Fully Secure Functional Encryption for Inner Products, from Standard Assumptions

Benoît Libert and Damien Stehlé

Ecole Normale Supérieure de Lyon
Laboratoire d'Informatique du Parallélisme
46 Allée d'Italie, 69364 Lyon Cedex 07, France

Abstract. Functional encryption is a modern public-key paradigm where a master private key can be used to derive sub-keys SK_F associated with certain functions F in such a way that the decryption operation reveals $F(M)$, if M is the encrypted message, and nothing else. Recently, Abdalla *et al.* gave simple and efficient realizations of the primitive for the computation of linear functions on encrypted data: given an encryption of a vector $\boldsymbol{y} \in \mathbb{Z}_q^\ell$, a private key $SK_{\boldsymbol{x}}$ for the vector $\boldsymbol{x} \in \mathbb{Z}_q^\ell$ allows computing $\langle \boldsymbol{x}, \boldsymbol{y} \rangle$. Their technique surprisingly allows for instantiations under standard assumptions, like the hardness of the Decision Diffie-Hellman (DDH) and Learning-with-Errors (LWE) problems. Their constructions, however, are only proved secure against *selective* adversaries, which have to declare the challenge messages M_0 and M_1 at the outset of the game. In this paper, we provide constructions that provably achieve security against more realistic *adaptive* attacks (where the messages M_0 and M_1 may be chosen in the challenge phase, based on the previously collected information) for the same inner product functionality. Our constructions are obtained from hash proof systems endowed with homomorphic properties over the key space. They are as efficient as those of Abdalla *et al.* and rely on the same assumptions. As a result of independent interest, we prove the security of our LWE-based system via a new result on the hardness of the extended LWE problem, where the distinguisher receives hints about the noise distribution.

Keywords. Functional encryption, adaptive security, standard assumptions, DDH, LWE, extended LWE.

1 Introduction

Functional encryption (FE) [69, 17] is an emerging public-key paradigm, where the master private key msk allows deriving a private sub-key SK_F associated with a function F. When SK_F is used to decrypt a ciphertext C, the key holder only learns $F(X)$, where X is the encrypted data and nothing else. In some cases, the message $X = (\mathsf{ind}, m)$ consists of an index ind (which is essentially a set of descriptive attributes) and a message m, which is sometimes called "payload". The concept finds numerous applications. In cloud computing platforms, users can store encrypted data on a remote server and subsequently provide the server

with a key SK_F which allows it to compute a specific function F of encrypted data. If m is the encrypted message, the server can thus return $F(m)$ without learning anything about the data. One distinguishes FE systems with public index, where ind is publicly revealed by the ciphertext but m is hidden, from those with private index, where ind and m are both hidden. Constructions with public index tend to be more efficient, expressive and easier to obtain than those in the private-index setting.

The usual security requirement is called "collusion-resistance" and it captures the intuition that no collection of private keys for functions $F_1, \ldots, F_q$ should make it possible to decrypt a ciphertext that no individual such key can decrypt. In the area, the holy grail is a technique allowing to construct FE schemes for any polynomial-time-computable function F. Unfortunately, truly efficient realizations have only been found for very restricted classes of functions thus far. Examples include identity-based encryption (IBE) [71, 15], attribute-based encryption (ABE) [69, 45] and predicate encryption (PE) [52]. These primitives enable the fine-grained distribution of sensitive content [69, 45] and privacy-preserving searches over encrypted data [14, 1, 52].

Constructions for General Functionalities. Until recently, all results on functional encryption for general functionalities, such as [42, 41], were limited to only provide bounded collusion-resistance: namely, the adversary was restricted to make an a priori bounded number of queries and, in turn, this upper bound affects the efficiency of the system. Without this restriction, the most expressive functionalities that could be handled were those that can be expressed in terms of shallow (i.e., logarithmic depth) circuits. This was the case of the inner product functionality considered for the first time by Katz, Sahai and Waters [52]. Specifically, ciphertexts and private keys were both associated with a vector of attributes and decryption works whenever these two vectors are orthogonal. This state-of-affairs changed in 2013 with the appearance of attribute-based encryption schemes for arbitrary polynomial-size circuits [34, 43, 16]. However, as in all earlier ABE schemes, their functionality is to reveal the payload message m entirely (rather than a function of it) if and only if the (public) index ind of the ciphertext satisfies the circuit F associated with the key. Gorbunov, Vaikuntanathan and Wee [44] recently extended these results to the private index setting. However, their predicate encryption system still does not provide a construction of functional encryption for general functionalities as the decryption algorithm reveals the entire plaintext when the circuit of the key accepts the hidden attribute set ind of the ciphertext.

Using the machinery of multi-linear maps [32], Garg *et al.* [32] gave a first theoretical solution of general functional encryption [33]. Due to the use of indistinguishability obfuscation [6], their solution is quite far from being practical as it incurs huge ciphertexts and keys, even for very simple circuits. Moreover, it relies on very *ad hoc* assumptions in groups with a multi-linear map, the security of which is not well-understood yet (at least with currently known candidates). Indeed, the two candidates [32, 22] put forth in 2013 were recently found to be insecure [21, 49]. Even if these cryptanalyses do not impact constructions of in-

distinguishability obfuscation, they motivate the search for (efficient) solutions based on well-studied hardness assumptions, even for a smaller range of functions.

Functional Encryption for Inner Product Evaluations. As noticed by Katz, Sahai and Waters [52], inner product relations suffice to express conjunctions and disjunctions of simple atomic conditions as well as CNF/DNF formulas involving a small number of variables. In the private index setting, the inner product encryption (IPE) systems of [52, 53, 60, 3, 61, 62] make it possible to test, e.g., whether the hidden ciphertext attributes ind belong to a specific set encoded in the key, if they are roots of some polynomial or if they satisfy a small CNF/DNF formula. A common feature of all of these works is that the realized functionality is to test whether the index ind of the ciphertext – which is a vector $\boldsymbol{y}$ of attributes – is orthogonal to another vector $\boldsymbol{x}$ hard-coded in the decryption key $SK_{\boldsymbol{x}}$ without revealing any further information. This functionality (intentionally) does not compute the actual inner product value $\langle \boldsymbol{x}, \boldsymbol{y} \rangle$ when the latter is non-zero.

Recently, Abdalla, Bourse, De Caro and Pointcheval [2] considered a slightly different functionality which computes the actual value of the inner product instead of testing its cancellation (this functionality was considered in a different model by an independent work of Naveed *et al.* [59]). When a ciphertext C encrypts a vector $\boldsymbol{y} \in \mathcal{D}^{\ell}$ over some domain $\mathcal{D}$, a private key for the vector $\boldsymbol{x} \in \mathcal{D}^{\ell}$ allows computing $\langle \boldsymbol{x}, \boldsymbol{y} \rangle$ and nothing else about $\boldsymbol{y}$. Abdalla *et al* [2] pointed out that the inner product functionality suffices for the computation of linear functions (e.g., sums or averages) over encrypted data. By encoding ℓ-bit messages $m = m[1] \dots m[\ell]$ as vectors $\boldsymbol{y} = (m[1], \dots, m[\ell])$, inner products also allow for the computation of Hamming weights using private keys $\mathsf{sk}_{\boldsymbol{x}}$ for the all-one vector $\boldsymbol{x} = (1, 1, \dots, 1)$. As mentioned in the earlier work of Katz, Sahai and Waters [52], inner products also enable the evaluation of polynomials over encrypted data. To do this, we can simply encode a message m as a vector $\boldsymbol{y} = (1, m, m^2, \dots, m^d) \in \mathcal{D}^{d+1}$ and a degree-d polynomial $P[X] = \sum_{i=0}^{d} p_i X^i$ is encoded as a vector $\boldsymbol{x} = (p_0, p_1, \dots, p_d) \in \mathcal{D}^{d+1}$ for which the key $SK_{\boldsymbol{x}}$ is generated. Using a similar encoding, we can also evaluate multivariate polynomials of the form $P[X_1, \dots, X_d] = \prod_{i=1}^{d} (X_i - I_i)$ of small degree $d = O(\log \ell)$. A difference with [52] is that the functionality of [2] makes it possible to compute the exact value of the polynomial whereas [52] only tests if the encrypted message is a root of the polynomial or not.

More surprisingly, Abdalla *et al.* showed that [2] this specific functionality allows for very simple and efficient realizations under standard assumptions like the Decision Diffie-Hellman (DDH) – which was unexpected since DDH is not known to easily lend itself to the design of such primitives – and Learning-with-Errors (LWE) assumptions [66]. Their constructions can be seen as instantiations of a general paradigm based on encryption schemes (like Elgamal [28] or Regev [66]) with additive homomorphic properties over the message space *and* the key space. They also leveraged the fact that, as shown in [7], such encryption schemes sometimes make it possible to safely recycle random encryption coins

across different encryptions computed under different keys.

Using the additively homomorphic variant of Elgamal [28], Abdalla *et al* described a very simple FE scheme where the master public key consists of elements $\left(g, \{h_i = g^{s_i}\}_{i=1}^{\ell}\right)$ in a cyclic group $\mathbb{G}$ and vectors $\boldsymbol{y} = (y_1, \ldots, y_\ell) \in \mathbb{Z}_q^\ell$ are encrypted as $(C_0, \{C_i\}_{i=1}^{\ell}) = (g^r, \{g^{y_i} \cdot h_i^r\}_{i=1}^{\ell})$. A private key for the vector $\boldsymbol{x} = (y_1, \ldots, y_\ell) \in \mathbb{Z}_q^\ell$ is obtained as $SK_{\boldsymbol{x}} = \langle \boldsymbol{s}, \boldsymbol{x} \rangle$ and allows computing $g^{\langle \boldsymbol{y}, \boldsymbol{x} \rangle} = (\prod_{i=1}^{\ell} C_i^{x_i})/C_0^{SK_{\boldsymbol{x}}}$, which in turn yields the inner product $\langle \boldsymbol{y}, \boldsymbol{x} \rangle$ as long as the latter lands in a small interval where discrete logarithms are computable in reasonable time. Abdalla *et al* also applied the same design principle to Regev's cryptosystem [66] and obtained a variant based on the LWE assumption.

At first glance, the constructions of [2] seem limited to only provide bounded collusion resistance since, in their DDH-based system for example, an adversary that obtains private keys $SK_{\boldsymbol{x}} = \langle \boldsymbol{s}, \boldsymbol{x} \rangle$ for ℓ independent vectors $\boldsymbol{x}$ can reconstruct $\boldsymbol{s}$. However, as observed in [2], this is limitation is inherent to the functionality: indeed, an adversary that can use an encryption of $\boldsymbol{y} \in \mathbb{Z}_q^\ell$ to compute inner products $\langle \boldsymbol{y}, \boldsymbol{x}_i \rangle$ for ℓ independent vectors $\boldsymbol{x}_i \in \mathbb{Z}_q^\ell$ can always reconstruct $\boldsymbol{y}$, no matter how the functionality is realized. Said otherwise, bounded collusion-resistance is the best we can hope for as far as the exact computation of inner products is concerned. In the context of identity-based encryption, this relaxed notion of collusion-resistance was already known [27, 48, 40, 72] to enable constructions based on standard (non-pairing-related) assumptions. The results of [2] thus provided the first solutions beyond the mere IBE functionality.

On the downside, Abdalla *et al.* [2] only proved their schemes to be secure against *selective* adversaries, that have to declare the challenge messages M_0, M_1 of the semantic security game upfront, before seeing the master public key mpk. The security proofs of [2] rely on a technique that appeared for the first time in the work of Boneh and Boyen [10, 12], where the reduction (obliviously to the adversary) partitions the function space in two sub-spaces. As discussed by Sahai and Waters [69], this technique makes it hard to cope with adaptive adversaries when we aim at expressiveness.

Selective vs Adaptive Security. Intuitively, selective security only guarantees the security of messages that are fixed before the adversary starts its interaction with the system. While sufficient for some applications [19, 20], it is generally believed too restrictive for most practical scenarios, where the notion of full (a.k.a. adaptive) security appears much more realistic as it ensures security for adaptively chosen messages.

Historically, most flavors of functional encryption have been first realized for selective adversaries [10, 69, 45, 52, 33] before being upgraded to attain full security. Boneh and Boyen [12] observed that a standard complexity leveraging argument can be used to argue that a selectively-secure system is also adaptively secure. However, this argument is not satisfactory in general as the reduction incurs an exponential security loss in the message length.

The first examples of fully secure functional encryption appeared in the setting of IBE schemes, with the techniques of Boneh and Boyen [11] and Waters [73], where the security reductions proceed by randomly partitioning the

message space in two parts independently of the adversary's view. This random partitioning paradigm turned out to be insufficient for proving full security in more powerful forms of FE systems, like hierarchical IBE [39] for polynomially-many levels and attribute-based encryption [69], via a polynomial reduction. This problem remained open until the work of Gentry and Halevi [36] and the more general "dual system" encryption methodology developed by Waters [74]. The latter was subsequently refined so as to develop fully secure attribute-based encryption [53, 60] and predicate encryption systems [60–62].

In more general forms of functional encryption, the first adapatively secure constructions were based on multi-linear maps [35] and indistinguishability obfuscation [75]. Quite recently, Ananth, Brakerski, Segev and Vaikuntanathan [5] described an elegant generic method of building adaptively secure functional encryption systems from selectively secure ones. Their method uses the hybrid encryption paradigm and combines a selectively secure public-key FE system and an adaptively secure secret-key FE system.[1] At a high level, it uses the encryption algorithm of the selectively secure public-key system to encrypt a secret key for the private-key FE scheme, the actual message being encrypted using the one-time secret key of the symmetric FE scheme. Using similar tricks, Ananth *et al.* [5] also gave a bootstrapping technique which allows constructing a FE system for any polynomial-size circuit out of any system for shallow circuits.

While powerful, the technique of [5] incurs some overhead as the key generation algorithm of the selectively secure system is used to generate private keys for the key generation circuit of the symmetric FE component. Also, since the decryption algorithm uses the decryption procedure of the public-key component in order to generate a key for the symmetric component, it requires to begin with a selectively secure FE scheme where the decryption algorithm computes functions with a large image. For this reason, it cannot be applied to the DDH-based realization of Abdalla *et al.* [2] which can only decrypt inner products in a small interval. In order to retain the simplicity and the efficiency of constructions based on standard assumptions (like [2]), we still need to prove full security via dedicated techniques.

Our Contributions. In this paper, we describe fully secure functional encryption systems for the evaluation of inner products on encrypted data. Our constructions are essentially as efficient as those of Abdalla *et al.* [2] and rely on the same standard assumptions. In addition, our LWE-based realization allows computing inner products over $\mathbb{Z}_p$, for some prime p, whereas its predecessor [2, Section 6] evaluates them over the integers, in a smaller interval than $\mathbb{Z}_p$. As a consequence, we can evaluate polynomial in $\mathbb{Z}_p[X]$ over encrypted data.

Our DDH-based construction and its security proof implicitly build on hash proof systems [24]. It involves public parameters comprised of group elements $\big(g, h, \{h_i = g^{s_i} \cdot h^{t_i}\}_{i=1}^{\ell}\big)$, where g, h generate a cyclic group $\mathbb{G}$ of prime order q, and the master secret key is $\mathsf{msk} = (\boldsymbol{s}, \boldsymbol{t}) \in \mathbb{Z}_q^{\ell} \times \mathbb{Z}_q^{\ell}$. On input of a vector $\boldsymbol{y} = (y_1, \ldots, y_\ell) \in \mathbb{Z}_q^{\ell}$, the encryption algorithm computes $(g^r, h^r, \{g^{y_i} \cdot h_i^r\}_{i=1}^{\ell})$

[1] The latter secret-key component is implied by the results of Gorbunov *et al.* [42] as it only needs to satisfy a weak security level.

in such a way that a private key of the form $SK_{\boldsymbol{x}} = (\langle \boldsymbol{s}, \boldsymbol{x} \rangle, \langle \boldsymbol{t}, \boldsymbol{x} \rangle)$ allows computing $g^{\langle \boldsymbol{y}, \boldsymbol{x} \rangle}$ in the same way as in [2].

Despite its simplicity and its efficiency (only one more group element than in [2] is needed in the ciphertext), we show that the above system can be proved fully secure using arguments – akin to those of Cramer and Shoup [23] – which consider what the adversary knows about the master secret key $(\boldsymbol{s}, \boldsymbol{t}) \in \mathbb{Z}_q^\ell \times \mathbb{Z}_q^\ell$ in the information theoretic sense. The security proof is even simpler than its counterpart in the selective case [2]. As in all security proofs based on hash proof systems, it uses the fact that the private key is known to the reduction at any time, which makes it simpler to handle private key queries without knowing the adversary's target messages $\boldsymbol{y}_0, \boldsymbol{y}_1 \in \mathbb{Z}_q^\ell$ in advance.

Our LWE-based construction builds on the dual Regev encryption scheme from [37]. Its security analysis requires some more work. The master public key contains a random matrix $\mathbf{A} \in \mathbb{Z}_q^{m \times n}$. For simplicity, we restrict ourselves to plaintext vectors and private key vectors with binary coordinates. Each vector coordinate $i \in \{1, \ldots, \ell\}$ requires a master public key component $\boldsymbol{u}_i^T = \boldsymbol{z}_i^T \cdot \mathbf{A} \in \mathbb{Z}_q^n$, for a small norm vector $\boldsymbol{z}_i \in \mathbb{Z}^m$ made of Gaussian entries which will be part of the master secret key $\mathsf{msk} = \{\boldsymbol{z}_i\}_{i=1}^\ell$. Each $\{\mathbf{u}_i\}_{i=1}^\ell$ can be seen as a syndrome in the GPV trapdoor function for which vector $\mathbf{z}_i$ is a pre-image. Our security analysis will rely on the fact that each GPV syndrome has a large number of pre-images and, conditionally on $\mathbf{u}_i \in \mathbb{Z}_q^n$, each $\mathbf{z}_i$ retains a large amount of entropy. In the security proof, this will allow us to apply arguments similar to those of hash proof systems [24] when we will generate the challenge ciphertext using $\{\mathbf{z}_i\}_{i=1}^\ell$. More precisely, when the first part $\mathbf{c}_0 \in \mathbb{Z}_q^m$ of the ciphertext is a random vector instead of an actual LWE sample $\mathbf{c}_0 = \mathbf{A} \cdot \mathbf{s} + \mathbf{e}_0$, the action of $\{\mathbf{z}_i\}_{i=1}^\ell$ on $\mathbf{c}_0 \in \mathbb{Z}_q^m$ produces vectors that appear statistically uniform to any legitimate adversary.

In order to properly simulate the challenge ciphertext using the master secret key $\{\mathbf{z}_i\}_{i=1}^\ell$, we use a variant of the extended LWE assumption [63] (eLWE) so as to have the (hint) values $\{\langle \mathbf{z}_i, \mathbf{e}_0 \rangle\}_{i=1}^\ell$ at disposal. One difficulty is that the reductions from LWE to eLWE proved in [4] and [18] handle a single hint vector $\boldsymbol{z}$. Fortunately, we extend the techniques of Brakerski *et al.* [18] using the gadget matrix from [55] to obtain a reduction from LWE to the multi-hint variant of eLWE that we use in the security proof. More specifically, if we encrypt vectors $\boldsymbol{y} \in \mathbb{Z}_p^\ell$, for some prime modulus p, we prove that the multi-hint variant of eLWE remains as hard as LWE when the adversary obtains as many as $\ell \lceil \log p \rceil$ hints, provided the dimension n of the LWE secret is $\geq 2\ell \lceil \log p \rceil$.

Our results leave a few interesting open problems. One of them is to build efficient chosen-ciphertext-secure variants of our schemes in the standard model. To techniques suggested in [20, 45, 76] do not readily apply in our setting since they either require key delegation capabilities or they are designed for the public index setting. Our construction based on DDH can be made chosen-ciphertext secure by applying the Naor-Yung paradigm [58] and resorting to pairing-based NIZK proofs [46, 47]. Still, it requires to switch to groups endowed with an asymmetric bilinear map. It would be interesting to achieve chosen-ciphertext security

in standard (i.e., non-pairing friendly) abelian groups. Another interesting open question is how to build a ring-based [56] variant of our LWE-based solution.

2 Background

In this section, we recall the functionality and security definitions of functional and non-interactive controlled functional encryption schemes, as well as the hardness assumptions underlying the security of the schemes we will describe.

2.1 Hardness assumptions

Our first scheme relies on the standard DDH assumption in ordinary (i.e., non-pairing-friendly) cyclic groups.

Definition 1. *In a cyclic group $\mathbb{G}$ of order q, the* **Decision Diffie-Hellman** *(DDH) problem is to distinguish the distributions*

$$D_0 = \{(g, g^a, g^b, g^{ab}) \mid g \hookleftarrow \mathbb{G},\ a, b \hookleftarrow \mathbb{Z}_q\}$$

and $D_1 = \{(g, g^a, g^b, g^c) \mid g \hookleftarrow \mathbb{G},\ a, b, c \hookleftarrow \mathbb{Z}_q\}$.

Our second construction builds on the Learning-With-Errors (LWE) problem, which is known [67, 18] to be at least as hard as certain standard lattice problems in the worst case.

Definition 2. *Let q, α, m be functions of a parameter n. For a secret $\mathbf{s} \in \mathbb{Z}_q^n$, the distribution $A_{q,\alpha,\mathbf{s}}$ over $\mathbb{Z}_q^n \times \mathbb{Z}_q$ is obtained by sampling $\mathbf{a} \hookleftarrow \mathbb{Z}_q^n$ and an $e \hookleftarrow D_{\mathbb{Z},\alpha q}$, and returning $(\mathbf{a}, \langle \mathbf{a}, \mathbf{s} \rangle + e) \in \mathbb{Z}_q^{n+1}$. The Learning With Errors problem $\mathsf{LWE}_{q,\alpha,m}$ is as follows: For $\mathbf{s} \hookleftarrow \mathbb{Z}_q^n$, the goal is to distinguish between the uniform distribution over $\mathbb{Z}_q^{n+1}$ and distribution $A_{q,\alpha,\mathbf{s}}$, given access to m independent samples. We say that a PPT algorithm $\mathcal{A}$ solves $\mathsf{LWE}_{q,\alpha}$ if it distinguishes the two distributions with non-negligible advantage (over the random coins of $\mathcal{A}$ and the randomness of the samples), with non-negligible probability over the randomness of $\mathbf{s}$.*

2.2 Definitions for functional encryption

We now recall the syntax of Functional Encryption, as defined by Boneh, Sahai and Waters [17], and their indistinguishability-based security definition.

Definition 3 ([17]). *A functionality F defined over $(\mathcal{K}, \mathcal{Y})$ is a function $F : \mathcal{K} \times \mathcal{Y} \to \Sigma \cup \{\perp\}$, where $\mathcal{K}$ is a key space, $\mathcal{Y}$ is a message space and Σ is an output space, which does not contain the special symbol $\perp$.*

Definition 4. *A functional encryption (FE) scheme for a functionality F is a tuple $\mathcal{FE} = (\mathsf{Setup}, \mathsf{Keygen}, \mathsf{Encrypt}, \mathsf{Decrypt})$ of algorithms with the following specifications:*

Setup(1^λ)**:** *Takes as input a security parameter* 1^λ *and outputs a master key pair* $(\mathsf{mpk}, \mathsf{msk})$.

Keygen(msk, K)**:** *Given the master secret key* msk *and a key (i.e., a function)* $K \in \mathcal{K}$*, this algorithm outputs a key* sk_K.

Encrypt(mpk, Y)**:** *On input of a message* $Y \in \mathcal{Y}$ *and the master public key* mpk*, this randomized algorithm outputs a ciphertext* C.

Decrypt$(\mathsf{mpk}, \mathsf{sk}_K, C)$**:** *Given the master public key* mpk*, a ciphertext* C *and a key* sk_K*, this algorithm outputs* $v \in \Sigma \cup \{\perp\}$.

It is required that, for all $(\mathsf{mpk}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda)$, all keys $K \in \mathcal{K}$ and all messages $Y \in \mathcal{Y}$, if $\mathsf{sk}_K \leftarrow \mathsf{Keygen}(\mathsf{msk}, K)$ and $C \leftarrow \mathsf{Encrypt}(\mathsf{mpk}, Y)$, with overwhelming probability, we have $\mathsf{Decrypt}(\mathsf{mpk}, \mathsf{sk}_K, C) = F(K, Y)$ whenever $F(K, Y) \neq \perp$.

Indistinguishability-based security. From a security standpoint, what we expect from a FE scheme is that, given $C \leftarrow \mathsf{Encrypt}(\mathsf{mpk}, Y)$, the only thing revealed by a private key sk_K about the underlying Y is the function evaluation $F(K, Y)$. In the natural definition of indistinguishability-based security (see, e.g., [17]), one asks that no efficient adversary be able to differentiate encryption of Y_0 and Y_1 without obtaining private keys sk_K such that $F(K, Y_0) \neq F(K, Y_1)$.

Definition 5 (Indistinguishability-based security). *A functional encryption scheme* $\mathcal{FE} = (\mathsf{Setup}, \mathsf{Keygen}, \mathsf{Encrypt}, \mathsf{Decrypt})$ *provides semantic security under chosen-plaintext attacks (or IND-CPA security) if no PPT adversary has non-negligible advantage in the following game, where* $q_1 \leq q \in \mathsf{poly}(\lambda)$*:*

1. *The challenger runs* $(\mathsf{mpk}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda)$ *and the master public key* mpk *is given to the adversary* $\mathcal{A}$.
2. *The adversary adaptively makes private key queries to the challenger. At each query,* $\mathcal{A}$ *chooses a key* $K \in \mathcal{K}$ *and obtains* $\mathsf{sk}_K \leftarrow \mathsf{Keygen}(\mathsf{msk}, K)$.
3. $\mathcal{A}$ *chooses distinct messages* Y_0, Y_1 *subject to the restriction that, if* $\{K_i\}_{i=1}^{q_1}$ *denotes the set of private key queries made by* $\mathcal{A}$ *at stage* 2*, it holds that* $F(K_i, Y_0) = F(K_i, Y_1)$ *for each* $i \in \{1, \ldots, q_1\}$*. Then, the challenger flips a fair coin* $\beta \hookleftarrow \{0, 1\}$ *and computes* $C^\star \leftarrow \mathsf{Encrypt}(\mathsf{mpk}, Y_\beta)$ *which is sent as a challenge to* $\mathcal{A}$.
4. $\mathcal{A}$ *makes further private key queries for arbitrary keys* $K \in \mathcal{K}$*. However, it is required that* $F(K, Y_0) = F(K, Y_1)$ *at each query* $K \in \{K_{q_1+1}, \ldots, K_q\}$.
5. $\mathcal{A}$ *eventually outputs a bit* $\beta' \hookleftarrow \{0, 1\}$ *and wins if* $\beta' = \beta$.

The adversary's advantage is defined to be $\mathbf{Adv}_{\mathcal{A}}(\lambda) := |\Pr[\beta' = \beta] - 1/2|$*, where the probability is taken over all coin tosses.*

Definition 5 captures *adaptive* security in that the adversary is allowed to choose the messages Y_0, Y_1 at stage 3. In [2], Abdalla *et al.* considered a weaker security notion, called *selective* security, where the adversary has to declare the messages Y_0, Y_1 at the very beginning of the game, before even seeing mpk (note that, in this scenario, the adversary can receive the challenge ciphertext at the

same time as the public key). In this work, our goal will be to meet the strictly stronger requirements of adaptive security.

Boneh, Sahai and Waters [17] pinpointed shortcomings of indistinguishability-based definitions in the case of general functionalities, where they may fail to rule out intuitively insecure systems. Boneh *et al.* [17] advocated the use of simulation-based definitions, but also proved them hard to satisfy as their first natural candidate was shown [17] impossible to realize in the standard model.

For a wide range of practically interesting specific functionalities (including inner product evaluations), however, indistinguishability-based security is believed to suffice. Moreover, De Caro *et al.* gave [26] a general method of constructing FE schemes that achieve a meaningful definition of simulation-based security from systems that are only proved secure in the sense of indistinguishability-based definitions. In the following, we will thus aim at *full* security in the sense of Definition 5.

Chosen-ciphertext security. In the chosen-ciphertext scenario, the adversary is additionally granted access to a decryption oracle. At each decryption query, the adversary specifies a ciphertext C and a function $K \in \mathcal{K}$ (which may not satisfy $F(K, Y_0) = F(K, Y_1)$) and obtains the value $F(K, Y)$ obtained by decrypting C using a private key sk_K generated for K. Of course, no such decryption query is allowed for the challenge ciphertext $C^\star$.

3 Fully secure functional encryption for inner products from DDH

In this section, we show that a simple adaptation of the DDH-based construction of Abdalla *et al.* [2] provides full security under the standard DDH assumption.

In comparison with the solution of Abdalla *et al.*, we only introduce one more group element in the ciphertext and all operations are just as efficient as in [2]. Our scheme is obtained by modifying [2] in the same way as Damgård's encryption scheme [25] was obtained from the Elgamal cryptosystem. The original DDH-based system of [2] encrypts a vector $\boldsymbol{y} = (y_1, \ldots, y_\ell) \in \mathbb{Z}_q^\ell$ by computing $(g^r, \{g^{y_i} \cdot h_i^r\}_{i=1}^\ell)$, where $\{h_i = g^{s_i}\}_{i=1}^\ell$ are part of the master public key and $\mathsf{sk}_{\boldsymbol{x}} = \sum_{i=1}^\ell s_i \cdot x_i \bmod q$ is the private key associated with the vector $\boldsymbol{x} = (x_1, \ldots, x_\ell) \in \mathbb{Z}_q^\ell$. Here, we encrypt $\boldsymbol{y}$ in the fashion of Damgård's Elgamal, by computing $(g^r, h^r, \{g^{y_i} \cdot h_i^r\}_{i=1}^\ell)$. The decryption algorithm uses private keys of the form $\mathsf{sk}_{\boldsymbol{x}} = (\sum_{i=1}^\ell s_i \cdot x_i, \sum_{i=1}^\ell t_i \cdot x_i)$, where $h_i = g^{s_i} \cdot h^{t_i}$ for each $i \in \{1, \ldots, \ell\}$ and $\boldsymbol{s} = (s_1, \ldots, s_\ell) \in \mathbb{Z}_q^\ell$ and $\boldsymbol{t} = (t_1, \ldots, t_\ell) \in \mathbb{Z}_q^\ell$ are part of the master secret key msk.

The scheme and its security proof also build on ideas from the Cramer-Shoup cryptosystem [23, 24] in that the construction can also be seen as an applying a hash proof system [24] with homomorphic properties over the key space.

Setup$(1^\lambda, 1^\ell)$**:** Choose a cyclic group $\mathbb{G}$ of prime order $q > 2^\lambda$ with generators $g, h \hookleftarrow \mathbb{G}$. Then, for each $i \in \{1, \ldots, \ell\}$, choose $s_i, t_i \hookleftarrow \mathbb{Z}_q$ and compute

$h_i = g^{s_i} \cdot h^{t_i}$. Define

$$\mathsf{mpk} := \left(\mathbb{G}, g, h, \{h_i\}_{i=1}^{\ell}\right)$$

and $\mathsf{msk} := \{(s_i, t_i)\}_{i=1}^{\ell}$.

Keygen$(\mathsf{msk}, \boldsymbol{x})$**:** To generate a key for the vector $\boldsymbol{x} = (x_1, \ldots, x_\ell) \in \mathbb{Z}_q^\ell$, compute $\mathsf{sk}_{\boldsymbol{x}} = (s_{\boldsymbol{x}}, t_{\boldsymbol{x}}) = (\sum_{i=1}^{\ell} s_i \cdot x_i, \sum_{i=1}^{\ell} t_i \cdot x_i) = (\langle \boldsymbol{s}, \boldsymbol{x} \rangle, \langle \boldsymbol{t}, \boldsymbol{x} \rangle)$.

Encrypt$(\mathsf{mpk}, \boldsymbol{y})$**:** To encrypt a vector $\boldsymbol{y} = (y_1, \ldots, y_\ell) \in \mathbb{Z}_q^\ell$, choose $r \hookleftarrow \mathbb{Z}_q$ and compute

$$\begin{aligned} C &= g^r, \\ D &= h^r, \\ E_i &= g^{y_i} \cdot h_i^r, \qquad \forall i \in \{1, \ldots, \ell\}. \end{aligned}$$

Return $C_{\boldsymbol{y}} = (C, D, E_1, \ldots, E_\ell)$.

Decrypt$(\mathsf{mpk}, \mathsf{sk}_{\boldsymbol{x}}, C_{\boldsymbol{y}})$**:** Given $\mathsf{sk}_{\boldsymbol{x}} = (s_{\boldsymbol{x}}, t_{\boldsymbol{x}})$, compute

$$E_{\boldsymbol{x}} = \prod_{i=1}^{\ell} E_i^{x_i} / (C^{s_{\boldsymbol{x}}} \cdot D^{t_{\boldsymbol{x}}}).$$

Then, compute and output $\log_g(E_{\boldsymbol{x}})$.

The decryption algorithm requires to compute a discrete logarithms in an interval $\{0, \ldots, L\}$, which takes time $\Theta(L^{1/2})$ using Pollard's kangaroo method [65]. Galbraith and Ruprai [31] gave an improved algorithm with complexity $\Theta(L^{1/2})$. As reported in [9], this can be reduced to $\Theta(L^{1/3})$ operations by precomputing a table of size $\Theta(L^{1/3})$.

Like [2], our scheme requires the inner product value to live in a small, polynomial-size interval in order to enable efficient decryption. Before proceeding with the security proof, we would like to clarify that, although the scheme of [2] only decrypts values in a polynomial-size space, the usual complexity leveraging argument does not prove it fully secure via a polynomial reduction. Indeed, when ℓ is polynomial in λ, having the inner product $\langle \boldsymbol{y}, \boldsymbol{x} \rangle$ in a small interval does not mean that original vector $\boldsymbol{y} \in \mathbb{Z}_q^\ell$ lives in a polynomial-size universe.

The security analysis uses similar arguments to those of Cramer and Shoup [23, 24] in that it exploits the fact that mpk does not reveal too much information about the master secret key. At some step, the challenge ciphertext is generated using msk instead of the public key and, as long as msk retains a sufficient amount of entropy from the adversary's view, it will perfectly hide which vector among $\boldsymbol{y}_0, \boldsymbol{y}_1$ is actually encrypted.

The reason why we can prove adaptive security is the fact that, as usual in security proofs relying on hash proof systems [23, 24], the reduction knows the master secret key at any time. It can thus correctly answer all private key queries without knowing the challenge messages $\boldsymbol{y}_0, \boldsymbol{y}_1$ beforehand.

The DDH-based construction can easily be generalized so as to rely on weaker variants of DDH, like the Decision Linear assumption [13], the k-linear assumption [70] for $k > 2$ or the Cascade assumption [29]. We believe that it can be

generalized to rely on other key homomorphic hash proof systems, like those [24] based on Paillier's cryptosystem [64].

Theorem 1. *The scheme provides full security under the* DDH *assumption.*

Proof. The proof uses a sequence of games that begins with the real game and ends with a game where the adversary's advantage is zero. For each i, we denote by S_i the event that the adversary wins in Game i.

Game 0: This is the real game. In this game, the adversary $\mathcal{A}$ is given mpk. In the challenge phase, $\mathcal{A}$ chooses two distinct vectors $\boldsymbol{y}_0, \boldsymbol{y}_1 \in \mathbb{Z}_q^\ell$ and obtains an encryption of $\boldsymbol{y}_\beta = (y_{\beta,1}, \ldots, y_{\beta,\ell})$ for a random bit $\beta \hookleftarrow \{0,1\}$ chosen by the challenger $\mathcal{B}$. At the end of the game, $\mathcal{A}$ outputs $\beta' \in \{0,1\}$ and we denote by S_0 the event that $\beta' = \beta$. For any vector $\boldsymbol{x} \in \mathbb{Z}_q^\ell$ submitted to the private key extraction oracle, it must be the case that $\langle \boldsymbol{x}, \boldsymbol{y}_0 \rangle = \langle \boldsymbol{x}, \boldsymbol{y}_1 \rangle$.

Game 1: We modify the generation of the challenge $C_{\boldsymbol{y}_\beta} = (C, D, E_1, \ldots, E_\ell)$. Namely, the challenger $\mathcal{B}$ first computes

$$C = g^r, \qquad\qquad D = h^r, \tag{1}$$

for a randomly chosen $r \hookleftarrow \mathbb{Z}_q$. Then, it uses $\mathsf{msk} := \{(s_i, t_i)\}_{i=1}^\ell$ to compute

$$E_i = g^{y_{\beta,i}} \cdot C^{s_i} \cdot D^{t_i} \tag{2}$$

Clearly, $C_{\boldsymbol{y}_\beta} = (C, D, E_1, \ldots, E_\ell)$ has the same distribution as in Game 0 and we have $\Pr[S_1] = \Pr[S_0]$.

Game 2: In this game, we modify again the generation of $C_{\boldsymbol{y}_\beta} = (C, D, E_1, \ldots, E_\ell)$ in the challenge phase. Namely, instead of computing the pair (C, D) as in (1), the challenger $\mathcal{B}$ chooses $r, \tilde{r} \hookleftarrow \mathbb{Z}_q$ and sets

$$C = g^r, \qquad\qquad D = h^{\tilde{r}},$$

The ciphertext components $(E_1, \ldots, E_\ell)$ are still computed as per (2). Under the DDH assumption, this modification should not significantly affect $\mathcal{A}$'s view and a straightforward reduction shows that $|\Pr[S_2] - \Pr[S_1]| \leq \mathbf{Adv}_{\mathcal{B}}^{\mathsf{DDH}}(\lambda)$.

In Game 2, we claim that $\Pr[S_2] = 1/2$, so that the adversary has no advantage at all. To see this, we first remark that the pair (C, D) can be written $(C, D) = (g^r, h^{r+r'})$ for some uniformly random $r' \hookleftarrow \mathbb{Z}_q$. So, for each $i \in \{1, \ldots, \ell\}$, we also have

$$E_i = g^{y_{\beta,i}} \cdot C^{s_i} \cdot D^{t_i} = g^{y_{\beta,i} + \omega \cdot r' \cdot t_i} \cdot h_i^r,$$

where $\omega = \log_g(h)$. If we define $t_i' = t_i + (\omega \cdot r')^{-1} \cdot (y_{\beta,i} - y_{1-\beta,i})$ for each $i \in \{1, \ldots, \ell\}$, we also have

$$E_i = g^{y_{1-\beta,i} + \omega \cdot r' \cdot t_i'} \cdot h_i^r.$$

In other words, the vector

$$(E_1, \ldots, E_\ell) = g^{\boldsymbol{y}_\beta + \omega \cdot r' \cdot \boldsymbol{t}} \cdot (h_1^r, \ldots, h_\ell^r) \tag{3}$$

can also be written

$$(E_1, \ldots, E_\ell) = g^{\boldsymbol{y}_{1-\beta} + \omega \cdot r' \cdot \boldsymbol{t}'} \cdot (h_1^r, \ldots, h_\ell^r) \tag{4}$$

if we define $\boldsymbol{t}' = \boldsymbol{t} + (\omega \cdot r')^{-1} \cdot (\boldsymbol{y}_\beta - \boldsymbol{y}_{1-\beta}) \bmod q$. Note that, in all private keys $\mathsf{sk}_{\boldsymbol{x}}$ involving vectors $\boldsymbol{x}$ such that $\langle \boldsymbol{x}, \boldsymbol{y}_\beta \rangle = \langle \boldsymbol{x}, \boldsymbol{y}_{1-\beta} \rangle$, we have $\langle \boldsymbol{t}', \boldsymbol{x} \rangle = \langle \boldsymbol{t}, \boldsymbol{x} \rangle$. Moreover, in each private key $\mathsf{sk}_{\boldsymbol{x}} = (\langle \boldsymbol{s}, \boldsymbol{x} \rangle, \langle \boldsymbol{t}, \boldsymbol{x} \rangle)$, the information $\langle \boldsymbol{s}, \boldsymbol{x} \rangle$ is redundant with $\langle \boldsymbol{t}, \boldsymbol{x} \rangle$ since it is uniquely determined by $\langle \boldsymbol{t}, \boldsymbol{x} \rangle$ and $\prod_{i=1}^{\ell} h_i^{x_i}$. In fact, together with $\{h_i\}_{i=1}^{\ell}$, $\boldsymbol{t}'$ determines the vector $\boldsymbol{s}' = (s'_1, \ldots, s'_\ell) \in \mathbb{Z}_q^\ell$ such that $s'_i = s_i + \frac{1}{r'} \cdot (y_{1-\beta,i} - y_{\beta,i}) \bmod q$, which satisfies $h_i = g^{s'_i} \cdot h^{t'_i}$ for each $i \in \{1, \ldots, \ell\}$ and $\langle \boldsymbol{s}', \boldsymbol{x} \rangle = \langle \boldsymbol{s}, \boldsymbol{x} \rangle$ for any vector $\boldsymbol{x}$ that can be legally submitted to the key extraction oracle. It comes that situations (3) and (4) are equally likely in $\mathcal{A}$'s view as long as all revealed private keys $\mathsf{sk}_{\boldsymbol{x}}$ involve vectors $\boldsymbol{x}$ such that $\langle \boldsymbol{x}, \boldsymbol{y}_0 \rangle = \langle \boldsymbol{x}, \boldsymbol{y}_1 \rangle$. We conclude that $\Pr[S_2] = 1/2$, as claimed. In turn, this implies $\Pr[S_0] \leq \mathbf{Adv}_{\mathcal{B}}^{\mathrm{DDH}}(\lambda) + 1/2$, which yields the announced result. □

The construction can be modified to provide chosen-ciphertext security in several ways. One option is to use the random oracle model [8] and the Fujisaki-Okamoto transformation [30]. In the standard model, another option is to switch to groups endowed with an asymmetric bilinear map $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ (where the DDH assumption is believed to hold in $\mathbb{G}_1$ and $\mathbb{G}_2$) and apply the Naor-Yung/Sahai paradigm [58, 68]. In the latter case, it is particularly convenient to apply quasi-adaptive NIZK proofs of plaintext equality [50] since the constructions of [51, 54] allow for proof lengths independent of the dimension ℓ of encrypted vectors.

4 Full security under the LWE assumption

In the description hereunder, we consider the message space $\mathcal{P} = \mathbb{Z}_p^\ell$ and the key vector space $\mathcal{V} = \mathbb{Z}_p^\ell$, for some prime p and integer ℓ. Further, we define $k = \lceil \log_2 p \rceil$. Note that modulus p is related to plaintexts. We use another modulus $q = p \cdot p'$ for ciphertexts, with $p' \neq p$ prime.

We use similar notations to [38]. Namely, if $\boldsymbol{x} = (x_1, \ldots, x_\ell) \in \mathbb{Z}_p^\ell$ and $\boldsymbol{y} = (y_1, \ldots, y_\ell) \in \mathbb{Z}_p^\ell$ are vectors over $\mathbb{Z}_p$, then $\mathsf{PowersOfTwo}(\boldsymbol{y})$ stands for the vector

$$[1, 2, \ldots, 2^{k-1}] \otimes \boldsymbol{y} = (y_1, 2 \cdot y_1, \ldots, 2^{k-1} \cdot y_1, \ldots, y_\ell, 2 \cdot y_\ell, \ldots, 2^{k-1} \cdot y_\ell) \in \mathbb{Z}_p^{k\ell}$$

while $\mathsf{BitDecomp}(\boldsymbol{x})$ denotes the vector $(x_{11}, \ldots, x_{1k}, \ldots, x_{\ell 1}, \ldots, x_{\ell k}) \in \{0,1\}^{k\ell}$ such that $x_i = \sum_{j=1}^{k} x_{ij} \cdot 2^{j-1}$ for each $i \in \{1, \ldots, \ell\}$. Hence, we have the equality $\langle \mathsf{BitDecomp}(\boldsymbol{x}), \mathsf{PowersOfTwo}(\boldsymbol{y}) \rangle = \langle \boldsymbol{x}, \boldsymbol{y} \rangle \bmod p$.

For the sake of simplicity, the reader may instantiate the scheme and its analysis presented below with binary plaintext and key vectors, evaluated modulo

$p \geq 2\sqrt{\ell}$ (i.e., evaluated over the integers). Then one may set $k = 1$: in that case, the functions $\mathsf{BitDecomp}$ and $\mathsf{PowersOfTwo}$ are unnecessary. In the more general case, their purpose is to reduce the necessary size of q to ensure that the noise term in decryption does not interfere with the plaintext (see the proof of correctness of the scheme). In this simplified setup, one may also set q prime, which allows to avoid several technical hurdles (linear algebra arguments simplify for q prime).

$\mathsf{Setup}(1^n, p, 1^\ell)$: Set integers m, q, real α and distribution τ over $\mathbb{Z}^{(k\ell)\times m}$ as explained below. Sample $\mathbf{A} \hookleftarrow \mathbb{Z}_q^{m\times n}$ and $\mathbf{Z} \hookleftarrow \tau$. Compute $\mathbf{U} = \mathbf{Z} \cdot \mathbf{A} \in \mathbb{Z}_q^{k\ell\times n}$. Define $\mathsf{mpk} := (\mathbf{A}, \mathbf{U})$ and $\mathsf{msk} := \mathbf{Z}$.

$\mathsf{Keygen}(\mathsf{msk}, \boldsymbol{x})$: Given a vector $\boldsymbol{x} \in \mathcal{V}$, generate a private key as follows. Compute and return $\mathsf{sk}_{\boldsymbol{x}} = \boldsymbol{z}_{\boldsymbol{x}} := \mathsf{BitDecomp}(\boldsymbol{x})^T \cdot \mathbf{Z} \in \mathbb{Z}^m$.

$\mathsf{Encrypt}(\mathsf{mpk}, \boldsymbol{y})$: To encrypt a vector $\boldsymbol{y} \in \mathcal{P}$, sample $\boldsymbol{s} \hookleftarrow \mathbb{Z}_q^n$, $\boldsymbol{e}_0 \hookleftarrow D_{\mathbb{Z},\alpha q}^m$ and $\boldsymbol{e}_1 \hookleftarrow D_{\mathbb{Z},\alpha q}^{k\ell}$ and compute

$$\boldsymbol{c}_0 = \mathbf{A} \cdot \boldsymbol{s} + \boldsymbol{e}_0 \in \mathbb{Z}_q^m,$$
$$\boldsymbol{c}_1 = \mathbf{U} \cdot \boldsymbol{s} + \boldsymbol{e}_1 + (q/p) \cdot \mathsf{PowersOfTwo}(\boldsymbol{y}) \in \mathbb{Z}_q^{k\ell}.$$

Then, return $C := (\boldsymbol{c}_0, \boldsymbol{c}_1)$.

$\mathsf{Decrypt}(\mathsf{mpk}, \boldsymbol{x}, \mathsf{sk}_{\boldsymbol{x}}, C)$: Given $C := (\boldsymbol{c}_0, \boldsymbol{c}_1)$ and a secret key $\mathsf{sk}_{\boldsymbol{x}} = \boldsymbol{z}_{\boldsymbol{x}}$ for $\boldsymbol{x} \in \mathcal{V}$, compute $\mu' = \langle \mathsf{BitDecomp}(\boldsymbol{x}), \boldsymbol{c}_1 \rangle - \langle \boldsymbol{z}_{\boldsymbol{x}}, \boldsymbol{c}_0 \rangle \bmod q$ and output the value $\mu \in \{-p+1, \ldots, p-1\}$ that minimizes $|(q/p) \cdot \mu - \mu'|$.

Setting the parameters. Let B_τ be such that with probability $\geq 1 - n^{-\omega(1)}$, each sample from τ has norm B_τ. As explained just below, correctness may be ensured by setting $q = pp'$ with $p' \neq p$ prime, and:

$$\alpha^{-1} \geq 8\ell k B_\tau \omega(\sqrt{\log n})p, \quad \text{and } q \geq 2p.$$

The choice of τ is driven by the reduction from LWE to mheLWE, as summarized in Theorem 3 (the precise description of τ is given in Lemma 2). Each entry coefficient of matrix τ is an independent discrete Gaussian $\tau_{i,j} = D_{\mathbb{Z},\sigma_{i,j},\boldsymbol{c}_{i,j}}$, with $\sigma_{i,j} \geq \Omega(\sqrt{mn \log m})$ for all i, j, and we have $B_\tau \leq O(n^4 m^2 \log^{5/2} n)$.

To ensure security based on $\mathsf{LWE}_{q,\alpha',m}$ in dimension $\geq n/2$ via Theorems 2 and 3 below, one may further impose that $k\ell \leq n/2$, $q \leq n^{O(1)}$ and $m = \Theta(n \log n)$, to obtain $\alpha' = \Omega(\alpha/(n^6 \log^{9/2} n))$.

Decryption correctness. To show the correctness of the scheme, we first observe that, modulo q:[2]

$$\begin{aligned}\mu' &= \langle \mathsf{BitDecomp}(\boldsymbol{x}), \boldsymbol{c}_1 \rangle - \langle \boldsymbol{z}_{\boldsymbol{x}}, \boldsymbol{c}_0 \rangle \\ &= \langle \mathsf{BitDecomp}(\boldsymbol{x}), \boldsymbol{e}_1 \rangle + (q/p) \cdot \langle \mathsf{BitDecomp}(\boldsymbol{x}), \mathsf{PowersOfTwo}(\boldsymbol{y}) \rangle - \langle \boldsymbol{z}_{\boldsymbol{x}}, \boldsymbol{e}_0 \rangle \\ &= (q/p) \cdot \langle \boldsymbol{x}, \boldsymbol{y} \rangle + \langle \mathsf{BitDecomp}(\boldsymbol{x}), \boldsymbol{e}_1 \rangle - \langle \boldsymbol{z}_{\boldsymbol{x}}, \boldsymbol{e}_0 \rangle.\end{aligned}$$

[2] In this sequence of equalities, we use the fact that p divides q. If we had chosen a modulus q that is not a multiple of p, the inner product $\langle \boldsymbol{x}, \boldsymbol{y} \rangle$ would "spill over" and lead to an extra noise term. This phenomenon could be easily taken care of for decryption correctness, but may lead to security issues.

Note that the term $e' := \langle \mathsf{BitDecomp}(\boldsymbol{x}), \boldsymbol{e}_1 \rangle - \langle \boldsymbol{z}_{\boldsymbol{x}}, \boldsymbol{e}_0 \rangle$ is an inner product between an integer Gaussian vector of dimension $(k\ell+m)$ and standard deviation parameter αq, and a vector of norm $\leq 2k\ell B_\tau$ with probability $\geq 1 - n^{-\omega(1)}$. Standard techniques show that $|e'| \leq k\ell B_\tau \alpha q \cdot \omega(\sqrt{\log n})$ holds with probability $\geq 1 - n^{-\omega(n)}$. Thanks to the choices of α and q, the latter upper bound is $\leq (q/p)/4$. This suffices to guarantee decryption correctness.

Full security. In order to prove the security of the scheme, we use the extended-LWE problem introduced by O'Neill, Peikert and Water [63] and further investigated in [4, 18]. At a high level, the extended-LWE problem can be seen as $\mathsf{LWE}_{\alpha,q}$ with a fixed number m of samples, for which some extra information on the LWE noises is provided: the adversary is provided a given linear combination of the noise terms. More concretely, the problem is to distinguish between the distributions

$$\big(\mathbf{A},\ \mathbf{A} \cdot \boldsymbol{s} + \boldsymbol{e},\ \boldsymbol{z},\ \langle \boldsymbol{e}, \boldsymbol{z} \rangle\big) \ \text{ and } \ \big(\mathbf{A},\ \boldsymbol{u},\ \boldsymbol{z},\ \langle \boldsymbol{e}, \boldsymbol{z} \rangle\big),$$

where $\mathbf{A} \hookleftarrow \mathbb{Z}_q^{m \times n}, \boldsymbol{s} \hookleftarrow \mathbb{Z}_q^n, \boldsymbol{b} \hookleftarrow \mathbb{Z}_q^m, \boldsymbol{e} \hookleftarrow D^m_{\mathbb{Z},\alpha q}$, and $\boldsymbol{z}$ is sampled from a specified distribution. Note that in [63], a noise was added to the term $\langle \boldsymbol{e}, \boldsymbol{z} \rangle$. The LWE to extended-LWE reductions from [4, 18] do not require such an extra noise term.

We will use a variant of extended-LWE for which multiple hints $(\boldsymbol{z}_i, \langle \boldsymbol{e}, \boldsymbol{z}_i \rangle)$ are given, for the same noise vector $\boldsymbol{e}$.

Definition 6 (Multi-hint extended-LWE). *Let q, m, t be integers, α be a real and τ be a distribution over $\mathbb{Z}^{t \times m}$, all of them functions of a parameter n. The Multi-hint extended-LWE problem $\mathsf{mheLWE}_{q,\alpha,m,t,\tau}$ is to distinguish between the distributions of the tuples*

$$\big(\mathbf{A},\ \mathbf{A} \cdot \boldsymbol{s} + \boldsymbol{e},\ \mathbf{Z},\ \mathbf{Z} \cdot \boldsymbol{e}\big) \ \textit{ and } \ \big(\mathbf{A},\ \boldsymbol{u},\ \mathbf{Z},\ \mathbf{Z} \cdot \boldsymbol{e}\big),$$

where $\mathbf{A} \hookleftarrow \mathbb{Z}_q^{m \times n}, \boldsymbol{s} \hookleftarrow \mathbb{Z}_q^n, \boldsymbol{u} \hookleftarrow \mathbb{Z}_q^m, \boldsymbol{e} \hookleftarrow D^m_{\mathbb{Z},\alpha q}$, and $\mathbf{Z} \hookleftarrow \tau$.

Theorem 2. *Assume that $k\ell \leq n^{O(1)}$ and $m \geq 2n \log_2 q$. Then the functional encryption scheme above is fully secure, under the $\mathsf{mheLWE}_{q,\alpha,m,k\ell,\tau}$ hardness assumption.*

Proof. The proof proceeds with a sequence of games that starts with the real game and ends with a game in which the adversary's advantage is negligible. For each i, we call S_i the event that the adversary wins in Game i.

Game 0: This is the genuine full security game. Namely: the adversary $\mathcal{A}$ is given the master public key mpk; in the challenge phase, adversary $\mathcal{A}$ comes up with two distinct vectors $\boldsymbol{y}_0, \boldsymbol{y}_1 \in \mathcal{P}$ and receives an encryption C of $\boldsymbol{y}_\beta$ for $\beta \hookleftarrow \{0, 1\}$ sampled by the challenger; when $\mathcal{A}$ halts, it outputs $\beta' \in \{0, 1\}$ and S_0 is the event that $\beta' = \beta$. Note that any vector $\boldsymbol{x} \in \mathcal{V}$ queried by $\mathcal{A}$ to the secret key extraction oracle must satisfy $\langle \boldsymbol{x}, \boldsymbol{y}_0 \rangle = \langle \boldsymbol{x}, \boldsymbol{y}_1 \rangle \bmod p$ if $\mathcal{A}$ is a legitimate adversary.

Game 1: We modify the generation of $C = (\boldsymbol{c}_0, \boldsymbol{c}_1)$ in the challenge phase. Namely, at the outset of the game, the challenger picks $\boldsymbol{s} \hookleftarrow \mathbb{Z}_q^n$, $\boldsymbol{e}_0 \hookleftarrow D_{\mathbb{Z},\alpha q}^m$ (which may be chosen ahead of time) as well as $\mathbf{Z} \hookleftarrow \tau$. The master public key mpk is computed by setting $\mathbf{U} = \mathbf{Z} \cdot \mathbf{A} \bmod q$. In the challenge phase, the challenger picks a random bit $\beta \hookleftarrow \{0,1\}$ and encrypts $\boldsymbol{y}_\beta$ by computing (modulo q)

$$\begin{aligned}\boldsymbol{c}_0 &= \mathbf{A} \cdot \boldsymbol{s} + \boldsymbol{e}_0,\\ \boldsymbol{c}_1 &= \mathbf{Z} \cdot \boldsymbol{c}_0 - \mathbf{Z} \cdot \boldsymbol{e}_0 + \boldsymbol{e}_1 + (q/p) \cdot \mathsf{PowersOfTwo}(\boldsymbol{y}_\beta),\end{aligned}$$

with $\boldsymbol{e}_1 \hookleftarrow D_{\mathbb{Z},\alpha q}^{k\ell}$.

As the distribution of C is the same as in Game 0, we have $\Pr[S_1] = \Pr[S_0]$.

Game 2: We modify again the generation of $C = (\boldsymbol{c}_0, \boldsymbol{c}_1)$ in the challenge phase. Namely, the challenger picks $\boldsymbol{u} \hookleftarrow \mathbb{Z}_q^m$, sets $\boldsymbol{c}_0 = \boldsymbol{u}$ and computes $\boldsymbol{c}_1$ using $\boldsymbol{c}_0, \mathbf{Z}$ and $\boldsymbol{e}_0$ as in Game 1.

Under the mheLWE hardness assumption with $t = k\ell$, this modification has no noticeable effect on the behavior of $\mathcal{A}$. Below, we prove that $\Pr[S_2] \approx 1/2$, which completes the proof of the theorem.

In Game 2, we have $\boldsymbol{c}_1 = \mathbf{Z}\boldsymbol{u} - \boldsymbol{f} + (q/p) \cdot \mathsf{PowersOfTwo}(\boldsymbol{y}_\beta)$, with $\boldsymbol{f} := -\mathbf{Z}\boldsymbol{e}_0 + \boldsymbol{e}_1$. Let $\boldsymbol{x}^i \in \mathcal{V}$ be the vectors corresponding to the secret key queries made by $\mathcal{A}$. As $\mathcal{A}$ is a legitimate adversary, we have $\langle \boldsymbol{x}^i, \boldsymbol{y}_0 \rangle = \langle \boldsymbol{x}^i, \boldsymbol{y}_1 \rangle \bmod p$ for all i. This implies that $\langle \boldsymbol{x}^i, \boldsymbol{y}_0 - \boldsymbol{y}_1 \rangle = 0 \bmod p$, for all i. Let $\mathbf{X}_{top}^{(p)} \in \mathbb{Z}_p^{(k\ell-1)\times k\ell}$ be a uniformly chosen basis of the $\mathbb{Z}_p$-vector subspace

$$\{\boldsymbol{x} \in \mathbb{Z}_p^{k\ell} : \langle \boldsymbol{x}, \mathsf{PowersOfTwo}(\boldsymbol{y}_0 - \boldsymbol{y}_1) \rangle = 0\}.$$

We choose $\mathbf{X}_{top}^{(p')} \in \mathbb{Z}_{p'}^{(k\ell-1)\times k\ell}$ full-rank, and let $\mathbf{X}_{top} \in \mathbb{Z}_q^{(k\ell-1)\times k\ell}$ be such that $\mathbf{X}_{top} = \mathbf{X}_{top}^{(p)} \mod p$ and $\mathbf{X}_{top} = \mathbf{X}_{top}^{(p')} \mod p'$. Let $\mathbf{X}_{bot} \in \mathbb{Z}_q^{1\times k\ell}$ be such that the matrix $\mathbf{X} \in \mathbb{Z}_q^{k\ell\times k\ell}$ obtained by putting $\mathbf{X}_{top}$ on top of $\mathbf{X}_{bot}$ is invertible (modulo q). We have:

$$\boldsymbol{c}_1 = \mathbf{X}^{-1} \cdot \mathbf{X} \cdot (\mathbf{Z}\boldsymbol{u} - \boldsymbol{f} + (q/p) \cdot \mathsf{PowersOfTwo}(\boldsymbol{y}_\beta)).$$

We are to show that the distribution of $\mathbf{X} \cdot \boldsymbol{c}_1$ is (almost) independent of β. As we built $\mathbf{X}$ so that it is independent of β and invertible, this implies that the distribution of $\boldsymbol{c}_1$ is (almost) independent of β and $\Pr[S_2] \approx 1/2$.

The first $k\ell-1$ entries of $\mathbf{X}\cdot\boldsymbol{c}_1$ do not depend on β because we have (modulo q)

$$\begin{aligned}(q/p)\mathbf{X}_{top} \cdot \mathsf{PowersOfTwo}(\boldsymbol{y}_0) &= (q/p)\mathbf{X}_{top}^{(p)} \cdot \mathsf{PowersOfTwo}(\boldsymbol{y}_0)\\ &= (q/p)\mathbf{X}_{top}^{(p)} \cdot \mathsf{PowersOfTwo}(\boldsymbol{y}_1)\\ &= (q/p)\mathbf{X}_{top} \cdot \mathsf{PowersOfTwo}(\boldsymbol{y}_1).\end{aligned}$$

It remains to study the last entry of $\mathbf{X} \cdot \boldsymbol{c}_1$. Note that, by Lemma 6 in Appendix A, we have that the distribution of the pair $((\mathbf{A}|\boldsymbol{u}), \mathbf{Z}(\mathbf{A}|\boldsymbol{u}))$ is within statistical distance $n^{-\omega(1)}$ from the uniform distribution over $\mathbb{Z}_q^{m\times(n+1)} \times \mathbb{Z}_q^{k\ell\times(n+1)}$.

As a consequence, the distribution of the tuple $(\mathbf{A}, \boldsymbol{u}, \mathbf{ZA}, \mathbf{XZ}\boldsymbol{u})$ is within statistical distance $n^{-\omega(1)}$ from the uniform distribution over $\mathbb{Z}_q^{m\times n} \times \mathbb{Z}_q^m \times \mathbb{Z}_q^{k\ell\times n} \times \mathbb{Z}_q^{k\ell}$. This implies that given $\mathbf{A}, \boldsymbol{u}, \mathbf{ZA}$ and $\mathbf{X}_{top}\mathbf{Z}\boldsymbol{u}$, the distribution of $\mathbf{X}_{bot}\mathbf{Z}\boldsymbol{u}$ is within statistical distance $n^{-\omega(1)}$ from the uniform distribution over $\mathbb{Z}_q$. This completes the security proof. □

4.1 Hardness of multi-hint extended-LWE

In this section, we prove the following theorem, which shows that for some parameters, the mheLWE problem is no easier than the LWE problem.

Theorem 3. *Let $n \geq 100$, $q \geq 2$, $t < n$ and m with $m = \Omega(n \log n)$ and $m \leq n^{O(1)}$. There exists $\xi \leq O(n^4 m^2 \log^{5/2} n)$ and a distribution τ over $\mathbb{Z}^{t\times m}$ such that the following statements hold:*

- *There is a reduction from $\mathsf{LWE}_{q,\alpha,m}$ in dimension $n-t$ to $\mathsf{mheLWE}_{q,\alpha\xi,m,t,\tau}$ that reduces the advantage by at most $2^{\Omega(t-n)}$,*
- *It is possible to sample from τ in time polynomial in n,*
- *Each entry coefficient of matrix τ is an independent discrete Gaussian $\tau_{i,j} = D_{\mathbb{Z},\sigma_{i,j},\mathbf{c}_{i,j}}$ for some $\mathbf{c}_{i,j}$ and $\sigma_{i,j} \geq \Omega(\sqrt{mn \log m})$,*
- *With probability $\geq 1 - n^{-\omega(1)}$, all rows from a sample from τ have norms $\leq \xi$.*

Our reduction from LWE to mheLWE proceeds as the reduction from LWE to extended-LWE from [18], using the matrix gadget from [55] to handle the multiple hints. We first reduce LWE to the following variant of LWE in which the first samples are noise-free. This problem generalizes the first-is-errorless LWE problem from [18].

Definition 7 (First-are-errorless LWE). *Let q, α, m, t be functions of a parameter n. The first-are-errorless LWE problem $\mathsf{faeLWE}_{q,\alpha,m,t}$ is defined as follows: For $\mathbf{s} \hookleftarrow \mathbb{Z}_q^n$, the goal is to distinguish between the following two scenarios. In the first, all m samples are uniform over $\mathbb{Z}_q^n \times \mathbb{Z}_q$. In the second, the first t samples are from $A_{q,\{0\},\boldsymbol{s}}$ (where $\{0\}$ denotes the distribution that is deterministically zero) and the rest are from $A_{q,\alpha,\boldsymbol{s}}$.*

Lemma 1. *For any $n > t$, $m, q \geq 2$, and $\alpha \in (0,1)$, there is an efficient reduction from $\mathsf{LWE}_{q,\alpha,m}$ in dimension $n-t$ to $\mathsf{faeLWE}_{q,\alpha,m,t}$ in dimension n that reduces the advantage by at most 2^{-n+t+1}.*

The proof, postponed to the appendices, is a direct adaptation of the one of [18, Le. 4.3].

In our reduction from faeLWE to mheLWE, we use the following gadget matrix from [55, Cor. 10]. It generalizes the matrix construction from [18, Claim 4.6].

Lemma 2. *Let n, m_1, m_2 with $100 \leq n \leq m_1 \leq m_2 \leq n^{O(1)}$. Let $\sigma_1, \sigma_2 > 0$ be standard deviation parameters such that $\sigma_1 \geq \Omega(\sqrt{m_1 n \log m_1})$, $m_1 \geq \Omega(n \log(\sigma_1 n))$ and $\sigma_2 \geq \Omega(n^{5/2}\sqrt{m_1}\sigma_1^2 \log^{3/2}(m_1\sigma_1))$. Let $m = m_1 + m_2$. There exists a probabilistic polynomial time algorithm that given n, m_1, m_2 (in unary) and σ_1, σ_2 as inputs, outputs $\mathbf{G} \in \mathbb{Z}^{m\times m}$ such that:*

- *The top $n \times m$ submatrix of $\mathbf{G}$ is within statistical distance $2^{-\Omega(n)}$ of $\tau = D_{\mathbb{Z},\sigma_1}^{n \times m_1} \times (D_{\mathbb{Z}^{m_2},\sigma_2,\boldsymbol{\delta}_1} \times \ldots \times D_{\mathbb{Z}^{m_2},\sigma_2,\boldsymbol{\delta}_n})^T$ with $\boldsymbol{\delta}_i$ denoting the ith canonical unit vector,*
- *We have $|\det(\mathbf{G})| = 1$ and $\|\mathbf{G}^{-1}\| \leq O(\sqrt{nm_2}\sigma_2)$, with probability $\geq 1 - 2^{-\Omega(n)}$.*

Lemma 3. *Let $n, m_1, m_2, m, \sigma_1, \sigma_2, \tau$ be as in Lemma 2, and $\xi \geq \Omega(\sqrt{nm_2}\sigma_2)$. Let $q \geq 2$, $t \leq n$, $\alpha \geq \Omega(\sqrt{n}/q)$. Let τ_t be the distribution obtained by keeping only the first t rows from a sample from τ. There is a (dimension-preserving) reduction from $\mathsf{faeLWE}_{q,\alpha,m,t}$ to $\mathsf{mheLWE}_{q,2\alpha\xi,m,t,\tau_t}$ that reduces the advantage by at most $2^{-\Omega(n)}$.*

Proof. Let us first describe the reduction. Let $(\mathbf{A}, \boldsymbol{b}) \in \mathbb{Z}_q^m \times \mathbb{Z}_q$ be the input, which is either sampled from the uniform distribution, or from distribution $A_{q,\{0\},\boldsymbol{s}}^t \times A_{q,\alpha,\boldsymbol{s}}^{m-t}$ for some fixed $\boldsymbol{s} \hookleftarrow \mathbb{Z}_q^n$. Our objective is to distinguish between the two scenarios, using an mheLWE oracle. We compute $\mathbf{G}$ as in Lemma 2 and let $\mathbf{U} = \mathbf{G}^{-1}$. We let $\mathbf{Z} \in \mathbb{Z}^{t \times m}$ denote the matrix formed by the top t rows of $\mathbf{G}$, and let $\mathbf{U}' \in \mathbb{Z}^{m \times (m-t)}$ denote the matrix formed by the right $m - t$ columns of $\mathbf{U}$. By construction, we have $\mathbf{ZU}' = \mathbf{0}$. We define $\mathbf{A}' = \mathbf{U} \cdot \mathbf{A} \bmod q$. We sample $\boldsymbol{f} \hookleftarrow D_{\alpha q(\xi^2 \mathbf{I} - \mathbf{U}'\mathbf{U}'^T)^{1/2}}$ (thanks to Lemma 2 and the choice of ξ, the matrix $\xi^2\mathbf{I} - \mathbf{U}'\mathbf{U}'^T$ is positive definite). We sample $\boldsymbol{e}'$ from $\{0\}^t \times D_{\alpha q}^{m-t}$ and define $\boldsymbol{b}' = \mathbf{U} \cdot (\boldsymbol{b} + \boldsymbol{e}') + \boldsymbol{f}$. We then sample $\boldsymbol{c} \hookleftarrow D_{\mathbb{Z}^m - \boldsymbol{b}', \sqrt{2}\alpha\xi q}$, and define $\boldsymbol{h} = \mathbf{Z}(\boldsymbol{f} + \boldsymbol{c})$.

Finally, the reduction calls the mheLWE oracle on input $(\mathbf{A}', \boldsymbol{b}' + \boldsymbol{c}, \mathbf{Z}, \boldsymbol{h})$, and outputs the reply.

Correctness is obtained by showing that distribution $A_{q,\{0\},\boldsymbol{s}}^t \times A_{q,\alpha,\boldsymbol{s}}^{m-t}$ is mapped to the mheLWE "LWE" distribution and that the uniform distribution is mapped to the mheLWE "uniform" distribution, up to $2^{-\Omega(n)}$ statistical distances (we do not discuss these tiny statistical discrepancies below). The proof is identical to the reduction analysis in the proof of [18, Le. 4.7]. □

Theorem 3 is obtained by combining Lemmas 1, 2 and 3.

Acknowledgements

This work has been supported in part by ERC Starting Grant ERC-2013-StG-335086-LATTAC. Part of this work was also funded by the "Programme Avenir Lyon Saint-Etienne de l'Université de Lyon" in the framework of the programme "Inverstissements d'Avenir" (ANR-11-IDEX-0007).

References

1. M. Abdalla, M. Bellare, D. Catalano, E. Kiltz, T. Kohno, T. Lange, J. Malone-Lee, G. Neven, P. Paillier, and H. Shi. Searchable encryption revisited: Consistency properties, relation to anonymous ibe, and extensions. In *Proc. of Crypto*, volume 3621 of *LNCS*, pages 205–222. Springer, 2005.

2. M. Abdalla, F. Bourse, A. De Caro, and D. Pointcheval. Simple functional encryption schemes for inner products. In *Proceedings of PKC*, volume 9020 of *LNCS*, pages 733–751. Springer, 2015.
3. S. Agrawal, D. M. Freeman, and V. Vaikuntanathan. Functional encryption for inner product predicates from learning with errors. In *Proc. of Asiacrypt*, volume 7073 of *LNCS*, pages 21–40. Springer, 2011.
4. J. Alperin-Sheriff and C. Peikert. Circular and kdm security for identity-based encryption. In *Proceedings of PKC*, volume 7293 of *LNCS*, pages 334–352. Springer, 2012.
5. P. Ananth, Z. Brakerski, G. Segev, and V. Vaikuntanathan. The trojan method in functional encryption: From selective to adaptive security, generically. In *Proc. of Crypto*, LNCS. Springer, 2015.
6. B. Barak, O. Goldreich, R. Impagliazzo, S. Rudich, A. Sahai, S. Vadhan, and K. Yang. On the (im)possibility of obfuscating programs. In *Proc. of Crypto*, volume 2139 of *LNCS*, pages 1–18. Springer, 2001.
7. M. Bellare, A. Boldyreva, K. Kurosawa, and J. Staddon. Multi-recipient encryption schemes: How to save on bandwidth and computation without sacrificing security. *IEEE Transactions on Information Theory*, 53(11):3927–3943, November 2007.
8. M. Bellare and P. Rogaway. Random oracles are practical: a paradigm for designing efficient protocols. In *Proc. of the 1st CCS*, pages 62–73. ACM Press, 1993.
9. D. Bernstein and T. Lange. Computing small discrete logarithms faster. In *Proc. of Indocrypt'12*, volume 7668 of *LNCS*, pages 317–338. Springer, 2012.
10. D. Boneh and X. Boyen. Efficient selective-id secure identity-based encryption without random oracles. In *Proc. of Eurocrypt*, volume 3027 of *LNCS*, pages 223–238. Springer, 2004.
11. D. Boneh and X. Boyen. Secure identity-based encryption without random oracles. In *Proc. of Crypto*, volume 3152 of *LNCS*, pages 443–459. Springer, 2004.
12. D. Boneh and X. Boyen. Efficient selective identity-based encryption without random oracles. *Journal of Cryptology*, 24(4):659–693, 2011.
13. D. Boneh, X. Boyen, and H. Shacham. Short group signatures. In *Proc. of Crypto*, volume 3152 of *LNCS*, pages 41–55. Springer, 2004.
14. D. Boneh, G. Di Crescenzo, R. Ostrovsky, and G. Persiano. Public key encryption with keyword search. In *Proc. of Eurocrypt*, volume 3027 of *LNCS*, pages 506–522. Springer, 2004.
15. D. Boneh and M. Franklin. Identity-based encryption from the Weil pairing. *SIAM J. Comput*, 32(3):586–615 (electronic), 2003.
16. D. Boneh, C. Gentry, S. Gorbunov, S. Halevi, V. Nikolaenko, G. Segev, V. Vaikuntanathan, and D. Vinayagamurthy. Fully key-homomorphic encryption, arithmetic abe and compact garbled circuits. In *Proc. of Eurocrypt*, volume 8441 of *LNCS*, pages 533–556. Springer, 2014.
17. D. Boneh, A. Sahai, and B. Waters. Functional encryption: Definitions and challenges. In *Proc. of TCC*, volume 6597 of *LNCS*, pages 253–273. Springer, 2011.
18. Z. Brakerski, A. Langlois, C. Peikert, Regev. O., and D. Stehlé. On the classical hardness of learning with errors. In *Proc. of STOC*, pages 575–584. ACM, 2013.
19. R. Canetti, S. Halevi, and J. Katz. A forward-secure public-key encryption scheme. In *Proc. of Eurocrypt*, volume 2656 of *LNCS*, pages 255–271. Springer, 2003.
20. R. Canetti, S. Halevi, and J. Katz. Chosen-ciphertext security from identity-based encryption. In *Proc. of Eurocrypt*, volume 3027 of *LNCS*, pages 207–222. Springer, 2004.

21. J.-H. Cheon, K. Han, C. Lee, H. Ryu, and D. Stehlé. Cryptanalysis of the multilinear map over the integers. In *Proc. of Eurocrypt*, volume 9056 of *LNCS*, pages 3–12. Springer, 2015.
22. J.-S. Coron, T. Lepoint, and M. Tibouchi. Practical multilinear maps over the integers. In *Proc. of Crypto*, volume 8042 of *LNCS*, pages 476–493. Springer, 2013.
23. R. Cramer and V. Shoup. A practical public key cryptosystem provably secure against adaptive chosen ciphertext attack. In *Proc. of CRYPTO*, volume 1462 of *LNCS*, pages 13–25. Springer, 1998.
24. R. Cramer and V. Shoup. Universal hash proofs and a paradigm for adaptive chosen ciphertext secure public-key encryption. In *Proc. of Eurocrypt*, volume 2332 of *LNCS*, pages 45–64. Springer, 2002.
25. I Damgård. Towards practical public key systems secure against chosen ciphertext attacks. In *Proc. of CRYPTO*, volume 576 of *LNCS*, pages 445–456. Springer, 1991.
26. A. De Caro, V. Iovino, A. Jain, A. O'Neill, O. Paneth, and G. Persiano. On the achievability of simulation-based security for functional encryption. In *Crypto'13*, volume 8043 of *LNCS*, pages 519–535. Springer, 2013.
27. Y. Dodis, J. Katz, S. Xu, and M. Yung. Key-insulated public key cryptosystems. In *Proceedings of Eurocrypt*, volume 2332 of *LNCS*, pages 65–82. Springer, 2002.
28. T. ElGamal. A public key cryptosystem and a signature scheme based on discrete logarithms. *IEEE Trans. Inf. Th.*, 31(4):469–472, 1985.
29. A. Escala, G. Herold, E. Kiltz, C. Ràfols, and G. Villard. An algebraic framework for diffie-hellman assumptions. In *Proc. of Crypto*, volume 8043 of *LNCS*, pages 129–147. Springer, 2013.
30. E. Fujisaki and T. Okamoto. How to enhance the security of public-key encryption at minimum cost. In *Proc. of PKC '99*, volume 1560 of *LNCS*, pages 53–68, 1999.
31. S. Galbraith and R. Ruprai. Using equivalence classes to accelerate solving the discrete logarithm problem in a short interval. In *Proc. of PKC'10*, volume 6056 of *LNCS*, pages 368–383. Springer, 2010.
32. S. Garg, C. Gentry, and S. Halevi. Candidate multilinear maps from ideal lattices and applications. In *Proc. of Eurocrypt*, LNCS. Springer, 2013.
33. S. Garg, C. Gentry, S. Halevi, M. Raykova, A. Sahai, and B. Waters. Candidate indistinguishability obfuscation and functional encryption for all circuits. In *Proc. of FOCS*, pages 40–49, 2013.
34. S. Garg, C. Gentry, S. Halevi, A. Sahai, and B. Waters. Attribute-based encryption for circuits from multilinear maps. In *Proc. of Crypto*, volume 8043 of *LNCS*, pages 479–499. Springer, 2013.
35. S. Garg, C. Gentry, S. Halevi, and M. Zhandry. Fully secure functional encryption without obfuscation. Cryptology ePrint Archive: Report 2014/666, 2014.
36. C. Gentry and S. Halevi. Hierarchical identity based encryption with polynomially many levels. In *Proc. of TCC*, volume 5444 of *LNCS*, pages 437–456. Springer, 2009.
37. C. Gentry, C. Peikert, and V. Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In *Proc. of STOC*, pages 197–206. ACM, 2008.
38. C. Gentry, A. Sahai, and B. Waters. Homomorphic encryption from learning with errors: Conceptually-simpler, asymptotically-faster, attribute-based. In *Proc. of CRYPTO*, volume 8042 of *LNCS*, pages 75–92. Springer, 2013.
39. C. Gentry and A. Silverberg. Hierarchical id-based cryptography. In *Proc. of Asiacrypt*, volume 2501 of *LNCS*, pages 548–566. Springer, 2002.

40. S. Goldwasser, A. Lewko, and D. Wilson. Bounded-collusion ibe from key homomorphism. In *Proceedings of TCC*, volume 7194 of *LNCS*, pages 564–581. Springer, 2012.
41. S. Goldwasser, Y. Tauman Kalai, R. Popa, V. Vaikuntanathan, and N. Zeldovich. Reusable garbled circuits and succinct functional encryption. In *Proc. of STOC*, pages 555–564. ACM Press, 2013.
42. S. Gorbunov, V. Vaikuntanathan, and H. Wee. Functional encryption with bounded collusions via multi-party computation. In *Proc. of Crypto*, volume 7417 of *LNCS*, pages 162–179. Springer, 2012.
43. S. Gorbunov, V. Vaikuntanathan, and H. Wee. Attribute-based encryption for circuits. In *Proc. of STOC*, pages 545–554. ACM Press, 2013.
44. S. Gorbunov, V. Vaikuntanathan, and H. Wee. Predicate encryption for circuits from lwe. In *Proc. of Crypto*, LNCS. Springer, 2015.
45. V. Goyal, O. Pandey, A. Sahai, and B. Waters. Attribute-based encryption for fine-grained access control of encrypted data. In *Proc. of ACM-CCS'06*, pages 89–98. ACM Press, 2006.
46. J. Groth. Simulation-sound nizk proofs for a practical language and constant size group signatures. In *Proc. of Asiacrypt*, volume 4284 of *LNCS*, pages 444–459. Springer, 2006.
47. J. Groth and A. Sahai. Efficient non-interactive proof systems for bilinear groups. In *Proc. of Eurocrypt*, volume 4965 of *LNCS*, pages 415–432. Springer, 2008.
48. S.-H. Heng and K. Kurosawa. k-resilient identity-based encryption in the standard model. In *Proceedings of CT-RSA*, volume 2964 of *LNCS*, pages 67–80. Springer, 2004.
49. Y. Hu and J. Huiwen. Cryptanalysis of ggh map. Cryptology ePrint Archive: Report 2015/301, 2015.
50. C. Jutla and A. Roy. Shorter quasi-adaptive nizk proofs for linear subspaces. In *Proc. of Asiacrypt '13*, volume 8617 of *LNCS*, pages 1–20, 2013.
51. C. Jutla and A. Roy. Switching lemma for bilinear tests and constant-size nizk proofs for linear subspaces. In *Proc. of Crypto '14*, volume 8617 of *LNCS*, pages 295–312, 2014.
52. J. Katz, A. Sahai, and B. Waters. Predicate encryption supporting disjunctions, polynomial equations, and inner products. In *Proc. of EUROCRYPT*, volume 4965 of *LNCS*, pages 146–162. Springer, 2008.
53. A. Lewko, E. Okamoto, A. Sahai, K. Takashima, and B. Waters. Fully secure functional encryption: Attribute-based encryption and (hierarchical) inner product encryption. In *Proc. of Eurocrypt'10*, volume 6110 of *LNCS*, pages 62–91. Springer, 2010.
54. B. Libert, T. Peters, M. Joye, and M. Yung. Non-malleability from malleability: Simulation-sound quasi-adaptive nizk proofs and cca2-secure encryption from homomorphic signatures. In *Proc. of Eurocrypt '14*, volume 8441 of *LNCS*, pages 514–532, 2014.
55. S. Ling, D. H. Phan, D. Stehlé, and R. Steinfeld. Hardness of k-LWE and Applications in Traitor Tracing. In *Proc. of CRYPTO*, volume 8616 of *LNCS*, pages 315–334. Springer, 2014.
56. V. Lyubashevsky, C. Peikert, and O. Regev. On ideal lattices and learning with errors over rings. In *Proc. of Eurocrypt*, volume 6110 of *LNCS*, pages 1–23. Springer, 2010.
57. D. Micciancio and C. Peikert. Trapdoors for lattices: Simpler, tighter, faster, smaller. In *Proc. of Eurocrypt*, volume 7237 of *LNCS*, pages 700–718. Springer, 2012.

58. M. Naor and M. Yung. Public-key cryptosystems provably secure against chosen ciphertext attacks. In *Proc. of the 22nd STOC*, pages 427–437. ACM Press, New York, 1990.
59. M. Naveed, S. Agrawal, M. Prabhakaran, X. Wang, E. Ayday, J.-P. Hubaux, and C. Gunter. Controlled functional encryption. In *Proc. of ACM-CCS*, pages 1280–1291. ACM Press, 2014.
60. E. Okamoto and K. Takashima. Fully secure functional encryption with general relations from the decisional linear assumption. In *Proc. of Crypto'10*, volume 6223 of *LNCS*, pages 191–208. Springer, 2010.
61. E. Okamoto and K. Takashima. Adaptively attribute-hiding (hierarchical) inner product encryption. In *Proc. of Eurocrypt'12*, volume 7237 of *LNCS*, pages 591–608. Springer, 2012.
62. E. Okamoto and K. Takashima. Fully secure unbounded inner-product and attribute-based encryption. In *Proc. of Asiacrypt'12*, volume 7658 of *LNCS*, pages 349–366. Springer, 2012.
63. A. O'Neill, C. Peikert, and B. Waters. Bi-deniable public-key encryption. In *Proceedings of Crypto*, volume 6841 of *LNCS*, pages 525–542. Springer, 2011.
64. P. Paillier. Public-key cryptosystems based on composite degree residuosity classes. In *Proc. of Eurocrypt*, volume 1592 of *LNCS*, pages 223–238. Springer, 1999.
65. J. Pollard. Kangaroos, monopoly and discrete logarithms. *Journal of Cryptology*, 13:433–447, 2000.
66. O. Regev. On lattices, learning with errors, random linear codes, and cryptography. In *Proc. of STOC*, pages 84–93. ACM, 2005.
67. O. Regev. On lattices, learning with errors, random linear codes, and cryptography. *J. ACM*, 56(6), 2009.
68. A. Sahai. Non-malleable non-interactive zero knowledge and adaptive chosen-ciphertext security. In *Proc. of FOCS*, pages 543–553, 1999.
69. A. Sahai and B. Waters. Fuzzy identity-based encryption. In *Proc. of EUROCRYPT*, volume 3494 of *LNCS*, pages 457–473. Springer, 2005.
70. H. Shacham. A cramer-shoup encryption scheme from the linear assumption and from progressively weaker linear variants. Cryptology ePrint Archive: Report 2007/074, 2007.
71. A. Shamir. Identity-based cryptosystems and signature schemes. In *Proc. of Crypto*, volume 196 of *LNCS*, pages 47–53. Springer, 1984.
72. S. Tessaro and D. Wilson. Bounded-collusion identity-based encryption from semantically-secure public-key encryption: Generic constructions with short ciphertexts. In *Proceedings of PKC*, volume 8383 of *LNCS*, pages 257–274. Springer, 2014.
73. B. Waters. Efficient identity-based encryption without random oracles. In *Proc. of EUROCRYPT*, volume 3494 of *LNCS*, pages 114–127. Springer, 2005.
74. B. Waters. Dual system encryption: Realizing fully secure ibe and hibe under simple assumptions. In *Proc. of Crypto*, volume 5677 of *LNCS*, pages 619–636. Springer, 2009.
75. B. Waters. A punctured programming approach to adaptively secure functional encryption. In *Proc. of Crypto*, LNCS. Springer, 2015.
76. S. Yamada, N. Attrapadung, G. Hanaoka, and N. Kunihiro. Generic constructions for chosen-ciphertext secure attribute based encryption. In *Proc. of PKC'11*, volume 6571 of *LNCS*, pages 71–89. Springer, 2011.

A Missing material from Section 4

Let Λ be a non-zero lattice. We recall that the smoothing parameter of Λ is defined as $\eta_\varepsilon(\Lambda) = \min(s > 0 : \sum_{\hat{\boldsymbol{b}} \in \hat{\Lambda}} \exp(-\pi \|\hat{\boldsymbol{b}}\|^2 / s^2) \leq 1 + \varepsilon)$, where $\hat{\Lambda}$ refers to the dual of Λ. Further, for a matrix $\mathbf{A} \in \mathbb{Z}_q^{m \times n}$ for some integers m, n, q, we define the lattice $\Lambda^\perp(\mathbf{A}) = \{\boldsymbol{x} \in \mathbb{Z}^m : \boldsymbol{x}^T \cdot \mathbf{A} = \mathbf{0} \bmod q\}$.

Lemma 4 (Adapted from [57, Le. 2.4]). *Let $n, m, q \geq 2$ be positive integers, and $\varepsilon, \delta > 0$. Assume that $q = p \cdot p'$ for $p \neq p'$ primes, and that*

$$m \geq \max\left(n + \frac{\log(12/(\delta\varepsilon))}{\log \min(p, p')}, \frac{n \log q + \log(4/(\delta\varepsilon))}{\log 2}\right).$$

Then $\eta_\varepsilon(\Lambda^\perp(\mathbf{A})) \leq 2\sqrt{\ln(2m(1 + 2/(\delta\varepsilon)))/\pi}$, except with probability $\leq \delta$ over the uniform choice of $\mathbf{A} \in \mathbb{Z}_q^{m \times n}$.

Lemma 5 (Adapted from [37, Le. 5.2]). *Assume the rows of $\mathbf{A} \in \mathbb{Z}_q^{m \times n}$ generate $\mathbb{Z}_q^n$ and let $\varepsilon \in (0, 1/2)$, $s_i \geq \eta_\varepsilon(\Lambda^\perp(\mathbf{A}))$ for $i \in [m]$, and $\boldsymbol{c} \in \mathbb{Z}^m$. Then for $\boldsymbol{e} \in \mathbb{Z}^m$ with each coordinate sampled independently from $D_{\mathbb{Z}, s_i, c_i}$, the distribution of the syndrome $\boldsymbol{e}^T \cdot \mathbf{A} \bmod q$ is within statistical distance 2ε of uniform over $\mathbb{Z}_q^n$.*

Using the two lemmas above, we obtain the following result, that we use in the proof of Theorem 2.

Lemma 6. *Let $n, m, q \geq 2$ be positive integers. Assume that $q = p \cdot p'$ for $p \neq p'$ primes, and that $m \geq 2n \log_2 q$. Let $s_i \geq \omega(\sqrt{\log m})$ for $i \in [m]$, and $\boldsymbol{c} \in \mathbb{Z}^m$. Then for $\mathbf{A} \in \mathbb{Z}_q^{m \times n}$ sampled uniformly and $\boldsymbol{e} \in \mathbb{Z}^m$ with each coordinate sampled independently from $D_{\mathbb{Z}, s_i, c_i}$, the distribution of the pair $(\mathbf{A}, \boldsymbol{e}^T \cdot \mathbf{A})$ is within statistical distance $n^{-\omega(1)}$ of uniform over $\mathbb{Z}_q^{m \times n} \times \mathbb{Z}_q^n$.*

Proof of Lemma 1. The reduction from LWE to faeLWE starts by sampling $\mathbf{A}' \hookleftarrow \mathbb{Z}_q^{t \times n}$. It aborts if it is not full-rank (modulo q): this happens with probability

$$\leq \prod_{p \text{ prime}, p | q} \left(1 - \prod_{0 \leq i < t} (1 - p^{-n+i})\right) \leq \prod_{p \text{ prime}, p | q} (4p^{-n+t-1}) \leq 2^{-n+t+1}.$$

Else, the reduction computes $\mathbf{R} \in \mathbb{Z}_q^{n \times n}$ which is invertible and whose top $t \times n$ submatrix is $\mathbf{A}'$. The reduction also samples $\boldsymbol{s}' \hookleftarrow \mathbb{Z}_q^t$. The first t output samples are $(\boldsymbol{a}'_i, s'_i)$ (for $i \leq t$), where $\boldsymbol{a}'_i$ denote the ith row of $\mathbf{A}'$. The remaining samples are produced by taking a sample $(\boldsymbol{a}, b) \in \mathbb{Z}_q^{n-t} \times \mathbb{Z}_q$ from the given oracle, picking a fresh uniformly random $\boldsymbol{d} \in \mathbb{Z}_q^t$, and returning $(\mathbf{R}^T(\boldsymbol{d}|\boldsymbol{a}), b + \langle \boldsymbol{s}', \boldsymbol{d} \rangle)$.

Given uniform samples, the reduction outputs uniform samples up to statistical distance 2^{-n+t+1}. Given samples from $A_{q,\alpha,\boldsymbol{s}}$, the reduction outputs t samples from $A_{q,\{0\},\boldsymbol{s}''}$ and the remaining samples from $A_{q,\alpha,\boldsymbol{s}''}$ up to statistical distance 2^{-n+t+1}, with $\boldsymbol{s}'' = \mathbf{R}^{-1} \cdot (\boldsymbol{s}'|\boldsymbol{s})^T \bmod q$. This proves correctness since $\mathbf{R}$ induces a bijection on $\mathbb{Z}_q^n$. □