Fuzzy Identity Based Encryption from Lattices

Shweta Agrawal* Xavier Boyen† Vinod Vaikuntanathan‡ Panagiotis Voulgaris§
Hoeteck Wee¶

Abstract

Cryptosystems based on the hardness of lattice problems have recently acquired much importance due to their average-case to worst-case equivalence, their conjectured resistance to quantum cryptanalysis, their ease of implementation and increasing practicality, and, lately, their promising potential as a platform for constructing advanced functionalities.

In this work, we construct "Fuzzy" Identity Based Encryption from the hardness of the standard Learning With Errors (LWE) problem. We give CPA and CCA secure variants of our construction, for small and large universes of attributes. All are secure against selective-identity attacks in the standard model.

Our construction is made possible by observing certain special properties that secret sharing schemes need to satisfy in order to be useful for Fuzzy IBE. We discuss why further extensions are not as easy as they may seem. As such, ours is among the first examples of advanced-functionality cryptosystem from lattices that goes "beyond IBE".

*UCLA. Research supported in part from a DARPA/ONR PROCEED award, and NSF grants 1118096, 1065276, 0916574 and 0830803. E-mail: shweta.a@gmail.com

†PARC—Palo Alto Research Center. E-mail: xb@boyen.org

‡Microsoft Research and University of Toronto. E-mail: vinodv@alum.mit.edu

§UCSD. E-mail: pvoulgar@cs.ucsd.edu

¶CUNY. E-mail: hoeteck@alum.mit.edu

1 Introduction

Lattices have recently emerged as a powerful mathematical platform on which to build a rich variety of cryptographic primitives. Starting from the work of Ajtai [4], lattices have been used to construct one-way functions and collision-resistant hash functions [4, 27], public-key encryption [6, 32, 33], identity-based encryption schemes [22, 16, 1, 2], trapdoor functions [22] and even fully homomorphic encryption [20, 21, 15]. Lattice-based cryptography is attractive not only as a fallback in case factoring and discrete-log turn out to be easy (which they are on quantum computers), but it is also an end in its own right — lattice-based systems resist quantum and sub-exponential attacks, and they are efficient, admit highly parallel implementations and are potentially quite practical.

At the same time, encryption schemes have grown more and more sophisticated, and able to support complex access policies. Specifically, the idea of *functional encryption* has emerged as a new paradigm for encryption. In functional encryption in its broad sense, a secret key allows its holder to unlock data (or some piece or function of the data) based on policies and logic, rather than by merely addressing the recipient(s). The usefulness of such a primitive is evident — access to encrypted data moves beyond mere enumeration to potentially arbitrary functions.

Since its introduction with Fuzzy Identity-Based Encryption by Sahai and Waters [34], several systems have emerged that move beyond the traditional "designated recipient(s)" paradigm of encryption. In this line of work, the key (or, in some variants, the ciphertext) is associated with a predicate, say f, while the ciphertext (or the key) is associated with an attribute vector, say x. Decryption succeeds if and only if $f(x) = 1$. Specifically, *attribute-based encryption* [23, 29, 9, 17, 25, 26] specifically refers to the case where the predicate is a Boolean formula to which the attributes provide binary inputs. Fuzzy IBE is a special case where f is a k-out-of-ℓ threshold function. In *predicate encryption* [24, 25], the predicate f is to be evaluated without leaking anything about the attributes other than the binary output of $f(x)$, i.e., achieving *attribute hiding* along with the standard *payload hiding*; known constructions are currently limited to inner-product predicates between embedded constants and attributes living in some field, though.

Notably, all known instantiations of Functional Encryption are based on bilinear maps on elliptic curves — and most are based on the IBE framework by Boneh and Boyen [10]. Non-pairing constructions have remained elusive, even though factoring-based IBE has been known since 2001 [18, 12] and lattice-based IBE since 2008 [22]. This is even more notable in the lattice world, where we now have an array of sophisticated (hierarchical) IBE schemes [22, 3, 16, 1, 2], but the construction of more expressive functional encryption schemes has been lagging far behind.

Our contributions. We take the first step in this direction by constructing a fuzzy identity-based encryption (fuzzy IBE) scheme based on lattices. A fuzzy IBE scheme is exactly like an identity-based encryption scheme except that (considering identities as bit-vectors in $\{0,1\}^n$) a ciphertext encrypted under an identity $\mathsf{id}_{\mathsf{enc}}$ can be decrypted using the secret key corresponding to any identity $\mathsf{id}_{\mathsf{dec}}$ that is "close enough" to $\mathsf{id}_{\mathsf{enc}}$. Examples arise when using one's biometric information as the identity, but also in general access control systems that permit access as long as the user satisfies a certain number of conditions.

Our construction is secure in the selective security model under the learning with errors (LWE) assumption and thus, by the results of [33, 31], secure under the worst-case hardness of "short vector problems" on arbitrary lattices. We then extend our construction to handle large universes, and to resist chosen ciphertext (CCA) attacks. Finally, we point out some difficulties involved in extending our approach to functional encryption systems.

Perspective. As reported by Boneh, Sahai and Waters in [14], the inability to move beyond inner products in Predicate Encryption stems from the "bi" in Bilinear maps. This calls to mind the well-known situation of yore, in partially homomorphic encryption [13] where the "bi" of bilinear maps allowed the authors to compute only a single multiplication of ciphertexts. Gentry overcame this obstacle [20, 21] by using *lattices* to construct fully homomorphic encryption. Thus, it is quite likely, in our opinion, that lattices might be the tool of choice for taking predicate encryption beyond inner products.

1.1 Overview of our Construction

Our construction borrows ideas from the pairing-based fuzzy IBE scheme of Sahai and Waters [34] and the lattice identity-based encryption scheme of [3, 16], together with an interesting observation about the Shamir secret-sharing scheme and the Lagrange interpolation formula.

First, consider the setting where the identities are ℓ-bit strings. This corresponds to the setting where there are ℓ attributes, and each attribute can take two values (either 0 or 1). Decryption using $\mathsf{SK}_{\mathsf{id}}$ succeeds on a ciphertext encrypted under identity id' if the bitwise difference of id and id' has Hamming weight at most k. We then show how to extend it to the case where the universe of attributes is (exponentially) large in a rather generic way.

Previous lattice-based IBE. We begin by recalling the IBE schemes of [3, 16], which we view as fuzzy IBE schemes where $k = \ell$. The public parameters consist of 2ℓ matrices $(\mathbf{A}_{1,0}, \mathbf{A}_{1,1}, \ldots, \mathbf{A}_{\ell,0}, \mathbf{A}_{\ell,1}) \in \mathbb{Z}_q^{n\times m}$ (where n is the security parameter, q is a small prime, and $m \approx n \log q$ is a parameter of the system) and a vector $\mathbf{u} \in \mathbb{Z}_q^n$. The master secret key then consists of the trapdoors $\mathbf{T}_{i,b}$ corresponding to each matrix $\mathbf{A}_{i,b}$.

We view the secret key derivation in the IBE scheme as a two-step procedure that proceeds as follows: on input an identity id:

1. First, *secret-share* the vector $\mathbf{u}$ into ℓ vectors $\mathbf{u}_1, \ldots, \mathbf{u}_\ell$ which are uniformly random in $\mathbb{Z}_q^n$ subject to the condition that $\sum_{i=1}^{\ell} \mathbf{u}_i = \mathbf{u}$.
2. The secret key $\mathsf{SK}_{\mathsf{id}}$ is then a vector $(\mathbf{e}_1, \ldots, \mathbf{e}_\ell) \in (\mathbb{Z}^m)^\ell$, where
$$\mathsf{SK}_{\mathsf{id}} \doteq (\mathbf{e}_1, \ldots, \mathbf{e}_\ell) \quad \text{and} \quad \mathbf{A}_{i,\mathsf{id}_i}\mathbf{e}_i = \mathbf{u}_i$$
The secret key $\mathbf{e}_i$ is computed using the trapdoor $\mathbf{T}_{i,\mathsf{id}_i}$ using the Gaussian sampling algorithm of [22].

This is a different, yet completely equivalent, way to view the secret key derivation in the IBE schemes of [3, 16].

To encrypt for an identity id in these schemes, one chooses a vector $\mathbf{s} \in \mathbb{Z}_q^n$ and "small error terms" $\mathbf{x}_1, \ldots, \mathbf{x}_\ell \in \mathbb{Z}^m$ and $x' \in \mathbb{Z}$, and outputs

$$\mathsf{CT}_{\mathsf{id}} \doteq \mathsf{IBE.Enc}(\mathsf{id}, b \in \{0,1\}) \doteq (\mathbf{A}_{1,\mathsf{id}_1}^T\mathbf{s} + \mathbf{x}_i, \ldots, \mathbf{A}_{\ell,\mathsf{id}_\ell}^T\mathbf{s} + \mathbf{x}_\ell, \mathbf{u}^T\mathbf{s} + x' + b\lfloor q/2 \rfloor)$$

The key observation in decryption is that if $\mathsf{id} = \mathsf{id}'$, then "pairing" each component of $\mathsf{CT}_{\mathsf{id}'}$ and $\mathsf{SK}_{\mathsf{id}}$ gives us a number that is approximately $\mathbf{u}_i^T \mathbf{s}$. Namely,

$$\mathbf{e}_i^T (\mathbf{A}_{i,\mathsf{id}_i}^T \mathbf{s} + \mathbf{x}_i) = (\mathbf{A}_{i,\mathsf{id}_i}^T \mathbf{e}_i)^T \mathbf{s} + \mathbf{e}_i^T \mathbf{x}_i = \mathbf{u}_i^T \mathbf{s} + \mathbf{e}_i^T \mathbf{x}_i \approx \mathbf{u}_i^T \mathbf{s} \tag{1}$$

By linearity, we can then add up these terms and obtain (approximately) $\mathbf{u}^T \mathbf{s}$. The "approximation" we get here is not terrible, since the error terms $\mathbf{e}_i^T \mathbf{x}_i$ are small, and we add up only ℓ of them. Thus, the magnitude of the error remains much smaller than $q/2$, which is sufficient for decryption.

Our approach. A natural thought to extend this methodology to fuzzy IBE is to use Shamir's k-out-of-ℓ secret-sharing scheme in the first step of the key derivation procedure. Since reconstructing the secret in Shamir's scheme involves computing a linear combination of the shares, we can hope to do decryption as before. As it turns out, the resulting scheme is in fact *neither correct nor secure*. For simplicity, we focus on the issue of correctness in this section.

Recall that correctness of the previous lattice-based IBE schemes lies in bounding the decryption "error terms" $\mathbf{e}_i^T \mathbf{x}_i$. More concretely, the analysis bounds the "cummulative error term"

$$x - \sum_{i=1}^{k} \mathbf{e}_i^T \mathbf{x}_i$$

by $q/4$. Upon instantiating the previous schemes with Shamir's secret-sharing scheme, we need to bound a new cummulative error term, which is given by:

$$x - \sum_{i \in S} L_i \mathbf{e}_i^T \mathbf{x}_i$$

Here, L_i are the fractional Lagrangian coefficients used in reconstructing the secret, interpreted as elements in $\mathbb{Z}_q$ and S identifies the subset of shares used in reconstruction. Indeed, while we can bound both the numerator and denominator in L_i as a fraction of integers, once interpreted as an element in $\mathbb{Z}_q$, the value L_i may be arbitrarily large.

The key idea in our construction is to "clear the denominators". Let $D := (\ell!)^2$ be a sufficiently large constant, so that $DL_i \in \mathbb{Z}$ for all i. Then, we multiply D into the noise vector, that is, the ciphertext is now generated as follows:

$$\mathsf{CT}_{\mathsf{id}} \doteq \mathsf{IBE.Enc}(\mathsf{id}, b \in \{0,1\}) \doteq (\mathbf{A}_{1,\mathsf{id}_1}^T \mathbf{s} + D\mathbf{x}_i, \ldots, \mathbf{A}_{\ell,\mathsf{id}_\ell}^T \mathbf{s} + D\mathbf{x}_\ell, \mathbf{u}^T \mathbf{s} + Dx' + b\lfloor q/2 \rfloor)$$

For correctness, it now suffices to bound the expression:

$$Dx - \sum_{i \in S} DL_i \mathbf{e}_i^T \mathbf{x}_i$$

by $q/4$. Now, further observe that each DL_i is an integer bounded by D^2, so it suffices to pick the noise vectors so that they are bounded by $q/4D\ell$ with overwhelming probability.

Thus, for appropriate parameter settings, we get a fuzzy IBE scheme based on the classical hardness of computing a sub-exponential approximation to "short vector problems" on arbitrary lattices.

Additional related work. The idea of using Shamir's secret-sharing scheme in lattice-based cryptography appears in the work of Bendlin and Damgård [8] on threshold cryptosystems. The security of their scheme, as with ours, relies on the hardness of computing sub-exponential approximation for lattice problems. In more detail, their scheme uses a pseudorandom secret-sharing from [19] in order to share a value in some interval, for which they do not have to address the issue of bounding the size of Lagrangian coefficients. Our idea of "clearing the denominator" is inspired by the work on factoring-based threshold cryptography (e.g. [35]), where the technique is used to handle a different technical issue: evaluating fractional Lagrangian coefficients over an "unknown" modulus $\phi(N)$, where N is a public RSA modulus.

2 Preliminaries

Notation: We use uppercase boldface alphabet for matrices, as in $\mathbf{A}$, lowercase boldface characters for vectors, as in $\mathbf{e}$, and lowercase regular characters for scalars, as in v. We say that a function $f : \mathbb{R}^+ \to \mathbb{R}^+$ is negligible if for all $d > d_0$ we have $f(\lambda) < 1/\lambda^d$ for sufficiently large λ. We write $f(\lambda) < \text{negl}(\lambda)$. For any ordered set $S = \{\mathbf{s}_1, \ldots, \mathbf{s}_k\} \in \mathbb{R}^m$ of linearly independent vectors, we define $\|\widetilde{\mathbf{S}}\| = \max_j \|\tilde{\mathbf{s}_j}\|$, where $\tilde{\mathbf{S}} = \{\tilde{\mathbf{s}_1}, \ldots, \tilde{\mathbf{s}_k}\}$ refers to the Gram-Schmidt orthogonalization of $\mathbf{S}$, and $\| \cdot \|$ refers to the euclidean norm. We let $\sigma_{\text{TG}} := O(\sqrt{n \log q}\,)$ denote the maximum (w.h.p.) Gram-Schmidt norm of a basis produced by $\mathsf{TrapGen}(q, n)$.

2.1 Definition: Fuzzy IBE

A Fuzzy Identity Based Encryption scheme consists of the following four algorithms:

$\mathsf{Fuzzy.Setup}(\lambda, \ell) \to (\mathsf{PP}, \mathsf{MK})$: This algorithm takes as input the security parameter λ and the maximum length of identities ℓ. It outputs the public parameters PP and a master key MK.

$\mathsf{Fuzzy.Extract}(\mathsf{MK}, \mathsf{PP}, \mathsf{id}, k) \to \mathsf{SK}$: This algorithm takes as input the master key MK, the public parameters PP, an identity id and the threshold $k \leq \ell$. It outputs a decryption key $\mathsf{SK}_{\mathsf{id}}$.

$\mathsf{Fuzzy.Enc}(\mathsf{PP}, b, \mathsf{id}') \to \mathsf{CT}$: This algorithm takes as input: a message bit b, an identity id', and the public parameters PP. It outputs the ciphertext $\mathsf{CT}_{\mathsf{id}'}$.

$\mathsf{Fuzzy.Dec}(\mathsf{PP}, \mathsf{CT}_{\mathsf{id}'}, \mathsf{SK}_{\mathsf{id}}) \to b$: This algorithm takes as input the ciphertext $\mathsf{CT}_{\mathsf{id}'}$, the decryption key $\mathsf{SK}_{\mathsf{id}}$ and the public parameters PP. It outputs the message b if $|\mathsf{id} \cap \mathsf{id}'| \geq k$.

2.2 Security Model for Fuzzy IBE

We consider the notion of ciphertext privacy, which implies both semantic security and recipient anonymity. Ciphertext privacy against a selective security, chosen plaintext attack is defined by the following game.

Target: The adversary declares the challenge identity, id^*, that he wishes to be challenged upon.

Setup: The challenger runs the Setup algorithm of Fuzzy-IBE and gives the public parameters to the adversary.

Phase 1: The adversary is allowed to issue queries for private keys for identities id_j of its choice, as long as $|\mathsf{id}_j \cap \mathsf{id}^*| < k; \forall j$

Challenge: The adversary submits a message to encrypt. The challenger encrypts the message with the challenge identity id^* and then flips a random coin r. If $r = 1$, the ciphertext is given to the adversary, otherwise a random element of the ciphertext space is returned.

Phase 2: Phase 1 is repeated.

Guess: The adversary outputs a guess r' of r. The advantage of an adversary A in this game is defined as $|\Pr[r' = r] - \frac{1}{2}|$

A Fuzzy Identity Based Encryption scheme is secure in the Selective-Set model of security if all polynomial time adversaries have at most a negligible advantage in the Selective-Set game.

The adaptive version of the above game is identical except it does not have the target step, hence the adversary is allowed to choose an attack identity adversarially.

3 Preliminaries: Lattices

Throughout the paper, we let the parameters $q = q(\lambda), m = m(\lambda)$, $n = n(\lambda)$ are polynomial functions of the security parameter λ.

3.1 Random Integer Lattices

Definition 1. Let $\mathbf{B} = \left[\ \mathbf{b}_1 \ \middle|\ \ldots \ \middle|\ \mathbf{b}_m\ \right] \in \mathbb{R}^{m \times m}$ be an $m \times m$ matrix whose columns are linearly independent vectors $\mathbf{b}_1, \ldots, \mathbf{b}_m \in \mathbb{R}^m$. The m-dimensional full-rank lattice Λ generated by $\mathbf{B}$ is the infinite periodic set,

$$\Lambda = \mathcal{L}(\mathbf{B}) = \left\{ \ \mathbf{y} \in \mathbb{R}^m \quad \text{s.t.} \quad \exists \mathbf{s} = (s_1, \ldots, s_m) \in \mathbb{Z}^m \ , \ \ \mathbf{y} = \mathbf{B}\,\mathbf{s} = \sum_{i=1}^{m} s_i\,\mathbf{b}_i \ \right\}$$

Here, we are interested in integer lattices, i.e, infinite periodic subsets of $\mathbb{Z}^m$, that are invariant under translation by multiples of some integer q in each of the coordinates.

Definition 2. For q prime and $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ and $\mathbf{u} \in \mathbb{Z}_q^n$, define:

$$\begin{aligned} \Lambda_q^{\perp}(\mathbf{A}) &= \left\{ \ \mathbf{e} \in \mathbb{Z}^m \quad \text{s.t.} \quad \mathbf{A}\,\mathbf{e} = 0 \pmod{q} \ \right\} \\ \Lambda_q^{\mathbf{u}}(\mathbf{A}) &= \left\{ \ \mathbf{e} \in \mathbb{Z}^m \quad \text{s.t.} \quad \mathbf{A}\,\mathbf{e} = \mathbf{u} \pmod{q} \ \right\} \end{aligned}$$

3.2 Trapdoors for Lattices: The algorithm $\mathsf{TrapGen}$

Ajtai [5] showed how to sample an essentially uniform matrix $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ with an associated full-rank set $\mathbf{T_A} \subset \Lambda^{\perp}(\mathbf{A})$ of *low-norm vectors*. We will use an improved version of Ajtai's basis sampling algorithm due to Alwen and Peikert [7]:

Proposition 3 ([7])**.** Let $n = n(\lambda), q = q(\lambda), m = m(\lambda)$ be positive integers with $q \geq 2$ and $m \geq 5n \log q$. There exists a probabilistic polynomial-time algorithm $\mathsf{TrapGen}$ that outputs a pair $\mathbf{A} \in \mathbb{Z}_q^{n\times m}, \mathbf{T_A} \in \mathbb{Z}_q^{m\times m}$ such that $\mathbf{A}$ is statistically close to uniform and $\mathbf{T_A}$ is a basis for $\Lambda^{\perp}(\mathbf{A})$ with length $L = \|\widetilde{\mathbf{T_A}}\| \leq m \cdot \omega(\sqrt{\log m})$ with all but $n^{-\omega(1)}$ probability.

3.3 Discrete Gaussians

Definition 4. Let $m \in \mathbb{Z}_{>0}$ be a positive integer and $\Lambda \subset \mathbb{R}^m$ an m-dimensional lattice. For any vector $\mathbf{c} \in \mathbb{R}^m$ and any positive parameter $\sigma \in \mathbb{R}_{>0}$, we define:

$\rho_{\sigma,c}(\mathbf{x}) \ = \exp\left(-\pi\frac{\|\mathbf{x}-\mathbf{c}\|^2}{\sigma^2}\right)$: a Gaussian-shaped function on $\mathbb{R}^m$ with center $\mathbf{c}$ and parameter σ,

$\rho_{\sigma,\mathbf{c}}(\Lambda) \ = \sum_{\mathbf{x}\in\Lambda}\rho_{\sigma,\mathbf{c}}(\mathbf{x})$: the (always converging) discrete integral of $\rho_{\sigma,\mathbf{c}}$ over the lattice Λ,

$\mathcal{D}_{\Lambda,\sigma,\mathbf{c}}$: the discrete Gaussian distribution over Λ with center $\mathbf{c}$ and parameter σ,

$$\forall \mathbf{y} \in \Lambda \quad , \quad \mathcal{D}_{\Lambda,\sigma,\mathbf{c}}(\mathbf{y}) = \frac{\rho_{\sigma,\mathbf{c}}(\mathbf{y})}{\rho_{\sigma,\mathbf{c}}(\Lambda)}$$

For notational convenience, $\rho_{\sigma,0}$ and $\mathcal{D}_{\Lambda,\sigma,0}$ are abbreviated as ρ_σ and $\mathcal{D}_{\Lambda,\sigma}$.

3.3.1 Sampling Discrete Gaussians over Lattices

Gentry, Peikert and Vaikuntanathan [22] construct the following algorithm for sampling from the discrete Gaussian $\mathcal{D}_{\Lambda,\sigma,\mathbf{c}}$, given a basis $\mathbf{B}$ for the m-dimensional lattice Λ with $\sigma \geq \|\widetilde{\mathbf{B}}\| \cdot \omega(\sqrt{\log m})$. Specialized to the case of random lattices, they show an algorithm

Algorithm $\textsf{SamplePre}(\mathbf{A}, \mathbf{T_A}, \mathbf{u}, \sigma)$: On input a matrix $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ with 'short' trapdoor basis $\mathbf{T_A}$ for $\Lambda_q^{\perp}(\mathbf{A})$, a target image $\mathbf{u} \in \mathbb{Z}_q^n$ and a Gaussian parameter $\sigma \geq \|\widetilde{\mathbf{T_A}}\| \cdot \omega(\sqrt{\log m})$, outputs a sample $\mathbf{e} \in \mathbb{Z}_q^m$ from a distribution that is within negligible statistical distance of $\mathcal{D}_{\Lambda_q^{\mathbf{u}}(\mathbf{A}),\sigma}$.

3.4 Sampling from an "Encryption" matrix

We will also need the following algorithm defined in [1]:

Algorithm $\mathsf{SampleLeft}(\mathbf{A}, \mathbf{M}_1, \mathbf{T_A}, \mathbf{u}, \sigma)$:

Inputs: a rank n matrix $\mathbf{A}$ in $\mathbb{Z}_q^{n\times m}$ and a matrix $\mathbf{M}_1$ in $\mathbb{Z}_q^{n\times m_1}$,
a "short" basis $\mathbf{T_A}$ of $\Lambda_q^{\perp}(\mathbf{A})$ and a vector $\mathbf{u} \in \mathbb{Z}_q^n$,
a gaussian parameter $\sigma > \|\widetilde{\mathbf{T_A}}\| \cdot \omega(\sqrt{\log(m+m_1)})$. (2)

Output: Let $\mathbf{F}_1 := (\mathbf{A} \mid \mathbf{M}_1)$. The algorithm outputs a vector $\mathbf{e} \in \mathbb{Z}^{m+m_1}$ sampled from a distribution statistically close to $\mathcal{D}_{\Lambda_q^{\mathbf{u}}(\mathbf{F}_1),\sigma}$. In particular, $\mathbf{e} \in \Lambda_q^{\mathbf{u}}(\mathbf{F}_1)$.

3.5 Two Lemmas to Bound Norms

The following lemma about the distribution $\overline{\Psi}_\alpha$ will be needed to show that decryption works correctly. The proof is implicit in [22, Lemma 8.2].

Lemma 5. *Let* $\mathbf{e}$ *be some vector in* $\mathbb{Z}^m$ *and let* $\mathbf{y} \overset{R}{\leftarrow} \overline{\Psi}_\alpha^m$*. Then the quantity* $|\mathbf{e}^\top \mathbf{y}|$ *treated as an integer in* $[0, q-1]$ *satisfies*

$$|\mathbf{e}^\top \mathbf{y}| \leq \|\mathbf{e}\|\, q\alpha\, \omega(\sqrt{\log m}\,) + \|\mathbf{e}\|\sqrt{m}/2$$

with all but negligible probability in m*.*

Micciancio and Regev showed that the norm of vectors sampled from discrete Gaussians is small with high probability.

Proposition 6 ([28])**.** For any lattice Λ of integer dimension m, any lattice point $\mathbf{c}$, and any two reals $\epsilon \in (0,1)$ and $\sigma \geq \omega(\sqrt{\log m})$,

$$\Pr\left\{ \mathbf{x} \sim \mathcal{D}_{\Lambda,\sigma,\mathbf{c}} \;:\; \|\mathbf{x} - \mathbf{c}\| > \sqrt{m}\,\sigma \right\} \;\leq\; \frac{1+\epsilon}{1-\epsilon}\, 2^{-m}$$

3.6 Hardness Assumption

The LWE (learning with errors) problem was first defined by [33], and has since been extensively studied and used. For polynomially bounded modulus q, the computational and decisional versions are equivalent. We give the following convenient restatement of the latter:

Definition 7. Consider a prime q, a positive integer n, and a distribution χ over $\mathbb{Z}_q$, all public. An $(\mathbb{Z}_q, n, \chi)$-LWE problem instance consists of access to an unspecified challenge oracle $\mathcal{O}$, being, either, a noisy pseudo-random sampler $\mathcal{O}_s$ carrying some constant random secret key $\mathbf{s} \in \mathbb{Z}_q^n$, or, a truly random sampler $\mathcal{O}_\$$, whose behaviors are respectively as follows:

$\mathcal{O}_s$**:** outputs noisy pseudo-random samples of the form $(\mathbf{w}_i, v_i) = \big(\mathbf{w}_i,\ \mathbf{w}_i^T\,\mathbf{s} + x_i\big) \in \mathbb{Z}_q^n \times \mathbb{Z}_q$, where, $\mathbf{s} \in \mathbb{Z}_q^n$ is a uniformly distributed persistent secret key that is invariant across invocations, $x_i \in \mathbb{Z}_q$ is a freshly generated ephemeral additive noise component with distribution χ, and $\mathbf{w}_i \in \mathbb{Z}_q^n$ is a fresh uniformly distributed vector revealed as part of the output.

$\mathcal{O}_\$$**:** outputs truly random samples $\big(\mathbf{w}_i, v_i\big) \in \mathbb{Z}_q^n \times \mathbb{Z}_q$, drawn independently uniformly at random in the entire domain $\mathbb{Z}_q^n \times \mathbb{Z}_q$.

The $(\mathbb{Z}_q, n, \chi)$-LWE problem statement, or LWE for short, allows an unspecified number of queries to be made to the challenge oracle $\mathcal{O}$, with no stated prior bound. We say that an algorithm $\mathcal{A}$ decides the $(\mathbb{Z}_q, n, \chi)$-LWE problem if $\big|\Pr[\mathcal{A}^{\mathcal{O}_s} = 1] - \Pr[\mathcal{A}^{\mathcal{O}_\$} = 1]\big|$ is non-negligible for a random $s \in \mathbb{Z}_q^n$.

The confidence in the hardness of the LWE problem stems in part from a result of Regev [33] which shows that the for certain noise distributions χ, the LWE problem is as hard as the worst-case SIVP and GapSVP under a quantum reduction (see also [30]). A classical reduction with related parameters was later obtained by Peikert [31].

Proposition 8 ([33])**.** Consider a real parameter $\alpha = \alpha(n) \in (0,1)$ and a prime $q = q(n) > 2\sqrt{n}/\alpha$. Denote by $\mathbb{T} = \mathbb{R}/\mathbb{Z}$ the group of reals $[0,1)$ with addition modulo 1. Denote by Ψ_α the distribution over $\mathbb{T}$ of a normal variable with mean 0 and standard deviation $\alpha/\sqrt{2\pi}$ then reduced modulo 1. Denote by $\lfloor x \rceil = \lfloor x + \frac{1}{2} \rfloor$ the nearest integer to the real $x \in \mathbb{R}$. Denote by $\bar{\Psi}_\alpha$ the discrete distribution over $\mathbb{Z}_q$ of the random variable $\lfloor q\,X \rceil \mod q$ where the random variable $X \in \mathbb{T}$ has distribution Ψ_α.

Then, if there exists an efficient, possibly quantum, algorithm for deciding the $(\mathbb{Z}_q, n, \bar{\Psi}_\alpha)$-LWE problem, there exists a quantum $q \cdot \mathsf{poly}(n)$-time algorithm for approximating the SIVP and GapSVP problems, to within $\widetilde{O}(n/\alpha)$ factors in the ℓ_2 norm, in the worst case.

Since the best known algorithms for 2^k-approximations of gapSVP and SIVP run in time $2^{\widetilde{O}(n/k)})$, it follows from the above that the LWE problem with the noise ratio $\alpha = 2^{-n^\epsilon}$ is likely hard for some constant $\epsilon < 1$.

4 The Fuzzy IBE Scheme

Let $\lambda \in \mathbb{Z}^+$ be a security parameter. Let $q = q(\lambda)$ be a prime, $n = n(\lambda)$ and $m = m(\lambda)$ two positive integers, and $\sigma = \sigma(\lambda)$ and $\alpha = \alpha(\lambda)$ two positive Gaussian parameters. We assume that $\mathsf{id} \in \{0,1\}^\ell$ for some $\ell \in \mathbb{N}$.

4.1 Construction

Fuzzy.Setup$(1^\lambda, 1^\ell)$**:** On input a security parameter λ, and identity size ℓ, do:

1. Use algorithm $\mathsf{TrapGen}(1^\lambda)$ (from Proposition 3) to select 2ℓ uniformly random $n \times m$-matrices $\mathbf{A}_{i,b} \in \mathbb{Z}_q^{n \times m}$ (for all $i \in [\ell], b \in \{0,1\}$) together with a full-rank set of vectors $\mathbf{T}_{i,b} \subseteq \Lambda_q^\perp(\mathbf{A}_{i,b})$ such that $\|\widetilde{\mathbf{T}_{i,b}}\| \leq m \cdot \omega(\sqrt{\log m})$.
2. Select a uniformly random vector $\mathbf{u} \in \mathbb{Z}_q^n$.
3. Output the public parameters and master key,

$$\mathsf{PP} = \Big(\{\mathbf{A}_{i,b}\}_{i \in [\ell], b \in \{0,1\}}, \mathbf{u} \Big) \quad ; \quad \mathsf{MK} = \Big(\{\mathbf{T}_{i,b}\}_{i \in [\ell], b \in \{0,1\}} \Big)$$

Fuzzy.Extract$(\mathsf{PP}, \mathsf{MK}, \mathsf{id}, k)$**:** On input public parameters PP, a master key MK, an identity $\mathsf{id} \in \{0,1\}^\ell$ and threshold $k \leq \ell$, do:

1. Construct ℓ shares of $\mathbf{u} = (u_1, ..., u_n) \in \mathbb{Z}_q^n$ using a Shamir secret-sharing scheme applied to each co-ordinate of $\mathbf{u}$ independently. Namely, for each $j \in [n]$, choose a uniformly random polynomial $p_j \in \mathbb{Z}_q[x]$ of degree $k - 1$ such that $p_j(0) = u_j$.
   Construct the j^{th} share vector

$$\hat{\mathbf{u}}_j = (\hat{u}_{j,1}, \ldots, \hat{u}_{j,n}) \stackrel{\text{def}}{=} (p_1(j), p_2(j), \ldots, p_n(j)) \in \mathbb{Z}_q^n$$

   Looking ahead (to decryption), note that for all $J \subset [\ell]$ such that $|J| \geq k$, we can compute fractional Lagrangian coefficients L_j such that $\mathbf{u} = \sum_{j \in J} L_j \cdot \hat{\mathbf{u}}_j \pmod{q}$. That is, we interpret L_j as a fraction of integers, which we can also evaluate $\pmod{q}$.

2. Using trapdoor MK and the algorithm $\mathsf{SamplePre}$ from Section 3.3.1, find $\mathbf{e}_j \in \mathbb{Z}^m$ such that $\mathbf{A}_{j,\mathsf{id}_j} \cdot \mathbf{e}_j = \hat{\mathbf{u}}_j$, for $j \in [\ell]$.
3. Output the secret key for id as $(\mathbf{e}_1, \ldots, \mathbf{e}_\ell)$.

Fuzzy.Enc$(\mathsf{PP}, \mathsf{id}, b)$**:** On input public parameters PP, an identity id, and a message $b \in \{0,1\}$, do:

1. Let $D \stackrel{\text{def}}{=} (\ell!)^2$.
2. Choose a uniformly random $\mathbf{s} \stackrel{R}{\leftarrow} \mathbb{Z}_q^n$.
3. Choose a noise term $x \leftarrow \chi_{\{\alpha,q\}}$ and $\mathbf{x}_i \leftarrow {\chi_{\{\alpha,q\}}}^m$,
4. Set $c_0 \leftarrow \mathbf{u}^\top \mathbf{s} + Dx + b\lfloor \frac{q}{2} \rfloor \in \mathbb{Z}_q$.
5. Set $\mathbf{c}_i \leftarrow {\mathbf{A}_{i,\mathsf{id}_i}}^\top \mathbf{s} + D\mathbf{x}_i \in \mathbb{Z}_q^m$ for all $i \in [\ell]$.
6. Output the ciphertext $\mathsf{CT}_{\mathsf{id}} := (c_0, \{\mathbf{c}_i\}_{i \in [\ell]})$.

Fuzzy.Dec$(\mathsf{PP}, \mathsf{SK}_{\mathsf{id}}, \mathsf{CT}_{\mathsf{id}'})$**:** On input parameters PP, a private key $\mathsf{SK}_{\mathsf{id}}$, and a ciphertext $\mathsf{CT}_{\mathsf{id}'}$:

1. Let $J \subset [\ell]$ denote the set of matching bits in id and id'. If $|J| < k$, output $\perp$. Otherwise, we can compute fractional Lagrangian coefficients L_j so that
$$\sum_{j \in J} L_j \mathbf{A}_j \mathbf{e}_j = \mathbf{u} \pmod q$$
2. Compute $r \leftarrow c_0 - \sum_{j \in J} L_j \cdot \mathbf{e}_j^\top \mathbf{c}_j \pmod q$. View it as the integer $r \in [-\lfloor \frac{q}{2} \rfloor, \lfloor \frac{q}{2} \rfloor) \subset \mathbb{Z}$.
3. If $|r| < \frac{q}{4}$, output 0, else output 1.

4.1.1 Correctness

To establish correctness for decryption, we only need to consider the case $|J| \geq k$. Let L_j be the fractional Lagrangian coefficients as described above. Then,

$$\begin{aligned} r &= c_0 - \sum_{j \in J} L_j\, \mathbf{e}_j^\top\, \mathbf{c}_j \pmod q \qquad (3) \\ &= \mathbf{u}^\top \mathbf{s} + Dx + b\left\lfloor \frac{q}{2} \right\rfloor - \sum_{j \in J} L_j\, \mathbf{e}_j^\top ({\mathbf{A}_j}^\top \mathbf{s} + D \cdot \mathbf{x}_j) \pmod q \\ &= b\left\lfloor \frac{q}{2} \right\rfloor + \underbrace{\Big(\mathbf{u}^\top \mathbf{s} - \sum_{j \in J} (L_j\, \mathbf{A}_j \mathbf{e}_j)^\top \mathbf{s}\Big)}_{=0 \pmod q} + \underbrace{\Big(Dx - \sum_{j \in J} D L_j \mathbf{e}_j^\top \mathbf{x}_j\Big)}_{\approx 0} \pmod q \qquad \approx b\left\lfloor \frac{q}{2} \right\rfloor \end{aligned}$$

It suffices to set the parameters so that with overwhelming probability,

$$|Dx - \sum_{j \in J} D L_j \mathbf{e}_j^\top \mathbf{x}_j| \leq D|x| + \sum_{j \in J} D^2 |\mathbf{e}_j^\top \mathbf{x}_j| < q/4 \qquad (4)$$

For the first inequality, we use the following lemma on Lagrangian coefficients which states that the numbers DL_j are integers bounded above by $D^2 \leq (\ell!)^4$.

Lemma 9. *Let $D = (\ell!)^2$. Given $k \leq \ell$ numbers $I_1, \ldots, I_k \in [1 \ldots \ell]$, define the Lagrangian coefficients*

$$L_j = \prod_{i \neq j} \frac{-I_i}{(I_j - I_i)}$$

Then, for every $1 \leq j \leq k$, DL_j is an integer, and $|DL_j| \leq D^2 \leq (\ell!)^4$.

Proof. To see this, note that the denominator of the j^{th} Lagrange coefficient L_j is of the form

$$d_j = \prod_{i \neq j} (I_j - I_i)$$

The numbers $|I_j - I_i|$ lie in the interval $[-(\ell-1), \ldots, (\ell-1)]$, and they can repeat at most twice (namely, for every number $n \in [\ell]$, there are at most two i, i' such that $|I_j - I_i| = |I_j - I_{i'}|$).

Since each of the factors $I_j - I_i$ can appear at most twice in absolute value, $(\ell!)^2$ divides d_j. Thus, DL_j is an integer. Also,

$$|DL_j| \leq D \cdot \big| \prod_{j \neq i} (-I_i) \big| \leq (\ell!)^3$$

□

4.2 Proof of Security

We show that the Fuzzy IBE construction provides ciphertext privacy under a selective identity attack as in Definition 2.2. Recall that ciphertext privacy means that the challenge ciphertext is indistinguishable from a random element in the ciphertext space. More precisely, we have the following theorem:

Theorem 10. *If there exists a PPT adversary $\mathcal{A}$ with advantage $\epsilon > 0$ against the selective security game for the Fuzzy IBE scheme of Section 4.1, then there exists a PPT algorithm $\mathcal{B}$ that decides the LWE problem with advantage $\epsilon/(\ell + 1)$.*

Proof. Recall from Definition 7 that an LWE problem instance is provided as a sampling oracle $\mathcal{O}$ which can be either truly random $\mathcal{O}_\$$ or noisy pseudo-random $\mathcal{O}_s$ for some secret key $s \in \mathbb{Z}_q^n$. The simulator $\mathcal{B}$ uses the adversary $\mathcal{A}$ to distinguish between the two, and proceeds as follows:

Instance. $\mathcal{B}$ requests from $\mathcal{O}$ and receives $(\ell m + 1)$ LWE samples that we denote as $(\mathbf{w}_1, v_1)$, $\{(\mathbf{w}_1^1, v_1^1), (\mathbf{w}_1^2, v_1^2), \ldots, (\mathbf{w}_1^m, v_1^m)\}, \ldots \{(\mathbf{w}_\ell^1, v_\ell^1), (\mathbf{w}_\ell^2, v_\ell^2), \ldots, (\mathbf{w}_\ell^m, v_\ell^m)\} \in \{\mathbb{Z}_q^n \times \mathbb{Z}_q\}^{(\ell m+1)}$.

Targeting. $\mathcal{A}$ announces to $\mathcal{B}$ the identity it intends to attack, namely id^*.

Setup. $\mathcal{B}$ constructs the system's public parameters PP as follows:

1. The ℓ matrices $\mathbf{A}_{i,\mathsf{id}_i^*}$, $i \in [\ell]$ are chosen from the LWE challenge $\{(\mathbf{w}_i^1), (\mathbf{w}_i^2), \ldots, (\mathbf{w}_i^m)\}_{i \in [\ell]}$. The ℓ matrices $\mathbf{A}_{i,\overline{\mathsf{id}_i^*}}$, $i \in [\ell]$ are chosen using $\mathsf{TrapGen}$ with a trapdoor $\mathbf{T}_{i,\overline{\mathsf{id}_i^*}}$.
2. The vector $\mathbf{u}$ is constructed from the LWE challenge, $\mathbf{u} = \mathbf{w}_1$.

The public parameters are returned to the adversary.

Queries. $\mathcal{B}$ answers each private-key extraction query for identity id as follows:

1. Let $\mathsf{id} \cap \mathsf{id}^* := I \subset [\ell]$ and let $|I| = t < k$. Then, note that $\mathcal{B}$ has trapdoors for the matrices corresponding to the set $\bar{I}$, where $|\bar{I}| = \ell - t$. W.l.o.g., we assume that the first t bits of id are equal to id^*.
2. Represent the shares of $\mathbf{u}$ symbolically as $\hat{\mathbf{u}}_i = \mathbf{u} + \mathbf{a}_1 i + \mathbf{a}_2 i^2 + \ldots + \mathbf{a}_{k-1} i^{k-1}$ where $\mathbf{a}_1, \ldots, \mathbf{a}_{k-1}$ are vector variables of length n each.
3. For i s.t. $\mathsf{id}^*_i = \mathsf{id}_i$, pick $\mathbf{e}_i$ randomly using algorithm $\mathsf{SampleGaussian}$. Set $\hat{\mathbf{u}}_i := \mathbf{A}_{i,\mathsf{id}_i}\mathbf{e}_i$; $i \in [t]$.
4. Since $t \leq k-1$, and there are $k-1$ variables $\mathbf{a}_1 .... \mathbf{a}_{k-1}$, by choosing $k-1-t$ shares $\hat{\mathbf{u}}_{t+1}, \ldots, \hat{\mathbf{u}}_{k-1}$ randomly, the values for $\mathbf{a}_1 .... \mathbf{a}_{k-1}$ are determined. This determines all ℓ shares $\hat{\mathbf{u}_1}, \ldots, \hat{\mathbf{u}_\ell}$.
5. To find $\mathbf{e}_j$ s.t. $\mathbf{A}_{j,\mathsf{id}_j}\mathbf{e}_j = \hat{\mathbf{u}}_j$ for $j = t+1, \ldots \ell$, invoke $\mathsf{SamplePre}(\mathbf{A}_{j,\mathsf{id}_j}, \mathbf{T}_{j,\mathsf{id}_j}, \hat{\mathbf{u}}_j, \sigma)$.
6. Return $(\mathbf{e}_1, \ldots, \mathbf{e}_\ell)$.

Note that the distribution of the public parameters and keys in the real scheme is statistically indistinguishable from that in the simulation.

Challenge. $\mathcal{A}$ outputs a message bit $b^* \in \{0,1\}$. $\mathcal{B}$ responds with a challenge ciphertext for id^*:

1. Let $c_0 = Dv_1 + b\lfloor q/2 \rceil$.
2. Let $\mathbf{c}_i = (Dv_i^1, Dv_i^2 ..... Dv_i^m)$ for $i \in [\ell]$.

Guess. The adversary $\mathcal{A}$ outputs a guess b'. The simulator $\mathcal{B}$ uses that guess to determine an answer on the LWE oracle: Output "`genuine`" if $b' = b^*$, else output "`random`". $\square$

4.3 Parameters

We set the parameters to ensure that the decoding works with high probability, and that the security reductions are meaningful. Our security parameter is n, and given (an upper bound on) ℓ, the size of the universe, the rest of the parameters are set under the following constraints:

1. For the lattice trapdoor generation algorithm of Alwen and Peikert [7], we need $m \geq 5n \log q$.

   Given this constraint on m, the $\mathsf{TrapGen}$ algorithm outputs a basis of (Gram-Schmidt) length at most $m \cdot \sqrt{\log m}$. Using the $\mathsf{SamplePre}$ algorithm, the secret key vectors $\mathbf{e}_j$ are drawn from a discrete Gaussian with standard deviation $\sigma \geq m \cdot \log m$ (using the $\mathsf{SamplePre}$ algorithm), and thus, by Proposition 6, have length at most $\sigma\sqrt{m} \leq m^{1.5} \cdot \log m$ with all but exponentially small probability.

2. We set the noise distribution $\chi = \overline{\Psi}_\alpha^m$, where $\alpha \geq 2\sqrt{m}/q$ in order to apply Regev's reduction (see Lemma 8). A vector $\mathbf{x}$ sampled from this distribution has length $O(\alpha q \sqrt{m}) \leq 2m$ with all but exponentially small probability.

3. For the correctness to hold, we need to satisfy equation 4. Since $D = (\ell!)^2$, we have

$$\begin{aligned} D|x| + \sum_{j\in J} D^2|\mathbf{e}_j^\top \mathbf{x}_j| &\le D \cdot \alpha q\sqrt{m} + \ell \cdot D^2 \cdot (\alpha q\sqrt{m} \cdot m^{1.5}\log m \cdot \sqrt{m}) \\ &\le 4 \cdot m^3 \log m \cdot \ell(\ell!)^4 \le m^3 \log m \cdot 2^{5\ell} \end{aligned}$$

where we used the fact that $(\ell!)^4 \le (\ell)^{4\ell} \le 2^{5\ell}$. Setting $q \ge m^3 \log m \cdot 2^{5\ell}$ ensures correctness.

As for concrete parameters settings under these constraints, given a constant $\epsilon \in (0,1)$, we set:

- The security parameter $n = \ell^{1/\epsilon}$.
- The modulus q to be a prime in the interval $[n^6 2^{5\ell}, 2 \cdot n^6 2^{5\ell}]$.
- $m = n^{1.5} \ge 5n \log q$, satisfying (1) above.

Putting together the last two bullets, we see that $q \ge m^3 \log m \cdot 2^{5\ell}$, satisfying (3) above.

- The noise parameter $\alpha = 2\sqrt{m}/q = 1/(2^{5n^\epsilon} \cdot \mathsf{poly}(n))$.

Combining this with the worst-case to average-case connection (Proposition 8), we get security under the hardness of $2^{O(n^\epsilon)}$-approximating gapSVP or SIVP on n-dimensional lattices using algorithms that run in time $q \cdot \mathsf{poly}(n) = 2^{O(n^\epsilon)}$. With our state of knowledge on lattice algorithms and algorithms for LWE, security holds for $\epsilon < 1/2$.

5 Construction for Identities in a Large Universe

The construction outlined above can only support identities that are binary vectors of length ℓ. We desire to have the identities live in a larger space so that they capture more expressive attributes.

At a high level, we shall combine our small-universe Fuzzy IBE with a compatible standard-model IBE, such as [3, 16, 1], to construct a Fuzzy IBE that can support large-universe identities. In the scheme outlined here, we use the efficient IBE from Agrawal, Boneh, and Boyen [1] to provide large-universe entities. Our identities are now ℓ-vectors of attributes in $\mathbb{Z}_q^n$, while our parameters are linear in ℓ (ℓ depends on n however; see Section 4.3). Our large-universe construction follows:

Fuzzy.Setup$(1^\lambda, 1^\ell)$**:** On input a security parameter λ, and identity size ℓ, do these steps:

1. Select a uniformly random n-vector $\mathbf{u} \stackrel{R}{\leftarrow} \mathbb{Z}_q^n$.
2. For $(i = 1, \ldots, \ell)$
   (a) Use algorithm $\mathsf{TrapGen}(q, n)$ to select a uniformly random $n \times m$-matrix $\mathbf{A}_{0,i} \in \mathbb{Z}_q^{n\times m}$ with a basis $\mathbf{T}_{\mathbf{A}_{0,i}}$ for $\Lambda_q^\perp(\mathbf{A}_{0,i})$ such that $\|\widetilde{T_{\mathbf{A}_{0,i}}}\| \le O(\sqrt{n \log q})$
   (b) Select two uniformly random $n \times m$ matrices $\mathbf{A}_{1,i}$ and $\mathbf{B}_i$ in $\mathbb{Z}_q^{n\times m}$.
3. Output the public parameters and master key,

$$\mathsf{PP} = \Big(\ \{\mathbf{A}_{0,i}, \mathbf{A}_{1,i}, \mathbf{B}_i\}_{i\in[\ell]}, \qquad \mathbf{u}\ \Big) \qquad ; \qquad \mathsf{MK} = \Big(\ \{T_{\mathbf{A}_{0,i}}\}_{i\in[\ell]}\ \Big)$$

Fuzzy.Extract$(\mathsf{PP}, \mathsf{MK}, \mathsf{id}, k)$**:** On input public parameters PP, a master key MK, an attribute vector or identity $\mathsf{id} = (\mathsf{id}_1, \mathsf{id}_2, \ldots, \mathsf{id}_\ell)$ where $\mathsf{id}_i \in \mathbb{Z}_q^n$ for each $i \in [\ell]$, and a threshold $k \leq \ell$, do:

1. Construct ℓ shares of $\mathbf{u} = (u_1, ..., u_n) \in \mathbb{Z}_q^n$ using a Shamir secret-sharing scheme applied to each co-ordinate of $\mathbf{u}$ independently. Namely, for each $i \in [n]$, choose a uniformly random polynomial $p_i \in \mathbb{Z}_q[x]$ of degree $k-1$ such that $p_i(0) = u_j$.
2. Construct the j^{th} share vector

$$\hat{\mathbf{u}}_j = (\hat{u}_{j,1}, \ldots, \hat{u}_{j,n}) \stackrel{\text{def}}{=} (p_1(j), p_2(j), \ldots, p_n(j)) \in \mathbb{Z}_q^n$$

   Note that by the linearity of the Shamir secret-sharing scheme, there are co-efficients $L_j \in \mathbb{Z}_q$ such that $\mathbf{u} = \sum_{j=1}^{\ell} L_j \cdot \hat{\mathbf{u}}_j$. In fact, linear reconstruction is possible whenever there are k or more shares available.
3. For $i = 1, \ldots, \ell$, do:
   (a) For id_i, construct the *encryption* matrix $\mathbf{F}_{\mathsf{id}_i} = [\mathbf{A}_{0,i} | \mathbf{A}_{1,i} + \mathbf{H}(\mathsf{id}_i)\mathbf{B}_i]$ as in [1]. Here, $\mathbf{H}$ is some fixed Full-Rank Difference (FRD) map, s.t., for any $\mathsf{id}_1 \neq \mathsf{id}_2$ in some exponential-size domain, $\mathbf{H}(\mathsf{id}_1) - \mathbf{H}(\mathsf{id}_2)$ is a full-rank matrix.
   (b) Sample $\mathbf{e}_i \in \mathbb{Z}^{2m}$ as $\mathbf{e}_i \leftarrow \mathsf{SampleLeft}(\mathbf{A}_{0,i},\ \ \mathbf{A}_{1,i} + \mathbf{H}(\mathsf{id}_i)\,\mathbf{B}_i,\ \ \mathbf{T}_{\mathbf{A}_{0,i}},\ \ \hat{\mathbf{u}},\ \ \sigma)$
4. Output the secret key $\mathsf{SK}_{\mathsf{id}} = (\mathbf{e}_1, \ldots, \mathbf{e}_\ell)$.

Fuzzy.Enc$(\mathsf{PP}, \mathsf{id}, b)$**:** On input PP, identity $\mathsf{id} = (\mathsf{id}_1, \mathsf{id}_2, \ldots, \mathsf{id}_\ell) \in (\mathbb{Z}_q^n)^\ell$, and message $b \in \{0, 1\}$:

1. Let $D = (\ell!)^2$.
2. Choose a uniformly random $\mathbf{s} \stackrel{R}{\leftarrow} \mathbb{Z}_q^n$.
3. For $(i = 1, \ldots, \ell)$, do:
   (a) Construct the *encryption* matrix $\mathbf{F}_{\mathsf{id}_i} = [\mathbf{A}_{0,i} | \mathbf{A}_{1,i} + \mathbf{H}(\mathsf{id}_i)\mathbf{B}_i] \in \mathbb{Z}_q^{n \times m}$ as above.
   (b) Choose a uniformly random $m \times m$ matrix $R \stackrel{R}{\leftarrow} \{-1, 1\}^{m \times m}$.
   (c) Choose noise vector $y \stackrel{\bar{\Psi}_\alpha^m}{\longleftarrow} \mathbb{Z}_q^m$, and set $z \leftarrow R^\top y \in \mathbb{Z}_q^m$.
   (d) Set $\mathbf{c}_i \leftarrow \mathbf{F}_{\mathsf{id}_i}{}^\top \mathbf{s} + D \begin{bmatrix} y \\ z \end{bmatrix} \in \mathbb{Z}_q^{2m}$ for all $i \in [\ell]$.
4. Choose a noise term $x \stackrel{\chi_{\{\alpha,q\}}}{\longleftarrow} \mathbb{Z}_q$.
5. Set $\mathbf{c}_0 \leftarrow \mathbf{u}^\top \mathbf{s} + Dx + b \left\lfloor \frac{q}{2} \right\rfloor \in \mathbb{Z}_q$.
6. Output the ciphertext $\mathsf{CT}_{\mathsf{id}} := (c_0, \{\mathbf{c}_i\}_{i \in [\ell]})$.

Fuzzy.Dec$(\mathsf{PP}, \mathsf{SK}_{\mathsf{id}}, \mathsf{CT}_{\mathsf{id}'})$**:** On input parameters PP, a private key $\mathsf{SK}_{\mathsf{id}}$, and a ciphertext $\mathsf{CT}_{\mathsf{id}'}$:

1. Let $J \subset [\ell]$ denote the set of matching elements in id and id'. If $|J| \geq k$ we can compute Lagrange coefficients L_j so that

$$\sum_{j \in J} L_j \hat{\mathbf{u}}_j = \sum_{j \in J} L_j \mathbf{F}_{\mathsf{id}_j}{}^\top \mathbf{e}_j = \mathbf{u}$$

2. Compute $r \leftarrow c_0 - \sum_{j \in J} L_j \cdot \mathbf{e}_j^\top \mathbf{c}_j \pmod q$. View it as the integer $r \in [-\lfloor \frac{q}{2} \rfloor, \lfloor \frac{q}{2} \rfloor) \subset \mathbb{Z}$.
3. If $|r| < \frac{q}{4}$, output 0, else output 1.

Correctness. Correctness of the scheme follows from Section 4.1.1 and the observation that in the large-universe construction, the encryption matrix $\mathbf{F}_{\mathsf{id}_i}$ behaves exactly the same way as the matrix $\mathbf{A}_{i,\mathsf{id}_i}$ in the small universe construction. In particular $\mathbf{F}_{\mathsf{id}_i}\mathbf{e}_i = \hat{\mathbf{u}}_i$, hence we can apply Lagrange interpolation to reconstruct the vector $\mathbf{u}$ and decrypt as before.

Security. Because of the preceding observation, and because the IBE scheme underlying the construction is secure from [1], the proof of security is a straightforward adaptation of the proof in Section 4.2. The details are deferred to the full paper.

6 Connections to Attribute Based Encryption

A natural question that arises from this work is whether the construction can be generalized to Attribute-Based Encryption (ABE) for more expressive access structures. Specifically, we could ask that the secret key for a user be associated with a set of her attributes (e.g., "PhD Student at University X", "Ran in Boston marathon") represented by some vector $\vec{x}$, and the ciphertext be created with respect to an access policy, represented by a (polynomial-size) Boolean circuit C, so that decryption works if and only if $C(\vec{x}) = 1$. (Conversely, we could instead bind the policy C to a user and the attributes $\vec{x}$ to a ciphertext.) In the world of bilinear maps, many constructions are known [23, 29, 9, 17, 25, 26], the most general being for access policies that can be described using *Boolean formulas.*

The difficulty of generalizing our construction to handle arbitrary Boolean formulas is quite subtle. To see this, recall that Fuzzy IBE is a particular type of ABE where the policy is restricted to a single k-out-of-n threshold gate. Since any monotone Boolean formula has an associated linear secret sharing scheme (LSSS), we might imagine generalizing the Fuzzy IBE construction as follows:

1. During $\mathsf{ABE.Setup}$, sample ℓ matrices $\mathbf{A}_1, \ldots, \mathbf{A}_\ell$ with trapdoors.
2. During $\mathsf{ABE.Extract}$, given a formula f, represent it as a LSSS matrix $\mathbf{M}$, share $\mathbf{u}$ according to $\mathbf{M}$ to obtain $\hat{\mathbf{u}}_1, \ldots, \hat{\mathbf{u}}_\ell$ (instead of using Shamir secret sharing). Compute $\mathbf{e}_i, i \in [\ell]$ such that $\mathbf{A}_i\mathbf{e}_i = \hat{\mathbf{u}}_i \bmod q$ and release $\mathbf{e}_1, \ldots \mathbf{e}_\ell$.
3. During $\mathsf{ABE.Enc}$: Say γ is a binary vector representing attributes. Then let $\mathbf{c}_i = \mathbf{A}_i^\top \mathbf{s} + \mathbf{x}$ for i *s.t.* $\gamma_i = 1$. Let $c_0 = \mathbf{u}^\top \mathbf{s} + y + b\lceil \frac{q}{2} \rceil$ as before ($\mathbf{x}, y$ is Gaussian noise and b is the bit being encrypted).
4. During $\mathsf{ABE.Dec}$, if attributes γ satisfy f, we can find low norm coefficients ρ_i so that $\rho_i\hat{\mathbf{u}}_i = \mathbf{u}$ and decrypt by computing $c_0 - \sum_i \rho_i \mathbf{e}_i^\top \mathbf{c}_i$ as before.

The problem with this scheme is that the shares $\hat{\mathbf{u}}_i, \hat{\mathbf{u}}_j$ may be correlated; for, e.g. it is possible to get $\mathbf{u}_1 = \mathbf{u}_2$ for queries such as $(x_1 \vee x_2) \wedge x_3$ and $(x_1 \vee x_2) \wedge x_5$, etc. Then, their preimages $\mathbf{e}_1$ and $\mathbf{e}_2$ can be combined to form a short vector in the null-space of $[\mathbf{A}_1|\mathbf{A}_2]$. Over several such queries, the attacker can then construct a full basis for $\Lambda^\perp([\mathbf{A}_1|\mathbf{A}_2])$, that can be used to break the challenge ciphertext for a target attribute vector such as $1100\ldots00$.

This problem does not arise in our Fuzzy IBE approach since we enforce the policy using secret sharing based on Reed Solomon (RS) codes. RS codes have the property that given k shares, either the shares are sufficient to reconstruct the vector $\mathbf{u}$, or they look jointly uniformly random. This property is crucial in the Fuzzy IBE simulation, and is not satisfied by the ABE generalization outlined above. Thus, we suspect that new techniques will be required to construct Attribute-Based Encryption from lattices.

7 Conclusion

We constructed a Fuzzy Identity-Based Encryption scheme, selectively secure in the standard model, from the hardness of the standard Learning With Errors problem. Ours is the first realization of attribute-based encryption from lattices, and indeed, the first and only "*post-quantum, beyond-IBE*" cryptosystem known to date. Extending the system by showing full security, or transforming it to support more expressive attributes or predicates, are important open problems.

References

[1] Shweta Agrawal, Dan Boneh, and Xavier Boyen. Efficient lattice (H)IBE in the standard model. In *Advances in Cryptology—EUROCRYPT 2010*, volume 6110 of *LNCS*, pages 553–572. Springer, 2010.

[2] Shweta Agrawal, Dan Boneh, and Xavier Boyen. Lattice basis delegation in fixed dimension and shorter-ciphertext hierarchical IBE. In *Advances in Cryptology—CRYPTO 2010*, volume 6223 of *LNCS*, pages 98–115. Springer, 2010.

[3] Shweta Agrawal and Xavier Boyen. Identity-based encryption from lattices in the standard model. Manuscript, July 2009. `http://www.cs.stanford.edu/~xb/ab09/`.

[4] M. Ajtai. Generating hard instances of lattice problems (extended abstract). In *STOC '96: Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, pages 99–108, New York, NY, USA, 1996. ACM.

[5] Miklos Ajtai. Generating hard instances of the short basis problem. In Jir Wiedermann, Peter van Emde Boas, and Mogens Nielsen, editors, *ICALP*, volume 1644 of *Lecture Notes in Computer Science*, pages 1–9. Springer, 1999.

[6] Miklós Ajtai and Cynthia Dwork. A public-key cryptosystem with worst-case/average-case equivalence. In *STOC*, pages 284–293, 1997.

[7] Joël Alwen and Chris Peikert. Generating shorter bases for hard random lattices. In *STACS*, pages 75–86, 2009.

[8] Rikke Bendlin and Ivan Damgård. Threshold decryption and zero-knowledge proofs for lattice-based cryptosystems. In *TCC*, pages 201–218, 2010.

[9] John Bethencourt, Amit Sahai, and Brent Waters. Ciphertext-policy attribute-based encryption. In *SP '07: Proceedings of the 2007 IEEE Symposium on Security and Privacy*, pages 321–334, Washington, DC, USA, 2007. IEEE Computer Society.

[10] Dan Boneh and Xavier Boyen. Efficient selective-id secure identity-based encryption without random oracles. In *Advances in Cryptology (EUROCRYPT 2004)*, volume 3027 of *LNCS*, pages 223–238. Springer, 2004.

[11] Dan Boneh, Ran Canetti, Shai Halevi, and Jonathan Katz. Chosen-ciphertext security from identity-based encryption. *Siam Journal on Computing*, 36:1301–1328, 2007.

[12] Dan Boneh, Craig Gentry, and Michael Hamburg. Space-efficient identity based encryption without pairings. In *Proceedings of FOCS 2007*, pages 647–657, 2007.

[13] Dan Boneh, Eu-Jin Goh, and Kobbi Nissim. Evaluating 2-DNF formulas on ciphertexts. In *Theory of Cryptography*, pages 325–341, 2005.

[14] Dan Boneh, Amit Sahai, and Brent Waters. Functional encryption: Definitions and challenges. In *TCC*, 2011.

[15] Zvika Brakerski and Vinod Vaikuntanathan. Fully homomorphic encryption from ring-LWE and security for key dependent messages. To Appear in CRYPTO 2011, 2011.

[16] David Cash, Dennis Hofheinz, Eike Kiltz, and Chris Peikert. Bonsai trees or, how to delegate a lattice basis. In *Proceedings of Eurocrypt, 2010*, 2010. `http://eprint.iacr.org/`.

[17] Ling Cheung and Calvin C. Newport. Provably secure ciphertext policy ABE. In *ACM Conference on Computer and Communications Security*, pages 456–465, 2007.

[18] Clifford Cocks. An identity based encryption scheme based on quadratic residues. In *Proceedings of the 8th IMA International Conference on Cryptography and Coding*, pages 26–8, 2001.

[19] Ronald Cramer, Ivan Damgård, and Yuval Ishai. Share conversion, pseudorandom secret-sharing and applications to secure computation. In *TCC*, pages 342–362, 2005.

[20] Craig Gentry. Fully homomorphic encryption using ideal lattices. In *STOC*, pages 169–178, 2009.

[21] Craig Gentry. Toward basing fully homomorphic encryption on worst-case hardness. In *CRYPTO*, pages 116–137, 2010.

[22] Craig Gentry, Chris Peikert, and Vinod Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In Richard E. Ladner and Cynthia Dwork, editors, *STOC*, pages 197–206. ACM, 2008.

[23] Vipul Goyal, Omkant Pandey, Amit Sahai, and Brent Waters. Attribute-based encryption for fine-grained access control of encrypted data. In *CCS '06: Proceedings of the 13th ACM conference on Computer and communications security*, pages 89–98, New York, NY, USA, 2006. ACM.

[24] Jonathan Katz, Amit Sahai, and Brent Waters. Predicate encryption supporting disjunctions, polynomial equations, and inner products. In *EUROCRYPT'08: Proceedings of the theory and applications of cryptographic techniques 27th annual international conference on Advances in cryptology*, pages 146–162, Berlin, Heidelberg, 2008. Springer-Verlag.

[25] Allison B. Lewko, Tatsuaki Okamoto, Amit Sahai, Katsuyuki Takashima, and Brent Waters. Fully secure functional encryption: Attribute-based encryption and (hierarchical) inner product encryption. In *EUROCRYPT*, pages 62–91, 2010.

[26] Allison B. Lewko and Brent Waters. Unbounded HIBE and attribute-based encryption. In *EUROCRYPT*, pages 547–567, 2011.

[27] Daniele Micciancio. Generalized compact knapsacks, cyclic lattices, and efficient one-way functions from worst-case complexity assumptions. In *FOCS*, pages 356–365, 2002.

[28] Daniele Micciancio and Oded Regev. Worst-case to average-case reductions based on gaussian measures. In *FOCS '04: Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science*, pages 372–381, Washington, DC, USA, 2004. IEEE Computer Society.

[29] Rafail Ostrovsky, Amit Sahai, and Brent Waters. Attribute-based encryption with non-monotonic access structures. In *CCS '07: Proceedings of the 14th ACM conference on Computer and communications security*, pages 195–203, New York, NY, USA, 2007. ACM.

[30] C. Peikert. Public-key cryptosystems from the worst-case shortest vector problem. In *STOC '09*, 2009.

[31] Chris Peikert. Public-key cryptosystems from the worst-case shortest vector problem: extended abstract. In *STOC '09: Proceedings of the 41st annual ACM symposium on Theory of computing*, pages 333–342. ACM, 2009.

[32] Oded Regev. New lattice-based cryptographic constructions. *J. ACM*, 51(6):899–942, 2004.

[33] Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. In *STOC '05: Proceedings of the thirty-seventh annual ACM symposium on Theory of computing*, pages 84–93, New York, NY, USA, 2005. ACM.

[34] Amit Sahai and Brent Waters. Fuzzy identity-based encryption. In *EUROCRYPT*, pages 457–473, 2005.

[35] Victor Shoup. Practical threshold signatures. In *EUROCRYPT*, pages 207–220, 2000.

A CCA security

Both our small-universe and the large-universe schemes can be lifted from CPA to CCA security using standard methods [11]. Here we describe the extension for our small universe construction; details for the large universe construction follow directly.

Specifically, we make use of a one-time strongly unforgeable signature scheme S_0 to augment the underlying FuzzyIBE scheme. The $\mathsf{Fuzzy.Setup}$ and $\mathsf{Fuzzy.Extract}$ algorithms remain unchanged.

During $\mathsf{Fuzzy.Enc}$, the encryptor runs $\mathsf{S}_0.\mathsf{KeyGen}$ to obtain a public-secret key pair, which we denote by $(\mathsf{VK}, \mathsf{SK})$. We assume that VK is represented as a binary string. Then, the encryptor picks the identity id he wants to encrypt to, and sets $\mathsf{id}' = (\mathsf{id}|\mathsf{VK})$. Let $\mathsf{CT}_{\mathsf{id}'} \leftarrow \mathsf{Fuzzy.Enc}(\mathsf{PP}, b, \mathsf{id}')$. Next, the encryptor sets $\sigma \leftarrow \mathsf{S}_0.\mathsf{Sign}(\mathsf{CT}_{\mathsf{id}'}, \mathsf{SK})$ and returns the tuple $(\sigma, \mathsf{VK}, \mathsf{CT}_{\mathsf{id}'})$.

During $\mathsf{Fuzzy.Dec}$, the decryptor first checks that $\mathsf{S_0.Verify}(\mathsf{CT}_{\mathsf{id}'}, \sigma, \mathsf{VK}) = \top$, and rejects if not. Next, she uses her secret key $\mathsf{SK}_{\mathsf{id}_1}$ to derive a secret key $\mathsf{SK}_{\mathsf{id}''}$ for the "delegated" identity $\mathsf{id}'' \leftarrow (\mathsf{id}_1|\mathsf{VK})$. Such delegation can be done using the standard technique from [16]. Note that if the Hamming weight $|\mathsf{id} - \mathsf{id}_1| \leq k$, then $|\mathsf{id}' - \mathsf{id}''| \leq k$, and conversely. Hence, if the decryptor is authorized to decrypt in the underlying scheme, she can use her extended key $\mathsf{SK}_{\mathsf{id}''}$ to decrypt in the augmented scheme, and only then. The details are deferred to the full paper.