Black Box Accountable Authority Identity-Based Encryption

Vipul Goyal[*]
Department of Computer Science
University of California, Los Angeles
vipul@cs.ucla.edu

Steve Lu[†]
Department of Mathematics
University of California, Los Angeles
stevelu@math.ucla.edu

Amit Sahai[‡]
Department of Computer Science
University of California, Los Angeles
sahai@cs.ucla.edu

Brent Waters[§]
Department of Computer Science
University of Texas at Austin
bwaters@cs.utexas.edu

Nov 03, 2008

Abstract

A well-known concern in the setting of identity based encryption is that the PKG is all powerful and has to be completely trusted. To mitigate this problem, the notion of Accountable Authority Identity-Based Encryption (A-IBE) was recently introduced by Goyal. Goyal provided constructions to realize the notion of A-IBE only in the white box and weak black box models. However, the security guarantees provided by these models fall short of those required in practice.

In this paper, we resolve the main open question left in Goyal's work by providing a construction of a fully black box A-IBE system. Our construction is based on the Decisional Bilinear Diffie-Hellman assumption and uses techniques from key policy attribute based encryption.

1 Introduction

Shamir [Sha84] introduced the concept of identity based encryption (IBE) as an approach to simplify public key and certificate management in a public key infrastructure (PKI). The first practical and fully functional IBE scheme was proposed by Boneh and Franklin [BF01] in the random oracle model. Following that work, a rapid development of identity based PKI has taken place (see [CHK03, BB04a, BB04b, BBG05, Wat05, Gen06] and the references therein).

In an IBE system, the public key of a user may be an arbitrary string like an e-mail address or other identifier. Of course, users are not capable of generating a private key for an identity themselves. For this reason, there is a trusted party called the private key generator (PKG) who does the system setup. To obtain a private key for his identity, a user would go to the PKG and

[*] Supported in part by a Microsoft Research Fellowship and the grants of the third author mentioned below.

[†] Supported by NSF grants 0430254, 0716835, 0716389, and NSF VIGRE grant DMS-0502315.

[‡] Supported in part from grants from the NSF ITR and Cybertrust programs (including grants 0627781, 0456717, and 0205594), a subgrant from SRI as part of the Army Cyber-TA program, an equipment grant from Intel, an Alfred P. Sloan Foundation Fellowship, and an Okawa Foundation Research Grant.

[§] Supported by NSF CNS-0749931, CNS-0524252, CNS-0716199; the U.S. Army Research Office under the CyberTA Grant No. W911NF-06-1-0316; and the U.S. Department of Homeland Security under Grant Award Number 2006-CS-001-000001.

prove his identity. The PKG would then generate the appropriate private key and pass it on to the user.

Such a setting, however, leads to the following problem. Since the PKG is able to compute the private key corresponding to any identity, it has to be completely trusted. The PKG is free to engage in malicious activities without any risk of being confronted in a court of law. The malicious activities could include: decrypting and reading messages meant for any user, or worse still: generating and distributing private keys for any identity. This, in fact, has been cited as a reason for the slow adoption of IBE despite its nice properties in terms of usability. It has been argued that due to the inherent key escrow problem, the use of IBE is restricted to small and closed groups where a central trusted authority is available [ARP03, LBD$^+$04, Gen03].

Accountable Authority Identity Based Encryption. Goyal [Goy07] introduced the notion of Accountable Authority Identity Based Encryption (A-IBE) as a new approach to mitigate the above problem of trust. Informally speaking, the simplified view of the approach is as follows:

1. In the IBE scheme, there will be an exponential (or super-polynomial) number of possible decryption keys corresponding to every identity ID.

2. Given one decryption key for an identity, it is intractable to find any other.

3. A users gets the decryption key corresponding to his identity from the PKG using a secure *key generation protocol.* The protocol allows the user to obtain a single decryption key d_{ID} for his identity *without letting the PKG know which key he obtained.*

4. Now if the PKG generates a decryption key d'_{ID} for that identity for malicious usage, with all but negligible probability, it will be different from the key d_{ID} which the user obtained. Hence the key pair $(d_{\mathsf{ID}}, d'_{\mathsf{ID}})$ is a cryptographic proof of malicious behavior by the PKG (since in normal circumstances, only one key per identity should be in circulation).

Thus, this approach severely restricts the PKG as far as malicious distribution of the private keys is concerned. The knowledge of the key d'_{ID} enables an entity E to go to the honest user U (with identity ID and having key d_{ID}) and together with him, sue the PKG by presenting the pair $(d'_{\mathsf{ID}}, d_{\mathsf{ID}})$ as a proof of fraud.

The Right Model for A-IBE. Goyal [Goy07] presented two constructions towards achieving the notion of A-IBE. However, his security proofs could only provide a limited guarantee: that the PKG cannot maliciously distribute a *well-formed* decryption key. As noted by Goyal, while this is a starting point, these kind of "white box" guarantees are completely insufficient in practice. The PKG could, for example, release an obfuscated program (or simply a decryption box) which successfully decrypts the ciphertexts and yet does not contain the decryption key in any canonical form. Furthermore trivial constructions can satisfy the "white box" security guarantee and clearly be insecure in practice: For instance, if we take any IBE scheme and force the user to also obtain a blind signature from the PKG on a random message (which is checked by the decryption algorithm), this would already satisfy the "white box" security definition. Obviously this scheme would be broken in practice since the PKG could release a box that decrypts for an identity but doesn't contain a signature (and therefore isn't well-formed).

Goyal also showed how to extend his constructions to achieve security guarantees according to a *weak black box model* in which, a malicious PKG has to output a decryption box just after running the key generation protocol with the honest user. However, this security model is also insufficient. It is conceivable that the PKG (or a party colluding with the PKG) could trick the user into decrypting a maliciously prepared ciphertext and see the result (in an attempt to learn more information about the decryption key which the user selected during the key generation protocol). Indeed, if such decryption queries are allowed, the weak black box scheme of [Goy07] can be compromised with only a small number of queries.

In what we call the *full black box model*, the PKG is given access to decryption queries and no assumptions are made regarding how the decryption box works. In particular, just by observing the input/output behavior of the given decryption box, a judge should be able to decide if the box was created by the actual user or by a dishonest PKG. The construction of an A-IBE scheme in the full black box model - the model which we believe provides the "right" real world security guarantees - was left as an important open problem in [Goy07].

Our Contribution. In this work, we resolve the above open question and provide a construction of (fully black box) A-IBE based on the Decisional Bilinear Diffie-Hellman (DBDH) assumption. The main technical difficulty is resolving the tension between the information leaked as part of the decryption queries and the success of the exoneration procedure. That is, on one hand we require that during regular operation, the outcome of the decryption of a ciphertext should not leak information about the which decryption key the user selected. On the other hand, during exoneration, a judge should be able to extract enough information about the user key selection from the black box in order to determine that the user could not have generated the box (and therefore the PKG must be at fault).

The key idea in our construction is to first design a scheme having *imperfect completeness*. That is, for every possible decryption key, there exist a negligible fraction of (valid) ciphertexts which cannot be decrypted by this key. On one hand, this property is helpful in tracing: a judge (given the decryption box and the decryption key of the user) can probe the box exactly on those ciphertexts which the user key should not be able to decrypt. On the other hand, this does not seem to create a problem for decryption queries since the chance that a malicious PKG will hit such a ciphertext (with a polynomial number of queries) is negligible.

We construct such a scheme using ideas from key-policy attribute-based encryption (KP-ABE) [SW05, GPSW06]. Very roughly, we label each ciphertext as well as a decryption key with a list of dummy attributes. There exists a policy which decides whether or not a ciphertext will be decrypted by a particular private key. To achieve statistical completeness, for every decryption key, all but a negligible fraction of ciphertexts will satisfy this policy.

While we take the approach of constructing such an A-IBE scheme with imperfect completeness, we will later show how to run a "complementary" system in parallel with such a scheme so that the resulting system also achieves the property of perfect completeness (while also maintaining the functionality of our tracing procedure).

Related Work. The idea of an accountable authority IBE was introduced by Goyal [Goy07] as a mitigation to the problem of trust in the PKG. Au et. al. [AHL$^+$08] extended this work by introducing a retrieval algorithm which causes the PKG's master secret key to be revealed if more than one key per identity is released. The motivation is to penalize the PKG without the users having to go to the court. However, this work is orthogonal to ours since their security proofs are in the white box model of security (as opposed to black box or even weakly black box) and require the PKG to release a well formed decryption key. To our knowledge, these are the only known mitigation approaches without using multiple PKGs. On the multiple PKGs side, Boneh and Franklin [BF01] proposed an efficient approach to make the PKG distributed in their scheme using techniques from threshold cryptography. Lee *et al* [LBD$^+$04] proposed a variant of this approach using multiple key privacy agents (KPAs).

Organization. In Section 2 we review background information pertaining to our constructions. In Section 3 we formally define the model for an accountable authority identity based encryption scheme. In Section 4 we give a construction of such a scheme and prove that it satisfies the definitions in our model. The construction will have statistical completeness and we describe in Appendix B how to achieve perfect completeness. Finally, in Section 5 we conclude with interesting open problems for future work.

2 Preliminaries

2.1 Bilinear Maps

We present a few facts related to groups with efficiently computable bilinear maps.

Let $\mathbb{G}_1$ and $\mathbb{G}_2$ be two multiplicative cyclic groups of prime order p. Let g be a generator of $\mathbb{G}_1$ and e be a bilinear map, $e : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$. The bilinear map e has the following properties:

1. **Bilinearity:** For all $u, v \in \mathbb{G}_1$ and $a, b \in \mathbb{Z}_p$, we have $e(u^a, v^b) = e(u, v)^{ab}$.
2. **Non-degeneracy:** $e(g, g) \neq 1$.

We say that $\mathbb{G}_1$ is a bilinear group if the group operation in $\mathbb{G}_1$ and the bilinear map $e : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ are both efficiently computable. Notice that the map e is symmetric since $e(g^a, g^b) = e(g, g)^{ab} = e(g^b, g^a)$.

2.2 Complexity Assumptions

We state our complexity assumptions below.

Decisional Bilinear Diffie-Hellman (DBDH) Assumption Let $a, b, c, z \in \mathbb{Z}_p$ be chosen at random and g be a generator of $\mathbb{G}_1$. The Decisional BDH assumption [BB04a, SW05] is that no probabilistic polynomial-time algorithm $\mathcal{B}$ can distinguish the tuple $(A = g^a, B = g^b, C = g^c, e(g, g)^{abc})$ from the tuple $(A = g^a, B = g^b, C = g^c, e(g, g)^z)$ with more than a negligible advantage. The advantage of $\mathcal{B}$ is

$$\left|\Pr[\mathcal{B}(A, B, C, e(g, g)^{abc}) = 0] - \Pr[\mathcal{B}(A, B, C, e(g, g)^z) = 0]\right|$$

where the probability is taken over the random choice of the generator g, the random choice of a, b, c, z in $\mathbb{Z}_p$, and the random bits consumed by $\mathcal{B}$.

2.3 Fully Simulatable k-out-of-n Oblivious Transfer

Informally speaking, a k-out-of-n oblivious transfer protocol (see [EGL85]) allows a receiver to choose and receive exactly k of the n string from the sender, such that the remaining strings are hidden from the receiver and the choice of the receiver is hidden from the sender. We require the oblivious transfer protocol to be fully simulatable (i.e. satisfy the standard Ideal/Real world definition of security, see Canetti [Can00] for more details). Various efficient constructions of k-out-of-n oblivious transfer are known based on specific assumptions such as DBDH and DDH [Lin08, GH07, CNS07].

2.4 Attribute Based Encryption

The notion of attribute-based encryption (ABE) was introduced by Sahai and Waters [SW05] who considered a user having a set of attributes ($\mathcal{I}$) associated to him or her. Similarly, when encrypting, the ciphertext also has a set of attributes ($\mathcal{J}$) associated to it. At a high level view, this scheme allowed a PKG to distribute user keys such that a user can only decrypt when their set of attributes "properly matched" the set of attributes in the ciphertext. The original Sahai-Waters work gave constructions for threshold policies (i.e. $|\mathcal{I} \cap \mathcal{J}| > \tau$ for some threshold τ). This was further generalized by Goyal et. al. [GPSW06] with the introduction of key policy attribute based encryption (KP-ABE) supporting advanced policies including those representable by trees of threshold functions. Our constructions are partially based off of these schemes; we will also have sets associated to the user and the ciphertext, and it will be convenient to keep the notion of "attributes" in mind. We refer the reader to [GPSW06] for the details of the construction of an attribute-based encryption scheme.

3 The Definitions and the Model

An Accountable Authority Identity Based Encryption (A-IBE) scheme consists of five components. These definitions are adapted from Goyal [Goy07] with a critical enhancement to account for fully black-box tracing.

Setup: There is a randomized algorithm $\mathsf{Setup}(\lambda)$ that takes as input the security parameter λ, and it outputs the public parameters PK and a master key MK.
Key Generation Protocol: There is an interactive protocol KeyGen between the public parameter generator PKG and the user U. The common input to PKG and U are: the public parameters PK and the identity ID (of U) for which the decryption key has to be generated. The private input to PKG is the master key MK. Additionally, PKG and U may use a sequence of random coin tosses as private inputs. At the end of the protocol, U receives a decryption key d_{ID} as its private output. At any time, either party may abort.
Encryption: There is a randomized algorithm $\mathsf{Encrypt}(M, \mathsf{ID}, \mathrm{PK})$ that takes as input: a message M, an identity ID, and the public parameters PK. It outputs the ciphertext C.
Decryption: There is an algorithm $\mathsf{Decrypt}(C, \mathsf{ID}, d_{\mathsf{ID}})$ that takes as input: the ciphertext C that was encrypted under the identity ID, the decryption key d_{ID} for ID and the public parameters PK. It outputs a message M or $\perp$.
Trace: There is a randomized algorithm $\mathsf{Trace}^{\mathsf{D}}(\mathsf{ID}, d_{\mathsf{ID}}, \epsilon)$ that takes as input an identity ID, a "well-formed" decryption key d_{ID} (where "well formed" means that the decryption key passes a "key sanity check" described as part of the key generation protocol), a parameter ϵ (which must be polynomially related to λ), and has black-box access to an ϵ-useful decoder box D. It runs in time polynomial in λ and $1/\epsilon$ and outputs PKG, User, or Fail.

Loosely speaking, the idea behind the tracing algorithm is to allow an honest user to present her decryption key along with a captured decoder box (which decrypts her messages) to a judge to implicate the PKG of wrongdoing. At the same time, the tracing algorithm should also prevent a dishonest user from being able to falsely implicate the PKG of having created the decoder box.

To define security for an accountable authority identity based encryption system, we first define the three following games.

The IND-ID-CPA game. The IND-ID-CPA game for A-IBE is very similar to the IND-ID-CPA game for standard IBE [BF01].

- **Setup** The challenger runs the Setup algorithm of A-IBE and gives the public parameters PK to the adversary.

- **Phase 1** The adversary runs the Key Generation protocol with the challenger for several distinct adaptively chosen identities $\mathsf{ID}^1, \ldots, \mathsf{ID}^q$ and gets the decryption keys $d_{\mathsf{ID}^1}, \ldots, d_{\mathsf{ID}^q}$.

- **Challenge** The adversary submits two equal length messages m_0 and m_1 and an identity ID not equal to any of the identities' queries in Phase 1. The challenger flips a random coin b and encrypts m_b with ID. The ciphertext C is passed on to the adversary.

- **Phase 2** This is identical to Phase 1 except that the adversary is not allowed to ask for a decryption key for ID.

- **Guess** The adversary outputs a guess b' of b.

The advantage of an adversary $\mathcal{A}$ in this game is defined as $\Pr[b' = b] - \frac{1}{2}$.

We note that the above game can extended to handle chosen-ciphertext attacks in the natural way by allowing for decryption queries in Phase 1 and Phase 2. We call such a game to be the IND-ID-CCA game.

We now define two games which should model the usefulness of the tracing algorithm; any decoder box D should trace back to the person who created it.

The DishonestPKG game. The intuition behind this game is that an adversarial PKG attempts to create a decoder box which will frame the user. Both the adversary and challenger are given the security parameter λ as input. A second parameter $\epsilon = \frac{1}{poly(\lambda)}$ is also given as input. The DishonestPKG game for A-IBE is defined as follows.

- **Setup** The adversary (acting as an malicious PKG) generates and passes the public parameters PK and an identity ID on to the challenger. The challenger checks that PK and ID are well-formed and aborts if the check fails.

- **Key Generation** The challenger and the adversary then engage in the key generation protocol to generate a decryption key for the identity ID. If neither party aborts, then the challenger gets the decryption key d_{ID} as output.

- **Decryption Queries** The adversary adaptively queries ciphertexts $C_1, \ldots, C_q$ to the challenger and the challenger replies with the decrypted values.

- **Create Decoder Box** The adversary outputs a decoder box D.

Let SF denote the event that the adversary wins this game, which happens if the following two conditions hold:

- The decoder box D is ϵ-useful for ID, i.e.

$$\Pr[\mathsf{D}(\mathsf{Encrypt}(M, \mathsf{ID}, \mathrm{PK})) = M] > \epsilon$$

- The tracing algorithm incorrectly implicates the user, i.e. $\mathsf{Trace}^{\mathsf{D}}(\mathsf{ID}, d_{\mathsf{ID}}, \epsilon) = \mathsf{User}$

The advantage of an adversary $\mathcal{A}$ in this game is defined as $\Pr[SF]$ where the probability is taken over the random coins of Trace.

We note that unlike the weak black box model in Goyal [Goy07], our model includes a *decryption queries* phase where the adversary adaptively queries the challenger with a sequence of ciphertexts. This phase could potentially help the adversary deduce information about the decryption key of d_{ID} if it is able to present a *maliciously formed* ciphertext and get the challenger try to decrypt it.

The Selective-ID DishonestUser game. The intuition behind this game is that some colluding set of users $\mathsf{ID}_1, \dots, \mathsf{ID}_q$ attempts to create a decoder box which will frame the PKG. Both the adversary and challenger are given the security parameter λ as input. A second parameter $\epsilon = \frac{1}{poly(\lambda)}$ is also given as input. The Selective-ID $\mathsf{DishonestUser}$ game for A-IBE is defined as follows.

- **Select ID** The adversary announces an $\mathsf{ID}^\star$ to the challenger.
- **Setup** The challenger runs the Setup algorithm of A-IBE and sends the public parameters PK to the adversary.
- **Key Generation Queries** The adversary runs the Key Generation protocol with the challenger for several *distinct* adaptively chosen identities $\mathsf{ID}_1, \dots, \mathsf{ID}_q$ and gets the decryption keys $d_{\mathsf{ID}_1}, \dots, d_{\mathsf{ID}_q}$.
- **Create Decoder Box** The adversary outputs a decryption key $d_{\mathsf{ID}^\star}$ and a decoder box D for the identity $\mathsf{ID}^\star$ announced in the Select ID phase.

Let DF denote the event that the adversary wins this game, which happens if the following two conditions hold:

- The decoder box D is ϵ-useful for ID, i.e.

$$\Pr[\mathsf{D}(\mathsf{Encrypt}(M, \mathsf{ID}, \mathrm{PK})) = M] > \epsilon$$

- The tracing algorithm incorrectly implicates the PKG, i.e. $\mathsf{Trace}^{\mathsf{D}}(\mathsf{ID}, d_{\mathsf{ID}}, \epsilon) = \mathsf{PKG}$

The advantage of an adversary $\mathcal{A}$ in this game is defined as $\Pr[DF]$ where the probability is taken over the random coins of Trace.

We note that one can also define a full $\mathsf{DishonestUser}$ game where the adversary does not have to declare $\mathsf{ID}^\star$ in advance. Our construction is only proven secure with the Selective-ID $\mathsf{DishonestUser}$ game, and this weakening can be seen as similar to weakening of the $\mathsf{IND\text{-}ID\text{-}CPA}$ game by some previously published papers [CHK03, BB04a, SW05, GPSW06].

Definition 1 *An Accountable Authority Identity-Based Encryption scheme is secure if for any polynomial time adversary $\mathcal{A}$ and any parameter $\epsilon = \frac{1}{poly(\lambda)}$, $\mathcal{A}$ has at most a negligible advantage (in λ) in the* $\mathsf{IND\text{-}ID\text{-}CPA}$ *game, the* $\mathsf{DishonestPKG}$ *game and the Selective-ID* $\mathsf{DishonestUser}$ *game.*

4 The Main Construction

In this section, we give a construction of a secure A-IBE scheme based on the decisional BDH assumption. The construction will borrow ideas from the second construction of Goyal [Goy07] and the attribute-based encryption schemes of Sahai-Waters [SW05] and Goyal et. al. [GPSW06]. It will be helpful to keep in mind that there will be a set of attributes associated with each decryption key as well as a set of attributes associated with each ciphertext. In the context of attribute-based encryption these attributes are viewed as meaningful meta-data; however for our purposes, most of the attributes only serve as a tool to enable us to determine who is held accountable for creating a captured decoder box. For this reason, we will refer to these as the *dummy attributes* (cf. dummy attributes in Goyal [Goy07]).

4.1 Main Idea

The main idea in our construction is to create a policy on the dummy attributes in such a way that any randomly chosen decryption key can decrypt *almost all* ciphertexts. The tracing algorithm will hone in on the ciphertexts the key cannot decrypt in attempt to catch a dishonest PKG. The structure of attributes in a user key is formed as: a portion connected to the ID, and then m "parallel" repetitions each consisting of of k (out of n) dummy attributes. Thus, the user's set of attributes will loosely look like $(\mathsf{ID}, \mathcal{I}_1, \ldots, \mathcal{I}_m)$, where each $\mathcal{I}_j$ will consist of k attributes. A ciphertext will have a similar attribute structure, which we can loosely write as $(\mathsf{ID}, \mathcal{J}_1, \ldots, \mathcal{J}_m)$ where each $\mathcal{J}_j$ will also consist of k attributes. The policy can be stated as: A user can decrypt a ciphertext if and only if (the ID portion matches) AND ($\mathcal{I}_1 \cap \mathcal{J}_1$ contains at least τ attributes) AND ...AND ($\mathcal{I}_m \cap \mathcal{J}_m$ contains at least τ attributes). To enforce this policy, our construction will make use of the key-policy attribute-based encryption scheme of Goyal et. al. [GPSW06]. By appropriately choosing the number of dummy attributes k (in the decryption key and ciphertext) and the threshold τ, we guarantee that a randomly encrypted ciphertext can be decrypted with high probability (we will later present a modification of the scheme that achieves perfect completeness as well). An example is provided in Appendix A demonstrating how to appropriately choose these parameters.

Our construction will focus on satisfying the security of the DishonestPKG game and the Selective-ID security of the DishonestUser game. As for satisfying IND-ID-CPA security, we demonstrate how to combine our scheme with a IND-ID-CPA secure IBE scheme (such as the ones found in Waters [Wat05] or Gentry [Gen06]). We now present our main construction.

4.2 The Construction

$\mathbb{G}_1$ is a bilinear group of prime order p, and let g be a generator of $\mathbb{G}_1$. In addition, let $e : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ denote a bilinear map. We define the Lagrange coefficient $\Delta_{i,S}$ for $i \in \mathbb{Z}_p$ and some set $S \subset \mathbb{Z}_p$ to be

$$\Delta_{i,S}(x) := \prod_{j \in S \setminus \{i\}} \frac{x - j}{i - j}.$$

We represent the identities as strings of length ℓ (since an identity $\mathsf{ID} \in \mathbb{Z}_p$, ℓ is the number of bits required to represent an element in $\mathbb{Z}_p$). Let n and m be chosen as "deterrence" parameters: looking ahead, our proofs will show that a malicious PKG can only succeed with probability negligible in n. For the sake of our proofs, we set n to be equal to the global security parameter λ and m be super-logarithmic in n, say $m = \log^2(n)$. We shall denote the sets $\{1, \ldots, \ell\}, \{1, \ldots, n\}, \{1, \ldots, m\}$ by $[\ell], [n], [m]$, respectively, and the ith bit of the identity ID with ID_i. We furthermore fix a number of dummy attributes k that is a constant fraction of n, and a decryption threshold τ as explained above (in Appendix A, we give an example using explicit values). Our scheme is as follows:

Setup For each $i \in [\ell]$, choose two numbers $u_{i,0}$ and $u_{i,1}$ uniformly at random from $\mathbb{Z}_p$ such that all 2ℓ numbers are different. In addition, for each $i \in [n]$ and $j \in [m]$ choose a $t_{i,j}$ uniformly at random from $\mathbb{Z}_p$. Also choose a number y uniformly at random in $\mathbb{Z}_p$.

The public parameters are:

$$\mathrm{PK} = \Big[\{(U_{i,j} = g^{u_{i,j}}) : i \in [\ell], j \in \{0,1\}\}, \{(T_{i,j} = g^{t_{i,j}}) : i \in [n], j \in [m]\}, Y = e(g,g)^y, g\Big]$$

The master key is:

$$\mathrm{MK} = \Big[\{u_{i,j} : i \in [\ell], j \in \{0,1\}\}, \{(t_{i,j}) : i \in [n], j \in [m]\}, y\Big]$$

Key Generation Protocol The high level idea of our key generation protocol is to allow the user to obliviously choose which dummy attributes he wants (using a k-out-of-n oblivious transfer) on each "repetition". The attributes are represented by distinct elements in $\mathbb{Z}_p$, thus attribute 1 in the first repetition will be represented by a different element than attribute 1 in the second repetition. These repetitions are performed in parallel and will be viewed as individual components of our key. We want a policy that he can only decrypt when the ciphertext shares τ of these attributes (for each component). Additional care needs to be taken to ensure the simulatability of this protocol (which is crucial to our security proofs) while still keeping it as efficient as possible. The key generation protocol between PKG and a user U (with the identity ID) proceeds as follows.

1. U aborts if the published values in the public key are not all different.

2. PKG generates $m+1$ random numbers $y_0, \dots, y_m$ from $\mathbb{Z}_p$ such that $y_0 + \cdots + y_m = y$. We will use y_0 to tie in the identity and $y_1, \dots, y_m$ for the the dummy attribute sets.

3. PKG generates ℓ random numbers $r_1, \dots, r_\ell$ from $\mathbb{Z}_p$ such that $r_1 + \cdots + r_\ell = y_0$.

4. PKG generates m random polynomials (of degree $\tau - 1$) $q_1, \dots, q_m$ with $q_j(0) = y_j$.

5. PKG computes the key components $d_i = g^{r_i/u_{i,\mathsf{ID}_i}}$ for all $i \in [\ell]$ and sends them to U. It also computes key components $d_{i,j} = g^{q_j(i)/t_{i,j}}$ for all $i \in [n], j \in [m]$ and stores them.

6. PKG chooses random permutations $\pi_1, \dots, \pi_m \in S_n$. Looking ahead, this step will help the simulator (in the proof of security) enforce a particular choice of the dummy attributes on him. We denote $\pi = (\pi_1, \dots, \pi_m)$.

7. PKG and U then engage in m executions of a k-out-of-n oblivious transfer protocol where PKG acts as the sender and U acts as the receiver. In the jth execution, the private input of PKG is the key components $\{d_{\pi_j(i),j}\}_{i=1}^n$ and the private input of U is a set $\mathcal{I}_j$ of k randomly selected dummy attributes. The private output of U is the key component $\{\pi_j(i), d_{\pi_j(i),j}\}_{i \in \mathcal{I}_j}$.

8. PKG sends U the permutation list π. U checks if he got the right key components as per π (and aborts if the check fails).

9. U sets $d = (\{d_i\}_{i \in [\ell]}, \{(\mathcal{I}_j, \{d_{i,j}\}_{i \in \mathcal{I}_j})\}_{j \in [m]})$ and runs a key sanity check on d, which we will define. U aborts if the check fails. Finally, U sets the decryption key $d_{\mathsf{ID}} = d$.

Key Sanity Check We specifically name this subroutine of the key generation protocol for later reference in the security analysis (Section 4.4). Given a decryption key $d_{\mathsf{ID}} = (\{d_i\}_{i\in[\ell]}, \{(\mathcal{I}_j, \{d_{i,j}\}_{i\in\mathcal{I}_j})\}_{j\in[m]})$ for an identity ID, we define a (deterministic) algorithm to check the well-formedness of this key.

1. For each $j \in [m]$, let S be the first τ elements of $\mathcal{I}_j$. Verify that every point $x \in \mathcal{I}_j$ lies on the polynomial interpolated by the points in S:
$$e(d_{x,j}, T_{x,j}) \overset{?}{=} \prod_{i\in S} e(d_{i,j}, T_{i,j})^{\Delta_{i,S}(x)}$$
2. Set $Y_j = \prod_{i\in S} e(d_{i,j}, T_{i,j})^{\Delta_{i,S}(0)}$.
3. Finally, check that
$$Y \overset{?}{=} \prod_{i\in[\ell]} e(U_{i,\mathsf{ID}_i}, d_i) \prod_{j\in[m]} Y_j$$

If all of the above are verified, then the key sanity check passes, otherwise it fails.

Encryption To encrypt a message $M \in \mathbb{G}_2$ under an identity ID, choose a random value $s \in \mathbb{Z}_p$ and a subset $\mathcal{J}_j \subset [n]$ of size k for each $j \in [m]$. Compute the ciphertext C as follows.

$$C = (\{\mathcal{J}_j\}_{j\in[m]}, c = M \cdot Y^s, \{(C_i = {U_{i,\mathsf{ID}_i}}^s) : i \in [\ell]\}, \{(C_{i,j} = {T_{i,j}}^s) : j \in [m], i \in \mathcal{J}_j\})$$

The key generation for ID was set up so that if on each component $j \in [m]$ the user's dummy attributes ($\mathcal{I}_j$) intersect the ciphertext's dummy attributes ($\mathcal{J}_j$) by more than τ then the user can decrypt the message.

Decryption To decrypt the ciphertext

$$C = (\{\mathcal{J}_j\}_{j\in[m]}, c, \{C_i\}, \{C_{i,j}\})$$

using $d_{\mathsf{ID}} = (\{d_i\}_{i\in[\ell]}, \{(\mathcal{I}_j, \{d_{i,j}\}_{i\in\mathcal{I}_j})\}_{j\in[m]})$, first run a ciphertext sanity check on C, which we will define.

If the check fails, output $\perp$. Otherwise, recover the message M by selecting (for each $j \in [m]$) a set $S_j \subset \mathcal{I}_j \cap \mathcal{J}_j$ of threshold size τ and performing the following computations:

$$\begin{aligned}
&c / \prod_{i\in[\ell]} e(C_i, d_i) \prod_{j\in[m]} \prod_{i\in S_j} (e(C_{i,j}, d_{i,j}))^{\Delta_{i,S_j}(0)} \\
&= M \cdot e(g,g)^{sy} / \prod_{i\in[\ell]} e(g^{su_{i,\mathsf{ID}_i}}, g^{r_i/u_{i,\mathsf{ID}_i}}) \prod_{j\in[m]} \prod_{i\in S_j} (e(g^{st_{i,j}}, g^{q_j(i)/t_{i,j}}))^{\Delta_{i,S_j}(0)} \\
&= M \cdot e(g,g)^{sy} / e(g,g)^{sy_0} \prod_{j\in[m]} e(g,g)^{sq_j(0)} \\
&= M \cdot e(g,g)^{sy} / e(g,g)^{sy_0} \prod_{j\in[m]} e(g,g)^{sy_j} \\
&= M
\end{aligned}$$

The decryption algorithm outputs $\perp$ if there exists a j such that $|\mathcal{I}_j \cap \mathcal{J}_j| < \tau$. In Appendix A, we show an instantiation of the parameters such that this case only happens with negligible probability.

Ciphertext Sanity Check We specifically name this subroutine of the decryption algorithm for later reference in the security analysis (Section 4.4). Our ciphertext sanity check is similar to that in [Goy07]. Given a ciphertext $C = (\{\mathcal{J}_j\}_{j \in [m]}, c, \{C_i\}, \{C_{i,j}\})$ for an identity ID, we define a (deterministic) algorithm to check the well-formedness of this ciphertext. Verify that

$$e(C_i, U_{1,\mathsf{ID}_1}) \overset{?}{=} e(U_{i,\mathsf{ID}_i}, C_1), \quad i \in [\ell], \quad \text{and}$$

$$e(C_{i,j}, U_{1,\mathsf{ID}_1}) \overset{?}{=} e(T_{i,j}, C_1), \quad j \in [m], i \in \mathcal{J}_j$$

If all of the above are verified, then the ciphertext sanity check passes, otherwise it fails.

Trace This algorithm takes an identity ID, a well-formed decryption key d_{ID} (i.e., passing the key sanity check) and a decoder box D which is ϵ-useful (where ϵ is polynomially related to the security parameter). For convenience, we assume that the message space is sufficiently large so that the probability of guessing a randomly chosen message is negligible in the security parameter. We note that the algorithm can be easily extended to the general case (and additionally, it would require the input ϵ to be "non-trivial", i.e., noticeably higher than the probability with which a randomly chosen message can be guessed correctly). Our tracing algorithm will run in time polynomial in the number of repetitions m and $\frac{1}{\epsilon}$. For each $j \in [m]$ fix an $X_j \subset ([n] \setminus \mathcal{I}_j)$ of size $1 + k - \tau$. Note that if C is a ciphertext with $\mathcal{J}_j \supset X_j$ for any j, then this ciphertext cannot be decrypted by an honest user. We will use this fact to attempt to catch the PKG cheating because the PKG is oblivious to an honest user's key. The tracing algorithm will repeat the following experiment $\eta = (\frac{6m}{\epsilon})^2$ times:

1. Iterate $j^\star \in [m]$ and perform the following test:
   (a) Choose a random $\mathcal{J}_{j^\star} \supset X_{j^\star}$, and the remaining $\{\mathcal{J}_j\}_{j \neq j^\star}$ at random.
   (b) Encrypt a random message using $\{\mathcal{J}_j\}$ as the ciphertext dummy attributes.
   (c) Test if the box correctly decrypts the message. If it does, immediately implicate the PKG by returning PKG and stop, otherwise continue.

If at the end of the experiment the PKG has not been implicated, then the algorithm implicates the user by returning User and stops. In the next section, we show that the above simple tracing mechanism works except with negligible probability even though the ciphertexts on which we probe the box are coming from a *special distribution* (rather than simply being random ciphertexts for the given identity).

In Appendix B, we show how to modify the above scheme to achieve perfect completeness by running a "complementary scheme" in parallel.

4.3 A Modification For IND-ID-CPA Security

We describe a simple method to augment our construction to achieve IND-ID-CPA security. We can secret share our message M by choosing a random M_1 and setting $M_2 = M - M_1$. We then encrypt M_1 using our construction and encrypt M_2 using a IND-ID-CPA secure IBE scheme. Because both our construction and the Waters IBE scheme rely on the decisional BDH assumption, we may achieve IND-ID-CPA security in our scheme by including the public parameters of the Waters IBE scheme into our own and modifying the encryption and decryption schemes to secret share the message as just described.

4.4 Security Proofs

We now prove the security of the scheme described above. Recall that there is a global security parameter λ for which we implicitly refer to whenever we mention "negligible" or "polynomial". We begin by addressing the IND-ID-CPA security of the scheme.

Theorem 1 *The advantage of an adversary in the* IND-ID-CPA *game is negligible for the above A-IBE scheme under the decisional BDH assumption.*

The above theorem follows trivially from the IND-ID-CPA security of Waters construction [Wat05]. Given an adversary to break the IND-ID-CPA security of our construction, it is straightforward to construct an adversary to break the IND-ID-CPA security of Waters construction. We shall omit the details from this paper.

Similar to [Wat05], we remark that with small modifications, it is possible to achieve IND-ID-CCA security by using techniques of Canetti, Halevi and Katz [CHK04]. We can also use other methods [BK05, BMW05] to achieve greater efficiency. We can also run different IBE schemes such as the Gentry IBE scheme [Gen06] so long as we make the necessary additional security assumptions.

Theorem 2 *Assuming that the underlying k-out-of-n oblivious transfer protocol is secure as per the ideal/real world security definition* [1] *[Can00], the advantage of an adversary in the* DishonestPKG *game is negligible for the above scheme.*

Proof: Assume towards a contradiction that an adversary $\mathcal{A}_0$ has some non-negligible probability ε of success. We will eventually work to contradict a combinatorial lemma (Lemma 5). We begin by replacing the oblivious transfer protocol by an ideal OT functionality. By the security of the simulation of the oblivious transfer protocol, the adversary may lose at most a negligible advantage moving between the two worlds. This can be stated as the following lemma:

Lemma 1 (Composition theorem (Canetti [Can00])) *For every adversary $\mathcal{A}_0$ which succeeds with probability ε in the real world, there exists an $\mathcal{A}$ which succeeds in the ideal OT world which succeeds with probability δ where $|\varepsilon - \delta| < \nu_1$ and ν_1 is negligible.*

Let SUCC be the event that the adversary $\mathcal{A}$ succeeds in this game. Let $r_{\mathcal{A}}$ denote the randomness for this adversary, and r_C denote the randomness for the challenger. Recall that the only randomness used by the challenger is during the key generation protocol where it selects a set of dummy attributes. We henceforth identify r_C as also being a set of dummy attributes $\{\mathcal{I}_j\}$. Let $\mathcal{E}_1$ be the event that the execution of the adversary will not cause an abort in the key generation phase with probability at least $\delta/2$. That is to say, $\mathcal{E}_1$ holds for the set of $r_{\mathcal{A}}$ on which $Pr[\mathsf{Ch} \text{ finishes KeyGen}] \geq \delta/2$ where the probability is taken over the randomness of the challenger.

Lemma 2 *The probability that event $\mathcal{E}_1$ occurs is at least $\delta/2$.*

Proof: Observe that when $\mathcal{E}_1$ does not occur, $\mathcal{A}$ has at most a $\delta/2$ chance of success due to the fact that the challenger will abort in the key generation phase with at least a $1 - \delta/2$ probability. Thus we can lower bound the probability of $\mathcal{E}_1$ occurring by $\delta/2$ using Markov's inequality. ■

In other words, a $\delta/2$ fraction of all possible dummy attribute choices for ID will result in a well-formed decryption key. We now focus on the executions on which $\mathcal{E}_1$ occur. The expected success

[1] As discussed in Section 2, the existence of such k-out-of-n oblivious transfer is implied by the decisional BDH assumption.

probability of the adversary must still be at least δ because every execution where $\mathcal{E}_1$ does not occur will fail with probability at least $\delta/2$. Because the challenger selects dummy attributes uniformly at random in the key generation protocol, this implies at least a $\delta/2$ fraction of all dummy attribute sets will lead to the challenger receiving a well-formed decryption key. We shall argue that even after the decryption query phase, there are still too many possible choices of dummy attributes for the adversary's decoder box to succeed against a non-negligible fraction of them.

Let $\mathcal{E}_2$ be the event that the challenger did not abort in the key generation phase. Indeed, the success probability of the adversary can only increase if we condition on $\mathcal{E}_2$ occurring: anytime $\mathcal{E}_2$ does not occur, the adversary immediately loses. If the challenger did not abort in the key generation phase, then the final check (Step 9) in the key generation protocol guarantees that

$$Y = \prod_{i \in [\ell]} e(U_{i,\mathsf{ID}_i}, d_i) \prod_{j \in [m]} Y_j$$

where

$$Y_j = \prod_{i \in S} e(d_{i,j}, T_{i,j})^{\Delta_{i,S}(0)}.$$

As a reminder, the key sanity check guarantees that the $d_{i,j}$ implicitly define a *unique* degree $\tau - 1$ polynomial q_j for each j. If in the decryption query phase a ciphertext

$$C = (\{\mathcal{J}_j\}_{j \in [m]}, c, \{C_i\}, \{C_{i,j}\})$$

is asked to the challenger, the ciphertext sanity check in the decryption algorithm guarantees that there is some unique r such that $C_i = U^r_{i,\mathsf{ID}_i}$ and $C_{i,j} = T^r_{i,j}$. Regardless of which $d_{i,j}$ are used to decrypt (as noted above, they all define a unique polynomial), one can see by algebraic manipulation that the decryption will always return c/Y^r provided that $|\mathcal{I}_j \cap \mathcal{J}_j| \geq \tau$ for all $j \in [m]$. Note that for any fixed ciphertext, over a random choice of all the user's dummy attributes $\{\mathcal{I}_j\}$, there is only a negligible probability that the user cannot decrypt. This is inherent by the proper construction of k,n,τ and m. We will call this negligible quantity ν_2.

Let $\mathcal{E}_3$ be the event that all well-formed ciphertexts were properly decrypted (i.e. the challenger did not fail on any query to decrypt due to insufficient intersection of dummy attributes). We now analyze the probability of this event occurring and how it affects the view of the adversary. We shall argue that the probability that $\mathcal{E}_3$ does not occur given $\mathcal{E}_1 \wedge \mathcal{E}_2$ is negligible. We define this quantity to be ν_3.

We stratify $\mathcal{E}_3$ as the conjunction of the events "Ch did not fail on query i". For some random tape r_C of the challenger, let $\{\mathcal{I}_j\}$ be the dummy attributes defined by it, and let $\{\mathcal{J}^i_j\}$ be the dummy attributes in the ith ciphertext query, we define GOOD_i to be the event that $|\mathcal{I}_j \cap \mathcal{J}^i_j| \geq \tau$ for all $j \in [m]$. Define $\mathcal{F}_i = \mathsf{GOOD}_1 \wedge \ldots \wedge \mathsf{GOOD}_i$. First, we prove a lemma about the view of the adversary.

Lemma 3 *Fix a random tape $r_{\mathcal{A}}$ of the adversary such that $\mathcal{E}_1$ occurs. Let r_C, r'_C be any two arbitrary elements from the set $\{r_C : \mathcal{E}_2 \wedge \mathcal{F}_{i^*-1}$ holds$\}$. Before query i^* is made, the view of the adversary in the execution where Ch uses r_C as its random tape is* identical *to the view in the execution where Ch uses r'_C as its random tape.*

Proof: After the key generation phase, the adversary learns only whether or not the challenger aborted. Up to this point, because we are in the ideal OT world, this is the only information the adversary learns. Event $\mathcal{E}_2$ occurring means the challenger did not abort and received a well-formed key. On every query i before the i^*th query, if the ciphertext is malformed, the challenger will reject regardless of its own dummy attributes, and if it is well-formed, then GOOD_i guarantees that there are sufficiently many attributes in the intersection between the challenger's dummy attributes and the ones in the ciphertext, and so the challenger will reply with c/Y^r. $\blacksquare$

We now prove the statement that for any fixed random tape $r_{\mathcal{A}}$ of the adversary such that $\mathcal{E}_1$ occurs, $Pr[\neg\mathcal{E}_3|\mathcal{E}_2] \leq \frac{2q\nu_2}{\delta/2}$ where the probability is taken over the random tapes of the challenger.

Lemma 4 *Fix a random tape $r_{\mathcal{A}}$ of the adversary such that $\mathcal{E}_1$ occurs, then $Pr_{r_C}[\neg\mathcal{E}_3|\mathcal{E}_2] \leq \frac{2q\nu_2}{\delta/2}$. We define the negligible quantity on the right hand side to be ν_3.*

Proof: Let p_2 be the probability that event $\mathcal{E}_2$ occurs. Because we fixed an execution where $\mathcal{E}_1$ occurs, we have that $p_2 \geq \delta/2$. We shall prove inductively that $Pr[\mathcal{E}_2 \wedge \mathcal{F}_i] \geq p_2 - i\nu_2$.

By Lemma 3, before the first ciphertext query, the adversary has no information about r_C other than $\mathcal{E}_2$ occurred. Hence, the first ciphertext is independent of any r_C for which $\mathcal{E}_2$ holds. Recall that for any ciphertext, ν_2 is the negligible fraction of user keys that cannot decrypt it. The probability that a uniformly selected r_C conditioned on $\mathcal{E}_2$ will fail on the first ciphertext query is $Pr[\neg\mathsf{GOOD}_1|\mathcal{E}_2] \leq \frac{\nu_2}{p_2}$. Thus $Pr[\mathsf{GOOD}_1|\mathcal{E}_2] \geq 1 - \frac{\nu_2}{p_2}$ and so there is at least a $(1-\frac{\nu_2}{p_2}) \cdot p_2 = p_2 - \nu_2$ fraction of the random tapes remaining which satisfy $\mathcal{E}_2 \wedge \mathsf{GOOD}_1$.

On the ith query, the queried ciphertext once again cannot be decrypted by a ν_2 fraction of all possible r_C's. In the worst case, this fraction is disjoint from the ones excised by the first $i-1$ queries. By Lemma 3, the adversary has no information about r_C other than $\mathcal{E}_2 \wedge \mathcal{F}_{i-1}$ occurred. The ith query is independent of any r_C for which $\mathcal{E}_2 \wedge \mathcal{F}_{i-1}$ holds. By induction, this accounts for at least a $p_2 - (i-1)\nu_2$ fraction of all possible r_C's. The probability that a uniformly selected r_C conditioned on $\mathcal{E}_2 \wedge \mathcal{F}_{i-1}$ will fail to decrypt the ith ciphertext query is $Pr[\neg\mathsf{GOOD}_i|\mathcal{E}_2 \wedge \mathcal{F}_{i-1}] \leq \frac{\nu_2}{p_2-(i-1)\nu_2}$. Consequently, we calculate that $Pr[\mathcal{E}_2 \wedge \mathcal{F}_i]$ is at least $p_2 - i\nu_2$.

Eventually after q queries, we have $Pr[\mathcal{E}_2 \wedge \mathcal{E}_3] = Pr[\mathcal{E}_2 \wedge \mathcal{F}_q]$ is at least $p_2 - q\nu_2$. So $Pr[\mathcal{E}_3|\mathcal{E}_2] \geq 1 - \frac{q\nu_2}{p_2} \geq 1 - \frac{2q\nu_2}{\delta}$ from which the lemma follows. $\blacksquare$

Finally, the adversary must output a decoder box D. We show that *any* decoder box can implicate the user in only a negligible fraction of dummy attribute sets. We call this negligible quantity ν_4. Our main lemma is as follows:

Lemma 5 *Let $\epsilon = \frac{1}{poly(\lambda)}$ and D be an ϵ-useful decoder box. If $\{\mathcal{I}_j\}_{j\in[m]}$ is a dummy attribute set for the user, we consider the following experiment:*

- *Select a dummy attribute set $\{\mathcal{J}_j\}_{j\in[m]}$ at random such that $|\mathcal{I}_j \cap \mathcal{J}_j| < \tau$ for some j.*
- *Select a random message M and encrypt M using $\{\mathcal{J}_j\}$ as the dummy attributes.*
- *The decoder box outputs some $M' = \mathsf{D}(C)$.*

Define the event DBox to hold when $M' = M$. The lemma states that for all but a negligible fraction, ν_4, of choices for $\{\mathcal{I}_j\}_{j\in[m]}$ we have

$$\Pr[\mathsf{DBox}] > \frac{\epsilon}{24m}$$

In particular, the tracing algorithm will implicate the PKG for all but a negligible fraction of choices of dummy attributes.

Assuming the lemma above, we continue our proof by contradiction. By Lemma 3, the view of the adversary after events $\mathcal{E}_2 \wedge \mathcal{E}_3$ will be identical for all the remaining $(\delta/2) - \nu_3$ fraction of r_C's. Thus, the adversary creates this box independent of r_C other than the fact that $\mathcal{E}_2 \wedge \mathcal{E}_3$ hold. Because any of these dummy attribute sets remain equally likely, the probability that D will succeed is at most $\frac{\nu_4}{(\delta/2)-\nu_3}$. We summarize this contradiction in the following equations:

$$\begin{aligned}
\delta &= Pr[\mathsf{SUCC}] \\
&\leq Pr[\mathsf{SUCC}|\mathcal{E}_1 \wedge \mathcal{E}_2] \\
&= Pr[\mathsf{SUCC}|\mathcal{E}_1 \wedge \mathcal{E}_2 \wedge \mathcal{E}_3]Pr[\mathcal{E}_3|\mathcal{E}_1 \wedge \mathcal{E}_2] \\
&\quad + Pr[\mathsf{SUCC}|\mathcal{E}_1 \wedge \mathcal{E}_2 \wedge \neg\mathcal{E}_3]Pr[\neg\mathcal{E}_3|\mathcal{E}_1 \wedge \mathcal{E}_2] \\
&\leq \frac{\nu_4}{(\delta/2) - \nu_3} \cdot 1 \\
&\quad + 1 \cdot \nu_3 \\
&= negl.
\end{aligned}$$

We now prove the main lemma:

Proof: For the purposes of this proof, we fix an identity ID and ignore all portions of the decryption key except for the dummy attributes contained in it: $\{\mathcal{I}_j\}_{j\in[m]}$. Similarly, we ignore all portions of the ciphertext except which dummy attributes are contained in it: $\{\mathcal{J}_j\}_{j\in[m]}$. We will refer to $\mathcal{I}_j$ (resp. $\mathcal{J}_j$) as the jth component or index of the dummy attributes in a user key (resp. ciphertext). For notational purposes, we will write for the user (resp. ciphertext) $U = (\mathcal{I}_1, \ldots, \mathcal{I}_m)$ (resp. $Z = (\mathcal{J}_1, \ldots, \mathcal{J}_m)$). We can imagine both U and Z as being subsets of the same universe $\mathcal{K}$ which contains all m-tuples of k-sized sets. We simply refer to these as the "user set" and the "ciphertext set".

Recall in the A-IBE scheme, that each $\mathcal{I}_j$ and each $\mathcal{J}_j$ is a subset of $[n]$ containing k elements. A user can decrypt if and only if each component in the intersection between the user set and the ciphertext set is at least some threshold (τ). Fix a decryption box D which we fix to be ϵ-useful. For each ciphertext set Z we can have some probability p_Z the that the box will decrypt on it (note that since we ignore the message, this is taken over the randomness used in the encryption except for the selection of the ciphertext set).

Consider how one randomly samples ciphertexts which cannot be decrypted by the user. We may think of choosing ciphertext set that intersects (with the user set) on the jth component U_j by less than τ attributes by first choosing a set of $\beta = 1 + k - \tau$ attributes disjoint from $\mathcal{I}_j$, then selecting the remaining $k - \beta$ attributes at random. Because these β-sized set of attributes will be important for us, it is useful to think of any arbitrary β-sized subset as an atomic object, which we will call a *bundle*. To clarify the description, every set of β attributes on the jth repetition is a *j-bundle*. Let B_j be the set of all j-bundles. In sampling random ciphertexts which cannot be decrypted by the user, the idea is to select a bundle which avoid a user set, then select a ciphertext set which contains that bundle. For each bundle $b \in B_j$, we can associate to it a set of ciphertexts whose attribute set contains it: $\mathcal{K}_b^j := \{Z = (\mathcal{J}_1, \ldots, \mathcal{J}_m) | b \subseteq \mathcal{J}_j\}$. Conversely, for each ciphertext $Z = (\mathcal{J}_1, \ldots, \mathcal{J}_m)$ we can associate to it a set of j-bundles which it contains: $\mathcal{B}_Z^j := \{b \in B_j | b \subseteq \mathcal{J}_j\}$. We may similarly define sets for users: $\mathcal{V}_b^j := \{U = (\mathcal{I}_1, \ldots, \mathcal{I}_m) | b \cap \mathcal{I}_j = \emptyset\}$ and $\mathcal{A}_U^j := \{b \in B_j | b \cap \mathcal{I}_j = \emptyset\}$ (users that avoid a bundle, and bundles that avoid a user, respectively).

We first make the observation that by symmetry, the size of the sets $\mathcal{K}_b^j$ (as well as the sets $\mathcal{B}_Z^j, \mathcal{V}_b^j, \mathcal{A}_U^j$) are independent of b, j, Z, and U. Thus, we may speak of the value $|\mathcal{K}_b^j|$ even outside the scope of a well-defined b or j. We define the set of all bundles to be $B := \bigcup_{j\in[m]} B_j$ and we can similarly define the set of all bundles contained in (resp. avoided by) a ciphertext (resp. a user) as $\mathcal{B}_Z := \bigcup_{j\in[m]} B^j_Z$ (resp. $\mathcal{A}_U = \bigcup_{j\in[m]} A^j_U$). By symmetry, selecting a ciphertext set $Z \in \mathcal{K}$ uniformly at random then selecting a bundle it contains $b \in \mathcal{B}_Z$ uniformly at random generates the same distribution on pairs (Z, b) as selecting a bundle $b \in \mathcal{B}$ at random (say it is a j-bundle) followed by selecting a ciphertext set containing it $Z \in \mathcal{K}^j_b$ uniformly at random. This can be combinatorially written as $|\mathcal{K}| \cdot |\mathcal{B}_Z| = |\mathcal{B}| \cdot |\mathcal{K}^j_b|$.

Definition 2 *Let $b \in B_j$ be a (j-)bundle. Define $p^j_b := \frac{\sum_{Z\in\mathcal{K}^j_b} p_Z}{|\mathcal{K}^j_b|}$ which is the average probability that a randomly selected ciphertext set containing this bundle is decrypted by D. A bundle $b \in B_j$ is said to be* heavy on j *if $p^j_b > \frac{\epsilon}{6m}$. We define $\mathcal{L}_j$ to be the set of bundles which are light (not heavy) on j, and $\mathcal{L} = \bigcup_{j\in[m]} \mathcal{L}_j$.*

We first show a combinatorial lemma.

Lemma 6 (Marking Lemma) *Consider an arbitrary marking on bundles where over $\frac{1}{2}$ the bundles in B_j are marked for each $j = 1, \ldots, m/2$. Then with probability at least $1 - \left(\frac{3}{4}\right)^{m/2}$ a randomly sampled ciphertext set Z will have the property that for some j, a $\frac{1}{3}$ fraction of $\mathcal{B}^j_Z$ will be marked.*

Proof: Consider any fixed j between 1 and $m/2$. For a ciphertext set Z we say Z is $MARK_j$ if at least a $\frac{1}{3}$ fraction of $\mathcal{B}^j_Z$ is marked. Let p be the probability over a randomly chosen Z that Z is $MARK_j$. Then we have

$$\begin{aligned}\frac{1}{2} &\le Pr_{Z,b\leftarrow\mathcal{B}^j_Z}[b \text{ is marked}]\\ &\le Pr[b \text{ is marked}|Z \text{ is } MARK_j] \cdot Pr[Z \text{ is } MARK_j]\\ &+ Pr[b \text{ is marked}|Z \text{ is not } MARK_j] \cdot Pr[Z \text{ is not } MARK_j]\\ &\le 1 \cdot p + \frac{1}{3} \cdot (1-p)\end{aligned}$$

Solving for p we get $p \ge \frac{1}{4}$. Since j was arbitrary, the probability that no j from 1 to $m/2$ have $MARK_j$ is $\left(\frac{3}{4}\right)^{m/2}$ which is negligible in n as long as m is super-logarithmic in n. ■

Claim 1 *Let D be an ϵ-useful decoder box. Either (1) on over half the repetitions, more than half the bundles are heavy (on those components) or (2) on at least half the repetitions, at least half the bundles are light. We claim that Case (1) implies Lemma 5, our main lemma. We further claim Case (2) will contradict the usefulness of D.*

Proof:
Case (1): WLOG we may assume the first $m/2$ repetitions have more than half the bundles heavy. Select a user set U uniformly at random. The Trace algorithm behaves by first fixing a random j-bundle from each $j \in [m]$ which avoids the user key on that component. Note that because the user was selected at random, each of these fixed bundles has probability $\frac{1}{2}$ of being heavy. Thus if m is super-logarithmic, then there is an all but negligible probability that at least one bundle is heavy. Then it will repeatedly iterate through every component sampling a ciphertext set at random which contains the previously fixed bundle in that component. Thus, because there is at least one bundle that is heavy, by randomly sampling $(1/\frac{\epsilon}{6m})^2 = \eta$ ciphertext sets that contain this bundle, a decryption will occur with high probability and thus we will implicate the PKG. Furthermore, if one considers sampling a random ciphertext set that U cannot decrypt by:

1. Selecting a component $j \in [m]$
2. Selecting a bundle $b \in \mathcal{A}_U^j$ in avoiding the user on that component uniformly at random
3. Selecting a ciphertext set $Z \in \mathcal{K}_b^j$ that contains that bundle uniformly at random

then we see that there is a $\frac{1}{2}$ probability of selecting a component with over half the bundles heavy in the first step, a $\frac{1}{2}$ probability that the bundle selected in the second step is heavy, and finally by the definition of heavy, the ciphertext selected in the third step will be decrypted by D with at least a $\frac{\epsilon}{6m}$ probability. Combined, this gives a probability of $\epsilon/24m$ as claimed in Lemma 5.

Case (2): WLOG we may assume the first $m/2$ repetitions have more than half the bundles light. By the marking lemma, if we mark the light bundles, we know that with all but a negligible probability, a randomly sampled ciphertext will have at least $\frac{1}{3}$ fraction marked (i.e. light) bundles on some component. We say Z has the property $LIGHT_j$ if on component j there are at least $\frac{1}{3}$ fraction of light bundles. We condition only on the ciphertext sets which has $LIGHT_j$ for some j, as only negligibly many do not have this property. Consider the space of all pairs (Z, b) where Z is a ciphertext set which contains a light j-bundle b (for some j). We will define two probability distributions $\mathcal{D}_1$ and $\mathcal{D}_2$ on this space. The first distribution is:

1. Select a random Z.
2. For every component of Z, place all light bundles it contains into a set S_Z. In other words, set $S_Z = \bigcup_{j\in[m]}(\mathcal{B}_Z^j \cap \mathcal{L}_j)$. Select a random bundle $b \in S_Z$.

The second distribution is:

1. Select a random light bundle $b \in \mathcal{L}$.
2. Select a random $Z \in K_b^j$.

Observe that if we fix some (Z', b'), the probability that $\mathcal{D}_1$ selects it is $\frac{1}{|\mathcal{K}|} \cdot \frac{1}{|S_Z|}$. As mentioned above, at least a third of all the bundles it contains (on some component) are light, and because each component contains the same number of bundles, at least $\frac{1}{3m}$ *total* bundles it contains will be light. Thus, the probability that $\mathcal{D}_1$ selects (Z', b') is somewhere between $\frac{1}{|\mathcal{K}|} \cdot \frac{3m}{|\mathcal{B}_Z|}$ and $\frac{1}{|\mathcal{K}|} \cdot \frac{1}{|\mathcal{B}_Z|}$.

For the distribution $\mathcal{D}_2$, it will select (Z', b') with probability $\frac{1}{|\mathcal{L}|} \cdot \frac{1}{|\mathcal{K}_b^j|}$. By assumption, there are at least half the bundles light on the first half of the repetitions, so overall, the light bundles make up over a quarter fraction of all bundles. Thus this probability is between $\frac{4}{|\mathcal{B}|} \cdot \frac{1}{|\mathcal{K}_b^j|}$ and $\frac{1}{|\mathcal{B}|} \cdot \frac{1}{|\mathcal{K}_b^j|}$.

By the earlier observation that $|\mathcal{K}| \cdot |\mathcal{B}_Z| = |\mathcal{B}| \cdot |\mathcal{K}_b^j|$, the probability that a ciphertext set is selected in $\mathcal{D}_1$ is at most $3m$ times as likely as that ciphertext set is selected in $\mathcal{D}_2$. However, the probability that D decrypts a ciphertext containing a ciphertext set sampled from the first distribution is ϵ, while by definition of "light", the probability that D decrypts a ciphertext containing a ciphertext set sampled from the second distribution is at most $\frac{\epsilon}{6m}$. Then we have

$$\begin{aligned}
\epsilon &= \sum_{(Z,b)} (\text{(Z,b) is sampled by } \mathcal{D}_1) \cdot p_Z \\
&\leq \sum_{(Z,b)} 3m \cdot (\text{(Z,b) is sampled by } \mathcal{D}_2) \cdot p_Z \\
&\leq 3m \cdot \sum_{(Z,b)} (\text{(Z,b) is sampled by } \mathcal{D}_2) \cdot p_Z \leq 3m \cdot \frac{\epsilon}{6m}
\end{aligned}$$

which leads to a contradiction. ■

Theorem 3 *The advantage of an adversary in the Selective-ID* $\mathsf{DishonestUser}$ *game is negligible for the above A-IBE scheme under the decisional BDH assumption.*

There are similarities between the Selective-ID $\mathsf{DishonestUser}$ game and the selective-set $\mathsf{IND\text{-}ID\text{-}CCA}$ game of Goyal et. al. [GPSW06]. With some critical modifications, one may adapt the proof of security in Goyal et. al. [GPSW06] to show a direct reduction of the Selective-ID $\mathsf{DishonestUser}$ game to the decisional BDH assumption. Instead, we pinpoint these critical modifications by giving a reduction from the Selective-ID $\mathsf{DishonestUser}$ game to the selective-set $\mathsf{IND\text{-}ID\text{-}CCA}$ game of Goyal et. al. [GPSW06].

Proof Sketch: Assume towards a contradiction that there is an adversary $\mathcal{A}_0$ which wins the $\mathsf{DishonestUser}$ game with advantage ε. As in Theorem 2, we argue that by the composition theorem of Canetti [Can00], there exists an adversary $\mathcal{A}$ which has advantage Adv in the OT-Hybrid model where the oblivious transfer in the key generation is replaced by an ideal functionality. This new advantage Adv only differs from ε by a negligible quantity. We use $\mathcal{A}$ to play against a selective-set ABE challenger $\mathcal{B}$. Our construction was based off of the Goyal et. al. [GPSW06] scheme so there is a one-to-one correspondence between the parameters in that scheme and the parameters in our scheme. Thus it makes sense when we speak of directly passing the parameters from $\mathcal{B}$ to $\mathcal{A}$.

In detail, we consider the universe of attributes to be of size $2\ell + mn$. The 2ℓ attributes $(A_{1,0}, A_{1,1}, \ldots, A_{\ell,0}, A_{\ell,1})$ will be for the identity and the remaining attributes will be for the dummy attributes. In our scheme, a user will have ℓ attributes corresponding to his identity (i.e. he will have A_{i,ID_i} if the ith bit of his identity is ID_i) and k dummy attributes for each of the m repetitions. This choice of attributes naturally defines the associated policy (in the sense of Goyal et. al. [GPSW06] that a ciphertext can be decrypted only if the identity attributes match and there is at least a τ number of attributes matching in each of the m repetitions. We will use this natural correspondence in the key generation query phase of the $\mathsf{DishonestUser}$ game.

We now give a reduction from the $\mathsf{DishonestUser}$ game to the selective-set $\mathsf{IND\text{-}ID\text{-}CCA}$ game of Goyal et. al. [GPSW06].

Select ID: The adversary $\mathcal{A}$ selects an ID^* as the challenge. We select the set of attributes corresponding to ID^* (namely, $\{A_{i,\mathsf{ID}_i^*}\}_{i=1}^{\ell}$) and a random set of dummy attributes $\{\mathcal{J}_j^*\}_{j\in[m]}$ and send the union as the selected set to $\mathcal{B}$. These will be the attributes used in the challenge ciphertexts in the selective-set game we are playing with $\mathcal{B}$.

Setup: Then $\mathcal{B}$ sends us public parameters, which we pass on to $\mathcal{A}$.

Key Generation Queries: Because we are now in the simulation-based model of OT, we know the private inputs in the key generation protocol and so we can learn the dummy attribute set. If $\mathcal{A}$ queries for a key on $\mathsf{ID} \neq \mathsf{ID}^*$ then simply pass the corresponding user policy as a key query to $\mathcal{B}$ which returns a well-formed key which we pass back to $\mathcal{A}$.

On the other hand if $\mathsf{ID} = \mathsf{ID}^*$, since we know the private inputs, we may select permutations $\pi_1 \ldots, \pi_m$ (as per Step 6 in the key generation protocol) in a way such that the key received by $\mathcal{A}$ will not be able to decrypt a ciphertext containing our previously selected attributes. We then query $\mathcal{B}$ for this key and pass it back to $\mathcal{A}$. Note that we must argue that this deviation from the protocol does not affect $\mathcal{A}$'s view. But indeed, this is the case because of symmetry: selecting a set of dummy attributes for a ciphertext uniformly at random then selecting a user's dummy attribute set that cannot decrypt this ciphertext uniformly at random induces the uniform distribution on the user's dummy attribute set.

Create Decoder Box: $\mathcal{A}$ now must output a decryption key d_{ID^*} and a decoder box D. If $\mathcal{A}$ wins the $\mathsf{DishonestUser}$ game then decoder box will implicate the PKG which can only occur when there is a non-negligible probability that D decrypts a random ciphertext that cannot be decrypted by d_{ID^*}. We randomly select two messages M_0, M_1 and send them to $\mathcal{B}$ which then sends us a challenge ciphertext C under the previously selected set. If d_{ID^*} can decrypt this message then we immediately do so and send the correct guess to $\mathcal{B}$. On the other hand, if d_{ID^*} cannot decrypt then C can be viewed as a random ciphertext that ID^* cannot decrypt, and therefore whenever $\mathcal{A}$ wins the DishonestUser game, D must have a non-negligible advantage in decrypting the ciphertext.

Thus, we have a non-negligible advantage in the selective-set game against $\mathcal{B}$. This contradicts the security of the ABE scheme under the decisional BDH assumption. ■

5 Conclusion and Open Problems

In this paper, we proposed a model of a secure accountable authority identity based encryption scheme which handles black-box decoders. This model is a critical improvement over the original Goyal [Goy07] model. We gave a construction of an A-IBE scheme in this enhanced model under the decisional BDH assumption where the security was respect to the IND-ID-CPA, DishonestPKG, and Selective-ID DishonestUser games. It may be worth noting that the construction can be viewed as "attachable" to any IBE scheme by secret sharing the message, so we may achieve better security or a more efficient underlying scheme as we choose.

There are several interesting open problems to be explored. We prove our construction to be secure in the Selective-ID DishonestUser game. This is seemingly due to the underlying connection to the Goyal et. al. [GPSW06] scheme which is only provably select-set secure. Even if there is some inherent difficulty in proving the full security of attribute-based encryption schemes such as Sahai-Waters [SW05] or Goyal et. al. [GPSW06], there may be other tricks that can be done for our construction.

Important questions arise when dealing with the users' decryption keys. The security in both Goyal [Goy07] and our construction only hold when a one decryption key is generated per user (with an explicit break if more than one is made available). This means that if the user loses his key, the user needs to get a new identity ID' to request a new key. Can we make a A-IBE scheme that allows a single ID to generate polynomially many keys?

Our tracing algorithm takes as input a user's decryption key. If a user lost the key or is deliberately uncooperative in court, then we cannot implicate the PKG or the user. One interesting open problem is to consider the possibility of tracing a box using only a public tracing key, or with the assistance of a tracing authority. What would be the proper additional modifications to the model of accountable authority IBE to account for this?

Finally, we mention the issue of efficiency in our scheme. We view this in terms of the cost of turning an IBE scheme into an A-IBE scheme by secret sharing the message with our construction. Each ciphertext and decryption key will now have an additional $\ell + mk$ group elements and an additional mk elements to represent the attributes. In our construction, there was a single global parameter λ which governed these parameters (of accountability) as well as the security of the scheme. One can imagine having a second parameter γ which will determine the *accountability* rather than the security of the scheme which will allow us to adjust the level of accountability in the scheme. The creation of an A-IBE scheme with only a logarithmic or constant sized decryption key and ciphertext remains as a broad open question.

References

[AHL$^+$08] Man Ho Au, Qiong Huang, Joseph K. Liu, Willy Susilo, Duncan S. Wong, and Guomin Yang. Traceable and retrievable identity-based encryption. In *Applied Cryptography* *and Network Security*, volume 5037 of *Lecture Notes in Computer Science*, pages 94–110. Springer Berlin / Heidelberg, 2008.

[ARP03] Sattam S. Al-Riyami and Kenneth G. Paterson. Certificateless public key cryptography. In Chi-Sung Laih, editor, *ASIACRYPT*, volume 2894 of *Lecture Notes in Computer Science*, pages 452–473. Springer, 2003.

[BB04a] Dan Boneh. and Xavier Boyen. Efficient Selective-ID Secure Identity Based Encryption Without Random Oracles. In *Advances in Cryptology – Eurocrypt*, volume 3027 of *LNCS*, pages 223–238. Springer, 2004.

[BB04b] Dan Boneh and Xavier Boyen. Secure identity based encryption without random oracles. In Matthew K. Franklin, editor, *CRYPTO*, volume 3152 of *Lecture Notes in Computer Science*, pages 443–459. Springer, 2004.

[BBG05] Dan Boneh, Xavier Boyen, and Eu-Jin Goh. Hierarchical identity based encryption with constant size ciphertext. In Cramer [Cra05], pages 440–456.

[BF01] D. Boneh and M. Franklin. Identity Based Encryption from the Weil Pairing. In *Advances in Cryptology – CRYPTO*, volume 2139 of *LNCS*, pages 213–229. Springer, 2001.

[BK05] Dan Boneh and Jonathan Katz. Improved efficiency for cca-secure cryptosystems built using identity-based encryption. In *CT-RSA*, pages 87–103, 2005.

[BMW05] Xavier Boyen, Qixiang Mei, and Brent Waters. Direct chosen ciphertext security from identity-based techniques. In *ACM Conference on Computer and Communications Security*, pages 320–329, 2005.

[Can00] Ran Canetti. Security and composition of multiparty cryptographic protocols. *Journal of Cryptology*, 13(1):143–202, 2000.

[CHK03] R. Canetti, S. Halevi, and J. Katz. A Forward-Secure Public-Key Encryption Scheme. In *Advances in Cryptology – Eurocrypt*, volume 2656 of *LNCS*. Springer, 2003.

[CHK04] R. Canetti, S. Halevi, and J. Katz. Chosen Ciphertext Security from Identity Based Encryption. In *Advances in Cryptology – Eurocrypt*, volume 3027 of *LNCS*, pages 207–222. Springer, 2004.

[CNS07] Jan Camenisch, Gregory Neven, and Abhi Shelat. Simulatable adaptive oblivious transfer. In Moni Naor, editor, *EUROCRYPT*, volume 4515 of *Lecture Notes in Computer Science*, pages 573–590. Springer, 2007.

[Cra05] Ronald Cramer, editor. *Advances in Cryptology - EUROCRYPT 2005, 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Aarhus, Denmark, May 22-26, 2005, Proceedings*, volume 3494 of *Lecture Notes in Computer Science*. Springer, 2005.

[EGL85] Shimon Even, Oded Goldreich, and Abraham Lempel. A randomized protocol for signing contracts. *Commun. ACM*, 28(6):637–647, 1985.

[Gen03] Craig Gentry. Certificate-based encryption and the certificate revocation problem. In Eli Biham, editor, *EUROCRYPT*, volume 2656 of *Lecture Notes in Computer Science*, pages 272–293. Springer, 2003.

[Gen06] Craig Gentry. Practical identity-based encryption without random oracles. In Serge Vaudenay, editor, *EUROCRYPT*, volume 4004 of *Lecture Notes in Computer Science*, pages 445–464. Springer, 2006.

[GH07] Matthew Green and Susan Hohenberger. Blind identity-based encryption and simulatable oblivious transfer. Cryptology ePrint Archive, 2007. http://eprint.iacr.org/2007/235.

[Goy07] Vipul Goyal. Reducing Trust in the PKG in Identity Based Cryptosystems. In *Advances in Cryptology - CRYPTO 2007*, volume 4622 of *LNCS*, pages 430–447. Springer, 2007.

[GPSW06] Vipul Goyal, Omkant Pandey, Amit Sahai, and Brent Waters. Attribute-based encryption for fine-grained access control of encrypted data. In Ari Juels, Rebecca N. Wright, and Sabrina De Capitani di Vimercati, editors, *ACM Conference on Computer and Communications Security*, pages 89–98. ACM, 2006.

[LBD$^+$04] Byoungcheon Lee, Colin Boyd, Ed Dawson, Kwangjo Kim, Jeongmo Yang, and Seungjae Yoo. Secure key issuing in id-based cryptography. In James M. Hogan, Paul Montague, Martin K. Purvis, and Chris Steketee, editors, *ACSW Frontiers*, volume 32 of *CRPIT*, pages 69–74. Australian Computer Society, 2004.

[Lin08] A. Y. Lindell. Efficient Fully-Simulatable Oblivious Transfer. In *CR-RSA 2007*, LNCS. Springer, 2008.

[Sha84] A. Shamir. Identity Based Cryptosystems and Signature Schemes. In *Advances in Cryptology – CRYPTO*, volume 196 of *LNCS*, pages 37–53. Springer, 1984.

[SW05] A. Sahai and B. Waters. Fuzzy Identity Based Encryption. In *Advances in Cryptology – Eurocrypt*, volume 3494 of *LNCS*, pages 457–473. Springer, 2005.

[Wat05] Brent Waters. Efficient identity-based encryption without random oracles. In Cramer [Cra05], pages 114–127.

A An Instantiation of the Parameters

We give an explicit example of how to choose appropriate key sizes and threshold sizes for decryption. For simplicity, we will only focus on one component, i.e. we set $j = 1$ when looking at $\mathcal{I}_j$ in the decryption key and $\mathcal{J}_j$ in the ciphertext. Each of these sets is of size k which we choose to be a fixed constant fraction of n. For example, we may choose $k = \frac{3}{5} \cdot n$. From this, we can determine the expected number of dummy attributes in their intersection: $\frac{3}{5} \cdot \frac{3}{5} \cdot n$. By Chernoff bounds, it can be seen that the probability that this intersection falls below a constant fraction of $\frac{9}{25} \cdot n$ will be negligible in n. Thus if we set our threshold to be $\frac{7}{25} \cdot n$ then a random ciphertext can be decrypted by the user except with negligible probability.

B Perfect Completeness

To achieve perfect completeness, we make use of second "helper" ciphertext. Observe that if the intersection between the dummy attributes in a user key (which has k out of n elements) and those in a ciphertext (also k out of n) is less than the threshold τ, then attributes in the user key must intersect those in the complement of the ciphertext by at least $k - \tau + 1$. This can be thought of as "joining together" two A-IBE schemes in a way so that the ciphertexts can either be decrypted only in one scheme or the other. We formalize this as follows:

As before, $\mathbb{G}_1$ is a bilinear group of prime order p, and let g and g' be generators of $\mathbb{G}_1$. We use the same notation as in the previous section, noting that k (out of a total of n) is the number of dummy attributes in each of the m repetitions and that identities are of length ℓ. The example in the appendix shows how these may be chosen.

Setup For each $i \in [\ell]$, choose four numbers $u_{i,0}, u_{i,1}$ and $u'_{i,0}, u'_{i,1}$ uniformly at random from $\mathbb{Z}_p$ such that all 2ℓ numbers are different. In addition, for each $i \in [n]$ and $j \in [m]$ choose $t_{i,j}, t'_{i,j}$ uniformly at random from $\mathbb{Z}_p$. Also choose a numbers $y_0, \dots, y_m$ and $y'_0, \dots, y'_m$ uniformly at random in $\mathbb{Z}_p$.

The published public parameters are:

$$\mathrm{PK} = \Big[\{(U_{i,j} = g^{u_{i,j}}) : i \in [\ell], j \in \{0,1\}\}, \{(T_{i,j} = g^{t_{i,j}}) : i \in [n], j \in [m]\}, \{Y_i = e(g,g)^y\}_{i=0}^m, g\Big]$$

$$\mathrm{PK}' = \Big[\{(U'_{i,j} = g'^{u'_{i,j}}) : i \in [\ell], j \in \{0,1\}\}, \{(T'_{i,j} = g'^{t'_{i,j}}) : i \in [n], j \in [m]\}, \{Y'_i = e(g',g')^y\}_{i=0}^m, g'\Big]$$

The master key is:

$$\mathrm{MK} = \Big[\{u_{i,j}, u'_{i,j} : i \in [\ell], j \in \{0,1\}\}, \{t_{i,j}, t'_{i,j} : i \in [n], j \in [m]\}, \{y_i, y'_i\}_{i=0}^m\Big]$$

Key Generation Protocol

1. U aborts if the published values in the public key are not all different.

2. PKG generates ℓ random numbers $r_1, \dots, r_\ell$ from $\mathbb{Z}_p$ such that $r_1 + \cdots + r_\ell = y_0$. Similarly, choose r'_i so that $r'_1 + \cdots + r'_\ell = y'_0$.

3. PKG generates m random polynomials $q_1, \dots, q_m$ of degree $\tau - 1$ with $q_j(0) = y_j$. It also generates m random polynomials $q'_1, \dots, q'_m$ of degree $k - \tau$ with $q'_j(0) = y'_j$

4. PKG computes the key components $d_i = g^{r_i/u_{i,\mathsf{ID}_i}}$ for all $i \in [\ell]$ and sends them to U. It also computes key components $d_{i,j} = g^{q_j(i)/t_{i,j}}$ for all $i \in [n], j \in [m]$ and stores them. Similarly, it computes d'_i and $d'_{i,j}$.

5. PKG chooses a random permutations $\pi_1, \dots, \pi_m \in S_n$.

6. PKG and U then engage in m executions of a k-out-of-n oblivious transfer protocol where PKG acts as the sender and U acts as the receiver. In the jth execution, the private input of PKG is the key components $\{d_{\pi_j(i),j}, d'_{\pi_j(i),j}\}_{i=1}^n$ and the private input of U is a set $\mathcal{I}_j$ of k randomly selected dummy attributes. The private output of U is the key component $\{d_{\pi_j(i),j}\}_{i \in \mathcal{I}_j}$ and the complementary $\{d'_{\pi_j(i),j}\}_{i \notin \mathcal{I}_j}$.

7. PKG sends U the permutation list π. U checks if he got the right key components as per π (and aborts if the check fails).

8. U sets $d = (\{d_i\}_{i \in [\ell]}, \{(\mathcal{I}_j, \{d_{i,j}\}_{i \in \mathcal{I}_j})\}_{j \in [m]})$ and d' similarly. We perform a sanity check on the decryption key as in the original scheme.

   Finally, U sets its decryption key $d_{\mathsf{ID}} = (d, d')$.

Encryption To encrypt a message $M \in \mathbb{G}_2$ under an identity ID, choose a random value $s \in \mathbb{Z}_p$ and a subset $\mathcal{J}_j \subset [n]$ of size k for each $j \in [m]$. Create $m+1$ random shares for the message $M = M_0 + \cdots + M_m$. Compute the ciphertext $\chi = (\{\mathcal{J}_j\}_{j \in [m]}, C, C')$ as follows:

$$C = \Big[\{c_i = M_i \cdot Y_i^s\}_{i \in [\ell]}, \{(C_i = U_{i,\mathsf{ID}_i}{}^s) : i \in [\ell]\}, \{(C_{i,j} = T_{i,j}{}^s) : j \in [m], i \in \mathcal{J}_j\}\Big]$$

$$C' = \Big[\{c'_i = M_i \cdot Y_i'^s\}_{i \in [\ell]}, \{(C'_i = U'_{i,\mathsf{ID}_i}{}^s) : i \in [\ell]\}, \{(C'_{i,j} = T'_{i,j}{}^s) : j \in [m], i \in ([n] \setminus \mathcal{J}_j)\}\Big]$$

Observe that in the second "helper" encryption the complement of the dummy attributes are selected. This will allow the user to recover the share in the second encryption if and only if he is not able to do so in the first.

Decryption To decrypt the ciphertext $\chi = (\{\mathcal{J}_j\}_{j \in [m]}, C, C')$ we first run a ciphertext sanity check as in the previous construction

If the ciphertext sanity check succeeds, recover the message M by selecting (for each $j \in [m]$) a set $S_j \subset \mathcal{I}_j \cap \mathcal{J}_j$ of threshold size τ or a set $S_j \subset \mathcal{I}_j \cap ([n] \setminus \mathcal{J}_j)$ of threshold size $k - \tau + 1$. One of these will always be possible by construction. The same decryption operations are performed as in the original scheme to recover all of the shares of the message.

Trace The tracing algorithm will be the same as before, except in the "helper" ciphertext we encrypt random messages.

We omit the proof of security and we mention that the intuition as to why it is secure is because the helper ciphertext is only useful a negligible fraction of the time.