General *Ad Hoc* Encryption from Exponent Inversion IBE

Xavier Boyen

Voltage Inc.
Palo Alto
xb@boyen.org

Abstract. Among the three broad classes of Identity-Based Encryption schemes built from pairings, the exponent inversion paradigm tends to be the most efficient, but also the least extensible: currently there are no hierarchical or other known extension of IBE based on those schemes. In this work, we show that such extensions can be realized from IBE systems that conform to a certain abstraction of the exponent inversion paradigm. Our method requires no random oracles, and is simple and efficient.

1 Introduction

Since the first practical constructions of the identity-based encryption (IBE) primitive appeared a few years ago [18,9,4], a large body of work has been devoted to creating better realizations of the basic primitive, and to extending it in many interesting ways. With the notable exception of Cocks' basic IBE scheme [9], virtually all IBE-like constructions known to date make more or less extensive use of bilinear pairings on elliptic curves.

The many extensions that have been proposed in the recent years have the common goal to extend the notion of identity from its original atomic meaning, to complex constructs of identity components on which certain operations can be performed. In particular, we mention hierarchical identities [13], fuzzy identities [16], and identities as attributes [12] among the most significant of these extensions. Fortunately, and unlike the original idea of IBE [19] which remained without construction for many years, most of the IBE extensions that have been suggested also have a known construction. However, to temper this optimism, we should note that for many of these extensions, the only realizations we know of all derive from the same basic IBE paradigm, despite the availability of alternatives. In particular, an entire family of very efficient IBE constructions does not seem to support any of the extensions afforded by other families.

Our current knowledge of pairing-based IBE schemes can be partitioned in three broad families: (1) full-domain hash, (2) exponent inversion, and (3) commutative blinding—with little doubt that others will be invented in the future. The connotations behind this taxonomy shall be explicited later on. Each of these

M. Naor (Ed.): EUROCRYPT 2007, LNCS 4515, pp. 394–411, 2007.

categories defines a general construction template, by which encryption and key derivation are matched in an identity-based manner using a bilinear pairing. The one thing that these families have in common is their use of a pairing—but not how they use it. Indeed, the shape of the template greatly affects how the schemes can be extended, and their security proved.

Among the three families, the commutative blinding method originated with BB_1-IBE [2] has distinguished itself as the most fertile ground for generalizing IBE, based on the number of extensions that it currently supports, such as forward secure hierarchies [3], partial-match or fuzzy identities [16], and complex attribute-based policies [12]. It is followed rather distantly by the full-domain hash family, defined by BF-IBE [4], which contains fewer but nevertheless interesting extensions, including hierarchies [11] also with forward security [21]. In stark contrast, based on our current state of knowledge, the exponent inversion family does not seem to have any useful extension, despite the fact that the basic IBE functionality performs more efficiently in this family, based on BB_2-IBE [2] and SK-IBE [7,17], than in the other two. This situation strikes us as odd, as there is no obvious reason why the exponent inversion family should be less accommodating than the other two.

The aim of this paper is to show that the exponent inversion paradigm is more flexible than has been previously recognized. To this end, we first give an abstraction of exponent inversion schemes such as BB_2-IBE and SK-IBE, that captures functional properties such as linearity in the exponent, and which we call Linear IBE. We also define certain security properties that such schemes should satisfy depending on the final goal of the construction; these properties have to do with simultaneous or parallel instances of the IBE running at once, which is a general technique we use in all our constructions. We then apply the method to transform any black-box Linear IBE with suitable security properties into a hierarchical, fuzzy, attribute-based, or distributed system, under generic security reductions to the underlying base IBE abstraction.

The transformations are syntactically black-box, but their security requires the parallel simulation of several base instances, hence our requirement that the underlying scheme be secure in such conditions. In general, the transformations preserve the gist of the security properties of the underlying scheme, *e.g.*, in the random oracle or standard model, and under selective or adaptive security, but keeping in mind that it requires (and consumes) the supplemental notion of parallel IBE security already mentioned. The method is quite simple and preserves the efficiency of the underlying scheme, with a multiplier that depends on the particulars of what the transformation seeks to achieve. In practice, this new approach seems appealing, as it allows the very efficient but barebones SK and BB_2 schemes to become more flexible and thus we hope more useful.

We call *ad hoc cryptosystem* any such public-key system that supports private sub-keys with designated restricted capabilities. This includes IBE and its extensions.

2 A Classification of IBE Schemes

The following is a rough classification of the known identity-based encryption schemes. All of them support at least a basic security reduction to a well-formulated complexity assumption, either in the standard model or in the random oracle model.

"Quadratic Residuosity" IBE (without pairings). We mention Cocks' [9] scheme as the only known example of IBE based on quadratic residuosity in RSA groups; it is inefficient in terms of bandwidth and has no known extension.

"Full Domain Hash" IBE. This is the class of the Boneh-Franklin identity-based encryption [4], and to which the earlier Sakai-Ohgishi-Kasahara identity-based key exchange [18] also belongs.

In BF encryption and the constructions that are based on it, such as [11,21], the session keys are of the form $\mathbf{e}(H(\mathsf{Id}), \hat{g}^{\alpha})^s$ where Id is the recipient identity, α is the master secret, and H is a full-domain hash function into the bilinear group, viewed as a random oracle. In SOK key exchange, the session key $\mathbf{e}(H(\mathsf{Id}_A), H(\mathsf{Id}_B)^{\alpha})^s$ is computed interactively from the identities of both parties, but also involves the master key α and a random oracle as in BF encryption.

"Exponent Inversion" IBE. This approach to IBE can be traced to an idea of Mitsunary, Sakai, and Kasahara in the context of traitor tracing [14]. For IBE, the principle is to obtain a session key of the form $\mathbf{e}(g, \hat{g})^s$ based on a ciphertext $(g^{f(\mathsf{Id})})^s$ and a private key $g^{1/f(\mathsf{Id})}$, where $f(\mathsf{Id})$ is a secret function of the recipient identity but $g^{f(\mathsf{Id})}$ is computable publicly. A benefit of this type of construction is that there is no need to hash directly on the curve. Notice also that the master key cancels out completely from the session key.

This category includes the Sakai-Kasahara scheme originally described in [17] and later proven secure in [7] in the random oracle model. The category also includes the second of two IBE schemes proposed by Boneh and Boyen [2], which has a selective-identity proof of security in the standard model. All these schemes rely on the fairly strong BDHI complexity assumption [2], which was first used in another context by Mitsunary, Sakai, and Kasahara [14]. This assumption, called Bilnear Diffie-Hellman Inversion (BDHI), has been further analyzed in [8].

Recently, Gentry [10] proposed another construction that has superficial similarities to the others in this category, but with a proof of security in the adaptive-identity model (based on an even stronger assumption). Gentry's IBE scheme appears to belong in the exponent inversion category, although the case is not clear-cut because the session key is not of the form $\mathbf{e}(g, \hat{g})^s$, but of the form $\mathbf{e}(g, \hat{h})^s$, where $\hat{h}$ is created by the initial setup procedure. Although $\hat{h}$ remains statistically independent of the secret key, it is not intended to be constant from one instance of the system to the next, and Gentry's security proof no longer applies if $\hat{h}$ and thus $\mathbf{e}(g, \hat{h})$ is fixed.

"Commutative Blinding" IBE. The last category of IBE systems descends from BB_1, the first scheme given in the Boneh-Boyen paper [2]. These systems are

based on the same BDH assumption as the Boneh-Franklin scheme [4], but use a mechanism that avoids random oracles. Very roughly, the general principle is to create blinding factors from two secret coefficients in a way that makes them "commute" (*i.e.*, not depend on the application order), thanks to the pairing.

The algebraic versatility exhibited by the BB_1 approach has given rise to a fair number of extensions to the original scheme; see for example [3,16,20,15,1]. Virtually all constructions in the commutative blinding paradigm have session keys of the form $\mathbf{e}(g, \hat{g}^{\alpha})^s$, where α is part of the master key, and s is chosen by the sender.

It is likely that the coming years will see the emergence of additional families of schemes. In this paper, we are concerned with the Exponent Inversion family, which tends to be the most computationally efficient and arguably requires the least bandwidth, but currently lacks the flexibility of the other pairing-based families (such as Commutative Blinding especially).

3 Exponent Inversion Abstractions

We now describe an abstraction of IBE that captures the properties of the exponent inversion paradigm that we need. Our abstraction is sufficiently powerful to support a wide variety of generic constructions, and sufficiently general to encompass all IBE schemes known to date that do not "obviously" fall outside of the exponent inversion paradigm.

3.1 Linear IBE Schemes

Based on the properties that our semi-generic construction will require, we define the following abstraction of IBE schemes that use the exponent inversion principle. Two basic schemes mentioned earlier (BB_2 and SK) fit particularly nicely within this abstraction.

Intuitively, we exploit two facets of the "linearity" exhibited by exponent inversion IBE. All such schemes construct their identity-based trapdoor from a secret polynomial $\theta(\mathsf{Id})$, and publish enough information to allow anyone to compute $g^{\theta(\mathsf{Id})}$ but not $\hat{g}^{1/\theta(\mathsf{Id})}$. The latter can serve as private key for Id, and the trapdoor arises from the cancellation of the exponents on both sides of the pairing: $\mathbf{e}(g^{\theta(\mathsf{Id})}, \hat{g}^{1/\theta(\mathsf{Id})}) = \mathbf{e}(g, \hat{g})$. To get an IBE scheme, the encryptor needs to pick a randomization exponent s; the ciphertext becomes $g^{\theta(\mathsf{Id})\, s}$ and the session key $\mathbf{e}(g, \hat{g})^s$. Because session keys constructed this way are linear in both the private key and the ciphertext, it will be easy to construct secret sharing schemes in the exponent either in the ciphertext or on the private key side. This is the first property we need (we shall precise and generalize it momentarily).

Our second property is the independence of session keys with respect to the master secret. As in any IBE scheme, the master secret is needed to construct the private keys, but here it need not affect the choice of session keys. Indeed, if the generators g and $\hat{g}$ are imposed externally, the only degree of freedom in the session key $\mathbf{e}(g, \hat{g})^s$ is the exponent s chosen by the encryptor. (This is very much unlike full-domain hash and commutative blinding IBE schemes, in which session keys are respectively of the form $\mathbf{e}(H(\mathsf{Id}), \hat{g}^{\alpha})^s$ and $\mathbf{e}(g, \hat{g}^{\alpha})^s$ and necessarily involve the master key α.)

As already mentioned, Gentry's IBE scheme uses session keys of the form $\mathbf{e}(g, \hat{h})^s$ rather than $\mathbf{e}(g, \hat{g})^s$, where $\hat{h}$ is created at random by the initial setup procedure. Although our template requires $\hat{h}$ to be fixed, the current proof of Gentry's IBE does not tolerate it, and so we provisionally include Gentry-IBE as a "syntactic" Linear IBE scheme until the question can be settled.

A Template for Exponent Inversion IBE. Toward formalizing the requirements above, we first define the particular template that candidate IBE schemes must obey.

$\mathsf{Setup}(\mathbf{e}, g, \hat{g}, v, \omega)$ on input a pairing $\mathbf{e} : \mathbb{G} \times \hat{\mathbb{G}} \to \mathbb{G}_t$, generators $g \in \mathbb{G}$, $\hat{g} \in \hat{\mathbb{G}}$, $v \in \mathbb{G}_t$, and a random seed ω, outputs a master key pair $(\mathsf{Msk}, \mathsf{Pub})$ where $\mathsf{Pub} = (\mathbf{e}, g, \hat{g}, v, ...)$.

We require key pairs generated from independent random seeds $\omega_1, \omega_2, ...$ to be mutually independent. We allow key pairs generated from the same inputs $\mathbf{e}, g, \hat{g}, v, \omega$ to be mutually independent, as the setup algorithm is permitted to use its own source of randomness.

$\mathsf{Extract}(\mathsf{Msk}, \mathsf{Id})$ on input Msk and an identity Id, outputs a private key $\mathsf{Pvk}_{\mathsf{Id}} = (\mathsf{Id}, R, \boldsymbol{d})$, which can be deterministic or randomized.

Here, $\mathsf{Id} \in \mathcal{I}d$, the domain of identities; $R \in \mathcal{R}d$, some non-empty auxiliary domain; and $\boldsymbol{d} = (d_1, ..., d_n) \in \mathcal{D}$, a vector space of n coordinates, each a copy of one of $\mathbb{F}_p$, $\mathbb{G}$, $\hat{\mathbb{G}}$, $\mathbb{G}_t$.

$\mathsf{Encrypt}(\mathsf{Pub}, \mathsf{Id}, \mathsf{Msg}, s)$ on input Pub, a recipient Id, a plaintext Msg, and a randomization exponent $s \in \mathbb{F}_p$, outputs a ciphertext $\mathsf{Ctx} = (\mathsf{Id}, S, c_0, \boldsymbol{c})$.

Here we require that $\mathsf{Msg} \in \mathbb{G}_t$, that $c_0 = \mathsf{Msg} \cdot v^s$, and that $\boldsymbol{c} = (c_1, ..., c_m) \in \mathcal{C}$, where $\mathcal{C}$ is a vector space of m coordinates, each being a copy of $\mathbb{F}_p$, $\mathbb{G}$, $\hat{\mathbb{G}}$, or $\mathbb{G}_t$. Finally, we assume that $S \in \mathcal{S}d$, with $\mathcal{S}d$ some non-empty auxiliary domain.

$\mathsf{Decrypt}(\mathsf{Pub}, \mathsf{Pvk}_{\mathsf{Id}}, \mathsf{Ctx})$ on input Pub, a private key $\mathsf{Pvk}_{\mathsf{Id}} = (\mathsf{Id}, R, \boldsymbol{d})$, and a ciphertext $\mathsf{Ctx} = (\mathsf{Id}, S, \mathsf{Msg} \cdot v^s, \boldsymbol{c})$, outputs Msg provided the inputs are well-formed and the identities match.

The purpose of ω given to setup is to allow the creation of multiple instances of a single scheme with related keys; this may enable certain schemes (potentially Gentry's) to fit the template, provided that other security conditions are met. Normally, ω is ignored by the underlying scheme and all key pairs are independent.

Based on this template, we define the notion of Linear IBE to capture the intuitive linearity properties of the session keys that we discussed.

Definition 1. *A Linear IBE scheme,* (Setup, Extract, Encrypt, Decrypt), *is a quadruple of algorithms that follows the template above, and further satisfies the two properties below.*

1. *There exists a (publicly) efficiently computable function,* $\mathbf{f}_{\mathsf{Pub}} : \mathcal{I}d \times \mathcal{R}d \times \mathcal{S}d \times \mathcal{C} \times \mathcal{D} \to \mathbb{G}_t$*, linear in each of its last two arguments, such that, for all well-formed* $\mathsf{Pvk}_{\mathsf{Id}} = (\mathsf{Id}, R, \boldsymbol{d})$ *and* $\mathsf{Ctx} = (\mathsf{Id}, S, c_0, \boldsymbol{c})$*,*

$$\mathbf{f}_{\mathsf{Pub}}\,(\mathsf{Id},\ R,\ S,\ \boldsymbol{c},\ \boldsymbol{d}) = v^{-s}\ ,$$

where we recall that v *is the generator of* $\mathbb{G}_t$ *given as input to the* Setup *function, and thus independent of the choice of* Msk*.*
Note that the decryption algorithm reduces to: $\mathsf{Decrypt}(\mathsf{Pvk}_{\mathsf{Id}}, \mathsf{Ctx}) \leftarrow c_0 \cdot \mathbf{f}_{\mathsf{Pub}}(\mathsf{Id}, R, S, \boldsymbol{c}, \boldsymbol{d})$*.*

2. *For any two possibly identical public keys* Pub_1 *and* Pub_2 *derived from the same parameters* $(\mathbf{e}, g, \hat{g}, v, \omega)$*, for any auxiliary values* R'_1 *and* R'_2*, and for any identities* Id_1 *and* Id_2 *such that* $\mathsf{Pub}_1 \neq \mathsf{Pub}_2 \vee \mathsf{Id}_1 \neq \mathsf{Id}_2$*, one can publicly and efficiently find two "reciprocal private keys"* $\boldsymbol{d}'_1 = (\hat{d}'_{1,1}, ..., \hat{d}'_{1,n})$ *and* $\boldsymbol{d}'_2 = (\hat{d}'_{2,1}, ..., \hat{d}'_{2,n})$ *such that:*
   (a) *For* $i, j = 1, 2$*, let* $[\boldsymbol{d}_{ij} : (\mathsf{Id}_j, R, \boldsymbol{d}_{ij}) \leftarrow \mathsf{Extract}(\mathsf{Msk}_j, \mathsf{Id}_j) \mid R = R_i]$ *be the conditional distribution induced by sampling the extraction algorithm and retaining outcomes with the stated auxiliary value* R_i*. There must exist a non-trivial linear combination with coefficients* $t_{ij} \in \mathbb{F}_p$*, allowed to depend on the* R_i *and* R'_j*, that renders these random variables statistically indistinguishable,*

$$[\boldsymbol{d}'_1] \sim [(\boldsymbol{d}_{11})^{t_{11}}\,(\boldsymbol{d}_{12})^{t_{12}}]\ ,$$
$$[\boldsymbol{d}'_2] \sim [(\boldsymbol{d}_{21})^{t_{21}}\,(\boldsymbol{d}_{22})^{t_{22}}]\ .$$

   (b) *For any two well-formed ciphertexts* $\mathsf{Ctx}_1 = (\mathsf{Id}_1,\ S_1,\ \mathsf{Msg}_1 \cdot v^s,\ \boldsymbol{c}_1)$ *and* $\mathsf{Ctx}_2 = (\mathsf{Id}_2,\ S_2,\ \mathsf{Msg}_2 \cdot v^s,\ \boldsymbol{c}_2)$*, for identities* Id_1 *and* Id_2*, and built with the same randomization exponent* s*, we have,*

$$\mathbf{f}_{\mathsf{Pub}}(\mathsf{Id}_1,\ R'_1,\ S_1,\ \boldsymbol{c}_1,\ \boldsymbol{d}'_1) \cdot \mathbf{f}_{\mathsf{Pub}}(\mathsf{Id}_2,\ R'_2,\ S_2,\ \boldsymbol{c}_2,\ \boldsymbol{d}'_2) = v^0 = 1\ .$$

Property 1 expresses our two earlier requirements: first, that the session keys be bilinear functions of both the private keys and the ciphertexts (represented by $\boldsymbol{c}$ and $\boldsymbol{d}$); and second, that session keys be of the form v^{-s} for externally fixed v, and thus independent of the master key.

Property 2 asks that anyone be able to produce $\boldsymbol{d}'_1$ and $\boldsymbol{d}'_2$ that cancel out when used as private keys. The private keys Pvk_1 and Pvk_2 and the linear coefficients $t_{11}, ..., t_{22}$ must provably exist, but they need not and should not be efficiently computable from public information (as this would be incompatible with IBE security). Requirement 2a serves to ensures that $\boldsymbol{d}'_1$ and $\boldsymbol{d}'_2$ are properly randomized and compatible with the function $\mathbf{f}_{\mathsf{Pub}}$. Requirement 2b implies a generalization to arbitrary linear combinations of keys $\boldsymbol{d}'_1, ..., \boldsymbol{d}'_k$ for any number k of identities (and auxiliary values): cancellation would then occur in a k-wise product under the chosen linear combination. We shall see this in action in the HIBE scheme of Section 5.1.

3.2 Parallel IBE Security

The preceding notion of Linear IBE must be strengthened slightly in order to be useful. What we need is a weak notion of parallelism for the IBE scheme that extends to the simulation proofs, but that does not necessarily entail full concurrency.

Essentially, we want the ability to run multiple instances of the IBE at once, in a way that the session keys be all the same (though the identities might be different). For this, we need all the instances to use the same target group generator $v \in \mathbb{G}_t$ (which need not be specified externally), and allow them to use the same random exponent s to create the common session key v^s.

We define the notion of parallel semantic security under selective-identity chosen plaintext attack using the following game played against an attacker $\mathcal{A}$.

Target: $\mathcal{A}$ announces the identities $\mathsf{Id}_1^*, ..., \mathsf{Id}_\ell^*$ it intends to attack.

Setup: The challenger generates a set of public bilinear parameters $(\mathbf{e}, g, \hat{g}, v)$ and a secret random seed ω, and makes ℓ independent calls to the IBE setup algorithm $(\mathsf{Msk}_i, \mathsf{Pub}_i) \leftarrow \mathsf{Setup}(\mathbf{e}, g, \hat{g}, v, \omega)$ using these inputs, but with different internal randomness if Setup uses any. $\mathcal{A}$ is given $(\mathbf{e}, g, \hat{g}, v)$ and the ℓ public keys $\mathsf{Pub}_1, ..., \mathsf{Pub}_\ell$, which may or may not be the same.

Queries I: $\mathcal{A}$ adaptively submits private key extraction queries on each IBE scheme. For any query Id made with respect to the i-th IBE public key Pub_i, we require that $\mathsf{Id} \neq \mathsf{Id}_i^*$. The challenger answers such a query with $\mathsf{Pvk}_{\mathsf{Id},i} \leftarrow \mathsf{Extract}(\mathsf{Msk}_i, \mathsf{Id})$, recalling $\mathsf{Pvk}_{\mathsf{Id},i}$ from storage if it has been computed already.

Challenge: $\mathcal{A}$ then outputs two messages Msg_1 and Msg_2 on which it wishes to be challenged. The challenger selects $b \in \{1, 2\}$ at random, draws a random exponent $s \in \mathbb{F}_p$, and creates ℓ ciphertexts $\mathsf{Ctx}_i \leftarrow \mathsf{Encrypt}(\mathsf{Pub}_i, \mathsf{Id}_i^*, \mathsf{Msg}_b, s)$ using the same message Msg_b. The challenge given to $\mathcal{A}$ is the ℓ ciphertexts $\mathsf{Ctx}_1, ..., \mathsf{Ctx}_\ell$.

Queries II: $\mathcal{A}$ makes additional queries under the same constraints as before, to which the challenger responds as before. The total number of queries to each IBE subsystem in phases I and II may not exceed q.

Guess: $\mathcal{A}$ eventually outputs a guess $b' \in \{1, 2\}$, and wins the game if $b' = b$.

Definition 2. *We say that an IBE scheme is $(q, \ell, \tau, \epsilon)$-Par-IND-sID-CPA secure if there is no adversary $\mathcal{A}$ that and wins the preceding game in time τ with probability at least $\frac{1}{2} + \epsilon$.*

We say that an IBE scheme is $(q, \ell, \tau, \epsilon)$-Par-IND-ID-CPA secure in the same conditions, if the Target phase is moved to the beginning of the Challenge phase.

We further strengthen the security notion by offering an additional type of key extraction query, which captures the intuition that the challenger is able to create linear relations between arbitrary private keys, including the ones on the target identities (albeit without revealing what those are). We define this security property separately because it is not needed for all generic constructions. In Query phases I and II, we add a "parallel simulation" query, which goes as follows:

> **Queries I' & II':** $\mathcal{A}$ can make adaptive "parallel simulation" queries across all IBE instances. To query, $\mathcal{A}$ outputs $k+1$ pairs (i_j, Id_{i_j}) where $\{i_0, ..., i_k\} \subseteq \{1, ..., \ell\}$. We require $\mathsf{Id}_{i_j} \neq \mathsf{Id}^*_{i_j}$ for $j = 1, ..., k$ but allow $\mathsf{Id}_{i_0} = \mathsf{Id}^*_{i_0}$. To respond, $\mathcal{B}$ picks a random $\gamma \in_\$ \mathbb{F}_p^\times$; for $j = 0, ..., k$, it computes $\mathsf{Pvk}_{i_j} = (\mathsf{Id}_{i_j}, R_{i,j}, \boldsymbol{d}_{i,j}) \leftarrow \mathsf{Extract}(\mathsf{Msk}_{i_j}, \mathsf{Id}_{i_j})$, or recalls it from storage if is was computed before; it then outputs $(\mathsf{Id}_{i_j}, R_{i,j}, (\boldsymbol{d}_{i,j})^\gamma)$ for $j = 0, ..., k$.
>
> Each new needed call to $\mathsf{Extract}$ counts toward the quota of q private key queries; no $\mathsf{Pvk}_{\mathsf{Id},i}$ is ever recomputed under different randomizations.

The above game augmented with the "parallel simulation" query defines the following security notion.

Definition 3. *We say that an IBE scheme is $(q, \ell, \tau, \epsilon)$-ParSim-IND-sID-CPA secure if there is no adversary $\mathcal{A}$ that and wins the augmented game in time τ with probability at least $\frac{1}{2} + \epsilon$.*

We similarly define adaptive-identity $(q, \ell, \tau, \epsilon)$-ParSim-IND-ID-CPA security, if the Target phase is postponed to the beginning of the Challenge phase.

We short-handedly say that an IBE scheme is Exponent Inversion Compliant (or EI-compliant) if it satisfies Definitions 1 and 3, and thus 2 (with parameters that are understood from context).

4 Concrete Instantiations

In this section, we prove that the canonical examples of IBE schemes that intuitively fall under the exponent inversion umbrella are, indeed, Linear IBE schemes per our formal definition, and also fulfil the Parallel Simulation IBE security property (albeit in different ways). For completeness, we briefly review the workings of each scheme, and refer to the literature for the details.

4.1 BB_2-IBE

Our first example is the second of two IBE constructions given by Boneh and Boyen in [2], or BB_2. It was originally proven secure against selective-identity attacks from the BDHI assumption [14,2] in the standard model.

- BB_2.Setup outputs the master key $\mathsf{Msk} \leftarrow (a,\, b)$ and the public parameters $\mathsf{Pub} \leftarrow \big(g,\, g_a = g^a,\, g_b = g^b,\, v = \mathbf{e}(g, \hat{g})\big)$ where $a, b \in_\$ \mathbb{F}_p$.
- BB_2.$\mathsf{Extract}(\mathsf{Msk}, \mathsf{Id})$ outputs $\mathsf{Pvk}_{\mathsf{Id}} \leftarrow \Big(r_{\mathsf{Id}} = r,\, \hat{d}_{\mathsf{Id}} = \hat{g}^{\frac{-1}{a+\mathsf{Id}+b\,r}}\Big)$ for $r \in_\$ \mathbb{F}_p$.

- $\mathsf{BB}_2.\mathsf{Encrypt}(\mathsf{Pub}, \mathsf{Id}, \mathsf{Msg}, s)$ outputs $\mathsf{Ctx} \leftarrow (c_0, c_1, c_2)$ where $c_0 = \mathsf{Msg} \cdot v^s$, $c_1 = (g_a\, g^{\mathsf{Id}})^s$, $c_2 = g_b^s$ for the given s.
- $\mathsf{BB}_2.\mathsf{Decrypt}(\mathsf{Pub}, \mathsf{Pvk}_{\mathsf{Id}}, \mathsf{Ctx})$ outputs $\mathsf{Msg}' \leftarrow c_0 \cdot \mathbf{e}(c_1\, c_2^{r_{\mathsf{Id}}},\, \hat{d}_{\mathsf{Id}}) \in \mathbb{G}_t$.

Note that the setup seed ω is not used; the master key (a, b) is generated from internal randomness.

Lemma 1. *BB_2-IBE is a Linear IBE scheme.*[1]

Proof. For key and ciphertext with matching identities, we find that $\mathsf{Msg}' = (\mathsf{Msg} \cdot v^s) \cdot v^{-s} = \mathsf{Msg}$. Towards Property 1, if follows that,

$$\mathbf{f}_{\mathsf{Pub}}\Big(\mathsf{Id},\ R = (r_{\mathsf{Id}}),\ S = \bot,\ \boldsymbol{c} = (c_1, c_2),\ \boldsymbol{d} = (\hat{d}_{\mathsf{Id}})\Big) = \mathbf{e}(c_1\, c_2^{r_{\mathsf{Id}}},\, \hat{d}_{\mathsf{Id}}) = v^{-s}\ .$$

Linearity in the last two arguments is then easy to show. In particular,

$$\mathbf{f}_{\mathsf{Pub}}\Big(\mathsf{Id},\ R = (r_{\mathsf{Id}}),\ \bot,\ \boldsymbol{c}^\alpha = (c_1^\alpha, c_2^\alpha),\ \boldsymbol{d}^\beta = (\hat{d}_{\mathsf{Id}})^\beta\Big) = v^{-s\,\alpha\,\beta}\ .$$

For Property 2, given Id_1, Id_2, and any $r'_1, r'_2 \in \mathbb{F}_p$, set $\boldsymbol{d}'_1 = (\hat{g}_{a_2}\, \hat{g}_{b_2}^{r'_2}\, \hat{g}^{\mathsf{Id}_2})$ and $\boldsymbol{d}'_2 = (\hat{g}_{a_1}\, \hat{g}_{b_1}^{r'_1}\, \hat{g}^{-\mathsf{Id}_1})^{-1}$, taking (g_{a_1}, g_{b_1}) from Pub_1 and (g_{a_2}, g_{b_2}) from Pub_2, which are not necessarily distinct. Then, for actual private keys $\mathsf{Pvk}_{\mathsf{Id}_1} = (r_1, \boldsymbol{d}_1)$ and $\mathsf{Pvk}_{\mathsf{Id}_2} = (r_2, \boldsymbol{d}_2)$, we have,

$$\begin{aligned} \boldsymbol{d}'_1 &= (\boldsymbol{d}_1)^{(a_1 + b_1\, r_1 + \mathsf{Id}_1)\,(a_2 + b_2\, r'_2 + \mathsf{Id}_2)} \cdot (\boldsymbol{d}_2)^0\ , \\ \boldsymbol{d}'_2 &= (\boldsymbol{d}_1)^0 \cdot (\boldsymbol{d}_2)^{-(a_2 + b_2\, r_2 + \mathsf{Id}_2)\,(a_1 + b_1\, r'_1 + \mathsf{Id}_1)}\ , \end{aligned}$$

and, for any $\boldsymbol{c}_1 = \big((g_{a_1}\, g^{\mathsf{Id}_1})^s, g_{b_1}^s\big)$ and $\boldsymbol{c}_2 = \big((g_{a_2}\, g^{\mathsf{Id}_2})^s, g_{b_2}^s\big)$, we have that, $\mathbf{f}_{\mathsf{Pub}}\,(\mathsf{Id}_1,\, r'_1,\, \bot,\, \boldsymbol{c}_1,\, \boldsymbol{d}'_1) \cdot \mathbf{f}_{\mathsf{Pub}}\,(\mathsf{Id}_2,\, r'_2,\, \bot,\, \boldsymbol{c}_2,\, \boldsymbol{d}'_2) = 1, \forall s,$ as required.

The following lemma generalizes the BB_2 security theorem from [2] to the notion of parallel IBE semantic security defined in Section 3.2.

Lemma 2. *BB_2-IBE is $(q, \ell, \tau, \epsilon)$-$\mathsf{ParSim}$-$\mathsf{IND}$-$\mathsf{sID}$-$\mathsf{CPA}$ secure in any bilinear context that satisfies the Decision (q', τ', ϵ)-BDHI assumption with $q' > q\,\ell$ and $\tau' < \tau - \Theta(q^2\,\ell^2)$.*

In other words, BB_2 is secure under a selective-identity, parallel simulation attack, in the standard model, provided that the BDHI assumption holds in the relevant bilinear context.

4.2 SK-IBE

The second scheme we describe is adapted from the identity-based key encapsulation mechanism (IBKEM) given in [7] and attributed to Sakai and Kasahara [17]. Its security proof is set in the random oracle model. For consistency with our definitions, we present an IBE version of the scheme, and call it SK.

[1] See Remark 1 concerning implementations in asymmetric bilinear groups.

- $\mathsf{SK.Setup}$ outputs the master key $\mathsf{Msk} \leftarrow a \in_{\$} \mathbb{F}_p$ and the public key $\mathsf{Pub} \leftarrow (g,\, g_a = g^a,\, v = \mathbf{e}(g, \hat{g}),\, H : \{0,1\}^* \to \mathbb{F}_p)$.
- $\mathsf{SK.Extract}(\mathsf{Msk}, \mathsf{Id})$ outputs the private key $\mathsf{Pvk}_{\mathsf{Id}} \leftarrow \hat{g}^{\frac{1}{a+H(\mathsf{Id})}}$.
- $\mathsf{SK.Encrypt}(\mathsf{Pub}, \mathsf{Id}, \mathsf{Msg}, s)$ outputs $\mathsf{Ctx} \leftarrow \big(c_0 = \mathsf{Msg} \cdot v^s,\ c_1 = (g_a\, g^{H(\mathsf{Id})})^s\big)$.
- $\mathsf{SK.Decrypt}(\mathsf{Pub}, \mathsf{Pvk}_{\mathsf{Id}}, \mathsf{Ctx})$ outputs $\mathsf{Msg}' \leftarrow c_0 \,/\, \mathbf{e}(c_1,\ \mathsf{Pvk}_{\mathsf{Id}}) \in \mathbb{G}_t$.

As in BB_2, the setup seed ω is not used; the master key a is generated from internal randomness.

Lemma 3. *SK-IBE is a Linear IBE scheme.*[1]

Proof. SK-IBE clearly fits the Linear IBE template with $v = \mathbf{e}(g, \hat{g})$. Property 1 is easily verified; in particular, for $\boldsymbol{c}^\alpha = (c_1^\alpha)$ and $\boldsymbol{d}^\beta = (\mathsf{Pvk}_{\mathsf{Id}}^\beta)$,

$$\mathbf{f}_{\mathsf{Pub}}\,(\mathsf{Id},\ \bot,\ \bot,\ \boldsymbol{c},\ \boldsymbol{d}) = \mathbf{e}(c_1,\ \mathsf{Pvk}_{\mathsf{Id}})^{-\alpha\,\beta} = \mathbf{e}(g, \hat{g})^{-s\,\alpha\,\beta} = v^{-s\,\alpha\,\beta}\ .$$

For Property 2, given Id_1 and Id_2 anyone can pick $\boldsymbol{d}'_1 = (\hat{g}_{a_2}\, \hat{g}^{H(\mathsf{Id}_2)})$ and $\boldsymbol{d}'_2 = (\hat{g}_{a_1}^{-1}\, \hat{g}^{-H(\mathsf{Id}_1)})$, so,

$$\begin{aligned} \boldsymbol{d}'_1 &= (\mathsf{Pvk}_{\mathsf{Id}_1})^{t_{11}} \cdot (\mathsf{Pvk}_{\mathsf{Id}_2})^{t_{12}} \qquad & t_{11} &= (a_1 + H(\mathsf{Id}_1))\,(a_2 + H(\mathsf{Id}_2)), \quad t_{12} = 0\ , \\ \boldsymbol{d}'_2 &= (\mathsf{Pvk}_{\mathsf{Id}_1})^{t_{21}} \cdot (\mathsf{Pvk}_{\mathsf{Id}_2})^{t_{22}} \qquad & t_{21} &= 0, \quad t_{22} = -t_{11}\ , \end{aligned}$$

and $\forall s,\ \mathbf{f}_{\mathsf{Pub}}\,\big(\mathsf{Id}_1,\ \bot,\ \bot,\ (g_{a_1}\, g^{H(\mathsf{Id}_1)})^s,\ \boldsymbol{d}'_1\big) \cdot \mathbf{f}_{\mathsf{Pub}}\,\big(\mathsf{Id}_2,\ \bot,\ \bot,\ (g_{a_2}\, g^{H(\mathsf{Id}_2)})^s,\ \boldsymbol{d}'_2\big) = \mathbf{e}(g^{(a_1+H(\mathsf{Id}_1))\,s},\ \hat{g}^{a_2+H(\mathsf{Id}_2)}) \cdot \mathbf{e}(g^{(a_2+H(\mathsf{Id}_2))\,s},\ \hat{g}^{-a_1-H(\mathsf{Id}_1)}) = \mathbf{e}(g, \hat{g})^0 = 1$, as required.

Lemma 4. *SK-IBE is $(q, \ell, \tau, \epsilon)$-ParSim-IND-ID-CPA secure in any bilinear context that satisfies the Decision (q', τ', ϵ')-BDHI assumption with $q' > q\,\ell$ and $\tau' < \tau - \Theta(q^2\,\ell^2)$, in the random oracle model, where $\epsilon'/\epsilon \geq \prod_{i=1}^{\ell} Q_i$, where Q_i is the number of adversarial queries to the random oracle that hashes the identities in the i-th IBE subsystem.*

Notice that the above lemma pertains to a full adaptive-identity, parallel simulation attack. The security is not tight, however, and the security losses mount exponentially with the number of IBE subsystems in the experiment.

Proof. The security proof is similar to (and a simpler version of) the proof of Lemma 2.

4.3 The Case of the Gentry IBE

The ambiguity of Gentry's IBE as an exponent inversion candidate presents an intriguing open problem. Recall from [10] that it uses a powerful security reduction that gives it tight security under adaptive-identity attacks, albeit under a strong assumption. On the one hand, the Gentry IBE has much in common with the exponent inversion family, such as the use of session keys $\mathbf{e}(g, \hat{h})^s$ that do not involve the master secret. On the other hand, the scheme uses two generators, g

and $\hat{h}$, chosen at random by the master key generator. The security proof breaks when both g and $\hat{h}$ are fixed externally, or even when chosen randomly but reused across parallel instances in the sense of Section 3.2. Thus, Gentry-IBE currently fails the exponent inversion litmus test that session keys be of the form v^s for fixed v; it remains open whether this can be remedied using a different proof.

Since the HIBE transformation we describe next preserves adaptive-identity security, extending Gentry's proof to work in the exponent inversion setting would resolve the long-standing problem of realizing fully secure HIBE for broad and deep hierarchies. Meanwhile, the very existence of such schemes remains an open problem.

Remark 1 (Asymmetric Implementations)
Lemmas 1 and 3 tacitly assume that for each element g, $g_a = g^a$, $g_b = g^b \in \mathbb{G}$ published in Pub, the corresponding element $\hat{g}$, $\hat{g}_a = \hat{g}^a$, $\hat{g}_b = \hat{g}^b \in \hat{\mathbb{G}}$ is made available for the creation of $\boldsymbol{d}'_1$ and $\boldsymbol{d}'_2$. This is automatically true if we assume that $\mathbb{G} = \hat{\mathbb{G}}$, as was the case in the original descriptions of BB_2 [2] and SK [7,17]. Otherwise, the relevant elements will need to be published explicitly, *e.g.*, in the public key, which is harmless to the security of any scheme that was already secure under the assumption that $\mathbb{G} = \hat{\mathbb{G}}$.

5 Generic Constructions

Let an abstract scheme $\mathsf{IBE} = (\mathsf{IBE.Setup}, \mathsf{IBE.Extract}, \mathsf{IBE.Encrypt}, \mathsf{IBE.Decrypt})$ with "parallel" semantic security against selective-identity chosen-plaintext attacks, that has an appropriate linear structure as above. We show how to turn it into generalizations of IBE that are semantically secure against (the appropriate notion of) selective-identity chosen-plaintext attacks.

5.1 Hierarchical Identities

In the HIBE primitive [13,11], identities are arranged in a hierarchy, and the private keys can be derived per the hierarchy without involving the global master secret. HIBE is essentially a delegation mechanism with a single root (the private key generator). We construct such a scheme generically as follows.

$\mathsf{HIBE.Setup}(L)$**.** Given a security parameter and the desired number L of levels in the hierarchy:

1. Create bilinear group parameters, $\mathbf{e}, g, \hat{g}, v$, at the desired level of security. Also pick an ephemeral shared random seed ω which is kept secret.
2. Generate L sets of IBE master key pairs with common bilinear parameters, $\mathbf{e}, g, \hat{g}, v$, by making L calls to setup $(\mathsf{IBE.Msk}_i, \mathsf{IBE.Pub}_i) \leftarrow \mathsf{IBE.Setup}(\mathbf{e}, g, \hat{g}, v, \omega)$ for $i = 1, ..., L$.
3. Select L collision-resistant hash functions (or UOWHFs) from vectors of IBE identities to single identities, $H_i : \mathcal{I}^i \rightarrow \mathcal{I}$ for $i = 1, ..., L$, where $\mathcal{I}$ is the domain of IBE identities.

4. Output the HIBE master key pair:

$$\mathsf{HIBE.Msk} = (\mathsf{IBE.Msk}_1,\ ...,\ \mathsf{IBE.Msk}_L)\ ,$$
$$\mathsf{HIBE.Pub} = (\mathsf{IBE.Pub}_1,\ ...,\ \mathsf{IBE.Pub}_L,\ H_1,\ ...,\ H_L)\ .$$

$\mathsf{HIBE.Extract}(\mathsf{Msk}, \mathsf{Id})$. Given $\mathsf{HIBE.Msk}$ and a target identity $\mathsf{Id} = (I_1, ..., I_\ell)$ at level $\ell \leq L$:

1. $\forall i = 1, ..., \ell$, let $h_i = H_i(I_1, ..., I_i)$ be the hash of the first i components.
2. $\forall i = 1, ..., \ell$, extract an IBE key $(h_i, R_i, \boldsymbol{d}_i) \leftarrow \mathsf{Extract}(\mathsf{IBE.Msk}_i, h_i)$.
3. Select $r_1, ..., r_\ell \in \mathbb{F}_p$ under the constraint that $\sum_{i=1}^{\ell} r_i = 1 \pmod p$.
4. Output the HIBE private key:

$$\mathsf{HIBE.Pvk}_{\mathsf{Id}} = ((I_1, R_1, \boldsymbol{d}_1^{r_1}),\ ...,\ (I_\ell, R_\ell, \boldsymbol{d}_\ell^{r_\ell}))\ .$$

Observe that all the components of the private key are bound to each other via the constraint $\sum_{i=1}^{\ell} r_i = 1 \pmod p$. Without it, the key would be utterly random and therefore useless. The mutual binding of the components also ensures that private keys given to different users are impervious to collusion attacks.

$\mathsf{HIBE.Derive}(\mathsf{Pvk}_{\mathsf{Id}}, I')$. Given $\mathsf{HIBE.Pvk}_{\mathsf{Id}}$ for an ℓ-level HIBE "parent" identity Id with $\ell < L$, and an IBE identity I' to act as the $(\ell + 1)$-th component of the HIBE "child" identity:

1. Decompose $\mathsf{HIBE.Pvk}_{\mathsf{Id}}$ as a list of triples $(I_i, R_i, \boldsymbol{d}_i)$ for $i = 1, ..., \ell$. Let also $I_{\ell+1} = I'$.
2. For each $i = 1, ..., \ell + 1$, let $h_i = H_i(I_1, ..., I_i)$ be the hash of the first i components.
3. For each $i = 1, ..., \ell$:
   (a) Find two vectors $\boldsymbol{d}'_{1,i}$ and $\boldsymbol{d}'_{2,i}$ that satisfy Property 2 for $\mathsf{Id}_1 = h_i$ and $\mathsf{Id}_2 = h_{i+1}$ (and the auxiliary R_i and R_{i+1}) relative to the public keys $\mathsf{IBE.Pub}_i$ and $\mathsf{IBE.Pub}_{i+1}$.
   (b) Select $r_i \in \mathbb{F}_p^\times$ and observe that $(\boldsymbol{d}'_{1,i})^{r_i}$ and $(\boldsymbol{d}'_{2,i})^{r_i}$ also satisfy Property 2.
4. For $i = 1, ..., \ell + 1$, define $\boldsymbol{d}''_i = \begin{cases} (\boldsymbol{d}'_{1,1})^{r_1} & \text{if } i = 1 \\ (\boldsymbol{d}'_{2,i-1})^{r_{i-1}}\,(\boldsymbol{d}'_{1,i})^{r_i} & \text{if } 2 \leq i \leq \ell \\ (\boldsymbol{d}'_{2,\ell})^{r_\ell} & \text{if } i = \ell + 1 \end{cases}$.
5. Output the HIBE private key:

$$\mathsf{HIBE.Pvk}_{\mathsf{Id}'} = \big((I_1, R_1, \boldsymbol{d}_1 \cdot \boldsymbol{d}''_1),\ ...,\ (I_\ell, R_\ell, \boldsymbol{d}_\ell \cdot \boldsymbol{d}''_\ell),\ (I_{\ell+1}, R_{\ell+1}, \boldsymbol{d}''_{\ell+1})\big)$$

Notice that the derived private key is fully randomized (its distribution is the same as if it had been created by $\mathsf{HIBE.Extract}$), it will decrypt correctly (because of Property 2), and its creation required only the parent private key and not the master key.

$\mathsf{HIBE.Encrypt}(\mathsf{Pub}, \mathsf{Id}, \mathsf{Msg})$. Given $\mathsf{HIBE.Pub}$, an ℓ-level identity $\mathsf{Id} = (I_1, ..., I_\ell)$ where $\ell \leq L$, and a message $\mathsf{Msg} \in \mathbb{G}_t$:

1. Pick a random exponent $s \in \mathbb{F}_p$.

2. $\forall i = 1, ..., \ell$, let $h_i = H_i(I_1, ..., I_i)$ be the hash of the first i components.
3. $\forall i = 1, ..., \ell$, use s to construct an IBE ciphertext $\mathsf{Ctx}_i = (h_i, S_i, c_0, \boldsymbol{c}_i) \leftarrow \mathsf{Encrypt}(\mathsf{IBE.Pub}_i, h_i, \mathsf{Msg}, s)$.
4. Output the HIBE ciphertext:

$$\mathsf{HIBE.Ctx} = ((h_1, ..., h_\ell),\ c_0,\ (S_1, ..., S_\ell),\ (\boldsymbol{c}_1, ..., \boldsymbol{c}_\ell)) \ .$$

Notice that $c_0 = \mathsf{Msg} \cdot v^s$ is the same in all the IBE ciphertexts.

$\mathsf{HIBE.Decrypt}(\mathsf{Pub}, \mathsf{Pvk}_{\mathsf{Id}}, \mathsf{Ctx})$**.** Given the public key $\mathsf{HIBE.Pub}$, a private key $\mathsf{Pvk}_{\mathsf{Id}} = (\mathsf{Pvk}_1, ..., \mathsf{Pvk}_\ell)$ for some hierarchical identity, and a ciphertext $\mathsf{Ctx} = ((h_1, ..., h_\ell), c_0, (S_1, ..., S_\ell), (\boldsymbol{c}_1, ..., \boldsymbol{c}_\ell))$ for the same identity:

1. $\forall i = 1, ..., \ell$, assemble $\mathsf{Ctx}_i = (h_i, 1, S_i, \boldsymbol{c}_i)$, using $1 \in \mathbb{G}_t$ in lieu of c_0.
2. $\forall i = 1, ..., \ell$, IBE-decrypt $v_i \leftarrow \mathsf{IBE.Decrypt}(\mathsf{IBE.Pub}_i, \mathsf{Pvk}_i, \mathsf{Ctx}_i)$.
3. Output the decrypted plaintext:

$$\mathsf{Msg} = c_0 \cdot \prod_{i=1}^{\ell} v_i \ .$$

By Property 1, we know that $v_i = v^{-s\, r_i}$ provided that the algorithm inputs are as expected. Since $\sum_i r_i = 1$, we obtain the desired result.

The collision-resistant hash functions $H_1, ..., H_L$ serve to enforce the "inheritance" requirement that identity components of higher index be dependent on the components of lower index. The hash functions do this by creating a precedence ordering over the indices in a construction that would otherwise be indifferent to it. The schemes we build next have no such requirement.

The above construction is quite efficient. If we instantiate it using BB_2 or SK, we respectively obtain two HIBE systems that only require ℓ pairings for decryption at level ℓ, which is marginally faster than most previously known HIBE systems [11,2,3]. The specialized construction from [3] offers faster decryption for identities of depth $\ell \geq 3$.

We can prove selective-identity security of the scheme if the underlying scheme meets the weaker version of "parallel" selective-identity IBE security (from Definition 2).

Theorem 1. *The generic* HIBE *scheme is* $(q, \ell, \tau, \epsilon)$*-IND-sHID-CPA secure [5] provided that the underlying* IBE *scheme is a Linear IBE that satisfies* $(q, \ell, \tau', \epsilon)$*-Par-IND-sID-CPA security for some* $\tau' \approx \tau$*.*

We have essentially the same result in the adaptive-identity models.

Corollary 1. *The generic* HIBE *scheme is* $(q, \ell, \tau, \epsilon)$*-IND-HID-CPA secure [11] provided that the underlying* IBE *scheme is a Linear IBE that satisfies* $(q, \ell, \tau', \epsilon)$*-Par-IND-ID-CPA security for some* $\tau' \approx \tau$*.*

5.2 Fuzzy Identities

In the Fuzzy IBE primitive [16], private keys and ciphertexts pertain to multiple identities (or attributes) at once, and decryption is predicated on meeting certain threshold of matching attributes. The collusion resistance property stipulates that private keys containing different sets of attributes cannot be combined to obtain a larger set than any of them provided by itself.

Two versions of the primitive are defined in [16]: a "small universe" version which supports an enumerated set of possible attributes, and a "large universe" version, where exponentially many attributes are representable but only a constant number at a time. In both versions the attributes are boolean (either present or absent), which we call "small domain".

Here, we give a "large domain" generalization of "small universe" Fuzzy IBE, where the enumerated attributes are now key/value pairs that range in all of $\mathbb{F}_p$. This could be useful in applications of Fuzzy IBE that require non-boolean attributes, such as a biometric system with attributes such as the height of a person.

The small-universe, large-domain, generic Fuzzy IBE construction is as follows.

$\mathsf{FuzzyIBE.Setup}(n)$**.** Given a security parameter, and the number n of attribute types to support:

1. Create bilinear group parameters, $\mathbf{e}, g, \hat{g}, v$, at the desired level of security, and a secret random string ω.
2. Generate n independent IBE master key pairs with shared bilinear parameters, $\mathbf{e}, g, \hat{g}, v$, by executing setup n times, $(\mathsf{IBE.Msk}_i,\ \mathsf{IBE.Pub}_i) \leftarrow \mathsf{IBE.Setup}(\mathbf{e}, g, \hat{g}, v, \omega)$ for $i = 1, ..., n$.
3. Output the Fuzzy IBE master key pair:

$$\mathsf{FuzzyIBE.Msk} = (\mathsf{IBE.Msk}_1,\ ...,\ \mathsf{IBE.Msk}_n)\ ,$$
$$\mathsf{FuzzyIBE.Pub} = (\mathsf{IBE.Pub}_1,\ ...,\ \mathsf{IBE.Pub}_n)\ .$$

$\mathsf{FuzzyIBE.Extract}(\mathsf{Msk}, \mathsf{Id}, t)$**.** On input a master key $\mathsf{FuzzyIBE.Msk}$, a vector $\mathsf{Id} = (I_1, ..., I_n)$ of (positionally sensitive) attributes $I_i \in \mathbb{F}_p$, and a threshold parameter t with $1 \le t \le n$:

1. Pick $f_1, ..., f_{t-1} \in \mathbb{F}_p$ and let $f(x) = 1 + \sum_{i=1}^{t-1} f_i\, x^i$ of degree $t-1$. Note that $f(0) = 1$.
2. $\forall i = 1, ..., n$, extract an IBE key $(I_i, R_i, \boldsymbol{d}_i) \leftarrow \mathsf{Extract}(\mathsf{IBE.Msk}_i, I_i)$,
3. Output the Fuzzy IBE private key:

$$\mathsf{FuzzyIBE.Pvk}_{\mathsf{Id}} = \Big(t,\ (I_1,\ R_1,\ \boldsymbol{d}_1^{f(1)}),\ ...,\ (I_n,\ R_n,\ \boldsymbol{d}_n^{f(n)})\Big)\ .$$

$\mathsf{FuzzyIBE.Encrypt}(\mathsf{Pub}, \mathsf{Id}, \mathsf{Msg})$**.** On input a public key $\mathsf{FuzzyIBE.Pub}$, a vector $\mathsf{Id} = (I_1, ..., I_n)$ of (positionally sensitive) attributes $I_i \in \mathbb{F}_p$, and a message $\mathsf{Msg} \in \mathbb{G}_t$:

1. Pick a random exponent $s \in \mathbb{F}_p$.
2. For all $i = 1, ..., n$, build an IBE ciphertext $\mathsf{Ctx}_i = (I_i, S_i, c_0, \boldsymbol{c}_i) \leftarrow \mathsf{Encrypt}(\mathsf{IBE.Pub}_i, I_i, \mathsf{Msg}, s)$.

3. Output the Fuzzy IBE ciphertext (using $c_0 = \mathsf{Msg} \cdot v^s$ common to all IBE ciphertexts):

$$\mathsf{FuzzyIBE.Ctx} = (\mathsf{Id},\ c_0,\ (S_1, ..., S_n),\ (\boldsymbol{c}_1, ..., \boldsymbol{c}_n)) \ .$$

$\mathsf{FuzzyIBE.Decrypt}(\mathsf{Pub}, \mathsf{Pvk}_{\mathsf{Id}}, \mathsf{Ctx})$**.** Given $\mathsf{FuzzyIBE.Pub}$, a private key $\mathsf{Pvk}_{\mathsf{Id}}$, and a ciphertext Ctx:

1. Determine t attributes $I_{i_1}, ..., I_{i_t}$ that appear in both $\mathsf{Pvk}_{\mathsf{Id}}$ and Ctx in matching positions.
   (a) If there are fewer than t "key/value" matches, then output $\perp$ and halt.
   (b) Else, select any t matching attributes $I_{i_1}, ..., I_{i_t}$ and define $T = \{i_1, ..., i_t\}$.
2. For $j = 1, ..., t$:
   (a) Extract the IBE private key $(I_{i_j}, R_{i_j}, \boldsymbol{d}_{i_j})$ from $\mathsf{Pvk}_{\mathsf{Id}}$ and call it Pvk_j.
   (b) Assemble the IBE ciphertext $(I_{i_j}, 1, S_{i_j}, \boldsymbol{c}_{i_j})$ from Ctx and call it Ctx_j.
   (c) Let $\Lambda_{T,i}(x) = \prod_{i' \in T \setminus \{i\}} \frac{x-i'}{i-i'}$ be the Lagrange interpolation coefficients from T to x.
   (d) Perform the IBE decryption $v_j \leftarrow \mathsf{IBE.Decrypt}(\mathsf{IBE.Pub}_j, \mathsf{Pvk}_j, \mathsf{Ctx}_j)$.
3. Output the plaintext:

$$\mathsf{Msg} = c_0 \cdot \prod_{j=1}^{t} {v_j}^{\Lambda_{T,i_j}(0)} \ .$$

By Property 1, we know that $v_j = v^{-s\, f(i_j)}$ if the inputs to the algorithm are as expected. The result follows by using Lagrange polynomial interpolation, $\sum_j f(i_j)\, \Lambda_{T,i_j}(0) = f(0) = 1$, "in the exponent".

The efficiency of the scheme is comparable to that of (the "small universe" version of) [16] when instantiated with BB_2 or SK, even though this is a "large domain" construction.

Theorem 2. *The generic* FuzzyIBE *scheme is* (q, n, τ, ϵ)*-IND-sFuzID-CPA secure [16] provided that the base* IBE *scheme is a Linear IBE with* (q, n, τ', ϵ)*-ParSim-IND-sID-CPA security for* $\tau' \approx \tau$*.*

5.3 Attribute-Based Encryption

Attribute-based encryption (ABE) is a powerful generalization of Fuzzy IBE that was recently proposed in [12]. Instead of allowing decryption conditionally on the satisfaction of a single threshold gate (whose inputs are the matching attributes in the ciphertext and the key), ABE allows the condition to be defined by a tree of threshold gates. The construction given in [12] generalizes the Fuzzy IBE construction of [16] in the commutative blinding approach, and is based on the use of not one but multiple interpolation polynomials $f(x)$, each of which applies to a subset of the input attributes. The degrees of the random polynomials and their inputs determine the access structure in the ABE scheme; in Key-Policy (KP) ABE, they are chosen by the authority.

Our generic framework can mirror the KP-ABE construction of [12], in the same way that our Fuzzy IBE construction retains the structure of the construction in [16]. The main difference is that, since our method is to build an independent instance of the underlying IBE for each attribute, we obtain a "large domain" generalization of ABE, with attributes as key/value pairs instead of booleans.

5.4 Multiple Independent Key Generators

Our generic construction immediately generalizes to the case of multiple independent key generators, which can be useful in many applications. For example, when using Fuzzy IBE for encrypting under someone's biometric readings, one may wish to use one set of attributes constructed from fingerprints and another from iris scans, and require a combination of both to decrypt. It is quite possible in this scenario that the authority issuing fingerprint-based private keys would be different than the one issuing keys based on iris scans.

Depending on the nature of the underlying IBE system, it is possible to base our generic Fuzzy IBE construction on independent subsystems that share only the bilinear groups and generators, thereby facilitating their setup. Whether independent setup is allowed (in a commonly agreed upon bilinear group), depends on the use that the $\mathsf{IBE.Setup}$ function makes of the common random ω. For instance, since the BB_2 and SK schemes achieve our notion of parallel simulation security without using ω, they are suitable for building a multi-authority system without shared secret.

The only remaining difficulty lies in the final assembly of private keys given to the users, because the separate authorities will have to agree on a suitable random polynomial $f(x)$ in order to create a new key. Some amount of coordination between the servers will be required (possibly mediated by the key recipient), but since the polynomial to be agreed upon is ephemeral and decoupled from the master keys, this is an orthogonal problem that can be solved in many standard ways. In particular, Chase [6] showed how to construct a multi-authority attribute-based scheme, in the commutative blinding framework, where multiple authorities can vouch for separate attributes under the auspices of a central authority that handles the sharing of ephemerals.

6 Conclusion

We have shown that the family of identity-based encryption schemes based on the exponent inversion principle can be leveraged into building more powerful systems. We first presented an abstraction to capture a number of useful properties shared by such schemes. We then showed how to use this abstraction to construct generalizions of IBE. We described Hierarchical and Fuzzy IBE as concrete examples, as each of them illustrates a specific feature of exponent inversion schemes, but many other generalizations are possible based on the same abstraction. Our approach is fairly lightweight and is also compatible with decentralized authorities.

These results have practical implications, since the few known exponent inversion IBE schemes tend to be marginally more efficient than competing constructions, although they require stronger complexity assumptions. Our formalism has no effect on these benefits and drawbacks, but it extends the range of applicability of the relevant schemes.

Acknowledgements

The author thanks Michel Abdalla and Brent Waters for discussions on some of these ideas, and anonymous referees for valuable comments.

References

1. Michel Abdalla, Dario Catalano, Alexander W. Dent, John Malone-Lee, Gregory Neven, and Nigel P. Smart. Identity-based encryption gone wild. In *Proceedings of ICALP 2006*, volume 4051 of *Lecture Notes in Computer Science*, pages 300–11. Springer-Verlag, 2006.
2. 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.
3. 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.
4. 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*.
5. 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*, pages 207–22. Springer-Verlag, 2004.
6. Melissa Chase. Multi-authority attribute based encryption. In *Proceedings of TCC 2007*, Lecture Notes in Computer Science. Springer-Verlag, 2007.
7. Liqun Chen, Zhaohui Cheng, John Malone-Lee, and Nigel P. Smart. An efficient ID-KEM based on the Sakai-Kasahara key construction. Cryptology ePrint Archive, Report 2005/224, 2005. `http://eprint.iacr.org/2005/224/`.
8. Jung Hee Cheon. Security analysis of the strong Diffie-Hellman problem. In *Advances in Cryptology—EUROCRYPT 2006*, volume 4004 of *Lecture Notes in Computer Science*, pages 1–13. Springer-Verlag, 2006.
9. Clifford Cocks. An identity based encryption scheme based on quadratic residues. In *Proceedings of the 8th IMA International Conference on Cryptography and Coding*, 2001.
10. Craig Gentry. Practical identity-based encryption without random oracles. In *Advances in Cryptology—EUROCRYPT 2006*, Lecture Notes in Computer Science. Springer-Verlag, 2006.
11. Craig Gentry and Alice Silverberg. Hierarchical ID-based cryptography. In *Proceedings of ASIACRYPT 2002*, Lecture Notes in Computer Science. Springer-Verlag, 2002.

12. Vipul Goyal, Omkant Pandey, Amit Sahai, and Brent Waters. Attribute-based encryption for fine-grained access control of encrypted data. In *ACM Conference on Computer and Communications Security—CCS 2006*, 2006.
13. 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.
14. Shigeo Mitsunari, Ryuichi Sakai, and Masao Kasahara. A new traitor tracing. *IEICE Transactions on Fundamentals*, E85-A(2):481–4, 2002.
15. David Naccache. Secure and practical identity-based encryption. Cryptology ePrint Archive, Report 2005/369, 2005. `http://eprint.iacr.org/2005/369/`.
16. Amit Sahai and Brent Waters. Fuzzy identity-based encryption. In *Advances in Cryptology—EUROCRYPT 2005*, volume 3494 of *Lecture Notes in Computer Science*. Springer-Verlag, 2005.
17. Ryuichi Sakai and Masao Kasahara. ID based cryptosystems with pairing over elliptic curve. Cryptology ePrint Archive, Report 2003/054, 2003. `http://eprint.iacr.org/2003/054/`.
18. Ryuichi Sakai, Kiyoshi Ohgishi, and Masao Kasahara. Cryptosystem based on pairing. In *Symposium on Cryptography and Information Security—SCIS 2000*, Okinawa, Japan, 2000.
19. 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.
20. 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.
21. 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.