Simulation without the Artificial Abort: Simplified Proof and Improved Concrete Security for Waters' IBE Scheme

Mihir Bellare and Thomas Ristenpart

Dept. of Computer Science & Engineering 0404, University of California San Diego
9500 Gilman Drive, La Jolla, CA 92093-0404, USA
{mihir,tristenp}@cs.ucsd.edu
http://www-cse.ucsd.edu/users/{mihir,tristenp}

Abstract. Waters' variant of the Boneh-Boyen IBE scheme is attractive because of its efficency, applications, and security attributes, but suffers from a relatively complex proof with poor concrete security. This is due in part to the proof's "artificial abort" step, which has then been inherited by numerous derivative works. It has often been asked whether this step is necessary. We show that it is not, providing a new proof that eliminates this step. The new proof is not only simpler than the original one but offers better concrete security for important ranges of the parameters. As a result, one can securely use smaller groups, resulting in significant efficiency improvements.

1 Introduction

The importance of identity-based encryption (IBE) as a cryptographic primitive stems from its widespread deployment and the numerous applications enabled by it. Since the initial work on providing realizations of IBE [8, 15], improving the efficiency, security, and extensibility of the fundamental primitive has consequently received substantial attention from the research community. A challenging problem has been to arrive at a practical IBE scheme with a tight security reduction under standard assumptions. (The most attractive target being DBDH without relying on random oracles.) While a typical approach for progressing towards this goal is proposing new constructions, in this paper we take another route: improving the concrete security of existing constructions. This requires providing better proofs of security and analyzing the impact of their tighter reductions.

Why concrete security? Informally speaking, consider an IBE scheme with a security reduction showing that attacking the scheme in time t with success probability ϵ implies breaking some believed-to-be hard problem in time $t + \omega_1$ with success probability $\epsilon' \geq \epsilon/\omega_2$. Tightness of the reduction refers to the value of ω_1 (the overhead in time needed to solve the hard problem using the scheme attacker) and of ω_2 (the amount by which the success probability decreases). Unlike asymptotic treatments, provably-secure IBE has a history of utilizing

concrete security, meaning specifying ω_1 and ω_2 explicitly. Concrete-security for IBE started with Boneh and Franklin [8] and has been continued in subsequent works, e.g. [6, 7, 9, 31, 19, 23] to name just a few.

As Gentry points out [19], concrete security and tight reductions are not just theoretical issues for IBE, rather they are of utmost practical import: the speed of implementations increases as ω_1 and/or ω_2 decrease. This is because security guarantees are lost unless the size of groups used to implement a scheme grow to account for the magnitude of these values. In turn group size dictates performance: exponentiations in a group whose elements can be represented in r bits takes roughly $\mathcal{O}(r^3)$ time. As a concrete example, this means that performing four 160-bit group exponentiations can be significantly faster than a *single* 256-bit group exponentiation. In practice even a factor of two efficiency slow-down is considered significant (let alone a factor of four), so finding as-tight-as-possible reductions is crucial.

Overview of IBE approaches. All practical IBE systems currently known are based on bilinear pairings. We can partition the space of such systems along two dimensions, as shown in the left table of Figure 1. In one dimension is whether one utilizes random oracles or not. In the other is the flavor of hard problem used, whether it be interactive (for example the q-BDHI assumption [6]) or non-interactive (for example bilinear Diffie-Hellman, or BDH, style assumptions [8]). Of note is that Katz and Wang [23], in the "random oracle/BDH" setting, and Gentry [19], in the "no random oracle/interactive setting", have essentially solved the problem of finding practical schemes with tight reductions. On the other hand, finding practical schemes with tight reductions in the "no random oracle/BDH" setting represents a hard open problem mentioned in numerous works [6, 7, 31, 19]. This last setting turns out to be attractive for two reasons. First, from a security perspective, it is the most conservative (and consequently most challenging) with regard to choice of assumptions. Second, schemes thus far proposed in this setting follow a framework due to Boneh and Boyen [6] (so-called "commutative blinding") that naturally supports many valuable extensions: hierarchical IBE [22], attribute-based IBE [29], direct CCA-secure encryption [10, 24], etc.

Progress in this setting is summarized in the right table of Figure 1. Boneh and Boyen initiated work here with the $\mathsf{BB_1}$ scheme (the first scheme in [6]). They prove it secure under the decisional BDH (DBDH) assumption, but in the selective-ID attack model of [11] in which adversaries must commit to a targeted identity before seeing the IBE system's parameters. Boneh and Boyen show how to prove full security, but the reduction doing so is exponentially loose (briefly, because it requires guessing the hash of the to-be-attacked identity).

Waters' proposed a variant of $\mathsf{BB_1}$ that we'll call Wa [31]. This variant requires larger public parameters, but can be proven fully secure with a polynomial reduction to DBDH that does not use random oracles. The relatively complex security proof relies on a novel "artificial abort" step, that, while clever, is unintuitive. It also significantly hurts the concrete security and, thereby, efficiency of the scheme. Many researchers in the community have asked whether artificial

	Interactive	BDH
RO model	SK	BF , KW
Standard model	BB_2 , Ge	BB_1 , Wa

Scheme	Security	Reduction
BB_1	selective-ID	polynomial
BB_1	full	exponential
Wa	full	polynomial

Fig. 1. A comparison of practical IBE schemes. BF is the Boneh-Franklin scheme [8]; SK is the Sakai-Kasahara scheme [30]; KW is the Katz-Wang scheme [23]; BB_1 and BB_2 are the first and second Boneh-Boyen schemes from [6]; Wa is Waters' scheme [31]; and Ge is Gentry's scheme [19]. **(Left)** The assumptions (an interactive assumption versus bilinear Diffie-Hellman) and model (random oracles or not) used to prove security of the schemes. **(Right)** Types of security offered by standard model BDH-based systems and asymptotic reduction tightness.

aborts can be dispensed with, but the general consensus seems to have been that the answer is "no" and that the technique is somehow fundamental to proving security. This folklore assessment (if true) is doubly unfortunate because Wa, inheriting the flexibility of the Boneh-Boyen framework, has been used in numerous diverse applications [10, 1, 5, 27, 12, 13, 20, 24]. As observed in [24], some of these subsequent works offer difficult to understand (let alone verify) proofs, due in large part to their use of the artificial abort technique in a more-or-less black-box manner. They also inherit its concrete security overhead.

This paper. Our first contribution is to provide a novel proof of Waters' variant that completely eliminates the artificial abort step. The proof, which uses several new techniques and makes crucial use of code-based games [4], provides an alternate and (we feel) more intuitive and rigorous approach to proving the security of Wa. Considering the importance of the original proof (due to its direct or indirect use in [10, 1, 5, 27, 12, 13, 20, 24]), a more readily understood proof is already a significant contribution. Our reduction (like Waters') is not tight, but as we see below it offers better concrete security for many important parameter choices, moving us closer to the goal of standard model BDH-based schemes with tight reductions. The many Waters'-derived works [10, 1, 5, 27, 12, 13, 20, 24] inherit the improvements in concrete security. We briefly describe these derivatives in Appendix A.

We now have the BB_1 and Wa schemes, the former with an exponentially-loose reduction and the latter with now two polynomial reductions each having a complex concrete security formula. What is the most efficient approach for providing provably-secure DBDH-based IBE? Since we want to account for the impact of reduction tightness, answering this question requires work. We offer a framework for computing the concrete efficiency of reductions, adopting techniques from [26, 25, 18]. Efficiency is measured by mapping desired (provable) security levels to requisite group sizes. Not only does this approach provide a metric for comparing different reductions, it also allows comparing the resultant bit-operation speed of schemes when each is instantiated in groups of size sufficient to account for the reduction. Providing such a framework that simul-

κ	ϵ	q	s_{BB}	s_W	s_{BR}	$\mathbf{T}_{\mathsf{Enc}}(s_W)/\mathbf{T}_{\mathsf{Enc}}(s_{BR})$
60	2^{-20}	2^{20}	192	192	128	**9**
70	2^{-20}	2^{20}	256	192	128	**9**
80	2^{-30}	2^{30}	256	256	192	**5**
90	2^{-30}	2^{30}	–	256	192	**5**
100	2^{-10}	2^{10}	–	128	192	1/9
100	2^{-40}	2^{40}	–	256	192	**5**
192	2^{-40}	2^{40}	–	256	–	–

Fig. 2. Table showing the security level of the pairing setups required to achieve κ-bits of security for the BB_1 and Wa encryption schemes when adversaries achieve ϵ success probability using q key extraction queries. Loosely speaking, the security level of the pairing setup is $(\log p)/2$ where p is the size of the first pairing group. Here s_{BB}, s_W, s_{BR} are, respectively, the securities of the pairing setups for BB_1, Wa under Waters' reduction, and Wa under the new reduction. The final column represents the (approximate) ratio of encryption times for Wa as specified by the two reductions. A dash signifies that one needs a pairing setup of security greater than 256.

taneously provides simplicity, accuracy, and fairness (i.e. not biased towards particular schemes/reductions) turned out to be very challenging.

Let us first mention the high-level results, before explaining more. In the end our framework implies that Waters' variant usually provides faster standard model encryption (than BB_1). Our new proof provides a better reduction for low to mid range security parameters, while Waters' reduction is tighter for higher security parameters. The new reduction in fact drastically improves efficiency in the former category, offering up to 9 times faster encryption for low parameters and 5 times faster encryption for mid-range security levels. Where Waters' reduction is tighter, we can continue to choose group size via it; the new reduction never *hurts* efficiency.

BB_1 does better than Wa when identities are short, such as $n = 80$ bits. We have, however, focused on providing IBE with arbitrary identity spaces, which provides the most versatility. Supporting long identities (e.g. email addresses such as `john.doe123@anonymous.com`) requires utilizing a collision-resistant hash function to compress identities. In this case, the birthday bound mandates that the bit length n of hash outputs be double the desired security level, and this affects the BB_1 scheme more due to its reduction being loose by a factor of 2^n.

Framework details. We present some results of applying our framework in Figure 2. Let us explain briefly what the numbers signify and how we derived them. (Details are in the full version of this paper [3].) By a *setup* we mean groups $\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T$ admitting a bilinear map $\mathbf{e}$: $\mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_T$. The setup provides security s (bits) if the best known algorithms to solve the discrete logarithm (DL) problem take at least 2^s time in any of the three groups. We assume (for these estimates but not for the proof!) that the best algorithm for

solving DBDH is solving DL in one of the groups. An important practical issue in pairings-based cryptography is that setups for arbitrary security are not known. Accordingly, we will restrict attention to values $s = 80$, 112, 128, 192, and 256, based on information from [28, 25, 26, 16, 17]. Now we take as our target that the IBE scheme should provide κ bits of security. By this we mean that any adversary making at most $q = 1/\epsilon$ **Extract** queries and having running time at most $\epsilon 2^{\kappa}$ should have advantage at most ϵ. For each scheme/reduction pair we can then derive the security s of the underlying pairing setup required to support the desired level of security. See Figure 2 for $\mathsf{BB_1}$ (s_{BB}), Wa under Waters' reduction (s_W), and under the new reduction (s_{BR}).

Other related work and open problems. Recently Hofheinz and Kiltz describe programmable hash functions [21]. Their main construction uses the same hash function (originally due to Chaum et al. [14]) as Waters', and they provide new proof techniques that provide a $\sqrt{n}$ (n is the length of identities) improvement on certain bounds that could be applicable to Wa. But this will only offer a small concrete security improvement compared to ours. Moreover, their results are asymptotic and hide (seemingly very large) unknown constants.

As mentioned, providing a scheme based on DBDH that has a tight security reduction (without random oracles) is a hard open problem, and one that remains after our work. (One reason we explain this is that we have heard it said that eliminating the artificial abort would solve the open problem just mentioned, but in fact the two seem to be unrelated.) Finding a tight reduction for Waters' (or another $\mathsf{BB_1}$-style scheme) is of particular interest since it would immediately give a hierarchical IBE (HIBE) scheme with security beyond a constant number of levels (the best currently achievable). From a practical point of view we contribute here, since better concrete security improves the (constant) number of levels achievable. From a theoretical perspective, this remains an open problem.

Viewing proofs as qualitative. We measure efficiency of schemes when one sets group size according to the best-known reduction. However, the fact that a proof implies the need for groups of certain size to guarantee security of the scheme does not mean the scheme is necessarily insecure (meaning there is an attack) over smaller groups. It simply means that the proof tells us nothing about security in these smaller groups. In the context of standards it is sometimes suggested one view a proof as a qualitative rather than quantitative guarantee, picking group sizes just to resist the best known attack. Our sense is that this procedure is not viewed as ideal even by its proposers but rather forced on them by the looseness of reductions. To rectify this gap, one must find tighter reductions, and our work is a step to this end.

2 Definitions and Background

Notation. We fix *pairing parameters* $\mathsf{GP} = (\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, p, \mathbf{e}, \psi)$ where $\mathbb{G}_1$, $\mathbb{G}_2$, $\mathbb{G}_T$ are groups of prime order p; $\mathbf{e}$: $\mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ is a non-degenerate, effi-

ciently computable bilinear map; and ψ: $\mathbb{G}_2 \to \mathbb{G}_1$ is an efficiently computable isomorphism [8]. Let $\mathsf{T}_{\exp}(\mathbb{G})$ denote the time to compute an exponentiation in a group $\mathbb{G}$. Similarly, let $\mathsf{T}_{\mathrm{op}}(\mathbb{G})$ denote the time to compute a group operation in a group $\mathbb{G}$. Let T_ψ denote the time to compute ψ. Let $\mathbb{G}^* = \mathbb{G} - \{\mathbf{1}\}$ denote the set of generators of $\mathbb{G}$ where $\mathbf{1}$ is the identity element of $\mathbb{G}$.

Vectors are written in boldface, e.g. $\mathbf{u} \in \mathbb{Z}_p^{n+1}$ is a vector of $n+1$ values each in $\mathbb{Z}_p$. We denote the i^{th} component of a vector $\mathbf{u}$ by $\mathbf{u}[i]$. If $S \in \{0,1\}^*$ then $|S|$ denotes its length and $S[i]$ denotes its i^{th} bit. For integers i, j we let $[i\,..\,j] = \{i, \ldots, j\}$. The running time of an adversary $\mathcal{A}$ is denoted $\mathbf{T}(\mathcal{A})$. We use big-oh notation with the understanding that this hides a small, fixed, machine-dependent constant.

GAMES. Our security definitions and proofs use code-based games [4], and so we recall some background from [4]. A game (look at Figure 3 for examples) has an **Initialize** procedure, procedures to respond to adversary oracle queries, and a **Finalize** procedure. A game G is executed with an adversary $\mathcal{A}$ as follows. First, **Initialize** executes, and its outputs are the inputs to $\mathcal{A}$. Then $\mathcal{A}$ executes, its oracle queries being answered by the corresponding procedures of G. When $\mathcal{A}$ terminates, its output becomes the input to the **Finalize** procedure. The output of the latter is called the output of the game, and we let $\mathrm{G}^{\mathcal{A}} \Rightarrow y$ denote the event that this game output takes value y. The boolean flag bad is assumed initialized to false. Games G_i, G_j are identical-until-bad if their code differs only in statements that follow the setting of bad to true. We let "$\mathrm{G}_i^{\mathcal{A}}$ sets bad" denote the event that game G_i, when executed with adversary $\mathcal{A}$, sets bad to true (and similarly for "$\mathrm{G}_i^{\mathcal{A}}$ doesn't set bad"). It is shown in [4] that if $\mathrm{G}_i, \mathrm{G}_j$ are identical-until-bad and $\mathcal{A}$ is an adversary, then

$$\Pr\left[\,\mathrm{G}_i^{\mathcal{A}} \text{ sets } \mathsf{bad}\,\right] = \Pr\left[\,\mathrm{G}_j^{\mathcal{A}} \text{ sets } \mathsf{bad}\,\right]. \tag{1}$$

The fundamental lemma of game-playing [4] says that if $\mathrm{G}_i, \mathrm{G}_j$ are identical-until-bad then for any y

$$\Pr\left[\,\mathrm{G}_i^{\mathcal{A}} \Rightarrow y\,\right] - \Pr\left[\,\mathrm{G}_j^{\mathcal{A}} \Rightarrow y\,\right] \le \Pr\left[\,\mathrm{G}_i^{\mathcal{A}} \text{ sets } \mathsf{bad}\,\right].$$

This lemma is useful when the probability that bad is set is small, but in our setting this probability will be close to one. We will instead use the following variant:

Lemma 1. Let $\mathrm{G}_i, \mathrm{G}_j$ be identical-until-bad games and let $\mathcal{A}$ be an adversary. Then for any y

$$\Pr\left[\,\mathrm{G}_i^{\mathcal{A}} \Rightarrow y \wedge \mathrm{G}_i^{\mathcal{A}} \text{ doesn't set } \mathsf{bad}\,\right] = \Pr\left[\,\mathrm{G}_j^{\mathcal{A}} \Rightarrow y \wedge \mathrm{G}_j^{\mathcal{A}} \text{ doesn't set } \mathsf{bad}\,\right]. \ \square$$

Lemma 1 is implicit in the proof of the fundamental lemma of [4].

DBDH PROBLEM. The Decisional Bilinear Diffie-Hellman (DBDH) assumption (in the asymmetric setting) [6] is captured by the game described in Figure 3. We define the dbdh-advantage of an adversary $\mathcal{A}$ against $\mathsf{GP} = (\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, p, \mathbf{e}, \psi)$ by

$$\mathbf{Adv}^{\mathrm{dbdh}}_{\mathsf{GP}}(\mathcal{A}) = 2 \cdot \Pr\left[\,\mathrm{DBDH}^{\mathcal{A}}_{\mathsf{GP}} \Rightarrow \mathsf{true}\,\right] - 1\,. \tag{2}$$

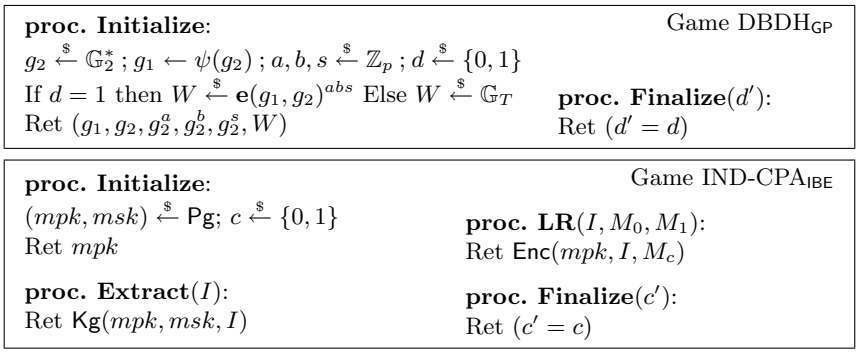

Fig. 3. The DBDH and IND-CPA games.

IDENTITY-BASED ENCRYPTION. An *identity-based encryption (IBE) scheme* is a tuple of algorithms $\mathsf{IBE} = (\mathsf{Pg}, \mathsf{Kg}, \mathsf{Enc}, \mathsf{Dec})$ with associated identity space $\mathsf{IdSp} \subseteq \{0,1\}^*$ and message space MsgSp. The key-issuing center runs the parameter generation algorithm Pg (which takes no input) generates a master public key mpk and a master secret key msk. The former is publicly distributed. The key generation algorithm Kg takes as input mpk, msk, I, where $I \in \mathsf{IdSp}$, and outputs a secret key sk for party I. The encryption algorithm Enc takes inputs mpk, I, M, where $I \in \mathsf{IdSp}$ and $M \in \mathsf{MsgSp}$, and outputs a ciphertext C. The deterministic decryption algorithm Dec takes inputs mpk, sk, I, C and outputs either $\perp$ or a plaintext M. We require the usual consistency, namely that $\mathsf{Dec}(mpk, sk, I, \mathsf{Enc}(mpk, I, M)) = M$ with probability one for all $I \in \mathsf{IdSp}$ and $M \in \mathsf{MsgSp}$, where the probability is over $(mpk, msk) \xleftarrow{\$} \mathsf{Pg}$; $sk \xleftarrow{\$} \mathsf{Kg}(mpk, msk, I)$ and the coins used by Enc. We use the notion of privacy from [8], namely indistinguishability under chosen-plaintext attack (ind-cpa). The ind-cpa advantage of an adversary $\mathcal{A}$ against an IBE scheme IBE is defined by

$$\mathbf{Adv}^{\text{ind-cpa}}_{\mathsf{IBE}}(\mathcal{A}) = 2 \cdot \Pr\left[\,\text{IND-CPA}^{\mathcal{A}}_{\mathsf{IBE}} \Rightarrow \mathsf{true}\,\right] - 1\,, \tag{3}$$

where game IND-CPA is shown in Figure 3. We only allow *legitimate* adversaries, where adversary $\mathcal{A}$ is legitimate if it makes only one query (I^*, M_0, M_1) to **LR**, for some $I^* \in \mathsf{IdSp}$ and $M_0, M_1 \in \mathsf{MsgSp}$ with $|M_0| = |M_1|$, and never queries I^* to **Extract**. Here $|M|$ denotes the length of some canonical string encoding of a message $M \in \mathsf{MsgSp}$. (In the schemes we consider messages are group elements.)

WATERS' IBE SCHEME. Let n be a positive integer. Define the hash family H: $\mathbb{G}_1^{n+1} \times \{0,1\}^n \to \mathbb{G}_1$ by $H(\mathbf{u}, I) = \mathbf{u}[0] \prod_{i=1}^{n} \mathbf{u}[i]^{I[i]}$ for any $\mathbf{u} \in \mathbb{G}_1^{n+1}$ and any $I \in \{0,1\}^n$. The Waters IBE scheme $\mathsf{Wa} = (\mathsf{Pg}, \mathsf{Kg}, \mathsf{Enc}, \mathsf{Dec})$ associated to GP and n has associated identity space $\mathsf{IdSp} = \{0,1\}^n$ and message space $\mathsf{MsgSp} = \mathbb{G}_T$, and its first three algorithms are as follows:

proc. Pg	**proc.** $\mathsf{Kg}(mpk, msk, I)$	**proc.** $\mathsf{Enc}(mpk, I, M)$
$A_1 \xleftarrow{\$} \mathbb{G}_1 \,;\, g_2 \xleftarrow{\$} \mathbb{G}_2^*$	$(g_2, A_1, B_2, \mathbf{u}) \leftarrow mpk$	$(g_2, A_1, B_2, \mathbf{u}) \leftarrow mpk$
$b \xleftarrow{\$} \mathbb{Z}_p \,;\, B_2 \leftarrow g_2^b \,;\, \mathbf{u} \xleftarrow{\$} \mathbb{G}_1^{n+1}$	$K \leftarrow msk;\ r \xleftarrow{\$} \mathbb{Z}_p$	$s \xleftarrow{\$} \mathbb{Z}_p$
$mpk \leftarrow (g_2, A_1, B_2, \mathbf{u})$	Ret $(K \cdot H(\mathbf{u}, I)^r, g_2^r)$	$c_1 \leftarrow \mathbf{e}(A_1, B_2)^s \cdot M$
$msk \leftarrow A_1^b$		$(c_2, c_2) \leftarrow (g_2^s, H(\mathbf{u}, I)^s)$
Ret (mpk, msk)		Ret (c_1, c_2, c_3)

Above, when we write $(g_2, A_1, B_2, \mathbf{u}) \leftarrow mpk$ we mean mpk is parsed into its constituent parts. We do not specify the decryption algorithm since it is not relevant to IND-CPA security; it can be found in [31].

In [31] the scheme is presented in the symmetric setting where $\mathbb{G}_1 = \mathbb{G}_2$. While this makes notation simpler, we work in the asymmetric setting because it allows pairing parameters for higher security levels [17].

The hash function used by Waters' scheme has restricted domain. One can extend to $\mathsf{IdSp} = \{0,1\}^*$ by first hashing an identity with a collision-resistant hash function to derive an n-bit string. To ensure security from birthday attacks, the output length n of the CR function must have bit-length at least twice that of the desired security parameter.

WATERS' RESULT. Waters [31] proves the security of the Wa scheme associated to GP, n under the assumption that the DBDH problem in GP is hard. Specifically, let $\mathcal{A}$ be an ind-cpa adversary against Wa that runs in time at most t, makes at most $q \in [1\,..\,p/4n]$ queries to its **Extract** oracle and has advantage $\epsilon = \mathbf{Adv}^{\text{ind-cpa}}_{\mathsf{Wa}}(\mathcal{A})$. Then [31, Theorem 1] presents a dbdh-adversary $\mathcal{B}_{\mathsf{Wa}}$ such that

$$\mathbf{Adv}^{\text{dbdh}}_{\mathsf{GP}}(\mathcal{B}_{\mathsf{Wa}}) \geq \frac{\epsilon}{32(n+1)q} \ , \text{ and} \tag{4}$$

$$\mathbf{T}(\mathcal{B}_{\mathsf{Wa}}) = \mathbf{T}(\mathcal{A}) + \mathbf{T}_{\text{sim}}(n, q) + \mathbf{T}_{\text{abort}}(\epsilon, n, q) \tag{5}$$

where

$$\mathbf{T}_{\text{sim}}(n, q) = \mathcal{O}\left(\mathsf{T}_\psi + (n+q)\cdot \mathsf{T}_{\text{exp}}(\mathbb{G}_1) + q \cdot \mathsf{T}_{\text{exp}}(\mathbb{G}_2) + qn + \mathsf{T}_{\text{op}}(\mathbb{G}_T)\right) \tag{6}$$

$$\mathbf{T}_{\text{abort}}(\epsilon, n, q) = \mathcal{O}(q^2 n^2 \epsilon^{-2} \ln(\epsilon^{-1}) \ln(qn)) \,. \tag{7}$$

An important factor in the "looseness" of the reduction is the $\mathbf{T}_{\text{abort}}(\epsilon, n, q)$ term, which can be very large, making $\mathbf{T}(\mathcal{B}_{\mathsf{Wa}})$ much more than $\mathbf{T}(\mathcal{A})$. This term arises from the "artificial abort" step. (In [31], the $\mathbf{T}_{\text{abort}}(\epsilon, n, q)$ term only has a qn factor in place of the $q^2 n^2$ factor we show. However, the step requires performing q times up to n operations over the integers modulo $4q$ for each of the $\ell = \mathcal{O}(qn\epsilon^{-2} \ln \epsilon^{-1} \ln qn)$ vectors selected, so our term represents the actual cost.)

3 New Proof of Waters' IBE without Artificial Aborts

We start with some high-level discussion regarding Waters' original proof and the reason for the artificial abort. First, one would hope to specify a simulator

that, given IND-CPA adversary $\mathcal{A}$ that attacks the IBE scheme using q **Extract** queries and gains advantage ϵ, solves the DBDH problem with advantage not drastically worse than ϵ. But $\mathcal{A}$ can make **Extract** queries that force any conceivable simulator to fail, i.e. have to abort. This means that the advantage against DBDH is conditioned on $\mathcal{A}$ not causing an abort, and so it could be the case that $\mathcal{A}$ achieves ϵ advantage in the normal IND-CPA experiment but almost always causes aborts for the simulator. In this (hypothetical) case, the simulator could not effectively make use of the adversary, and the proof fails.

On the other hand, if one can argue that the lower and upper bounds on the probability of $\mathcal{A}$ causing an abort to occur are close (i.e. the case above does not occur), then the proof would go through. As Waters' points out [31], the natural simulator (that only aborts when absolutely necessarily) fails to provide such a guarantee. To compensate, Waters' introduced "artificial aborts". At the end of a successful simulation for the IBE adversary, the simulator $\mathcal{B}_{\mathsf{Wa}}$ used by Waters' generates $\mathcal{O}(qn\epsilon^{-2} \ln \epsilon^{-1} \ln qn)$ random vectors. These are used to estimate the probability that the **Extract** queries made by $\mathcal{A}$ cause an abort during any given execution of the simulator. The simulator then artificially aborts with some related probability. Intuitively, this forces the probability of aborting to be independent of $\mathcal{A}$'s particular queries. Waters' shows that $\mathcal{B}_{\mathsf{Wa}}$ provides the aforementioned guarantee of close lower and upper bounds and the proof goes through.

The artificial abort step seems strange because $\mathcal{B}_{\mathsf{Wa}}$ is forcing itself to fail even when it appears to have succeeded. The concrete security also suffers because the running time of the simulator goes up by $\mathbf{T}_{\text{abort}}(\epsilon, n, q)$ as shown in (7).

The rest of this section is devoted to proving the next theorem, which establishes the security of the Waters' IBE scheme without relying on an artificial abort step.

Theorem 1. Fix pairing parameters $\mathsf{GP} = (\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, p, \mathbf{e}, \psi)$ and an integer $n \geq 1$, and let $\mathsf{Wa} = (\mathsf{Pg}, \mathsf{Kg}, \mathsf{Enc}, \mathsf{Dec})$ be the Waters IBE scheme associated to GP and n. Let $\mathcal{A}$ be an ind-cpa adversary against Wa which has advantage $\epsilon = \mathbf{Adv}^{\text{ind-cpa}}_{\mathsf{Wa}}(\mathcal{A}) > 0$ and makes at most $q \in [1 \,..\, p\epsilon/9n]$ queries to its **Extract** oracle. Then there is a dbdh adversary $\mathcal{B}$ such that

$$\mathbf{Adv}^{\text{dbdh}}_{\mathsf{GP}}(\mathcal{B}) \geq \frac{\epsilon^2}{27qn + 3\epsilon} \text{ , and} \tag{8}$$

$$\mathbf{T}(\mathcal{B}) = \mathbf{T}(\mathcal{A}) + \mathbf{T}_{\text{sim}}(n, q) \tag{9}$$

where $\mathbf{T}_{\text{sim}}(n, q)$ was defined by (6). □

The limitations on q—namely, $1 \leq q \leq p/4n$ in Waters' result and $1 \leq q \leq p\epsilon/9n$ in ours—are of little significance since in practice $p \geq 2^{160}$, $\epsilon \geq 2^{-80}$, and $n = 160$. For $q = 0$ there is a separate, tight reduction. The remainder of this section is devoted to the proof of Theorem 1.

Some definitions. Let $m = \lceil 9q/\epsilon \rceil$ and let $X = [-n(m-1) \,..\, 0] \times [0 \,..\, m-1] \times \cdots \times [0 \,..\, m-1]$ where the number of copies of $[0 \,..\, m-1]$ is n. For $\mathbf{x} \in X$,

$\mathbf{y} \in \mathbb{Z}_p^{n+1}$ and $I \in \{0,1\}^n$ we let

$$\mathsf{F}(\mathbf{x}, I) = \mathbf{x}[0] + \sum_{i=1}^{n} \mathbf{x}[i]I[i] \quad \text{and} \quad \mathsf{G}(\mathbf{y}, I) = \mathbf{y}[0] + \sum_{i=1}^{n} \mathbf{y}[i]I[i] \bmod p\,. \tag{10}$$

Note that while the computation of G above is over $\mathbb{Z}_p$, that of F is over $\mathbb{Z}$.

Adversary $\mathcal{B}$. Our DBDH adversary $\mathcal{B}$ is depicted in Figure 4, where the *simulation subroutines* KgS and EncS are specified below. There are two main differences between our adversary and that of Waters'. The first is that in our case the parameter m is $\mathcal{O}(q/\epsilon)$ while in Waters' case it is $\mathcal{O}(q)$. The second difference of course is that Waters' adversary $\mathcal{B}_{\mathsf{Wa}}$, unlike ours, includes the artificial abort step. Once $\mathcal{A}$ has terminated, this step selects $l = \mathcal{O}(qn\epsilon^{-2}\ln(\epsilon^{-1})\ln(qn))$ new random vectors $\mathbf{x}_1, \ldots, \mathbf{x}_l$ from X. Letting $I_1, \ldots, I_q$ denote the identities queried by $\mathcal{A}$ to its **Extract** oracle and I_0 the identity queried to the **LR** oracle, it then evaluates $\mathsf{F}(\mathbf{x}_i, I_j)$ for all $1 \le i \le l$ and $0 \le j \le q$, and uses these values to approximate the probability that bad is set. It then aborts with some related probability. Each computation of F takes $\mathcal{O}(n)$ time, and there are q such computations for each of the l samples, accounting for the estimate of (7). In addition there are some minor differences between the adversaries. For example, $\mathbf{x}$ is chosen differently. (In [31] it is taken from $[0\,..\,m-1]^{n+1}$, and an additional value $k \in [0\,..\,n]$, which we do not have, is mixed in.)

We note that our adversary in fact *never* aborts. Sometimes, it is clearly returning incorrect answers (namely $\perp$) to $\mathcal{A}$'s queries. Adversary $\mathcal{A}$ will recognize this, and all bets are off as to what it will do. Nonetheless, $\mathcal{B}$ continues the execution of $\mathcal{A}$. Our analysis will show that $\mathcal{B}$ has the claimed properties regardless.

An analysis of the running time of $\mathcal{B}$, justifying equations (6) and (9), is given in the full version of the paper [3].

Simulation subroutines. We define the subroutines that $\mathcal{B}$ utilizes to answer **Extract** and **LR** queries. We say that $(g_1, g_2, A_2, A_1, B_2, B_1, \mathbf{x}, \mathbf{y}, \mathbf{u}, S, W)$ are *simulation parameters* if: $g_2 \in \mathbb{G}_2^*$; $g_1 = \psi(g_2) \in \mathbb{G}_1^*$; $A_2 \in \mathbb{G}_2$; $A_1 = \psi(A_2) \in \mathbb{G}_1$; $B_2 \in \mathbb{G}_2$; $B_1 = \psi(B_2) \in \mathbb{G}_1$; $\mathbf{x} \in X$; $\mathbf{y} \in \mathbb{Z}_p^{n+1}$; $\mathbf{u}[j] = B_1^{\mathbf{x}[j]} g_1^{\mathbf{y}[j]}$ for $j \in [0\,..\,n]$; $S \in \mathbb{G}_2$; and $W \in \mathbb{G}_T$. We define the following procedures:

proc. $\mathsf{KgS}(g_1, g_2, A_2, A_1, B_1, \mathbf{x}, \mathbf{y}, I)$

$r \xleftarrow{\$} \mathbb{Z}_p$; $w \leftarrow \mathsf{F}(\mathbf{x}, I)^{-1} \bmod p$

$L_1 \leftarrow B_1^{\mathsf{F}(\mathbf{x},I)\cdot r} g_1^{\mathsf{G}(\mathbf{y},I)\cdot r} A_1^{-\mathsf{G}(\mathbf{y},I)w}$

$L_2 \leftarrow g_2^r A_2^{-w}$

Ret (L_1, L_2)

proc. $\mathsf{EncS}(S, W, M, \mathbf{y}, I)$

$C_1 \leftarrow W \cdot M$

$C_2 \leftarrow S$; $C_3 \leftarrow \psi(S)^{\mathsf{G}(\mathbf{y},I)}$

Ret (C_1, C_2, C_3)

Note that if $\mathsf{F}(\mathbf{x}, I) \neq 0$ then $\mathsf{F}(\mathbf{x}, I) \not\equiv 0 \pmod p$ so the quantity w computed by KgS is well-defined whenever $\mathsf{F}(\mathbf{x}, I) \neq 0$. This is because the absolute value of $\mathsf{F}(\mathbf{x}, I)$ is at most

$$n(m-1) = n\left(\left\lceil \frac{9q}{\epsilon} \right\rceil - 1\right) < \frac{9nq}{\epsilon} \le p\,, \tag{11}$$

Adversary $\mathcal{B}(g_1, g_2, A_2, B_2, S, W)$:
$c \xleftarrow{\$} \{0,1\}$; $A_1 \leftarrow \psi(A_2)$; $B_1 \leftarrow \psi(B_2)$
For $j = 0, \ldots, n$ do
  $\mathbf{y}[j] \xleftarrow{\$} \mathbb{Z}_p$
  If $j = 0$ then $\mathbf{x}[j] \xleftarrow{\$} [-n(m-1) \,..\, 0]$
  Else $\mathbf{x}[j] \xleftarrow{\$} [0 \,..\, m-1]$
  $\mathbf{u}[j] \leftarrow B_1^{\mathbf{x}[j]} g_1^{\mathbf{y}[j]}$
$mpk \leftarrow (g_2, A_1, B_2, \mathbf{u})$
Run $\mathcal{A}(mpk)$, answering queries by

query Extract(I):
$sk(I) \leftarrow \perp$
If $\mathsf{F}(\mathbf{x}, I) = 0$ then $\mathsf{bad} \leftarrow \mathsf{true}$
Else $sk(I) \xleftarrow{\$} \mathsf{KgS}(g_1, g_2, A_2, A_1, B_1, \mathbf{x}, \mathbf{y}, I)$
Ret $sk(I)$

query LR(I, M_0, M_1):
$C \leftarrow \perp$
If $\mathsf{F}(\mathbf{x}, I) \neq 0$ then $\mathsf{bad} \leftarrow \mathsf{true}$
Else $C \leftarrow \mathsf{EncS}(S, W, M_c, \mathbf{y}, I)$
Ret C

$\mathcal{A}$ finishes, returning bit c'
If $\mathsf{bad} = \mathsf{true}$ then $c' \xleftarrow{\$} \{0,1\}$
If $c = c'$ then Ret 1 else Ret 0

Fig. 4. Adversary $\mathcal{B}$.

the last because of the restriction on q in the theorem statement. The next lemma captures two facts about the simulation subroutines, which we will use in our analysis.

Lemma 2. Let $(g_1, g_2, A_2, A_1, B_2, B_1, \mathbf{x}, \mathbf{y}, \mathbf{u}, S, W)$ be simulation parameters. Let $I \in \{0,1\}^n$. Let $mpk = (g_2, A_1, B_2, \mathbf{u})$. Let b be the discrete log of B_1 to base g_1 and let $msk = A_1^b$. Let s be the discrete log of S to base g_2. Then if $\mathsf{F}(\mathbf{x}, I) \neq 0$ the outputs of $\mathsf{KgS}(g_1, g_2, A_2, A_1, B_1, \mathbf{x}, \mathbf{y}, I)$ and $\mathsf{Kg}(mpk, msk, I)$ are identically distributed. Also if $\mathsf{F}(\mathbf{x}, I) = 0$ then for any $M \in \mathsf{MsgSp}$, the output of $\mathsf{EncS}(S, W, M, \mathbf{y}, I)$ is $(W \cdot M, S, H(\mathbf{u}, I)^s)$. $\square$

The proof of Lemma 2, which follows arguments given in [31], is given in the full version [3].

Overview. Consider executing $\mathcal{B}$ in game $\mathrm{DBDH}_{\mathsf{GP}}$. If $d = 1$, then Lemma 2 implies that adversary $\mathcal{B}$ correctly answers oracle queries as long as it does not set bad. On the other hand if $d = 0$ then $\mathcal{B}$'s output is a random bit. Attempting to conclude by showing that bad is seldom set fails, however, because in fact it will be set with probability close to 1. Alternatively, if one could show that the setting of bad is independent of the correctness of $\mathcal{B}$'s output, then one could

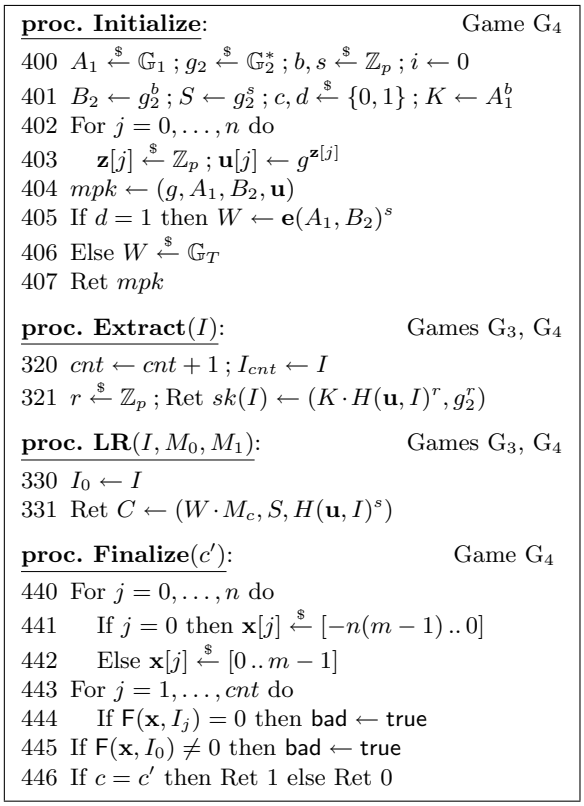

Fig. 5. The game G_4.

conclude by multiplying the probabilities of these events. The difficulty in the proof is that this independence does not hold. Waters' artificial abort step is one way to compensate. However, we have dropped this (expensive) step and propose nonetheless to push an argument through. We will first use the game sequence G_0–G_4 to arrive at a game where the choice of $\mathbf{x}$ is independent of the game output. The subtle point is that this *still* does not provide independence between setting bad and the game output, because the identities chosen by $\mathcal{A}$ for its oracle queries affect both events. The first step in addressing this is a conditioning argument based on Lemma 4 which allows us to express a lower bound on the advantage of $\mathcal{B}$ in terms of probabilities $\gamma(\mathbf{I})$ associated to different queried identities. The crucial insight is that Lemma 5 gives upper and lower bounds on these probabilities that are very close, specifically within a factor of $1 - \epsilon$ of each other, due to our choice of $m = \mathcal{O}(q/\epsilon)$ rather than merely the $m = \mathcal{O}(q)$ of [31]. Using this allows us to conclude easily.

THE GAME PLAYING SEQUENCE. Assume without loss of generality that $\mathcal{A}$ always makes exactly q queries to its **Extract** oracle rather than at most q. The proof

starts using a sequence of games $\mathrm{G}_0 - \mathrm{G}_4$ to move from $\mathcal{B}$ running in the DBDH experiment to a game G_4 (shown in Figure 5) that is essentially the IND-CPA experiment, though with some additional bookkeeping. This transition is critical since it moves to a setting where the choice of $\mathbf{x}$ is clearly independent of $\mathcal{A}$'s choices. For brevity, we capture this game playing sequence via the following lemma whose proof is given in the full version [3]. Let GD_4 denote the event that $\mathrm{G}_4^{\mathcal{A}}$ does not set bad.

Lemma 3. $\mathbf{Adv}^{\mathrm{dbdh}}_{\mathsf{GP}}(\mathcal{B}) = 2 \cdot \Pr\left[\, \mathrm{G}_4^{\mathcal{A}} \Rightarrow d \wedge \mathsf{GD}_4 \,\right] - \Pr\left[\, \mathsf{GD}_4 \,\right]$ $\square$

We now reach a subtle point. Consider the following argument: "The event GD_4 depends only on $\mathbf{x}$, which is chosen at lines 440–442 *after* the adversary and game outputs are determined. So GD_4 is independent of the event that $\mathrm{G}_4^{\mathcal{A}} \Rightarrow d$." If we buy this, the probability of the conjunct in Lemma 3 becomes the product of the probability of the constituent events, and it is quite easy to conclude. However, the problem is that the argument in quotes above is wrong. The reason is that GD_4 also depends on $I_0, \ldots, I_q$ and these are adversary queries whose values are not independent of the game output. Waters' compensates for this via the artificial abort step, but we do not have this step in $\mathcal{B}$ and propose to complete the analysis anyway.

Conditional independence lemma. Let

$$\mathsf{ID} = \{(I_0, \ldots, I_q) \in (\{0,1\}^n)^{q+1} \;:\; \forall i \in [1\,..\,q]\ (I_0 \neq I_i)\}\,.$$

For $(I_0, \ldots, I_q) \in \mathsf{ID}$ let

$$\gamma(I_0, \ldots, I_q) = \Pr\left[\, \mathsf{F}(\mathbf{x}, I_0) = 0 \wedge \mathsf{F}(\mathbf{x}, I_1) \neq 0 \wedge \cdots \wedge \mathsf{F}(\mathbf{x}, I_q) \neq 0 \,\right]$$

where the probability is taken over $\mathbf{x} \xleftarrow{\$} X$. This is the probability of GD_4 under a particular sequence of queried identities $I_0, \ldots, I_q$. (We stress that here we first fix $I_0, \ldots, I_q$ and then choose $\mathbf{x}$ at random.) If $\gamma(I_0, \ldots, I_q)$ were the same for all $(I_0, \ldots, I_q) \in \mathsf{ID}$ then the problem discussed above would be resolved. The difficulty is that $\gamma(I_0, \ldots, I_q)$ varies with $I_0, \ldots, I_q$. Our next lemma is the main tool to resolve the independence problem. Roughly it says that if we consider the conditional space obtained by conditioning on a particular sequence $I_0, \ldots, I_q$ of queried identities, then independence does hold. To formalize this, let $\mathsf{Q}(\mathbf{I})$ be the event that the execution of G_4 with $\mathcal{A}$ results in the identities $I_0, \ldots, I_q$ being queried by $\mathcal{A}$, where $\mathbf{I} = (I_0, \ldots, I_q)$. Then:

Lemma 4. For any $\mathbf{I} \in \mathsf{ID}$,

$$\Pr\left[\, \mathrm{G}_4^{\mathcal{A}} \Rightarrow d \wedge \mathsf{GD}_4 \wedge \mathsf{Q}(\mathbf{I}) \,\right] = \gamma(\mathbf{I}) \cdot \Pr\left[\, \mathrm{G}_4^{\mathcal{A}} \Rightarrow d \wedge \mathsf{Q}(\mathbf{I}) \,\right] \tag{12}$$

$$\Pr\left[\, \mathsf{GD}_4 \wedge \mathsf{Q}(\mathbf{I}) \,\right] = \gamma(\mathbf{I}) \cdot \Pr\left[\, \mathsf{Q}(\mathbf{I}) \,\right] \ \square \tag{13}$$

Proof. The set of coin tosses underlying the execution of G_4 with $\mathcal{A}$ can be viewed as a cross product $\Omega = \Omega' \times X$, meaning each member ω of Ω is a pair $\omega = (\omega', \mathbf{x})$ where $\mathbf{x}$ is the choice made at lines 440–442 and ω' is all the rest of the game and adversary coins. For any $\mathbf{I} \in \mathsf{ID}$ let $\Omega'(\mathbf{I})$ be the set of all $\omega \in \Omega'$ such that the execution with ω produces $\mathbf{I}$ as the sequence of queried

identities. (Which $\mathbf{I}$ is produced depends only on ω' since $\mathbf{x}$ is chosen after $\mathcal{A}$ has terminated.) Let Ω'_{out} be the set of all $\omega \in \Omega'$ on which the execution outputs d. (Again, this is determined only by ω' and not $\mathbf{x}$.) Let $X_{\text{gd}}(\mathbf{I})$ be the set of all $\mathbf{x} \in X$ such that

$$\mathsf{F}(\mathbf{x}, I_0) = 0 \wedge \mathsf{F}(\mathbf{x}, I_i) \neq 0 \wedge \cdots \wedge \mathsf{F}(\mathbf{x}, I_q) \neq 0\,,$$

where $\mathbf{I} = (I_0, \ldots, I_q)$. Now observe that the set of coins leading to $\mathrm{G}_4^{\mathcal{A}} \Rightarrow d$ is $\Omega'_{\text{out}} \times X$ and the set of coins leading to $\mathsf{GD}_4 \wedge \mathsf{Q}(\mathbf{I})$ is $\Omega'(\mathbf{I}) \times X_{\text{gd}}(\mathbf{I})$. So

$$\begin{aligned}
\Pr\left[\,\mathrm{G}_4^{\mathcal{A}} \Rightarrow d \wedge \mathsf{GD}_4 \wedge \mathsf{Q}(\mathbf{I})\,\right] &= \frac{|(\Omega'_{\text{out}} \times X) \cap (\Omega'(\mathbf{I}) \times X_{\text{gd}}(\mathbf{I}))|}{|\Omega' \times X|} \\
&= \frac{|(\Omega'_{\text{out}} \cap \Omega'(\mathbf{I})) \times X_{\text{gd}}(\mathbf{I})|}{|\Omega' \times X|} = \frac{|\Omega'_{\text{out}} \cap \Omega'(\mathbf{I})| \cdot |X_{\text{gd}}(\mathbf{I})|}{|\Omega'| \cdot |X|} \\
&= \frac{|\Omega'_{\text{out}} \cap \Omega'(\mathbf{I})| \cdot |X|}{|\Omega'| \cdot |X|} \cdot \frac{|X_{\text{gd}}(\mathbf{I})|}{|X|} = \frac{|(\Omega'_{\text{out}} \cap \Omega'(\mathbf{I})) \times X|}{|\Omega' \times X|} \cdot \frac{|X_{\text{gd}}(\mathbf{I})|}{|X|}\,.
\end{aligned}$$

But the first term above is $\Pr\left[\,\mathrm{G}_4^{\mathcal{A}} \Rightarrow d \wedge \mathsf{Q}(\mathbf{I})\,\right]$ while the second is $\gamma(\mathbf{I})$, establishing (12). For (13) we similarly have

$$\begin{aligned}
\Pr\left[\,\mathsf{GD}_4 \wedge \mathsf{Q}(\mathbf{I})\,\right] &= \frac{|\Omega'(\mathbf{I}) \times X_{\text{gd}}(\mathbf{I})|}{|\Omega' \times X|} = \frac{|\Omega'(\mathbf{I})|}{|\Omega'|} \cdot \frac{|X_{\text{gd}}(\mathbf{I})|}{|X|} \\
&= \frac{|\Omega'(\mathbf{I})| \cdot |X|}{|\Omega'| \cdot |X|} \cdot \frac{|X_{\text{gd}}(\mathbf{I})|}{|X|} = \frac{|\Omega'(\mathbf{I}) \times X|}{|\Omega' \times X|} \cdot \frac{|X_{\text{gd}}(\mathbf{I})|}{|X|}\,.
\end{aligned}$$

But the final terms above are $\Pr\left[\,\mathsf{Q}(\mathbf{I})\,\right]$ and $\gamma(\mathbf{I})$, respectively, establishing (13).

Analysis continued. Let $\gamma_{\min}$ be the smallest value of $\gamma(I_0, \ldots, I_q)$ taken over all $(I_0, \ldots, I_q) \in \mathsf{ID}$. Let $\gamma_{\max}$ be the largest value of $\gamma(I_0, \ldots, I_q)$ taken over all $(I_0, \ldots, I_q) \in \mathsf{ID}$. Using Lemma 3 we have that

$$\begin{aligned}
\mathbf{Adv}^{\text{dbdh}}_{\mathsf{GP}}(\mathcal{B}) &= 2 \cdot \Pr\left[\,\mathrm{G}_4^{\mathcal{A}} \Rightarrow d \wedge \mathsf{GD}_4\,\right] - \Pr\left[\,\mathsf{GD}_4\,\right] \\
&= \sum_{\mathbf{I} \in \mathsf{ID}} 2 \cdot \Pr\left[\,\mathrm{G}_4^{\mathcal{A}} \Rightarrow d \wedge \mathsf{GD}_4 \wedge \mathsf{Q}(\mathbf{I})\,\right] - \sum_{\mathbf{I} \in \mathsf{ID}} \Pr\left[\,\mathsf{GD}_4 \wedge \mathsf{Q}(\mathbf{I})\,\right]
\end{aligned}$$

and applying Lemma 4:

$$\begin{aligned}
\mathbf{Adv}^{\text{dbdh}}_{\mathsf{GP}}(\mathcal{B}) &= \sum_{\mathbf{I} \in \mathsf{ID}} 2\gamma(\mathbf{I}) \cdot \Pr\left[\,\mathrm{G}_4^{\mathcal{A}} \Rightarrow d \wedge \mathsf{Q}(\mathbf{I})\,\right] - \sum_{\mathbf{I} \in \mathsf{ID}} \gamma(\mathbf{I}) \cdot \Pr\left[\,\mathsf{Q}(\mathbf{I})\,\right] \\
&\geq \gamma_{\min} \underbrace{\sum_{\mathbf{I} \in \mathsf{ID}} 2 \cdot \Pr\left[\,\mathrm{G}_4^{\mathcal{A}} \Rightarrow d \wedge \mathsf{Q}(\mathbf{I})\,\right]}_{= 2 \cdot \Pr\left[\,\mathrm{G}_4^{\mathcal{A}} \Rightarrow d\,\right]} - \gamma_{\max} \underbrace{\sum_{\mathbf{I} \in \mathsf{ID}} \Pr\left[\,\mathsf{Q}(\mathbf{I})\,\right]}_{= 1} \\
&\geq 2\gamma_{\min} \cdot \Pr\left[\,\mathrm{G}_4^{\mathcal{A}} \Rightarrow d\,\right] - \gamma_{\max}\,. \qquad (14)
\end{aligned}$$

Now

$$\begin{aligned}\Pr\left[\,\mathrm{G}_4^{\mathcal{A}} \Rightarrow d\,\right] &= \Pr\left[\,\mathrm{G}_4^{\mathcal{A}} \Rightarrow 1 \mid\ d=1\,\right]\Pr\left[\,d=1\,\right] + \Pr\left[\,\mathrm{G}_4^{\mathcal{A}} \Rightarrow 0 \mid\ d=0\,\right]\Pr\left[\,d=0\,\right] \\ &= \frac{1}{2}\cdot\Pr\left[\,\mathrm{G}_4^{\mathcal{A}} \Rightarrow 1 \mid\ d=1\,\right] + \frac{1}{2}\cdot\Pr\left[\,\mathrm{G}_4^{\mathcal{A}} \Rightarrow 0 \mid\ d=0\,\right] \\ &= \frac{1}{2}\cdot\left(\frac{1}{2}+\frac{1}{2}\cdot\mathbf{Adv}^{\text{ind-cpa}}_{\mathsf{Wa}}(\mathcal{A})\right)+\frac{1}{2}\cdot\frac{1}{2} \qquad (15)\\ &= \frac{1}{4}\cdot\mathbf{Adv}^{\text{ind-cpa}}_{\mathsf{Wa}}(\mathcal{A})+\frac{1}{2} \qquad (16)\end{aligned}$$

where we justify (15) as follows. In the case that $d = 0$, the value W is uniformly distributed over $\mathbb{G}_T$ and hence line 331 gives $\mathcal{A}$ no information about the bit c. So the probability that $c = c'$ at line 446 is 1/2. On the other hand if $d = 1$ then G_4 implements the IND-CPA$_{\mathsf{Wa}}$ game, so $2\cdot\Pr\left[\,\mathrm{G}_4^{\mathcal{A}} \Rightarrow 1 \mid\ d=1\,\right]-1 = \mathbf{Adv}^{\text{ind-cpa}}_{\mathsf{Wa}}(\mathcal{A})$ by (3). We substitute (16) into (14) and get

$$\begin{aligned}\mathbf{Adv}^{\text{dbdh}}_{\mathsf{GP}}(\mathcal{B}) &\geq 2\gamma_{\min}\left(\frac{1}{4}\mathbf{Adv}^{\text{ind-cpa}}_{\mathsf{Wa}}(\mathcal{A})+\frac{1}{2}\right)-\gamma_{\max} \\ &= \frac{\gamma_{\min}}{2}\mathbf{Adv}^{\text{ind-cpa}}_{\mathsf{Wa}}(\mathcal{A})+(\gamma_{\min}-\gamma_{\max})\,. \qquad (17)\end{aligned}$$

To finish the proof, we use the following:

Lemma 5. $\dfrac{1}{n(m-1)+1}\left(1-\dfrac{q}{m}\right) \leq \gamma_{\min} \leq \gamma_{\max} \leq \dfrac{1}{n(m-1)+1}$ $\square$

The proof of Lemma 5, based on ideas in [31], is given in the full version of the paper [3]. Let $\alpha = 1/(n(m-1)+1)$. Recall that $m = \lceil 9q/\epsilon \rceil \geq 9q/\epsilon$ where $\epsilon = \mathbf{Adv}^{\text{ind-cpa}}_{\mathsf{Wa}}(\mathcal{A})$. Then, applying Lemma 5 to (17) we get

$$\begin{aligned}\mathbf{Adv}^{\text{dbdh}}_{\mathsf{GP}}(\mathcal{B}) &\geq \frac{\alpha}{2}\left(1-\frac{q}{m}\right)\epsilon+\alpha\left(1-\frac{q}{m}\right)-\alpha \;=\; \alpha\left[\frac{1}{2}\left(1-\frac{q}{m}\right)\epsilon-\frac{q}{m}\right] \\ &\geq \alpha\left[\frac{1}{2}\left(1-\frac{q\epsilon}{9q}\right)\epsilon-\frac{q\epsilon}{9q}\right] \;=\; \frac{\alpha\epsilon}{18}(7-\epsilon) \\ &\geq \frac{\alpha\epsilon}{3}\,. \qquad (18)\end{aligned}$$

Inequality (18) is justified by the fact that $\epsilon \leq 1$. Using the fact that $m = \lceil 9q/\epsilon \rceil \leq 9q/\epsilon + 1$ and substituting in for α, we complete the derivation of our lower bound for $\mathcal{B}$:

$$\mathbf{Adv}^{\text{dbdh}}_{\mathsf{GP}}(\mathcal{B}) \geq \frac{\epsilon}{3}\cdot\frac{1}{n(m-1)+1} \geq \frac{\epsilon}{3}\cdot\frac{1}{n(9q/\epsilon)+1} = \frac{\epsilon^2}{27qn+3\epsilon}\,.$$

Acknowledgments

We thank Brent Waters for pointing out a bug in the proof of an earlier version of this paper. We thank Sarah Shoup for participating in early stages of this

work. We thank Dan Boneh and Xavier Boyen for comments on earlier drafts of this work. This work was supported in part by NSF grants CNS 0524765 and CNS 0627779 and a gift from Intel corporation.

References

1. M. Abdalla, D. Catalano, A. W. Dent, J. Malone-Lee, G. Neven, and N. P. Smart. Identity-based encryption gone wild. In M. Bugliesi, B. Preneel, V. Sassone, and I. Wegener, eds., *ICALP 2006*, vol. 4052 of *LNCS*, pp. 300–311, July 10–14, 2006. Springer-Verlag, Berlin, Germany.
2. M. Abdalla, E. Kiltz, and G. Neven. Generalized key delegation for hierarchical identity-based encryption. In *ESORICS*, vol. 4734 of *Lecture Notes in Computer Science*, pp. 139–154. Springer, 2007.
3. M. Bellare and T. Ristenpart. Simulation without the artificial abort: Simplified proof and improved concrete security for waters' ibe scheme (full version of this paper). Available from authors' home pages, Jan. 2009.
4. M. Bellare and P. Rogaway. The Security of Triple Encryption and a Framework for Code-Based Game-Playing Proofs. In S. Vaudenay, eds., *EUROCRYPT 2006*, vol. 4004 of *LNCS*, pp. 409–426, May 29 –June 1, 2006. Springer-Verlag, Berlin, Germany.
5. J. Birkett, A. W. Dent, G. Neven, and J. C. N. Schuldt. Efficient chosen-ciphertext secure identity-based encryption with wildcards. In J. Pieprzyk, H. Ghodosi, and E. Dawson, eds., *ACISP 07*, vol. 4586 of *LNCS*, pp. 274–292, July 2–4, 2007. Springer-Verlag, Berlin, Germany.
6. D. Boneh and X. Boyen. Efficient selective-id secure identity based encryption without random oracles. In C. Cachin and J. Camenisch, eds., *EUROCRYPT 2004*, vol. 3027 of *LNCS*, pp. 223–238, May 2–6, 2004. Springer-Verlag, Berlin, Germany.
7. D. Boneh and X. Boyen. Short signatures without random oracles. In C. Cachin and J. Camenisch, eds., *EUROCRYPT 2004*, vol. 3027 of *LNCS*, pp. 56–73, May 2–6, 2004. Springer-Verlag, Berlin, Germany.
8. D. Boneh and M. K. Franklin. Identity-based encryption from the Weil pairing. In J. Kilian, eds., *CRYPTO 2001*, vol. 2139 of *LNCS*, pp. 213–229, Aug. 19–23, 2001. Springer-Verlag, Berlin, Germany.
9. X. Boyen. General ad hoc encryption from exponent inversion IBE. In M. Naor, eds., *EUROCRYPT 2007*, vol. 4515 of *LNCS*, pp. 394–411, May 20–24 2007. Springer-Verlag, Berlin, Germany.
10. X. Boyen, Q. Mei, and B. Waters. Direct chosen ciphertext security from identity-based techniques. In V. Atluri, C. Meadows, and A. Juels, eds., *ACM CCS 05*, pp. 320–329, Nov. 7–11, 2005. ACM Press.
11. R. Canetti, S. Halevi, and J. Katz. Chosen-ciphertext security from identity-based encryption. In C. Cachin and J. Camenisch, eds., *EUROCRYPT 2004*, vol. 3027 of *LNCS*, pp. 207–222, May 2–6, 2004. Springer-Verlag, Berlin, Germany.
12. S. Chatterjee and P. Sarkar. Trading time for space: Towards an efficient IBE scheme with short(er) public parameters in the standard model. In D. Won and S. Kim, eds., *ICISC 05*, LNCS, pp. 424–440, Dec. 1–2, 2005. Springer-Verlag, Berlin, Germany.
13. S. Chatterjee and P. Sarkar. HIBE with short public parameters without random oracle. In *ASIACRYPT 2006*, LNCS, pp. 145–160. Springer-Verlag, Berlin, Germany, Dec. 2006.

14. D. Chaum, J.-H. Evertse, and J. van de Graaf. An improved protocol for demonstrating possession of discrete logarithms and some generalizations. In D. Chaum and W. L. Price, eds., *EUROCRYPT'87*, vol. 304 of *LNCS*, pp. 127–141, Apr. 13–15, 1987. Springer-Verlag, Berlin, Germany.
15. C. Cocks. An identity based encryption scheme based on quadratic residues. In B. Honary, eds., *Cryptography and Coding, 8th IMA International Conference*, vol. 2260 of *LNCS*, pp. 360–363, Dec. 17–19, 2001. Springer-Verlag, Berlin, Germany.
16. D. Freeman, M. Scott, and E. Teske. A taxonomy of pairing-friendly elliptic curves. Cryptology ePrint Archive, Report 2006/372, `http://eprint.iacr.org/`, 2007.
17. S. D. Galbraith, K. G. Paterson, and N. P. Smart. Pairings for cryptographers. Cryptology ePrint Archive, Report 2006/165. `http://eprint.iacr.org/`, 2006.
18. D. Galindo. The exact security of pairing based encryption and signature schemes. Based on a talk at Workshop on Provable Security, INRIA, Paris, 2004. `http://www.dgalindo.es/galindoEcrypt.pdf`, 2004.
19. C. Gentry. Practical identity-based encryption without random oracles. In S. Vaudenay, eds., *EUROCRYPT 2006*, vol. 4004 of *LNCS*, pp. 445–464, May 29 –June 1, 2006. Springer-Verlag, Berlin, Germany.
20. M. Green and S. Hohenberger. Blind identity-based encryption and simulatable oblivious transfer. In K. Kurosawa, eds., *ASIACRYPT 2007*, vol. 4833 of *LNCS*, pp. 265–282. Springer-Verlag, Berlin, Germany.
21. D. Hofheinz and E. Kiltz. Programmable hash functions and their applications. In D. Wagner, eds., *CRYPTO 2008*, vol. 5157 of *LNCS*, pp. 21–38, Aug. 17–21, 2008. Springer-Verlag, Berlin, Germany.
22. J. Horwitz and B. Lynn. Toward hierarchical identity-based encryption. In L. R. Knudsen, eds., *EUROCRYPT 2002*, vol. 2332 of *LNCS*, pp. 466–481, Apr. 28 – May 2, 2002. Springer-Verlag, Berlin, Germany.
23. J. Katz and N. Wáng. Efficiency improvements for signature schemes with tight security reductions. In *ACM CCS 03*, pp. 155–164, Oct. 27–30, 2003. ACM Press.
24. E. Kiltz and D. Galindo. Direct chosen-ciphertext secure IB-KEM without random oracles. In L. M. Batten and R. Safavi-Naini, eds., *ACISP 06*, vol. 4058 of *LNCS*, pp. 336–347, July 3–5, 2006. Springer-Verlag, Berlin, Germany.
25. A. K. Lenstra. Unbelievable security: Matching AES security using public key systems. In C. Boyd, eds., *ASIACRYPT 2001*, vol. 2248 of *LNCS*, pp. 67–86, Dec. 9–13, 2001. Springer-Verlag, Berlin, Germany.
26. A. K. Lenstra and E. R. Verheul. Selecting cryptographic key sizes. *Journal of Cryptology: the journal of the International Association for Cryptologic Research*, 14(4):255–293, 2001.
27. D. Naccache. Secure and practical identity-based encryption. Cryptology ePrint Archive, Report 2005/369, 2005. `http://eprint.iacr.org/`.
28. National Institute for Standards and Technology. Recommendation for Key Management Part 1: General (revised), 2005. NIST Special Publication 800-57.
29. A. Sahai and B. R. Waters. Fuzzy identity-based encryption. In R. Cramer, eds., *EUROCRYPT 2005*, vol. 3494 of *LNCS*, pp. 457–473, May 22–26, 2005. Springer-Verlag, Berlin, Germany.
30. R. Sakai and M. Kasahara. Id based cryptosystems with pairing on elliptic curve. Cryptology ePrint Archive, Report 2003/054, 2003. `http://eprint.iacr.org/`.
31. B. R. Waters. Efficient identity-based encryption without random oracles. In R. Cramer, eds., *EUROCRYPT 2005*, vol. 3494 of *LNCS*, pp. 114–127, May 22–26, 2005. Springer-Verlag, Berlin, Germany.

A Derivatives of Waters' IBE

A large body of research [10, 1, 5, 24, 27, 12, 13, 20] utilizes Waters' scheme. Recall that Waters' already proposed a heirarchical IBE scheme based on Wa in [31], and subsequently there have been numerous derivative works. All use the artificial abort, either due to a black-box reduction to Waters' (H)IBE or as an explicit step in a direct proof. Our new proof technique immediately benefits those schemes that utilize Waters' scheme directly (i.e. in a black-box manner). For the rest, we believe that our techniques can be applied but have not checked the details.

- Naccache [27] and Chatterjee and Sarkar [12, 13] independently and concurrently introduced a space-time trade-off for Wa that involves modifying the hash function utilized from $H(\mathbf{u}, I) = \mathbf{u}[0] \prod_{i=1}^{n} \mathbf{u}[i]^{I[i]}$ for $\mathbf{u} \in \mathbb{G}_1^{n+1}$ to $H'(\mathbf{u}, I) = \mathbf{u}[0] \prod_{i=1}^{\ell} \mathbf{u}[i]^{I[i]}$ where $\mathbf{u} \in \mathbb{G}_1^{\ell+1}$ and each $I[i]$ is now an n/ℓ-bit string. For appropriate choice of ℓ this will significantly reduce the number of elements included in the master public key. However the new choice of hash function impacts the reduction tightness, and since their proof includes just minor changes to Waters', our new reduction will increase the efficiency of this time/space trade-off for various security levels.
- In [10] BB_1- and Wa-based constructions of CCA-secure public-key encryption schemes and their proofs for the Wa case directly utilize artificial aborts.
- Kiltz and Galindo [24] propose a construction of CCA-secure identity-based key encapsulation that is a modified version of Wa.
- Wildcard IBE [1, 5] is a generalization of heirarchical IBE that allows encryption to identities that include wildcards, e.g. "*@anonymous.com". In [1] a wildcard IBE scheme is proposed that utilizes the Waters HIBE scheme, and the proof is black-box to it. In [5] a wildcard identity-based KEM is produced based (in a non-black-box manner) on Waters' IBE.
- Wicked IBE [2] allows generation of private keys for wildcard identities. These private keys can then be used to generate derivative keys that replace the wildcards with any concrete identity string. They suggest using the Waters' HIBE scheme to achieve full security in their setting.
- Blind IBE, as introduced by Green and Hohenberger [20], enables the "trusted" master key generator to generate a private key for an identity without learning anything about the identity. To prove a Waters'-based blind IBE scheme secure they utilize the Naccache proof [27] (mentioned above). They utilize blind IBE schemes to build efficient and fully-simulatable oblivious transfer protocols based on the non-interactive assumptions inherited from the BDH-based IBE schemes used.