Efficient Identity-Based Encryption with Tight Security Reduction

Nuttapong Attrapadung[1], Benoit Chevallier-Mames[2], Jun Furukawa[3],
Takeshi Gomi[1], Goichiro Hanaoka[4], Hideki Imai[1,4], and Rui Zhang[1]

[1] Institute of Industrial Science, University of Tokyo.
{nuts,takego,zhang}@imailab.iis.u-tokyo.ac.jp
imai@iis.u-tokyo.ac.jp

[2] Gemplus and Ecole Normale Supérieure, France,
benoit.chevallier-mames@gemplus.com

[3] NEC Corporation.
j-furukawa@ay.jp.nec.com

[4] Research Center for Information Security,
National Institute of Advanced Industrial Science and Technology.
hanaoka-goichiro@aist.go.jp

Abstract. In a famous paper of Crypto'01, Boneh and Franklin proposed the first identity-based encryption scheme (IBE), around fifteen years after the concept was introduced by Shamir. Their scheme security (more precisely, the notion of resistance against an IND-ID-CCA attacker) relies in the random oracle model. However, the reduction is far from being tight, and notably depends on the number of extractions queries.

In this paper, we present an efficient modification to the Boneh-Franklin scheme that provides a tight reduction. Our scheme is basically an IBE under two keys, one of which is (randomly) detained by the recipient. It can be viewed as a continuation of an idea introduced by Katz and Wang; we will however show how our construction improves this last scheme.

Our scheme features a tight reduction to the list bilinear Diffie-Hellman (LBDH) problem, which can be itself reduced tightly either to the gap bilinear Diffie-Hellman (GBDH) or the decisional bilinear Diffie-Hellman (DBDH) problems. Furthermore, for a relaxed notion of tightness (called *weak-tightness*) that we introduce and discuss in our paper, we show that there is a weakly tight reduction from our scheme to the computational bilinear Diffie-Hellman (CBDH) problem.

Our scheme is very efficient, as one can precompute most of the quantity involved in the encryption process. Furthermore, the ciphertext size is very short: for proposed parameters, they are $|M| + 330$ bits long.

keywords: ID-based encryption, tight security reduction

1 Introduction

Identity Based Encryption (IBE) provides a public key encryption mechanism where an arbitrary string, such as recipient's identity, can be served as a public key. The ability to use identities as public keys avoids the need to distribute public key certificates. Such a scheme is largely motivated by many applications such as to encrypt emails using recipient's email address or to encrypt messages for users that have not their proper key at the given moment.

Although the concept of identity based encryption was proposed two decades ago [14], it is only recently that the first fully functional schemes were proposed. Boneh and Franklin [3, 4] defined a security model namely IND-ID-CCA and gave the first efficient construction provably secure in the random oracle model based on the bilinear Diffie-Hellman (BDH) problem. A few years after, new schemes were shown to be secure without random oracles, but in a weaker model of security known as "Selective-ID" model [5, 1]. Such schemes in this weaker model are known to be secure also in the sense of IND-ID-CCA, but the proofs use an inefficient security reduction [1], which degrades reduction costs by a factor of the size of identities' space, which is indeed not polynomial in the security parameter. Boneh and Boyen [2] subsequently proposed the first scheme which is provably secure in the sense of IND-ID-CCA with a polynomial time reduction in the absence of random oracles, which was then simplified and improved by Waters [17].

However, for each of the above schemes, the security as in the sense of IND-ID-CCA is reduced only *loosely* to its underlying intractability assumption. An inefficient security reduction would imply either a lower security level or the requirement of larger key and ciphertext sizes to obtain the same security level.

It has been an open problem (as already posed in [17, 7]) whether efficient IBE systems can exist with their security in the sense of IND-ID-CCA being reduced *tightly* (*i.e.,* the factor between the difficulty of the underlying problem and the security of the scheme being only a constant term, as close to 1 as possible) to some reasonable intractability assumption. In the standard model, this problem is still open.

In the random oracle model, however, it has been partially solved by Katz and Wang [10]. However, their idea was just mentioned at the end of one of their papers and regarding a different subject, *i.e.,* the signature schemes, and so, some thoughts were let to the reader.

Our Contribution. In this paper, we remind identity-based encryption schemes of Boneh and Franklin, and of Katz and Wang. We show notably how the Katz and Wang solution does not achieve tight IND-ID-CCA security, even when used with the generic Fujisaki-Okamoto [6] transform. Then, we present our principle result, which is a new IBE scheme with a tight reduction to the list bilinear Diffie-Hellman (LBDH) problem. We also show how this problem can itself be tightly reduced to the gap bilinear Diffie-Hellman (GBDH) problem or the decisional bilinear Diffie-Hellman (DBDH) problem. Another point that we address is a relaxed definition of tightness (called *weak-tightness*), that we introduce and discuss; we then show that there is a weakly-tight reduction from our scheme to the computational bilinear Diffie-Hellman (CBDH) problem.

Our scheme is very efficient, as one can precompute most of the quantity during the encryption process, before knowing the message. Furthermore, the ciphertext size is very short: for proposed parameters, they are $|M| + 330$ bit long, which is comparable with the Boneh-Franklin IBE (whose ciphertexts are $|M| + 250$ bit long, for a *loose* reduction).

Outlines. Our paper is organized as follows: we begin in Section 2 with some definitions. Then, in Section 3, we remind the idea of Katz and Wang (which is itself a variant of the Boneh-Franklin IBE), and show how it allows a tight reduction from IND-ID-CPA attackers. However, we point out that without any additional construction step, the reduction does not succeed against IND-ID-CCA attackers. In Section 4, we introduce our new identity-based encryption scheme, and show how it achieves IND-ID-CCA security, with tight reduction. In Section 5, we show in fact our scheme is *weakly* reducible (more precise discussion given later) to the CBDH problem. In Section 6, we compare our scheme with existing ones. Finally, we conclude our work.

2 Definitions

We review the model and the security notion of an IBE scheme, the length-preserving IND-CCA symmetric key encryption, as well as the definitions of bilinear maps and related problems. We also discuss two flavors of tightness.

2.1 ID-Based Encryption

An IBE scheme $\mathcal{E}$ consists of four polynomial-time algorithms:

Setup: takes a security parameter k and returns params (system parameters) and master-key. The system parameters include a description of a finite message space $\mathcal{M}$, and a description of a finite ciphertext space $\mathcal{C}$. Intuitively, the system parameters will be publicly known, while the master-key will be known only to the private key generator.

Extract: takes as input params, master-key, and an arbitrary $\mathsf{ID} \in \{0,1\}^*$, and returns a private key sk. Here ID is an arbitrary string that will be used as a public key, and sk is the corresponding private decryption key. The Extract algorithm extracts a private key from the given public key.

Encrypt: takes as input params, ID, and $M \in \mathcal{M}$. It returns a ciphertext $C \in \mathcal{C}$.

Decrypt: takes as input params, $C \in \mathcal{C}$, and a private key sk. It returns $M \in \mathcal{M}$ or *“reject”*, which is a special symbol not in $\mathcal{M}$.

These algorithms must satisfy the standard consistency constraint; that is, if $(\mathsf{params}, \mathsf{master\text{-}key}, \mathcal{M}, \mathcal{C}) \leftarrow \mathbf{Setup}(1^k)$, then for all $M \in \mathcal{M}$ and for all ID, $M = \mathbf{Decrypt}(\mathsf{params}, \mathbf{Encrypt}(\mathsf{params}, \mathsf{ID}, M), \mathbf{Extract}(\mathsf{params}, \mathsf{master\text{-}key}, \mathsf{ID}))$.

Security Notion. The strongest security definition for IBE is chosen ciphertext security for IBE under a chosen identity attack ($\mathsf{IND\text{-}ID\text{-}CCA}$) [3, 4]. In this model, the adversaries are allowed to collude (chosen ID attack) and to access a decryption oracle. We first review the $\mathsf{IND\text{-}ID\text{-}CCA}$ game:

***Setup*:** The challenger takes a security parameter k and runs the **Setup** algorithm. It gives the adversary the resulting system parameters params. It keeps the $\mathsf{master\text{-}key}$ to itself.

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

- Extraction query $\langle \mathsf{ID}_i \rangle$: The challenger responds by running algorithm **Extract** to generate the private key sk_i corresponding to the public key $\langle \mathsf{ID}_i \rangle$. It sends sk_i to the adversary.
- Decryption query $\langle \mathsf{ID}_i, C_i \rangle$: The challenger responds by running algorithm **Extract** to generate the private key sk_i corresponding to ID_i. It then runs algorithm **Decrypt** to decrypt the ciphertext C_i using the private key sk_i. It sends the result 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 Extraction query in Phase 1. The challenger picks a random bit $\beta \in \{0, 1\}$, sets $C^* = \mathbf{Encrypt}(\mathsf{Params}, \mathsf{ID}^*, M_\beta)$, and sends C^* to the adversary.

***Phase 2*:** The adversary issues more queries $q_{m+1}, \cdots, q_{max}$ adaptively where each query is one of:

- Extraction query $\langle \mathsf{ID}_i \rangle$ where $\mathsf{ID}_i \neq \mathsf{ID}^*$: challenger responds as before.
- Decryption query $\langle \mathsf{ID}_i, C_i \rangle \neq \langle \mathsf{ID}^*, C^* \rangle$: challenger responds as before.

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

We define adversary $\mathcal{A}$'s advantage in attacking the scheme $\mathcal{E}$ as $\mathsf{AdvIBE}_{\mathcal{E}}(\mathcal{A}) = |\Pr[\beta = \beta'] - 1/2|$. We say that $\mathcal{A}$ is an (ϵ, t)-$\mathsf{IND\text{-}ID\text{-}CCA}$ adversary if $\mathsf{AdvIBE}_{\mathcal{E}}(\mathcal{A}) \geq \epsilon$ and its running time is at most t. We say that an IBE scheme $\mathcal{E}$ is (ϵ, t)-$\mathsf{IND\text{-}ID\text{-}CCA}$ secure if there exists no (ϵ, t)-$\mathsf{IND\text{-}ID\text{-}CCA}$ adversary.

2.2 Bilinear Maps

We briefly review several facts about bilinear maps. Throughout this paper, we let $\mathbb{G}_1$ and $\mathbb{G}_2$ be two multiplicative cyclic groups of prime order q and g be a generator of $\mathbb{G}_1$. A *bilinear map* $e : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ satisfies the following properties: (i) *Bilinearity*: For all $u, v \in \mathbb{G}_1$ and $a, b \in \mathbb{Z}$, $e(u^a, v^b) = e(u, v)^{ab}$. (ii) *Non-degeneracy*: $e(g, g) \neq 1$. (iii) *Computability*: There is an efficient algorithm to compute $e(u, v)$ for any $u, v \in \mathbb{G}_1$.

2.3 Underlying Hard Problems

We review hard problems related to bilinear maps which are those variants of *bilinear Diffie-Hellman* (BDH) problems: the *computational BDH* (CBDH) [3], the *list BDH* (LBDH), the *decisional BDH* (DBDH) [5], and the *gap BDH* (GBDH) [12] problems.

CBDH and LBDH Problems. The *ℓ-LBDH problem* is defined as follows: given a tuple $(g, g^a, g^b, g^c) \in (\mathbb{G}_1)^4$ as input, output a list $\mathcal{L}$ of length at most ℓ ($\ell \geq 1$) which contains $T \ \in \mathbb{G}_2$ such that $T = e(g, g)^{abc}$. Especially, 1-LBDH problem is referred to as the *CBDH problem.* We say that $\mathcal{A}$ is a (ϵ, t)-ℓ-LBDH algorithm if it runs with time at most t and outputs a list $\mathcal{L}$ of length at most ℓ which contains $T = e(g, g)^{abc}$ with probability at least ϵ, that is,

$$\Pr[\mathcal{A}(g, g^a, g^b, g^c) = \mathcal{L} \ \wedge \ e(g, g)^{abc} \in \mathcal{L} \ \wedge \ |\mathcal{L}| \leq \ell] \geq \epsilon,$$

where $|\mathcal{L}|$ denotes the number of elements of $\mathcal{L}$ and the probability is taken over the random choice of generator $g \in \mathbb{G}_1^*$, the random choice of $a, b, c \in \mathbb{Z}_q$, and random coins consumed by $\mathcal{A}$.

DBDH Problem. The *DBDH problem* is defined as follows: given a tuple $(g, g^a, g^b, g^c, T) \in (\mathbb{G}_1)^4 \times \mathbb{G}_2$ as input, outputs a bit $\beta \in \{0, 1\}$. We say that $\mathcal{A}$ is a (ϵ, t)-DBDH algorithm if it runs with time at most t, and distinguishes the BDH-tuple with advantage at least ϵ, that is,

$$\left|\Pr[\mathcal{A}(g, g^a, g^b, g^c, e(g, g)^{abc}) = 0] - \Pr[\mathcal{A}(g, g^a, g^b, g^c, T) = 0]\right| \geq \epsilon,$$

where the probability is taken over the random choice of generator $g \in \mathbb{G}_1^*$, the random choice of $a, b, c \in \mathbb{Z}_q$, the random choice of T in $\mathbb{G}_2$, and the random coins consumed by $\mathcal{A}$.

GBDH Problem. The *GBDH problem* is defined as follows: given a tuple $(g, g^a, g^b, g^c) \in (\mathbb{G}_1)^4$ as input, output $e(g, g)^{abc} \in \mathbb{G}_2$ with the help of a DBDH oracle $\mathcal{O}$ which for given $(g, g^a, g^b, g^c, T) \in (\mathbb{G}_1)^4 \times \mathbb{G}_2$, answers "*true*" if $T = e(g, g)^{abc}$, or "*false*" otherwise [12]. We say that $\mathcal{A}$ is a (ϵ, t)-GBDH algorithm if it runs with time at most t and succeeds in outputting $e(g, g)^{abc}$ with probability at least ϵ, that is,

$$\Pr[\mathcal{A}^{\mathcal{O}}(g, g^a, g^b, g^c) = e(g, g)^{abc}] \geq \epsilon,$$

where the probability is taken over the random choice of generator $g \in \mathbb{G}_1^*$, the random choice of $a, b, c \in \mathbb{Z}_q$, and random coins consumed by $\mathcal{A}$.

2.4 IND-CCA Length Preserving Symmetric Key Encryption

A (deterministic) symmetric key encryption (SKE) scheme $\mathcal{E} = (\text{Enc}, \text{Dec})$ contains two algorithms: an encryption algorithm $\text{Enc} : \mathcal{K} \times \mathcal{M} \to \mathcal{C}$ and a decryption algorithm $\text{Dec} : \mathcal{K} \times \mathcal{C} \to \mathcal{M}$, where $\mathcal{K}$, $\mathcal{M}$ and $\mathcal{C}$ are the spaces of keys, plaintexts and ciphertexts, respectively. Two algorithms are conform to the standard consistency constraint: for all $K \in \mathcal{K}, M \in \mathcal{M}$, $M = \text{Dec}(K, \text{Enc}(K, M))$. Moreover, if for all $K \in \mathcal{K}, M \in \mathcal{M}$, $|\text{Enc}(K, M)| = |M|$ then we say that $\mathcal{E}$ is *length preserving*. We often let $\text{Enc}_K(\cdot)$ denote $\text{Enc}(K, \cdot)$.

Security Notion. A challenger plays the following game with an adversary $\mathcal{A}$: The challenger randomly chooses a key $K \in \mathcal{K}$ and a bit γ. $\mathcal{A}$ is given access to two oracles $\mathrm{Enc}_K(\cdot)$ and $\mathrm{Dec}_K(\cdot)$. A chooses a pair (M_0, M_1) in M of the same length that were not submitted to $\mathrm{Enc}_K(\cdot)$ or obtained from $\mathrm{Dec}_K(\cdot)$, submits to $\mathrm{Enc}_K(\cdot)$ and gets $C^* = \mathrm{Enc}_K(M_\gamma)$. $\mathcal{A}$ can further query the oracles as before but is not allowed to ask $\mathrm{Dec}_K(C^*)$, $\mathrm{Enc}_K(M_0)$ or $\mathrm{Enc}_K(M_1)$. Finally, $\mathcal{A}$ outputs a bit γ'. The advantage of $\mathcal{A}$ is defined by $\mathsf{AdvSKE}_{\mathcal{E}}(\mathcal{A}) = |\Pr[\gamma = \gamma'] - 1/2|$. We say that $\mathcal{A}$ is an (ϵ, t)-IND-CCA adversary if $\mathsf{AdvSKE}_{\mathcal{E}}(\mathcal{A}) \geq \epsilon$ and it runs in time at most t. We say that a SKE scheme $\mathcal{E}$ is (ϵ, t)-IND-ID-CCA secure if there exists no (ϵ, t)-IND-CCA adversary.

We will use a length preserving IND-CCA-secure SKE in our construction.[‡] Such a scheme can be built, for example, by applying CMC [8] or EME [9] mode of operation to a block cipher, if the underlying block cipher is modeled as (strong) pseudorandom permutation, e.g. AES. Though the above formulation of IND-CCA security differs from that of [8], one can show by some standard arguments that it is implied by the definition given in [8].

2.5 On the Notions of Tight Reduction

Informally, we say that the security of a scheme can be reduced to an underlying problem *tightly in the conventional sense* if, there exists a $t_{\mathcal{B}}$-time algorithm $\mathcal{B}$ who can solve the underlying problem with the probability $\epsilon_{\mathcal{B}}$ when there exists a $t_{\mathcal{A}}$-time adversary $\mathcal{A}$ who can break the scheme with the probability $\epsilon_{\mathcal{A}}$, where both $\epsilon_{\mathcal{A}} \simeq \epsilon_{\mathcal{B}}$ and $t_{\mathcal{A}} \simeq t_{\mathcal{B}}$ hold.

In addition to such conventional definition of tightness, we also propose a definition of relaxed tightness. We say that the security of a scheme can be reduced to an underlying problem *tightly in the weak sense* if, there exists a $t_{\mathcal{B}}$-time algorithm $\mathcal{B}$ who can solve the underlying problem with the probability $\epsilon_{\mathcal{B}}$ when there exists a $t_{\mathcal{A}}$-time adversary $\mathcal{A}$ who can break the scheme with the probability $\epsilon_{\mathcal{A}}$, where $t_{\mathcal{B}}/e_{\mathcal{B}} \simeq t_{\mathcal{A}}/\epsilon_{\mathcal{A}}$ holds. If this condition holds, we have that the expected running time of $\mathcal{A}$ is roughly the same as $\mathcal{B}$. This is the intuition as to why we consider this kind of reduction as weakly tight. Similar notion was also considered by Pointcheval and Stern in [13].

In this paper, our main result shows that the security of our scheme can be reduced tightly in the conventional sense to standard hard problems, namely, the LBDH problem, and also to the GBDH problem and the DBDH problem. As an independent interest, we also show a tight reduction from the LBDH problem to the CBDH problem in the weak sense. Thus the security of our scheme can be tightly reduced to the problem (also in the weak sense). This observation brings more confidence to the security of our scheme.

[‡] Indeed our scheme does not need the full power of the IND-CCA-secure SKE. More precisely, as it will become clear, we do not need the encryption oracles at all.

3 Boneh-Franklin IBE and Its Katz-Wang Variant

In this section, we remind the construction of Boneh and Franklin, and its variant by Katz and Wang.

3.1 Boneh-Franklin Identity Based Encryption

The Boneh-Franklin [3, 4] ID-based encryption scheme (more precisely, its basic variant) is defined in Table 1. In Tables 1 and 2, M denotes a plaintext, $G : \mathbb{G}_2 \rightarrow \{0,1\}^n$ and $H : \{0,1\}^* \rightarrow \mathbb{G}_1$ denote random oracles. We refer to [3, 4] for a more precise study of its security. In this subsection, we just remind that the basic version of the Boneh-Franklin IBE is IND-ID-CPA secure, while using Fujisaki-Okamoto [6] transform, one gets the full version of the Boneh-Franklin IBE, which is IND-ID-CCA secure. All these reductions are in the random oracle model.

The Boneh-Franklin Identity Based Encryption	
Setup (1^k)**:** $s \leftarrow \mathbb{Z}_q^*$; $g_{pub} := g^s$ $\mathsf{params} := \langle q, \mathbb{G}_1, \mathbb{G}_2, e, n, g, g_{pub}, G, H \rangle$ $\mathsf{master\text{-}key} := s$ $return$ $(\mathsf{params},\ \mathsf{master\text{-}key})$	**Extract** $(\mathsf{ID},\ \mathsf{params},\ \mathsf{master\text{-}key})$: $h_{\mathsf{ID}} := H(\mathsf{ID})$ $d_{\mathsf{ID}} := (h_{\mathsf{ID}})^s$ $return\ d_{\mathsf{ID}}$
Encrypt $(\mathsf{ID}, \mathsf{params}, M)$**:** $h_{\mathsf{ID}} := H(\mathsf{ID})$; $\quad r \leftarrow \mathbb{Z}_q^*$ $w := e(g_{pub}, h_{\mathsf{ID}})^r$ $C := \langle g^r,\ G(w) \oplus M \rangle$ $return\ C$	**Decrypt** $(C, \mathsf{params}, d_{\mathsf{ID}})$**:** parse $C = \langle u, V \rangle$ $w' := e(u, d_{\mathsf{ID}})$ $M := V \oplus G(w')$ $return\ M$

Table 1. The Boneh-Franklin Identity Based Encryption

Unfortunately, the reduction of the Boneh-Franklin IBE scheme is very loose, as there is a factor equal to the number of extract queries that an attacker can make, between the security of the underlying problem (*i.e.,* the CBDH) and the security of the scheme. Roughly, this factor is due to the fact that the reduction must *guess* which of the identity will be used in the challenge, as for this special identity, it must return a special H output, while for other identities, it must return another type of H output, to be able to answer extract queries.

3.2 Katz and Wang's Variant of Boneh-Franklin IBE

This problem of tightness of IBE has been partially solved by Katz and Wang, at the end of a paper [10] whose subject was quite different. Hence, these authors only gave few points of their ideas, and let the rest to the reader. In this subsection, we explain what we believe that Katz and Wang meant, even if we might be subject to errors in the interpretation.

Katz and Wang proposed that, for each identity, there should be two corresponding public keys: instead of using $H(\mathsf{ID})$ as in the Boneh-Franklin, they proposed to use both $H(\mathsf{ID}, 0)$ and $H(\mathsf{ID}, 1)$. However, only one of the corresponding private key is known to the designator. With this trick, the reduction does not need to guess which of the identity will be used in the challenge: for each identity, one of the two hash output (let say the one with bit b_{ID}) is controlled in order the simulator to be able to answer to extract queries, while the other is let to be used in case the identity is the one that appears in the challenge. Hence, for the identity $\mathsf{ID}^\star$ of the challenge, if the bit $b_{\mathsf{ID}^\star}$ is absolutely indistinguishable to the attacker, with a chance of one half, $H(\mathsf{ID}^\star, \bar{b}_{\mathsf{ID}^\star})$ will be used by the attacker and the simulator will succeed in solving the underlying problem.

More precisely, the idea of Katz and Wang is depicted in the Table 2.

The Katz-Wang Identity Based Encryption	
Setup (1^k)**:** $s \leftarrow \mathbb{Z}_q^*$; $g_{pub} := g^s$ $\mathsf{params} := \langle q, \mathbb{G}_1, \mathbb{G}_2, e, n, g, g_{pub}, G, H\rangle$ $\mathsf{master\text{-}key} := s$ *return* $(\mathsf{params}, \mathsf{master\text{-}key})$	**Extract**† $(\mathsf{ID}, \mathsf{params}, \mathsf{master\text{-}key})$: $b_{\mathsf{ID}} \leftarrow \{0, 1\}$ $h_{\mathsf{ID}} := H(\mathsf{ID}, b_{\mathsf{ID}})$ $d_{\mathsf{ID}} := (h_{\mathsf{ID}})^s$ $sk_{\mathsf{ID}} := (d_{\mathsf{ID}}, b_{\mathsf{ID}})$ *return* sk_{ID}
Encrypt $(\mathsf{ID}, \mathsf{params}, M)$**:** $h_{\mathsf{ID},0} := H(\mathsf{ID}, 0)$ $h_{\mathsf{ID},1} := H(\mathsf{ID}, 1)$ $r_0 \leftarrow \mathbb{Z}_q^*$ $r_1 \leftarrow \mathbb{Z}_q^*$ $w_0 := e(g_{pub}, h_{\mathsf{ID},0})^{r_0}$ $w_1 := e(g_{pub}, h_{\mathsf{ID},1})^{r_1}$ $C := \langle g^{r_0},\ G(w_0) \oplus M, g^{r_1},\ G(w_1) \oplus M\rangle$ *return* C	**Decrypt** $(C, \mathsf{params}, b_{\mathsf{ID}}, d_{\mathsf{ID}})$**:** parse $C = \langle u_0, V_0, u_1, V_1\rangle$ $w' := e(u_{b_{\mathsf{ID}}}, d_{\mathsf{ID}})$ $M := V_{b_{\mathsf{ID}}} \oplus G(w')$ *return* M

†**Extract** first checks to see if sk_{ID} has been generated before. If it has, the previously-generated sk_{ID} is output.

Table 2. The Katz-Wang Identity Based Encryption

A disadvantage of this scheme is its cost: roughly, the Katz-Wang IBE ciphertexts are twice as much as in the Boneh-Franklin IBE, and the encryption process is twice longer (*i.e.,* two exponentiations and two pairing computations).

From [10], the security of this scheme against IND-ID-CPA can be *tightly* reduced to the Gap Bilinear Diffie-Hellman problem. Unfortunately, the use of Fujisaki-Okamoto [6] transform for this scheme is unclear. Katz and Wang did not explain how to achieve a tight IND-ID-CCA security with their scheme.

More precisely, to achieve ID-CCA security (either OW-ID-CCA or IND-ID-CCA), it is necessary that during the decryption, the user can test the equality of the messages in the two parts of the ciphertext. Else, the adversary would get a challenge $C = \langle u_0, V_0, u_1, V_1\rangle$ (of a message M that he wants to recover), and create another valid ciphertext $C'_0 = \langle u_0, V_0, u_2, V_2\rangle$ or $C'_1 = \langle u_2, V_2, u_1, V_1\rangle$, depending on a random bit b: for this, he takes a random message M_2, picks $r_2 \leftarrow \mathbb{Z}_q^*$, and computes $w_2 = e(g_{pub}, h_{\mathsf{ID},b})^{r_2}$, $u_2 = g^{r_2}$ and $v_2 = G(w_2) \oplus M_2$. With overwhelming probability, M_2 is not equal to M. Then, by querying the decryption of C'_b to the simulator or the legitimate user, with probability $\frac{1}{2}$, the adversary would learn the message M.

We now conclude the above discussion. On one hand, the technique of double encryption in which exactly one key for each ID is known by the simulator enables the simulation of the key exposure oracle and results in tight security reduction. On the other hand, this very technique itself also allows the CCA adversary to successfully break the scheme.[§] This contradictory implication of straightforward application of the Katz-Wang technique suggests that more sophisticated techniques are needed.

In our scheme, we propose a solution to these problems: namely, our scheme features a tight IND-ID-CCA security; furthermore, our scheme is roughly as efficient as the Boneh-Franklin scheme in term of ciphertext size, and in term of encryption and decryption timing. Our scheme is the subject of the next section.

4 Our IBE Scheme

4.1 Proposed Scheme (TightIBE)

Let k be a given security parameter. Let $\mathbb{G}_1$ and $\mathbb{G}_2$ be two groups of order q (which is a k-bit prime number) and g be a generator of $\mathbb{G}_1$. Let $e : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ be a bilinear map. Let $\mathcal{E} = (\mathrm{Enc}, \mathrm{Dec})$ be a SKE that the key space is $\mathcal{K}$ and the message space is $\mathcal{M}$. Let $G, H, \hat{H}$ be cryptographic hash functions $G : \{0,1\}^* \to \{0,1\}^{k_1}$ for some k_1, $H : \{0,1\}^* \to \mathbb{G}_1$, $\hat{H} : \{0,1\}^* \to \mathbb{Z}_q \times \mathcal{K}$ respectively. The TightIBE scheme consists of the four algorithms which are shown in Table 3.

4.2 Security

Theorem 1. *Suppose that the hash functions $G, H, \hat{H}$ are random oracles. Suppose there exists an (ϵ_{ibe},t_{ibe})-IND-ID-CCA adversary $\mathcal{A}$ against TightIBE. Suppose $\mathcal{A}$ makes at most q_G G-queries, q_H H-queries, $q_{\hat{H}}$ $\hat{H}$-queries, q_D decryption queries, and q_E extraction queries. Suppose that $\mathcal{E}$ is an (ϵ_{sym},t_{sym})-IND-CCA secure SKE. Then there exists an (ϵ_{lbdh},t_{lbdh})-($q_G + q_D$)-LBDH algorithm where*

$$\epsilon_{lbdh} \geq \frac{1}{2}\epsilon_{ibe} - \epsilon_{sym} - \frac{q_{\hat{H}}}{2^{k_1+1}},$$
$$t_{lbdh} \leq t_{ibe} + (3q_H + q_G + 3q_E + 10q_D)\tau + q_{\hat{H}}\tau' + q_D t_{sym},$$

where, τ is the maximum time among times for computing an exponentiation in $\mathbb{G}_1, \mathbb{G}_2$ and pairing e, and τ' is the time for responding to an $\hat{H}$-query.

Proof. The proof is provided by a sequence of games. Let $(g, g_1 = g^a, g_2 = g^b, g_3 = g^c)$ be a random instance of the LBDH problem, for which we do not know a, b, c.

[§] Even if combined with the Fujisaki-Okamoto [6] transform that was used to ensure full security of the Boneh-Franklin IBE, one can not ensure the equality of messages.

TightIBE	
Setup (1^k)**:** $s \leftarrow \mathbb{Z}_q^*$; $g_{pub} := g^s$ $\mathsf{params} := \langle q, \mathcal{G}_1, \mathcal{G}_2, e, n,$ $g, g_{pub}, G, H, \hat{H} \rangle$ $\mathsf{master\text{-}key} := s$ *return* $(\mathsf{params}, \mathsf{master\text{-}key})$	**Extract**† $(\mathsf{ID}, \mathsf{params}, \mathsf{master\text{-}key})$: $b_{\mathsf{ID}} \leftarrow \{0, 1\}$ $h_{\mathsf{ID}, b_{\mathsf{ID}}} := H(\mathsf{ID}, b_{\mathsf{ID}})$; $d_{\mathsf{ID}} := (h_{\mathsf{ID}, b_{\mathsf{ID}}})^s$ $sk_{\mathsf{ID}} := (d_{\mathsf{ID}}, b_{\mathsf{ID}})$ *return* sk_{ID}
Encrypt $(\mathsf{ID}, \mathsf{params}, M)$**:** $h_{\mathsf{ID},0} := H(\mathsf{ID}, 0)$ $h_{\mathsf{ID},1} := H(\mathsf{ID}, 1)$ $R \leftarrow \{0, 1\}^{k_1}$ $r \Vert K := \hat{H}(R, \mathsf{ID})$ $w_0 := e(g_{pub}, h_{\mathsf{ID},0})^r$ $w_1 := e(g_{pub}, h_{\mathsf{ID},1})^r$ $u := g^r$ $V_0 := G(w_0, \mathsf{ID}, 0) \oplus R$ $V_1 := G(w_1, \mathsf{ID}, 1) \oplus R$ $\alpha := \mathrm{Enc}_{\mathrm{K}}(M)$ $C := \langle u, V_0, V_1, \alpha \rangle$ *return* C	**Decrypt** $(C, \mathsf{params}, sk_{\mathsf{ID}})$**:** parse $C = \langle u, V_0, V_1, \alpha \rangle$ $w'_{b_{\mathsf{ID}}} := e(u, d_{\mathsf{ID}})$ $R_{b_{\mathsf{ID}}} := V_{b_{\mathsf{ID}}} \oplus G(w'_{b_{\mathsf{ID}}}, \mathsf{ID}, b_{\mathsf{ID}})$ $r' \Vert K := \hat{H}(R_{b_{\mathsf{ID}}}, \mathsf{ID})$ $w'_{\bar{b}_{\mathsf{ID}}} := e(g_{pub}, h_{\mathsf{ID}, \bar{b}_{\mathsf{ID}}})^{r'}$ $R_{\bar{b}_{\mathsf{ID}}} := V_{\bar{b}_{\mathsf{ID}}} \oplus G(w'_{\bar{b}_{\mathsf{ID}}}, \mathsf{ID}, \bar{b}_{\mathsf{ID}})$ if $R_{b_{\mathsf{ID}}} \neq R_{\bar{b}_{\mathsf{ID}}} \vee u \neq g^{r'}$ *return* "*reject*" else $M := D_K(\alpha)$ *return* M

†**Extract** first checks to see if sk_{ID} has been generated before. If it has, the previously-generated sk_{ID} is output.

Table 3. The algorithms of TightIBE

Game $\mathbf{G}_0$*:* This is the real IND-ID-CCA game. We denote by S_0 the event that $\beta' = \beta$ and use a similar notation S_i in any $\mathbf{G}_i$ below. By definition, we have

$$\Pr[\mathsf{S}_0] = \frac{1}{2} + \epsilon_{ibe}.$$

Game $\mathbf{G}_1$*:* In this game, one makes classical simulation of the random oracles, with random answers for any new query, as shown in Figure 1. Moreover, it maintains the evaluation of b_{ID} for each ID by randomly choosing from $\{0, 1\}$ for the first-time evaluation and using the same value after that. This game is clearly identical to the previous one, hence $\Pr[\mathsf{S}_0] = \Pr[\mathsf{S}_1]$.

Game $\mathbf{G}_2$*:* In this game, we change the simulation of the H-oracle:

▶ **Rule** $\mathsf{H}^{(2)}$

- If $b = b_{\mathsf{ID}}$, then randomly choose $\pi_{\mathsf{ID}} \in_R \mathbb{Z}_q$ and set $h = g^{\pi_{\mathsf{ID}}}$. Record $(\mathsf{ID}, b_{\mathsf{ID}}, \pi_{\mathsf{ID}}, h)$ in the H-list;
- Else, randomly choose $\tau_{\mathsf{ID}} \in_R \mathbb{Z}_q$ and set $h = g_2^{\tau_{\mathsf{ID}}}$. Record $(\mathsf{ID}, \bar{b}_{\mathsf{ID}}, \tau_{\mathsf{ID}}, h)$ in the H-list.

The two games $\mathbf{G}_1$ and $\mathbf{G}_2$ are perfectly indistinguishable: $\Pr[\mathsf{S}_1] = \Pr[\mathsf{S}_2]$.

Simulation

G, H, $\hat{\mathsf{H}}$ oracles

Query $G(w, \mathsf{ID}, b)$: if a record (w, ID, b, g) appears in the G-list, the answer is g. Otherwise g is chosen randomly in $\{0,1\}^{k_1}$ and the record (w, ID, b, g) is added in the G-list.

Query $H(\mathsf{ID}, b)$: if a record $(\mathsf{ID}, b, *, h)$ appears in the H-list, the answer is h. Otherwise do the following.

▶ **Rule** $\mathsf{H}^{(1)}$

| The answer h is chosen randomly in $\mathcal{G}_1$ and the record $(\mathsf{ID}, b, *, h)$ is added in the H-list.

Query $\hat{H}(R, \mathsf{ID})$: if a record (R, ID, r, K) appears in the $\hat{H}$-list, the answer is $r||K$. Otherwise the answer (r, K) is chosen randomly in $\mathbb{Z}_q \times \mathcal{K}$ and the record (R, ID, r, K) is added in the $\hat{H}$-list.

Ext-Oracle

Query $\textsc{Extract}(\mathsf{ID})$: the answer $(b_{\mathsf{ID}}, d_{\mathsf{ID}})$ is defined by the following rules.

▶ **Rule** $\mathsf{Extract}^{(1)}$

| Compute $d_{\mathsf{ID}} = H(\mathsf{ID}, b_{\mathsf{ID}})^s$.

Decryption-Oracle

Query $\textsc{Decrypt}(\mathsf{ID}, u, V_0, V_1, \alpha)$: the answer M is defined by the following rules. First get the secret key d_{ID} by using Extract rule.

▶ **Rule** $\mathsf{Decrypt\text{–}Exception}^{(1)}$

| Do nothing.

Then compute:

(D1) $w'_{b_{\mathsf{ID}}} = e(u, d_{\mathsf{ID}})$, $\qquad R_{b_{\mathsf{ID}}} = V_{b_{\mathsf{ID}}} \oplus G(w'_{b_{\mathsf{ID}}}, \mathsf{ID}, b_{\mathsf{ID}})$,

(D2) $r' \| K = \hat{H}(R_{b_{\mathsf{ID}}}, \mathsf{ID})$,

(D3) $w'_{\bar{b}_{\mathsf{ID}}} = e(g_{pub}, H(\mathsf{ID}, \bar{b}_{\mathsf{ID}}))^{r'}$, $R_{\bar{b}_{\mathsf{ID}}} = V_{\bar{b}_{\mathsf{ID}}} \oplus G(w'_{\bar{b}_{\mathsf{ID}}}, \mathsf{ID}, \bar{b}_{\mathsf{ID}})$,

(D4) if $R_{b_{\mathsf{ID}}} \neq R_{\bar{b}_{\mathsf{ID}}}$ or $u \neq g^{r'}$, then return "*reject*"
else compute $M = D_K(\alpha)$ and return M.

Challenge

For two messages (M_0, M_1) and identity $\mathsf{ID}^\star$, flip a coin β and set $M^\star = M_\beta$, choose randomly $R^\star \in \{0,1\}^{k_1}$, and then answer $(u^\star, V_0^\star, V_1^\star, \alpha^\star)$ where

▶ **Rule** $\mathsf{Chal\text{–}DEM\text{–}Key}^{(1)}$

| Compute $r^\star \| K^\star := \hat{H}(R^\star, \mathsf{ID}^\star)$, then let $K^\ddagger = K^\star$.

▶ **Rule** $\mathsf{Chal\text{–}KEM}^{(1)}$

| $u^\star = g^{r^\star}$,
| $w_0^\star = e(g_{pub}, H(\mathsf{ID}^\star, 0))^{r^\star}$, $\qquad V_0^\star = G(w_0^\star, \mathsf{ID}^\star, 0) \oplus R^\star$,
| $w_1^\star = e(g_{pub}, H(\mathsf{ID}^\star, 1))^{r^\star}$, $\qquad V_1^\star = G(w_1^\star, \mathsf{ID}^\star, 1) \oplus R^\star$.

▶ **Rule** $\mathsf{Chal\text{–}DEM\text{–}Enc}^{(1)}$

| Let $\alpha^\star = \mathrm{Enc}_{K^\ddagger}(M^\star)$.

Fig. 1. The formal simulation of the IND–ID–CCA game

Game $\mathbf{G}_3$: From now, we change the setup, as well as $\mathsf{Extract}$ rule. Instead of using $g_{pub} = g^s$, for a chosen $s \in \mathbb{Z}_q$, we use $g_{pub} = g_1$ (for which we do not know the value a such that $g_1 = g^a$). Furthermore our $\mathsf{Extract}$ rule becomes:

▶ **Rule** $\mathsf{Extract}^{(3)}$

Ask $H(\mathsf{ID}, b_{\mathsf{ID}})$ to the H-oracle. Find $(\mathsf{ID}, b_{\mathsf{ID}}, \pi_{\mathsf{ID}}, h)$ in the H-list and let $d_{\mathsf{ID}} = g_1^{\pi_{\mathsf{ID}}}$.

One can see that d_{ID} is valid: $d_{\mathsf{ID}} = H(\mathsf{ID}, b_{\mathsf{ID}})^a$. This is since $H(\mathsf{ID}, b_{\mathsf{ID}}) = g^{\pi_{\mathsf{ID}}}$. The two games $\mathbf{G}_2$ and $\mathbf{G}_3$ are perfectly indistinguishable: $\Pr[\mathsf{S}_2] = \Pr[\mathsf{S}_3]$.

Game $\mathbf{G}_4$: In this game, we make a conceptual modification for the decryption oracle. This modification will be useful in game $\mathbf{G}_6$ below.

▶ **Rule** $\mathsf{Decrypt–Exception}^{(4)}$

- If $(\mathsf{ID}, u, V_0, V_1) = (\mathsf{ID}^\star, u^\star, V_0^\star, V_1^\star)$ but $\alpha \neq \alpha^*$, then return $\mathrm{Dec}_{K^\ddagger}(\alpha)$.
- If $u \neq u^\star$ and $V_{b_{\mathsf{ID}^\star}} \oplus G(e(u, d_{\mathsf{ID}^\star}), \mathsf{ID}^\star, b_{\mathsf{ID}^\star}) = R^\star$, return "reject".

The two games $\mathbf{G}_3$ and $\mathbf{G}_4$ are perfectly indistinguishable since the change is only conceptual. The first one is verified by observing that from $u = u^\star$ we have $r = r^*$ which then leads to $R = R^\star$ due to (D1) and the above condition. Hence $K = K^\star = K^\ddagger$ due to (D2) and the $\mathsf{Chal–DEM–Key}$ rule. The second one is verified by first assuming that such a query is valid. Since $u \neq u^\star$, then $r \neq r^\star$. From the above constraint we must have $r||K = \hat{H}(R^\star, \mathsf{ID}^\star) = r^\star||*$ hence a contradiction. Thus such a query must be invalid. Therefore $\Pr[\mathsf{S}_3] = \Pr[\mathsf{S}_4]$.

Game $\mathbf{G}_5$: In this game, we modify the challenge rule, by simplifying its KEM component to:

▶ **Rule** $\mathsf{Chal–KEM}^{(5)}$

$$\begin{array}{ll} u^\star = g_3, & \\ G_0^\dagger \leftarrow \{0,1\}^{k_1}, & V_0^\star = G_0^\dagger \oplus R^\star, \\ G_1^\dagger \leftarrow \{0,1\}^{k_1}, & V_1^\star = G_1^\dagger \oplus R^\star. \end{array}$$

The two games $\mathbf{G}_4$ and $\mathbf{G}_5$ are perfectly indistinguishable unless at least one of the following events occurs:

$$\begin{array}{ll} \mathsf{AskGoodG} : (e(g,g)^{abc\tau_{\mathsf{ID}^\star}}, \mathsf{ID}^*, \bar{b}_{\mathsf{ID}^\star}) \text{ is asked to } G\text{-oracle;} \\ \mathsf{AskBadG} \;\; : (e(g_1,g_3)^{\pi_{\mathsf{ID}^\star}}, \mathsf{ID}^*, b_{\mathsf{ID}}) \text{ is asked to } G\text{-oracle} \end{array}$$

either by the adversary or the decryption oracle. By the difference lemma (see [16]), we thus have

$$|\Pr[\mathsf{S}_4] - \Pr[\mathsf{S}_5]| \leq \Pr[\mathsf{AskGoodG}] + \Pr[\mathsf{AskBadG}] \leq 2\Pr[\mathsf{AskGoodG}],$$

where the last inequality is due to the claim below. Before proving the claim, we will conclude the result from this game by constructing an algorithm $\mathcal{B}$ for solving the LBDH problem. Assume that $\mathsf{AskGoodG}$ occurs. Let L be a list which is empty at first. From each record $(w, \mathsf{ID}^\star, \bar{b}_{\mathsf{ID}^\star}, g)$ in the G-list, algorithm $\mathcal{B}$ adds $g^{1/\tau_{\mathsf{ID}^\star}}$ to the L list and output this list. Since $\mathsf{AskGoodG}$ occurs, L contains $e(g,g)^{abc}$. This implies $\Pr[\mathsf{AskGoodG}] \leq \epsilon_{lbdh}$. Hence, $|\Pr[\mathsf{S}_4] - \Pr[\mathsf{S}_5]| \leq 2\epsilon_{lbdh}$.

Claim. $\Pr[\mathsf{AskGoodG}] = \Pr[\mathsf{AskBadG}]$.

Proof. (of the claim) It is sufficient to prove that the adversary's view is independent of the variable $b_{\mathsf{ID}^\star}$. Moreover, since the only variables that are possibly dependent on $b_{\mathsf{ID}^\star}$ are those responses from the decryption oracle, it is sufficient to prove that there exists no ciphertext such that its decryption result may become different values according to the value of b_{ID}. We assume for the sake of contradiction that there exists $C = \langle u, V_0, V_1, \alpha\rangle$ such that $\textsc{Decrypt}_{b_{\mathsf{ID}^\star}=0}(\mathsf{ID}^\star, C) \neq \textsc{Decrypt}_{b_{\mathsf{ID}^\star}=1}(\mathsf{ID}^\star, C)$, where the subscripts denote the conditional events. Without loss of generality, we assume that the value on the left is M which is not "*reject*". Let $r = \log_g u$. As in (D1) (when $b_{\mathsf{ID}^\star} = 0$), we let $\bar{R} := V_0 \oplus G(e(g_{pub}, H(\mathsf{ID}^\star, 0))^r, \mathsf{ID}^\star, 0)$ and as in (D2) we let $r' \| K := \hat{H}(\bar{R}, \mathsf{ID})$. Then we have $\alpha = E_K(M)$ from (D4).

Since M is not"*reject*", we have $\bar{R} = V_1 \oplus G(e(g_{pub}, H(\mathsf{ID}^\star, 1))^{r'}, \mathsf{ID}^\star, 1)$ and $r = r'$ due to (D3) and (D4). Thus

$$V_1 = G(e(g_{pub}, H(\mathsf{ID}^\star, 1))^r, \mathsf{ID}^\star, 1) \oplus \bar{R}. \qquad (1)$$

Now the decryption oracle conditioned on $b_{\mathsf{ID}^\star} = 1$ will decrypt C by executing (D1) (when $b_{\mathsf{ID}^\star} = 1$) and obtaining $\bar{R}$ due to Eq.(1). From (D2), we thus obtain the same K as above. The condition in (D4) is true by the definition of $\bar{R}$. Thus the oracle will return M, a contradiction. This completes the proof of the claim.

<u>Game $\mathbf{G}_6$</u>*:* In this game, we modify the challenge rule, by simplifying its DEM component to:

▶ **Rule** $\mathsf{Chal\text{–}DEM\text{–}Key}^{(6)}$

| Randomly choose $K^\dagger \in_R \mathcal{K}$, then let $K^\ddagger = K^\dagger$.

The two games $\mathbf{G}_5$ and $\mathbf{G}_6$ are perfectly indistinguishable unless the query $(R^\star, \mathsf{ID}^\star)$ is asked to the $\hat{H}$-oracle, by either the adversary or the decryption oracle. But the latter case is not possible. This is since such a decryption query must be $(\mathsf{ID}^\star, u, V_0, V_1, \alpha)$ such that $R^\star = V_{b_{\mathsf{ID}^\star}} \oplus G(u, d_{\mathsf{ID}^\star}, b_{\mathsf{ID}^\star})$ in order to force the decryption oracle to ask $(R^\star, \mathsf{ID}^\star)$ to the $\hat{H}$-oracle. If $u = u^\star$, then this leads to $(\mathsf{ID}, u, V_0, V_1) = (\mathsf{ID}^\star, u^\star, V_0^\star, V_1^\star)$. Hence in this case the decryption query is either the challenge ciphertext itself (so it will be rejected) or its process for decryption falls into the first $\mathsf{Decrypt\text{–}Exception}$ rule (so the decryption oracle will not ask such a $\hat{H}$-oracle query). If $u \neq u^\star$, then such a query will be rejected due to the second $\mathsf{Decrypt\text{–}Exception}$ rule (and so in particular, the decryption oracle will not ask such a $\hat{H}$-oracle query). Therefore, from the difference lemma, we have

$$|\Pr[\mathsf{S}_5] - \Pr[\mathsf{S}_6]| \leq \frac{q_{\hat{H}}}{2^{k_1}}$$

which is the probability that the adversary correctly guesses R^* in one of $q_{\hat{H}}$ times. The adversary is forced to simply guess since the other information about R^* is perfectly hiding thanks to the independent random values $G_0^\dagger$ and $G_1^\dagger$.

Game $\mathbf{G}_7$: In this game, we further modify the challenge rule, by replacing the challenge message by another fixed message with the same length:

▶ **Rule** $\mathsf{Chal\text{–}DEM\text{–}Enc}^{(7)}$

| Let $\alpha^\star = \mathrm{Enc}_{K^\ddagger}(0^{|M^\star|})$.

The output of the adversary follows from a distribution that does not depend on β. Accordingly, $\Pr[\mathsf{S}_7] = 1/2$. We also claim that

$$|\Pr[\mathsf{S}_6] - \Pr[\mathsf{S}_7]| \leq 2\epsilon_{sym}.$$

We prove this by constructing an algorithm $\mathcal{S}$ which has an IND-CCA advantage for the symmetric encryption scheme of exactly $(\Pr[\mathsf{S}_6] - \Pr[\mathsf{S}_7])/2$. Algorithm $\mathcal{S}$ first asks $(0^{|M^\star|}, M^\star)$ to obtain the challenge ciphertext ψ^+, and will try to guess the bit γ. Algorithm $\mathcal{S}$ runs the IBE adversary by providing the simulation in exactly the same way as done by the challenger in game $\mathbf{G}_6$ except only for the rules which produce or use $K^\ddagger$, which are (1) the Chal–DEM–Key rule (producing $K^\ddagger$), (2) the Chal–DEM–Enc rule (using $\mathrm{Enc}_{K^\ddagger}(\cdot)$), and (3) the first Decrypt–Exception rule (using $\mathrm{Dec}_{K^\ddagger}(\cdot)$). For those exceptions, $\mathcal{S}$ does nothing for (1), let $\alpha^\star = \psi^+$ for (2), and queries to its decryption oracle for (3). Finally if $\beta' = \beta$, then $\mathcal{S}$ output 1; else 0. It is clear that $\Pr[\gamma' = 1|\gamma = 1] = \Pr[\mathsf{S}_6]$ and $\Pr[\gamma' = 1|\gamma = 0] = \Pr[\mathsf{S}_7]$. Hence $\Pr[\gamma' = \gamma] - 1/2 = (\Pr[\mathsf{S}_6] - \Pr[\mathsf{S}_7])/2$ as claimed.

From all the results above, we now can conclude that $\epsilon_{ibe} \leq 2\epsilon_{lbdh} + 2\epsilon_{sym} + \frac{q_{\hat{H}}}{2^{k_1}}$, which completes the proof. The running time can be easily verified. □

We next state somewhat straightforward reductions from the ℓ–LDBH problem to the DBDH, GBDH, and CBDH problems. The first two are tight, while in the last one the security is degraded by factor ℓ.

Lemma 1. *Suppose that there exists an $(\epsilon_{lbdh}, t_{lbdh})$-$\ell$-LBDH algorithm $\mathcal{L}$. Then there exists an $(\epsilon_{dbdh}, t_{dbdh})$-DBDH algorithm $\mathcal{D}$, $(\epsilon_{gbdh}, t_{gbdh})$-GBDH algorithm $\mathcal{G}$, and $(\epsilon_{cbdh}, t_{cbdh})$-CBDH algorithm $\mathcal{C}$ such that*

$$\begin{array}{ll} \epsilon_{dbdh} \geq \epsilon_{lbdh} - \ell/|\mathbb{G}_2|, & t_{dbdh} \leq t_{lbdh} + \ell\tau_1, \\ \epsilon_{gbdh} \geq \epsilon_{lbdh}, & t_{gbdh} \leq t_{lbdh} + \ell\tau_2, \\ \epsilon_{cbdh} \geq \epsilon_{lbdh}/\ell, & t_{cbdh} \leq t_{lbdh} + \tau_3, \end{array}$$

where τ_1 is the time required to check an equality of two elements in $\mathbb{G}_2$, τ_2 is the time required to access the DBDH oracle (as provide for $\mathcal{G}$), and τ_3 is the time required to randomly choose one element from a list of size ℓ.

Proof. The description of the algorithms $\mathcal{D}, \mathcal{G}, \mathcal{C}$ are as follows. Given these descriptions, the above relations can be easily verified.

- The algorithm $\mathcal{D}$, upon input (g, g_1, g_2, g_3, T), runs $\mathcal{L}$ on the input (g, g_1, g_2, g_3) and, in response, obtains a list, which contains $e(g_1, g_2)^{\log_g g_3}$ with probability ϵ_{lbdh}. Then $\mathcal{D}$ outputs 1 if the list contains T, and 0 otherwise.
- The algorithm $\mathcal{G}$ forwards its input to $\mathcal{L}$ and get a list. Then it tests all the elements in the list by calling the DBDH oracle $\mathcal{O}$. If the oracle returns 1 for some query, then $\mathcal{G}$ outputs that query.
- The algorithm $\mathcal{C}$ forwards its input to $\mathcal{L}$ and get a list. Then it randomly chooses one element in the list and outputs it. □

From this lemma and Theorem 1, the following main result is immediate.

Theorem 2. *Given the same supposition as in Theorem 1, we have that there exists an $(\epsilon_{dbdh}, t_{dbdh})$-DBDH algorithm $\mathcal{D}$, an $(\epsilon_{gbdh}, t_{gbdh})$-GBDH algorithm $\mathcal{G}$, and an $(\epsilon_{cbdh}, t_{cbdh})$-CBDH algorithm $\mathcal{C}$ such that*

$$\begin{aligned}
\epsilon_{dbdh} &\geq \frac{1}{2}\epsilon_{ibe} - \epsilon_{sym} - \frac{q_{\hat{H}}}{2^{k_1+1}} - \frac{q_G + q_D}{|\mathbb{G}_2|},\\
t_{dbdh} &\leq t_{ibe} + (3q_H + q_G + 3q_E + 10q_D)\tau + q_{\hat{H}}\tau' + q_D t_{sym} + (q_G + q_D)\tau_1,\\
\epsilon_{gbdh} &\geq \frac{1}{2}\epsilon_{ibe} - \epsilon_{sym} - \frac{q_{\hat{H}}}{2^{k_1+1}},\\
t_{gbdh} &\leq t_{ibe} + (3q_H + q_G + 3q_E + 10q_D)\tau + q_{\hat{H}}\tau' + q_D t_{sym} + (q_G + q_D)\tau_2,\\
\epsilon_{cbdh} &\geq \frac{1}{q_G + q_D}\left(\frac{1}{2}\epsilon_{ibe} - \epsilon_{sym} - \frac{q_{\hat{H}}}{2^{k_1+1}}\right),\\
t_{cbdh} &\leq t_{ibe} + (3q_H + q_G + 3q_E + 10q_D)\tau + q_{\hat{H}}\tau' + q_D t_{sym} + \tau_3,
\end{aligned}$$

where τ, τ' are defined as in Theorem 1 and τ_1, τ_2, τ_3 are defined as in Lemma 1.

5 (Weakly) Tight Reduction to CBDH

In this section, we prove that security of our proposed scheme can be also tightly reduced to the CBDH problem in the sense of *weak* tightness (See Section 2.5). Our reduction technique is due to [15] which is based on random self reducibility of the given problem.

Lemma 2. *If there exists an $(\epsilon_{lbdh}, t_{lbdh})$-$\ell$-LBDH algorithm, then there exists an $(\epsilon_{cbdh}, t_{cbdh})$-CBDH algorithm $\mathcal{A}$ such that*

$$\epsilon_{cbdh} \geq \begin{cases} \frac{1}{4}\left(1 - \frac{\ell^2}{\epsilon_{lbdh}^2 (q-1)}\right) & \textit{if } \epsilon_{lbdh} \leq 1/2 \\ \frac{1}{4}\left(1 - \frac{4\ell^2}{q-1}\right) & \textit{if } \epsilon_{lbdh} > 1/2 \end{cases}, \qquad t_{cbdh} \leq \begin{cases} \frac{t_{lbdh}}{\epsilon_{lbdh}} & \textit{if } \epsilon_{lbdh} \leq 1/2 \\ 2t_{lbdh} & \textit{if } \epsilon_{lbdh} > 1/2 \end{cases}.$$

Proof. $\mathcal{A}$ runs the $(\epsilon_{lbdh}, t_{lbdh})$-$\ell$-LBDH algorithm for N times where N will be determined later. In the i-th time run, $\mathcal{A}$ chooses $x_i, y_i \in \{1, ..., q-1\}$ at random and inputs $((g^a)^{x_i} g^{y_i}, g^b, g^c)$ to the LBDH algorithm, obtaining a list $L_i = \langle h_{i,1}, \ldots, h_{i,\ell} \rangle$ of elements in the group $\mathbb{G}_2$. For each list L_i we construct another list $L'_i = \langle t_{i,1}, \ldots, t_{i,\ell} \rangle$ where we let

$$t_{i,j} = (h_{i,j} \cdot e(g^b, g^c)^{-y_i})^{(x_i^{-1} \bmod q)}. \tag{2}$$

Next, $\mathcal{A}$ tests if there exist $1 \leq r < s \leq N$ such that there exists a unique pair (u, v) where $1 \leq u \leq \ell$ and $1 \leq v \leq \ell$ such that $t_{r,u} = t_{s,v}$ (where uniqueness is in the sense that for all $1 \leq u' \leq \ell$ and $1 \leq v' \leq \ell$ such that $u' \neq u, v' \neq v$ we have that $t_{r,u'} \neq t_{s,v'}$). If this is satisfied, we output $t_{r,u}$; otherwise, $\mathcal{A}$ reports failure.

We now analyze the correctness. Let $h = e(g, g)$ (a generator in $\mathbb{G}_2$). Precisely, we want a lower bound the probability of the following event:

$$\exists(r, s, u, v)[t_{r,u} = t_{s,v} = h^{abc}] \wedge \not\exists(r', s', u', v')[t_{r',u'} = t_{s',v'} \neq h^{abc}] \qquad (3)$$

We first claim that the event $\exists(r, u)\ t_{r,u} = h^{abc}$ is exactly the event that the LBDH algorithm succeeds at least once (namely, the r-th run). This is since, due to Eq.(2), $t_{r,u} = h^{abc}$ if and only if $h_{r,u} = h^{(ax_r + y_r)bc}$ and we have that $((g^a)^{x_r} g^{y_r}, g^b, g^c, h^{(ax_r+y_r)bc})$ is a BDH tuple. From the claim we thus have that the event $\exists(r, u, s, v)\ t_{r,u} = t_{s,v} = h^{abc}$ is exactly the event the LBDH algorithm succeeds at least twice (namely, the r-th and s-th runs). This happens with probability at least $1 - (1 - \epsilon)^N - N\epsilon(1 - \epsilon)^{N-1}$.

Next we will bound the conditional probability of the event $\not\exists(r', s', u', v')$ $[t_{r',u'} = t_{s',v'} \neq h^{abc}]$. Denote the event in the given part as A. Let $z_i = ax_i + y_i$, $\alpha = \log_h h_{r',u'}$ and $\beta = \log_h h_{s',v'}$. For any r', s', u', v', we have

$$\begin{aligned}
\Pr[t_{r',u'} = t_{s',v'} \neq h^{abc}] &= \Pr_{\substack{x_{r'}, y_{r'},\\ x_{s'}, y_{s'}}}[(\alpha - bcy_{r'})(x_{r'}^{-1} \bmod q) = (\beta - bcy_{s'})(x_{s'}^{-1} \bmod q) \\
&\qquad\qquad \mid ax_{r'} + y_{r'} = z_{r'} \wedge ax_{s'} + y_{s'} = z_{s'}] \\
&= \Pr_{x_{r'}, x_{s'}}[x_{s'}(\alpha - bcz_{r'}) = x_{r'}(\beta - bcz_{s'})] \\
&= \Pr_x[(\alpha - bcz_{r'})x - (\beta - bcz_{s'}) = 0] \leq \frac{1}{q-1}.
\end{aligned}$$

Therefore $\Pr[\mathsf{A}] \geq 1 - N^2(\ell^2/(q - 1))$. Combining these, we have

$$\Pr[(3)] \geq (1 - (1 - \epsilon_{lbdh})^N - N\epsilon_{lbdh}(1 - \epsilon_{lbdh})^{N-1})(1 - N^2 \cdot \ell^2/(q - 1)).$$

We choose N as the function of ϵ_{lbdh} as follows: let $N(\epsilon_{lbdh}) := \lceil 1/\epsilon_{lbdh} \rceil$ if $\epsilon_{lbdh} \leq 1/N^\star$; and $N(\epsilon_{lbdh}) := N^\star$ otherwise, where $N^\star \geq 2$ is a fixed value from $\mathbb{Z}$. We define

$$\begin{aligned}
f(\epsilon) &:= 1 - (1 - \epsilon)^{1/\epsilon} - (1 - \epsilon)^{1/\epsilon - 1}; \\
g(\epsilon) &:= 1 - (1 - \epsilon)^{N^\star} - N^\star\epsilon(1 - \epsilon)^{N^\star - 1}; \\
h(\epsilon) &:= 1 - (1 - \epsilon)^{N(\epsilon)} - N(\epsilon)\epsilon(1 - \epsilon)^{N(\epsilon) - 1}.
\end{aligned}$$

Observe that f is a monotone decreasing function while g is a monotone increasing function in the interval $[0, 1]$. Hence we have that $h(\epsilon)$ is minimum when $\epsilon = 1/N^\star$. Observe that $N(1/N^\star) = N^\star$, we thus have $h(\epsilon) \geq 1 - (1 - 1/N^\star)^{N^\star} - (1 - 1/N^\star)^{N^\star - 1}$. To maximize this lower bound, we will choose $N^\star$ as large as possible since it tends to its maximum, $1 - 2e^{-1}$ (where e is the base of natural logarithm), as $N^\star \to \infty$. However, for simplicity, choosing $N^\star = 2$ is sufficient for our purpose. Therefore $h(\epsilon) \geq 1/4$ and the the probability bound in the lemma statement holds. □

Theorem 3. *Given the same supposition as in Theorem 1, we have that there exists an* $(\epsilon_{cbdh}, t_{cbdh})$*-CBDH algorithm such that*

$$\frac{t_{cbdh}}{\epsilon_{cbdh}} \leq 4(1 - \frac{\ell^2}{\epsilon'^2_{lbdh}(q-1)})^{-1} \frac{t'_{lbdh}}{\epsilon'_{lbdh}},$$

where $\epsilon'_{lbdh} := \frac{1}{2}\epsilon_{ibe} - \epsilon_{sym} - \frac{q_{\hat{H}}}{2^{k_1+1}}$, $t'_{lbdh} := t_{ibe} + (3q_H + q_G + 3q_E + 10q_D)\tau +$ $+ q_{\hat{H}}\tau' + q_D t_{sym}$ *(which are the parameters from Theorem 1), and* τ, τ' *are as defined in Theorem 1.*

Proof. It follows from Theorem 1, Lemma 2, and the fact that $\epsilon'_{lbdh} \leq 1/2$. □

Since $\ell^2/(\epsilon'^2_{lbdh}(q-1))$ is negligible (since ϵ'_{lbdh} is non-negligible by the supposition), we have that $t_{cbdh}/\epsilon_{cbdh} \simeq 8t_{ibe}/\epsilon_{ibe}$.

6 Performance

The following table compares the performance of our scheme with other IBE schemes which their security proofs are done in the random oracle model.

Scheme	Security as IND-ID-X			|Ciphertext|
	Assumption	X	Reduction Cost	(bits)
BF01(FullIdent) [3]	CBDH	CCA	$O(1/q_h^2)$ ¶	$\lvert M\rvert + 250$
	GBDH	CCA	$O(1/q_h)$	
G05(NewFull-Ident) [7]	CBDH	CCA	$O(1/q_h^2)$	$\lvert M\rvert + 250$
LQ05 [11]	GBDH	CCA	$O(1/q_e)$	$\lvert M\rvert + 170$
BF01(BasicIdent) [3] +KW03 [10]	GBDH	CPA	$O(1)$	$2\lvert M\rvert + 340$
BF01(FullIdent) [3] +KW03 [10]	GBDH	CPA	$O(1)$	$2\lvert M\rvert + 500$
TightIBE	DBDH	CCA	$O(1)$	$\lvert M\rvert + 330$
	GBDH	CCA	$O(1)$	
	CBDH	CCA	$O(1/q_h)$	
	CBDH	CCA	$O(1)$ (weak)	

q_e,q_h : the number of queries to the Extraction oracle and the random oracle respectively.

Table 4. Comparison among IBE schemes in the random oracle model.

The Boneh-Franklin IBE scheme was proven secure by assuming the CBDH problem is hard, while one can make a stronger (e.g. GBDH) assumption and have a tighter reduction. A flawed step in the proof of the Boneh-Franklin scheme was pointed by [7], and a modified proof was proposed. We present these reduction results in the table.

Applying the Katz-Wang technique to the IND-ID-CCA version of the Boneh-Franklin IBE scheme does not result in an IND-ID-CCA secure, but an IND-ID-CPA secure IBE scheme.

Here parameters are chosen as: $|\mathbb{G}_1| = 170$ bits, $|R| = 80$ bits. We note that the security parameter $|R| = 80$ is enough to achieves security comparable to that of $|\mathbb{G}_1| = 170$ bits. Moreover, for those schemes without tight security reductions, the security parameters have to be chosen larger in order to compensate such a security loss. Taking account of all these factors, we conclude that our scheme is the most efficient among these schemes.

7 Conclusion

In Crypto'01, Boneh and Franklin introduced the first ID-based encryption scheme. Their scheme security (more precisely, the notion of resistance against an IND-ID-CCA attacker) relies in the random oracle model, but the reduction is far from being tight, and notably depends on the number of extractions queries.

In this paper, we have presented an efficient modification to the Boneh-Franklin scheme that provides a tight reduction. Our scheme is basically an IBE under two keys, one of which is (randomly) detained by the recipient. Our scheme is a continuation of an idea introduced by Katz and Wang. However, we have shown how to deal with the problem of IND-ID-CCA security, while it was quite unclear with the original description of Katz and Wang.

Our scheme features a tight reduction to the LBDH problem, which can be itself reduced tightly either to the GBDH or the DBDH problems. Furthermore, for a relaxed definition of tightness (called weak-tightness) that we have introduced and discussed, we have shown that there is a weakly-tight reduction from our scheme to the CBDH problem.

Our scheme is very efficient, as one can precompute most of the quantities involved in the encryption process. Furthermore, contrarily to the Katz-Wang IBE, in our scheme, the ciphertext size and the encryption timing are roughly equivalent to the Boneh-Franklin one's. Unfortunately, our decryption process is twice as much as that of the Boneh-Franklin IBE or that of the Katz-Wang IBE.

It is still an open problem to build chosen ciphertext secure IBE that obtain tight security reductions under reasonable assumptions in the standard model.

Acknowledgement

Rui Zhang is supported by a JSPS Fellowship. Benoit Chevallier-Mames would like to thank his PhD advisor, David Pointcheval, as well as his coauthors that allowed him to take place in this on-going work.

[¶] A simple recounting of the reduction with Galindo's strategy [7] is adopted here, while the original reduction given in [7] is $O(1/q_h^3)$.

References

1. D. Boneh and X. Boyen, “Efficient Selective-ID Secure Identity-Based Encryption Without Random Oracles,” In Advances in Cryptology–Eurocrypt’04, LNCS 3027, pp.223-238, 2004. The full version is available as IACR ePrint Report 2004/172.
2. D. Boneh and X. Boyen, “Secure Identity Based Encryption Without Random Oracles,” In Advances in Cryptology–Crypto’04, LNCS 3152, pp.443-459, 2004.
3. D. Boneh and M. Franklin, “Identity Based Encryption from the Weil Pairing,” In Advances in Cryptology–Crypto’01, LNCS 2139, pp.213-229, 2001.
4. D. Boneh and M. Franklin, “Identity Based Encryption from the Weil Pairing,” SIAM Journal of Computing 32(3):586-615, 2003, full version of [3].
5. R. Canetti, S. Halevi and J. Katz, “A Forward-Secure Public-Key Encryption Scheme,” In Advances in Cryptology–Eurocrypt’03, LNCS 2656, pp.255-271, 2003.
6. E. Fujisaki and T. Okamoto, “Secure Integration of Asymmetric and Symmetric Encrytion Schemes,” In Advances in Cryptology–Crypto’99, LNCS 1666, pp.537-554, 1999.
7. D. Galindo, “Boneh-Franklin Identity Based Encryption Revisited,” In ICALP’05, LNCS 3580, pp.791-802, 2005.
8. S. Halevi and P. Rogaway, “A Tweakable Enciphering Mode” In Advances in Cryptology–Crypto’03, LNCS 2729, pp.482-499, 2003.
9. S. Halevi and P. Rogaway, “A Parallelizable Enciphering Mode.” In CT-RSA 2004, LNCS 2964, pp.292-304, 2004.
10. J. Katz and N. Wang, “Efficiency Improvements for Signature Schemes with Tight Security Reductions,” In ACM-CCS’03, pp.155-164, 2003.
11. B. Libert and J. Quisquater, “Identity Based Encryption Without Redundancy,” In ACNS’05, pp.285-300, 2005.
12. T. Okamoto and D. Pointcheval, “The Gap-Problems: a New Class of Problems for the Security of Cryptographic Schemes,” In PKC’01, LNCS 1992, pp.104-118, 2001.
13. D. Pointcheval and J. Stern, “Security arguments for digital signatures and blind signatures,” In Journal of Cryptology, 13(3), pp.361-396, 2000.
14. A. Shamir, “Identity-Based Cryptosystems and Signature Schemes,” In Advances in Cryptology–Crypto’84, LNCS 293, pp.341-349, 1984.
15. V. Shoup, “Lower Bounds for Discrete Logarithms and Related Problems,” In Eurocrypt’97, LNCS 1233, pp.256-266, 1997.
16. V. Shoup, “Sequences of Games: A Tool for Taming Complexity in Security Proofs,” IACR ePrint Report 2004/332.
17. B. Waters,“Efficient Identity-Based Encryption Without Random Oracles,” In Advances in Cryptology–Eurocrypt’05, LNCS 1666, pp.114-127, 2005. The full version is available as IACR ePrint Report 2004/180.