Anonymous Hierarchical Identity-Based Encryption (Without Random Oracles)

Xavier Boyen[1] and Brent Waters[2]

[1] Voltage Inc., Palo Alto — `xb@boyen.org`
[2] SRI International — `bwaters@csl.sri.com`

Abstract. We present an identity-based cryptosystem that features fully anonymous ciphertexts and hierarchical key delegation. We give a proof of security in the standard model, based on the mild Decision Linear complexity assumption in bilinear groups. The system is efficient and practical, with small ciphertexts of size linear in the depth of the hierarchy. Applications include search on encrypted data, fully private communication, *etc.*

Our results resolve two open problems pertaining to anonymous identity-based encryption, our scheme being the first to offer provable anonymity in the standard model, in addition to being the first to realize fully anonymous HIBE at all levels in the hierarchy.

1 Introduction

The cryptographic primitive of Identity-Based Encryption (IBE) allows a sender to encrypt a message for a receiver using only the receiver's identity as a public key. Recently, there has been interest in "anonymous" identity-based encryption systems, where the ciphertext does not leak the identity of the recipient. In addition to their obvious privacy benefits, anonymous IBE systems can be leveraged to construct Public key Encryption with Keyword Search (PEKS) schemes, as was first observed by Boneh *et al.* [9] and later formalized by Abdalla *et al.* [1]. Roughly speaking, PEKS is a form of public key encryption that allows an encryptor to make a document serarchable by keywords, and where the capabilities to search on particular keywords are delegated by a central authority. Anonymous HIBE further enables sophisticated access policies for PEKS and ID-based PEKS.

Prior to this paper, the only IBE system known to be inherently anonymous was that of Boneh and Franklin [10]. Although they did not state it explicitly, the anonymity of their scheme followed readily from their proof of semantic security. This was noticed by Boyen [12], who gave an ID-based signcryption with a formalization of sender and recipient anonymity. One drawback of the Boneh-Franklin IBE paradigm is that its security proofs are set in the random oracle model. More recently, a number of IBE schemes [14, 5, 6, 31, 16, 26] have been proven secure outside of the random oracle model, but none of these schemes is anonymous. In particular, in the efficient schemes of Boneh and Boyen [5] and Waters [31], the identity is deterministically encoded in a simple manner within the exponent of an element of the bilinear group $\mathbb{G}$. When these schemes are implemented using a "symmetric" bilinear pairing $\mathbf{e} : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$, it becomes trivial to test whether a given ciphertext was encrypted for a candidate identity.

A tempting workaround to this problem is to use an "asymmetric" pairing $\mathbf{e} : \mathbb{G} \times \hat{\mathbb{G}} \to \mathbb{G}_T$ in the schemes that allow it, such as Boneh and Boyen's "BB_1" and "BB_2", and Waters' by extension. In those schemes, and under the additional assumption that Decision Diffie-Hellman is hard in $\mathbb{G}$, one may prevent the use of the pairing as a direct test of whether a ciphertext is for a particular identity. Unfortunately, turning this observation into a formal security reduction would at the very least require making a strong assumption that is patently false in bilinear groups with symmetric pairings, and the approach would still fail to generalize to hierarchical IBE for fundamental reasons that are discussed later. Ideally, one would like a scheme that works indifferently with symmetric and asymmetric pairings, and generalizes to hierarchical identities.

The first anonymous IBE without random oracles was unveiled at the Rump Session of CRYPTO'05 by one of the authors, and is now described in Section 4. In a nutshell, the identity is split randomly into two blind components to prevent its recognition by the bilinear map, without making unduly strong assumptions. A second anonymous IBE without random oracles was recently proposed by Gentry [19], based on a different principle. In Gentry's scheme, the identity of a ciphertext cannot be tested because a crucial element of the ciphertext lives in the target group $\mathbb{G}_T$ rather than the bilinear group $\mathbb{G}$. Gentry's scheme is very efficient and has a number of advantages, but unfortunately relies on a strong complexity assumption and does not generalize to hierarchical IBE.

In spite of these recent achievements, creating an Anonymous Hierarchical IBE (A-HIBE) scheme has remained a challenge. Even if we avail ourselves of random oracles, there simply does not exist any known hierarchical identity-based encryption scheme which is also anonymous. In particular, the Gentry-Silverberg [20] HIBE scheme is not anonymous, despite the fact that it derives from the Boneh-Franklin IBE scheme, which is anonymous. The numerous applications to searching on encrypted data motivated Abdalla *et al.* [1], in their CRYPTO'05 paper, to ask for the creation of an Anonymous HIBE system, preferably without random oracles, as an important open research problem.

1.1 Our Results

Our contribution is twofold. First, we build a simple and efficient Anonymous IBE scheme, and give a proof of security without random oracles. Second, we generalize our construction into a fully Anonymous HIBE scheme (*i.e.*, anonymous at all levels in the hierarchy), again with a proof without random oracles. Our approach gives a very efficient system in the non-hierarchical case, and remains practical for the shallow hierarchies that are likely to be encountered in most applications. The security of our systems is based on Boneh's *et al.* [8] Decision Linear assumption, which is arguably one of the mildest useful complexity assumptions in the realm of bilinear groups.

At first sight, our construction bears a superficial resemblance to Boneh and Boyen's "BB_1" HIBE scheme [5, §4] — but with at least two big differences. First, we perform "linear splittings" on various portions of the ciphertext, to thwart the trial-and-error identity guessing to which other schemes fell prey. This idea gives us provable anonymity, even under symmetric pairings. Second, we use multiple parallel HIBE systems and re-randomize the keys between them upon each delegation. This is what lets us use the linear splitting technique at all levels of the hierarchy, but also poses a technical challenge in the security reduction which must now simulate multiple interacting HIBE systems *at once*. Solving this problem was the crucial step that gave us a hierarchy without destroying anonymity.

1.2 Related Work

The concept of identity-based encryption was first proposed by Shamir [28] two decades ago. However, it was not until much later that Boneh and Franklin [10] and Cocks [17] presented the first practical solutions. The Boneh-Franklin IBE scheme was based on groups with efficiently computable bilinear maps, while the Cocks scheme was proven secure under the quadratic residuosity problem, which relies on the hardness of factoring. The security of either scheme was only proven in the random oracle model.

Canetti, Halevi, and Katz [14] suggested a weaker security notion for IBE, known as selective identity or selective-ID, relative to which they were able to build an inefficient but secure IBE scheme without using random oracles. Boneh and Boyen [5] presented two very efficient IBE systems ("BB_1" and "BB_2") with selective-ID security proofs, also without random oracles. The same authors [6] then proposed a coding-theoretic extension to their "BB_1" scheme that allowed them to prove security for the full notion of adaptive identity or adaptive-ID security without random oracles, but the construction was impractical. Waters [31] then proposed a much simpler extension to "BB_1" also with an adaptive-ID security proof without random oracles; its efficiency was further improved in two recent independent papers, [16] and [26].

The notion of hierarchical identity-based encryption was first defined by Horwitz and Lynn [22], and a construction in the random oracle model given by Gentry and Silverberg [20]. Canetti, Halevi, and Katz [14] give the first HIBE with a (selective-ID) security proof without random oracles, but that is not efficient. The first efficient HIBE scheme to be provably secure without random oracles is the "BB_1" system of Boneh and Boyen; further improvements include the HIBE scheme by Boneh, Boyen, and Goh [7], which features shorter ciphertexts and private keys.

Nominally adaptive-ID secure HIBE schemes have been proposed, although all constructions known to date [20, 31, 16, 26] are depth-limited because they suffer from an exponential security degradation with the depth of the hierarchy. Qualitatively, this is no different than taking an HIBE scheme with tight selective-ID security, such as BB_1 or BBG, and using one of the generic transformations from [5, §7] to make it adaptive-ID secure. Quantitatively, the rate of decay will differ between those approaches, which means that the number of useful hierarchy levels will evolve similarly but not identically in function of the chosen group size and the desired security bit strength. Accordingly, it remains an important open problem in identity-based cryptography to devise an adaptive-ID secure HIBE scheme whose security degrades at most polynomially with the depth of the hierarchy, under reasonable assumptions. (In this paper, we mostly leave aside this issue of adaptive-ID security for HIBE.)

Encrypted search was studied by Song, Wagner, and Perrig [30], who presented the first scheme for searching on encrypted data. Their scheme is in the symmetric-key setting where the same party that encrypted the data would generate the keyword search capabilities. Boneh *et al.* [9] introduced Public Key Encryption with Keyword Search (PEKS), where any party with access to a public key could make an encrypted document that was searchable by keyword; they realized their construction by applying the Boneh-Franklin IBE scheme. Abdalla *et al.* [1] recently formalized the notion of Anonymous IBE and its relationship to PEKS. Additionally, they formalized the notion of Anonymous HIBE and mentioned different applications for it. Using the GS system as a starting point, they also gave an HIBE scheme that was anonymous at the first level, in the random oracle model. Another view of Anonymous IBE is as a combination of identity-based encryption with the property of key privacy, which was introduced by Bellare *et al.* [4].

1.3 Applications

In this section we discuss various applications of our fully anonymous HIBE system. The main applications can be split into several broad categories.

Fully Private Communication. The first compelling application of anonymous IBE is for fully private communication. Bellare *et al.* [4] argue that public key encryption systems that have the "key privacy" property can be used for anonymous communication: for example, if one wishes to hide the identity of a recipient one can encrypt a ciphertext with an anonymous IBE system and post it on a public bulletin board. By the anonymity property, the ciphertext will betray neither sender nor recipient identity, and since the bulletin board is public, this method will also be resistant to traffic analysis. To compound this notion of key privacy, identity-based encryption is particularly suited for untraceable anonymous communication, since, contrarily to public-key infrastructures, the sender does not even need to query a directory for the public key of the recipient. For this reason, anonymous IBE provides a very convincing solution to the problem of secure anonymous communication, as it makes it harder to conduct traffic analysis attack on directory lookups.

Search on Encrypted Data. The second main application of anonymous (H)IBE is for encrypted search. As mentioned earlier, anonymous IBE and HIBE give several application in the Public-key Encryption with Keyword Search (PEKS) domain, proposed by Boneh *et al.* [9], and further discussed by Abdalla *et al.* [1]. As a simple example of real-world application of our scheme, PEKS is a useful primitive for building secure audit logs [32, 18]. Furthermore, one can leverage the hierarchical identities in our anonymous HIBE in several interesting ways. For example, we can use a two-level anonymous HIBE scheme where the first level is an identity and the second level is a keyword. This gives us the first implementation of the Identity-Based Encryption with Keyword Search (IBEKS) primitive asked for in [1]. With this primitive, someone with the private key for an identity can delegate out search capabilities for encryptions to their identity, without requiring a central authority to act as the delegator. Conversely, by using certain keywords such as "Top Secret" at the first level of the hierarchy, it is possible to broadcast innocent-looking ciphertexts that require a certain clearance to decrypt, without even hinting at the fact that their payload might be valuable. We can create more refined search capabilities with a deeper hierarchy.

As the last applications we mention, forward-secure public-key encryption [14] and forward-secure HIBE [33] are not hard to construct from HIBE systems with certain algebraic properties [7]. Without going into details, we mention that we can implement Anonymous fs-HIBE with our scheme by embedding a time component within the hierarchy, while preserving the anonymity property.

2 Background

Recall that a pairing is an efficiently computable [25], non-degenerate function, $\mathbf{e} : \mathbb{G} \times \hat{\mathbb{G}} \to \mathbb{G}_T$, with the bilinearity property that $\mathbf{e}(g^r, \hat{g}^s) = \mathbf{e}(g, \hat{g})^{r\,s}$. Here, $\mathbb{G}$, $\hat{\mathbb{G}}$, and $\mathbb{G}_T$ are all multiplicative groups of prime order p, respectively generated by g, $\hat{g}$, and $\mathbf{e}(g, \hat{g})$. It is *asymmetric* if $\mathbb{G} \neq \hat{\mathbb{G}}$.

We call *bilinear instance* a tuple $\mathbf{G} = [p, \mathbb{G}, \hat{\mathbb{G}}, \mathbb{G}_T, g, \hat{g}, \mathbf{e}]$. We assume an efficient generation procedure that on input a security parameter $\Sigma \in \mathbb{N}$ outputs $\mathbf{G} \xleftarrow{\$} \mathrm{Gen}(1^{\Sigma})$ where $\log_2(p) = \Theta(\Sigma)$. We write $\mathbb{Z}_p = \mathbb{Z}/p\mathbb{Z}$ for the set of residues $\bmod p$ and $\mathbb{Z}_p^\times = \mathbb{Z}_p \setminus \{0\}$ for its multiplicative group.

2.1 Assumptions

Since bilinear groups first appeared in cryptography half a decade ago [23], several years after their first use in cryptanalysis [24], bilinear maps or pairings have been used in a large variety of ways under many different complexity assumptions. Some of them are very strong; others are weaker. Informally, we say that an assumption is *mild* if it is tautological in the generic group model [29], and also "efficiently falsifiable" [27] in the sense that its problem instances are stated non-interactively and concisely (*e.g.*, independently of the number of adversarial queries or such large quantity). Most IBE and HIBE schemes mentioned in Introduction (except "BB_2" and the Factoring-based system by Cocks) are based on *mild* bilinear complexity assumptions, such as BDH [23, 10] and Linear [8]. In this paper, our goal is to rely only on mild assumptions.

Decision BDH: The Bilinear DH assumption was first used by Joux [23], and gained popularity for its role in the Boneh-Franklin IBE system [10]. The decisional assumption posits the hardness of the D-BDH problem, which we state in asymmetric bilinear groups as:

Given a tuple $[g, g^{z_1}, g^{z_3}, \hat{g}, \hat{g}^{z_1}, \hat{g}^{z_2}, Z] \in \mathbb{G}^3 \times \hat{\mathbb{G}}^3 \times \mathbb{G}_T$ for random exponents $[z_1, z_2, z_3] \in (\mathbb{Z}_p)^3$, decide whether $Z = \mathbf{e}(g, \hat{g})^{z_1\, z_2\, z_3}$.

Decision Linear: The Linear assumption was first proposed by Boneh, Boyen, and Shacham for group signatures [8]. Its decisional form posits the hardness of the D-Linear problem, which can be stated in asymmetric bilinear groups as follows:

Given a tuple $[g, g^{z_1}, g^{z_2}, g^{z_1\, z_3}, g^{z_2\, z_4}, \hat{g}, \hat{g}^{z_1}, \hat{g}^{z_2}, Z] \in \mathbb{G}^5 \times \hat{\mathbb{G}}^3 \times \mathbb{G}$ for random $[z_1, z_2, z_3, z_4] \in (\mathbb{Z}_p)^4$, decide whether $Z = g^{z_3+z_4}$.

We remark that the elements $\hat{g}, \hat{g}^{z_1}, \hat{g}^{z_2} \in \hat{\mathbb{G}}^3$ were not explicitly included in Boneh's *et al.* original formulation.

"Hard" means algorithmically non-solvable with probability $1/2 + \Omega(\text{poly}(\Sigma)^{-1})$ in time $\mathcal{O}(\text{poly}(\Sigma))$ for "bilinear instances" $[p, \mathbb{G}, \hat{\mathbb{G}}, \mathbb{G}_T, g, \hat{g}, \mathbf{e}] \xleftarrow{\$} \text{Gen}(1^{\Sigma})$ that are generated at random using an efficient algorithm, as $\Sigma \to +\infty$.

These assumptions allow but not require the groups $\mathbb{G}$ and $\hat{\mathbb{G}}$ to be distinct, and similarly we make no representation one way or the other regarding the existence of computable homomorphisms between $\mathbb{G}$ and $\hat{\mathbb{G}}$, in either direction. This is the most general formulation. It has two main benefits: (1) since it comes with fewer restrictions, it is potentially more robust and increases our confidence in the assumptions we make; and (2) it gives us the flexibility to implement the bilinear pairing on a broad variety of algebraic curves with attractive computational characteristics [2], whereas symmetric pairings tend to be confined to supersingular curves, to name this one distinction.

Note that if we let $\mathbb{G} = \hat{\mathbb{G}}$ and $g = \hat{g}$, our assumptions regain their familiar "symmetric" forms:

Given $[g, g^{z_1}, g^{z_2}, g^{z_3}, Z] \in \mathbb{G}^4 \times \mathbb{G}_T$ for random $[z_1, z_2, z_3] \in (\mathbb{Z}_p)^3$, decide whether $Z = \mathbf{e}(g, g)^{z_1\, z_2\, z_3}$.
Given $[g, g^{z_1}, g^{z_2}, g^{z_1\, z_3}, g^{z_2\, z_4}, Z] \in \mathbb{G}^5 \times \mathbb{G}$ for random $[z_1, z_2, z_3, z_4] \in (\mathbb{Z}_p)^4$, decide if $Z = g^{z_3+z_4}$.

As a rule of thumb, the remainder of this paper may be read in the context of symmetric pairings, simply by dropping all "hats" ($\hat{}$) in the notation. Also note that D-Linear trivially implies D-BDH.

2.2 Models

We briefly precise the security notions that are implied by the concept of Anonymous IBE or HIBE. We omit the formal definitions, which may be found in the literature [10, 1].

Confidentiality: This is the usual security notion of semantic security for encryption. It means that no non-trivial information about the message can be feasibly gleaned from the ciphertext.

Anonymity: Recipient anonymity is the property that the adversary be unable to distinguish the encryption of a chosen message for a first chosen identity from the encryption of the same message for a second chosen identity. Equivalently, the adversary must be unable to decide whether a ciphertext was encrypted for a chosen identity, or for a random identity.

3 Intuition

Before we present our scheme we first explain why it is difficult to implement anonymous IBE without random oracles, as well as any form of anonymous HIBE even in the random oracle model. We then give some intuition behind our solution.

3.1 The Difficulty

Recall that in the basic Boneh-Franklin IBE system [10], an encryption of a message Msg to some identity Id, takes the following form,

$$\mathsf{CT} = [\, C_1,\ C_2 \,] = [\, g^r,\ \mathbf{e}(\mathcal{H}(\mathsf{Id}), Q)^r \, \mathsf{Msg} \,] \ \in\ \mathbb{G} \times \mathbb{G}_T \ ,$$

where $\mathcal{H}$ is a random oracle, r is a random exponent, and g and Q are public system parameters. A crucial observation is that the one element of the ciphertext in the bilinear group $\mathbb{G}$, namely, g^r, is just a random element that gives no information about the identity of the recipient. The reason why only one element in $\mathbb{G}$ is needed is because private keys in the Boneh-Franklin scheme are deterministic — there will be no randomness in the private key to cancel out. Since the proof of semantic security is based on the fact that C_2 is indistinguishable from random without the private key for ID, it follows that the scheme is also anonymous since C_2 is the only part of the ciphertext on which the recipient identity has any bearing.

More recently, there have been a number of IBE schemes proven secure without random oracles, such as BTE from [14], BB_1 and BB_2 from [5], and Waters' [31]. However, in all these schemes the proof of security requires that randomness be injected into the private key generation. Since the private keys are randomized, some extra information is needed in the ciphertext in order to cancel out the randomness upon decryption. To illustrate, consider the encryption of a message Msg to an identity Id in the BB_1 Boneh-Boyen system,

$$\mathsf{CT} = [\, C_1,\ C_2,\ C_3 \,] = \left[\, g^r,\ (g_1^{\mathsf{Id}}\, g_3)^r,\ \mathbf{e}(g_1, \hat{g}_2)^r \, \mathsf{Msg} \,\right] \ \in\ \mathbb{G}^2 \times \mathbb{G}_T \ ,$$

where r is chosen by the encryptor and g, g_1, g_3, and $\mathbf{e}(g_1, \hat{g}_2)$ are public system parameters. Notice, there are now two elements in $\mathbb{G}$, and between them there is enough redundancy to determine whether a ciphertext was intended for a given identity Id, simply by testing whether the tuple $[g, g_1^{\mathsf{Id}}\, g_3, C_1, C_2]$ is Diffie-Hellman, using the bilinear map,

$$\mathbf{e}(C_1, \hat{g}_1^{\mathsf{Id}}\, \hat{g}_3) \stackrel{?}{=} \mathbf{e}(C_2, \hat{g}) \ .$$

We see that the extra ciphertext components which are seemingly necessary in IBE schemes without random oracles, in fact contribute to leaking the identity of the intended recipient of a ciphertext.

A similar argument can be made for why none of the existing HIBE schemes is anonymous, even though some of them use random oracles. Indeed, all known HIBE schemes, including the Gentry-Silverberg system in the random oracle model, rely on randomization in order to properly delegate private keys down the hierarchy in a collusion-resistant manner. Since the randomization is performed not just by the master authority, but by anyone who has the power to delegate a key, the elements needed for it are distributed as part of the public parameters. Because of this, we end up in the same situation as above, where the extra components needed to either perform or cancel the randomization will also provide a test for the addressee's identity.

Since having randomized keys seems to be fundamental to designing (H)IBE systems without random oracles, we aim to design a system where the necessary extra information will be hidden to a computationally bounded adversary. Thus, even though we cannot prevent the ciphertext from containing information about the recipient, we can design our system such that this information cannot be easily tested from the public parameters and ciphertext alone.

3.2 Our Approach

As mentioned in the introduction, we can prevent a single-level identity to be testable by performing some sort of blinding, by splitting the identity into two randomized complementary components. Indeed, building a "flat" anonymous IBE system turns out to be reasonably straightforward using our linear splitting technique to hide the recipient identity behind some randomization.

Complications arise when one tries to support hierarchical key generation. In a nutshell, to prevent collusion attacks in HIBE, "parents" must independently re-randomize the private keys they give to their "children". In all known HIBE schemes, re-randomization is enabled by listing a number of supplemental components in the public system parameters. Why this breaks anonymity is because the same mechanism that allows private keys to be publicly re-randomized, also allows ciphertexts to be publicly tested for recipient identities. Random oracles offer no protection against this.

To circumvent this obstable, we need to make the re-randomization elements non-public, and tie them to each individual private key. In practical terms, this means that private keys must convey extra components (although not too many). The real difficulty is that each set of re-randomization components constitutes a full-fledged HIBE in its own right, which must be simulated together with its peers in the security proof (their number grows linearly with the maximal depth). Because these systems are not independent but interact with each other, we are left with the task of simulating multiple HIBE subsystems that are globally constrained by a set of linear relations. A novelty of our proof technique is a method to endow the simulator with enough degrees of freedom to reduce a system of unknown keys to a single instance of the presumed hard problem.

A notable feature of our construction is that it can be implemented using all known instantiations of the bilinear pairing (whether symmetric or asymmetric, with our without a computable or invertible homomorphism, *etc.*). To cover all grounds, we first describe a "flat" anonymous IBE using the symmetric notation, for ease of exposition, and then move to the full HIBE using the general asymmetric notation without assuming any homomorphism, for maximum generality.

4 A Primer : Anonymous IBE

We start by describing an Anonymous IBE scheme that is semantically secure against selective-ID chosen plaintext attacks. This construction will illustrate our basic technique of "splitting" the bilinear group elements into two pieces to protect against the attacks described in the previous section.

For simplicity, and also to show that we get anonymity even when using symmetric pairings, we describe the IBE system (and the IBE system only) in the special case where $\mathbb{G} = \hat{\mathbb{G}}$:

Setup The setup algorithm chooses a random generator $g \in \mathbb{G}$, random group elements $g_0, g_1 \in \mathbb{G}$, and random exponents $\omega, t_1, t_2, t_3, t_4 \in \mathbb{Z}_p$. It keeps these exponents as the master key, Msk. The corresponding system parameters are published as:

$$\mathsf{Pub} \leftarrow \left[\ \Omega = \mathbf{e}(g,g)^{t_1 t_2 \omega},\ g,\ g_0,\ g_1,\ v_1 = g^{t_1},\ v_2 = g^{t_2},\ v_3 = g^{t_3},\ v_4 = g^{t_4}\ \right]$$

Extract$(\mathsf{Msk}, \mathsf{Id})$ To issue a private key for identity Id, the key extraction authority chooses two random exponents $r_1, r_2 \in \mathbb{Z}_p$, and computes the private key as: $\mathsf{Pvk}_{\mathsf{Id}} = [d_0, d_1, d_2, d_3, d_4] \leftarrow$

$$\left[\ g^{r_1 t_1 t_2 + r_2 t_3 t_4},\ g^{-\omega t_2}(g_0 g_1^{\mathsf{Id}})^{-r_1 t_2},\ g^{-\omega t_1}(g_0 g_1^{\mathsf{Id}})^{-r_1 t_1},\ (g_0 g_1^{\mathsf{Id}})^{-r_2 t_4},\ (g_0 g_1^{\mathsf{Id}})^{-r_2 t_3}\ \right]$$

Encrypt$(\mathsf{Pub}, \mathsf{Id}, M)$ Encrypting a message $\mathsf{Msg} \in \mathbb{G}_T$ for an identity $\mathsf{Id} \in \mathbb{Z}_p^\times$ works as follows. The algorithm chooses random exponents $s, s_1, s_2 \in \mathbb{Z}_p$, and creates the ciphertext as:

$$\mathsf{CT}\ = [\ C', C_0, C_1, C_2, C_3, C_4\ ] \leftarrow \left[\ \Omega^s M,\ (g_0 g_1^{\mathsf{Id}})^s,\ v_1^{s-s_1},\ v_2^{s_1},\ v_3^{s-s_2},\ v_4^{s_2}\ \right]$$

Decrypt$(\mathsf{Pvk}_{\mathsf{Id}}, C)$ The decryption algorithm attempts to decrypt a ciphertext CT by computing:

$$C'\,\mathbf{e}(C_0, d_0)\,\mathbf{e}(C_1, d_1)\,\mathbf{e}(C_2, d_2)\,\mathbf{e}(C_3, d_3)\,\mathbf{e}(C_4, d_4) = \mathsf{Msg}\ .$$

Proving Security. We prove security using a hybrid experiment.

Let $[C', C_0, C_1, C_2, C_3, C_4]$ denote the challenge ciphertext given to the adversary during a real attack. Additionally, let R be a random element of $\mathbb{G}_T$, and R', R'' be random elements of $\mathbb{G}$. We define the following hybrid games which differ on what challenge ciphertext is given by the simulator to the adversary:

Γ_0 : The challenge ciphertext is $\mathsf{CT}_0 = [C', C_0, C_1, C_2, C_3, C_4]$.
Γ_1 : The challenge ciphertext is $\mathsf{CT}_1 = [R, C_0, C_1, C_2, C_3, C_4]$.
Γ_2 : The challenge ciphertext is $\mathsf{CT}_2 = [R, C_0, R', C_2, C_3, C_4]$.
Γ_3 : The challenge ciphertext is $\mathsf{CT}_3 = [R, C_0, R', C_2, R'', C_4]$.

We remark that the challenge ciphertext in Γ_3 leaks no information about the identity since it is composed of six random group elements, whereas in Γ_0 the challenge is well formed. We show that the transitions from Γ_0 to Γ_1 to Γ_2 to Γ_3 are all computationally indistinguishable.

Lemma 1 (semantic security). *Under the (t, ϵ)-Decision BDH assumption, there is no adversary running in time t that distinguishes between the games Γ_0 and Γ_1 with advantage greater than ϵ.*

Proof. The proof from this lemma essentially follows from the security of the Boneh-Boyen selective-ID scheme. Suppose there is an adversary that can distinguish between game Γ_0 and Γ_1 with advantage ϵ. Then we build a simulator that plays the Decision BDH game with advantage ϵ.

The simulator receives a D-BDH challenge $[g, g^{z_1}, g^{z_2}, g^{z_3}, Z]$ where Z is either $\mathbf{e}(g,g)^{z_1 z_2 z_3}$ or a random element of $\mathbb{G}_T$ with equal probability. The game proceeds as follows:

⋄ *Init:* The adversary announces the identity Id^* it wants to be challenged upon.

⋄ *Setup:* The simulator chooses random exponents $t_1, t_2, t_3, t_4, y \in \mathbb{Z}_p$. It retains the generator g, and sets $g_0 = (g^{z_1})^{-\mathsf{Id}^*} g^y$ and $g_1 = g^{z_1}$. The public parameters are published as:

$$\mathsf{Pub} \leftarrow \left[\ \Omega = \mathbf{e}(g^{z_1}, g^{z_2})^{t_1 t_2},\ g,\ g_0,\ g_1,\ v_1 = g^{t_1},\ v_2 = g^{t_2},\ v_3 = g^{t_3},\ v_4 = g^{t_4}\ \right] .$$

Note that this implies that $\omega = z_1 z_2$.

⋄ *Phase 1:* Suppose the adversary requests a key for identity $\mathsf{Id} \neq \mathsf{Id}^*$. The simulator picks random exponents $r_1, r_2 \in \mathbb{Z}_p$, and issues a private key as: $\mathsf{Pvk}_{\mathsf{Id}} = [d_0, d_1, d_2, d_3, d_4] \leftarrow$

$$\left[\begin{array}{c} (g^{z_2})^{\frac{-1}{\mathsf{Id}-\mathsf{Id}^*}} g^{r_1} g^{r_2 t_3 t_4},\ ((g^{z_2})^{\frac{y}{\mathsf{Id}-\mathsf{Id}^*}} (g_0 g_1^{\mathsf{Id}})^{r_1})^{-t_2},\ ((g^{z_2})^{\frac{y}{\mathsf{Id}-\mathsf{Id}^*}} (g_0 g_1^{\mathsf{Id}})^{r_1})^{-t_1}, \\ (g_0 g_1^{\mathsf{Id}})^{-r_2 t_4},\ (g_0 g_1^{\mathsf{Id}})^{-r_2 t_3} \end{array} \right] .$$

This is a well formed secret key for random exponents $\tilde{r_1} = r_1 - z_2/(\mathsf{Id} - \mathsf{Id}^*)$ and $\tilde{r_2} = r_2$.

⋄ *Challenge:* Upon receiving a message Msg from the adversary, the simulator chooses $s_1, s_2 \in \mathbb{Z}_p$, and outputs the challenge ciphertext as:

$$\mathsf{CT}\ = [\,C', C_0, C_1, C_2, C_3, C_4\,] \leftarrow \left[\begin{array}{c} Z^{-t_1 t_2}\, M,\ (g^{z_3})^y,\ (g^{z_3})^{t_1} g^{-s_1 t_1},\ g^{s_1 t_2}, \\ (g^{z_3})^{t_3} g^{-s_2 t_3},\ g^{s_2 t_4} \end{array} \right] .$$

We can let $s = z_3$ and see that if $Z = \mathbf{e}(g,g)^{z_1 z_2 z_3}$ the simulator is playing game Γ_0 with the adversary, otherwise the simulator is playing game Γ_1 with the adversary.

⋄ *Phase 2:* The simulator answers the queries in the same way as Phase 1.

⋄ *Guess:* The simulator outputs a guess γ, which the simulator forwards as its own guess for the D-BDH game.

Since the simulator plays game Γ_0 if and only the given D-BDH instance was well formed, the simulator's advantage in the D-BDH game is exactly ϵ.

Lemma 2 (anonymity, part 1). *Under the (t,ϵ)-Decision linear assumption, no adversary that runs in time t can distinguish between the games Γ_1 and Γ_2 with advantage greater than ϵ.*

Proof. Suppose the existence of an adversary $\mathcal{A}$ that distinguishes between the two games with advantage ϵ. Then we construct a simulator that wins the Decision Linear game as follows.

The simulator takes in a D-Linear instance $[g, g^{z_1}, g^{z_2}, g^{z_1 z_3}, g^{z_2 z_4}, Z]$, where Z is either $g^{z_3+z_4}$ or random in $\mathbb{G}$ with equal probability. For convenience, we rewrite this as $[g, g^{z_1}, g^{z_2}, g^{z_1 z_3}, Y, g^s]$ for s such that $g^s = Z$, and consider the task of deciding whether $Y = g^{z_2(s-z_3)}$ which is equivalent. The simulator plays the game in the following stages.

⋄ *Init:* The adversary $\mathcal{A}$ gives the simulator the challenge identity Id^*.

⋄ *Setup:* The simulator first chooses random exponents $\alpha, y, t_3, t_4, \omega$. It lets g in the simulation be as in the instance, and sets $v_1 = g^{z_2}$ and $v_2 = g^{z_1}$. The public key is published as:

$$\mathsf{Pub} \leftarrow \left[\begin{array}{c} \Omega = \mathbf{e}(g^{z_1}, g^{z_2})^{\omega},\ g,\ g_0 = (g^{z_2})^{-\mathsf{Id}^* \alpha} g^{y},\ g_1 = (g^{z_2})^{\alpha}, \\ v_1 = (g^{z_2}),\ v_2 = (g^{z_1}),\ v_3 = g^{t_3},\ v_4 = g^{t_4} \end{array} \right] .$$

If we pose $t_1 = z_2$ and $t_2 = z_1$, we note that the public key is distributed as in the real scheme.

⋄ *Phase 1:* To answer a private key extraction query for an identity $\mathsf{Id} \neq \mathsf{Id}^*$, the simulator chooses random exponents $r_1, r_2 \in \mathbb{Z}_p$, and outputs:

$$\mathsf{Pvk}_{\mathsf{Id}} \leftarrow \left[\begin{array}{c} (g^{z_1})^{r_1} g^{r_2 t_3 t_4},\ (g^{z_1})^{-\omega - \alpha(\mathsf{Id}-\mathsf{Id}^*) r_1},\ (g^{z_2})^{-\omega - \alpha(\mathsf{Id}-\mathsf{Id}^*) r_1}, \\ (g^{z_1})^{\frac{-r_1 y}{t_3}} (g_0 g_1^{\mathsf{Id}})^{-r_2 t_4},\ (g^{z_1})^{\frac{-r_1 y}{t_4}} (g_0 g_1^{\mathsf{Id}})^{-r_2 t_3} \end{array} \right] .$$

If, instead of r_1 and r_2, we consider this pair of uniform random exponents,

$$\tilde{r_1} = \frac{r_1 \alpha(\mathsf{Id} - \mathsf{Id}^*)}{\alpha(\mathsf{Id} - \mathsf{Id}^*) z_2 + y} , \qquad \tilde{r_2} = r_2 + \frac{y z_1 r_1}{(t_3 t_4)(\alpha(\mathsf{Id} - \mathsf{Id}^*) z_2 + y)} ,$$

then we see that the private key is well formed, since it can be rewritten as:

$$\left[\ g^{\tilde{r_1} t_1 t_2 + \tilde{r_2} t_3 t_4},\ g^{-\omega t_2} (g_0 g_1^{\mathsf{Id}})^{-\tilde{r_1} t_2},\ g^{-\omega t_1} (g_0 g_1^{\mathsf{Id}})^{-\tilde{r_1} t_1},\ (g_0 g_1^{\mathsf{Id}})^{-\tilde{r_2} t_4},\ (g_0 g_1^{\mathsf{Id}})^{-\tilde{r_2} t_3}\ \right] .$$

⋄ *Challenge:* The simulator gets from the adversary a message M which it can discard, and responds with a challenge ciphertext for the identity Id^*. Pose

$s_1 = z_3$. To proceed, the simulator picks a random exponent $s_2 \in \mathbb{Z}_p$ and a random element $R \in \mathbb{G}_T$, and outputs the ciphertext as:

$$\mathsf{CT} = [\, C', C_0, C_1, C_2, C_3, C_4 \,] \leftarrow \big[\, R,\ (g^s)^y,\ Y,\ (g^{z_1 z_3}),\ (g^s)^{t_3} g^{-s_2 t_3},\ g^{s_2 t_4} \,\big] \ .$$

If $Y = g^{z_2(s-z_3)}$, *i.e.*, $g^s = Z = g^{z_3+z_4}$, then $C_1 = v_1^{s-s_1}$ and $C_2 = v_2^{s_1}$; all parts of the challenge but C' are thus well formed, and the simulator behaved as in game Γ_1. If instead Y is independent of z_1, z_2, s, s_1, s_2, which happens when Z is random, then the simulator responded as in game Γ_2.

⋄ *Phase 2:* The simulator answer the query in the same way as Phase 1.

⋄ *Output:* The adversary outputs a bit γ to guess which hybrid game the simulator has been playing. To conclude, the simulator forwards γ as its own answer in the Decision-Linear game.

By the simulation setup the advantage of the simulator will be exactly that of the adversary.

Lemma 3 (anonymity, part 2). *Under the (t, ϵ)-Decision linear assumption, no adversary that runs in time t can distinguish between the games Γ_2 and Γ_3 with advantage greater than ϵ.*

Proof. This argument follows almost identically to that of Lemma 2, except where the simulation is done over the parameters v_3 and v_4 in place of v_1 and v_2. The other difference is that the g^ω term that appeared in d_1, d_2 without interfering with the simulation, does not even appear in d_3, d_4.

5 The Scheme : Anonymous HIBE

We now describe our full Anonymous HIBE scheme without random oracles. Anonymity is provided by the splitting technique and hybrid proof introduced in the previous section. In addition, to thwart the multiple avenues for user collusion enabled by the hierarchy, the keys are re-randomized between all siblings and all children. Roughly speaking, this is done by using several parallel HIBE systems, which are recombined at random every time a new private key is issued. In the proof of security, this extra complication is handled by a "multi-secret simulator", that is able to simulate multiple interacting HIBE systems under a set of constraints. This is an information theoretic proof that sits on top of the hybrid argument, which is computational.

For the most part, we focus on security against selective-identity, chosen plaintext attacks, though we will briefly mention how to secure the scheme against adaptive-ID and CCA2 adversaries. Our (selective-ID) Anonymous HIBE scheme consists of the following algorithms:

Setup$(1^\Sigma, D)$ To generate the public system parameters and the corresponding master secret key, given a security parameter $\Sigma \in \mathbb{N}$ in unary, and the hierarchy's maximum depth $D \in \mathbb{N}$, the setup algorithm first generates a bilinear instance $\mathbf{G} = [p, \mathbb{G}, \hat{\mathbb{G}}, \mathbb{G}_T, g, \hat{g}, \mathbf{e}] \xleftarrow{\$} \mathrm{Gen}(1^\Sigma)$. Then:

1. Select $7 + 5\,D + D^2$ random integers modulo p (some of them non-zero):

$$\omega,\ [\ \alpha_n,\ \beta_n,\ [\ \theta_{n,\ell}\ ]_{\ell=0,\ldots,D}\ ]_{n=0,\ldots,1+D} \quad \in_\$ \mathbb{Z}_p^\times \times ((\mathbb{Z}_p^\times)^2 \times (\mathbb{Z}_p)^{1+D})^{2+D}\ ;$$

2. Publish $\mathbf{G}$ and the system parameters $\mathsf{Pub} \in \mathbb{G}_T \times \mathbb{G}^{2\,(1+D)\,(2+D)}$ as:

$$\Omega \ \leftarrow\ \mathbf{e}(g, \hat{g})^\omega \quad ,$$

$$[\,[\ a_{n,\ell},\ b_{n,\ell}\ ]_{\ell=0,\ldots,D}\ ]_{n=0,\ldots,1+D} \ \leftarrow\ \left[\ [\ g^{\alpha_n\,\theta_{n,\ell}},\ g^{\beta_n\,\theta_{n,\ell}}\ ]_{\ell=0,\ldots,D}\ \right]_{n=0,\ldots,1+D} \quad .$$

3. Retain the master secret key $\mathsf{Msk} \in \hat{\mathbb{G}}^{1+(3+D)\,(2+D)}$ as the elements:

$$\hat{w} \ \leftarrow\ \hat{g}^\omega \quad ,$$

$$[\ \hat{a}_n,\ \hat{b}_n,\ [\ \hat{y}_{n,\ell}\ ]_{\ell=0,\ldots,D}\ ]_{n=0,\ldots,1+D} \ \leftarrow\ \begin{bmatrix} \hat{g}^{\alpha_n},\ \hat{g}^{\beta_n}, \\ [\ \hat{g}^{\alpha_n\,\beta_n\,\theta_{n,\ell}}\ ]_{\ell=0,\ldots,D} \end{bmatrix}_{n=0,\ldots,1+D} \quad .$$

Extract$(\mathsf{Pub}, \mathsf{Msk}, \mathsf{Id})$ To extract from the master key Msk a private key for an identity $\mathsf{Id} = [I_0, I_1, \ldots, I_L] \in (\mathbb{Z}_p^\times)^{1+L}$ where $L \in \{1, \ldots, D\}$ and $I_0 = 1$:

1. Pick $6 + 5\,D + D^2$ random integers:

$$[\ \rho_n,\ [\ \rho_{n,m}\ ]_{m=0,\ldots,1+D}\ ]_{n=0,\ldots,1+D} \quad \in_\$ \ (\mathbb{Z}_p)^{(3+D)\,(2+D)}\ .$$

2. Output the private key $\mathsf{Pvk}_{\mathsf{Id}} \in \hat{\mathbb{G}}^{(5+3\,D-L)\,(3+D)}$ constituted of the following three subkeys (for decryption, re-randomization, and delegation):
   (a) $\mathsf{Pvk}_{\mathsf{Id}}^{\mathrm{decrypt}} = k_0, [k_{n,(a)}, k_{n,(b)}]_{n=0,\ldots,1+D}$

$$\leftarrow\ \hat{w} \prod_{n=0}^{1+D} \prod_{\ell=0}^{L} (\hat{y}_{n,\ell}^{I_\ell})^{\rho_n},\ \left[\ \hat{a}_n^{-\rho_n},\ \hat{b}_n^{-\rho_n}\ \right]_{n=0,\ldots,1+D} \quad .$$

   (b) $\mathsf{Pvk}_{\mathsf{Id}}^{\mathrm{rerand}} = [f_{m,0}, [f_{m,n,(a)}, f_{m,n,(b)}]_{n=0,\ldots,1+D}]_{m=0,\ldots,1+D}$

$$\leftarrow\ \left[\ \prod_{n=0}^{1+D} \prod_{\ell=0}^{L} (\hat{y}_{n,\ell}^{I_\ell})^{\rho_{n,m}},\ \left[\ \hat{a}_n^{-\rho_{n,m}},\ \hat{b}_n^{-\rho_{n,m}}\ \right]_{n=0,\ldots,1+D}\ \right]_{m=0,\ldots,1+D} \quad .$$

   (c) $\mathsf{Pvk}_{\mathsf{Id}}^{\mathrm{deleg}} = [h_\ell, [h_{m,\ell}]_{m=0,\ldots,1+D}]_{\ell=1+L,\ldots,D}$

$$\leftarrow\ \left[\ \prod_{n=0}^{1+D} (\hat{y}_{n,\ell})^{\rho_n},\ \left[\ \prod_{n=0}^{1+D} (\hat{y}_{n,\ell})^{\rho_{n,m}}\ \right]_{m=0,\ldots,1+D}\ \right]_{\ell=1+L,\ldots,D} \quad .$$

A more visual way to represent the private key is as a $(3 + D) \times (5 + 3\,D - L)$ array of elements in $\hat{\mathbb{G}}$, with $\mathsf{Pvk}_{\mathsf{Id}}^{\mathrm{decrypt}}$ as the upper left partial row, $\mathsf{Pvk}_{\mathsf{Id}}^{\mathrm{rerand}}$ as the lower left rectangle, and $\mathsf{Pvk}_{\mathsf{Id}}^{\mathrm{deleg}}$ as the entire right side block:

$$\begin{matrix} \left[\begin{matrix} k_0 & k_{1,(a)} & k_{1,(b)} & \cdots & k_{1+D,(a)} & k_{1+D,(b)} \end{matrix}\right] \\ \begin{bmatrix} f_{0,0} & f_{0,0,(a)} & f_{0,0,(a)} & \cdots & f_{0,1+D,(a)} & f_{0,1+D,(a)} \\ f_{1,0} & f_{1,0,(a)} & f_{1,0,(a)} & \cdots & f_{1,1+D,(a)} & f_{1,1+D,(a)} \\ \vdots & & & \ddots & & \\ f_{1+D,0} & f_{1+D,0,(a)} & f_{1+D,0,(a)} & \cdots & f_{1+D,1+D,(a)} & f_{1+D,1+D,(a)} \end{bmatrix} \end{matrix} \begin{bmatrix} h_{1+L} & \cdots & h_D \\ h_{0,1+L} & \cdots & h_{0,D} \\ h_{1,1+L} & \cdots & h_{1,D} \\ & & \ddots \\ h_{1+D,1+L} & \cdots & h_{1+D,D} \end{bmatrix} \quad .$$

Each row on the left can be viewed as a private key in an independent HIBE system (with generalized linear splitting as in Section 4). The main difference is that only $\mathsf{Pvk}_{\mathsf{Id}}^{\mathrm{decrypt}}$ contains the secret $\hat{w}$. The rows of $\mathsf{Pvk}_{\mathsf{Id}}^{\mathrm{rerand}}$ are independent HIBE keys for the same Id that do not permit decryption. The elements on the right side provide the delegation functionality: each column in $\mathsf{Pvk}_{\mathsf{Id}}^{\mathrm{deleg}}$ extends the hierarchy down one level. Delegation works as follows:

Derive$(\mathsf{Pub}, \mathsf{Pvk}_{\mathsf{Id}|L-1}, \mathsf{Id}_L)$ To derive a private key for $\mathsf{Id} = [I_0, I_1, \ldots, I_L] \in (\mathbb{Z}_p^{\times})^{1+L}$ where $L \in \{2, \ldots, D\}$ and $I_0 = 1$, given the parent's private key, $\mathsf{Pvk}_{\mathsf{Id}|L-1} = [k_0, [k_{n,(a)}, k_{n,(b)}], [f_{m,0}, [f_{m,n,(a)}, f_{m,n,(b)}]], [h_\ell, [h_{m,\ell}]]_{\ell=L,\ldots,D}]$, (where n, m range over $\{0, \ldots, 1+D\}$), do the following:

1. Pick $6 + 5\,D + D^2$ random integers:

$$[\ \pi_m,\ [\ \pi_{m,m'}\ ]_{m'=0,\ldots,1+D}\ ]_{m=0,\ldots,1+D} \quad \in_\$ \ (\mathbb{Z}_p)^{(3+D)\,(2+D)}\ .$$

2. Output the subordinate private key $\mathsf{Pvk}_{\mathsf{Id}} \in \hat{\mathbb{G}}^{(5+3\,D-L)\,(3+D)}$ comprised of $\mathsf{Pvk}_{\mathsf{Id}}^{\mathrm{decrypt}}$, $\mathsf{Pvk}_{\mathsf{Id}}^{\mathrm{rerand}}$, and $\mathsf{Pvk}_{\mathsf{Id}}^{\mathrm{deleg}}$, where:
   (a) To build $\mathsf{Pvk}_{\mathsf{Id}}^{\mathrm{decrypt}} = k'_0, [k'_{n,(a)}, k'_{n,(b)}]_{n=0,\ldots,1+D}$, we set, for all n:

$$k'_0 \ \leftarrow\ (k_0 \prod_{m=0}^{1+D} (f_{m,0})^{\pi_m})\,(h_\ell \prod_{m=0}^{1+D} (h_{m,\ell})^{\pi_m})^{I_L}$$

$$k'_{n,(a)},\ k'_{n,(b)} \quad \leftarrow \quad k_{n,(a)} \prod_{m=0}^{1+D} (f_{m,n,(a)})^{\pi_m},\ \ k_{n,(b)} \prod_{m=0}^{1+D} (f_{m,n,(b)})^{\pi_m}$$

   (b) For $\mathsf{Pvk}_{\mathsf{Id}}^{\mathrm{rerand}} = [f'_{m',0}, [f'_{m',n,(a)}, f'_{m',n,(b)}]_{n=0,\ldots,1+D}]_{m'=0,\ldots,1+D}$:

$$f'_{m',0} \quad \leftarrow \quad (\prod_{m=0}^{1+D} (f_{m,0})^{\pi_{m,m'}})\,(\prod_{m=0}^{1+D} (h_{m,\ell})^{\pi_{m,m'}})^{I_L}$$

$$f'_{m',n,(a)},\ f'_{m',n,(b)} \quad \leftarrow \quad \prod_{m=0}^{1+D} (f_{m,n,(a)})^{\pi_{m,m'}},\ \prod_{m=0}^{1+D} (f_{m,n,(b)})^{\pi_{m,m'}}$$

   (c) And for $\mathsf{Pvk}_{\mathsf{Id}}^{\mathrm{deleg}} = [h'_\ell, [h'_{m',\ell}]_{m'=0,\ldots,1+D}]_{\ell=1+L,\ldots,D}$, we assign:

$$h'_\ell \ \leftarrow\ h_\ell \prod_{m=0}^{1+D} (h_{m,\ell})^{\pi_m}$$

$$h'_{m',\ell} \ \leftarrow\ \prod_{m=0}^{1+D} (h_{m,\ell})^{\pi_{m,m'}}$$

We note that *Derive* and *Extract* create private keys with the same structure and distribution. Notice that the *Derive* algorithm can be interpreted as the combination of two distinct operations: hierarchical delegation and key re-randomization.

– We start by re-randomizing the parent key, conceptually speaking, by performing random linear combinations of all its rows (in the array representation shown earlier). The first row enjoys special treatment: its coefficient into other rows' re-randomization is 0, and its own coefficient is 1.
– We then delegate by transforming the leftmost elements of $\mathsf{Pvk}_{\mathsf{Id}}^{\text{decrypt}}$ and $\mathsf{Pvk}_{\mathsf{Id}}^{\text{rerand}}$, in which identities are encoded. Suppose that re-randomization has already occurred, and imagine the resulting $\mathsf{Pvk}_{\mathsf{Id}}^{\text{decrypt}}$, $\mathsf{Pvk}_{\mathsf{Id}}^{\text{rerand}}$, $\mathsf{Pvk}_{\mathsf{Id}}^{\text{deleg}}$. Delegation to a child identity I_L will "consume" the first column of $\mathsf{Pvk}_{\mathsf{Id}}^{\text{deleg}}$: each element is raised to the power of I_L, and the result is aggregated into the leftmost element of $\mathsf{Pvk}_{\mathsf{Id}}^{\text{decrypt}}$ or $\mathsf{Pvk}_{\mathsf{Id}}^{\text{rerand}}$ on the same row, as follows:

$$\begin{bmatrix} k_0 & \cdots & k_{n,(a)} & k_{n,(b)} & \cdots \end{bmatrix}\begin{bmatrix} h_L & \cdots & h_D \end{bmatrix} \to \begin{bmatrix} k'_0 & \cdots \end{bmatrix}\begin{bmatrix} \bullet & h'_{1+L} & \cdots & h'_D \end{bmatrix}$$
$$\begin{bmatrix} f_{0,0} & \cdots & f_{0,n,(a)} & f_{0,n,(b)} & \cdots \\ \vdots & \ddots & & & \\ f_{m,0} & \cdots & f_{m,n,(a)} & f_{m,n,(b)} & \cdots \\ \vdots & & \ddots & & \end{bmatrix}\begin{bmatrix} h_{0,L} & \cdots & h_{0,D} \\ \vdots & \ddots & \\ h_{m,L} & \cdots & h_{m,D} \\ \vdots & & \ddots \end{bmatrix} \to \begin{bmatrix} f'_{0,0} & \cdots \\ \vdots & \ddots \\ f'_{m,0} & \cdots \\ \vdots & \end{bmatrix}\begin{bmatrix} \bullet & h'_{0,1+L} & \cdots & h'_{0,D} \\ & \ddots & & \\ \bullet & h'_{m,1+L} & \cdots & h'_{m,D} \\ & & & \ddots \end{bmatrix}$$

We now turn to the encryption and decryption methods.

Encrypt$(\mathsf{Pub}, \mathsf{Id}, \mathsf{Msg})$ To encrypt a given message encoded as a group element $\mathsf{Msg} \in \mathbb{G}_T$ for a given identity $\mathsf{Id} = [I_0(=1), I_1, \ldots, I_L]$ at level L:
1. Select $3 + D$ random integers: $r, [r_n]_{n=0,\ldots,1+D} \in_\$ (\mathbb{Z}_p)^{3+D}$.
2. Output the ciphertext $\mathsf{CT} = E, c_0, [c_{n,(a)}, c_{n,(b)}]_{n=0,\ldots,1+D} \in \mathbb{G}_T \times \mathbb{G}^{5+2\,D}$, computed as:

$$\mathsf{CT} \leftarrow \;\; \mathsf{Msg} \cdot \Omega^{-r}, \;\; g^r, \;\; \left[\left(\prod_{\ell=0}^{L} b_{n,\ell}^{I_\ell} \right)^{r_n}, \left(\prod_{\ell=0}^{L} a_{n,\ell}^{I_\ell} \right)^{r-r_n} \right]_{n=0,\ldots,1+D} .$$

Decrypt$(\mathsf{Pub}, \mathsf{Pvk}_{\mathsf{Id}}, \mathsf{CT})$ To decrypt a ciphertext CT, using the decryption subkey a private key, $\mathsf{Pvk}_{\mathsf{Id}}^{\text{decrypt}} = [k_0, [k_{n,(a)}, k_{n,(b)}]_{n=0,\ldots,1+D}]$, compute:

$$\hat{\mathsf{Msg}} \leftarrow E \cdot \mathbf{e}(c_0, k_0) \prod_{n=0}^{1+D} \mathbf{e}(c_{n,(a)}, k_{n,(a)})\, \mathbf{e}(c_{n,(b)}, k_{n,(b)}) \;\in\; \mathbb{G}_T \;.$$

Encryption can be made very cheap with a bit of caching since the exponentiation bases never change. Decryption is also fairly efficient since all the pairings in the product can be computed at once using a "multi-pairing" approach [21], which is similar to multi-exponentiation. One can also exploit the fact that all the $k_{\cdots}$ are fixed for a given recipient to perform advantageous pre-computations [3].

6 Consistency and Security

The following theorems state that extracted and delegated private keys are identically distributed, and that extraction, encryption, and decryption, are consistent. We remark that Theorem 1 is not essential for the security model, but it is nice to have and it is also useful to prove Theorem 2.

Theorem 1. *Private keys calculated by* Derive *and* Extract *have the same distribution.*

Theorem 2. *The Anonymous HIBE scheme is internally consistent.*

We now state the basic security theorems for the A-HIBE scheme. The selective-ID security reductions are almost tight and hold in the standard model. We only consider recipient anonymity, since sender anonymity is trivially attainable in an unauthenticated encryption scheme.

Theorem 3 (Confidentiality). *Suppose that* $\mathbf{G}$ *upholds the* (τ, ϵ)*-Decision BDH assumption. Then, against a selective-ID adversary that makes at most* q *private key extraction queries, the HIBE scheme of Section 5 is* $(q, \tilde{\tau}, \tilde{\epsilon})$*-IND-sID-CPA secure in* $\mathbf{G}$ *with* $\tilde{\tau} \approx \tau$ *and* $\tilde{\epsilon} = \epsilon - (3 + D)\, q/p$.

Theorem 4 (Anonymity). *Suppose that* $\mathbf{G}$ *upholds the* (τ, ϵ)*-Decision Linear assumption. Then, against a selective-ID adversary that makes* q *private key extraction queries, the HIBE scheme of Section 5 is* $(q, \tilde{\tau}, \tilde{\epsilon})$*-ANON-sID-CPA secure in* $\mathbf{G}$ *with* $\tilde{\tau} \approx \tau$ *and* $\tilde{\epsilon} = \epsilon - (2 + D)\,(7 + 3\,D)\, q/p$.

For completeness, we mention that based on the above theorems it is easy to secure the scheme against active adversaries, *i.e.*, adaptive-ID and CCA2. Adaptive-identity security can be obtained using the Waters [31] technique, or by using random oracles [5, §7], although these methods only work for shallow hierarchies. Adaptive chosen-ciphertext security can be achieved very effectively using one of several techniques [15, 11, 13], all of which are applicable here.

7 Conclusion

We presented a provably anonymous IBE and HIBE scheme without random oracles, which resolves an open question from CRYPTO 2005 regarding the existence of anonymous HIBE systems.

Our constructions make use of a novel "linear-splitting" technique which prevents an attacker from testing the intended recipient of ciphertexts, yet allows for the use of randomized private IBE keys. In the hierarchical case, we add to this a new "multi-simulation" proof device that permits multiple HIBE subsystems to concurrently re-randomize each other. Security is based solely on the Linear assumption in bilinear groups.

Our basic scheme is very efficient, a factor two slower than (non-anonymous) Boneh-Boyen BB_1 and BB_2 encryption, and quite faster than Boneh-Franklin. The full hierarchical scheme remains practical with its quadratic private key size, and its linear ciphertext size, encryption time, and decryption time, as functions of the depth of the hierarchy.

Acknowledgements

The authors would like to thank Mihir Bellare, Dan Boneh, and Hovav Shacham for helpful discussions, as well as the anonymous referees for useful comments.

References

1. Michel Abdalla, Mihir Bellare, Dario Catalano, Eike Kiltz, Tadayoshi Kohno, Tanja Lange, John Malone-Lee, Gregory Neven, Pascal Paillier, and Haixia Shi. Searchable encryption revisited: Consistency properties, relation to anonymous IBE, and extensions. In *Advances in Cryptology—CRYPTO 2005*, Lecture Notes in Computer Science, pages 205–22. Springer-Verlag, 2005.
2. Paulo S. L. M. Barreto and Michael Naehrig. Pairing-friendly elliptic curves of prime order. Cryptology ePrint Archive, Report 2005/133, 2005. `http://eprint.iacr.org/`.
3. Paulo S.L.M. Barreto, Hae Y. Kim, Ben Lynn, and Michael Scott. Efficient algorithms for pairing-based cryptosystems. Cryptology ePrint Archive, Report 2002/008, 2002. `http://eprint.iacr.org/`.
4. Mihir Bellare, Alexandra Boldyreva, Anand Desai, and David Pointcheval. Key-privacy in public-key encryption. In *Proceedings of ASIACRYPT 2001*, Lecture Notes in Computer Science, pages 566–82. Springer-Verlag, 2001.
5. Dan Boneh and Xavier Boyen. Efficient selective-ID secure identity based encryption without random oracles. In *Advances in Cryptology—EUROCRYPT 2004*, volume 3027 of *Lecture Notes in Computer Science*, pages 223–38. Springer-Verlag, 2004.
6. Dan Boneh and Xavier Boyen. Secure identity based encryption without random oracles. In *Advances in Cryptology—CRYPTO 2004*, volume 3152 of *Lecture Notes in Computer Science*, pages 443–59. Springer-Verlag, 2004.
7. Dan Boneh, Xavier Boyen, and Eu-Jin Goh. Hierarchical identity based encryption with constant size ciphertext. In *Advances in Cryptology—EUROCRYPT 2005*, volume 3494 of *Lecture Notes in Computer Science*, pages 440–56. Springer-Verlag, 2005.
8. Dan Boneh, Xavier Boyen, and Hovav Shacham. Short group signatures. In *Advances in Cryptology—CRYPTO 2004*, volume 3152 of *Lecture Notes in Computer Science*, pages 41–55. Springer-Verlag, 2004.
9. Dan Boneh, Giovanni Di Crescenzo, Rafail Ostrovsky, and Giuseppe Persiano. Public key encryption with keyword search. In *Advances in Cryptology—EUROCRYPT 2004*, volume 3027 of *Lecture Notes in Computer Science*, pages 506–22. Springer-Verlag, 2004.
10. Dan Boneh and Matthew Franklin. Identity-based encryption from the Weil pairing. *SIAM Journal of Computing*, 32(3):586–615, 2003. Extended abstract in *Advances in Cryptology—CRYPTO 2001*.
11. Dan Boneh and Jonathan Katz. Improved efficiency for CCA-secure cryptosystems built using identity based encryption. In *Proceedings of CT-RSA 2005*, volume 3376 of *Lecture Notes in Computer Science*. Springer-Verlag, 2005.
12. Xavier Boyen. Multipurpose identity-based signcryption: A Swiss Army knife for identity-based cryptography. In *Advances in Cryptology—CRYPTO 2003*, volume 2729 of *Lecture Notes in Computer Science*, pages 383–99. Springer-Verlag, 2003.
13. Xavier Boyen, Qixiang Mei, and Brent Waters. Direct chosen ciphertext security from identity-based techniques. In *ACM Conference on Computer and Communications Security—CCS 2005*. ACM Press, 2005.
14. Ran Canetti, Shai Halevi, and Jonathan Katz. A forward-secure public-key encryption scheme. In *Advances in Cryptology—EUROCRYPT 2003*, volume 2656 of *Lecture Notes in Computer Science*. Springer-Verlag, 2003.

15. Ran Canetti, Shai Halevi, and Jonathan Katz. Chosen-ciphertext security from identity-based encryption. In *Advances in Cryptology—EUROCRYPT 2004*, volume 3027 of *Lecture Notes in Computer Science*. Springer-Verlag, 2004.
16. Sanjit Chatterjee and Palash Sarkar. Trading time for space: Towards an efficient IBE scheme with short(er) public parameters in the standard model. In *Proceedings of ICISC 2005*, 2005.
17. Clifford Cocks. An identity based encryption scheme based on quadratic residues. In *Proceedings of the 8th IMA International Conference on Cryptography and Coding*, 2001.
18. Darren Davis, Fabian Monrose, and Michael K. Reiter. Time-scoped searching of encrypted audit logs. In *Proceedings of ICICS 2004*, pages 532–45, 2004.
19. Craig Gentry. Practical identity-based encryption without random oracles. In *Advances in Cryptology—EUROCRYPT 2006*, Lecture Notes in Computer Science. Springer-Verlag, 2006.
20. Craig Gentry and Alice Silverberg. Hierarchical ID-based cryptography. In *Proceedings of ASIACRYPT 2002*. Springer-Verlag, 2002.
21. Robert Granger and Nigel P. Smart. On computing products of pairings. Cryptology ePrint Archive, Report 2006/172, 2006. `http://eprint.iacr.org/`.
22. Jeremy Horwitz and Ben Lynn. Towards hierarchical identity-based encryption. In *Advances in Cryptology—EUROCRYPT 2002*, Lecture Notes in Computer Science, pages 466–81. Springer-Verlag, 2002.
23. Antoine Joux. A one round protocol for tripartite Diffie-Hellman. *Journal of Cryptology*, 17(4):263–76, 2004. Extended abstract in *Proceedings of ANTS IV, 2000*.
24. Alfred Menezes, Tatsuaki Okamoto, and Scott Vanstone. Reducing elliptic curve logarithms in a finite field. *IEEE Transactions on Information Theory*, 39(5):1639–46, 1993.
25. Victor Miller. The Weil pairing, and its efficient calculation. *Journal of Cryptology*, 17(4), 2004.
26. David Naccache. Secure and practical identity-based encryption. Cryptology ePrint Archive, Report 2005/369, 2005. `http://eprint.iacr.org/`.
27. Moni Naor. On cryptographic assumptions and challenges. In *Advances in Cryptology—CRYPTO 2003*. Springer-Verlag, 2003.
28. Adi Shamir. Identity-based cryptosystems and signature schemes. In *Advances in Cryptology—CRYPTO 1984*, volume 196 of *Lecture Notes in Computer Science*, pages 47–53. Springer-Verlag, 1984.
29. Victor Shoup. Lower bounds for discrete logarithms and related problems. In *Advances in Cryptology—EUROCRYPT 1997*, volume 1233 of *Lecture Notes in Computer Science*. Springer-Verlag, 1997.
30. Dawn Xiaodong Song, David Wagner, and Adrian Perrig. Practical techniques for searches on encrypted data. In *Proceedings of the IEEE Symposium on Security and Privacy—SP 2000*. IEEE Computer Society, 2000.
31. Brent Waters. Efficient identity-based encryption without random oracles. In *Advances in Cryptology—EUROCRYPT 2005*, volume 3494 of *Lecture Notes in Computer Science*. Springer-Verlag, 2005.
32. Brent Waters, Dirk Balfanz, Glenn Durfee, and Diana Smetters. Building an encrypted and searchable audit log. In *Proceedings of NDSS 2004*, 2004.
33. Danfeng Yao, Nelly Fazio, Yevgeniy Dodis, and Anna Lysyanskaya. ID-based encryption for complex hierarchies with applications to forward security and broadcast encryption. In *ACM Conference on Computer and Communications Security—CCS 2004*, pages 354–63, 2004.