Identity-Based Encryption Schemes with Tight Security Reductions

Nuttapong Attrapadung* Jun Furukawa*† Takeshi Gomi* Goichiro Hanaoka‡
Hideki Imai*‡ Rui Zhang*

*Institute of Industrial Science, University of Tokyo.
{nuts,takego,zhang}@imailab.iis.u-tokyo.ac.jp
imai@iis.u-tokyo.ac.jp
†NEC Corporation.
j-furukawa@ay.jp.nec.com
‡Research Center for Information Security,
National Institute of Advanced Industrial Science and Technology.
hanaoka-goichiro@aist.go.jp

Abstract

We present three variants of the Boneh-Franklin identity-based encryption scheme so as to have tight security reductions in the full security notion (to be precise, in the sense of IND-ID-CCA) in the random oracle model based on the technique suggested by Katz and Wang. The first one is based on the Gap Bilinear Diffie-Hellman (Gap BDH) assumption. The other two are based on the Decision BDH assumption, one of which has more compact ciphertext size while the other enjoys a publicly verifiability of ciphertexts.

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.

Although the concept of identity based encryption was proposed two decades ago [16], it is only recently that the first fully functional schemes were proposed. Boneh and Franklin [5, 6] defined a security model namely IND-ID-CCA[1] and gave the first efficient construction provably secure in the random oracle model based on the Bilinear Diffie-Hellman (BDH) problem. Since then, there have been schemes shown to be secure without random oracles, but in a weaker model of security known as "Selective-ID" model [7, 2]. 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 [3], 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 [4] 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 [18].

However, for each of the above schemes, its 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 the lower security level or the requirement of larger key size to obtain the same security level.

[1] INDistinguishability under Chosen-Ciphertext Attack for ID-based encryption

It has been an open problem (as already posed in [18, 12]) whether efficient IBE systems can exist with their security in the sense of IND-ID-CCA being reduced *tightly* (i.e., the reduction cost being only a constant term) to some reasonable intractability assumption. In the standard model, this is still an open problem. On the other hand, in the random oracle model, it has been partially solved by Katz and Wang [13].

Katz and Wang [13] suggested an elegant technique for constructing a variant of Boneh-Franklin which is secure in the sense of IND-ID-CPA with tight security reduction. Although they did not go into details since this issue was not the main focus of their paper, it is straightforward to prove its chosen-plaintext security (CPA).

However, to achieve CCA security, one might attempt to apply some CCA conversion with tight security reduction to the scheme modified by Katz and Wang so that we could have an overall IBE scheme tightly secure in the CCA sense. We explain in the following why this straightforward combination is essentially insufficient. To this end, the problem motivates us to find a proper modification so as to obtain CCA security while seamlessly integrate with Katz-Wang technique. In this paper, we present three such variants. To the best of our knowledge, these thus give the first explicit CCA-secure IBE schemes with tight security reduction.

1.1 Some Attempts

We now point out why the straightforward combination as described above does not work. The technique from [13] requires to modify the full Boneh-Franklin scheme (but now converted from the basic scheme by some CCA-conversion with tight reduction) so that for any ID there are two "public keys" $H(\mathsf{ID}||0)$ and $H(\mathsf{ID}||1)$; furthermore, to encrypt a message to user ID, a sender now encrypts the message with respect to both of these public keys. The private key generator (PKG), however, gives to ID only one of the corresponding keys, say the key corresponding to $H(\mathsf{ID}||b_{\mathsf{ID}})$, where b_{ID} is a secret fixed bit corresponding to ID and is maintained by the PKG. Note that a single key is sufficient to enable decryption. Following the proof technique in [13], a simulation can be set up so that the simulator knows exactly one secret key for every ID, in particular $H(\mathsf{ID}||b_{\mathsf{ID}})$. This enables the simulator to simulate the key exposure oracle for all queries while ensuring that encryption to any non-exposed ID remains secret. Let ID^* be the challenge identity. Up to some point, in order to gather some information about the challenge ciphertext the adversary will essentially come up with a random oracle query related to the encryption corresponding to either $H(\mathsf{ID}^*||b_{\mathsf{ID}^*})$ or $H(\mathsf{ID}^*||\bar{b}_{\mathsf{ID}^*})$, in which the simulator embeds the underlying problem instance in the latter. Therefore, without knowing b_{ID^*}, the adversary will ask the latter query with probability $\frac{1}{2}$, and the simulator can solve the underlying problem instance with the reduction cost $\frac{1}{2}$. The proof relies essentially on the fact that b_{ID^*} is perfectly hiding.

However, the following simple adversary A in the chosen-ciphertext attack scenario can cause the above simulator S, who uses A as a subroutine and tries to break the underlying assumption, to *always fail*. A will query the decryption oracle for $Enc_{H(\mathsf{ID}^*||0)}(m_0)||Enc_{H(\mathsf{ID}^*||1)}(m_1)$ for some $m_0 \neq m_1$. If the oracle returns m_0, A knows that $b_{\mathsf{ID}^*} = 0$, otherwise $b_{\mathsf{ID}^*} = 1$. (Since S has only the key corresponding to $H(\mathsf{ID}^*||b_{\mathsf{ID}^*})$). Now A will just ask the random oracle query related to the encryption corresponding to $H(\mathsf{ID}^*||b_{\mathsf{ID}^*})$ (and never to that of $H(\mathsf{ID}^*||\bar{b}_{\mathsf{ID}^*})$, the one with the underlying problem instance embedded), resulting in the failure of S.

Even worse, the following adversary C can successfully attack in the real chosen-ciphertext game. Similar to the adversary A, C will be able to distinguish b^*_{ID}. Then she submits $m^*_0, m^*_1, \mathsf{ID}^*$ to the challenge encryption oracle and gets back $C^*_0||C^*_1 = Enc_{H(\mathsf{ID}^*||0)}(m_\beta)||Enc_{H(\mathsf{ID}^*||1)}(m_\beta)$ and will try to guess β. Knowing b_{ID^*}, she now asks the decryption oracle for the ciphertext $C^*_0||Enc_{H(\mathsf{ID}^*||1)}(m')$ if $b^*_{\mathsf{ID}} = 0$ and $Enc_{H(\mathsf{ID}^*||0)}(m')||C^*_1$ if $b^*_{\mathsf{ID}} = 1$ for some $m' \neq m^*_0, m^*_1$. Clearly, this is a legitimate query since neither is equal to $C^*_0||C^*_1$. She will get m_β which means that she successfully attacks the scheme.

We now conclude the above discussion. On one hand, the technique of double encryption in which

Scheme	Security						Model	Size Efficiency		
	Basic: IND-X-ID-CPA			Full: IND-ID-CCA						
	X	Assume	Reduction	Conversion	Overall Reduc			Pub	Priv	Cipher
BF01 [5]	Adpt	CBDH	$O(\frac{1}{q_{ext} \cdot q_{H_1}})$	FO99 [11]	$O(\frac{1}{q_{ext} \cdot q_{H_1} \cdot q_{H_2}})$		RO	$O(1)$	$O(1)$	$O(1)$
CHK03 [7]	Sel	DBDH	1	CHK04 [8]	$\frac{1}{n}$		ST	$O(\log n)$	$O(\log n)$	$O(\log n)$
BB04(E1) [2]	Sel	DBDH	1	CHK04 [8]	$\frac{1}{n}$		ST	$O(1)$	$O(1)$	$O(1)$
BB04(E2) [2]	Sel	DBDH-inv	1	-	-		ST	$O(1)$	$O(1)$	$O(1)$
BB04(C) [4]	Adpt	DBDH	$O(\frac{1}{q_{ext}^2})$	CHK04 [8]	$O(\frac{1}{q_{ext}^2})$		ST	$O(\log^2 n)$	$O(\log^2 n)$	$O(\log^2 n)$
W05 [18]	Adpt	DBDH	$O(\frac{1}{q_{ext} \log n})$	CHK04 [8]	$O(\frac{1}{q_{ext} \log n})$		ST	$O(\log n)$	$O(1)$	$O(1)$
Ours 1	Adpt	GBDH	$\rightarrow$	-	$O(1)$		RO	$O(1)$	$O(1)$	$O(1)$
Ours 2,3	Adpt	DBDH	$\rightarrow$	-	$O(1)$		RO	$O(1)$	$O(1)$	$O(1)$

n: the size of identities' space; q_{ext}, q_H : the no. of queries to the extraction oracle and random oracle H respectively. RO is for random oracle; ST is for the standard model; Adapt is for the adaptive-ID model; Sel is for selective-ID model. CBDH, DBDH, GBDH are computational, decisional, gap BDH assumptions respectively.

Table 1: Comparison among existing schemes and our schemes.

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 know about b_{ID^*} and then to successfully break the scheme. This contradictory implication of straightforward application of the Katz-Wang technique suggests that more sophisticated techniques are needed.

1.2 Our Contributions

In this paper, we present the first three efficient variants of the Boneh-Franklin IBE schemes with tight security reductions in the random oracle model. We modify the double encryption technique by seamlessly integrating with new "special-purpose" CCA conversions which also overcome the weakness in use of the Katz-Wang technique in the CCA setting. Put in other words, we construct three schemes achieving CCA security from scratch, without using any existing conversions. Our first construction is based on the Gap BDH assumption. The other two are based on the Decision BDH assumption, one of which has more compact ciphertext size while the other has a publicly verifiability of ciphertexts. All of our schemes are efficient and are the first in the literature that simultaneously achieve public parameter size, private key size, ciphertext size and reduction cost as being constant terms (see Table 1).

2 Preliminaries

We first review the model and the security notion of IBE scheme. Next, we review bilinear maps which is used in our proposed schemes. Then, we give a brief review of related computational assumptions. The definitions run parallel with [5, 6].

2.1 ID-Based Encryption: Algorithms

An IBE scheme $\mathcal{E}$ is constructed by four efficient algorithms (**Setup, Extract, Encrypt, Decrypt**).

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" (PKG).

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"*.

These algorithms must satisfy the standard consistency constraint, namely when sk is the private key generated by algorithm Extract when it is given ID as the public key, then

$$\forall M \in \mathcal{M} : \mathsf{Decrypt}(\mathsf{params}, C, sk) = M \quad \text{where } C = \mathsf{Encrypt}(\mathsf{params}, \mathsf{ID}, M)$$

2.2 Security Notion

In [5, 6], Boneh and Franklin defined chosen ciphertext security for IBE under a chosen identity attack. In their model the adversary is allowed to collude with an adversary having other ID and access a decryption oracle.

We say that an IBE scheme $\mathcal{E}$ is semantically secure against an adaptive chosen ciphertext attack under a chosen identity attack (IND-ID-CCA) if no polynomially bounded adversary $\mathcal{A}_{ibe}$ has a non-negligible advantage against the challenger in the following IND-ID-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 master-key to itself.

Phase 1: The adversary issues several 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 $q_1, \cdots, q_{i-1}$.

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 $\beta \in \{0,1\}$ and sets $C^* = \text{Encrypt}(\mathsf{Params}, \mathsf{ID}^*, M_\beta)$. It sends C^* as the challenge to the adversary.

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

- Extraction query $\langle \mathsf{ID}_i \rangle$ where $\mathsf{ID}_i \neq \mathsf{ID}^*$: Challenger responds as in Phase 1.
- Decryption query $\langle \mathsf{ID}_i, C_i \rangle \neq \langle \mathsf{ID}^*, C^* \rangle$: 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 $\mathcal{A}_{ibe}$ as an IND-ID-CCA adversary. We define adversary $\mathcal{A}_{ibe}$'s advantage in attacking the scheme $\mathcal{E}$ as:

$$Adv_{\mathcal{E},\mathcal{A}_{ibe}} = \Pr[\beta = \beta'] - 1/2$$

The provability is over the random bits used by the challenger and the adversary.

Definition 1. We say that the IBE system $\mathcal{E}$ is $(t_{ibe}, \epsilon_{ibe})$-adaptive chosen ciphertext secure under a chosen identity attack if for any t_{ibe}-time IND-ID-CCA adversary $\mathcal{A}_{ibe}$, we have $Adv_{\mathcal{E},\mathcal{A}_{ibe}} < \epsilon_{ibe}$. As shorthand, we say that $\mathcal{E}$ is IND-ID-CCA secure.

2.3 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* $e : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ satisfies the following properties:

1. bilinearity: For all $u, v \in \mathbb{G}_1$ and $a, b \in \mathbb{Z}$, $e(u^a, v^b) = e(u, v)^{ab}$.
2. non-degeneracy: $e(g, g) \neq 1$.
3. computability: There is an efficient algorithm to compute $e(u, v)$ for any $u, v \in \mathbb{G}_1$.

Note that a bilinear map is symmetric since $e(g^a, g^b) = e(g^b, g^a) = e(g, g)^{ab}$.

2.4 Complexity Assumptions

Here, we review three complexity assumptions related to bilinear maps: the Computational Bilinear Diffie-Hellman (CBDH) assumption, the Decision Bilinear Diffie-Hellman (DBDH) assumption, and the Gap Bilinear Diffie-Hellman (GBDH) assumption. Here, we let $\mathbb{G}_1$ and $\mathbb{G}_2$ be two groups of order q and g be a generator of $\mathbb{G}_1$.

CBDH Assumption. The CBDH problem [5] in $\mathbb{G}_1$ is as follows: given a tuple $(g, g^a, g^b, g^c) \in (\mathbb{G}_1)^4$ as input, output $e(g, g)^{abc} \in \mathbb{G}_2$. An algorithm $\mathcal{A}_{cbdh}$ solves CBDH problem in $\mathbb{G}_1$ with the probability ϵ_{cbdh} if

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

where the probability is 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 $\mathcal{A}_{cbdh}$.

Definition 2. We say that the $(t_{cbdh}, \epsilon_{cbdh})$-*CBDH assumption* holds in $\mathbb{G}_1$ if no t_{cbdh}-time algorithm has advantage at least ϵ_{cbdh} in solving the CBDH problem in $\mathbb{G}_1$.

DBDH Assumption. The DBDH problem in $\mathbb{G}_1$ 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 $b \in \{0, 1\}$. An algorithm $\mathcal{A}_{dbdh}$ solves DBDH problem in $\mathbb{G}_1$ with advantage ϵ_{dbdh} if

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

where the probability is 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 $\mathcal{A}_{dbdh}$.

Definition 3. We say that the $(t_{dbdh}, \epsilon_{dbdh})$-*DBDH assumption* holds in $\mathbb{G}_1$ if no t_{dbdh}-time algorithm has advantage at least ϵ_{dbdh} in solving the DBDH problem in $\mathbb{G}_1$.

GBDH Assumption. The GBDH problem in $\mathbb{G}_1$ is 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 [15]. An algorithm $\mathcal{A}_{gbdh}$ solves GBDH problem in $\mathbb{G}_1$ with the probability ϵ_{gbdh} if

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

where the probability is 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 $\mathcal{A}_{gbdh}$.

Definition 4. We say that the $(t_{gbdh}, \epsilon_{gbdh})$-*GBDH assumption* holds in $\mathbb{G}_1$ if no t_{gbdh}-time algorithm has advantage at least ϵ_{gbdh} in solving the GBDH problem in $\mathbb{G}_1$.

3 Secure IBE Construction Based on the GBDH Assumption

In this section, we present an IBE scheme TightIBE1 whose IND-ID-CCA security can be reduced tightly to the difficulty of the GBDH problem in the random oracle model. Intuitively, the sender will doubly encrypt the message M using the Boneh-Franklin scheme to the identities $H(\mathsf{ID}||b_{\mathsf{ID}})$ and $H(\mathsf{ID}||\bar{b}_{\mathsf{ID}})$ in such a way that the receiver who has only one out of the two keys, say for the identity $H(\mathsf{ID}||b_{\mathsf{ID}})$, can decrypt *both* encryptions so that she can check the consistencies of the messages (and in particular, this overcomes the weakness of the straightforward application of Katz-Wang technique as described in Section 1.1). This sounds paradoxical in the first place since she does not know the other key, i.e., that of the identity $H(\mathsf{ID}||\bar{b}_{\mathsf{ID}})$. However we can manage to do this by encrypting also the "randomness" used in the other encryption simultaneously.

3.1 Construction: TightIBE1

Let $\mathbb{G}_1$ and $\mathbb{G}_2$ be two groups of order q (whose size is k) 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 $G, H, \hat{H}$ be cryptographic hash functions $G : \mathbb{G}_2 \to \{0,1\}^{n+k}$ for some n, $H : \{0,1\}^* \to \mathbb{G}_1$, $\hat{H} : \{0,1\}^* \to \{0,1\}^{k_1}$ for some k_1 respectively. The message space is $\mathcal{M} = \{0,1\}^n$. Let $a\|b$ denote the concatenation of a and b, and $a \oplus b$ denote the exclusive-OR of a and b. The TightIBE1 scheme consists of the four algorithms which are shown in Table 2.

3.2 Security

Now, we prove that the security of TightIBE1 can be tightly reduced to the GBDH assumption.

Theorem 1. *Suppose* $(t_{gbdh}, \epsilon_{gbdh})$*-GBDH assumption holds in* $\mathbb{G}_1$*. Suppose the hash functions* $G, H, \hat{H}$ *are random oracles. Then,* TightIBE1 *is* $(t_{ibe}, \epsilon_{ibe})$-IND-ID-CCA *secure such that*

$$\begin{aligned} \epsilon_{ibe} &\leq \epsilon_{gbdh} + \frac{q_D}{2^{k_1+1}} \\ t_{ibe} &\leq t_{gbdh} - \Theta(\tau(2q_H + 3q_E + 9q_D)) \end{aligned}$$

as long as IND-ID-CCA *adversary* $\mathcal{A}_{ibe}$ *makes at most* q_H H*-queries,* q_D *Decryption queries, and* q_E *Extraction queries. Here,* τ *is the maximum time among time for computing an exponentiation in* $\mathbb{G}_1, \mathbb{G}_2$*, and pairing* e*.*

Proof. We show how to construct an algorithm $\mathcal{A}_{gbdh}$ that solves the GBDH problem in $\mathbb{G}_1$ by using an adversary $\mathcal{A}_{ibe}$ that breaks IND-ID-CCA security of our scheme. The algorithm $\mathcal{A}_{gbdh}$ is given an instance $\langle g, g^a, g^b, g^c \rangle$ in $\mathbb{G}_1$ from the challenger and try to output $e(g,g)^{abc}$ using $\mathcal{A}_{ibe}$ and the DBDH oracle $\mathcal{O}$. Let $g_1 = g^a, g_2 = g^b, g_3 = g^c$. The algorithm $\mathcal{A}_{gbdh}$ works by interacting with $\mathcal{A}_{ibe}$ and $\mathcal{O}$ in an IND-ID-CCA game as follows:

TightIBE1	
Setup (1^k)**:** $s \leftarrow \mathbb{Z}_q^*$; $g_{pub} := g^s$ $\mathsf{params} := \langle q, \mathbb{G}_1, \mathbb{G}_2, e, n, g, g_{pub}, G, H, \hat{H} \rangle$ $\mathsf{master\text{-}key} := s$ *return* (params, master-key)	**Extract**† (ID, params, master-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 $(\mathsf{ID}, \mathsf{params}, M)$**:** $h_{\mathsf{ID}\|0} := H(\mathsf{ID}\|0)$; $h_{\mathsf{ID}\|1} := H(\mathsf{ID}\|1)$ $r_0, r_1 \leftarrow \mathbb{Z}_q^*$ $W_0 := e(g_{pub}, h_{\mathsf{ID}\|0})^{r_0}$ $W_1 := e(g_{pub}, h_{\mathsf{ID}\|1})^{r_1}$ $c_0 := \langle g^{r_0},\ G(W_0) \oplus (M\|r_1) \rangle$ $c_1 := \langle g^{r_1},\ G(W_1) \oplus (M\|r_0) \rangle$ $c_{\hat{H}} := \hat{H}(W_0, W_1, \mathsf{ID}, M, c_0, c_1)$ $C := (c_0, c_1, c_{\hat{H}})$ *return* C	**Decrypt** $(C, \mathsf{params}, sk_{\mathsf{ID}})$**:** parse $C = (\langle u_0, v_0 \rangle, \langle u_1, v_1 \rangle, \alpha)$ $W'_{b_{\mathsf{ID}}} := e(u_{b_{\mathsf{ID}}}, d_{\mathsf{ID}})$ $(M_{b_{\mathsf{ID}}}\|r_{\bar{b}_{\mathsf{ID}}}) := v_{b_{\mathsf{ID}}} \oplus G(W'_{b_{\mathsf{ID}}})$ $W'_{\bar{b}_{\mathsf{ID}}} := e(g_{pub}, h_{\mathsf{ID}\|\bar{b}_{\mathsf{ID}}})^{r_{\bar{b}_{\mathsf{ID}}}}$ $(M_{\bar{b}_{\mathsf{ID}}}\|r_{b_{\mathsf{ID}}}) := v_{\bar{b}_{\mathsf{ID}}} \oplus G(W'_{\bar{b}_{\mathsf{ID}}})$ if $M_{b_{\mathsf{ID}}} \neq M_{\bar{b}_{\mathsf{ID}}} \vee u_{b_{\mathsf{ID}}} \neq g^{r_{b_{\mathsf{ID}}}} \vee u_{\bar{b}_{\mathsf{ID}}} \neq g^{r_{\bar{b}_{\mathsf{ID}}}} \vee$ $\alpha \neq \hat{H}(W'_0, W'_1, \mathsf{ID}, M_{b_{\mathsf{ID}}}, c_0, c_1)$ *return* "*reject*" else $M := M_{b_{\mathsf{ID}}} (= M_{\bar{b}_{\mathsf{ID}}})$ *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 algorithms of TightIBE1

Setup: $\mathcal{A}_{gbdh}$ picks a random $\mu \in \mathbb{Z}_q^*$. Also, $\mathcal{A}_{gbdh}$ gives $\mathcal{A}_{ibe}$ the system parameter

$$\mathsf{params} = \langle q, \mathbb{G}_1, \mathbb{G}_2, e, n, g, g_1, G, H, \hat{H} \rangle.$$

Here, random oracles $G, H, \hat{H}$ are controlled by $\mathcal{A}_{gbdh}$ as described below.

G-queries: $\mathcal{A}_{ibe}$ issues up to q_G queries to the random oracle G. To respond to these queries algorithm $\mathcal{A}_{gbdh}$ forms a list of tuples $\langle W, x \rangle$ as explained below. We call this list G_{list}. The list is initially empty. When $\mathcal{A}_{ibe}$ gives $\mathcal{A}_{gbdh}$ a query W to the oracle G, $\mathcal{A}_{gbdh}$ responds as follows:

1. If the query W already appears on the G_{list} in a tuple $\langle W, x \rangle$, then outputs $G(W) = x$.
2. The algorithm $\mathcal{A}_{gbdh}$ performs the two verifications as follows:
   (i) $\mathcal{A}_{ibe}$ submits (g, g_1, g_2, g_3, W) to $\mathcal{O}$. If $\mathcal{O}$'s answer is *true*, $\mathcal{A}_{gbdh}$ outputs W as the solution of the given instance of GBDH problem, and *aborts the simulation.*
   (ii) $\mathcal{A}_{gbdh}$ checks whether $W = e(g_1, g)^{\mu}$ or not. If it is *true*, then *aborts the simulation.*
3. $\mathcal{A}_{gbdh}$ chooses a random $x \in \{0,1\}^{n+k}$.
4. $\mathcal{A}_{gbdh}$ stores the tuple $\langle W, x \rangle$ to the G_{list} and outputs $G(W) = x$.

H-queries: $\mathcal{A}_{ibe}$ issues up to q_H queries to the random oracle H. To respond to these queries algorithm $\mathcal{A}_{gbdh}$ forms a list of tuples $\langle \mathsf{ID}, b_{\mathsf{ID}}, h_{\mathsf{ID}\|\bar{b}_{\mathsf{ID}}}, h_{\mathsf{ID}\|b_{\mathsf{ID}}}, r_{\mathsf{ID}}, t_{\mathsf{ID}} \rangle$ as explained below. We call the list H_{list}. The list is initially empty. When $\mathcal{A}_{ibe}$ give $\mathcal{A}_{gbdh}$ a query $(\mathsf{ID}\|b)$ to the oracle H, $\mathcal{A}_{gbdh}$ responds as follows:

1. If the query ID already appears on the H_{list} in a tuple $\langle \mathsf{ID}, b_{\mathsf{ID}}, h_{\mathsf{ID}\|\bar{b}_{\mathsf{ID}}}, h_{\mathsf{ID}\|b_{\mathsf{ID}}}, r_{\mathsf{ID}}, t_{\mathsf{ID}} \rangle$, then outputs $H(\mathsf{ID}\|b) = h_{\mathsf{ID}\|b_{\mathsf{ID}}}$ if $b = b_{\mathsf{ID}}$ and $H(\mathsf{ID}\|b) = h_{\mathsf{ID}\|\bar{b}_{\mathsf{ID}}}$ otherwise.

2. $\mathcal{A}_{gbdh}$ picks a random bit $b_{\mathsf{ID}} \in \{0,1\}$ and chooses random $r_{\mathsf{ID}}, t_{\mathsf{ID}} \in \mathbb{Z}_q^*$.
3. $\mathcal{A}_{gbdh}$ computes $(h_{\mathsf{ID}\|\bar{b}_{\mathsf{ID}}}, h_{\mathsf{ID}\|b_{\mathsf{ID}}}) = (g_2^{r_{\mathsf{ID}}}, g^{t_{\mathsf{ID}}})$.
4. $\mathcal{A}_{gbdh}$ stores the tuple $\langle \mathsf{ID}, b_{\mathsf{ID}}, h_{\mathsf{ID}\|\bar{b}_{\mathsf{ID}}}, h_{\mathsf{ID}\|b_{\mathsf{ID}}}, r_{\mathsf{ID}}, t_{\mathsf{ID}} \rangle$ to the H_{list} and outputs $H(\mathsf{ID}\|b) = h_{\mathsf{ID}\|b_{\mathsf{ID}}}$ if $b = b_{\mathsf{ID}}$ and $H(\mathsf{ID}\|b) = h_{\mathsf{ID}\|\bar{b}_{\mathsf{ID}}}$ otherwise.

$\hat{H}$-queries: $\mathcal{A}_{ibe}$ issues up to $q_{\hat{H}}$ queries to the random oracle $\hat{H}$. To respond to these queries algorithm $\mathcal{A}_{gbdh}$ forms a list of tuples $\langle W_0, W_1, \mathsf{ID}, M, c_0, c_1, \gamma \rangle$ as described below. We call the list $\hat{H}_{list}$ and it is initially empty. When $\mathcal{A}_{ibe}$ give $\mathcal{A}_{gbdh}$ a query $(W_0, W_1, \mathsf{ID}, M, c_0, c_1)$ to the oracle $\hat{H}$, algorithm $\mathcal{A}_{gbdh}$ responds as follows:

1. If the query $(W_0, W_1, \mathsf{ID}, M, c_0, c_1)$ already appears on the $\hat{H}_{list}$ in a tuple $\langle W_0, W_1, \mathsf{ID}, M, c_0, c_1, \gamma \rangle$, then outputs $\hat{H}(W_0, W_1, \mathsf{ID}, M, c_0, c_1) = \gamma$.
2. The algorithm $\mathcal{A}_{gbdh}$ performs the two verifications as follows:
   (i) For $W = W_0$ and W_1, $\mathcal{A}_{ibe}$ submits (g, g_1, g_2, g_3, W) to $\mathcal{O}$. If $\mathcal{O}$'s answer is *true*, $\mathcal{A}_{gbdh}$ outputs W as the solution of the given instance of GBDH problem, and *aborts the simulation.*
   (ii) For $W = W_0$ and W_1, $\mathcal{A}_{gbdh}$ checks whether $W = e(g_1, g)^{\mu}$ or not. If it is *true*, then *aborts the simulation.*
3. $\mathcal{A}_{gbdh}$ chooses a random $\gamma \in \{0,1\}^{k_1}$.
4. $\mathcal{A}_{gbdh}$ stores the tuple $\langle W_0, W_1, \mathsf{ID}, M, c_0, c_1, \gamma \rangle$ to the $\hat{H}_{list}$ and output $\hat{H}(W_0, W_1, \mathsf{ID}, M, c_0, c_1) = \gamma$.

Extraction queries: $\mathcal{A}_{ibe}$ issues up to q_E Extraction queries. The simulator behaves same in both Phase 1 and Phase 2. When $\mathcal{A}_{ibe}$ gives a query ID, $\mathcal{A}_{gbdh}$ responds as follows:

1. $\mathcal{A}_{gbdh}$ runs the algorithm for responding to H-queries to obtain b_{ID} and t_{ID}.
2. $\mathcal{A}_{gbdh}$ sets $sk_{\mathsf{ID}} = ((g_1)^{t_{\mathsf{ID}}}, b_{\mathsf{ID}})$. Observe that $sk_{\mathsf{ID}} = ((g^{t_{\mathsf{ID}}})^a, b_{\mathsf{ID}})$ and therefore sk_{ID} is the private key corresponding to the ID.
3. $\mathcal{A}_{gbdh}$ outputs sk_{ID} to $\mathcal{A}_{ibe}$.

Decryption queries: $\mathcal{A}_{ibe}$ issues up to q_D Decryption queries. The simulator behaves same in both Phase 1 and Phase 2. When $\mathcal{A}_{ibe}$ gives a query (ID, C), $\mathcal{A}_{gbdh}$ responds as follows:

1. $\mathcal{A}_{gbdh}$ runs the algorithm for responding to Extraction queries to obtain the private key sk_{ID} corresponding to the ID.
2. Using the private key sk_{ID}, $\mathcal{A}_{gbdh}$ decrypts C.
3. $\mathcal{A}_{gbdh}$ outputs the result.

Challenge: Once algorithm $\mathcal{A}_{ibe}$ decides that Phase 1 is over, it outputs public key ID^* and two messages M_0, M_1 on which it wishes to be challenged. Algorithm $\mathcal{A}_{gbdh}$ responds as follows:

1. $\mathcal{A}_{gbdh}$ runs the algorithm for responding to H-queries to obtain r_{ID^*} and t_{ID^*} such that $H(\mathsf{ID}^* || \bar{b}_{\mathsf{ID}^*}) = g_2^{r_{\mathsf{ID}^*}}$ and $H(\mathsf{ID}^* || b_{\mathsf{ID}^*}) = g^{t_{\mathsf{ID}^*}}$.
2. $\mathcal{A}_{gbdh}$ sets $\mu^* = \mu \cdot t_{\mathsf{ID}^*}^{-1} \bmod q$, $\omega_0, \omega_1 \in_R \{0,1\}^{n+k}$ and $\alpha^* \in_R \{0,1\}^{k_1}$.

3. $\mathcal{A}_{gbdh}$ sets $c^*_{\bar{b}_{\mathsf{ID}^*}}, c^*_{b_{\mathsf{ID}^*}}, c^*_{\hat{H}}$ as follows:

$$\begin{aligned} c^*_{\bar{b}_{\mathsf{ID}^*}} &= \langle (g_3)^{r^{-1}_{\mathsf{ID}^*}}, \omega_0 \rangle \\ c^*_{b_{\mathsf{ID}^*}} &= \langle g^{\mu^*}, \omega_1 \rangle \\ c^*_{\hat{H}} &= \alpha^* \end{aligned}$$

where $r^{-1}_{\mathsf{ID}^*}$ is the inverse of r_{ID^*} mod q.

4. $\mathcal{A}_{gbdh}$ gives $C^* = \langle c^*_0, c^*_1, c^*_{\hat{H}} \rangle$ as the challenge ciphertext to $\mathcal{A}_{ibe}$.

Guess: When $\mathcal{A}_{ibe}$ decides that Phase 2 is over, $\mathcal{A}_{ibe}$ outputs its guess bit $\beta' \in \{0,1\}$. At the same time, algorithm $\mathcal{A}_{gbdh}$ terminates the simulation.

Claim 1. *If $\mathcal{A}_{gbdh}$ does not abort during the simulation, then algorithm $\mathcal{A}_{ibe}$'s view is identical to its view in the real attack. Furthermore, if $\mathcal{A}_{gbdh}$ does not abort then $Pr[\beta' = \beta] \geq 1/2 + \epsilon_{ibe}$.*

Proof. It is obvious that the responses to G, $\hat{H}$ queries are perfect. The responses to H are also as in the real attack since each response is uniformly and independently distributed in $\mathbb{G}_1^*$. Interestingly, the responses to Extraction and Decryption queries are perfect as well. Notice that $\mathcal{A}_{gbdh}$ can generate any user's private key including ID^*'s, and furthermore, can decrypt any ciphertext for any user by using the private keys. Finally, we show that the response to Challenge is perfect if $\mathcal{A}_{gbdh}$ does not *abort* the simulation. Let the response to Challenge be $C^* = (\langle u^*_0, v^*_0 \rangle, \langle u^*_1, v^*_1 \rangle, \alpha^*)$. Then, both u^*_0 and u^*_1 are uniformly and independently distributed in $\mathbb{G}_1^*$ due to randoms $\log_g g_3$ and μ, and therefore are as in the real attack. Obviously, v^*_0, v^*_1 and α^* are perfect. Therefore, by definition of $\mathcal{A}_{ibe}$, we have that $Pr[\beta' = \beta] \geq 1/2 + \epsilon_{ibe}$. □

Next, let us define by E_1 an event assigned to be true if and only if a G-query coincides with $e(g,g)^{abc}$ and by E_2 an event assigned to be true if and only if a G-query coincides with $e(g_1,g)^{\mu}$. Similarly, let us define by E_3 an event assigned to be true if and only if a $\hat{H}$-query coincides with $(e(g,g)^{abc}, *, *, *, *, *)$ or $(*, e(g,g)^{abc}, *, *, *, *)$ and by E_4 an event assigned to be true if and only if a $\hat{H}$-query coincides with $(e(g_1,g)^{\mu}, *, *, *, *, *)$ or $(*, e(g_1,g)^{\mu}, *, *, *, *)$, where $*$ denotes any bit string. We also define $E = E_1 \vee E_2 \vee E_3 \vee E_4$.

Claim 2. *We have that* $\Pr[\beta' = \beta | \neg E] \leq 1/2 + q_D/2^{k_1+1}$.

Proof. Since M_β is concealed by one-time pad, it is impossible to obtain any information of M_β unless $\mathcal{A}_{ibe}$ sends a query whose response contains information of $G(e(g,g)^{abc})$ or $G(e(g_1,g)^{\mu})$. There are only two ways to make such a query, that is, (i) submit $e(g,g)^{abc}$ or $e(g_1,g)^{\mu}$ directly to G, or (ii) submit a Decryption query such that $\mathcal{A}_{gbdh}$ by itself has to calculate $G(e(g,g)^{abc})$ or $G(e(g_1,g)^{\mu})$ and return a decryption result based on it.

Hence, if $E = \mathit{false}$, $\mathcal{A}_{ibe}$'s best strategy for guessing β is to observe a response to a Decryption query $(\mathsf{ID}, \langle \langle u_0, v_0 \rangle, \langle u_1, v_1 \rangle, \alpha \rangle)$ such that at least one of $e(h_{\mathsf{ID}||0}, u_0)^a$ and $e(h_{\mathsf{ID}||1}, u_1)^a$ is identical to $e(g,g)^{abc}$ or $e(g_1,g)^{\mu}$ without submitting $(e(g,g)^{abc}, *, *, *, *, *)$, $(*, e(g,g)^{abc}, *, *, *, *)$, $(e(g_1,g)^{\mu}, *, *, *, *, *)$ nor $(*, e(g_1,g)^{\mu}, *, *, *, *)$ to $\hat{H}$. Then, it is clear that the response to the query will be *reject* with probability $1 - 1/2^{k_1}$ since $\mathcal{A}_{ibe}$ has to guess α uniformly at random from $\{0,1\}^{k_1}$. Obviously, *reject* gives no information on the plaintext. Hence, $\mathcal{A}_{ibe}$ obtains no information with probability at least $(1-1/2^{k_1})^{q_D}$. Hence, we have

$$\begin{aligned} \Pr[\beta' = \beta | \neg E] &\leq \frac{1}{2} \cdot (1 - \frac{1}{2^{k_1}})^{q_D} + 1 \cdot (1 - (1 - \frac{1}{2^{k_1}})^{q_D}) \\ &\leq \frac{1}{2} + \frac{q_D}{2^{k_1+1}}, \end{aligned}$$

which proves the claim.

□

Claim 3. *We have that* $\Pr[E_1 \vee E_3] = \Pr[E_2 \vee E_4]$.

Proof. From the symmetricity, it is sufficient to prove that $\mathcal{A}_{ibe}$ can distinguish the value of b_{ID^*} with probability 1/2. Now, we prove that $\mathcal{A}_{ibe}$'s view is independent on the value of b_{ID} for any ID.

Since leakage of information on b_{ID} may occur only from responses to Decryption queries, 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}. Next, we prove this by contradiction. Assume that $C = \langle\langle u_0, v_0\rangle, \langle u_1, v_1\rangle, \alpha\rangle$ be a ciphertext such that $\mathbf{Decrypt}(\mathsf{params}, C, sk_0) \neq \mathbf{Decrypt}(\mathsf{params}, C, sk_1)$, where $sk_b = ((h_{\mathsf{ID}||b})^s, b)$, $b \in \{0,1\}$. Without loss of generality, we assume that $\mathbf{Decrypt}(\mathsf{params}, C, sk_0) = M(\neq \mathit{reject})$. Then, we have $\langle u_0, v_0\rangle = \langle g^{r_0}, G(e(g_{pub}, h_{\mathsf{ID}||0})^{r_0}) \oplus (M||r_1)\rangle$ for some $r_0, r_1 \in \mathbb{Z}_q^*$. Since $M \neq \mathit{reject}$, the following equations hold: $M' = M$, $u_0 = g^{r_0'}$, $u_1 = g^{r_1}$, and $\alpha = \hat{H}(e(u_0, (h_{\mathsf{ID}||0})^s), e(g_{pub}, h_{\mathsf{ID}||1})^{r_1}, M, \langle u_0, v_0\rangle, \langle u_1, v_1\rangle)$, where $(M'||r_0') = v_1 \oplus G(e(g_{pub}, h_{\mathsf{ID}||1})^{r_1})$. This means that $\langle u_1, v_1\rangle = \langle g^{r_1}, G(e(g_{pub}, h_{\mathsf{ID}||1})^{r_1}) \oplus (M||r_0)\rangle$, and it is obvious that $\mathbf{Decrypt}(\mathsf{params}, C, sk_1) = M$ which is a contradition. □

Finally, we calculate ϵ_{gbdh} from the above claims. From Claims 1 and 2, we have

$$\begin{aligned}
\Pr[\beta' = \beta] &= \Pr[\beta' = \beta | \neg E] \Pr[\neg E] + \Pr[\beta' = \beta | E] \Pr[E] \\
&\leq (\frac{1}{2} + \frac{q_D}{2^{k_1+1}})(1 - \Pr[E]) + \Pr[E] \\
&\leq \frac{1}{2} + \frac{1}{2}\Pr[E] + \frac{q_D}{2^{k_1+1}}.
\end{aligned}$$

From Claim 3, we have $\Pr[E_1 \vee E_3] \geq 1/2 \Pr[E]$, and therefore,

$$\epsilon_{gbdh} = \Pr[E_1 \vee E_3] \geq \frac{1}{2}\Pr[E].$$

Hence, we have that

$$\Pr[\beta' = \beta] \leq \frac{1}{2} + \epsilon_{gbdh} + \frac{q_D}{2^{k_1+1}},$$

and consequently,

$$\epsilon_{ibe} \leq \epsilon_{gbdh} + \frac{q_D}{2^{k_1+1}}.$$

From above discussions, it is easily seen that the claimed bound of the running-time of $\mathcal{A}_{gbdh}$ holds. This completes the proof of Theorem 1. □

4 Secure IBE Constructions Based on the DBDH Assumption

In this section, we present two IBE schemes TightIBE2 and TightIBE3 whose security is tightly reduced to the difficulty of the DBDH problem.

4.1 Construction: TightIBE2

IND-ID-CCA security of IBE scheme TightIBE2 can be reduced to the DBDH problem in the random oracle model. TightIBE2 adapts similar technique of TightIBE1, however, distinguishes itself from TightIBE1 in performance: its enjoys more compact ciphertext size when the message is short. Moreover, it is more computational efficient since it needs only 2 pairings and 2 scalar multiplications in encryption and 2 pairings and 2 scalar multiplications in decryption. The computation time is even comparable to the original Boneh-Franklin scheme, if parallel computation is considered.

Let $\mathbb{G}_1$ and $\mathbb{G}_2$ be two groups of order q (whose size is k) 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 $k = k_1 + k_2$, $G : \{0,1\}^* \to (Z_q^*)^2$, $H : \{0,1\}^* \to \mathbb{G}_1$ be two cryptographic hash functions, $\phi : \{0,1\}^k \to \mathbb{G}_2$ be one-to-one function that its inverse is efficiently computable. The message space is $\mathcal{M} = \{0,1\}^{k_1}$. The TightIBE2 scheme consists of the algorithms which are shown in Table 3.

TightIBE2	
Setup (1^k)**:** $s \leftarrow \mathbb{Z}_q^*$; $g_{pub} := g^s$ params $:= \langle q, \mathbb{G}_1, \mathbb{G}_2, e, k_1, g, g_{pub}, G, H, \phi \rangle$ master-key $:= s$ *return* (params, master-key)	**Extract**† (ID, params, master-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 $(\mathsf{ID}, \mathsf{params}, M)$**:** $h_{\mathsf{ID}\|0} := H(\mathsf{ID}\|0)$; $h_{\mathsf{ID}\|1} := H(\mathsf{ID}\|1)$ $R \leftarrow \{0,1\}^{k_2}$ $\bar{M} := \phi(M\|R)$ $(r_0\|r_1) := G(\mathsf{ID}, M, R)$ $c_0 := \langle g^{r_0}, \bar{M} \cdot e(g_{pub}, h_{\mathsf{ID}\|0})^{r_0} \rangle$ $c_1 := \langle g^{r_1}, \bar{M} \cdot e(g_{pub}, h_{\mathsf{ID}\|1})^{r_1} \rangle$ $C := (c_0, c_1)$ *return* C	**Decrypt** $(C, \mathsf{params}, sk_{\mathsf{ID}})$**:** parse $C = (\langle u_0, v_0 \rangle, \langle u_1, v_1 \rangle)$ $(M'\|\|R') := \phi^{-1}(v_{b_{\mathsf{ID}}} \cdot e(u_{b_{\mathsf{ID}}}, d_{\mathsf{ID}})^{-1})$ $(r'_0\|\|r'_1) := G(\mathsf{ID}, M', R')$ if $c_0 \neq \langle g^{r'_0}, \phi(M'\|\|R') \cdot e(g_{pub}, h_{\mathsf{ID}\|0})^{r'_0} \rangle \vee$ $c_1 \neq \langle g^{r'_1}, \phi(M'\|\|R') \cdot e(g_{pub}, h_{\mathsf{ID}\|1})^{r'_1} \rangle$ *return* "*reject*" else *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 TightIBE2

Theorem 2. *Suppose* $(t_{dbdh}, \epsilon_{dbdh})$*-DBDH assumption holds in* $\mathbb{G}_1$*. Suppose the hash functions* G, H *are random oracles. Then,* TightIBE2 *is* $(t_{ibe}, \epsilon_{ibe})$-IND-ID-CCA *secure such that*

$$\begin{aligned} \epsilon_{ibe} &\leq 2\epsilon_{dbdh} + \frac{q_G}{2^{k_2}} \\ t_{ibe} &\leq t_{dbdh} - \Theta(\tau(2q_H + 3q_E + 9q_D)) \end{aligned}$$

as long as IND-ID-CCA *adversary* $\mathcal{A}_{ibe}$ *makes at most* q_G *G-queries,* q_H *H-queries,* q_D *Decryption queries, and* q_E *Extraction queries. Here,* τ *is the maximum time among time for computing an exponentiation in* $\mathbb{G}_1, \mathbb{G}_2$*, and pairing* e*.*

The proof of Theorem 2 is given in Appendix A.

4.2 Construction: TightIBE3

Next, we propose a public verifiable IBE scheme TightIBE3 whose IND-ID-CCA security can be reduced tightly to the decision bilinear Diffie-Hellman assumption in the random oracle model. This is a

straightforward application of the idea given in [13] to the double encryption paradigm proposed in [14] with non-malleable and non-interactive zero-knowledge proof in the random oracle model.

Public verifiability of ciphertexts is especially effective in cases when its private key is distributed among a multiple of players and these players are required to collaborately prove the validity/invalidity of the ciphertext when they decrypt/reject it. Such a case can be found in some of mix-net schemes. These collaborations are tends to be complex task if ciphertexts are not publicly verifiable.

Let $\mathbb{G}_1$ and $\mathbb{G}_2$ be two groups of order q (whose size is k) 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 $H : \{0,1\}^* \to \mathbb{G}_1$ and $G : \{0,1\}^* \to \mathbb{Z}_q$ be a cryptographic hash functions. The message space $\mathcal{M} = \mathbb{G}_2$. The TightIBE3 scheme consists of the algorithms which are shown in Table 4.

TightIBE3	
Setup (1^k)**:** $s \leftarrow \mathbb{Z}_q^*$; $g_{pub} := g^s$ params $:= \langle q, \mathbb{G}_1, \mathbb{G}_2, e, g, g_{pub}, G, H \rangle$ master-key $:= s$ *return* (params, master-key)	**Extract**† (ID, params, master-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 (ID, params, M)**:** $h_{\mathsf{ID}\|0} := H(\mathsf{ID}\|0)$; $h_{\mathsf{ID}\|1} := H(\mathsf{ID}\|1)$ $r_0, r_1 \leftarrow \mathbb{Z}_q$ $c_0 = (u_0, v_0) := (g^{r_0}, M \cdot e(g_{pub}, h_{\mathsf{ID}\|0})^{r_0})$ $c_1 = (u_1, v_1) := (g^{r_1}, M \cdot e(g_{pub}, h_{\mathsf{ID}\|1})^{r_1})$ $s_0, s_1, r_M, s_M \leftarrow \mathbb{Z}_q$ $c_M := (1/M) \cdot e(g,g)^{r_M}$ $u'_0 := g^{s_0}$ $u'_1 := g^{s_1}$ $v'_0 := e(g_{pub}, h_{\mathsf{ID}\|0})^{s_0} \cdot e(g,g)^{s_M}$ $v'_1 := e(g_{pub}, h_{\mathsf{ID}\|1})^{s_1} \cdot e(g,g)^{s_M}$ $c := G(\mathbb{G}_1, \mathbb{G}_2, g, g_{pub}, \mathsf{ID}, c_0, c_1, c_M, u'_0, u'_1, v'_0, v'_1)$ $t_0 := r_0 c + s_0$ $t_1 := r_1 c + s_1$ $t_M := r_M c + s_M$ $C := (c_0, c_1, c_M, c, t_0, t_1, t_M)$ *return* C	**Decrypt** $(C, \mathsf{params}, sk_{\mathsf{ID}})$**:** parse $C = (\langle u_0, v_0 \rangle, \langle u_1, v_1 \rangle, c_M, c, t_0, t_1, t_M)$ $e_0 := g^{t_0} {u_0}^{-c}$ $e_1 := g^{t_1} {u_1}^{-c}$ $e_2 := e(g,g)^{t_m} e(g_{pub}, h_{\mathsf{ID}\|0})^{t_0} (v_0 c_M)^{-c}$ $e_3 := e(g,g)^{t_m} e(g_{pub}, h_{\mathsf{ID}\|1})^{t_1} (v_1 c_M)^{-c}$ if $c \neq G(\mathbb{G}_1, \mathbb{G}_2, g, g_{pub}, \mathsf{ID}, c_0, c_1, c_M, e_0, e_1, e_2, e_3)$ *return* "*reject*" else $M := v_{b_{\mathsf{ID}}} \cdot e(u_{b_{\mathsf{ID}}}, d_{\mathsf{ID}})^{-1}$ *return* M Note that no secret key is required to decide whether to accept the ciphertext or not, which is the property of publicly verifiability.

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

Table 4: The algorithms of TightIBE3

Theorem 3. *Suppose* $(t_{dbdh}, \epsilon_{dbdh})$*-DBDH assumption holds in* $\mathbb{G}_1$*. Suppose the hash functions* G, H *are random oracles. Then,* TightIBE3 *is* $(t_{ibe}, \epsilon_{ibe})$-IND-ID-CCA *secure such that*

$$\begin{aligned} \epsilon_{ibe} &\leq 2\epsilon_{dbdh} + \frac{2(q_G + q_H)}{q} \\ t_{ibe} &\leq t_{dbdh} - \Theta(\tau(2q_H + 3q_E + 13q_D)) \end{aligned}$$

as long as IND-ID-CCA *adversary* $\mathcal{A}_{ibe}$ *makes at most* q_G *G-queries,* q_H *H-queries,* q_D *Decryption queries, and* q_E *Extraction queries. Here,* τ *is the maximum time among times for computing an exponentiation in* $\mathbb{G}_1, \mathbb{G}_2$*, and pairing* e*.*

The proof of Theorem 3 is given in Appendix B.

5 Conclusion

We presented three new efficient and chosen ciphertext secure identity-based encryption schemes which are the first ones in the literature that enjoy tight security reductions in the random oracle model.

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

Acknowledgement

Rui Zhang is supported by a JSPS Fellowship.

References

[1] M. Bellare and P. Rogaway, "The Game-Playing Technique," available as IACR ePrint Report 2004/331.

[2] 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.

[3] D. Boneh and X. Boyen, "Efficient Selective-ID Secure Identity Based Encryption Without Random Oracles," the full version of [2], available as IACR ePrint Report 2004/172.

[4] D. Boneh and X. Boyen, "Secure Identity Based Encryption Without Random Oracles," In Advances in Cryptology–Crypto'04, LNCS 3152, pp.443-459, 2004.

[5] D. Boneh and M. Franklin, "Identity Based Encryption from the Weil Pairing," In Advances in Cryptology–Crypto'01, LNCS 2139, pp.213-229, 2001.

[6] D. Boneh and M. Franklin, "Identity Based Encryption from the Weil Pairing," SIAM Journal of Computing 32(3):586-615, 2003, full version of [5].

[7] 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.

[8] R. Canetti, S. Halevi and J. Katz, "Chosen-Ciphertext Security from Identity Based Encryption," In Advances in Cryptology–Eurocrypt'04, LNCS 3027, pp.207-222, 2004.

[9] J.S. Coron, "On the Exact Security of Full Domain Hash," In Advances in Cryptology–Crypto'00, LNCS 1880, pp.229-235, 2000.

[10] E. Fujisaki and T. Okamoto, "How to Enhance the Security of Public-Key Encryption at Minimum Cost," Proc. of PKC'99, LNCS 1560, pp.53-68, 1999.

[11] 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.

[12] D. Galindo, "Boneh-Franklin Identity Based Encryption Revisited," to appear in Proc. of ICALP'05, available as IACR ePrint Report 2005/117.

[13] J. Katz and N. Wang, "Efficiency Improvements for Signature Schemes with Tight Security Reductions," Proc. of ACM-CCS'03, pp.155-164, 2003.

[14] M. Naor and M. Yung, "Public-key Cryptosystems Provably Secure against Chosen Ciphertext Attacks," Proc. of STOC'90, pp.427-437, 1990.

[15] T. Okamoto and D. Pointcheval, "The Gap-Problems: a New Class of Problems for the Security of Cryptographic Schemes," Proc. of PKC'01, LNCS 1992, pp.104-118, 2001.

[16] A. Shamir, "Identity-Based Cryptosystems and Signature Schemes," In Advances in Cryptology–CRYPTO'84, LNCS 293, pp.341-349, 1984.

[17] V. Shoup, "Sequences of Games: A Tool for Taming Complexity in Security Proofs," available as IACR ePrint Report 2004/332.

[18] 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.

A The Proof of Theorem 2

Proof. We show how to construct an algorithm $\mathcal{A}_{dbdh}$ that solves the DBDH problem in $\mathbb{G}_1$ by using an adversary $\mathcal{A}_{ibe}$ that breaks IND-ID-CCA security of our scheme. The algorithm $\mathcal{A}_{dbdh}$ is given an instance $\langle g, g^a, g^b, g^c, T\rangle$ in $\mathbb{G}_1$ from the challenger and tries to distinguish whether it is a valid BDH tuple or not. Let $g_1 = g^a, g_2 = g^b, g_3 = g^c$. $\mathcal{A}_{dbdh}$ works by interacting with $\mathcal{A}_{ibe}$ in an IND-ID-CCA game as follows:

Setup: $\mathcal{A}_{dbdh}$ picks a random $R_0, R_1 \in_R \{0,1\}^{k_2}$, and flips coins β and $\mathcal{COIN} \in_R \{0,1\}$. Also, $\mathcal{A}_{dbdh}$ gives $\mathcal{A}_{ibe}$ the system parameter $\mathsf{params} = \langle q, \mathbb{G}_1, \mathbb{G}_2, e, k_1, g, g_1, G, H\rangle$. Here, random oracles G, H are controlled by $\mathcal{A}_{dbdh}$ as described below.

G-queries: $\mathcal{A}_{ibe}$ issues up to q_G queries to the random oracle G. To respond to these queries algorithm $\mathcal{A}_{dbdh}$ forms a list of tuples $\langle W, x\rangle$ as explained below. We call this list G_{list}. The list is initially empty. When $\mathcal{A}_{ibe}$ gives $\mathcal{A}_{dbdh}$ a query W to the oracle G, $\mathcal{A}_{dbdh}$ responds as follows:

1. If the query W already appears on the G_{list} in a tuple $\langle W, x\rangle$, then outputs $G(W) = x$.
2. If $W = (*, *, R_\beta)$, $\mathcal{A}_{dbdh}$ outputs "$T = e(g,g)^{abc}$" and *aborts the simulation.*
3. If $W = (*, *, R_{\bar{\beta}})$ and $\mathcal{COIN} = 1$, $\mathcal{A}_{dbdh}$ outputs "$T \neq e(g,g)^{abc}$" and *aborts the simulation.*
4. $\mathcal{A}_{dbdh}$ chooses a random $x \in \mathbb{Z}_q^*$.
5. $\mathcal{A}_{dbdh}$ stores the tuple $\langle W, x\rangle$ to the G_{list} and outputs $G(W) = x$.

H-queries: $\mathcal{A}_{ibe}$ issues up to q_H queries to the random oracle H. To respond to these queries algorithm $\mathcal{A}_{dbdh}$ forms a list of tuples $\langle \mathsf{ID}, b_{\mathsf{ID}}, h_{\mathsf{ID}\|\bar{b}_{\mathsf{ID}}}, h_{\mathsf{ID}\|b_{\mathsf{ID}}}, r_{\mathsf{ID}}, t_{\mathsf{ID}}\rangle$ as explained below. We call this list H_{list}. The list is initially empty. When $\mathcal{A}_{ibe}$ gives $\mathcal{A}_{dbdh}$ a query $(\mathsf{ID}\|b)$ to the oracle H, $\mathcal{A}_{dbdh}$ responds as follows:

1. If the query ID already appears on the H_{list} in a tuple $\langle \mathsf{ID}, b_{\mathsf{ID}}, h_{\mathsf{ID}\|\bar{b}_{\mathsf{ID}}}, h_{\mathsf{ID}\|b_{\mathsf{ID}}}, r_{\mathsf{ID}}, t_{\mathsf{ID}}\rangle$, then outputs $H(\mathsf{ID}\|b) = h_{\mathsf{ID}\|b_{\mathsf{ID}}}$ if $b = b_{\mathsf{ID}}$ and $H(\mathsf{ID}\|b) = h_{\mathsf{ID}\|\bar{b}_{\mathsf{ID}}}$ otherwise.
2. $\mathcal{A}_{dbdh}$ picks a random bit $b_{\mathsf{ID}} \in \{0,1\}$ and chooses random $r_{\mathsf{ID}}, t_{\mathsf{ID}} \in \mathbb{Z}_q^*$.
3. $\mathcal{A}_{dbdh}$ computes $(h_{\mathsf{ID}\|\bar{b}_{\mathsf{ID}}}, h_{\mathsf{ID}\|b_{\mathsf{ID}}}) = (g_2^{r_{\mathsf{ID}}}, g^{t_{\mathsf{ID}}})$.
4. $\mathcal{A}_{dbdh}$ stores the tuple $\langle \mathsf{ID}, b_{\mathsf{ID}}, h_{\mathsf{ID}\|\bar{b}_{\mathsf{ID}}}, h_{\mathsf{ID}\|b_{\mathsf{ID}}}, r_{\mathsf{ID}}, t_{\mathsf{ID}}\rangle$ to the H_{list} and outputs $H(\mathsf{ID}\|b) = h_{\mathsf{ID}\|b_{\mathsf{ID}}}$ if $b = b_{\mathsf{ID}}$ and $H(\mathsf{ID}\|b) = h_{\mathsf{ID}\|\bar{b}_{\mathsf{ID}}}$ otherwise.

Extraction queries: $\mathcal{A}_{ibe}$ issues up to q_E Extraction queries. The simulator behaves same in both Phase 1 and Phase 2. When $\mathcal{A}_{ibe}$ gives a query ID, $\mathcal{A}_{dbdh}$ responds as follows:

1. $\mathcal{A}_{dbdh}$ runs the algorithm for responding to H-queries to obtain b_{ID} and t_{ID}.
2. $\mathcal{A}_{dbdh}$ sets $sk_{\mathsf{ID}} = ((g_1)^{t_{\mathsf{ID}}}, b_{\mathsf{ID}})$. Observe that $sk_{\mathsf{ID}} = ((g^{t_{\mathsf{ID}}})^a, b_{\mathsf{ID}})$ and therefore sk_{ID} is the secret key corresponding to the ID.
3. $\mathcal{A}_{dbdh}$ outputs sk_{ID} to $\mathcal{A}_{ibe}$.

Decryption queries: $\mathcal{A}_{ibe}$ issues up to q_D Decryption queries. The simulator behaves same in both Phase 1 and Phase 2. When $\mathcal{A}_{ibe}$ gives a query (ID, C), $\mathcal{A}_{dbdh}$ responds as follows:

1. $\mathcal{A}_{dbdh}$ runs the algorithm for responding to Extraction queries to obtain the secret key sk_{ID}.
2. Using the private key sk_{ID}, $\mathcal{A}_{dbdh}$ decrypts C.

3. $\mathcal{A}_{dbdh}$ outputs the result.

Challenge: Once algorithm $\mathcal{A}_{ibe}$ decides that Phase 1 is over, it outputs public key ID^* and two messages M_0, M_1 on which it wishes to be challenged. Algorithm $\mathcal{A}_{dbdh}$ responds as follows:

1. $\mathcal{A}_{dbdh}$ runs the algorithm for responding to H-queries to obtain r_{ID^*} such that $H(\mathsf{ID}^*||\bar{b}_{\mathsf{ID}^*}) = g_2^{r_{\mathsf{ID}^*}}$.
2. $\mathcal{A}_{dbdh}$ sets $C^* = \langle c_0^*, c_1^* \rangle$ as follows:

$$\begin{aligned} c^*_{\bar{b}_{\mathsf{ID}^*}} &= \langle g_3^{r_{\mathsf{ID}^*}^{-1}}, \bar{M}_\beta \cdot T \rangle \\ c^*_{b_{\mathsf{ID}^*}} &= \langle g^r, \bar{M}_\beta \cdot e(h_{\mathsf{ID}^*||b_{\mathsf{ID}^*}}, g_1)^r \rangle \quad \text{if } \mathcal{COIN} = 0 \\ &= \langle g^r, \bar{M}_{\bar{\beta}} \cdot e(h_{\mathsf{ID}^*||b_{\mathsf{ID}^*}}, g_1)^r \rangle \quad \text{if } \mathcal{COIN} = 1, \end{aligned}$$

   where $\bar{M}_\beta = \phi(M_\beta||R_\beta)$, $\bar{M}_{\bar{\beta}} = \phi(M_{\bar{\beta}}||R_{\bar{\beta}})$, $r \in_R Z_q^*$ and $r_{\mathsf{ID}^*}^{-1}$ is the inverse of r_{ID^*} mod q.
3. $\mathcal{A}_{dbdh}$ gives $C^* = \langle c_0^*, c_1^* \rangle$ as the challenge ciphertext to $\mathcal{A}_{ibe}$.

Guess: When $\mathcal{A}_{ibe}$ decides that Phase 2 is over, $\mathcal{A}_{ibe}$ outputs its guess bit $\beta' \in \{0,1\}$. At the same time, algorithm $\mathcal{A}_{dbdh}$ outputs "$T = e(g,g)^{abc}$" if $\beta' = \beta$, or "$T \neq e(g,g)^{abc}$" otherwise. Without loss of generality, we can assume that $\mathcal{A}_{ibe}$ always outputs 0 or 1.

Next, let us define by E an event assigned to be true if and only if $\mathcal{A}_{dbdh}$ outputs "$T = e(g,g)^{abc}$".

Claim 4. *$\mathcal{A}_{ibe}$'s view is independent to the value of b_{ID} for any ID.*

Proof. Since leakage of information on b_{ID} may occur only from responses to Decryption queries, 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}. Next, we prove this by contradiction. Assume that $C = \langle \langle u_0, v_0 \rangle, \langle u_1, v_1 \rangle \rangle$ be a ciphertext such that $\mathbf{Decrypt}(\mathsf{params}, C, sk_0) \neq \mathbf{Decrypt}(\mathsf{params}, C, sk_1)$, where $sk_b = ((h_{\mathsf{ID}||b})^s, b)$, $b \in \{0,1\}$. Without loss of generality, we assume that $\mathbf{Decrypt}(\mathsf{params}, C, sk_0) = M (\neq \textit{reject})$. Then, we have $\langle u_0, v_0 \rangle = \langle g^{r_0}, \phi(M||R) \cdot e(g_{pub}, h_{\mathsf{ID}||0})^{r_0} \rangle$ for some $r_0 \in \mathbb{Z}_q^*$ and $R \in \{0,1\}^{k_2}$. Since $M \neq \textit{reject}$, following equations hold: $(r_0||r_1') = G(\mathsf{ID}, M, R)$, $u_0 = g^{r_0}$, $u_1 = g^{r_1'}$, $v_1 \cdot e(g_{pub}, h_{\mathsf{ID}||1})^{-r_1'} = \phi(M||R)$ for some $r_1' \in \mathbb{Z}_q^*$. This means that $\langle u_1, v_1 \rangle = \langle g^{r_1'}, \phi(M||R) \cdot e(g_{pub}, h_{\mathsf{ID}||1})^{r_1'} \rangle$, and it is obvious that $\mathbf{Decrypt}(\mathsf{params}, C, sk_1) = M$ which is a contradiction. □

Claim 4 guarantees symmetricity of c_0^* and c_1^* in C^*.

Claim 5. *We have that $\Pr[E|T = e(g,g)^{abc}, \mathcal{COIN} = 0] \geq 1/2 + Adv$, where Adv is $\mathcal{A}_{ibe}$'s advantage.*

Proof. When $T = e(g,g)^{abc}$, $\mathcal{COIN} = 0$ and $\mathcal{A}_{dbdh}$ doesn't abort, it is clear that $\mathcal{A}_{ibe}$'s view is identical to the real attack. Hence, we have

$$\frac{1}{2} + Adv = \Pr[\beta' = \beta | E_{vld} \wedge E_{abrt}] \cdot \Pr[E_{abrt}|E_{vld}] + \Pr[\beta' = \beta | E_{vld} \wedge \neg E_{abrt}] \cdot \Pr[\neg E_{abrt}|E_{vld}],$$

where E_{vld} denotes the event $(T = e(g,g)^{abc} \wedge \mathcal{COIN} = 0)$ and E_{abrt} denotes the event assigned to be true if and only if $\mathcal{A}_{dbdh}$ aborts during the simulation. Also, we have the following equation:

$$\begin{aligned} \Pr[E|E_{vld}] &= \Pr[E|E_{vld} \wedge E_{abrt}] \cdot \Pr[E_{abrt}|E_{vld}] + \Pr[E|E_{vld} \wedge \neg E_{abrt}] \cdot \Pr[\neg E_{abrt}|E_{vld}] \\ &= \Pr[E_{abrt}|E_{vld}] + \Pr[E|E_{vld} \wedge \neg E_{abrt}] \cdot \Pr[\neg E_{abrt}|E_{vld}]. \end{aligned}$$

From the above, we finally have

$$\begin{aligned}\Pr[E|E_{vld}] &\geq \Pr[E_{abrt}|E_{vld}] + (\frac{1}{2} + Adv - \Pr[\beta' = \beta|E_{vld} \wedge E_{abrt}] \cdot \Pr[E_{abrt}|E_{vld}]) \\ &= \frac{1}{2} + Adv + (1 - \Pr[\beta' = \beta|E_{vld} \wedge E_{abrt}]) \cdot \Pr[E_{abrt}|E_{vld}] \\ &\geq \frac{1}{2} + Adv,\end{aligned}$$

which proves the claim. □

Claim 6. *If $T = e(g,g)^{abc}$ and $\mathcal{COIN} = 1$, then $\mathcal{A}_{dbdh}$'s view is independent to β and hence,* $\Pr[E|T = e(g,g)^{abc},\ \mathcal{COIN} = 1] = 1/2$.

Proof. It is obvious from Claim 4. □

Claim 7. *We have that* $|\Pr[E|T \neq e(g,g)^{abc},\ \mathcal{COIN} = 0] - \Pr[\neg E|T \neq e(g,g)^{abc},\ \mathcal{COIN} = 1]| \leq q_G/2^{k_2}$.

Proof. If $T \neq e(g,g)^{abc}$, then $c^*_{\bar{b}_{\mathsf{ID}^*}}$ is uniformly distributed in $\mathbb{G}_1 \times \mathbb{G}_2$ due to the random $\log_g g_3$. Since

$$\begin{aligned}c^*_{\bar{b}_{\mathsf{ID}^*}} &= \langle g^r, \bar{M}_\beta \cdot e(h_{\mathsf{ID}^*||\bar{b}_{\mathsf{ID}^*}}, g_1)^r\rangle \quad \text{if } \mathcal{COIN} = 0 \\ &= \langle g^r, \bar{M}_{\bar{\beta}} \cdot e(h_{\mathsf{ID}^*||\bar{b}_{\mathsf{ID}^*}}, g_1)^r\rangle \quad \text{if } \mathcal{COIN} = 1,\end{aligned}$$

distribution of $c^*_{\bar{b}_{\mathsf{ID}^*}}$ for the case of $(\beta = \mathtt{b}) \wedge (\mathcal{COIN} = 0)$ is statistically indistinguishable from that for the case of $(\bar{\beta} = \bar{\mathtt{b}}) \wedge (\mathcal{COIN} = 1)$ for any $\mathtt{b} \in \{0,1\}$. We notice that $\mathcal{A}_{ibe}$ distinguishes the above distributions only when it submits a G-query $(*, *, R_{\mathtt{b}})$, where $*$ denotes any bit string. Hence, $|\Pr[E|T \neq e(g,g)^{abc},\ \mathcal{COIN} = 0] - \Pr[\neg E|T \neq e(g,g)^{abc},\ \mathcal{COIN} = 1]| \leq q_G/2^{k_2}$ holds. □

Now, we are back to the proof of the theorem. We calculate the advantage of $\mathcal{A}_{dbdh}$ by using the above claims.

$$\begin{aligned}\epsilon_{dbdh} &\geq |\Pr[E|T = e(g,g)^{abc}] - \Pr[E|T \neq e(g,g)^{abc}]| \\ &\geq \frac{1}{2}\Pr[E|T = e(g,g)^{abc}, \mathcal{COIN} = 0] + \frac{1}{2}\Pr[E|T = e(g,g)^{abc}, \mathcal{COIN} = 1] \\ &\quad -\frac{1}{2}\Pr[E|T \neq e(g,g)^{abc}, \mathcal{COIN} = 0] - \frac{1}{2}\Pr[E|T \neq e(g,g)^{abc}, \mathcal{COIN} = 1] \\ &\geq \frac{1}{2}\Pr[E|T = e(g,g)^{abc}, \mathcal{COIN} = 0] + \frac{1}{2}\Pr[E|T = e(g,g)^{abc}, \mathcal{COIN} = 1] \\ &\quad -(\frac{1}{2}\Pr[E|T \neq e(g,g)^{abc}, \mathcal{COIN} = 0] + (\frac{1}{2} - \frac{1}{2}\Pr[\neg E|T \neq e(g,g)^{abc}, \mathcal{COIN} = 1])) \\ &\geq \frac{1}{2}(\frac{1}{2} + Adv) + \frac{1}{2} \cdot \frac{1}{2} - (\frac{1}{2} + \frac{1}{2} \cdot \frac{q_G}{2^{k_2}}) \\ &= \frac{1}{2}Adv - \frac{q_G}{2^{k_2+1}}.\end{aligned}$$

Hence, we have that

$$2\epsilon_{dbdh} + \frac{q_G}{2^{k_2}} \geq Adv,$$

and consequently,

$$2\epsilon_{dbdh} + \frac{q_G}{2^{k_2}} \geq \epsilon_{ibe}.$$

From above discussions, it is easily seen that the claimed bound of the running-time of $\mathcal{A}_{dbdh}$ holds. This completes the proof of Theorem 2. □

B The proof of Theorem 3

Proof. The theorem follows from Claim 8 below. □

We prove this theorem by using "game-based technique"[1, 17] unlike the proofs of Theorems 1 and 2. We are able to reduce distinguishing of two successive games into solving a certain problem. Majority of procedures in these reductions are the same as those given in the proofs of Theorem 1 and 2. Hence, in the followings, we concentrate mostly on their differences.

Definition 1. *Game 0-4 are defined as in the following:*
Game 0:

Same as the real game. Let X_0 is an event that the adversary output β' which is equal to β. Events $X_1, \ldots, X_4$ in Game 1,2,3, and 4 are defined in the same way.
Game 1:

Suppose that (A, B, C) which is randomly chosen from uniform distribution of $(\mathbb{G}_1)^3$, and $D \in \mathbb{G}_2$, which is the solution of computational bilinear Diffie-Hellman problem with respect to (A, B, C), is given.

Game 1 is the same as Game 0 except in the following points:

Setup: *We let $g_{pub} = A$ instead of letting $g_{pub} = g^s$.*

H-queries: *H is an ordinary random oracle except that its output is chosen in the following way: For H query with respect to $\mathsf{ID}||b$, randomly choose $b_{\mathsf{ID}} \in_R \{0,1\}$ and $(r_{\mathsf{ID}}, t_{\mathsf{ID}}) \in_R (\mathbb{Z}_q)^2$ and then let $(h_{\mathsf{ID}||b_{\mathsf{ID}}}, d_{\mathsf{ID}}) = (g^{r_{\mathsf{ID}}}, A^{r_{\mathsf{ID}}})$ and $h_{\mathsf{ID}||\bar{b}_{\mathsf{ID}}} = B^{t_{\mathsf{ID}}}$.*

Challenge: *Challenge ciphertext is generated as follows: First, choose random $w \in_R \mathbb{Z}_q$ and generate*

$$\begin{aligned} c^*_{b^*_{\mathsf{ID}^*}} &= (g^w, M_b \cdot e(g_{pub}, h_{\mathsf{ID}^*||b^*_{\mathsf{ID}^*}})^w) \\ c^*_{\bar{b}^*_{\mathsf{ID}^*}} &= (C^{t_{\mathsf{ID}^*}}, M_b \cdot D) \end{aligned}$$

Other variables are simulated by choosing random oracles.

Game 2:

Same as Game 1 except that D is randomly chosen.
Game 3:

Same as Game 2 except that the distribution from which D is chosen is the same as that in Game 1 and that challenge ciphertext is generated as follows: Randomly choose $M \in \mathbb{G}_2$ and generate

$$\begin{aligned} c^*_{b^*_{\mathsf{ID}^*}} &= (g^w, M \cdot e(g_{pub}, h_{\mathsf{ID}^*||b^*_{\mathsf{ID}^*}})^w) \\ c^*_{\bar{b}^*_{\mathsf{ID}^*}} &= (C^{t_{\mathsf{ID}^*}}, M_b \cdot D). \end{aligned}$$

Other variables are simulated by choosing random oracles.
Game 4:

Same as Game 3 except that D is randomly chosen.

Claim 8. *The followings hold:*

1. $\epsilon_{ibe} - \Pr[X_0] = 0$
2. $|\Pr[X_0] - \Pr[X_1]| \leq q_G/q$

3. *There exists t_{dbdh}-time algorithm $\mathcal{A}_{dbdh}$ such that*
   $\epsilon_{dbdh} \geq |\Pr[X_1] - \Pr[X_2]|$ *and* $t_{dbdh} \leq t_{ibe} + \Theta(\tau(2q_H + 3q_E + 13q_D)$
4. $|\Pr[X_2] - \Pr[X_3]| \leq \frac{q_G + 2q_H}{q}$
5. *There exists t_{dbdh}-time algorithm $\mathcal{A}_{dbdh}$ such that*
   $\epsilon_{dbdh} \geq |\Pr[X_3] - \Pr[X_4]|$ *and* $t_{dbdh} \leq t_{ibe} + \Theta(\tau(2q_H + 3q_E + 13q_D)$
6. $\Pr[X_4] - 1/2 = 0$

Proof. 1. The 1st and the 6th relations clearly hold.

2. With respect to the 2nd relation, $\Pr[X_0] - \Pr[X_1] = 0$ holds unless the simulation fails. Therefore, from Claim 9, the result follows.

3. It is easy to see that the 3rd and the 5th relations hold. The algorithm $\mathcal{A}_{dbdh}$ needs $\Theta(\tau(2q_H + 3q_E + 13q_D)$ times more than $\mathcal{A}_{ibe}$ to simulate H oracle, the Decryption oracle, and the Extraction oracle. The simulations can be done similarly to those in the proofs of Theorems 1 and 2.

4. With respect to 4th relation, $\Pr[X_0] - \Pr[X_1] = 0$ holds unless the behavior of Decryption oracle depends on the value of b^*_{ID}. Therefore, from Claim 10, the result follows.

□

Claim 9. *Given (u_0, v_0, u_1, v_1), (c_M, c, t_0, t_1, t_M) are perfectly simulatable by choosing random oracle with the probability of at least $1 - q_G/q$.*

Proof. c_M, c, t_0, t_1, t_M are uniformly and randomly distributed in $\mathbb{G}_2 \times (\mathbb{Z}_q)^4$ because of random choice of r_M, s_0, c, s_1, s_M (by Encrypt and a random oracle). Hence, by randomly choosing c_M, c, t_0, t_1, t_M and a consistent random oracle, which succeeds with the probability of at least $1 - q_G/q$, perfect simulation is possible. □

Claim 10. *A ciphertext C will not be accepted with the probability more than $\frac{q_G + 2q_H}{q}$ unless its both decryptions by secret keys $((h_{\mathsf{ID}||0})^s, 0)$ and $((h_{\mathsf{ID}||1})^s, 1)$ coincide.*

Proof. We will show that no adversary whose resources are unbounded, with the exception that it may ask random oracles only polynomial number of times, is able to generates a ciphertext that can be accepted with non-negligible probability unless the results of two decryptions are the same.

Suppose that $r_0, r_1, r'_0, r'_1, r_M, s_0, s_1, s'_0, s'_1, s_M, s'_M, s''_M, \alpha_0$, and α_1 are such that

$$\begin{aligned}
u_0 &= g^{r_0} \\
v_0 &= M \cdot e(g_{pub}, h_{\mathsf{ID}||0})^{r'_0} = Me(g,g)^{\alpha_0 r'_0} \\
u_1 &= g^{r_1} \\
v_1 &= M \cdot e(g_{pub}, h_{\mathsf{ID}||1})^{r'_1} = Me(g,g)^{\alpha_1 r'_1} \\
c_M &= (1/M) \cdot e(g,g)^{r_M} \\
u'_0 &= g^{s_0} \\
u'_1 &= g^{s_1} \\
v'_0 &= e(g_{pub}, h_{\mathsf{ID}||0})^{s'_0} e(g,g)^{s_M} \\
v'_1 &= e(g_{pub}, h_{\mathsf{ID}||1})^{s'_1} e(g,g)^{s'_M} \\
c'_M &= M \cdot g^{s''_M}
\end{aligned}$$

hold. Then, for randomly chosen c, equations

$$\begin{aligned} t_0 &= r_0 c + s_0 \\ t_1 &= r_1 c + s_1 \\ t_M + \alpha_0 t_0 &= (\alpha_0 r'_0 + r_M)c + s_M + \alpha_0 s'_0 \\ t_M + \alpha_1 t_1 &= (\alpha_1 r'_1 + r_M)c + s'_M + \alpha_1 s'_1 \end{aligned}$$

must hold for the ciphertext to be accepted. That is,

$$\{\alpha_0(r'_0 - r_0) - \alpha_1(r'_1 - r_1)\}c + \{\alpha_0(s'_0 - s_0) - \alpha_1(s'_1 - s_1)\} + (s_M - s'_M) = 0$$

must holds for randomly chosen c. Hence, the ciphertext will be accepted at most with the probability of $\frac{q_G+2q_H}{q}$ unless equation

$$\alpha_0(r'_0 - r_0) = \alpha_1(r'_1 - r_1)$$

holds. In this case when the equation holds, the following equation

$$\begin{aligned} v_0 \cdot e(u_0, (h_{\mathsf{ID}||0})^s)^{-1} &= Me(g,g)^{\alpha_0 r'_0} e(g,g)^{-\alpha_0 r_0} \\ &= Me(g,g)^{\alpha_0 r'_0 - \alpha_0 r_0} \\ &= Me(g,g)^{\alpha_1 r'_1 - \alpha_1 r_1} \\ &= Me(g,g)^{\alpha_1 r'_1} e(g,g)^{-\alpha_1 r_1} \\ &= v_1 \cdot e(u_1, (h_{\mathsf{ID}||1})^s)^{-1} \end{aligned}$$

holds. Thus, the claim is proved. □