Direct Chosen-Ciphertext Secure Identity-Based Key Encapsulation without Random Oracles

Eike Kiltz[1] David Galindo[2]

August 4, 2009

[1] Cryptology and Information Security Research Theme
CWI Amsterdam, The Netherlands
kiltz@cwi.nl

[2] University of Luxembourg
david.galindo@uni.lu

Abstract

We describe a practical identity-based encryption scheme that is secure in the standard model against chosen-ciphertext attacks. Our construction applies "direct chosen-ciphertext techniques" to Waters' chosen-plaintext secure scheme and is not based on hierarchical identity-based encryption. Furthermore, we give an improved concrete security analysis for Waters' scheme. As a result, one can instantiate the scheme in smaller groups, resulting in efficiency improvements.

1 Introduction

An Identity-Based Encryption (IBE) scheme is a public-key encryption scheme where any string is a valid public key. In particular, email addresses and dates can be public keys. The ability to use identities as public keys minimizes the need to distribute public key certificates. The concept of IBE was proposed by Shamir [54] in the early eighties, but coming up with a satisfactory instantiation of it remained an open problem for almost two decades. It was not until 2001 that IBE systems were constructed by using bilinear maps [50, 12, 13]. In particular, Boneh and Franklin [12, 13] proposed formal security notions for IBE systems and designed a fully functional secure IBE scheme using bilinear maps. This scheme and the tools developed in its design have been successfully applied in numerous cryptographic settings, transcending by far the identity-based cryptography framework. An alternative but less efficient IBE construction based on quadratic residues was proposed by Cocks [20]. IBE is currently in the process of getting standardized — the new IEEE P1363.3 standard for "Identity-Based Cryptographic Techniques using Pairings" is currently in preparation [33], as well as the IETF memos RFC 5091, RFC 5408 and RFC 5409 [35].

All the above IBE schemes provide security against *chosen-ciphertext attacks* through the Fujisaki-Okamoto [24] transformation. In a chosen ciphertext attack [48], the adversary is given access to a decryption oracle that allows him to obtain the decryptions of ciphertexts of his choosing. Intuitively, security in this setting means that an adversary obtains effectively no information about encrypted messages, provided the corresponding ciphertexts are never submitted to the decryption oracle. For different reasons, the notion of chosen-ciphertext security has emerged as the "right" notion of security for encryption schemes. We stress that, in general, chosen-ciphertext security is a much stronger security requirement than chosen-plaintext attacks [2], where in the latter an attacker is not given access to the decryption oracle.

The drawback of the IBE scheme from Boneh-Franklin and Cocks is that security can only be guaranteed in the *random oracle* model [5], i.e., in an idealized world where all parties get black-box access to a truly random function. Unfortunately a proof in the random oracle model can only serve as a heuristic argument and has proved to possibly lead to insecure schemes in the standard model (see, e.g., [16]). More importantly, there exist results [22] indicating that even certain natural cryptographic schemes (such as full-domain hash signatures [6]) will always remain in the grey area of schemes having a proof in the random oracle yet are "provably unprovable" in the standard model.

Waters' IBE. To fill this gap Waters [58] presents the first practical Identity-Based Encryption scheme that is chosen-plaintext secure without random oracles. The proof of his scheme makes use of an algebraic method first used by Boneh and Boyen [8, 9] and security of the scheme is based on the Bilinear Decisional Diffie-Hellman (BDDH) assumption. However, Waters' plain IBE scheme is insecure against chosen-ciphertext attacks.

From 2-level Hierarchical IBE to chosen-chipertext secure IBE. Hierarchical identity-based encryption (HIBE) [32, 27] is a generalization of IBE allowing for hierarchical delegation of decryption keys. Recent results from Boneh, Canetti, Halevi, and Katz [17, 14, 11] show a *generic* transformation from any chosen-plaintext secure 2-level HIBE scheme to a chosen-ciphertext secure IBE scheme. We will refer to it as the BCHK transformation. Since Waters' IBE scheme can naturally be extended to a 2-level HIBE this implies a chosen-ciphertext secure IBE in the standard model. Key size, as well as the security reduction of the resulting scheme are comparable to the ones from Waters' IBE. However, the transformation involves some symmetric overhead to the ciphertext in form of a one-time signature or a MAC/commitment scheme with their respective keys.

Direct chosen-ciphertext techniques for public-key encryption. In [15, 38] "direct chosen-ciphertext" techniques were developed to improve efficiency of certain concrete public-key encryption schemes obtained from the BCHK transformation (applied to the IBE-schemes from [8]). Their methods are no longer generic but for particular encryption schemes [15, 38, 39] the overhead of the one-time signature or MAC can be completely avoided.

Identity-based Key Encapsulation. Instead of providing the full functionality of an IBE scheme, in many applications it is sufficient to let sender and receiver agree on a common random session key. This can be accomplished with an *identity-based key encapsulation mechanism* (IB-KEM) as formalized in [21, 7]. Any IB-KEM can be bootstrapped to a full IBE scheme by adding a symmetric encryption scheme (also called data encapsulation scheme — DEM) with appropriate security properties [21]. There are a numerous practical reasons to prefer a IB-KEM over an IBE scheme, which is why for traditional public-key encryption the modular KEM/DEM approach is incorporated in many recent standards (e.g., [55, 1, 34]).

1.1 Our Contributions

Our contributions can be summarized as follows.

A direct chosen-ciphertext secure IB-KEM based on Waters' IBE. Our main idea is to extend the "direct chosen-ciphertext" techniques from [15, 38] to the the identity-based setting. We enhance the IB-KEM version of Waters *chosen-plaintext* secure IBE by adding some redundant information to the ciphertext, consisting of a single group element, to make it *chosen-ciphertext* secure. This information is used to check whether a given IB-KEM ciphertext was "properly generated" by the encryption algorithm or not; if so decryption is done as before, otherwise the ciphertext is simply rejected. Intuitively, this "consistency check" is what gives us the necessary leverage to deal with the stronger chosen-ciphertext attacks. Unfortunately, implementing the consistency check is relatively expensive and an equivalent "implicit rejection" method is used to improve efficiency. This provides a direct construction of a chosen-ciphertext secure IB-KEM that is not explicitly derived from hierarchical techniques [11]. Like Waters' scheme, our scheme can be proved secure under the BDDH assumption in pairing groups. Furthermore, our IB-KEM scheme can be extended in a natural way to obtain a chosen-ciphertext secure HIB-KEM.

A tighter security reduction. In terms of concrete security, our security reduction is significantly tighter than the one given by Waters [58]. (Our new analysis can be applied to both Water's original scheme and our chosen-ciphertext secure IB-KEM.) More precisely, let $\mathcal{A}$ be an adversary against Waters' IBE scheme that runs in time at most $\mathbf{T}_{\mathcal{A}}$, makes at most q queries to its key-derivation oracle and has advantage $\varepsilon_{\mathcal{A}}$. Then [58, Theorem 1] presents a BDDH adversary $\mathcal{B}$ that runs in time at most $\mathbf{T}_{\mathcal{B}}$ and has advantage $\varepsilon_{\mathcal{B}}$ such that $\varepsilon_{\mathcal{B}} = \Omega(\varepsilon_{\mathcal{A}}/nq)$ and $\mathbf{T}_{\mathcal{B}} = \mathbf{T}_{\mathcal{A}} + \mathbf{T}_{\text{sim}} + \mathbf{T}_{\text{abort}}$, where: n is the bit-length of the identities; $\mathbf{T}_{\text{sim}} = q \cdot \mathbf{T}_{\mathbb{PG}}$ and $\mathbf{T}_{\mathbb{PG}}$ is the time for one exponentiation/pairing computation in $\mathbb{PG}$;

$\mathbf{T}_{\mathbb{Z}}$ is the time for one addition over integers smaller than $2q$; and ignoring log-terms

$$\mathbf{T}_{\text{abort}}(k) = \tilde{\mathcal{O}}\left(q^2 n^2 \cdot \varepsilon_{\mathcal{A}}^{-2}(k)\right) \cdot \mathbf{T}_{\mathbb{Z}}$$

Here $\mathbf{T}_{\text{abort}}$ denotes the time $\mathcal{B}$ needs to compute the probability whether it has to do an "artificial abort." Actually, $\mathbf{T}_{\text{abort}}$ as computed in [58] only shows a qn factor, but this was corrected in [4] to a q^2n^2 factor. By an improved analysis we can reduce the running time of $\mathbf{T}_{\text{abort}}$ to

$$\mathbf{T}_{\text{abort}}(k) = \tilde{\mathcal{O}}\big(n^3 \cdot \varepsilon_{\mathcal{A}}^{-2}(k)\big) \cdot \mathbf{T}_{\mathbb{Z}} \ ,$$

while the success probability stays the same. For concreteness, realistic values for $k = 80$ bit security are $q = 2^{30}$ and $n = 160$, so our reduction is significantly tighter than the one by Waters. As a result, one can securely use smaller groups, resulting in significant efficiency improvements. At a technical level our improved reduction makes use of a lower and an upper bound on the abortion probability during the execution of $\mathcal{B}$ (cf. Lemmas 6.2 and 6.3), whereas Waters only provides a lower bound. This makes it possible to substantially decrease the number of samples the simulator has to compute in order to approximate the proability it has to perform an artificial abort. We stress that our proof inherits the "artificial abort" technique by Waters.

A RIGOROUS GAME-BASED PROOF. The proof of Waters' IBE is already quite complex and has many technical parts that we found hard to verify. Additionally, many other results (e.g., [15, 18, 43]) already use ingredients of Waters' IBE, some more or less in a "black-box" manner which makes verification nearly impossible without having completely understood the original work. Motivated by this we give a rigorous, games-based proof of our result that can be easily understood and verified.

1.2 Comparison and Related Work

We carefully review all known chosen-ciphertext secure IBE constructions and make an extensive comparison with our scheme. It turns out that, to the best of our knowledge, our scheme is the most efficient chosen-ciphertext secure IBE scheme in the standard model based on the the BDDH assumption.

In (the full version of) [15] a technique is sketched how to avoid the BCHK transformation to get a direct chosen-ciphertext secure IB-KEM construction based on Waters' 2-level HIBE. Compared to our IBE, however, this construction has a weaker security reduction and nearly doubles the public key size. We mention other chosen-ciphertext secure IBE scheme that were proposed concurrently or after the publication of the extended abstract of this article [40]. The one by Gentry [26] relies on a much stronger security assumption, the q-ABDHE assumption, where the strength of the assumption degrades on the number of established user secret keys. Even though it has relatively short public-keys, the ciphertext size of Gentry's scheme is much larger, resulting in a bigger disadvantage. The scheme by Kiltz and Vahlis [37, 41] combines our direct chosen-ciphertext security techniques with the HIBE scheme from [10] to reduce the ciphertext size of the IBE scheme. The disadvantage is a slightly stronger security assumption. Similar results were obtained by Chatterjee and Sarkar [19, 52], who also propose a HIBE scheme from the BDDH assumption which is related to our proposal in Section 5.

Concurrently to the preparation of this article, Bellare and Ristenpart [4] present a new security analysis of Waters' IBE that completely avoids the artificial abort and therefore implies a tighter security reduction to the BDDH assumption.

1.3 Publication info

An extended abstract of this paper was published in the proceedings of ACISP 2006 [40]. This is the full version, containing improved concrete security bounds, missing proofs as well as a detailed comparison of our scheme with previous IB-KEM constructions.

2 Definitions

2.1 Notation

If x is a string, then $|x|$ denotes its length, while if S is a set then $|S|$ denotes its size. If $k \in \mathbb{N}$ then 1^k denotes the string of k ones. If S is a set then $s \xleftarrow{\$} S$ denotes the operation of picking an element s of S uniformly at random. We write $\mathcal{A}(x, y, \ldots)$ to indicate that $\mathcal{A}$ is an algorithm with inputs $x, y, \ldots$ and by $z \xleftarrow{\$} \mathcal{A}(x, y, \ldots)$ we denote the operation of running $\mathcal{A}$ with inputs $(x, y, \ldots)$ and letting z be the output. We write $\mathcal{A}^{\mathcal{O}_1, \mathcal{O}_2, \cdots}(x, y, \ldots)$ to indicate that $\mathcal{A}$ is an algorithm with inputs $x, y, \ldots$ and access to oracles $\mathcal{O}_1, \mathcal{O}_2, \ldots$ and by $z \xleftarrow{\$} \mathcal{A}^{\mathcal{O}_1, \mathcal{O}_2, \cdots}(x, y, \ldots)$ we denote the operation of running $\mathcal{A}$ with inputs $(x, y, \ldots)$ and access to oracles $\mathcal{O}_1, \mathcal{O}_2, \ldots$, and letting z be the output.

2.2 Identity Based Key Encapsulation

An *identity-based key-encapsulation mechanism* (IB-KEM) scheme [54, 13] $\mathcal{IBKEM} = (\mathsf{Kg}, \mathsf{Extract}, \mathsf{Encaps}, \mathsf{Decaps})$ consists of four polynomial-time algorithms. Via $(pk, sk) \xleftarrow{\$} \mathsf{Kg}(1^k)$ the randomized key-generation algorithm produces master keys for security parameter $k \in \mathbb{N}$; via $usk[id] \xleftarrow{\$} \mathsf{Extract}(sk, id)$ the master computes the secret key for identity id; via $(C, K) \xleftarrow{\$} \mathsf{Encaps}(pk, id)$ a sender creates a random session key K and a corresponding ciphertext C with respect to identity id; via $K \leftarrow \mathsf{Decaps}(pk, id, usk[id], C)$ the possessor of secret key sk decapsulates ciphertext C to get back a session key K. Associated to the scheme is a key space $\mathcal{K}$. For consistency, we require that for all $k \in \mathbb{N}$, all identities id, and all $(C, K) \xleftarrow{\$} \mathsf{Encaps}(pk, id)$, we have $\Pr[\mathsf{Decaps}(pk, id, \mathsf{Extract}(sk, id), C) = K] = 1$, where the probability is taken over the choice of $(pk, sk) \xleftarrow{\$} \mathsf{Kg}(1^k)$, and the coins of all the algorithms in the expression above.

The strongest and commonly accepted notion of security for an indentity-based key encapsulation scheme is that of *indistinguishability against an adaptive chosen ciphertext attack.* This notion, denoted IND-CCA, is defined using the following game between a challenger and an adversary $\mathcal{A}$. Let $\mathcal{IBKEM} = (\mathsf{Kg}, \mathsf{Extract}, \mathsf{Encaps}, \mathsf{Decaps})$ be an IB-KEM with associated key space $\mathcal{K}$.

The security we require the IBKEM is IND-CCA security [48]. For an adversary $\mathcal{A}$ we define the advantage function

$$\mathbf{Adv}^{ibkem\text{-cca}}_{\mathcal{IBKEM},\mathcal{A}}(k) \;=\; \left|\Pr[\mathbf{Exp}^{ibkem\text{-cca-}1}_{\mathcal{IBKEM},\mathcal{A}}(k) = 1] - \Pr[\mathbf{Exp}^{ibkem\text{-cca-}0}_{\mathcal{IBKEM},\mathcal{A}}(k) = 1]\right|$$

where, for $\gamma \in \{0, 1\}$, $\mathbf{Exp}^{ibkem\text{-cca-}\gamma}_{\mathcal{IBKEM},\mathcal{A}}$ is defined by the following experiment.

```
Experiment $\mathbf{Exp}^{ibkem\text{-cca-}\gamma}_{\mathcal{IBKEM},\mathcal{A}}(k)$
  $(pk, sk) \xleftarrow{\$} \mathsf{Kg}(1^k)$
  $(id^*, St) \xleftarrow{\$} \mathcal{A}^{\textsc{Ex}(\cdot),\textsc{Dec}(\cdot,\cdot)}(\mathit{find}, pk)$
  $K_0^* \xleftarrow{\$} \mathcal{K}$ ; $(C^*, K_1^*) \xleftarrow{\$} \mathsf{Encaps}(pk, id^*)$
  $\gamma' \xleftarrow{\$} \mathcal{A}^{\textsc{Ex}(\cdot),\textsc{Dec}(\cdot,\cdot)}(\mathit{guess}, K_\gamma^*, C^*, St)$
  Return $\gamma'$
```

The oracle $\textsc{Ex}(id)$ returns $usk[id] \xleftarrow{\$} \mathsf{Extract}(sk, id)$ with the restriction that $\mathcal{A}$ is not allowed to query oracle $\textsc{Ex}(\cdot)$ for the target identity id^*. The oracle $\textsc{Dec}(id, C)$ first computes $usk[id] \xleftarrow{\$} \textsc{Ex}(sk, id)$ and then returns $K \leftarrow \mathsf{Decaps}(pk, id, usk[id], C)$ with the restriction that in the guess stage $\mathcal{A}$ is not allowed to query oracle $\textsc{Dec}(\cdot, \cdot)$ for the tuple (id^*, C^*). Here the output of $\textsc{Ex}(id)$ is stored internally by the experiment and multiple queries to $\textsc{Dec}(id, \cdot)$ are answered with respect to the *same* user secret key $usk[id]$. The variable St represents some internal state information of adversary $\mathcal{A}$ and can be any polynomially-bounded string.

$\mathbf{Exp}^{ibkem\text{-cca-}1}_{\mathcal{IBKEM},\mathcal{A}}$ is called the *real CCA experiment* (with the real challenge key K_1^*), and $\mathbf{Exp}^{ibkem\text{-cca-}0}_{\mathcal{IBKEM},\mathcal{A}}$ is called the *random CCA experiment* (with a random challenge key K_0^*).

An IB-KEM $\mathcal{IBKEM}$ is said to be *secure against chosen-ciphertext attacks* (CCA secure) if the advantage functions $\mathbf{Adv}^{ibkem\text{-cca}}_{\mathcal{IBKEM},\mathcal{A}}(k)$ is a negligible function in k for all polynomial-time adversaries $\mathcal{A}$.

We remark that our security definition is given with respect to "full-identity" attacks, as opposed to the much weaker variant of "selective-identity" attacks where the adversary has to commit to its target identity id^* in advance, even before seeing the public key.

2.3 Target Collision Resistant Hash Functions

$\mathcal{TCR} = (\mathsf{TCR}_k)_{k\in\mathbb{N}}$ is a family of keyed hash function $\mathsf{TCR}_k^s : \mathbb{G} \to \mathbb{Z}_p$ for each $\ell(k)$-bit key s, where $\ell(\cdot)$ is a non-negative integer-valued polynomially-bounded function and p is a prime with polynomially-bounded bit-length. It is assumed target collision resistant (TCR) [21], which is captured by defining the tcr-advantage of an adversary $\mathcal{H}$ as

$$\mathbf{Adv}^{\mathrm{tcr}}_{\mathsf{TCR},\mathcal{H}}(k) = \Pr[\mathsf{TCR}^s(c^*) = \mathsf{TCR}^s(c) \wedge c \neq c^* \ : \ s \xleftarrow{\$} \{0,1\}^{\ell(k)} \ ; \ c^* \xleftarrow{\$} \mathbb{G} \ ; \ c \xleftarrow{\$} \mathcal{H}(s, c^*)]$$

Note TCR is a weaker requirement than collision-resistance, so that, in particular, any practical collision-resistant function can be used. Also note that our notion of TCR is related to the stronger notion of universal one-way hashing [46], where in the security experiment of the latter the target value c^* is chosen by the adversary (but before seeing the hash key s).

Commonly [21, 42] this function is implemented using a dedicated cryptographic hash function like MD5 or SHA, which is assumed to be target collision resistant. Alternatively, target collision resistant hashing can be constructed from any one-way function [46, 49]. However, these generic constructions are somewhat inefficient. Since in our case $|\mathbb{G}| = |\mathbb{Z}_p| = p$, we can alternatively also use a non-keyed bijective encoding function $\mathsf{TCR}^* : \mathbb{G} \to \mathbb{Z}_p$. In that case we have a perfect collision resistant hash function, i.e. $\mathbf{Adv}^{\mathrm{tcr}}_{\mathsf{TCR}^*,\mathcal{H}}(k) = 0$. Boyen, Mei and Waters [15] note that for bilinear maps defined on supersingular elliptic curves there exists a very efficient way to implement such injective mappings. We refer to [15] for more details.

3 Assumptions

3.1 Parameter generation algorithms for Bilinear Groups.

All pairing-based schemes will be parameterized by a *pairing parameter generator.* This is a randomized polynomial-time algorithm $\mathcal{G}$ that on input 1^k returns the description of an multiplicative cyclic group $\mathbb{G}_1$ of prime order p, where $2^{2k} < p$, the description of a multiplicative cyclic group $\mathbb{G}_T$ of the same order, and a non-degenerate bilinear pairing $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_T$. See [13] for a description of the properties of such pairings. We use $\mathbb{G}_1^*$ to denote $\mathbb{G}_1 \setminus \{0\}$, i.e. the set of all group elements except the neutral element. Throughout the paper we use $\mathbb{PG} = (\mathbb{G}_1, \mathbb{G}_T, p, \hat{e})$ as shorthand for the description of bilinear groups.

3.2 The BDDH assumption

Let $\mathbb{PG}$ be the description of a pairing group. Consider the following problem first put forward by Joux [36] and later formalized by Boneh and Franklin [13]: Given $(g, g^a, g^b, g^c, W) \in \mathbb{G}_1^4 \times \mathbb{G}_T$ as input, output yes if $W = \hat{e}(g,g)^{abc}$ and no otherwise. More formally, to a parameter generation algorithm for pairing-groups $\mathcal{G}$ and an adversary $\mathcal{B}$ we define the following advantage function

$$\mathbf{Adv}^{\mathrm{bddh}}_{\mathcal{G},\mathcal{B}}(k) = \left|\Pr[\mathcal{B}(g, g^a, g^b, g^c, W) = 1] - \Pr[\mathcal{B}(g, g^a, g^b, g^c, \hat{e}(g,g)^{abc}) = 1]\right|,$$

where $g, W \xleftarrow{\$} \mathbb{G}$ and $a, b, c, r \leftarrow \mathbb{Z}_p$.

We say that the *Bilinear Decision Diffie-Hellman (BDDH) assumption relative to generator* $\mathcal{G}$ holds if $\mathbf{Adv}^{\mathrm{bddh}}_{\mathcal{G},\mathcal{B}}$ is a negligible function in k for all polynomial time adversaries $\mathcal{B}$. The BDDH assumption was shown to hold in the generic group model in [10] and can be shown to be random self reducible by using similar techniques to those in [44].

4 A chosen-ciphertext secure IB-KEM based on BDDH

In this section we present our new chosen-ciphertext secure IB-KEM. From now on let $\mathbb{PG} = (\mathbb{G}_1, \mathbb{G}_T, p, \hat{e}, g)$ be public system parameters obtained by running the group parameter algorithm $\mathcal{G}(1^k)$.

```
$\mathsf{Kg}(1^k)$
    $u_1, u_2, \alpha \xleftarrow{\$} \mathbb{G}_1$ ; $z \leftarrow \hat{e}(g, \alpha)$
    $\mathsf{H} \xleftarrow{\$} \mathsf{HGen}(n)$
    $pk \leftarrow (u_1, u_2, z, \mathsf{H})$ ; $sk \leftarrow \alpha$
    Return $(pk, sk)$

$\mathsf{Extract}(pk, sk, id)$
    $s \xleftarrow{\$} \mathbb{Z}_p$
    $usk[id] \leftarrow (\alpha \cdot \mathsf{H}(id)^s, g^s) \in \mathbb{G}_1^2$
    Return $usk[id]$

$\mathsf{Encaps}(pk, id)$
    $r \xleftarrow{\$} \mathbb{Z}_p^*$
    $c_1 \leftarrow g^r$ ; $t \leftarrow \mathsf{TCR}(c_1)$
    $c_2 \leftarrow \mathsf{H}(id)^r$
    $c_3 \leftarrow (u_1^t u_2)^r$
    $K \leftarrow z^r \in \mathbb{G}_T$
    $C \leftarrow (c_1, c_2, c_3) \in \mathbb{G}_1^3$
    Return $(K, C)$

$\mathsf{Decaps}(pk, id, usk[id], C)$
    Parse $C$ as $(c_1, c_2, c_3)$
    Parse $usk[id]$ as $(d_1, d_2)$
    $t \leftarrow \mathsf{TCR}(c_1)$
    If $(g, c_1, u_1^t u_2, c_3)$ is not a DH tuple
    or $(g, c_1, \mathsf{H}(id), c_2)$ is not a DH tuple
        then return $K \xleftarrow{\$} \mathbb{G}_T$
    else return $K \leftarrow \hat{e}(c_1, d_1)/\hat{e}(c_2, d_2)$
```

Figure 1: Our chosen-ciphertext secure IB-KEM $\mathcal{IBKEM} = (\mathsf{Kg}, \mathsf{Extract}, \mathsf{Encaps}, \mathsf{Decaps})$.

4.1 Waters' Hash

We review the hash function $\mathsf{H} : \{0,1\}^n \to \mathbb{G}_1$ used in Waters' identity based encryption scheme [58]. On input of an integer n, the randomized hash key generator $\mathsf{HGen}(n)$ chooses $n+1$ random groups elements $h_0, \ldots, h_n \in \mathbb{G}_1$ and returns $h = (h_0, h_1, \ldots, h_n)$ as the public description of the hash function. The hash function $\mathsf{H} : \{0,1\}^n \to \mathbb{G}_1^*$ is evaluated on a string $id = (id_1, \ldots, id_n) \in \{0,1\}^n$ as the product

$$\mathsf{H}(id) = h_0 \prod_{i=1}^{n} h_i^{id_i} \ .$$

4.2 The IB-KEM Construction

Let $\mathsf{TCR} : \mathbb{G}_1 \to \mathbb{Z}_p$ be a target collision resistant hash function. Our IB-KEM with identity space $\mathit{IDSp} = \{0,1\}^n$ ($n = n(k)$) and key space $\mathcal{K} = \mathbb{G}_T$ is depicted in Figure 1. For simplicity we assume that TCR is a fixed hash function such as an injective encoding or SHA-1. Otherwise, if TCR is a keyed TCR function, a random key s has to be included in the scheme's public key.

A tuple $(h, h^a, h^b, h^c) \in \mathbb{G}_1^4$ is said to be a *Diffie-Hellman tuple* if $ab = c \bmod p$. Thanks to the properties of the bilinear pairing, a tuple $(g, u, v, w) \in \mathbb{G}_1^4$ is Diffie-Hellman if and only if $\hat{e}(g, w) = \hat{e}(v, u)$. Therefore the check in the decapsulation algorithm Decaps can be implemented by evaluating the bilinear map four times.

We now show correctness of the scheme, i.e., that the session key K computed in the encapsulation algorithm matches the K computed in the decapsulation algorithm. A correctly generated ciphertext for identity id has the form $C = (c_1, c_2, c_3) = (g^r, \mathsf{H}(id)^r, (u_1^t u_2)^r)$ and therefore $(g, c_1, u_1^t u_2, c_3) = (g, g^r, u_1^t u_2, (u_1^t u_2)^r)$ is always a DH tuple. A correctly generated secret key for identity id has the form $usk[id] = (d_1, d_2) = (\alpha \cdot \mathsf{H}(id)^s, g^s)$. Therefore the decapsulation algorithm computes the session key K as

$$\begin{aligned} K &= \hat{e}(c_1, d_1)/\hat{e}(c_2, d_2) \\ &= \hat{e}(g^r, \alpha \cdot \mathsf{H}(id)^s)/\hat{e}(\mathsf{H}(id)^r, g^s) \\ &= \hat{e}(g^r, \alpha) \cdot \hat{e}(g^r, \mathsf{H}(id)^s)/\hat{e}(\mathsf{H}(id)^r, g^s) \\ &= z^r \cdot \hat{e}(g^s, \mathsf{H}(id)^r)/\hat{e}(\mathsf{H}(id)^r, g^s) \\ &= z^r, \end{aligned}$$

as the key computed in the encapsulation algorithm. This shows correctness.

Let $C = (c_1, c_2, c_3) \in \mathbb{G}_1^3$ be a (possibly malformed) ciphertext. Ciphertext C is called *consistent* w.r.t the public key pk and identity id if $(g, c_1, u_1^t u_2, c_3)$ and $(g, c_1, \mathsf{H}(id), c_2)$ are Diffie-Hellman tuples, where $t = \mathsf{TCR}(c_1)$. Note that any ciphertext properly generated by the encapsulation algorithm is always consistent. The decapsulation algorithm tests for consistency of the ciphertext. Note that this consistency test can be performed by anybody knowing the public-key. We call this property "public verifiability" of the ciphertext. It is the key feature that allows building an efficient IB-KEM with non-interactive threshold decryption as it was proposed in [25].

4.3 More Efficient Decapsulation from Implicit Rejection

We now describe an alternative decapsulation algorithm which is more efficient but slightly more technically involved. The idea is to make the Diffie-Hellman consistency check implicit in the computation of the key K and it has already been used in [38]. This is done by choosing random integers $r_1, r_2 \in \mathbb{Z}_p^*$ and computing the session key as

$$K \leftarrow \frac{\hat{e}(c_1, d_1 \cdot (u_1^t u_2)^{r_1} \cdot H(id)^{r_2})}{\hat{e}(c_2, d_2 \cdot g^{r_2}) \cdot \hat{e}(g^{r_1}, c_3)} .$$

We claim that this is equivalent to first checking for consistency and returning a random key if not, and otherwise computing the key as $K \leftarrow \hat{e}(c_1, d_1)/\hat{e}(c_2, d_2)$ as in the original decapsulation algorithm.

To prove this claim we define the functions $\Delta_1(C) = \hat{e}(c_1, u_1^t u_2)/\hat{e}(g, c_3)$ and $\Delta_2(C) = \hat{e}(\mathsf{H}(id), c_1)/\hat{e}(g, c_2)$. Then $\Delta_1(C) = \Delta_2(C) = 1$ if and only if C is consistent. Consequently, for random $r_1, r_2 \in \mathbb{Z}_p^*$, $K = \hat{e}(c_1, d_1)/\hat{e}(c_2, d_2) \cdot (\Delta_1(C))^{r_1} \cdot (\Delta_2(C))^{r_2} \in \mathbb{G}_T^*$ evaluates to $\hat{e}(c_1, d_1)/\hat{e}(c_2, d_2) \in \mathbb{G}_T$ if C is consistent and to a random group element otherwise. The claim then follows by

$$\begin{aligned} K &= \hat{e}(c_1, d_1)/\hat{e}(c_2, d_2) \cdot \Delta_1(C)^{r_1} \cdot (\Delta_2(C))^{r_2} \\ &= \hat{e}(c_1, d_1)/\hat{e}(c_2, d_2) \cdot (\hat{e}(c_1, u_1^t u_2)/\hat{e}(g, c_3))^{r_1} \cdot (\hat{e}(\mathsf{H}(id), c_1)/\hat{e}(g, c_2))^{r_2} \\ &= \frac{\hat{e}(c_1, d_1 (u_1^t u_2)^{r_1} H(id)^{r_2})}{\hat{e}(c_2, d_2 \cdot g^{r_2}) \cdot \hat{e}(g^{r_1}, c_3)} . \end{aligned}$$

We remark that the alternative decapsulation algorithm saves four pairing operations in a naïve implementation (at the cost of four exponentiations).

4.4 Relation to existing schemes

Relation to Waters' IBE scheme. The ciphertext in our scheme is basically identical to the ciphertext from Waters' IBE scheme plus the redundant element c_3 used to check for consistency of the ciphertext. Hence Waters' IBE scheme is obtained by ignoring the computation of c_3 in encapsulation as well as the consistency check in decapsulation.

Relation to the Encryption Scheme from BMW. Clearly, IB-KEM implies traditional public-key encapsulation by simply ignoring all operations related to the identity. We remark that viewed in this light (i.e., ignoring the element c_2 in encapsulation/decapsulation and ignoring the key derivation algorithm) our IB-KEM is simplified to the chosen-ciphertext secure encryption scheme recently proposed in [15, 38].

4.5 Security

Theorem 4.1 Assume $\mathcal{TCR}$ is a family of target collision resistant hash functions. Under the Bilinear Decisional Diffie-Hellman (BDDH) assumption relative to generator $\mathcal{G}$, the IB-KEM from Section 4.2 is secure against chosen-ciphertext attacks.

In particular, given an adversary $\mathcal{A}$ attacking the chosen-ciphertext security of the IB-KEM with advantage $\varepsilon_{\mathcal{A}}(k) = \mathbf{Adv}^{ibkem\text{-}\mathrm{cca}}_{\mathcal{IBKEM},\mathcal{A}}(k)$ and running time $\mathbf{T}_{\mathcal{A}}(k)$, we construct an adversary $\mathcal{B}$ breaking the

BDDH assumption with advantage $\varepsilon_{\mathcal{B}}(k) = \mathbf{Adv}^{\mathrm{bddh}}_{\mathcal{G},\mathcal{B}}(k)$ and running time $\mathbf{T}_{\mathcal{B}}(k)$, and an adversary $\mathcal{H}$ breaking $\mathcal{TCR}$ with advantage $\varepsilon_{\mathcal{H}}(k) = \mathbf{Adv}^{\mathrm{tcr}}_{\mathsf{TCR},\mathcal{H}}(k)$ and running time $\mathbf{T}_{\mathcal{H}}(k) \approx \mathbf{T}_{\mathcal{A}}(k)$ with

$$\begin{aligned} \varepsilon_{\mathcal{B}}(k) &\geq \frac{\varepsilon_{\mathcal{A}}(k) - \varepsilon_{\mathcal{H}}(k)}{10nq} - \frac{q}{p}; \\ \mathbf{T}_{\mathcal{B}}(k) &\leq \mathbf{T}_{\mathcal{A}} + \mathcal{O}(n^3 \cdot \varepsilon_{\mathcal{A}}^{-2}(k) \cdot \ln((nq\varepsilon_{\mathcal{A}}(k))^{-1}) \cdot \mathbf{T}_{\mathbb{Z}} + q \cdot \mathbf{T}_{\mathbb{PG}}(k)), \end{aligned}$$

where $q < p/(2(n+1))$ is an upper bound on the number of key derivation/decryption queries made by adversary $\mathcal{A}$, $\mathbf{T}_{\mathbb{PG}}$ is the time for one exponentiation/pairing computation in $\mathbb{PG}$, and $\mathbf{T}_{\mathbb{Z}}$ is the time for one addition over integers smaller than $2q$.

The proof of Theorem 4.1 is deferred to Section 6.

5 Extensions

5.1 Chosen-ciphertext secure Hierarchical Identity-Based Key Encapsulation

Hierarchical identity-based key encapsulation (HIB-KEM) is a generalization of IB-KEM to identities supporting hierarchical structures [32, 27]. By the relation to Waters HIBE scheme it is easy to see that our technique can also be used to make chosen-ciphertext secure the KEM variant of Waters HIBE. To be more precise, we modify Waters' HIB-KEM and add one more element $(u_1^t u_2^\lambda u_3)^r$ to the the ciphertext, where t was computed by applying a target-collision hash function to g^r (here r is the randomness used to create the ciphertext). The additional element is used for a consistency check at decryption, with the novelty that the hierarchy's depth λ is encoded via u_2^λ. The security reduction is exponential in the depth λ of the hierarchy, i.e. it introduces, roughly, a multiplicative factor of $(nq)^\lambda$. Hence the scheme can only be securely instantiated for small hierarchies, say $\lambda \leq 4$.

More precisely, the new HIB-KEM setup algorithm chooses d different and independent hash functions $\mathsf{H}_j \overset{\$}{\leftarrow} \mathsf{HGen}(\mathbb{G}_1)$ for $1 \leq j \leq d$ and $u_1, u_2, u_3 \leftarrow \mathbb{G}_1$. The private key for the identity $\overrightarrow{id} = (id_1, \ldots, id_\lambda)$ of depth $1 \leq \lambda \leq d$ is defined as $usk[\overrightarrow{id}] = (d_0, d_1, \ldots, d_\lambda)$, where $d_j = g^{r_j}$ and $r_j \overset{\$}{\leftarrow} \mathbb{Z}_p$ for $1 \leq i \leq \lambda$, while $d_0 = \alpha \cdot (\prod_{j=1}^{\lambda} \mathsf{H}_j(id_j)^{r_j})$. Encapsulation with respect to $\overrightarrow{id}$ is defined as $C = (c_0, \ldots, c_\lambda, c_{\lambda+1})$, where $c_0 = g^r$ and $c_j = \mathsf{H}_j(id_j)^r$ for $r \overset{\$}{\leftarrow} \mathbb{Z}_p$, $1 \leq j \leq \lambda$, while $c_{\lambda+1} = (u_1^t u_2^\lambda u_3)^r$, with $t = \mathsf{TCR}(g^r)$. Finally, decapsulation K of a ciphertext $C = (c_0, c_1, \ldots, c_\lambda, c_{\lambda+1})$ with respect to $\overrightarrow{id}$ is obtained by choosing $s_0, \ldots, s_\lambda \overset{\$}{\leftarrow} \mathbb{Z}_p$ and computing

$$K = \frac{\hat{e}\big(c_0, d_0 \cdot (u_1^t u_2^\lambda u_3)^{s_0} \cdot \prod_{j=1}^{\lambda} \mathsf{H}_j(id_j)^{s_j}\big)}{\hat{e}(g^{s_0}, c_{\lambda+1}) \cdot \prod_{j=1}^{\lambda} \hat{e}(c_j, d_j \cdot g^{s_j})} .$$

Here we used our "implicit rejection" technique from Section 4.3 to decrease the number of pairings needed for decapsulation to $\lambda + 2$.

We note that the HIB-KEM construction mentioned in the proceedings version of this paper [40] was incorrect and hence not secure against chosen-ciphertext attacks. This was also independently discovered and fixed in [51], where details of the proof were worked out.

5.2 Identity-based Encryption

Given a IB-KEM and a symmetric encryption scheme, a hybrid identity-based encryption scheme can be obtained by using the IB-KEM to securely transport a random session key that is fed into the symmetric encryption scheme to encrypt the plaintext message. It is known that if both the IB-KEM and the symmetric encryption scheme are chosen-ciphertext secure, then the resulting hybrid encryption is also chosen-ciphertext secure [21, 7]. The security reduction is tight.

Using the "encrypt-then-mac" paradigm [3], a symmetric encryption scheme secure against chosen-ciphertext attacks can be built from relatively weak primitives, i.e. from any one-time symmetric encryption scheme, such as the one-time pad [57], by adding a message authentication code (MAC). Furthermore, Phan and Pointcheval [47] showed that *super pseudorandomn permutations* directly imply redundancy-free chosen-ciphertext secure symmetric encryption that avoid the use of the MAC. Such strong pseudorandom permutations can in turn be generated by applying a 2-round Feistel network to a pseudorandom function (and furthermore two pairwise independent permutations) [45]. However, it practice it seems reasonable to assume that modern block-ciphers such as AES are already strong pseudorandom permutations. Provided that the underlying block-cipher is a strong pseudorandom permutation, the modes of operation CMC [29], EME [30], and EME* [28] can be used to encrypt large messages. Hence a chosen-ciphertext secure IBE scheme can be built from our IB-KEM construction without any additional overhead: the ciphertext overhead of our IBE scheme, that is the difference between ciphertext and message size, is the asymmetric IB-KEM part, i.e. three group elements.

We note that for the natural task of securely generating a joint random session key, a IB-KEM is sufficient and a fully-fledged identity-based encryption scheme is not needed.

5.3 A Tradeoff between public key size and security reduction

As independently discovered in [18, 43], there exists an interesting trade-off between key-size of Waters' hash H and the security reduction of the IBE scheme.

The construction modifies Waters hash H as follows: Let the integer $l = l(k)$ be a new parameter of the scheme. In particular, we represent an identity $id \in \{0,1\}^n$ as an n/l-dimensional vector $id = (id_1, \ldots, id_{n/l})$, where each id_i is an l bit string. Waters hash is then redefined to $\mathsf{H} : \{0,1\}^n \to \mathbb{G}_1$, with $\mathsf{H}(id) = h_0 \prod_{i=1}^{n/l} h_i^{id_i}$ for random public elements $h_0, h_1, \ldots, h_{n/l} \in \mathbb{G}_1$. Waters' original hash function is obtained as the special case $l = 1$. It is easy to see that using this modification in our IBE scheme (i) reduces the size of the public key from $n + 4$ to $n/l + 4$ group elements, whereas (ii) it adds another multiplicative factor of 2^l to the security reduction of the IBE scheme (Theorem 4.1).[1]

5.4 Selective-identity chosen-ciphertext secure IB-KEM

For the definition of a selective-identity chosen-ciphertext secure IB-KEM we change the security experiment such that the adversary has to commit to the target idenity id^* before seeing the public key. Clearly, this is a weaker security requirement. We quickly note that (using an algebraic technique from [8]) by replacing Waters' hash H with $\mathsf{H}(id) = h_0 \cdot h_1^{id}$ (for $id \in \mathbb{Z}_p$) we get a selective-id chosen-ciphertext secure IB-KEM. Note that the size of the public-key of this scheme drops to 3 elements.

6 Security analysis

We give a game-based proof of Theorem 4.1. Our proof is mainly based on the one given by Waters [58], where we make some important modifications to be able to deal with chosen-ciphertext attacks. Moreover, we are able to substantially improve the bound on the running time of the BDDH adversary $\mathcal{B}$ compared to [58].

Intuitively, security can be best understood by observing that our scheme is a generalization of Waters' IBE scheme, as well as of the chosen-ciphertext secure *public-key* encapsulation scheme from [15, 38].

Before we give the proof we recall the "Difference Lemma" [56].

Lemma 6.1 Let X_1,X_2, B be events defined in some probability distribution, and suppose that $X_1 \wedge \neg B \Leftrightarrow X_2 \wedge \neg B$. Then $|\Pr[X_1] - \Pr[X_2]| \leq \Pr[B]$.

6.1 Proof of Theorem 4.1

Let $\mathcal{A}$ be an adversary on the CCA security of the IB-KEM. We will consider a sequence of games, Game 0, Game 2, ..., Game 10, each game involving $\mathcal{A}$. At the end of each game there is a well-defined output bit $\beta' \in \{0,1\}$. Let X_i be the event that in Game i, it holds that $\beta' = 1$.

Game 0. (Real CCA experiment) Let Game 0 be IB-KEM security experiment of Section 2.2 with $\gamma = 1$ (the real CCA experiment). While describing the experiment we will make a couple of conventions on how

[1] On the technical side our proof basically stays the same, only the bound from Lemma 6.2 needs to be adapted to take the modified Waters' hash into account.

the experiment chooses the values appearing in the game. These conventions will be purely conceptual and, compared to the original experiment, do not change the distribution of any value appearing during the experiment. We will also make a couple of definitions of values appearing during the experiment.

We assume that in the beginning the experiment chooses some values a, b, and c, uniformly distributed over $\mathbb{Z}_p$. The experiment will depend on these values (i.e., the key generation will depend on g^a, g^b, user secret key generation on g^{ab} and the challenge ciphertext will depend on c). In sequel games the experiment will "forget" the values g^{ab}, and c and instead only use the values g^a, g^b, and g^c. The dependencies of the different experiment phases "Challenge", "Extract", and "Decaps" on those values in Games 0-10 are depicted in the following table.

Game	0	1	2	3	4	5	6	7	8	9	10
Challenge	c	c	c	c	c	c	c	c	c	$\hat{e}(g,g)^{abc}$	—
Extract	g^{ab}	g^{ab}	g^{ab}	g^{ab}	g^{ab}	—	—	—	—	—	—
Decaps	g^{ab}	g^{ab}	g^{ab}	g^{ab}	g^{ab}	g^{ab}	g^{ab}	g^{ab}	—	—	—

Key Generation. Initially the experiment generates public-key $pk = (u_1, u_2, z, \mathsf{H})$ and secret-key $sk = \alpha$ using the IB-KEM key generation algorithm $\mathsf{Kg}(1^k)$. We make the convention that the public key is generated as

$$u_1 \leftarrow g^a, \quad u_2 \xleftarrow{\$} \mathbb{G}_1, \quad z \leftarrow \hat{e}(g^a, g^b), \quad h_0 \xleftarrow{\$} \mathbb{G}_1, \ldots, h_n \xleftarrow{\$} \mathbb{G}_1 \ , \qquad (1)$$

depending on the elements a, b. Note that the way the value $z = \hat{e}(g^a, g^b) = \hat{e}(g, g^{ab})$ from the public key is generated implies $\alpha = g^{ab}$. The public key is given to the adversary $\mathcal{A}$ to start its *find* phase.

Find Phase. During its execution adversary $\mathcal{A}$ makes a number of key derivation and decapsulation requests. If the adversary makes a key derivation query $\textsc{Ex}(id)$ then the experiment computes the secret key $sk[id]$ by using the master secret key α, and returns $sk[id]$ to the adversary. If the adversary makes a decapsulation query $\textsc{Dec}(id, C)$ the experiment (using α) decrypts the ciphertext and returns the corresponding key to the adversary.

Eventually, the adversary returns a target identity id^*. The experiment runs the encapsulation algorithm to create a real challenge key K_1^* together with the the challenge ciphertext $C^* = (c_1^*, c_2^*, c_3^*)$. We make the convention that challenge key/ciphertext are computed using randomness $c \in \mathbb{Z}_p$ as follows

$$c_1^* \leftarrow g^c, \quad c_2^* \leftarrow \mathsf{H}(id^*)^c, \quad c_3^* \leftarrow (u_1^{t^*} u_2)^c, \quad K_1^* \leftarrow z^c \ , \qquad (2)$$

where $t^* \leftarrow \mathsf{TCR}(c_1^*)$.

The experiment returns the challenge ciphertext C^* together with the real key K_1^* to adversary $\mathcal{A}$.

Guess Phase. The adversary continues to make its oracle queries, subsequent key derivation requests must be different from the target identity id^* and decapsulation requests must be different from (id^*, C^*). Finally, adversary $\mathcal{A}$ returns a bit $\gamma' \in \{0, 1\}$ and the experiment returns $\beta' = \gamma'$.

Note that the experiment behaves exactly as in the original real CCA IB-KEM security experiment with $\gamma = 1$, i.e., we have

$$\Pr[X_0] = \Pr[\mathbf{Exp}^{ibkem\text{-cca-}1}_{\mathcal{IBKEM},\mathcal{A}} = 1] \ .$$

Now a few important definitions are in place. During its execution $\mathcal{A}$ may query the key derivation oracle for some identity id or the decapsulation oracle for the identity/ciphertext pair (id, C). We collect all those identities used to make queries to the key derivation and decapsulation oracle in the set $\widetilde{ID}$. Note that $\widetilde{ID}$ may contain the target identity id^* or one identity more than once. Let ID be the subset of queried identities obtained by removing from $\widetilde{ID}$ all multiples and the target identity. We write $ID = \{id^{(1)}, \ldots, id^{(q_0)}\}$ (without any particular order) for some $q_0 \leq q$ such that $id^{(i)} \neq id^{(j)}$ for each $1 \leq i \neq j \leq q_0$ and $id^* \notin ID$. Furthermore, we define $ID^* = ID \cup \{id^*\} = \{id^{(1)}, \ldots, id^{(q_0)}, id^*\}$.

Game 1. (Eliminate hash collisions) Note that the values $c_1^* = C$ and $t^* = \mathsf{TCR}(C)$ from the challenge ciphertext Equation (2) are completely independent of the view of adversary $\mathcal{A}$ until $\mathcal{A}$ is in *guess* phase. This holds since C is simply not touched by the experiment before generating the challenge ciphertext. Therefore we may assume that the value c_1^* is already generated by the experiment before the key generation (and then before the seed s for $\mathsf{TCR} = \mathsf{TCR}_s$ is chosen).

In this game the experiment changes its answers to all decapsulation queries $\textsc{Dec}(id, C)$ made by $\mathcal{A}$ as follows: Let $C = (c_1, c_2, c_3)$ and $t = \mathsf{TCR}(c_1)$. If $t = t^*$ and $c_1 \neq c_1^*$, the experiment aborts and returns $\beta' = 1$. Otherwise it continues as in the last game. Let $\textsc{HashAbort}$ be the event that this new abortion rule applies. Until $\textsc{HashAbort}$ happens Game 0 and Game 1 are identical. Therefore by Lemma 6.1 we have

$$|\Pr[X_1] - \Pr[X_0]| \leq \Pr[\textsc{HashAbort}] .$$

Furthermore, there exists an adversary $\mathcal{H}$ against the target collision resistance of TCR running in time $\mathbf{T}_{\mathcal{H}}(k) \approx \mathbf{T}_{\mathcal{A}}(k)$ that succeeds with probability at least $\Pr[\textsc{HashAbort}]$, i.e.,

$$\Pr[\textsc{HashAbort}] \leq \mathbf{Adv}^{\mathrm{tcr}}_{\mathsf{TCR},\mathcal{H}}(k) .$$

This adversary $\mathcal{H}$ inputs a random $c_1^* = g^c$ and runs the real CCA experiment. Note that $\mathcal{H}$ can simulate the whole Game 0 depending only on $c_1^* = g^c$ by knowing a, b. Furthermore, $\mathcal{H}$ sets up the public-key such that it knows $\log_g(u_2)$ and $\log_g(h_i)$ and hence can create the challenge ciphertext from Equation (2). When $\textsc{HashAbort}$ happens, $\mathcal{H}$ simply outputs c_1 and terminates.

Game 2. (Change of public key) This is the same as Game 1 except that the experiment changes the generation of the public key pk from Equation (1) as follows.

Set $m = 2q$ (the choice of m will become clear later). Instead of generating the hash keys as in Equation (1) the experiment now chooses

$$\begin{aligned} x_0, x_1, \ldots, x_n &\overset{\$}{\leftarrow} \{0, \ldots, p-1\} \\ y_0', y_1, \ldots, y_n &\overset{\$}{\leftarrow} \{0, \ldots, m-1\} \\ \ell &\overset{\$}{\leftarrow} \{0, \ldots, n\} \qquad (3) \\ \text{and sets} & \\ y_0 &\leftarrow y_0' - \ell m . \end{aligned}$$

The public keys $h = (h_0, \ldots, h_n)$ of the hash function H are then defined as $h_i = g^{x_i} u_1^{y_i}$, for $0 \leq i \leq n$. By definition the public hash function evaluated in identity $id \in \{0,1\}^n$ is given as $\mathsf{H}(id) = h_0 \prod_{i=1}^n h_i^{id_i}$. From the experiments's point of view, however, the hash function evaluated in $id \in \{0,1\}^n$ looks like

$$\mathsf{H}(id) = g^{x(id)} u_1^{y(id)}, \qquad (4)$$

with $x(id) = x_0 + \sum_{i=1}^n id_i x_i$ and $y(id) = y_0 + \sum_{i=1}^n id_i y_i$ only known to the experiment. On the other hand, note that this change does not affect the distribution of the hash keys $h = (h_0, h_1, \ldots, h_n)$. Therefore we have

$$\Pr[X_2] = \Pr[X_1] .$$

Game 3. (Abort at the end of the game) Fix all the random variables adversary $\mathcal{A}$ gets to see during its execution, including its random coin tosses: fix pk, and the randomness used in answering the key derivation and decapsulation queries. Now adversary $\mathcal{A}$ can be seen as a deterministic algorithm, in particular the set of all queried (distinct) identities $ID^* = \{id^{(1)}, \ldots, id^{(q_0)}, id^*\}$ can be seen as fixed. By $view_{\mathcal{A}}$ we denote all these fixed variables.

Define $\mathbf{Y} = \{y_0', y_1, \ldots, y_n, \ell\}$, where the random variables $\{y_0', y_1, \ldots, y_n, \ell\}$ are distributed as in Equation (3). It is clear that once $view_{\mathcal{A}}$ is fixed, the random variable $\mathbf{Y}$ still has its original distribution (due to the random masks $x_i \in \mathbb{Z}_p$). Define the event

$$\textsc{ForcedAbort} : \bigvee_{i=1}^{q_0} \left(y(id^{(i)}) = 0 \bmod p \right) \vee y(id^*) \neq 0 \bmod p .$$

We call this abort *forced* since in sequel games the experiment is modified such that it always *has to* abort once this event happens. For fixed $view_{\mathcal{A}}$ we define

$$\eta(view_{\mathcal{A}}) := \Pr_{\mathbf{Y}}[\neg\textsc{ForcedAbort}] . \qquad (5)$$

The following lemma bounds $\eta(view_{\mathcal{A}})$.

Lemma 6.2 For every fixed $view_{\mathcal{A}}$, we have

$$\lambda_{\text{low}} := \frac{1}{4(n+1)q} \;\leq\; \eta(view_{\mathcal{A}}) \;\leq\; \frac{1}{2q} =: \lambda_{\text{up}} \; .$$

This lemma is as an extension of a lemma by Waters [58] who only proved the lower bound on $\eta(view_{\mathcal{A}})$. Its proof is quite technical and is postponed to Section 6.2.

Compared to Game 2 we will make two modifications to the experiment in Game 3. The experiment is exactly the same as in Game 2 until adversary $\mathcal{A}$ outputs his guess bit γ'. Since adversary $\mathcal{A}$ already terminated we can assume $view_{\mathcal{A}}$ to be fixed from now on.

First modification: add forced abort. After adversary $\mathcal{A}$ outputs his guess bit γ', the experiment checks if the event ForcedAbort occurs. If yes, the experiment returns $\beta' = 0$ and aborts. Otherwise, it continues as before, i.e., it returns $\beta' = \gamma'$.

Let us first make two unsuccessful attempts to meaningfully relate the two events X_3 and X_2. First, by the Difference Lemma (Lemma 6.1) we have that $|\Pr[X_3] - \Pr[X_2]| \leq \Pr[\text{ForcedAbort}]$ which is not meaningful since $\Pr[\text{ForcedAbort}] = 1 - \eta(view_{\mathcal{A}})$ is close to 1 (Lemma 6.2).

Second, since $\Pr[\beta' = 1 \mid \text{ForcedAbort}] = 0$ we have $\Pr[X_3] = \Pr[\beta' = 1] = \Pr[\gamma' = 1 \mid \neg\text{ForcedAbort}] \cdot \Pr[\neg\text{ForcedAbort}]$. Now we would like to continue with $\Pr[\gamma' = 1 \mid \neg\text{ForcedAbort}] = \Pr[\gamma' = 1] = \Pr[X_2]$. However, this is not correct since the experiment aborts with a probability $\eta(view_{\mathcal{A}})$ which is a function in $view_{\mathcal{A}}$, in particular in the choices of the identities $ID^* = \{id^{(1)}, \ldots, id^{(q_0)}, id^*\}$ queried by adversary $\mathcal{A}$. Hence the two events X_2 and $\neg$ForcedAbort cannot be considered as independent. In the worst case it may happen that $\Pr[\gamma' = 1 \mid \neg\text{ForcedAbort}] \approx 0$ even though $\Pr[\gamma' = 1]$ is non-negligible. Let $\lambda_{\text{low}} := \frac{1}{4(n+1)q}$ be the lower bound on the abortion probability computed in Lemma 6.2. To get rid of this unwanted dependence the experiment adds some *artificial abort* such that in total it *always* aborts with probability around $1 - \lambda_{\text{low}}$, independent of the view of the adversary $view_{\mathcal{A}}$.

Second modification: add artificial abort. After adversary $\mathcal{A}$ outputs his guess bit γ', the experiment checks if the event ForcedAbort occurs. If yes, the experiment returns $\beta' = 0$ and aborts. Otherwise, it continues as follows: first it samples (using sufficiently many samples) an estimate $\eta'(view_{\mathcal{A}})$ of the probability $\eta(view_{\mathcal{A}})$ (over $\mathbf{Y}$) that the event $\neg$ForcedAbort happens.[2] We want to stress that $view_{\mathcal{A}}$ is fixed at this point so sampling does not involve running adversary $\mathcal{A}$ again. By definition, this estimate $\eta'(view_{\mathcal{A}})$ is a random variable that only depends on the queried identities $id^{(1)}, \ldots, id^{(q_0)}, id^*$ (and the randomness used to sample).

Depending on the estimate $\eta'(view_{\mathcal{A}})$ the experiment distinguishes two cases:

Case $\eta'(view_{\mathcal{A}}) \leq \lambda_{\text{low}}$: the experiment does not abort and continues as before, outputting $\beta' = \gamma'$.

Case $\eta'(view_{\mathcal{A}}) > \lambda_{\text{low}}$: With probability $1 - \lambda_{\text{low}}/\eta'(view_{\mathcal{A}})$ the experiment aborts and outputs $\beta' = 0$. With probability $\lambda_{\text{low}}/\eta'(view_{\mathcal{A}})$ the experiment does not abort and continues as before, outputting $\beta' = \gamma'$.

This concludes the description of Game 3.

The following lemma relating the events X_2 and X_3 will be proved in Section 6.3. Compared to the corresponding lemma by Waters [58] it also makes use of the upper bound on $\eta(view_{\mathcal{A}})$ from Lemma 6.2 to show that a fewer number of samples is sufficient to compute the estimate $\eta'(view_{\mathcal{A}})$.

Lemma 6.3 Let $\rho(k) > 0$ be a function in k. If the experiment takes

$$s(k) := \mathcal{O}\big(n^2 \rho^{-2}(k) \ln\big((nq\rho(k))^{-1}\big)\,\big)$$

samples when computing the estimate $\eta'(view_{\mathcal{A}})$, then

$$|\Pr[X_2] - 4(n+1)q \cdot \Pr[X_3]| \leq \rho(k) \; .$$

Furthermore, the $s(k)$ samples can be computed in $\mathcal{O}(ns(k) \cdot \mathbf{T}_{\mathbb{Z}})$ time.

[2] Unfortunately, there seems not to be an efficient way to compute the exact value $\eta(view_{\mathcal{A}})$, we can only bound it using Lemma 6.2. If there was one we could greatly simplify our analysis.

The parameter $\rho(k)$ will be determined at the end of the proof.

Game 4. (Forced abort during the game I) Compared to the last game we make the following changes to the experiment: When identity $id \in ID$ is queried to the key derivation oracle, the experiment immediately aborts and returns $\beta' = 0$ if $y(id) = 0 \bmod p$. When receiving the challenge identity id^*, the experiment immediately aborts and returns $\beta' = 0$ if $y(id^*) \neq 0 \bmod p$. The artificial abort at the end of the experiment is the same as in the last game.

Clearly, this modification does not affect the adversary if there is no forced abort. In case there is a new forced abort the experiment outputs $\beta' = 0$ as in Game 3. Therefore we have

$$\Pr[\, X_4 \,] = \Pr[\, X_3 \,] \,.$$

Game 5. (Change key derivation oracle) The experiment changes its answers to all key derivation queries $\mathrm{Ex}(id)$ made by the adversary $\mathcal{A}$ as follows: By Eqn. (4) we have $\mathsf{H}(id) = g^{x(id)} u_1^{y(id)}$ for some values $x(id)$ and $y(id)$ known to the experiment.
Case $y(id) = 0 \bmod p$: The experiment aborts and returns $\beta' = 0$ (as in the last game).
Case $y(id) \neq 0 \bmod p$: The derived key $sk[id] = (d_1, d_2)$ is computed as follows:
For a random $r' \in \mathbb{Z}_p$, the experiment implicitly defines $r = -b/y(id) + r' \bmod p$ and computes

$$\begin{aligned} d_1 &\leftarrow (g^b)^{-x(id)/y(id)} g^{x(id)r'} u_1^{y(id)r'} \,, \\ d_2 &\leftarrow (g^b)^{-1/y(id)} \cdot g^{r'} \,. \end{aligned}$$

Note that the randomness r is not known to the experiment. Furthermore, the generation of the derived keys $sk[id] = (d_1, d_2)$ only depends on g^b and does not involve the knowledge of the secret key $\alpha = g^{ab}$ anymore. (However, the experiment still needs α to answer decapsulation queries.)

Lemma 6.4 $\Pr[X_5] = \Pr[X_4]$.

Proof: We have to verify that each derived key $sk[id] = (d_1, d_2)$ is identically distributed as in the last game. Let us abbreviate $x = x(id)$, and $y = y(id) \neq 0 \bmod p$. Clearly, if r' is uniform in $\mathbb{Z}_p$ so is r. Then by Equation (1) and since $r' = r + b/y$,

$$\begin{aligned} d_1 &= (g^b)^{-x/y} g^{xr'} u_1^{yr'} \\ &= g^{-bx/y} g^{xr'} u_1^{yr'} \\ &= g^{-bx/y} g^{x(r+b/y)} u_1^{y(r+b/y)} \\ &= g^{-bx/y} g^{xr+bx/y} u_1^{yr+b} \\ &= u_1^b \cdot g^{xr} u_1^{yr} \\ &= \alpha \cdot (g^x u_1^y)^r \\ &= \alpha \cdot (\mathsf{H}(id))^r \,, \end{aligned} \qquad \begin{aligned} d_2 &= (g^b)^{-1/y} \cdot g^{r'} \\ &= g^{-b/y} \cdot g^{r+b/y} \\ &= g^r \,, \end{aligned}$$

are distributed as in the last game (the original experiment). ▌

Game 6. (Change of the public key) In this game the experiment will modify the generation of the value u_2 from the public key pk. The experiment picks a random $d \in \mathbb{Z}_p$ and computes the value u_2 as $u_2 = (g^a)^{-t^*} g^d$, where $t^* = \mathsf{TCR}(c_1^*)$. To summarize, the public key $pk = (u_1, u_2, z, \mathsf{H})$ is now computed as

$$u_1 \leftarrow g^a, \quad u_2 \leftarrow (g^a)^{-t^*} g^d, \quad z \leftarrow \hat{e}(g^a, g^b) \,, \tag{6}$$

the hash keys as in Equation (3), and the secret key sk as $\alpha = g^{ab} = u_1^b$ that is still known to the experiment. The simulation of $\mathcal{A}$'s queries is done as before, using the secret key α. Note that the public key is identically distributed as in the last game. Therefore we have

$$\Pr[X_6] = \Pr[X_5] \; .$$

Game 7. (Forced abort during the game II) Compared to the last game we make the following changes to the experiment: When the tuple (id, C) is queried to the decapsulation oracle for $id \in ID \cup \{id^*\}$ and $C = (c_1, c_2, c_3)$ the experiment computes $t = \mathsf{TCR}(c_1)$ and immediately aborts if $y(id) = 0 \bmod p$, C is consistent, and $t = t^*$. In case of abort the experiment returns $\beta' = 0$.

Lemma 6.5 $|\Pr[X_7] - \Pr[X_6]| \leq \frac{q}{p}$.

Proof: Clearly, this modification does not affect the adversary if there is no new forced abort. Note that any new forced abort implies $c_1 = c_1^*$ since otherwise by $t = t^*$ the experiment already aborted in the last game (having found a collision in the hash function TCR). In case of a new forced abort we distinguish between two cases:
Case 1: the new forced abort happens during the *guess* stage. Recall that we call a ciphertext $C = (c_1, c_2, c_3)$ consistent if $(g, c_1, u_1^t u_2, c_3)$ is a Diffie-Hellman tuple (where $t = \mathsf{TCR}(c_1)$), i.e., if $(g, c_1, u_1^t u_2, c_3) = (g, g^r, u_1^t u_2, (u_1^t u_2)^r)$ for some value $r \in \mathbb{Z}_p$. Note that the way the public-key pk is generated by Eqn. (6) and since $c_1 = c_1^*$, and $t = t^*$, for a consistent ciphertext C we have

$$c_3 = (u_1^t u_2)^r = ((g^a)^{t-t^*} g^d)^r = (c_1^a)^{t-t^*} \cdot c_1^d = (c_1^*)^d = c_3^* \; , \qquad (7)$$

where $d \in \mathbb{Z}_p$ is only known to the experiment. If $id = id^*$ (i.e., if $\mathcal{A}$ queries the decapsulation oracle with the target identity) then $c_2^* = c_2$. Consequently $C = C^*$ and so the experiment rejects as in the original IB-KEM security experiment. If $id \neq id^*$ then, by definition, $id \in ID$ and the experiment outputs $\beta' = 0$ as in Game 6 where the abort was still done at the end of the experiment. Therefore, conditioned on case 1 this does not change the distribution of β' and we have $\Pr[X_7] = \Pr[X_6]$.

Case 2: the new forced abort happens during $\mathcal{A}$'s *find* stage. Since in the *find* stage the adversary has no information (in a statistical sense) about c_1^* from the challenge ciphertext C^*, and the adversary makes at most q decapsulation queries in its *find* stage, this implies

$$|\Pr[X_7] - \Pr[X_6]| \leq \frac{q}{p}$$

and concludes the proof. ▌

Game 8. (Change the answers to the decapsulation queries.) In the last game decapsulation queries were either aborted or answered using the secret key α, as in the original experiment. In this game the experiment changes its answers to its decapsulation queries $\textsc{Dec}(id, C)$ made by $\mathcal{A}$ as follows: By Eqn. (4) we have $\mathsf{H}(id) = g^{x(id)} u_1^{y(id)}$ for some values $x(id)$ and $y(id)$ known to the experiment.
Case $y(id) \neq 0 \bmod p$: the query is answered using $sk[id]$ obtained from the key derivation oracle.
Case $y(id) = 0 \bmod p$: the experiment simulates the decapsulation queries as follows: Let $C = (c_1, c_2, c_3)$ be the queried ciphertext and let $t = \mathsf{TCR}(c_1)$.

If the ciphertext is not consistent then return `reject`
If $t = t^*$ then the experiment aborts and returns $\beta' = 0$ (as in the last game)
if $t \neq t^*$ then return $K \leftarrow \hat{e}(c_3/c_1^d, g^b)^{(t-t^*)^{-1}}$.

Note that from this point on the experiment does not depend on the knowledge of $sk = g^{ab}$ anymore.

Lemma 6.6 $\Pr[X_8] = \Pr[X_7]$.

Proof: Let $C = (c_1, c_2, c_3)$ be an arbitrary ciphertext submitted to the decapsulation oracle with respect to identity id. If $y(id) \neq 0 \bmod p$ then decapsulation is done using the simulation of the key derivation oracle which we already showed to be correct so we may now assume $y(id) = 0 \bmod p$. Furthermore we may assume C is consistent because otherwise it gets rejected, as in the last game.

Case 1a: $t = t^*$ and $c_1 \neq c_1^*$. In this case the experiment has found a collision in the hash function TCR and returns $\beta' = 0$ (as in the last game).
Case 1b: $t = t^*$ and $c_1 = c_1^*$. In this case the experiment returns $\beta' = 0$ as in the forced abort introduced in the last game.

Case 2: $t \neq t^*$. Similar to Eqn. (7) consistency of C implies

$$c_3 = (u_1^t u_2)^r = ((g^a)^{t-t^*} g^d)^r = (c_1^a)^{t-t^*} \cdot c_1^d \ , \tag{8}$$

and we obtain

$$(c_3/c_1^d)^{(t-t^*)^{-1}} = ((c_1^a)^{t-t^*} c_1^d / c_1^d)^{(t-t^*)^{-1}} = c_1^a \ . \tag{9}$$

In the original IB-KEM decapsulation algorithm first the secret key for identity id is computed as $sk[id] = (d_1, d_2) = (\alpha \cdot \mathsf{H}(id)^s, g^s)$ for random $s = s(id)$, and then the session key K is reconstructed as

$$\begin{aligned} K = \hat{e}(c_1, d_1)/\hat{e}(c_2, d_2) &= \hat{e}(c_1, \alpha) \cdot \hat{e}(c_1, \mathsf{H}(id)^s)/\hat{e}(c_2, g^s) \\ &= \hat{e}(c_1^a, g^b) \cdot (\hat{e}(c_1, \mathsf{H}(id))/\hat{e}(c_2, g))^s \\ &\overset{(9)}{=} \hat{e}((c_3/c_1^d)^{(t-t^*)^{-1}}, g^b) \cdot (\hat{e}(c_1, \mathsf{H}(id)^s/\hat{e}(c_2, g))^s \\ &= \hat{e}(c_3/c_1^d, g^b)^{(t-t^*)^{-1}}, \end{aligned}$$

with $\Delta'(C) = \hat{e}(c_1, \mathsf{H}(id))/\hat{e}(c_2, g) = 1$ by consistency. This shows correctness of the new decapsulation algorithm. ▌

Game 9. (Modify the challenge) After $\mathcal{A}$'s *find* stage the experiment inputs the target identity id^* from $\mathcal{A}$. The experiment modifies the computation of the challenge ciphertext C^* follows:
Case $y(id^*) \neq 0 \bmod p$: The experiment aborts (as in the last game).
Case $y(id^*) = 0 \bmod p$: The experiment creates the challenge ciphertext $C^* = (c_1^*, c_2^*, c_3^*)$ and key K_1^* as

$$c_1^* \leftarrow g^c, \quad c_2^* \leftarrow (g^c)^{x(id^*)}, \quad c_3^* \leftarrow (g^c)^d, \quad K_1^* \leftarrow \hat{e}(g, g)^{abc} \ . \tag{10}$$

By virtue of Eqns. (4), (8), and since $\mathsf{TCR}(c_1^*) = t^*$ and $y(id^*) = 0 \bmod p$, C^* is a correctly distributed ciphertext of K_1^*. Note that the generation of the challenge ciphertext now only depends on g^c and $\hat{e}(g, g)^{abc}$, instead of c as in the last game. Clearly,

$$\Pr[X_9] = \Pr[X_8] \ .$$

Game 10. (Random CCA experiment) The experiment replaces the value K_1^* from the challenge C^* with K_0^*, where $K_0^* \overset{\$}{\leftarrow} \mathbb{G}_T$. This precisely models the IB-KEM CCA experiment with $\gamma = 0$ (random game) and hence we have

$$\Pr[X_{10}] = \Pr[\mathbf{Exp}_{\mathcal{IBKEM},\mathcal{A}}^{ibkem\text{-}\mathrm{cca}\text{-}0} = 1] \ .$$

Observe that Game 10 does not use the secret key anymore and that the whole simulation only depends on the values g^a, g^b, g^c. Game 9 and Game 10 are equal unless adversary $\mathcal{A}$ can distinguish $K_1^* = \hat{e}(g, g)^{abc}$ (in Game 9) from K_0^* (in Game 10), where $K_0 \overset{\$}{\leftarrow} \mathbb{G}_T$. Therefore we have

$$|\Pr[X_{10}] - \Pr[X_9]| \leq \mathbf{Adv}_{\mathcal{G},\mathcal{B}}^{\mathrm{bddh}}(k) \ ,$$

for any adversary $\mathcal{B}$ against the hardness of BDDH running in the same time as the experiment, i.e.,

$$\mathbf{T}_{\mathcal{B}}(k) = \mathbf{T}_{\mathcal{A}}(k) + \mathcal{O}(ns(k) \cdot \mathbf{T}_{\mathbb{Z}} + q \cdot \mathbf{T}_{\mathbb{PG}}(k)), \tag{11}$$

where $s(k)$ is the number of samples from Lemma 6.3 the experiment needs in order to compute $\eta'(view_{\mathcal{A}})$.

Analysis. Cellecting the probabilities relating the different games we have shown that given an adversary $\mathcal{A}$ that runs in time $\mathbf{T}_{\mathcal{A}}(k)$ and has advantage $\varepsilon_{\mathcal{A}}(k) = \mathbf{Adv}_{\mathcal{IBKEM},\mathcal{A}}^{ibkem\text{-}\mathrm{cca}}$, there exists an adversary $\mathcal{B}$ with advantage $\varepsilon_{\mathcal{B}}(k) = \mathbf{Adv}^{\mathrm{bddh}}_{\mathcal{G},\mathcal{B}}(k)$ and an adversary $\mathcal{H}$ that runs in time $\mathbf{T}_{\mathcal{H}}(k) \approx \mathbf{T}_{\mathcal{A}}(k)$ with advantage $\varepsilon_{\mathcal{H}}(k) = \mathbf{Adv}^{\mathrm{tcr}}_{\mathsf{TCR},\mathcal{H}}(k)$ such that

$$\begin{aligned}
\varepsilon_{\mathcal{A}}(k) &= \big|\Pr[X_0] - \Pr[X_{10}]\big| \\
&\leq \big|\Pr[X_1] - \Pr[X_{10}]\big| + \big|\Pr[X_0] - \Pr[X_1]\big| \\
&\leq \big|\Pr[X_1] - \Pr[X_{10}]\big| + \varepsilon_{\mathcal{H}}(k) \\
&\leq \big|4(n+1)q \cdot \Pr[\,X_3\,] - \Pr[X_{10}]\big| + \rho(k) + \varepsilon_{\mathcal{H}}(k) \\
&\leq \big|4(n+1)q \cdot \big(\Pr[\,X_7\,] + q/p\big) - \Pr[X_{10}]\big| + \rho(k) + \varepsilon_{\mathcal{H}}(k) \\
&\leq \big|4(n+1)q \cdot \big(\Pr[\,X_9\,] + q/p\big) - \Pr[X_{10}]\big| + \rho(k) + \varepsilon_{\mathcal{H}}(k) \\
&\leq 4(n+1)q \cdot (\varepsilon_{\mathcal{B}}(k) + q/p) + \rho(k) + \varepsilon_{\mathcal{H}}(k)\ .
\end{aligned}$$

The above implies

$$\varepsilon_{\mathcal{B}}(k) \geq \frac{\varepsilon_{\mathcal{A}}(k) - \varepsilon_{\mathcal{H}}(k) - \rho(k)}{5nq} - \frac{q}{p}\ . \tag{12}$$

Defining

$$\rho(k) := \frac{1}{2}\varepsilon_{\mathcal{A}}(k)\ , \tag{13}$$

we obtain

$$\varepsilon_{\mathcal{B}}(k) \quad \geq \quad \frac{\varepsilon_{\mathcal{A}}(k) - \varepsilon_{\mathcal{H}}(k)}{10nq} - \frac{q}{p}\ ,$$

where q is an upper bound on all (derivation plus decapsulation) queries made by $\mathcal{A}$. Using Equation (11) and the bound on $s(k)$ from Lemma 6.3 we bound $\mathcal{B}$'s running time as

$$\begin{aligned}
\mathbf{T}_{\mathcal{B}}(k) &= \mathbf{T}_{\mathcal{A}}(k) + \mathcal{O}(n \cdot s(k) \cdot \mathbf{T}_{\mathbb{Z}} + q \cdot \mathbf{T}_{\mathbb{PG}}(k)) \\
&= \mathbf{T}_{\mathcal{A}}(k) + \mathcal{O}(n \cdot n^2 \rho^{-2}(k) \ln\big((nq\rho(k))^{-1}\big) \cdot \mathbf{T}_{\mathbb{Z}} + q \cdot \mathbf{T}_{\mathbb{PG}}(k)) \\
&= \mathbf{T}_{\mathcal{A}}(k) + \mathcal{O}(n^3 \varepsilon_{\mathcal{A}}^{-2} \cdot \ln((nq\varepsilon_{\mathcal{A}})^{-1}) \cdot \mathbf{T}_{\mathbb{Z}}(k) + q \cdot \mathbf{T}_{\mathbb{PG}}(k))\ .
\end{aligned}$$

This concludes the proof of Theorem 4.1.

6.2 Proof of Lemma 6.2

Fix $view_{\mathcal{A}}$ and hence the queried identities $id^{(1)}, \ldots, id^{(q_0)}, id^*$. We abbreviate $\eta = \eta(view_{\mathcal{A}})$. For an integer t, define the event

$$\mathrm{E}_t : \quad \bigwedge_{i=1}^{q_0} (y(id^{(i)}) \neq 0 \bmod t) \wedge y(id^*) = 0 \bmod t\ .$$

With this notation recall that $\eta = \Pr_{\mathbf{Y}}[\mathrm{E}_p]$ and we intend to show that

$$\frac{1}{4(n+1)q} \ \leq\ \eta\ \leq\ \frac{1}{2q}\ , \tag{14}$$

(Also recall that $\mathbf{Y} = \{y'_0, y_1, \ldots, y_n, \ell\}$, where the random variables $\{y'_0, y_1, \ldots, y_n, \ell\}$ are distributed as in Equation (3).) Over the integers we have by Equation (3) $y(id) = y'_0 + \sum_{i=1}^{n} id_i y_i - \ell m$ for some integer $0 \leq \ell \leq n+1$, where $0 \leq y'_0 + \sum_{i=1}^{n} id_i y_i < (n+1)m < p$. If $\ell = \ell^* := \lfloor (y'_0 + \sum_{i=1}^{n} id^*_i y_i)/m \rfloor$

and $y(id^*) = 0 \bmod m$, then clearly $y(id^*) = 0 \bmod p$. On the other hand, if $y(id) \neq 0 \bmod m$ then $y(id) \neq 0 \bmod p$. Hence,

$$\begin{aligned} \eta = \Pr_{\mathbf{Y}}[\mathrm{E}_p] &\geq \Pr[\ell = \ell^*] \Pr_{\mathbf{Y}}[\mathrm{E}_p \mid \ell = \ell^*] \\ &= \frac{1}{n+1} \Pr_{\mathbf{Y}}[\mathrm{E}_p \mid \ell = \ell^*] \\ &\geq \frac{1}{n+1} \Pr_{\mathbf{Y}}[\mathrm{E}_m \mid \ell = \ell^*] \\ &= \frac{1}{n+1} \Pr_{\mathbf{Y}'}[\mathrm{E}_m] , \end{aligned}$$

where the probability space $\mathbf{Y}'$ contains the random variables $\{y_0', y_1, \ldots, y_n\}$ distributed according to Equation (3), for fixed ℓ. Define $\Pr_{\mathbf{Y}'}[\mathrm{E}_m] =: \eta_m$. Since trivially $\eta_m \geq \eta$, we obtain

$$\frac{1}{n+1} \cdot \eta_m \;\leq\; \eta \;\leq\; \eta_m \,. \tag{15}$$

It remains to compute an upper and lower bound on η_m.

Let $id \neq id'$ and $a, b \in \mathbb{Z}$. We collect some simple observations on function $y(\cdot)$ which essentially show that the $y(\cdot) \bmod m$ are pairwise independent:

$$\Pr_{\mathbf{Y}'}[y(id) = b \bmod m] \;=\; 1/m \tag{16}$$

$$\Pr_{\mathbf{Y}'}[y(id) = a \bmod m \mid y(id') = b \bmod m] \;=\; 1/m \,. \tag{17}$$

Equation (16) follows since for any choice of $y_1, ..., y_n$ there is a single choice of y_0' that will make the condition hold. To show Equation (17) assume there exists an index $1 \leq i \leq n$ such that $id_i = 1$ and $id_i' = 0$. Then fix all y_j's for $j \neq i$ except y_i so that $y(id') = b$. Therefore $\Pr\big[\, y(id) = a \mid y(id') = b \,\big] = 1/m$. If there is no such i then we can use Bayes to reverse roles of id and id'.

We continue to bound η_m with

$$\begin{aligned} \eta_m \;&=\; \Pr_{\mathbf{Y}'}[\bigwedge_{i=1}^{q_0} y(id^{(i)}) \neq 0 \bmod m \mid y(id^*) = 0 \bmod m] \cdot \Pr[y(id^*) = 0 \bmod m] \\ &\overset{(16)}{=} \frac{1}{m} \cdot \Pr_{\mathbf{Y}'}[\bigwedge_{i=1}^{q_0} y(id^{(i)}) \neq 0 \bmod m \mid y(id^*) = 0 \bmod m] \\ &= \frac{1}{m} \cdot (1 - \Pr_{\mathbf{Y}'}[\bigvee_{i=1}^{q_0} y(id^{(i)}) = 0 \bmod m \mid y(id^*) = 0 \bmod m]) \\ &\geq \frac{1}{m} \cdot (1 - \sum_{i=1}^{q_0} \Pr_{\mathbf{Y}'}[y(id^{(i)}) = 0 \bmod m \mid y(id^*) = 0 \bmod m]) \qquad (18) \\ &\overset{(17)}{=} \frac{1}{m} \cdot (1 - \sum_{i=1}^{q_0} \frac{1}{m}) \\ &= \frac{1}{m} \cdot (1 - \frac{q}{m}) \\ &= \frac{1}{4q} , \end{aligned}$$

where the last equation follows by our choice of $m = 2q$ which minimizes the term. Furthermore, we obtain $\eta_m \leq 1/m = 1/(2q)$ by replacing the union bound from Equation (18) by the trivial bound $(1 - \Pr_{\mathbf{Y}'}[\cdots]) \leq 1$. Together with Equation (15) this proves Equation (14).

6.3 Proof of Lemma 6.3

For the proof we can assume $\rho \leq 1$ since otherwise the lemma is trivially true.

Let ArtAbort be the event that the experiment artificially aborts at the end of the simulation. Let Abort = ArtAbort $\vee$ ForcedAbort be the event that it aborts artificially or forced. First we claim

Claim 6.7 For any fixed $view_{\mathcal{A}}$, $|\Pr[\neg\textsc{Abort}] - \lambda_{\text{low}}| \leq \lambda_{\text{low}}\rho$.

The proof of the claim is postponed until later. Since the claim holds for any fixed $view_{\mathcal{A}}$ it also remains true conditioned on $\gamma' = 1$:

$$|\Pr[\,\neg\textsc{Abort} \mid \gamma' = 1\,] - \lambda_{\text{low}}| \leq \lambda_{\text{low}}\rho\ . \tag{19}$$

In case of abort the experiment outputs $\beta' = 0$, otherwise it outputs $\beta' = \gamma'$. We continue computing $\Pr[\,X_3\,]$:

$$\begin{aligned}
\Pr[\,X_3\,] &= \Pr[\,\beta' = 1 \wedge \textsc{Abort}\,] + \Pr[\,\beta' = 1 \wedge \neg\textsc{Abort}\,] \\
&= 0 + \Pr[\,\beta' = 1 \wedge \neg\textsc{Abort}\,] \\
&= \Pr[\,\gamma' = 1 \wedge \neg\textsc{Abort}\,] \\
&= \Pr[\,\neg\textsc{Abort} \mid \gamma' = 1\,] \cdot \Pr[\,\gamma' = 1\,] \\
&= \Pr[\,\neg\textsc{Abort} \mid \gamma' = 1\,] \cdot \Pr[\,X_2\,]
\end{aligned}$$

where the last equation holds since $\Pr[\,X_2\,] = \Pr[\,\gamma' = 1\,]$, i.e., in Game 2 the experiment outputs whatever adversary $\mathcal{A}$ outputs.

Combining this with Equation (19) we get

$$\begin{aligned}
|\Pr[\,X_3\,] - \lambda_{\text{low}} \cdot \Pr[\,X_2\,]| &= \Pr[X_2] \cdot |\Pr[\,\neg\textsc{Abort} \mid \gamma' = 1\,] - \lambda_{\text{low}}| \\
&\leq \Pr[\,X_2\,] \cdot \lambda_{\text{low}}\rho \\
&\leq \lambda_{\text{low}}\rho\ .
\end{aligned}$$

It remains to prove Claim 6.7 which requires the following bound from [31].

Lemma 6.8 [Hoeffding's bound] Let $X_1, \ldots, X_s$ be independent random variables with $a \leq X_i \leq b$ and define $X = \frac{1}{s} \cdot \sum_{i=1}^{s} X_i$. Then, for any $t > 0$, we have the inequality

$$\Pr[|X - \mathbf{E}[X]| \geq t] \leq 2e^{-2s\left(\frac{t}{b-a}\right)^2},$$

where $\mathbf{E}[X]$ denotes the expected values of X.

Proof of Claim 6.7. We abbreviate $\eta = \eta(view_{\mathcal{A}})$ and $\eta' = \eta'(view_{\mathcal{A}})$. By construction the two events ArtAbort and ForcedAbort are independent and consequently we have

$$\Pr[\,\neg\textsc{Abort}\,] = \Pr[\,\neg\textsc{ForcedAbort}\,] \cdot \Pr[\,\neg\textsc{ArtAbort}\,] = \eta \cdot \Pr[\,\neg\textsc{ArtAbort}\,]\ . \tag{20}$$

We make

$$s(k) := 2q^2 \cdot (\lambda_{\text{low}}\rho)^{-2} \ln((\frac{1}{8}\lambda_{\text{low}}\rho)^{-1}) = \mathcal{O}(n^2\rho^{-2}(k) \ln\left((nq\rho(k))^{-1}\right)) \tag{21}$$

samples to compute an approximation η' of η, where $\lambda_{\text{low}} = 1/(4(n+1)q)$. For each sample we pick $y_0', y_1, \ldots, y_n, \ell$ independently according to the distribution $\mathbf{Y}$ from Equation (3) which defines the function $y(\cdot)$. Depending on $y(\cdot)$, each indicator variable X_i is defined as

$$X_i := \begin{cases} 1: & \bigwedge_{i=1}^{q_0} \left(y(id^{(i)}) \neq 0 \bmod p\right) \wedge y(id^*) = 0 \bmod p \\ 0: & \text{otherwise.} \end{cases} \tag{22}$$

By construction, $\lambda_{\text{low}} \leq \Pr[\, X_i = 1\,] \leq \lambda_{\text{up}}$. Finally, we make a majority decision over all X_i by computing $\eta' = \sum \frac{1}{s(k)} \cdot \sum_{i=1}^{s(k)} X_i$. By construction, $\mathbf{E}[\eta'] = \eta \geq \lambda_{\text{low}}$. Using Lemma 6.8 for the estimate η' of η, with $t := \eta\rho/4$ and $b - a \leq \lambda_{\text{up}} - \lambda_{\text{low}} \leq 1/(2q)$, we get[3]

$$\begin{aligned} \Pr[|\eta' - \eta| \geq \eta\rho/4] &< 2e^{-2s\left(\frac{\eta\rho q}{2}\right)^2} \\ &\overset{\eta \geq \lambda_{\text{low}}}{\leq} 2e^{-2s\left(\frac{\lambda_{\text{low}}\rho q}{2}\right)^2} \\ &\overset{(21)}{\leq} \frac{1}{4}\lambda_{\text{low}}\rho \,. \end{aligned}$$

Set $\rho' := \rho/4$. We call the approximation η' "good" if $|\eta' - \eta| \leq \eta\rho'$ and "bad" otherwise. Then the above establishes

$$\Pr[\eta' \text{ BAD}] \leq \lambda_{\text{low}}\rho'. \tag{23}$$

For every fixed good η' we have $\eta(1-\rho') \leq \max\{\lambda_{\text{low}}, \eta'\} \leq \eta(1+\rho')$. Since $\Pr[\,\neg\textsc{ArtAbort}\,] = \lambda_{\text{low}} / \max\{\lambda_{\text{low}}, \eta'\}$ we have

$$\frac{\lambda_{\text{low}}}{\eta(1+\rho')} \leq \Pr[\,\neg\textsc{ArtAbort} \mid \eta' \text{ GOOD}\,] \leq \frac{\lambda_{\text{low}}}{\eta(1-\rho')} \tag{24}$$

We first give a lower bound on $\Pr[\,\neg\textsc{Abort}\,]$.

$$\begin{aligned} \Pr[\,\neg\textsc{Abort}\,] &\overset{(20)}{=} \eta \cdot \Pr[\,\neg\textsc{ArtAbort}\,] \\ &\geq \eta \cdot \Pr[\,\neg\textsc{ArtAbort} \mid \eta' \text{ GOOD}\,] \Pr[\,\eta' \text{ GOOD}\,] \\ &\overset{(23),(24)}{\geq} \eta \cdot \frac{\lambda_{\text{low}}}{\eta(1+\rho')} \cdot (1 - \lambda_{\text{low}}\rho') \\ &\geq \lambda_{\text{low}} \cdot (1-\rho')^2 \geq \lambda_{\text{low}} \cdot (1-\rho) \,. \end{aligned}$$

We now turn to the upper bound on $\Pr[\,\neg\textsc{Abort}\,]$.

$$\begin{aligned} \Pr[\,\neg\textsc{Abort}\,] &\overset{(20)}{=} \eta \cdot (\Pr[\,\neg\textsc{ArtAbort} \mid \eta' \text{ GOOD}\,] \Pr[\,\eta' \text{ GOOD}\,] + \Pr[\,\neg\textsc{ArtAbort} \mid \eta' \text{ BAD}\,] \Pr[\,\eta' \text{ BAD}\,]) \\ &\leq \eta \cdot (\Pr[\,\neg\textsc{ArtAbort} \mid \eta' \text{ GOOD}\,] + \Pr[\,\eta' \text{ BAD}\,]) \\ &\overset{(23),(24)}{\leq} \eta \cdot \left(\frac{\lambda_{\text{low}}}{\eta(1-\rho')} + \lambda_{\text{low}}\rho' \right) \\ &\leq \lambda_{\text{low}} \cdot \left(\frac{1}{1-\rho'} + \rho' \right) \\ &\overset{\rho' \leq 1/4}{\leq} \lambda_{\text{low}} \cdot (1 + 4\rho') = \lambda_{\text{low}} \cdot (1+\rho) \,. \end{aligned}$$

To complete the proof we need to establish the bound on the running time necessary to compute the samples $X_1, \ldots, X_{s(k)}$. To compute one indicator variable X_i, one has to sample once from distribution $\mathbf{Y}$ and evaluate the functions $y(\cdot)$ on $y(id^*)$ and $id^{(1)}, \ldots, id^{(q)}$ according to Equation (22). Evaulating each of the functions $y(\cdot)$ takes $n+1$ additions modulo p. Hence, the samples and hence the approximation η' can be computed with $\mathcal{O}(qns(k))$ additions modulo p.

We now sketch how to compute the approximation η' with $\mathcal{O}(ns(k))$ additions modulo p. Note that for most of the X_i's it is sufficient to check $y(id^*) \neq 0 \bmod p$ in which case one can conclude $X_i = 0$. If $y(id^*) = 0 \bmod p$ then one also has to evaluate $y(\cdot)$ also on $id^{(1)}, \ldots, id^{(q)}$. However, since by Equation (16), $\Pr_{\mathbf{Y}}[y(id^*) = 0 \bmod p] \leq \Pr_{\mathbf{Y}}[y(id^*) = 0 \bmod m] \leq 1/2q$, this only has to be done on an expected q fraction of all the samples. We therefore modify the simulation such that it aborts whenever the total number of additions modulo p used to compute η' exceeds $c \cdot ns(k)$, for some fixed constant c. Since by the Hoeffding bound (Lemma 6.8) this additional abort only happens with negligible probability, it does not affect the adversary's overall success probability.

[3] At this point Waters [58] used $b \leq 1$ instead of our refined upper bound $b \leq \lambda_{\text{up}} = 1/(2q)$ from Lemma 6.2.

7 Comparison

In this section we compare our scheme with the previous chosen-ciphertext secure IBE schemes in the literature based on the BDDH assumption.

Previous to this work there were basically two ways of building chosen-ciphertext secure IBE schemes in the standard model. One way is to combine the IBE schemes [8, 58] with the generic transformation from [11], the other one stems from a remark from [15]. We will now carefully review both constructions and compare them to our proposed scheme.

7.1 IB-KEM scheme obtained by the generic BCHK transformation

We begin by reviewing the generic transformations from any chosen-plaintext secure 2-HIBE into a chosen-ciphertext secure IBE scheme by Boneh, Canetti, Halevi, and Katz [11] (BCHK), We describe the CHK transformation in terms of key encapsulation and note that this is not possible for the improved BK transformation.

The one-time signature based BCHK method transforms any two level HIB-KEM into an IB-KEM scheme as follows: the identity of the first level HIB-KEM becomes the identity of the IB-KEM scheme. To create a ciphertext of the IB-KEM a random pair of one-time signing/verification keys is chosen. A HIB-KEM ciphertext for the message is created with respect to the two-level identity consisting of the HIB-KEM identity at the first level and the verification key at the second level. The resulting HIB-KEM ciphertext is signed using the signing key. Finally, the IB-KEM ciphertext is then composed by the HIB-KEM ciphertext, the signature, and the corresponding verification key. For decapsulation first the validity of the signature is checked and then, conditioned it was correct, the HIB-KEM ciphertext is decapsulated using the hierarchical key-derivation algorithm for the 2-level "identity" consisting of id plus the verification key.

It was proved in [11] that any chosen-plaintext secure 2-HIB-KEM with selective-identity security with respect to the second level of the hierarchy and adaptive security at the first level is sufficient to obtain a chosen-ciphertext secure IBE. Consequently, as noted in [58], the most efficient instantiation of this transformation is obtained from the hybrid HIB-KEM using Waters IB-KEM [58] at the first level and Boneh/Boyen's IB-KEM [8] at the second level.

Combining the results from [11] with [58, 8] we get a chosen-ciphertext secure IB-KEM under the BDDH assumption. Similar to our scheme, the security reduction roughly comes with a multiplicative factor of $\approx nq$.

7.2 IB-KEM mentioned in BMW

In contrast to [11], Boyen, Mei, and Waters [15] propose a non black-box technique to obtain a chosen-ciphertext secure IB-KEM from a 2-level HIB-KEM without relying on additional primitives like signatures or MACs. For concreteness we cite their concrete statement (from Sec. 5.3 of the full version of [15]), referring to the two HIBE constructions from Boneh-Boyen [8] and Waters [58]:

> "It is easy to see that we obtain the desired result [i.e. a construction avoiding a signature/MAC] very simply, by extending the hierarchy in either HIBE construction by one level, and setting the "identity" for that last level to be the hash value of the previous ciphertext components. This gives us (in the Waters case) an adaptive-identity CCA2-secure HIBE, and (in the Boneh-Boyen case) a selective-identity CCA2-secure HIBE."

No theorem statement is given but it is clear that security relies on Waters 2-HIBE which has a loss-factor of roughly $(nq)^2$ in the reduction from BDDH. We want to stress that in their construction the "hashing the previous ciphertext" makes it basically impossible to replace the second level of Waters HIBE with the more efficient (but only weakly=selective-identity secure) IBE scheme from Boneh-Boyen. This is since the challenge ciphertext depends on the target identity which is used in the second-level of the HIBE scheme and the target identity is not known until the adversary outputs it.

Since the construction uses Waters 2-HIBE the public-key has to include two independent sets of hash public-keys, i.e. the public key contains roughly $2n$ elements from $\mathbb{G}_1$. For the same reason the security

Scheme	CCA?	Ciphertext	Encapsulation	Decapsulation	Keysize	Security
		Overhead	#pairings + #[multi,regular,fixed-base]-exp+...		pk	Reduction
Ours (§4)	$\surd$	3ℓ	$0+[1,3,1]$	$3+[1,0,2]$	$n+4$	nq
Hybrid + CHK (§7.1)	$\surd$	3ℓ+$25k$ (704)	$0+[1,3,1]$+Sig	$3+[1,0,1]$+Vfy	$n+4$	nq
BMW (§7.2)	$\surd$	$3\ell'$	$0+[0,5,1]$	$4+[0,1,0]$	$2n+3$	$(nq)^2$
Waters	—	2ℓ	$0+[0,3,1]$	$2+[0,0,0]$	$n+2$	nq

Figure 2: Efficiency comparison for CCA-secure IB-KEMs for identity-space $IDSp = \{0,1\}^n$. BMW is the IB-KEM as proposed in [15], Hybrid + BK is the Waters/BB hybrid HIB-KEM scheme applied to the CHK transformation as proposed in [58], and Waters is Waters' original chosen-plaintext secure IBE scheme [58]. The keysize is measured in terms of the number of group elements of the public key *pk*. Ciphertext overhead represents the difference in bits between the ciphertext length and the message length. ℓ is the length of the representation of an element in $\mathbb{G}_1$ with respect to the security reduction $O(nq)$, while ℓ' is the length of a $\mathbb{G}_1$ group element with respect to the security reduction $O(n^2q^2)$, and thus $\ell \ll \ell'$. For comparison we mention that relative timings for the various operations are as follows: bilinear pairing ≈ 5 [53], multi-exponentiation ≈ 1.5, regular exponentiation $= 1$, fixed-base exponentiation $\ll 0.2$.

reduction of the proposed IBE scheme depends on the security of Waters' 2-HIBE which is quadratic in q and n.

7.3 A comparison

An efficiency comparison between the above two schemes, plus Waters original scheme, and our IBE is given in Figure 2. We further discuss it in the following prose. We stress that the performance of the security reduction is a crucial parameter here. In light of the keysize/security reduction tradeoff from Section 5.3 we can also compare the BMW scheme to all other schemes by "normalizing" the security reduction for all schemes to $O(n^2q^2)$, i.e. by setting the tradeoff parameter l to $l = \log_2(nq) \approx 20+7 = 27$ (for very optimistic $< 2^{20}$ adversary queries and identities of $n = 160$ bits) we get a public-key size of $n/27 + 4 \approx 9$ group elements compared to the $2n + 3 = 323$ group elements of BMW with the *same security.*

The symmetric overhead of the BCHK transformation consists of a strong one-time signature plus a verification key which sums up to $\approx 160^2 = 25600$ ("security parameter squared") bits [23].

Since computing Waters hash requires computing $n/2$ products in $\mathbb{G}_1$ on the average, where $n \approx \log_2 p$, it can be seen as a single exponentiation. Therefore we count computing $\mathsf{H}(id)^r$ for random r as two exponentiations. In the decapsulation algorithm of our IB-KEM we assume $\mathsf{H}(id)$ to be precomputed. The size of the secret key *sk* is the same for all three schemes – a single element in $\mathbb{G}_1$.

To summarize, compared to the BMW IB-KEM from Section 7.2, our proposed chosen-ciphertext secure IBE scheme achieves better performance and public key sizes with half of the BMW public key size. In addition, the fact that our security reduction is more efficient than that of the BMW scheme means that, for concrete values of the security parameter, our ciphertexts are much shorter even if the two schemes have the same number of elements in the ciphertext.

In terms of computational efficiency our scheme has one fixed-based exponentiation more than the the Hybrid + CHK scheme from Section 7.1, but it does not have to resort on any kind of exogenous consistency check such as a signature or a MAC. Since one fixed-based exponentiation is $\ll 0.2$ regular exponentiations, we conclude that encapsulation/decapsulation in our scheme is as efficient as in the Hybrid + CHK scheme. The most striking difference, however, is that our scheme comes with shorter ciphertexts: for current security requirements the ciphertexts difference (a strong one-time signature plus a verification key) amounts to a couple of thousand bits [23].

We remark that, in order to get a direct full IBE scheme (in contrast to an IB-KEM) we can also apply the MAC-based BCHK transformation [11] to the construction from Section 7.1 and get a full IBE scheme with shorter ciphertexts. The latter construction significantly reduces the ciphertext overhead compared to the CHK-transformation by replacing the signature with a MAC. Compared to our construction, however, there is still a difference in the ciphertext size of a MAC tag plus a “commitment” which sums up to ≈ 576 bits [11].

References

[1] American National Standards Institute (ANSI) X9.F1 subcommittee. ANSI X9.63 Public key cryptography for the Financial Services Industry: Elliptic curve key agreement and key transport schemes, July 5, 1998. Working draft version 2.0. (Cited on page 2.)

[2] Mihir Bellare, Anand Desai, David Pointcheval, and Phillip Rogaway. Relations among notions of security for public-key encryption schemes. In Hugo Krawczyk, editor, *Advances in Cryptology – CRYPTO'98*, volume 1462 of *Lecture Notes in Computer Science*, pages 26–45. Springer-Verlag, Berlin, Germany, August 1998. (Cited on page 1.)

[3] Mihir Bellare and Chanathip Namprempre. Authenticated encryption: Relations among notions and analysis of the generic composition paradigm. In Tatsuaki Okamoto, editor, *Advances in Cryptology – ASIACRYPT 2000*, volume 1976 of *Lecture Notes in Computer Science*, pages 531–545. Springer-Verlag, Berlin, Germany, December 2000. (Cited on page 8.)

[4] Mihir Bellare and Thomas Ristenpart. Simulation without the artificial abort: Improved concrete security for waters' IBE scheme. In *EUROCRYPT 2009*, 2009. (Cited on page 3.)

[5] Mihir Bellare and Phillip Rogaway. Random oracles are practical: A paradigm for designing efficient protocols. In V. Ashby, editor, *ACM CCS 93: 1st Conference on Computer and Communications Security*, pages 62–73. ACM Press, November 1993. (Cited on page 1.)

[6] Mihir Bellare and Phillip Rogaway. The exact security of digital signatures: How to sign with RSA and Rabin. In Ueli M. Maurer, editor, *Advances in Cryptology – EUROCRYPT'96*, volume 1070 of *Lecture Notes in Computer Science*, pages 399–416. Springer-Verlag, Berlin, Germany, May 1996. (Cited on page 2.)

[7] Kamel Bentahar, Pooya Farshim, John Malone-Lee, and Nigel P. Smart. Generic constructions of identity-based and certificateless KEMs. *Journal of Cryptology*, 21(2):178–199, April 2008. (Cited on page 2, 8.)

[8] Dan Boneh and Xavier Boyen. Efficient selective-ID secure identity based encryption without random oracles. In Christian Cachin and Jan Camenisch, editors, *Advances in Cryptology – EUROCRYPT 2004*, volume 3027 of *Lecture Notes in Computer Science*, pages 223–238. Springer-Verlag, Berlin, Germany, May 2004. (Cited on page 2, 9, 20.)

[9] Dan Boneh and Xavier Boyen. Secure identity based encryption without random oracles. In Matthew Franklin, editor, *Advances in Cryptology – CRYPTO 2004*, volume 3152 of *Lecture Notes in Computer Science*, pages 443–459. Springer-Verlag, Berlin, Germany, August 2004. (Cited on page 2.)

[10] Dan Boneh, Xavier Boyen, and Eu-Jin Goh. Hierarchical identity based encryption with constant size ciphertext. In Ronald Cramer, editor, *Advances in Cryptology – EUROCRYPT 2005*, volume 3494 of *Lecture Notes in Computer Science*, pages 440–456. Springer-Verlag, Berlin, Germany, May 2005. (Cited on page 3, 5.)

[11] Dan Boneh, Ran Canetti, Shai Halevi, and Jonathan Katz. Chosen-ciphertext security from identity-based encryption. *SIAM Journal on Computing*, 36(5):915–942, 2006. (Cited on page 2, 20, 21, 22.)

[12] Dan Boneh and Matthew K. Franklin. Identity-based encryption from the Weil pairing. In Joe Kilian, editor, *Advances in Cryptology – CRYPTO 2001*, volume 2139 of *Lecture Notes in Computer Science*, pages 213–229. Springer-Verlag, Berlin, Germany, August 2001. (Cited on page 1.)

[13] Dan Boneh and Matthew K. Franklin. Identity based encryption from the Weil pairing. *SIAM Journal on Computing*, 32(3):586–615, 2003. (Cited on page 1, 4, 5.)

[14] Dan Boneh and Jonathan Katz. Improved efficiency for CCA-secure cryptosystems built using identity-based encryption. In Alfred Menezes, editor, *Topics in Cryptology – CT-RSA 2005*, volume 3376 of *Lecture Notes in Computer Science*, pages 87–103. Springer-Verlag, Berlin, Germany, February 2005. (Cited on page 2.)

[15] Xavier Boyen, Qixiang Mei, and Brent Waters. Direct chosen ciphertext security from identity-based techniques. In Vijayalakshmi Atluri, Catherine Meadows, and Ari Juels, editors, *ACM CCS 05: 12th Conference on Computer and Communications Security*, pages 320–329. ACM Press, November 2005. (Cited on page 2, 3, 5, 7, 9, 20, 21.)

[16] Ran Canetti, Oded Goldreich, and Shai Halevi. The random oracle methodology, revisited (preliminary version). In *30th Annual ACM Symposium on Theory of Computing*, pages 209–218. ACM Press, May 1998. (Cited on page 1.)

[17] Ran Canetti, Shai Halevi, and Jonathan Katz. Chosen-ciphertext security from identity-based encryption. In Christian Cachin and Jan Camenisch, editors, *Advances in Cryptology – EUROCRYPT 2004*, volume 3027 of *Lecture Notes in Computer Science*, pages 207–222. Springer-Verlag, Berlin, Germany, May 2004. (Cited on page 2.)

[18] Sanjit Chatterjee and Palash Sarkar. Trading time for space: Towards an efficient IBE scheme with short(er) public parameters in the standard model. In Dongho Won and Seungjoo Kim, editors, *ICISC 05: 8th International Conference on Information Security and Cryptology*, volume 3935 of *Lecture Notes in Computer Science*, pages 424–440. Springer-Verlag, Berlin, Germany, December 2005. (Cited on page 3, 9.)

[19] Sanjit Chatterjee and Palash Sarkar. New constructions of constant size ciphertext hibe without random oracle. In *ICISC*, pages 310–327, 2006. (Cited on page 3.)

[20] Clifford Cocks. An identity based encryption scheme based on quadratic residues. In Bahram Honary, editor, *Cryptography and Coding, 8th IMA International Conference*, volume 2260 of *Lecture Notes in Computer Science*, pages 360–363, Cirencester, UK, December 17–19, 2001. Springer-Verlag, Berlin, Germany. (Cited on page 1.)

[21] Ronald Cramer and Victor Shoup. Design and analysis of practical public-key encryption schemes secure against adaptive chosen ciphertext attack. *SIAM Journal on Computing*, 33(1):167–226, 2003. (Cited on page 2, 5, 8.)

[22] Yevgeniy Dodis, Roberto Oliveira, and Krzysztof Pietrzak. On the generic insecurity of the full domain hash. In Victor Shoup, editor, *Advances in Cryptology – CRYPTO 2005*, volume 3621 of *Lecture Notes in Computer Science*, pages 449–466. Springer-Verlag, Berlin, Germany, August 2005. (Cited on page 1.)

[23] Shimon Even, Oded Goldreich, and Silvio Micali. On-line/off-line digital signatures. *Journal of Cryptology*, 9(1):35–67, 1996. (Cited on page 21.)

[24] Eiichiro Fujisaki and Tatsuaki Okamoto. How to enhance the security of public-key encryption at minimum cost. In Hideki Imai and Yuliang Zheng, editors, *PKC'99: 2nd International Workshop on Theory and Practice in Public Key Cryptography*, volume 1560 of *Lecture Notes in Computer Science*, pages 53–68. Springer-Verlag, Berlin, Germany, March 1999. (Cited on page 1.)

[25] David Galindo and Eike Kiltz. Chosen-ciphertext secure threshold identity-based key encapsulation without random oracles. In Roberto De Prisco and Moti Yung, editors, *SCN 06: 5th International Conference on Security in Communication Networks*, volume 4116 of *Lecture Notes in Computer Science*, pages 173–185. Springer-Verlag, Berlin, Germany, September 2006. (Cited on page 7.)

[26] Craig Gentry. Practical identity-based encryption without random oracles. In Serge Vaudenay, editor, *Advances in Cryptology – EUROCRYPT 2006*, volume 4004 of *Lecture Notes in Computer Science*, pages 445–464. Springer-Verlag, Berlin, Germany, May / June 2006. (Cited on page 3.)

[27] Craig Gentry and Alice Silverberg. Hierarchical ID-based cryptography. In Yuliang Zheng, editor, *Advances in Cryptology – ASIACRYPT 2002*, volume 2501 of *Lecture Notes in Computer Science*, pages 548–566. Springer-Verlag, Berlin, Germany, December 2002. (Cited on page 2, 8.)

[28] Shai Halevi. EME*: Extending EME to handle arbitrary-length messages with associated data. In Anne Canteaut and Kapalee Viswanathan, editors, *Progress in Cryptology - INDOCRYPT 2004: 5th International Conference in Cryptology in India*, volume 3348 of *Lecture Notes in Computer Science*, pages 315–327. Springer-Verlag, Berlin, Germany, December 2004. (Cited on page 9.)

[29] Shai Halevi and Phillip Rogaway. A tweakable enciphering mode. In Dan Boneh, editor, *Advances in Cryptology – CRYPTO 2003*, volume 2729 of *Lecture Notes in Computer Science*, pages 482–499. Springer-Verlag, Berlin, Germany, August 2003. (Cited on page 9.)

[30] Shai Halevi and Phillip Rogaway. A parallelizable enciphering mode. In Tatsuaki Okamoto, editor, *Topics in Cryptology – CT-RSA 2004*, volume 2964 of *Lecture Notes in Computer Science*, pages 292–304. Springer-Verlag, Berlin, Germany, February 2004. (Cited on page 9.)

[31] W. Hoeffding. Probability inequalities for sums of bounded random variables. *J. Amer. Statist. Assoc.*, 58:13–30, 1963. (Cited on page 18.)

[32] Jeremy Horwitz and Ben Lynn. Toward hierarchical identity-based encryption. In Lars R. Knudsen, editor, *Advances in Cryptology – EUROCRYPT 2002*, volume 2332 of *Lecture Notes in Computer Science*, pages 466–481. Springer-Verlag, Berlin, Germany, April / May 2002. (Cited on page 2, 8.)

[33] IEEE P1363.3 Committee. IEEE 1363.3 / CfS — standard for identity-based cryptographic techniques using pairings. `http://grouper.ieee.org/groups/1363/index.html/`, February 2006. Call for submissions. (Cited on page 1.)

[34] IEEE P1363a Committee. IEEE P1363a / D9 — standard specifications for public key cryptography: Additional techniques. `http://grouper.ieee.org/groups/1363/index.html/`, June 2001. Draft Version 9. (Cited on page 2.)

[35] IETF. the internet engineering task force. `http://www.ietf.org/`. (Cited on page 1.)

[36] Antoine Joux. A one round protocol for tripartite diffie-hellman. In *Algorithmic Number Theory – ANTS IV*, volume 1838 of *LNCS*, pages 385–394. Springer-Verlag, 2000. (Cited on page 5.)

[37] Eike Kiltz. Chosen-ciphertext secure identity-based encryption in the standard model with short ciphertexts. Cryptology ePrint Archive, Report 2006/122, 2006. `http://eprint.iacr.org/`. (Cited on page 3.)

[38] Eike Kiltz. Chosen-ciphertext security from tag-based encryption. In Shai Halevi and Tal Rabin, editors, *TCC 2006: 3rd Theory of Cryptography Conference*, volume 3876 of *Lecture Notes in Computer Science*, pages 581–600. Springer-Verlag, Berlin, Germany, March 2006. (Cited on page 2, 7, 9.)

[39] Eike Kiltz. On the limitations of the spread of an IBE-to-PKE transformation. In Moti Yung, Yevgeniy Dodis, Aggelos Kiayias, and Tal Malkin, editors, *PKC 2006: 9th International Conference on Theory and Practice of Public Key Cryptography*, volume 3958 of *Lecture Notes in Computer Science*, pages 274–289. Springer-Verlag, Berlin, Germany, April 2006. (Cited on page 2.)

[40] Eike Kiltz and David Galindo. Direct chosen-ciphertext secure identity-based key encapsulation without random oracles. In *ACISP 2006*, volume 4058 of *Lecture Notes in Computer Science*. Springer-Verlag, 2006. (Cited on page 3, 8.)

[41] Eike Kiltz and Yevgeniy Vahlis. CCA2 secure IBE: Standard model efficiency through authenticated symmetric encryption. In Tal Malkin, editor, *Topics in Cryptology – CT-RSA 2008*, Lecture Notes in Computer Science, pages 221–238. Springer-Verlag, Berlin, Germany, April 2008. (Cited on page 3.)

[42] Kaoru Kurosawa and Yvo Desmedt. A new paradigm of hybrid encryption scheme. In Matthew Franklin, editor, *Advances in Cryptology – CRYPTO 2004*, volume 3152 of *Lecture Notes in Computer Science*, pages 426–442. Springer-Verlag, Berlin, Germany, August 2004. (Cited on page 5.)

[43] David Naccache. Secure and practical identity-based encryption. *IET Information Security*, 1(1):59–64, 2007. (Cited on page 3, 9.)

[44] Moni Naor and Omer Reingold. Number-theoretic constructions of efficient pseudo-random functions. In *38th Annual Symposium on Foundations of Computer Science*, pages 458–467. IEEE Computer Society Press, October 1997. (Cited on page 5.)

[45] Moni Naor and Omer Reingold. On the construction of pseudorandom permutations: Luby-Rackoff revisited. *Journal of Cryptology*, 12(1):29–66, 1999. (Cited on page 9.)

[46] Moni Naor and Moti Yung. Universal one-way hash functions and their cryptographic applications. In *21st Annual ACM Symposium on Theory of Computing*, pages 33–43. ACM Press, May 1989. (Cited on page 5.)

[47] Duong Hieu Phan and David Pointcheval. About the security of ciphers (semantic security and pseudo-random permutations). In Helena Handschuh and Anwar Hasan, editors, *SAC 2004: 11th Annual International Workshop on Selected Areas in Cryptography*, volume 3357 of *Lecture Notes in Computer Science*, pages 182–197. Springer-Verlag, Berlin, Germany, August 2004. (Cited on page 8.)

[48] Charles Rackoff and Daniel R. Simon. Non-interactive zero-knowledge proof of knowledge and chosen ciphertext attack. In Joan Feigenbaum, editor, *Advances in Cryptology – CRYPTO'91*, volume 576 of *Lecture Notes in Computer Science*, pages 433–444. Springer-Verlag, Berlin, Germany, August 1992. (Cited on page 1, 4.)

[49] John Rompel. One-way functions are necessary and sufficient for secure signatures. In *22nd Annual ACM Symposium on Theory of Computing*, pages 387–394. ACM Press, May 1990. (Cited on page 5.)

[50] R. Sakai, K. Ohgishi, and M. Kasahara. Cryptosystems based on pairing over elliptic curve (in japanese). In *Proceedings of the Symposium on Cryptography and Information Security — SCIS 2001*, jan 2001. (Cited on page 1.)

[51] Palash Sarkar and Sanjit Chatterjee. Construction of a hybrid (hierarchical) identity-based encryption protocol secure against adaptive attacks. Cryptology ePrint Archive, Report 2006/362, 2006. `http://eprint.iacr.org/`. (Cited on page 8.)

[52] Palash Sarkar and Sanjit Chatterjee. Construction of a hybrid hibe protocol secure against adaptive attacks. In Willy Susilo, Joseph K. Liu, and Yi Mu, editors, *Provable Security, First International Conference, ProvSec 2007*, volume 4784 of *Lecture Notes in Computer Science*, pages 51–67, Wollongong, Australia, November 1–2, 2007. Springer-Verlag, Berlin, Germany. (Cited on page 3.)

[53] Michael Scott. Faster pairings using an elliptic curve with an efficient endomorphism. Cryptology ePrint Archive, Report 2005/252, 2005. `http://eprint.iacr.org/`. (Cited on page 21.)

[54] Adi Shamir. Identity-based cryptosystems and signature schemes. In G. R. Blakley and David Chaum, editors, *Advances in Cryptology – CRYPTO'84*, volume 196 of *Lecture Notes in Computer Science*, pages 47–53. Springer-Verlag, Berlin, Germany, August 1985. (Cited on page 1, 4.)

[55] Victor Shoup. A proposal for an ISO standard for public key encryption (version 2.1). manuscript, 2001. Available on `http://shoup.net/papers/`. (Cited on page 2.)

[56] Victor Shoup. Sequences of games: a tool for taming complexity in security proofs. manuscript, 2004. Available from `http://shoup.net/papers/`. (Cited on page 9.)

[57] Gilbert S. Vernam. Cipher printing telegraph systems for secret wire and radio telegraphic communications. *Journal of the IEEE*, 55:109–115, 1926. (Cited on page 8.)

[58] Brent R. Waters. Efficient identity-based encryption without random oracles. In Ronald Cramer, editor, *Advances in Cryptology – EUROCRYPT 2005*, volume 3494 of *Lecture Notes in Computer Science*, pages 114–127. Springer-Verlag, Berlin, Germany, May 2005. (Cited on page 2, 3, 6, 9, 12, 19, 20, 21.)