Circular and KDM Security for Identity-Based Encryption

Jacob Alperin-Sheriff and Chris Peikert*

Georgia Institute of Technology

Abstract. We initiate the study of security for key-dependent messages (KDM), sometimes also known as "circular" or "clique" security, in the setting of identity-based encryption (IBE). Circular/KDM security requires that ciphertexts preserve secrecy even when they encrypt messages that may depend on the secret keys, and arises in natural usage scenarios for IBE.

We construct an IBE system that is circular secure for affine functions of users' secret keys, based on the learning with errors (LWE) problem (and hence on worst-case lattice problems). The scheme is secure in the standard model, under a natural extension of a selective-identity attack. Our three main technical contributions are (1) showing the circular/KDM-security of a "dual"-style LWE public-key cryptosystem, (2) proving the hardness of a version of the "extended LWE" problem due to O'Neill, Peikert and Waters (CRYPTO'11), and (3) building an IBE scheme around the dual-style system using a novel lattice-based "all-but-d" trapdoor function.

1 Introduction

Traditional notions of secure encryption, starting with semantic (or IND-CPA) security [22], assume that the plaintext messages do not depend on the secret decryption key (except perhaps indirectly, via the public key or other ciphertexts). In many settings, this may fail to be the case. One obvious scenario is, of course, careless or improper key management: for example, some disk encryption systems end up encrypting the symmetric secret key itself (or a derivative) and storing it on disk. However, there are also situations in which key-dependent messages are used as an integral part of an cryptosystem. For example, in their anonymous credential system, Camenisch and Lysyanskaya [13] use a cycle of key-dependent messages to discourage users from delegating their secret keys. More recently, Gentry's "bootstrapping" technique for obtaining a fully homomorphic cryptosystem [19] encrypts a secret key under the corresponding public key in order to support unbounded homomorphism; the cryptosystem therefore

* This material is based upon work supported by the National Science Foundation under CAREER Award CCF-1054495 and by the Alfred P. Sloan Foundation. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation or the Sloan Foundation.

M. Fischlin, J. Buchmann, and M. Manulis (Eds.): PKC 2012, LNCS 7293, pp. 334–352, 2012.

needs to be "circular secure." More generally, a system that potentially reveals encryptions of any party's secret key under any user's public key needs to be "clique" or "key-dependent message" (KDM) secure. This notion allows for proving formal symbolic soundness of cryptosystems having complexity-based security proofs [1].

Since Boneh *et al.*'s breakthrough work [9] giving a KDM-secure encryption scheme, in the standard model, from the Decision Diffie-Hellman assumption, a large number of results (mostly positive) have been obtained regarding circular- and KDM-secure encryption [23, 5, 6, 10, 4, 26, 11, 12]. However, all these works have dealt only with the symmetric or public-key settings; in particular, the question of circular/KDM security for *identity-based* cryptography has not yet been considered. Recall that in identity-based encryption [35], any string can serve as a public key, and the corresponding secret keys are generated and administered by a trusted Private Key Generator (PKG).

Circular Security for IBE. In this work we define and construct a circular/KDM-secure identity-based encryption (IBE) scheme. KDM security is well-motivated by some natural usage scenarios for IBE, as we now explain.

Recall that identity-based encryption gives a natural and lightweight solution to revocation, via expiring keys. The lifetime of the cryptosystem is divided into time periods, or "epochs." An identity string consists of a user's "true" identity (e.g., name) concatenated with an epoch; when encrypting, one uses the identity for the current epoch. To support revocation, the PKG gives out a user's secret key only for the current epoch, and only if the user is still authorized to be part of the system. Therefore, a user's privileges can be revoked by simply refusing to give out his secret key in future epochs; in particular, this revocation is transparent to the encrypter.

The above framework makes it necessary for users to periodically get new secret keys from the PKG, confidentially. The most lightweight method, which eliminates the need for the user to prove his identity every time, is simply for the PKG to encrypt the new secret key under the user's identity for the previous epoch. This can be proved secure, assuming the underlying IBE is CPA-secure, *as long as there are no cycles of encrypted keys.* However, if a user deletes or loses an old secret key and wants to decrypt a ciphertext from the corresponding epoch, it would be natural for the authority to provide the old secret key encrypted under the user's identity for the current epoch. But because the current secret key has also been encrypted (perhaps via a chain of encryptions) under the old identity, this may be unsafe unless the IBE is KDM-secure.

1.1 Our Contributions

As already mentioned, in this work we define a form of circular/KDM security for identity-based encryption, and give a standard-model construction based on the learning with errors (LWE) problem, hence on worst-case lattice problems via the reductions of [34, 32].

As in prior positive results on circular security [9, 5, 10], our definition allows the adversary to obtain encrypted "key cliques" for affine functions of the secret keys. More precisely, for any tuple of identities $(id_1, \ldots, id_d)$, the attacker may adaptively query encryptions of $f(sk_{id_i})$ under any of the identities id_j, where f is any affine function over the message space, and each sk_{id_i} is a secret key for identity id_i. Our attack model is in the style of a "selective identity" attack, wherein the adversary must declare the target identities $id_1, \ldots, id_d$ (but not the functions f) before seeing the public parameters of the scheme. While this is not the strongest security notion we might hope for, it appears to at least capture the main security requirements of the scenarios described above, because encrypted key cycles are only ever published for the same "real-world" identity at different time epochs. Therefore, just as in a standard selective-identity attack for IBE, the adversary is actually limited to attacking just a single real-world identity, on a set of d epochs (which could, for example, include all valid epochs). We also point out that by a routine hybrid argument, security also holds when attacking a *disjoint* collection of identity cliques (that are named before seeing the public parameters).

Our IBE construction is built from two components, both of which involve some novel techniques. First, we give an LWE-based *public-key* cryptosystem that is clique secure (even for an *unbounded* number of users) for affine functions, and is suitable for embedding into an IBE like the one of [20]. Second, we construct a lattice-based "all-but-d" trapdoor function that serves as the main foundation of the IBE. We elaborate on these two contributions next.

Clique-Secure Public-Key Cryptosystem. We first recall that Applebaum *et al.* [5] showed that a variant of Regev's so-called "primal" LWE cryptosystem [34] is clique secure for affine functions. Unfortunately, this primal-type system does not seem suitable as the foundation for identity-based encryption using the tools of [20]. The first reason is that the proof of clique security from [5] needs the users' public keys to be completely independent, rather than incorporating a shared random string (e.g., the public parameters in an IBE system). The second reason is a bit more technical, and is already described in [20]: in primal-style systems, the user-specific public keys are exponentially sparse pseudorandom values (with unique secret keys), and it is difficult to design an appropriate mapping from identities to valid public keys that actually admit usable underlying secret keys.

Therefore, we first need to obtain clique security for a so-called "dual"-type cryptosystem (using the terminology from [20]), in which *every* syntactically valid public key has a functional underlying secret key (actually, many such secret keys) that can be extracted by the PKG. It turns out that achieving this goal is quite a bit more technically challenging than it was for the "primal"-style schemes. This is primarily because the KDM-secure scheme from [5] (like the earlier one from [9]) has the nice property that given the public key alone, one can efficiently generate *statistically well-distributed* encryptions of the secret key (without knowing the corresponding encryption randomness). This immediately implies circular security for "self-loops," and clique security follows from a couple of other related techniques.

Unfortunately, this nice statistical property on ciphertexts does not seem attainable for dual-style LWE encryption, because now valid ciphertexts are exponentially sparse and hard to generate without knowing the underlying encryption randomness. In addition, because the adversary may obtain an *unbounded* number of key-dependent ciphertexts, we also cannot rely on any statistical entropy of the secret key conditioned on the public key, as is common in the security proofs of most dual-style cryptosystems.

We resolve the above issues by relying on computational assumptions twice in our security proof, first when changing the way that challenge ciphertexts are produced (i.e., by using knowledge of the secret key), and then again when changing the form of the public key. Notably, unlike prior works (e.g., [17, 31]) in which ciphertexts in intermediate games are created by "encrypting with an (information theoretically revealed) secret key," we are able to avoid the use of super-polynomially large noise to "overwhelm" the slight statistical difference between the two ways of generating ciphertexts. This lets us prove security under fully polynomial lattice/LWE assumptions, i.e., those involving a polynomially bounded modulus q and inverse error rate for the LWE problem, and therefore polynomial approximation factors for worst-case lattice problems. We do so by proving the hardness of a version of the *extended*-LWE problem, as defined and left open by the recent work of [31]. We believe that this result should be useful in several other contexts as well.

All-but-d trapdoor functions. We use the clique-secure cryptosystem described above as the foundation for a clique-secure IBE. To make the cryptosystem identity-based, as in [20] we need to embed a "strong" trapdoor into the public parameters so that the PKG can extract a secret key for any identity. Here we use the ideas behind the tag-based algebraic construction of [2], and follow the somewhat simpler and more efficient realization recently due to [28]. We remark that these trapdoor constructions are well-suited to security definitions in which the adversary attacks a *single* tag, because the trapdoor can be "punctured" (i.e., made useless for extracting secret keys, and useful for embedding an LWE challenge) at exactly one predetermined tag. Unfortunately, this does not appear to be sufficient for our purposes, because in the clique security game, the adversary is attacking d identities at once and can obtain challenge ciphertexts under all of them.

To resolve the insufficiency of a single puncture, we extend the trapdoor constructions of [2, 28] so that it is possible to puncture the trapdoor at up to d arbitrary, prespecified tags. To accomplish this, we show how to statistically hide in the public key a degree-d polynomial $f(\cdot)$ over a certain ring $\mathcal{R}$, so that $f(id_i) = 0$ for all the targeted tags (identities) id_i, while $f(id)$ is a unit in $\mathcal{R}$ (i.e., is invertible) for all other identities. The d components of the public key can be combined so as to homomorphically evaluate f on any desired tag. The underlying trapdoor is punctured exactly on tags id where $f(id) = 0$, and is effective for inversion whenever $f(id)$ is a unit in $\mathcal{R}$. Our construction is analogous to the one of [15] in the setting of prime-order groups with bilinear pairings (where arithmetic "in the exponent" happens in a field), and the all-but-d lossy

trapdoor functions of [24]. However, since lattice-based constructions do not work with fields or rings like $\mathbb{Z}_N$, we instead use techniques from the literature on secret sharing over groups and modules, e.g., [16, 18].

We remark that, for technical reasons relating to the number of "hints" for which we can prove the hardness of the extended-LWE problem, we have not been able to prove the KDM-security of our IBE under fully polynomial assumptions (as we were able to do for our basic public-key cryptosystem). We instead rely on the conjectured hardness of LWE for a slightly super-polynomial modulus q and inverse error rate $1/\alpha$, which translates via known reductions [34, 32] to the conjectured hardness of worst-case lattice problems for slightly super-polynomial approximation factors, against slightly super-polynomial-time algorithms. Known lattice algorithms are very far from disproving such conjectures.

2 Preliminaries

We denote the real numbers by $\mathbb{R}$ and the integers by $\mathbb{Z}$. For a positive integer d, we use $[d]$ to denote the set $\{1, \ldots, d\}$. We denote vectors over $\mathbb{R}$ and $\mathbb{Z}$ with lower-case bold letters (e.g. $\mathbf{x}$), and matrices by upper-case bold letters (e.g. $\mathbf{A}$). We say that a function is *negligible*, written $\mathrm{negl}(n)$, if it vanishes faster than the inverse of any polynomial in n. The *statistical distance* between two distributions X, Y over a finite or countable set D is $\Delta(X, Y) = \frac{1}{2}\sum_{w \in D} |X(w) - Y(w)|$. Statistical distance is a metric, and in particular obeys the triangle inequality. Let $\{X_n\}$ and $\{Y_n\}$ be ensembles of random variables indexed by the security parameter n. We say that X and Y are *statistically close* if $\Delta(X_n, Y_n) = \mathrm{negl}(n)$. For a matrix $\mathbf{X} \in \mathbb{R}^{n \times k}$, the *largest singular value* (also known as the *spectral norm*) of $\mathbf{X}$ is defined as $s_1(\mathbf{X}) = \max_{\|\mathbf{u}\|=1} \|\mathbf{X}\mathbf{u}\|$.

2.1 Lattices and Gaussians

A (full-rank) m-dimensional *integer lattice* Λ is an additive subgroup of $\mathbb{Z}^m$ with finite index. This work is concerned with the family of integer lattices whose cryptographic importance was first demonstrated by Ajtai [3]. For integers $n \geq 1$, modulus $q \geq 2$, an m-dimensional lattice from this family is specified by an "arity check" matrix $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$:

$$\Lambda^{\perp}(\mathbf{A}) = \{\mathbf{x} \in \mathbb{Z}^m : \mathbf{A}\mathbf{x} = \mathbf{0} \in \mathbb{Z}_q^n\} \subseteq \mathbb{Z}^m.$$

For any $\mathbf{y}$ in the subgroup of $\mathbb{Z}_q^n$ generated by the columns of $\mathbf{A}$, we also define the coset

$$\Lambda_{\mathbf{y}}^{\perp}(\mathbf{A}) = \{\mathbf{x} \in \mathbb{Z}^m : \mathbf{A}\mathbf{x} = \mathbf{y} \bmod q\} = \Lambda^{\perp}(\mathbf{A}) + \bar{\mathbf{x}},$$

where $\bar{\mathbf{x}} \in \mathbb{Z}^m$ is an arbitrary solution to $\mathbf{A}\bar{\mathbf{x}} = \mathbf{y}$.

We briefly recall Gaussian distributions over lattices (for more details see [29, 20]). For $s > 0$ and dimension $m \geq 1$, the Gaussian function $\rho_s : \mathbb{R}^m \to (0, 1]$ is defined as $\rho_s(\mathbf{x}) = \exp(-\pi\|\mathbf{x}\|^2/s^2)$. For a coset $\Lambda + \mathbf{c}$ of a lattice Λ, the *discrete*

Gaussian distribution $D_{\Lambda+\mathbf{c},s}$ (centered at zero) assigns probability proportional to $\rho_s(\mathbf{x})$ to each vector in the coset, and probability zero elsewhere.

We will need a few standard concepts and facts about discrete Gaussians over lattices. First, for $\epsilon > 0$ the *smoothing parameter* [29] $\eta_\epsilon(\Lambda)$ of an n-dimensional lattice is a positive real value. We will not need its precise definition, which depends on the notion of the *dual* lattice, but only recall the few relevant facts that we need; for details, see, e.g., [29, 20, 28].

Lemma 1. *Let $m \geq Cn \lg q$ for some constant $C > 1$.*

1. *For any $\omega(\sqrt{\log n})$ function, we have $\eta_\epsilon(\mathbb{Z}^n) \leq \omega(\sqrt{\log n})$ for some negligible $\epsilon(n) = \mathrm{negl}(n)$.*
2. *With all but $\mathrm{negl}(n)$ probability over the uniformly random choice of $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$, the following holds: For $\mathbf{e} \leftarrow D_{\mathbb{Z}^m,r}$ where $r = \omega(\sqrt{\log n})$, the distribution of $\mathbf{y} = \mathbf{Ae} \bmod q$ is within $\mathrm{negl}(n)$ statistical distance of uniform, and the conditional distribution of $\mathbf{e}$ given $\mathbf{y}$ is $D_{\Lambda_\mathbf{y}^\perp(\mathbf{A}),r}$.*
3. *For any m-dimensional lattice Λ, any $\mathbf{c} \in \mathbb{Z}^m$, and any $r \geq \eta_\epsilon(\Lambda)$ where $\epsilon(n) = \mathrm{negl}(n)$, we have $\|D_{\Lambda+\mathbf{c},r}\| \leq r\sqrt{m}$ with all but $\mathrm{negl}(n)$ probability. In addition, for $\Lambda = \mathbb{Z}$ we have $|D_{\mathbb{Z},r}| \leq r \cdot \omega(\sqrt{\log n})$ except with $\mathrm{negl}(n)$ probability.*
4. *For any $r > 0$, and for $\mathbf{R} \leftarrow D_{\mathbb{Z},r}^{n\times k}$, we have $s_1(\mathbf{R}) \leq r \cdot O(\sqrt{n} + \sqrt{k})$ except with $\mathrm{negl}(n)$ probability.*

Lemma 2. *For any real number $r = \omega(\sqrt{\log n})$ and $c \in \mathbb{Z}$, the statistical distance between $D_{\mathbb{Z},r}$ and $c + D_{\mathbb{Z},r}$ is $O(|c|/r)$.*

2.2 Trapdoors for Lattices

We recall the efficient trapdoor construction and associated sampling algorithm of Micciancio and Peikert [28]. This construction uses a universal public "gadget" matrix $\mathbf{G} \in \mathbb{Z}_q^{n\times w}$ for which there is an efficient discrete Gaussian sampling algorithm for any parameter $r \geq \omega(\sqrt{\log n}) \geq \eta_\epsilon(\Lambda^\perp(\mathbf{G}))$ (for some $\epsilon(n) = \mathrm{negl}(n)$), i.e., an algorithm that, given any $\mathbf{y} \in \mathbb{Z}_q^n$ and r, outputs a sample from $D_{\Lambda_\mathbf{y}^\perp(\mathbf{G}),r}$. For concreteness, as in [28] we take $\mathbf{G} = \mathbf{I}_n \otimes [1, 2, 4, \ldots, 2^{k-1}] \in \mathbb{Z}_q^{n\times nk}$ for $k = \lceil \lg q \rceil$.

Following [28], we say that an integer matrix $\mathbf{R} \in \mathbb{Z}^{(m-n)\times w}$ is a "strong" trapdoor with tag H for $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ if $\mathbf{A}\left[\begin{smallmatrix}\mathbf{R}\\ \mathbf{I}\end{smallmatrix}\right] = H(\mathbf{G})$ for some efficiently computable and invertible linear transformation H over $\mathbb{Z}_q^n$, which is applied column-wise to $\mathbf{G}$. Equivalently, in place of $H(\mathbf{G})$ we may write $\mathbf{H} \cdot \mathbf{G}$ for some invertible matrix $\mathbf{H} \in \mathbb{Z}_q^{n\times n}$, but in our constructions it will be more natural to work with invertible linear transformations, without explicitly referring to the matrices that represent them.

Lemma 3 ([28, Theorem 5.1]). *Let $\mathbf{R}$ be a strong trapdoor for $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$. There is an efficient randomized algorithm that, given $\mathbf{R}$, any $\mathbf{u} \in \mathbb{Z}_q^n$, and any $r \geq s_1(\mathbf{R}) \cdot \omega(\sqrt{\log n}) \geq \eta_\epsilon(\Lambda^\perp(\mathbf{A}))$ (for some $\epsilon(n) = \mathrm{negl}(n)$), samples from a distribution within $\mathrm{negl}(n)$ distance of $D_{\Lambda_\mathbf{u}^\perp(\mathbf{A}),r}$.*

2.3 Learning With Errors

The *learning with errors* (LWE) problem is parameterized by a dimension $n \geq 1$, an integer modulus $q \geq 2$ and an error distribution χ over $\mathbb{Z}$ (or its induced distribution over $\mathbb{Z}_q$). For a vector $\mathbf{s} \in \mathbb{Z}_q^n$, the distribution $A_{\mathbf{s},\chi}$ over $\mathbb{Z}_q^n \times \mathbb{Z}_q$ is sampled by choosing $\mathbf{a} \in \mathbb{Z}_q^n$ uniformly at random and outputting $(\mathbf{a}, \langle \mathbf{a}, \mathbf{s} \rangle + x)$, where $x \leftarrow \chi$.

The search version of LWE is to recover an arbitrary $\mathbf{s}$ given oracle access to $A_{\mathbf{s},\chi}$. The decision version of LWE is to distinguish, with non-negligible advantage, between samples from $A_{\mathbf{s},\chi}$ for uniformly random $\mathbf{s} \in \mathbb{Z}_q^n$ and uniformly random samples from $\mathbb{Z}_q^n \times \mathbb{Z}_q$. There are search-to-decision reductions for LWE for a variety of moduli q and parameter conditions ([34, 32, 5, 27, 28]). Of particular importance to us are the reductions from [5, 28] for $q = p^e$, where p is prime, $e \geq 1$ is an integer, and $\Pr_{\mathbf{x} \leftarrow \chi}[|\mathbf{x}| \geq p/2] = \mathrm{negl}(n)$. The reductions runs in time polynomial in n, p, and e.

For error distribution $\chi = D_{\mathbb{Z},\alpha q}$, where $\alpha q \geq 2\sqrt{n}$, the search version of LWE is at least as hard as *quantumly* approximating certain worst-case problems on n-dimensional lattices to within $\tilde{O}(n/a)$ factors [34]; for certain parameters, a classical reduction is known for a subset of these lattice problems [32]. Note that the original hardness result for search-LWE was for a continuous Gaussian error distribution, but this can be converted to a discrete Gaussian disribution with a suitable randomized rounding method [33].

We will need the transformation of Applebaum *et al.* [5] from the standard decision-LWE problem (where $\mathbf{s}$ is uniform) to one where the secret $\mathbf{s}$ is chosen from the error distribution χ.

Lemma 4 ([5, Lemma 2]). *Let $q = p^e$ be a prime power. There is a deterministic polynomial-time transformation that, for arbitrary $\mathbf{s} \in \mathbb{Z}_q^n$ and error distribution χ, maps $A_{\mathbf{s},\chi}$ to $A_{\bar{\mathbf{x}},\chi}$ where $\bar{\mathbf{x}} \leftarrow \chi^n$, and maps $U(\mathbb{Z}_q^n \times \mathbb{Z}_q)$ to itself. The transformation also produces an invertible square matrix $\bar{\mathbf{A}} \in \mathbb{Z}_q^{n \times n}$ and $\bar{\mathbf{b}} \in \mathbb{Z}_q^n$ that, when mapping $A_{\mathbf{s},\chi}$ to $A_{\bar{\mathbf{x}},\chi}$, satisfy $\bar{\mathbf{x}} = -\bar{\mathbf{A}}^t \mathbf{s} + \bar{\mathbf{b}}$.*

2.4 Key-Dependent Message Security

In defining key-dependent message security for public-key encryption and for identity-based encryption, we adapt the original definitions of Black *et al.* [7]. As in their definitions, the adversary plays a game with a challenger, and is able to make encryption queries for functions from a certain family $\mathcal{F}$ of the users' secret keys. (Technically, $\mathcal{F}$ is a family of sets of functions parameterized by the security parameter n and the number of users d.)

To simplify our security proofs, in our definition the adversary specifies two functions $(f_0, f_1) \in \mathcal{F}$ with each query, and must distinguish between encryptions of f_0 and encryptions of f_1. If $f(k_1, \ldots, k_d) = 0$ is contained in $\mathcal{F}$ (which should be the case if we want KDM security to imply standard semantic security), then this definition is at least as strong as (and is in fact equivalent to) the original.

To define KDM-security for identity-based encryption, we extend the standard definition of selective security for IBE from [14, 8]. Note that we add a parameter

d to the Setup algorithm denoting the maximum number of users in a clique (i.e., a set of users such that the secret key for any user in the clique may be safely encrypted under the identity for any user in the clique). An adversary plays a game with a challenger that answer encryption queries for functions of the secret keys for identities from a list $\mathcal{I}$, encrypted under identities from $\mathcal{I}$. For selective security, $\mathcal{I}$ must be specified before the adversary sees the public key and remains static throughout the game. In addition to (key-dependent) encryption queries, the adversary is also allowed to make extraction queries for any identity $id \notin \mathcal{I}$.

For an identity-based encryption scheme $(\mathsf{Setup}, \mathsf{Ext}, \mathsf{Enc}, \mathsf{Dec})$, the security game between an adversary and a challenger is parameterized by some $\beta \in \{0,1\}$ and proceeds as follows.

1. $\mathcal{A}(1^n, d)$ outputs a list of (distinct) target identities $\mathcal{I} = (id_1, id_2, \ldots id_\ell)$ for some $\ell \leq d$.
2. The challenger runs $(mpk, msk) \leftarrow \mathsf{Setup}(1^n, d)$. The adversary is given mpk. The challenger then extracts secret keys for each of the target identities, running $sk_i \leftarrow \mathsf{Ext}_{msk}(id_i)$ for each $i \in [\ell]$.
3. $\mathcal{A}$ then can make extraction and encryption queries, in the order of its choice.
   Extraction Queries: $\mathcal{A}$ can query $\mathsf{Ext}_{msk}(\cdot)$ for any identity $id \notin \mathcal{I}$
   Encryption Queries: $\mathcal{A}$ can make encryption queries of the form (f_0, f_1, i), where $f_0, f_1 \in \mathcal{F}$ and $1 \leq i \leq \ell$. The challenger computes $m \leftarrow f_\beta(sk_1, \ldots, sk_\ell)$ and $c \leftarrow \mathsf{Enc}(id_i, m)$, and returns c to $\mathcal{A}$.

We say that the scheme is selective-identity KDM-CPA secure with respect to $\mathcal{F}$ if the games for $\beta = 0, 1$ are computationally indistinguishable.

We define KDM-CPA security for a public-key scheme $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ in a similar manner. Starting at phase two above (since there are no identities to target), the challenger now runs Gen d times, and gives $pk_1, \ldots, pk_d$ to the adversary. In phase three, the adversary can only make encryption queries (since there are no identities to extract), and requests encryptions under public keys instead of identities. Everything else is exactly the same.

3 Hardness of Extended LWE

In this section we describe the *extended*-LWE problem (as originally defined in [31]), and give a reduction to it from the standard LWE problem (with polynomially bounded parameters), thus establishing its hardness under a mild assumption.

3.1 Background and the Problem

O'Neill, Peikert and Waters [31] introduced the extended-LWE problem as a simplifying tool for certain security proofs in which LWE is used in a "hash proof-like" fashion, and additional information about the secret key is revealed to the attacker. In prior works, dealing with such situations often involved adding some "overwhelming" (super-polynomial) extra noise in order to disguise a small

but noticeable statistical difference between, e.g., creating a ciphertext honestly according to an encryption algorithm, and creating one by combining the secret key with a challenge LWE instance. Unfortunately, the use of such overwhelming noise requires an underlying LWE problem with super-polynomial modulus q and inverse error rate $1/\alpha$, which corresponds to a substantially stronger assumption than is needed in the security proofs for many other cryptosystems.

Here we recall the formal definition of the extended-LWE problem. In addition to the usual n, q, and χ parameters for LWE, we also have a number $m = \text{poly}(n)$ of LWE samples, an efficiently sampleable "hint" distribution τ over $\mathbb{Z}^m$ (often, a discrete Gaussian $D^m_{\mathbb{Z},r}$ for some $r \geq 1$) and another Gaussian parameter $\beta > 0$. The problem is to distinguish, with non-negligible advantage, between the two experiments described next; the extended-LWE assumption is that this distinguishing problem is hard. In the ExptLWE experiment, the challenger chooses $\mathbf{A} \leftarrow \mathbb{Z}_q^{n \times m}$, a secret $\mathbf{s} \leftarrow \mathbb{Z}_q^n$ and error vector $\mathbf{x} \leftarrow \chi^m$ defining $\mathbf{b}^t = \mathbf{s}^t\mathbf{A} + \mathbf{x}^t$, along with a "hint" vector $\mathbf{z} \leftarrow \tau$ and error term $\tilde{x} \leftarrow D_{\mathbb{Z},\beta q}$, and outputs

$$(\mathbf{A}, \mathbf{b}, \mathbf{z}, b' = \langle \mathbf{x}, \mathbf{z} \rangle + \tilde{x}).$$

The first two components are just m LWE samples, while the latter two components may be seen as a hint about the error vector $\mathbf{x} \in \mathbb{Z}^m$ in the form of a (noisy) inner product with a vector $\mathbf{z} \in \mathbb{Z}^m$, which is not reduced modulo anything. The ExptUnif experiment is the same, except that $\mathbf{b}$ is defined to be uniformly random and independent of everything else.

Notice that because $\mathbf{A}$ and $\mathbf{z}$ are public, one can "amortize" the extended-LWE problem by including any $\text{poly}(n)$ number of vectors $\mathbf{b}_i^t = \mathbf{s}_i^t\mathbf{A} + \mathbf{x}_i^t$ and hints $b'_i = \langle \mathbf{x}_i, \mathbf{z} \rangle$, for independent $\mathbf{s}_i, \mathbf{x}_i$ (and the same $\mathbf{A}, \mathbf{z}$). By a routine hybrid argument, the two forms of the problem are equivalent, up to a $\text{poly}(n)$ factor in the distinguishing advantage. We use the amortized form of the problem in our security proof in Section 4.

As observed in [31] (and implicitly in prior works like [21, 17]), there is a straightforward reduction from LWE with $\chi = D_{\mathbb{Z},\alpha q}$ to extended-LWE where τ is any m-fold product distribution with variance r^2, if the ratio $\beta/(r \cdot \alpha)$ is superpolynomial in n. In fact, in this case we can securely give out an *unbounded* polynomial number of hints $\mathbf{z}_i, b'_i = \langle \mathbf{x}, \mathbf{z}_i \rangle + \tilde{x}_i$ about the error $\mathbf{x}$. This is simply because by Lemma 2, the terms $\tilde{x} \leftarrow D_{\mathbb{Z},\beta q}$ statistically hide the inner product $\langle \mathbf{x}, \mathbf{z} \rangle$, since the latter has magnitude $\approx r\|\mathbf{x}\| \leq r\alpha q\sqrt{m} = \beta q \cdot \text{negl}(n)$. As a result, the reduction can just simulate the hints $(\mathbf{z}, \langle \mathbf{x}, \mathbf{z} \rangle + \tilde{x})$ on its own. The disadvantage of this approach is that in order to be useful, the modulus q and inverse error rate $1/\alpha$ typically must be super-polynomially large in n, which corresponds to assuming the worst-case hardness of various lattice problems for super-polynomial approximation factors and running times.

We also point out that in the above setting, if the ratio $\beta q/r$ is polynomial in n and a sufficiently large $h = \text{poly}(n)$ number of hints are given out, then extended-LWE is *easy* to solve. To see this, view the h hints as $(\mathbf{Z} \in \mathbb{Z}^{m \times h}, \mathbf{y}^t := \mathbf{x}^t\mathbf{Z} + \tilde{\mathbf{x}}^t)$. With overwhelming probability, the singular values of $\mathbf{Z}$ will all be $r \cdot \Omega(\sqrt{h} - C\sqrt{n+m})$ for some universal constant $C > 0$ (see, e.g., [36, Theorem

5.39]). Thus, for sufficiently large $h = \text{poly}(n)$, with overwhelming probability the singular values of the right-inverse $\mathbf{Z}^+ \in \mathbb{R}^{h \times m}$ of $\mathbf{Z}$ will all be small enough so that $\lfloor \tilde{\mathbf{x}}^t \cdot \mathbf{Z}^+ \rceil = \mathbf{0}$. As a result, we can compute $\lfloor \mathbf{y}^t \mathbf{Z}^+ \rceil = \mathbf{x}^t$, which trivially allows for solving the extended-LWE problem.

In the full version, we contrast our results for extended-LWE with syntactically similar (but qualitatively different) results, such as the Goldreich-Levin theorem and those of [21, 17].

3.2 Reduction from LWE

Here we give a tight reduction from standard LWE to extended-LWE, which holds for the same parameters $n, q, \chi, m \geq n + \omega(\log n)$ in the two problems, and in which *no noise* is added to the hint $\langle \mathbf{z}, \mathbf{x} \rangle$ (i.e., $\beta = 0$). Our reduction imposes one requirement on the parameters: for $\mathbf{x} \leftarrow \chi^m$ and $\mathbf{z} \leftarrow \tau$, we need it to be the case that $|\langle \mathbf{x}, \mathbf{z} \rangle| < p$ with overwhelming probability, where p is the smallest prime divisor of the modulus q. For example, if $\chi = D_{\mathbb{Z},\alpha q}$ and $\tau = D^m_{\mathbb{Z},r}$, by standard tail inequalities it suffices to have $\alpha q \cdot r\sqrt{m+n} \cdot \omega(\sqrt{\log n}) < p$. In other words, the LWE inverse error rate is $1/\alpha > (q/p) \cdot r\sqrt{m+n}$, which is only polynomial in n when q, r, m are.

Theorem 1. *There exists a probabilistic polynomial-time oracle machine (a simulator) $\mathcal{S}$ such that for any adversary $\mathcal{A}$,*

$$\mathbf{Adv}_{\mathsf{LWE}}(\mathcal{S}^{\mathcal{A}}) \geq \tfrac{1}{2p-1} \cdot \mathbf{Adv}_{\mathsf{ELWE}}(\mathcal{A}) - \text{negl}(n),$$

where the parameters of the LWE *and extended-*LWE *problems satisfy the condition specified above.*

Proof. For the proof it is convenient to use the equivalent "knapsack" form of LWE, which is: given $\mathbf{H} \leftarrow \mathbb{Z}_q^{(m-n)\times m}$ and $\mathbf{c} \in \mathbb{Z}_q^{m-n}$, where $\mathbf{c}$ is either $\mathbf{c} = \mathbf{H}\mathbf{x}$ for $\mathbf{x} \leftarrow \chi^m$, or is uniformly random and independent of $\mathbf{H}$, determine (with non-negl(n) advantage) which is the case. The extended form of the problem also reveals a hint $(\mathbf{z}, \langle \mathbf{x}, \mathbf{z} \rangle + \tilde{x})$, analogously to extended-LWE. The equivalence between LWE and its knapsack form for $m \geq n + \omega(\log n)$, which also applies to their extended versions, has been noticed in several works; a proof appears in [27, Lemmas 4.8 and 4.9].

We construct the reduction $\mathcal{S}$ as follows. It receives an LWE instance (in knapsack form) $\mathbf{H} \in \mathbb{Z}_q^{(m-n)\times m}, \mathbf{c} \in \mathbb{Z}_q^{m-n}$. It samples $\mathbf{z} \leftarrow \tau$, $\mathbf{x}' \leftarrow \chi^m$, and $\mathbf{v} \leftarrow \mathbb{Z}_q^{m-n}$, then lets

$$\mathbf{H}' := \mathbf{H} - \mathbf{v}\mathbf{z}^t \in \mathbb{Z}_q^{(m-n)\times m}, \quad \mathbf{c}' = \mathbf{c} - \mathbf{v} \cdot \langle \mathbf{z}, \mathbf{x}' \rangle \in \mathbb{Z}_q^{m-n}.$$

It sends $(\mathbf{H}', \mathbf{b}', \mathbf{z}, \langle \mathbf{x}', \mathbf{z} \rangle)$ to $\mathcal{A}$ (an adversary for extended-LWE in knapsack form), and outputs what $\mathcal{A}$ outputs.

We now analyze the behavior of $\mathcal{S}$. First consider the case where $\mathbf{H}, \mathbf{c}$ are uniform and independent. Then it is clear that $\mathbf{H}', \mathbf{c}'$ are as well, and both $\mathbf{x}'$

and $\mathbf{z}$ are also chosen exactly as in ExptUnif, so $\mathcal{S}$ perfectly simulates ExptUnif to $\mathcal{A}$.

Now, consider the case where $\mathbf{H}, \mathbf{c}$ are drawn from the knapsack distribution, with $\mathbf{c} = \mathbf{H}\mathbf{x}$ for $\mathbf{x} \leftarrow \chi^m$. In this case, we have that $\mathbf{H}'$ is uniformly random (solely over the choice of $\mathbf{H}$), and

$$\mathbf{c}' = \mathbf{H}\mathbf{x} - \mathbf{v} \cdot \langle \mathbf{z}, \mathbf{x}' \rangle = \mathbf{H}'\mathbf{x} + \mathbf{v} \cdot \langle \mathbf{z}, \mathbf{x} - \mathbf{x}' \rangle.$$

So in the event that $\langle \mathbf{x}', \mathbf{z} \rangle = \langle \mathbf{x}, \mathbf{z} \rangle$, we have $\mathbf{c}' = \mathbf{H}'\mathbf{x}$ and so $\mathcal{S}$ perfectly simulates ExptLWE to $\mathcal{A}$. Whereas if $\langle \mathbf{z}, \mathbf{x} - \mathbf{x}' \rangle$ is a unit modulo q, then for any fixed choice of $\mathbf{H}'$, $\mathbf{z}$, $\mathbf{x}$, and $\mathbf{x}'$ (subject to this condition), we have that $\mathbf{c}'$ is uniformly random over the choice of $\mathbf{v}$ alone. Finally, since $\mathbf{x}$ and $\mathbf{x}'$ are identically distributed, it follows that $\mathcal{S}$ perfectly simulates ExptUnif to $\mathcal{A}$.

It remains to analyze the probabilities that $\langle \mathbf{z}, \mathbf{x} - \mathbf{x}' \rangle$ is zero or a unit (modulo q), respectively. First, by assumption $|\langle \mathbf{z}, \mathbf{x} - \mathbf{x}' \rangle| < p$ with overwhelming probability, so exactly one of the two cases holds; moreover, we have $\langle \mathbf{x}, \mathbf{z} \rangle = \langle \mathbf{x}', \mathbf{z} \rangle$ with probability at least $\frac{1}{2p-1} - \mathrm{negl}(n)$ because $\mathbf{x}$ and $\mathbf{x}'$ are independent. The theorem then follows from a routine calculation.

Normal Form. In our cryptosystems, we need to assume the hardness of extended-LWE in "normal form" (as in [30, 5]), where the secret $\mathbf{s} \leftarrow \chi^n$ is drawn from the *error* distribution, the matrix $\mathbf{A}$ and vector $\mathbf{b}^t$ have $m - n$ columns, and the hint is of the form $\mathbf{z} \leftarrow \tau$, $b' = \langle (\mathbf{s}, \mathbf{x}), \mathbf{z} \rangle \in \mathbb{Z}$. Suppose m is sufficiently large so that a uniformly random matrix from $\mathbb{Z}_q^{n \times m}$ contains an invertible n-by-n submatrix with overwhelming probability. Then the reduction from [30, 5] applies to extended-LWE in this form, with the slight modification that LWE samples in the first phase are never "thrown away" but are instead recycled to the second phase.

4 KDM-CPA Secure Public-Key Scheme

Here we present a "dual"-style LWE cryptosystem that is KDM-CPA secure for affine functions of the secret keys. In fact, by setting the parameters appropriately, the construction and security proof also encompass (a slight variant of) the cryptosystem from [25], which has somewhat smaller keys and ciphertexts than "primal" or "dual" systems. In Section 6 we build a KDM-CPA secure IBE around this system.

The cryptosystem involves a few parameters: a modulus $q = p^2$ for a prime p where the message space is $\mathbb{Z}_p$; integer dimensions n, m relating to the underlying LWE problems; and a Gaussian parameter r for key generation and encryption. To make embedding this scheme into our IBE more natural, Gen includes an additional parameter d, which will be used to specify the size of identity cliques in the IBE scheme, and outputs public keys $\mathbf{A}$ that are md columns wide. In the public-key scheme alone, the value d is unrelated to the number of public keys that the adversary can obtain in an attack (which is unbounded), and we would just fix $d = 1$.

- $\mathsf{Gen}(1^n, d)$: choose $\mathbf{A} \in \mathbb{Z}_q^{n \times md}$, $\mathbf{z}_0 \leftarrow D_{\mathbb{Z},r}^n$, $\mathbf{z}_1 \leftarrow D_{\mathbb{Z},r}^{md}$, and let $\mathbf{y} = \mathbf{z}_0 - \mathbf{A}\mathbf{z}_1 = [\mathbf{I}_n \mid -\mathbf{A}]\mathbf{z} \in \mathbb{Z}_q^n$ where $\mathbf{z} = (\mathbf{z}_0, \mathbf{z}_1) \in \mathbb{Z}^{n+md}$. The public key is $(\mathbf{A}, \mathbf{y})$ and the secret key is $\mathbf{z}_1$.
  (Notice that, unlike the dual-style encryption of [20], but like the scheme of [25], the public key component $\mathbf{y}$ is a *perturbed* value of $-\mathbf{A}\mathbf{z}_1$. This will be important in the proof of KDM security.)
- $\mathsf{Enc}(\mathbf{A}, \mathbf{y}, \mu)$: to encrypt a message $\mu \in \mathbb{Z}_p$, choose $\mathbf{x}_0 \leftarrow D_{\mathbb{Z},r}^n$, $\mathbf{x}_1 \leftarrow D_{\mathbb{Z},r}^{md}$ and $x' \leftarrow D_{\mathbb{Z},r}$. Output the ciphertext $\mathbf{c}^t = \mathbf{x}_0^t[\mathbf{A} \mid \mathbf{y}] + [\mathbf{x}_1^t \mid x'] + [\mathbf{0} \mid p \cdot \mu]$.
- $\mathsf{Dec}(\mathbf{z}_1, \mathbf{c})$: Compute $\mu' = \mathbf{c}^t \left[\begin{smallmatrix} \mathbf{z}_1 \\ 1 \end{smallmatrix}\right] \in \mathbb{Z}_q$. Output the $\mu \in \{0, \ldots, p-1\} = \mathbb{Z}_p$ such that μ' is closest to $(p\mu) \bmod q$.

For the public-key system alone, it suffices to take $m \geq n + \omega(\log n)$ by our use of the extended-LWE assumption and its proof of hardness as in Section 3. When embedding the system into an IBE scheme, however, we will use $m = \Theta(n \log q)$ because we need the public parameters to be statistically close to uniform over the choice of the master secret key. The error parameter r must be small enough (relative to q/p) so that decryption is correct with overwhelming probability, but large enough to satisfy the reductions to LWE from worst-case lattice problems [34, 32]; for the latter purpose, $r \geq 2\sqrt{n}$ suffices. (Note that even if part of the security proof relies on LWE in dimension $> n$, this problem is no easier than LWE in dimension n, and so we can still securely use $r = 2\sqrt{n}$ with the larger dimension.)

Here we give some example bounds. Let $r = 2\sqrt{n}$, let

$$p = r^2\sqrt{n + md} \cdot \omega(\sqrt{\log n}) = n\sqrt{n + md} \cdot \omega(\sqrt{\log n}),$$

and let $q = p^2$. Then decryption is correct except with probability $\mathrm{negl}(n)$: let $(\mathbf{A}, \mathbf{y}, \mathbf{z}) \leftarrow \mathsf{Gen}(1^n, d)$. For a ciphertext $\mathbf{c} \leftarrow \mathsf{Enc}(\mathbf{A}, \mathbf{y}, \mu)$, we have

$$\mathbf{c}^t \left[\begin{smallmatrix} \mathbf{z}_1 \\ 1 \end{smallmatrix}\right] = \mathbf{x}_0^t\mathbf{A}\mathbf{z}_1 + \langle \mathbf{x}_1, \mathbf{z}_1 \rangle + \langle \mathbf{x}_0, \mathbf{y} \rangle + x' + p \cdot \mu = \langle \mathbf{x}_0, \mathbf{z}_0 \rangle + \langle \mathbf{x}_1, \mathbf{z}_1 \rangle + x' + p \cdot \mu \bmod q,$$

so decryption is correct whenever $|\langle \mathbf{x}_0, \mathbf{z}_0 \rangle + \langle \mathbf{x}_1, \mathbf{z}_1 \rangle + x'| < p/2$. By known tail bounds on discrete Gaussians, this bound holds except with probability $\mathrm{negl}(n)$ (over the choice of all the random variables), as required.

A proof of the following appears in the full version.

Theorem 2. *The above cryptosystem is KDM-CPA secure with respect to the set of affine functions over $\mathbb{Z}_p$, under the extended-*LWE *assumption for parameters described above.*

5 All-But-*d* Trapdoor Functions

Here we develop a technique for constructing "all-but-d" (tag-based) trapdoor functions, which, informally, are trapdoor functions for which the trapdoor enables efficient inversion for all but (up to) d tags, which are specified at the time of key generation. This is the main tool we use for embedding our KDM-CPA-secure public-key cryptosystem into an identity-based encryption scheme.

Our construction is a generalization (to higher-degree polynomials) of the main technique from [2]. For simplicity and somewhat better efficiency, we follow the construction of [28], specifically the use of a fixed, public "gadget" matrix $\mathbf{G}$ as described in Section 2.2.

5.1 Algebraic Background

Let $n \geq 1$, $q \geq 2$, and $d = \text{poly}(n)$ be integers. Let $\mathcal{R}$ denote any commutative ring (with efficiently computable operations, including inversion of multiplicative units) such that the additive group $\mathbb{G} = \mathbb{Z}_q^n$ is an $\mathcal{R}$-module, and such that there are at least $d+1$ known elements $U = \{u_0 = 0, u_1, u_2, \ldots\} \subseteq \mathcal{R}$ where $u_i - u_j$ is invertible in $\mathcal{R}$ (i.e., a unit) for every $i \neq j$. In particular, we have an (efficiently computable) scalar multiplication operation $\mathcal{R} \times \mathbb{G} \to \mathbb{G}$. Note that multiplication by $u \in \mathcal{R}$ is an invertible linear transformation on $\mathbb{G}$ exactly when u is invertible (i.e., a unit). We extend scalar multiplication in the natural way to vectors and matrices, i.e., $R^{a \times b} \times \mathbb{G}^{b \times c} \to \mathbb{G}^{a \times c}$. To avoid confusion with vectors and matrices over $\mathbb{Z}_q$, we use $\boldsymbol{u}$ notation for vectors over $\mathcal{R}$, and V notation for matrices over $\mathcal{R}$.

To construct a suitable ring, we use ideas from the literature on secret sharing over groups and modules, e.g., [16, 18]. We use an extension ring $\mathcal{R} = \mathbb{Z}_q[x]/(F(x))$ for any monic, degree-n, irreducible $F(x) = F_0 + F_1 x + \cdots + F_{n-1}x^{n-1} + x^n \in \mathbb{Z}_q[x]$. Scalar multiplication $\mathcal{R} \times \mathbb{G} \to \mathbb{G}$ is defined by identifying each $\mathbf{a} = (a_0, \ldots, a_{n-1})^t \in \mathbb{G}$ with the polynomial $a(x) = a_0 + a_1 x + \cdots + a_{n-1}x^{n-1} \in \mathcal{R}$, multiplying in $\mathcal{R}$, then mapping back to $\mathbb{G}$. In other words, scalar multiplication is defined by the linear transformation $x \cdot (a_0, \ldots, a_{n-1})^t = (0, a_0, \ldots, a_{n-2})^t - a_{n-1}(F_0, F_1, \ldots, F_{n-1})^t$. It is easy to check that with this scalar product, $\mathbb{G}$ is an $\mathcal{R}$-module. In addition, by the Chinese remainder theorem, $r \in \mathcal{R}$ is a unit if and only if it is nonzero (as a polynomial residue) modulo every prime integer divisor p of q. (This is because $\mathbb{Z}_p[x]/(F(x))$ is a field by construction.) Letting p be the smallest such divisor of q, we can define the universe $U = \{u_0 = 0, u_1, u_2, \ldots\} \subseteq \mathcal{R}$ to consist of all the polynomial residues having coefficients in $\{0, \ldots, p-1\}$. Then $|U| = p^n \geq 2^n$ and $u_i - u_j$ is a unit for all $i \neq j$, as desired.

5.2 Basic Construction

As in [28], we fix a universal public "gadget" matrix $\mathbf{G} \in \mathbb{Z}_q^{n \times w}$ for which there is an efficient Gaussian preimage sampling algorithm for parameter $s \geq \omega(\sqrt{\log n})$, i.e., an algorithm that given any $\mathbf{u} \in \mathbb{Z}_q^n$ outputs a sample from $D_{\Lambda_{\mathbf{u}}^{\perp}(\mathbf{G}),s}$. E.g., we can let $\mathbf{G} = \mathbf{I}_n \otimes (1, 2, 4, \ldots, 2^{k-1}) \in \mathbb{Z}_q^{n \times nk}$ for $k = \lceil \lg q \rceil$.

As input, the trapdoor generator takes:

- an integer $d \geq 1$ and a monic degree-d polynomial $f(z) = c_0 + c_1 z + \cdots + z^d \in \mathcal{R}[z]$,
- a (usually uniformly random) matrix $\bar{\mathbf{A}} \in \mathbb{Z}_q^{(nd) \times \bar{m}}$ for some $\bar{m} \geq 1$, which is made up of stacked submatrices $\bar{\mathbf{A}}_i \in \mathbb{Z}_q^{n \times \bar{m}}$ for $i = 0, \ldots, d-1$.

– a "short" secret $\mathbf{R} \in \mathbb{Z}^{\bar{m} \times w}$ chosen at random from an appropriate distribution (typically, a discrete Gaussian) to serve as a trapdoor.

As output it produces a matrix $\mathbf{A} \in \mathbb{Z}_q^{(nd) \times (\bar{m}+w)}$ (which is statistically close to uniform, when the parameters and input $\bar{\mathbf{A}}$ are appropriately chosen). In addition, for each tag $u \in U$ there is an efficiently computable (from $\mathbf{A}$) matrix $\mathbf{A}_u \in \mathbb{Z}_q^{n \times (\bar{m}+w)}$ for which $\mathbf{R}$ *may* be a trapdoor, depending on the value of $f(u) \in \mathcal{R}$.

We write the coefficients of $f(z)$ as a column vector $\boldsymbol{c} = (c_0, c_1, \ldots, c_{d-1})^t \in \mathcal{R}^d$, and define

$$\mathbf{A}'_f := \left[\bar{\mathbf{A}} \; \boldsymbol{c} \otimes \mathbf{G}\right] = \begin{bmatrix} \bar{\mathbf{A}}_0 & c_0 \cdot \mathbf{G} \\ \vdots & \vdots \\ \bar{\mathbf{A}}_{d-1} & c_{d-1} \cdot \mathbf{G} \end{bmatrix} \in \mathbb{Z}_q^{(nd) \times (\bar{m}+w)}.$$

To hide the polynomial f, we output the public key

$$\mathbf{A} := \mathbf{A}'_f \cdot \begin{bmatrix} \mathbf{I} & -\mathbf{R} \\ & \mathbf{I} \end{bmatrix} = \left[\bar{\mathbf{A}} \; (\boldsymbol{c} \otimes \mathbf{G}) - \bar{\mathbf{A}}\mathbf{R}\right].$$

Note that as long as the distribution of $[\bar{\mathbf{A}} \mid -\bar{\mathbf{A}}\mathbf{R}]$ is statistically close to uniform, then so is $\mathbf{A}$ for any f.

The tag space for the trapdoor function is the set $U \subset \mathcal{R}$. For any tag $u \in U$, define the row vector $\boldsymbol{u}^t := (u^0, u^1, \cdots, u^{d-1}) \in \mathcal{R}^d$ (where $0^0 = 1$) and the derived matrix for tag u to be

$$\mathbf{A}_u := \boldsymbol{u}^t \cdot \mathbf{A} + \left[\mathbf{0} \; u^d \cdot \mathbf{G}\right] = \left[\boldsymbol{u}^t \cdot \bar{\mathbf{A}} \; f(u) \cdot \mathbf{G}\right] \cdot \begin{bmatrix} \mathbf{I} & -\mathbf{R} \\ & \mathbf{I} \end{bmatrix}.$$

By the condition in Lemma 3, $\mathbf{R}$ is a (strong) trapdoor for $\mathbf{A}_u$ exactly when $f(u) \in \mathcal{R}$ is a unit, because $\mathbf{A}_u \cdot \left[\begin{smallmatrix} \mathbf{R} \\ \mathbf{I} \end{smallmatrix}\right] = f(u) \cdot \mathbf{G}$ and $f(u)$ represents an invertible linear transformation when it is a unit.

5.3 Puncturing

In our cryptosystems and security proofs we will need to generate (using the above procedure) an all-but-d trapdoor function that is "punctured" at up to d tags. More precisely, we are given as input:

– a set of distinct tags $P = \{u_1, \ldots, u_\ell\} \subseteq U$ for some $\ell \leq d$,
– uniformly random matrices $\mathbf{A}_i^* \in \mathbb{Z}_q^{n \times \bar{m}}$ for $i \in [\ell]$ (which often come from an SIS or LWE challenge),
– a "short" secret $\mathbf{R} \in \mathbb{Z}^{\bar{m} \times w}$ chosen at random from an appropriate distribution (typically, a discrete Gaussian) to serve as a trapdoor,
– optionally, some uniformly random auxiliary matrices $\mathbf{Y}_i^* \in \mathbb{Z}_q^{n \times k}$ for $i \in [\ell]$ and some $k \geq 0$.

As output we produce a public key $\mathbf{A} \in \mathbb{Z}_q^{(nd)\times\bar{m}}$ and auxiliary matrix $\mathbf{Y} \in \mathbb{Z}_q^{(nd)\times k}$ so that:

1. Each $\mathbf{A}_{u_i}$ matches the challenge matrix $\mathbf{A}_i^*$, and $\mathbf{R}$ is only a "weak" trapdoor for $\mathbf{A}_{u_i}$. More precisely,
$$\mathbf{A}_{u_i} = \left[\mathbf{A}_i^* \ \mathbf{0}\right] \cdot \begin{bmatrix} \mathbf{I} & -\mathbf{R} \\ & \mathbf{I} \end{bmatrix}.$$
2. $\mathbf{R}$ is a (strong) trapdoor for $\mathbf{A}_u$ for any *nonzero* $u \in U \setminus P$, i.e., $f(u)$ is a unit.
3. The auxiliary matrix $\mathbf{Y}_{u_i} := \boldsymbol{u_i}^t \cdot \mathbf{Y}$ equals the auxiliary input $\mathbf{Y}_i^*$ for each $u_i \in P$.

We satisfy these criteria by invoking the above trapdoor generator with the following inputs f and $\bar{\mathbf{A}}$:

1. We define the monic degree-d polynomial
$$f(z) = z^{d-\ell} \cdot \prod_{i\in[\ell]} (z - u_i) \in \mathcal{R}[z]$$
and expand to compute its coefficients $c_i \in \mathcal{R}$. Note that $f(u_i) = 0$ for every $u_i \in P$, and $f(u)$ is a unit for any nonzero $u \in U \setminus P$ because $0 \in U$ and $u_i - u_j$ is a unit for every distinct $u_i, u_j \in U$.
2. We define $\bar{\mathbf{A}}$ using interpolation: let $\mathbf{A}^* \in \mathbb{Z}_q^{(n\ell)\times\bar{m}}$ denote the stack of challenge matrices $\mathbf{A}_i^*$, and let $V \in \mathcal{R}^{\ell\times d}$ be the Vandermonde matrix whose rows are the vectors $\boldsymbol{u_i}^t$ defined above. We then let $\bar{\mathbf{A}} \in \mathbb{Z}_q^{(nd)\times\bar{m}}$ be a uniformly random solution to $V \cdot \bar{\mathbf{A}} = \mathbf{A}^*$.
Such a solution exists, and is efficiently computable and uniformly random (over the uniformly random choice of $\mathbf{A}^*$ and the random solution chosen). To see this, extend V to an invertible $d \times d$ Vandermonde matrix over $\mathcal{R}$ having unit determinant $\prod_{i<j}(u_j - u_i) \in \mathcal{R}^*$, by adding $d - \ell$ additional rows $\boldsymbol{u_j}^t$ for arbitrary distinct $u_j \in U \setminus P$. Likewise, extend $\mathbf{A}^*$ to have dimension $(nd) \times \bar{m}$ by adding uniformly random rows. Then for any fixed choice of the (extended) matrix V, the (extended) matrix $\mathbf{A}^*$ and solution $\bar{\mathbf{A}}$ are in bijective correspondence, and so the latter is uniformly random because the former is.
3. We also define the auxiliary matrix $\mathbf{Y}$ similarly using interpolation, as a uniformly random solution to $V \cdot \mathbf{Y} = \mathbf{Y}^*$.

6 Circular-Secure IBE

Our IBE scheme is a generalization of the efficient IBE scheme of Agrawal *et al.* [2]. Other than some minor changes in the parameters, the main difference is the use of the all-but-d trapdoor construction, which allows us to "puncture"

the master public key at up to d identities in the security proof. The scheme has parameters modulus q, message space $\mathbb{Z}_p$ for some $p < q$, dimension m, and Gaussian parameters r and γ. Most of the parameters match those in the public-key encryption scheme of Section 4, with the additional constraint that r must be large enough that we can run the preimage sampling algorithm (Lemma 3) in Ext. Due to space considerations, the conditions on the parameters are described in the full version.

The identity space for the scheme is $U \setminus \{0\} \subset \mathcal{R}$, where U, $\mathcal{R}$ are constructed as in Section 5.

- $\mathsf{Setup}(1^n, d)$: On input security parameter 1^n and secret key clique size d:
  1. Sample $\mathbf{R} \leftarrow D^{md \times w}_{\mathbb{Z},\omega(\sqrt{\log n})}$, and for $i = 0, \ldots, d-1$, choose uniformly random $\mathbf{A}_i \leftarrow \mathbb{Z}_q^{n \times md}$, $\mathbf{y}_i \leftarrow \mathbb{Z}_q^n$ and let $\tilde{\mathbf{A}}_i = -\mathbf{A}_i\mathbf{R} \in \mathbb{Z}_q^{n \times w}$. (Note that this is simply calling the all-but-d trapdoor construction from Section 5 with an empty set of punctured tags.) Let $\mathbf{A}^t := \left[\mathbf{A}_0^t \cdots \mathbf{A}_{d-1}^t\right]$, $\tilde{\mathbf{A}}^t := \left[\tilde{\mathbf{A}}_0^t \cdots \tilde{\mathbf{A}}_{d-1}^t\right]$, $\mathbf{y}^t := \left[\mathbf{y}_0^t \cdots \mathbf{y}_{d-1}^t\right]$. Note that $\tilde{\mathbf{A}} = -\mathbf{A}\mathbf{R}$.
  2. The public key is $mpk = (\mathbf{A}, \tilde{\mathbf{A}}, \mathbf{y})$. The master secret key is $msk = (\mathbf{R})$.
- $\mathsf{Ext}(mpk, msk, u)$ On input mpk, msk and $u \in U \setminus \{0\} \subseteq \mathcal{R}$:
  1. Let $\boldsymbol{u}^t := (u^0, u^1, \ldots, u^{d-1})$, $\bar{\mathbf{A}}_u = \boldsymbol{u}^t \cdot \mathbf{A}$, $\mathbf{y}_u = \boldsymbol{u}^t \cdot \mathbf{y}$ and $\mathbf{A}_u = [\bar{\mathbf{A}}_u \mid u^d\mathbf{G} - \bar{\mathbf{A}}_u\mathbf{R}]$, as in Section 5.
  2. Sample $\mathbf{z}_0 \leftarrow D^n_{\mathbb{Z},r}$, $\mathbf{z}_1 \leftarrow D_{\Lambda^{\perp}_{\mathbf{z}_0 - \mathbf{y}_u}(\mathbf{A}_u),r}$ using the preimage sampling algorithm (Lemma 3), so that $\mathbf{y}_u = \mathbf{z}_0 - \mathbf{A}_u\mathbf{z}_1$ (as in the public-key cryptosystem from Section 4). Output $sk_u := \mathbf{z}_1$.
     Note that the above is possible because $u^d \in \mathcal{R}$ is a unit, and by our choice of r below, because $s_1(\mathbf{R}) = O(\sqrt{md}+\sqrt{w})\cdot\omega(\sqrt{\log n}) = O(\sqrt{md})\cdot\omega(\sqrt{\log n})$ with all but $\mathrm{negl}(n)$ probability by Lemma 1.
- $\mathsf{Enc}(mpk, u, \mu)$: On input master public key, identity $u \in U \setminus \{0\}$, and message $\mu \in \mathbb{Z}_p$ do:
  1. Let $\boldsymbol{u}^t := (u^0, u^1, \ldots, u^{d-1})$, $\mathbf{A}_u = [\boldsymbol{u}^t \cdot \mathbf{A} \mid u^d\mathbf{G} + \boldsymbol{u}^t \cdot \tilde{\mathbf{A}}] \in \mathbb{Z}_q^{n \times md+w}$, and $\mathbf{y}_u = \boldsymbol{u}^t \cdot \mathbf{y}$.
  2. Choose $\mathbf{s} \leftarrow D^n_{\mathbb{Z},r}$, $\mathbf{x}_0 \leftarrow D^{md}_{\mathbb{Z},r}, \mathbf{x}_1 \leftarrow D^w_{\mathbb{Z},\gamma}$, $x_2 \leftarrow D_{\mathbb{Z},r}$. Let $\mathbf{x}^t = [\mathbf{x}_0^t \mid \mathbf{x}_1^t]$.
  3. Output the ciphertext $\mathbf{c}^t = \mathbf{s}^t[\mathbf{A}_u \mid \mathbf{y}_u] + [\mathbf{x}^t \mid x_2] + [\mathbf{0} \mid p \cdot \mu]$.
- $\mathsf{Dec}(mpk, sk_u = \mathbf{z}_1, \mathbf{c})$: output the $\mu \in \mathbb{Z}_p$ such that $\mathbf{c}^t \left[\begin{smallmatrix}\mathbf{z}_1\\ 1\end{smallmatrix}\right]$ is closest to $p \cdot \mu$ modulo q.

Theorem 3. *For the above parameters, the above IBE scheme is selective identity KDM-CPA secure with respect to the set of affine functions over $\mathbb{Z}_p$, under the $\mathsf{LWE}_{q,\chi}$ assumption for $\chi = D_{\mathbb{Z},r}$, and the KDM-CPA security of the system from Section 4.*

Proof (Sketch). Here we give an overview of the proof strategy, deferring the formal proof to the full version. Game 0 is the actual attack game. In Game 1, we use the all-but-d trapdoor construction from Section 5 to generate the master public key, "puncturing" it at the targeted identities. Finally, in Game 2, we play the KDM-CPA security game against a challenger running the public-key encryption scheme from Section 4 and use the outputs of the challenger

to simulate Game 1. This requires some care because the IBE secret keys and ciphertexts have larger dimension by an additive term of w (the width of $\mathbf{G}$). To address this, we fill in the missing dimensions of the secret keys by choosing them ourselves, and use knowledge of the master secret key to fill in the missing dimensions of the ciphertexts (here is where we use the fact that noise with parameter γ "overwhelms" noise with parameter r).

Acknowledgments. We thanks Oded Regev for helpful comments, and for pointing out a subtle error in a prior version of our reduction from Section 3.

References

[1] Adão, P., Bana, G., Herzog, J., Scedrov, A.: Soundness of Formal Encryption in the Presence of Key-Cycles. In: de Capitani di Vimercati, S., Syverson, P.F., Gollmann, D. (eds.) ESORICS 2005. LNCS, vol. 3679, pp. 374–396. Springer, Heidelberg (2005)

[2] Agrawal, S., Boneh, D., Boyen, X.: Efficient Lattice (H)IBE in the Standard Model. In: Gilbert, H. (ed.) EUROCRYPT 2010. LNCS, vol. 6110, pp. 553–572. Springer, Heidelberg (2010)

[3] Ajtai, M.: Generating hard instances of lattice problems. Quaderni di Matematica 13, 1–32 (2004); Preliminary version in STOC 1996

[4] Applebaum, B.: Key-Dependent Message Security: Generic Amplification and Completeness. In: Paterson, K.G. (ed.) EUROCRYPT 2011. LNCS, vol. 6632, pp. 527–546. Springer, Heidelberg (2011)

[5] Applebaum, B., Cash, D., Peikert, C., Sahai, A.: Fast Cryptographic Primitives and Circular-Secure Encryption Based on Hard Learning Problems. In: Halevi, S. (ed.) CRYPTO 2009. LNCS, vol. 5677, pp. 595–618. Springer, Heidelberg (2009)

[6] Barak, B., Haitner, I., Hofheinz, D., Ishai, Y.: Bounded Key-Dependent Message Security. In: Gilbert, H. (ed.) EUROCRYPT 2010. LNCS, vol. 6110, pp. 423–444. Springer, Heidelberg (2010)

[7] Black, J., Rogaway, P., Shrimpton, T.: Encryption-Scheme Security in the Presence of Key-Dependent Messages. In: Nyberg, K., Heys, H.M. (eds.) SAC 2002. LNCS, vol. 2595, pp. 62–75. Springer, Heidelberg (2003)

[8] Boneh, D., Canetti, R., Halevi, S., Katz, J.: Chosen-ciphertext security from identity-based encryption. SIAM J. Comput. 36(5), 1301–1328 (2007)

[9] Boneh, D., Halevi, S., Hamburg, M., Ostrovsky, R.: Circular-Secure Encryption from Decision Diffie-Hellman. In: Wagner, D. (ed.) CRYPTO 2008. LNCS, vol. 5157, pp. 108–125. Springer, Heidelberg (2008)

[10] Brakerski, Z., Goldwasser, S.: Circular and Leakage Resilient Public-Key Encryption Under Subgroup Indistinguishability - (or: Quadratic Residuosity Strikes Back). In: Rabin, T. (ed.) CRYPTO 2010. LNCS, vol. 6223, pp. 1–20. Springer, Heidelberg (2010)

[11] Brakerski, Z., Goldwasser, S., Kalai, Y.T.: Black-Box Circular-Secure Encryption beyond Affine Functions. In: Ishai, Y. (ed.) TCC 2011. LNCS, vol. 6597, pp. 201–218. Springer, Heidelberg (2011)

[12] Brakerski, Z., Vaikuntanathan, V.: Fully Homomorphic Encryption from Ring-LWE and Security for Key Dependent Messages. In: Rogaway, P. (ed.) CRYPTO 2011. LNCS, vol. 6841, pp. 505–524. Springer, Heidelberg (2011)

[13] Camenisch, J., Lysyanskaya, A.: An Efficient System for Non-transferable Anonymous Credentials with Optional Anonymity Revocation. In: Pfitzmann, B. (ed.) EUROCRYPT 2001. LNCS, vol. 2045, pp. 93–118. Springer, Heidelberg (2001)
[14] Canetti, R., Halevi, S., Katz, J.: A Forward-Secure Public-key Encryption Scheme. J. Cryptology 20(3), 265–294 (2007); Preliminary version in EUROCRYPT 2003
[15] Chatterjee, S., Sarkar, P.: Generalization of the Selective-ID Security Model for HIBE Protocols. In: Yung, M., et al. (eds.) PKC 2006. LNCS, vol. 3958, pp. 241–256. Springer, Heidelberg (2006)
[16] Desmedt, Y., Frankel, Y.: Perfect homomorphic zero-knowledge threshold schemes over any finite abelian group. SIAM J. Discrete Math. 7(4), 667–679 (1994)
[17] Dodis, Y., Goldwasser, S., Tauman Kalai, Y., Peikert, C., Vaikuntanathan, V.: Public-Key Encryption Schemes with Auxiliary Inputs. In: Micciancio, D. (ed.) TCC 2010. LNCS, vol. 5978, pp. 361–381. Springer, Heidelberg (2010)
[18] Fehr, S.: Span programs over rings and how to share a secret from a module. Master's thesis, ETH Zurich, Institute for Theoretical Computer Science (1998)
[19] Gentry, C.: Fully homomorphic encryption using ideal lattices. In: STOC, pp. 169–178 (2009)
[20] Gentry, C., Peikert, C., Vaikuntanathan, V.: Trapdoors for hard lattices and new cryptographic constructions. In: STOC, pp. 197–206 (2008)
[21] Goldwasser, S., Kalai, Y.T., Peikert, C., Vaikuntanathan, V.: Robustness of the learning with errors assumption. In: ICS, pp. 230–240 (2010)
[22] Goldwasser, S., Micali, S.: Probabilistic encryption. J. Comput. Syst. Sci. 28(2), 270–299 (1982); Preliminary version in STOC 1982
[23] Haitner, I., Holenstein, T.: On the (Im)Possibility of Key Dependent Encryption. In: Reingold, O. (ed.) TCC 2009. LNCS, vol. 5444, pp. 202–219. Springer, Heidelberg (2009)
[24] Hemenway, B., Libert, B., Ostrovsky, R., Vergnaud, D.: Lossy Encryption: Constructions from General Assumptions and Efficient Selective Opening Chosen Ciphertext Security. In: Lee, D.H. (ed.) ASIACRYPT 2011. LNCS, vol. 7073, pp. 70–88. Springer, Heidelberg (2011)
[25] Lindner, R., Peikert, C.: Better Key Sizes (and Attacks) for LWE-Based Encryption. In: Kiayias, A. (ed.) CT-RSA 2011. LNCS, vol. 6558, pp. 319–339. Springer, Heidelberg (2011)
[26] Malkin, T., Teranishi, I., Yung, M.: Efficient Circuit-Size Independent Public Key Encryption with KDM Security. In: Paterson, K.G. (ed.) EUROCRYPT 2011. LNCS, vol. 6632, pp. 507–526. Springer, Heidelberg (2011)
[27] Micciancio, D., Mol, P.: Pseudorandom Knapsacks and the Sample Complexity of LWE Search-to-Decision Reductions. In: Rogaway, P. (ed.) CRYPTO 2011. LNCS, vol. 6841, pp. 465–484. Springer, Heidelberg (2011)
[28] Micciancio, D., Peikert, C.: Trapdoors for Lattices: Simpler, Tighter, Faster, Smaller. In: Pointcheval, D., Johansson, T. (eds.) EUROCRYPT 2012. LNCS, vol. 7237, pp. 700–718. Springer, Heidelberg (2012)
[29] Micciancio, D., Regev, O.: Worst-case to average-case reductions based on Gaussian measures. SIAM J. Comput. 37(1), 267–302 (2004); Preliminary version in FOCS 2004
[30] Micciancio, D., Regev, O.: Lattice-based cryptography. In: Post Quantum Cryptography, pp. 147–191. Springer (February 2009)
[31] O'Neill, A., Peikert, C., Waters, B.: Bi-Deniable Public-Key Encryption. In: Rogaway, P. (ed.) CRYPTO 2011. LNCS, vol. 6841, pp. 525–542. Springer, Heidelberg (2011)

[32] Peikert, C.: Public-key cryptosystems from the worst-case shortest vector problem. In: STOC, pp. 333–342 (2009)
[33] Peikert, C.: An Efficient and Parallel Gaussian Sampler for Lattices. In: Rabin, T. (ed.) CRYPTO 2010. LNCS, vol. 6223, pp. 80–97. Springer, Heidelberg (2010)
[34] Regev, O.: On lattices, learning with errors, random linear codes, and cryptography. J. ACM 56(6), 1–40 (2005); Preliminary version in STOC 2005
[35] Shamir, A.: Identity-Based Cryptosystems and Signature Schemes. In: Blakely, G.R., Chaum, D. (eds.) CRYPTO 1984. LNCS, vol. 196, pp. 47–53. Springer, Heidelberg (1985)
[36] Vershynin, R.: Introduction to the non-asymptotic analysis of random matrices (January 2011), `http://www-personal.umich.edu/~romanv/papers/non-asymptotic-rmt-plain.pdf` (last accessed February 4, 2011)