}CCA}$ adversary against $\mathcal{E}^2_{hy}$, we build an adversary $\mathcal{B}_2$ against $\mathcal{P}$, who interacts with $\mathcal{A}_2$ in an $\mathsf{IND\text{-}ID\text{-}CPA}$ game (Game 4) as follows:

Setup. $\mathcal{B}_2$ picks a random bit b, then generates mpk^* and mpk'_b as $\mathcal{B}_1$ does in Game 1, while msk_b is unknown to $\mathcal{B}_2$. In addition, $\mathcal{B}_2$ picks $msk_{\bar{b}}$ itself, and generates the associated $mpk'_{\bar{b}}$ accordingly.

H_1**-queries**. For a given identity ID, $\mathcal{B}_2$ programs $H_1(\mathsf{ID})$ to be an element in V_1 with probability 1/2, in V_2 with probability 1/2. Note that either way $H_1(\mathsf{ID})$ is uniform in V and is independent of $\mathcal{A}_2$'s current view.

H_2**-queries**. The same as the Game 2 presented in Section 3.

Phase 1 - Private key queries. When $\mathcal{A}_1$ queries the private key of ID,

- If $H_1(\mathsf{ID})$ belongs to V_1, $\mathcal{B}_2$ generates SK_{ID} with the trapdoor information as $\mathcal{B}_1$ does in Game 1, then responds with the private key $SK^2_{\mathsf{ID}} = (b, SK_{\mathsf{ID}})$ (related to msk_b), where b is the hidden bit.
- Otherwise, $\mathcal{B}_2$ generates SK_{ID} using $msk_{\bar{b}}$, then responds with the private key $SK^2_{\mathsf{ID}} = (\bar{b}, SK_{\mathsf{ID}})$.

It is clearly that $\mathcal{B}_2$ can answer all the private key queries.

Phase 1 - Decryption queries. Upon receiving the decryption query $\langle C, \mathsf{ID}\rangle$, $\mathcal{B}_2$ decrypts C using the private key SK_{ID} normally. Note that $\mathcal{B}_2$ can generate the private key for any identity, thus it can answer all the decryption queries.

Challenge. $\mathcal{A}_2$ submits two messages M_1, M_2, and an identity ID^* which it wishes to be challenged on. If $H_1(\mathsf{ID}^*) \in V_1$, $\mathcal{B}_2$ aborts. Otherwise, $\mathcal{B}_2$ picks a random bit β, random $\sigma \in \{0,1\}^n$, and computes $r^* = H_3(\sigma^*, M_\beta)$, $\mathcal{B}_2$ generates U^* and V_b^* the same way as $\mathcal{B}_1$ generates U^* and V^* in Game 1, while the only difference is replacing M_β with σ^*. $\mathcal{B}_2$ computes $V_{\bar{b}}^* = M_\beta \oplus H_2(k_{\bar{b}}^*)$ using the private key $SK_{\mathsf{ID}^*}^2 = (\bar{b}, SK_{\mathsf{ID}^*,\bar{b}})$, where $k_{\bar{b}}^* = \mathsf{Decaps}(mpk_{\bar{b}}, SK_{\mathsf{ID}^*,\bar{b}}, U^*)$. Finally, $\mathcal{B}_2$ sets $W^* = M_\beta \oplus H_4(\sigma^*)$. The challenge ciphertext $C^* = \langle U^*, V_b^*, V_{\bar{b}}^*, W^*\rangle$.

Phase 2 - Private key queries. Handled the same way as Phase 1.

Phase 2 - Decryption queries. Handled the same way as Phase 1.

Guess. $\mathcal{A}_2$ outputs its answer. $\mathcal{B}_2$ outputs its answer to $\mathcal{P}$ based on the entries in the L_2 list.

This finishes the description of Game 4. ■

Claim. $\mathcal{B}_2$ outputs the correct solution of $\mathcal{P}$ with advantage $\frac{1}{4}p_3\epsilon$.
Proof of claim. For adversary $\mathcal{B}_2$, we have:

– $p_1' = 1$. Since $\mathcal{B}_2$ can answer all the private key queries.
– $p_2' = 1/2$. For any challenge identity ID^*, $\mathcal{B}_2$ can generate the challenge ciphertext embedded with the challenge instance of $\mathcal{P}$ with probability $1/2$.
– $p_3' = p_3/2$. Since the bit b is information-theoretically hidden from $\mathcal{A}_2$.

This finishes the proof of Theorem 6.1. □

7 A Variant of BB$_1$-IBE with Tight Security Reduction

Boneh and Boyen [4] proposed two efficient IBE schemes BB_1-IBE and BB_2-IBE which are proven secure in the standard model. However, they only have selective-ID security [8]. Interestingly, BB_1-IBE can also be proven fully secure in the random oracle model if we model its identity map function as a random oracle. In this section, we apply the Refined Approach II to BB_1-IBE. The resulting scheme is as follows:
Setup. To generate system parameters, selects two random integers $x_0, x_1 \in \mathbb{Z}_p$, two random elements $g, Y \in \mathbb{G}$ and computes $X_0 = g^{x_0}$, $X_1 = g^{x_1}$. Next, picks four cryptographic hash functions $H_1 : \{0,1\}^* \to \mathbb{G}$, $H_2 : \mathbb{G}_T \to \{0,1\}^n$, $H_3 : \{0,1\}^n \times \{0,1\}^n \to \mathbb{Z}_p^*$, and $H_4 : \{0,1\}^n \to \{0,1\}^n$. The mpk is (g, X_0, X_1, Y). The msk is $(Y_0 = Y^{x_0}, Y_1 = Y^{x_1})$. The message space is $\mathcal{M} = \{0,1\}^n$. The ciphertext space is $\mathcal{C} = \mathbb{G} \times \mathbb{G} \times \{0,1\}^n \times \{0,1\}^n \times \{0,1\}^n$.

KeyGen. To generate the private key SK_{ID} for an identity $\mathsf{ID} \in \{0,1\}^*$, picks a random $r \in \mathbb{Z}_p$ and a random bit b, sets $SK_{\mathsf{ID}} = (d_0, d_1, d_2) = (b, Y_b Q^r, g^r)$, where $Q = H_1(\mathsf{ID})$ can be viewed as the public key of ID.

Encrypt. To encrypt a message M under the identity ID, picks a random $\sigma \in \{0,1\}^n$, computes $Q = H_1(\mathsf{ID})$ and $z = H_3(\sigma, M)$, sets the ciphertext to be: $C = \langle U, V, W_0, W_1, S\rangle = \langle g^z, Q^z, \sigma \oplus H_2(e(Q, X_0)^z), \sigma \oplus H_2(e(Q, X_1)^z), M \oplus H_4(\sigma)\rangle$.

Decrypt. To decrypt a given ciphertext $C = \langle U, V, W_0, W_1, S\rangle$ under ID using the private key $SK_{\mathsf{ID}} = (b, d_1, d_2)$ does:

1. Compute $e(d_1, U)/e(d_2, V) = e(Y_b Q^r, g^z)/e(g^r, Q^z) = e(X_b, Y)^z$;
2. Compute $W_b \oplus H_2(e(X_b, Y)^z) = \sigma$, $S \oplus H_4(\sigma) = M$, $H_3(\sigma, M) = z$;
3. Test that if $U = g^z$ and $V = Q^z$, if not, reject the ciphertext.

4. Compute $W_{\bar{b}} \oplus H_2(e(X_{\bar{b}}, Y)^z) = \sigma'$. If $\sigma = \sigma'$, output the plaintext M. Otherwise, reject the ciphertext.

Theorem 7.1 *The above variant of BB_1-IBE is* $\mathsf{IND\text{-}ID\text{-}CCA}$ *secure provided that H_1, H_2 are random oracles and the CBDH assumption holds in $\mathbb{G}$. Concretely, suppose there is an* $\mathsf{IND\text{-}ID\text{-}CCA}$ *adversary $\mathcal{A}$ that has advantage ϵ against the scheme. If $\mathcal{A}$ makes at most $Q_{h_2} > 0$ queries to H_2. Then there is an algorithm $\mathcal{B}$ that solves the CBDH problem with advantage at least:* $\mathrm{Adv}_{\mathcal{B}} \geq \epsilon / 2Q_{h_2}$.

Proof. Suppose $\mathcal{A}$ has advantage ϵ in attacking the variant of BB_1-IBE. We build an algorithm $\mathcal{B}$ that solves the BDH problem. Algorithm $\mathcal{B}$ is given as input a random 4-tuple $(g, g_1, g_2, g_3) = (g, g^x, g^y, g^z)$, with the goal to output $T = e(g, g)^{xyz}$. Algorithm $\mathcal{B}$ works by interacting with $\mathcal{A}$ in an $\mathsf{IND\text{-}ID\text{-}CCA}$ game as follows.

Setup. $\mathcal{B}$ randomly picks $b \in \{0, 1\}$, $s \in \mathbb{Z}_p$, set mpk to be (g, X_0, X_1, Y), where $X_b = g^x$, $X_{\bar{b}} = g^s$. The msk is $(Y_b = Y^x, Y_{\bar{b}} = Y^s)$, while Y_b is unknown to $\mathcal{B}$. From the perspective of the adversary $\mathcal{A}$ the distribution of the public parameters are identical to the real construction.

H_1**-queries**. At any time $\mathcal{A}$ can query the random oracle H_1. To respond to these queries $\mathcal{B}$ maintains a list of tuples $\langle \mathsf{ID}, v, w \rangle$ as explained below. We refer to this list as the L_1 list, which is initially empty. When $\mathcal{A}$ queries the oracle H_1 at a point $\hat{\mathsf{ID}}$ algorithm $\mathcal{B}$ responds as follows:

1. If $\hat{\mathsf{ID}}$ already appears on the L_1 list in a tuple $\langle \hat{\mathsf{ID}}, \hat{v}, \hat{w} \rangle$ then algorithm $\mathcal{B}$ responds with $H_1(\hat{\mathsf{ID}}) = g^{\hat{v}} Y^{\hat{w}} \in \mathbb{G}$.
2. Otherwise, $\mathcal{B}$ picks random $\hat{v} \in \mathbb{Z}_p$, $\hat{w} \in \{0, 1\}$ and adds the tuple $\langle \hat{\mathsf{ID}}, \hat{v}, \hat{w} \rangle$ to the L_1 list. $\mathcal{B}$ responds to $\mathcal{A}$ with $H_1(\hat{\mathsf{ID}}) = g^{\hat{v}} Y^{\hat{w}}$.

Note that $H_1(\mathsf{ID})$ is uniformly distributed over $\mathbb{G}$, and the bit $\hat{w}$ is perfectly hidden from $\mathcal{A}$'s view.

H_2**-queries**. $\mathcal{B}$ handles the queries to H_2 the obvious way, by producing a randomly sampled element from the appropriate codomain, and adding both query and answer to the L_2 list.

Phase 1 - Private key queries. Upon receiving the private key extraction query for an identity $\hat{\mathsf{ID}}$, $\mathcal{B}$ runs the above algorithm to obtain $H_1(\hat{\mathsf{ID}})$. Let $\langle \hat{\mathsf{ID}}, \hat{v}, \hat{w} \rangle$ be the corresponding tuple on the L_1 list, $\mathcal{B}$ randomly picks $r \in \mathbb{Z}_p$,

– If $\hat{w} = 0$, and constructs the private key using $msk_{\bar{b}} = Y_{\bar{b}}$ as

$$SK = (d_0, d_1, d_2) = (\bar{b}, Y_{\bar{b}} Q^r, g^r)$$

– Otherwise, $\mathcal{B}$ constructs the private key as follows: Let $\hat{r} = r - x$, we have

$$\begin{aligned} d_0 &= b \\ d_1 &= g^{\hat{v}r} X_b^{-\hat{v}} Y^r = Y^x (g^{\hat{v}} Y)^{r-x} = Y^x (H(\hat{\mathsf{ID}}))^{\hat{r}} = Y_b Q^{\hat{r}} \\ d_2 &= g^r X_b^{-1} = g^{r-x} = g^{\hat{r}} \end{aligned}$$

$d = (b, Y_b Q^{\hat{r}}, g^{\hat{r}})$ is a valid private key of ID for the real randomness $\hat{r}$.

Note that either way d is a valid private key of ID.

Phase 1 - Decryption queries. Upon receiving the decryption query $\langle C, \mathsf{ID} \rangle$, $\mathcal{B}$ decrypts M using the private key SK_{ID} normally. Note that $\mathcal{B}$ can generate the private key for any identity, thus it can answer all the decryption queries.

Challenge. $\mathcal{A}$ submits two messages M_0, M_1 and an identity ID where it wishes to be challenged. Suppose $\langle \mathsf{ID}^*, v^*, w^* \rangle$ is the corresponding entry on the L_1 list. If $w^* \neq 0$, $\mathcal{B}$ aborts and outputs a random element from $\mathbb{G}_T$ as the answer to the BDH challenge. Otherwise $\mathcal{B}$ generates the private key $SK_{\mathsf{ID}^*} = (\bar{b}, Y_{\bar{b}}Q^r, g^r)$ of ID^*, flips a fair coin $\beta \in \{0,1\}$, picks a random $\sigma^* \in \{0,1\}^n$, a random string $R \in \{0,1\}^n$, set the challenge ciphertext $C^* = (U^*, V^*, W_0^*, W_1^*, S^*)$ to be:

$$U^* = g^z, V^* = (g^z)^{v^*} = (Q^*)^z, W_b^* = \sigma^* \oplus R, W_{\bar{b}}^* = \sigma^* \oplus H_2(e(X_{\bar{b}}, Y)^z), S^* = M_\beta \oplus H_4(\sigma^*)$$

This operation implicitly sets $R = H_2(T)$, where $T = e(X_b, Y)^z$ is the solution to the BDH problem. Note that $V^* = (g^z)^{v^*} = (g^{v^*}Y^{w^*})^z = (Q^*)^z$ since $w^* = 0$. $\mathcal{B}$ obtains $e(X_{\bar{b}}, Y)^z$ via computing $e(d_1, U^*)/e(d_2, V^*)$ We remark that C^* is a not valid ciphertext. However, this does not affect the final result of security reduction.

Phase 2 - Private key queries. Handled the same way as Phase 1.

Phase 2 - Decryption queries. Handled the same way as Phase 1.

Guess. Finally, the adversary $\mathcal{A}$ outputs its guess β' for β. $\mathcal{B}$ randomly picks an entry $\langle T, R\rangle$ from the L_2 list and outputs T as its answer to the BDH problem.

Claim. The responses to H_1-queries, H_2-queries are as in the real attack. All responses to private key extraction queries and decryption queries are valid. The bit b is information-theoretically hidden from $\mathcal{A}$, therefore if $\mathcal{B}$ does not abort, the desired T will appear in some entry of the L_2 list with probability ϵ.
The probability that $\mathcal{B}$ aborts during the simulation is 1/2. This shows that $\mathcal{B}$'s advantage is at least $\epsilon/2Q_{h_2}$ as required. □

8 Further Discussions

Throughout this paper, all the security proofs follows the Type-2 style as previously remarked in the introduction part. We emphasize that Type-1 style proof can not be employed in the schemes characterized with Katz-Wang “double encryption” technique. Type-1 style proofs hold on the condition that when T is the right solution of $\mathcal{P}$ the adversary $\mathcal{A}$ has advantage ϵ to output the right β', otherwise its advantage is negligible since M_β is information-theoretically hidden from $\mathcal{A}$. After adopting the Katz-Wang technique (no matter whether combining with the Fujisaki-Okamoto transformation), the above condition does not hold anymore, because at least one part of the challenge is always a valid ciphertext (in the original scheme) of M_β, which is a natural result brought by the redundancy due to the Katz-Wang technique. In this situation the adversary's outputs $\beta' \in \{0,1\}$ could be totally independent of the challenge (suppose the adversary has the ability to “fully” decrypt the ciphertext, not just partially decrypt), thus the reduction fails.

Both our variant of BF-IBE presented in Section 5 and our variant of BB_1-IBE presented in Section 7 are proven secure based on the CBDH problem. Obviously, they can be tightly reduced to the decisional BDH (DBDH) problem or the gap BDH (GBDH) problem [16]. However, either DBDH assumption or GBDH assumption is stronger than CBDH assumption. An alternative approach to achieve tight security reduction without resorting to stronger assumptions is adapting the twin technique presented in [9].

We tabulate the efficiency of our variants described in 5 and 7, and compare them to the related schemes, in Table 1.

9 Conclusion

In this paper, we first presented a generic method based on the Katz-Wang technique which can greatly improve the security reductions for a category of IBE schemes. By employing the randomness reuse technique, we further proposed two refined approaches with respect to the

Scheme	Assumption	IND-ID-X	Ciphertext size	Reduction Cost	Encryption	Decryption
BF-IBE [6]	CBDH	CCA	$\|\mathbb{G}\| + \kappa + \|M\|$	$\mathcal{O}(1/Q_e Q_h)$	1P+2E	1P+1E
BF-IBE [6]+KW03 [15]	CBDH	CPA	$2\|\mathbb{G}\| + 2\|M\|$	$\mathcal{O}(1/Q_h)$	2P+2E	1P+1E
TightIBE [1]	CBDH	CCA	$\|\mathbb{G}\| + 2\kappa + \|M\|$	$\mathcal{O}(1/Q_h)$	2P+2E	2P+2E
BF-IBE [6]+Approach I	CBDH	CCA	$\|\mathbb{G}\| + 2\kappa + \|M\|$	$\mathcal{O}(1/Q_h)$	2P+2E	2P+2E
BB_1-IBE [4]	DBDH	CPA	$2\|\mathbb{G}\| + \kappa$	$\mathcal{O}(1/Q_h)$	3E	2P
BB_1-IBE [4]+KW03 [15]	DBDH	CPA	$4\|\mathbb{G}\| + 2\kappa$	$\mathcal{O}(1)$	6E	2P
BB_1-IBE [4]+Approach II	DBDH	CCA	$2\|\mathbb{G}\| + 2\kappa$	$\mathcal{O}(1)$	3E	2P+3E

We assume all the schemes are constructed on a symmetric pairing $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$. P denotes a pairing operation, and E denotes a group exponentiation in $\mathbb{G}$ or $\mathbb{G}_T$. κ is the security parameter. $|G|$ denotes the bits of an element of $\mathbb{G}$. If using point compression trick, we have $|\mathbb{G}| \approx \kappa$. There are also the computational costs due to the multiplication/inversion operations in group $\mathbb{G}$ or $\mathbb{G}_T$ and hash function/block cipher evaluations, but they can be done quite efficiently. For simplicity concern, we drop these terms in the table.

Table 1. Comparison between the related IBE Schemes

different constructions of the one-way function in the IBE schemes. Compared to the original schemes, the new schemes derived from them using the two refined approaches achieve tighter CCA reductions with the reasonable costs in ciphertext size and efficiency.

References

1. Attrapadung, N., Furukawa, J., Gomi, T., Hanaoka, G., Imai, H., Zhang, R.: Efficient Identity-Based Encryption with Tight Security Reduction. In: Cryptology and Network Security, 5th International Conference, CANS 2006. LNCS, vol. 4301, pp. 19–36 (2006)
2. Bellare, M., Ristenpart, T.: Simulation without the Artificial Abort: Simplified Proof and Improved Concrete Security for Waters' IBE Scheme. In: Advances in Cryptology - EUROCRYPT 2009. LNCS, vol. 5479, pp. 407–424 (2009)
3. Bellare, M., Rogaway, P.: Random Oracles are Practical: A Paradigm for Designing Efficient Protocols. ACM conference on Computers and Communication Security pp. 62–73 (1995)
4. Boneh, D., Boyen, X.: Efficient Selective-ID Secure Identity Based Encryption without Random Oracles. In: Advances in Cryptology - EUROCRYPT 2004. LNCS, vol. 3027, pp. 223–238 (2004)
5. Boneh, D., Franklin, M.: Identity-Based Encryption from the Weil Pairing. In: Advances in Cryptology - CRYPTO 2001. LNCS, vol. 2139, pp. 213–229 (2001)
6. Boneh, D., Franklin, M.K.: Identity-Based Encryption from the Weil Pairing. SIAM Journal on Computation 32, 586–615 (2003)
7. Boyen, X.: General ad hoc encryption from exponent inversion ibe. In: Advances in Cryptology - EUROCRYPT 2007. LNCS, vol. 4515, pp. 394–411 (2007)
8. Canetti, R., Halevi, S., Katz, J.: Chosen-ciphertext security from identity based encryption. In: Advances in Cryptology - Eurocrypt 2004. LNCS, vol. 3027, pp. 207–222 (2004)
9. Cash, D., Kiltz, E., Shoup, V.: The Twin Diffie-Hellman Problem and Applications. In: Advances in Cryptology - EUROCRYPT 2008. LNCS, vol. 4965, pp. 127–145 (2008)
10. Chen, L., Cheng, Z.: Security Proof of Sakai-Kasahara's Identity-Based Encryption Scheme. In: Cryptography and Coding, 10th IMA International Conference. LNCS, vol. 3796, pp. 442–459 (2005)
11. Fischlin, M., Lehmann, A., Ristenpart, T., Shrimpton, T., Stam, M., Tessaro, S.: Random Oracles With(out) Programmability. In: Advances in Cryptology - ASIACRYPT 2010. LNCS, vol. 6477, pp. 303–320 (2010)
12. Fujisaki, E., Okamoto, T.: Secure Integration of Asymmetric and Symmetric Encryption Schemes. In: Advances in Cryptology - CRYPTO 1999. LNCS, vol. 1666, pp. 537–554 (1999)
13. Gentry, C.: Practical Identity-Based Encryption Without Random Oracles. In: Advances in Cryptology - EUROCRYPT 2006. LNCS, vol. 4004, pp. 445–464 (2006)
14. Joux, A.: A One Round Protocol for Triparitite Diffie-Hellman. In: Algorithmic Number Theory, 4th International Symposium. ANTS-IV, vol. 1838, pp. 385–394 (2000)
15. Katz, J., Wang, N.: Efficiency improvements for signature schemes with tight security reductions. In: ACM Conference on Computer and Communications Security, CCS 2003. pp. 155–164 (2003)
16. Okamoto, T., Pointcheval, D.: The gap-problems: A new class of problems for the security of cryptographic schemes. In: Public Key Cryptography, 4th International Workshop on Practice and Theory in Public Key Cryptography, PKC 2001. LNCS, vol. 1992, pp. 104–118 (2001)

17. Sakai, R., Kasahara, M.: ID based Cryptosystems with Pairing on Elliptic Curve. Cryptology ePrint Archive, Report 2003/054 (2003), http://eprint.iacr.org/
18. Sakai, R., Ohgishi, K., Kasahara, M.: Cryptosystems based on pairing. The 2001 Symposium on Cryptography and Information Security, Japan 45, 26–28 (2001)
19. Waters, B.: Dual System Encryption: Realizing Fully Secure IBE and HIBE under Simple Assumptions. In: Advances in Cryptology - CRYPTO 2009. LNCS, vol. 5677, pp. 619–636 (2009)