# Practical Fully Secure Unrestricted Inner Product Functional Encryption modulo p

Guilhem Castagnos, Fabien Laguillaumie, Ida Tucker

# Practical Fully Secure Unrestricted Inner Product Functional Encryption modulo $p$

Guilhem Castagnos[1], Fabien Laguillaumie[2], and Ida Tucker[2]

[1] Université de Bordeaux, INRIA, CNRS, IMB UMR 5251, F-33405 Talence, France.
[2] Univ Lyon, CNRS, Université Claude Bernard Lyon 1, ENS de Lyon, INRIA, LIP UMR 5668, F-69007, LYON Cedex 07, France.

**Abstract.** Functional encryption is a modern public-key cryptographic primitive allowing an encryptor to finely control the information revealed to recipients from a given ciphertext. Abdalla, Bourse, De Caro, and Pointcheval (PKC 2015) were the first to consider functional encryption restricted to the class of linear functions, i.e. inner products. Though their schemes are only secure in the selective model, Agrawal, Libert, and Stehlé (CRYPTO 16) soon provided adaptively secure schemes for the same functionality. These constructions, which rely on standard assumptions such as the Decision Diffie-Hellman (DDH), the Learning-with-Errors (LWE), and Paillier's Decision Composite Residuosity (DCR) problems, do however suffer of various practical drawbacks. Namely, the DCR based scheme only computes inner products modulo an RSA integer which is oversized for many practical applications, while the computation of inner products modulo a prime $p$ either requires, for their (DDH) based scheme, that the inner product be contained in a sufficiently small interval for decryption to be efficient, or, as in the LWE based scheme, suffers of poor efficiency due to impractical parameters.
In this paper, we provide adaptively secure functional encryption schemes for the inner product functionality which are both efficient and allow for the evaluation of unbounded inner products modulo a prime $p$. Our constructions rely on new natural cryptographic assumptions in a cyclic group containing a subgroup where the discrete logarithm (DL) problem is easy which extend Castagnos and Laguillaumie's assumption (RSA 2015) of a DDH group with an easy DL subgroup. Instantiating our generic construction using class groups of imaginary quadratic fields gives rise to the most efficient functional encryption for inner products modulo an arbitrary large prime $p$. One of our schemes outperforms the DCR variant of Agrawal et al.'s protocols in terms of size of keys and ciphertexts by factors varying between 2 and 20 for a 112-bit security.

**Keywords :** Inner Product Functional Encryption, Adaptive Security, Diffie-Hellman Assumptions.

## 1 Introduction

Traditional public key encryption (PKE) provides an all-or-nothing approach to data access. This somewhat restricting property implies that a receiver can

either recover the entire message with the appropriate secret key, or learns nothing about the encrypted message. In many real life applications however, the encryptor may wish for a more subtle encryption primitive, allowing him to disclose distinct and restricted information on the encrypted data according to the receivers privileges. For instance, consider a cloud-based email service where users may want the cloud to perform spam filtering on their encrypted emails but learn nothing more about the contents of these emails. Here the user only wants the cloud to learn one bit indicating whether or not the message is spam, but nothing more.

Functional encryption (FE) [BSW11,O'N10] emerged from a series of refinements of PKE, starting with identity based encryption [Sha84], which was later extended to fuzzy identity-based encryption by Sahai and Waters [SW05]. This work also introduced attribute-based encryption, where a message is encrypted for all users that have a certain set of attributes. FE encompasses all three of these primitives, and goes further still, as it allows not only to devise policies regulating which users can decrypt, but also provides control over which piece or function of the data each user can recover. Specifically, FE allows for a receiver to recover a function $f(y)$ of the encrypted message $y$, without learning anything else about $y$. The primitive requires a trusted authority, which possesses a master secret key $msk$, to deliver secret keys $sk_{f_i}$ – associated to specific functionalities $f_i$ – to the appropriate recipients. The encryptor computes a single ciphertext associated to the plaintext $c = \mathsf{Encrypt}(y)$, from which any user, given a decryption key $sk_{f_i}$, can recover $f_i(y) = \mathsf{Decrypt}(sk_i, c)$.

There exist two main security definitions for FE, indistinguishability-based and a stronger simulation-based security. The former – which is the model we adopt throughout this paper – requires that no efficient adversary, having chosen plaintext messages $y_0$ and $y_1$, can guess, given the encryption of one of these, which is the underlying message with probability significantly greater than $1/2$. The adversary can query a key derivation oracle for functionalities $f$, with the restriction that $f(y_0) = f(y_1)$, otherwise one could trivially tell apart both ciphertexts. Though constructions for general FE have been put forth, these schemes are far from practical, and only allow for the adversary to request an a priori bounded number of secret keys [GKP$^+$13b,SS10], or rely on non-standard and ill-understood cryptographic assumptions such as indistinguishability obfuscation or multilinear maps [ABSV15,BGJS16,GKP$^+$13a,GVW12,Wat15,GGHZ16].

The problem thus arose of building efficient FE schemes for restricted classes of functions; such constructions could be of great use for many practical applications, while developing our understanding of FE.

*Inner Product Functional Encryption (IPFE).* The restriction of FE to linear functions, or equivalently to the inner product functionality yields many interesting applications. Among other uses, linear functions allow for the computation of weighted averages and sums which are of use for statistical analysis on encrypted data, where the statistical analysis itself has sensitive information. As mentioned by Katz, Sahai and Waters in [KSW08], another application is the evaluation of polynomials over encrypted data. Agrawal, Libert and Stehlé [ALS16, Section 6]

motivate FE for the computation of linear functions modulo a prime $p$ by demonstrating that such a scheme can be turned into a bounded collusion FE scheme for all circuits[3]. And as a final example, Agrawal, Bhattacherjee, Phan, Stehlé and Yamada provide a generic transformation from FE for linear functions to trace-and-revoke systems in [ABP+17]. Naturally as they are performing linear algebra, their transformation requires the modulus to be prime and preferably quite large (of the order of 128 or 256 bits).

The primitive can succinctly be defined as follows: plaintexts are vectors $\boldsymbol{y} \in \mathcal{R}^\ell$, where $\mathcal{R}$ is a given ring. Function specific secret keys $sk_{\boldsymbol{x}}$ are derived from vectors $\boldsymbol{x} \in \mathcal{R}^\ell$ and allow to recover $\langle \boldsymbol{y}, \boldsymbol{x} \rangle \in \mathcal{R}$ but reveal no further information about $\boldsymbol{y}$. It is worth noting that due to the linearity of inner products, if the adversary requests decryption keys derived from independent vectors $\boldsymbol{x}_i$ for $i \in \{1, \ldots, \ell\}$, it can recover $\boldsymbol{y}$ by resolving a simple system of linear equations resulting from $\langle \boldsymbol{y}, \boldsymbol{x}_i \rangle$ for $i \in \{1, \ldots, \ell\}$.

This specific line of research was initiated by Abdalla, Bourse, De Caro and Pointcheval in 2015 [ABDP15]. They provided the first IPFE schemes which rely on standard assumptions such as learning with errors (LWE) and decision Diffie Hellman (DDH). However their schemes are only secure in the *selective setting*, i.e. the adversary must commit to challenge messages before having access to the schemes' public parameters. Though of great theoretical interest, such schemes are not sufficiently secure for practical applications, indeed selective security is often considered a first step towards proving full *adaptive security*. The first fully secure schemes were put forth by Agrawal, Libert and Stehlé [ALS16] under the LWE, DDH and Paillier's Decision Composite Residuosity (DCR, *cf.* [Pai99]) assumptions. Abdalla *et al.* in [ABCP16] also put forth a generic construction achieving adaptive security and provide instantiations from the DDH, DCR and LWE assumptions. However, their instantiation from Elgamal gives the same construction as the DDH based scheme of [ALS16], and their obtained schemes from LWE are restricted to the computation of inner products over the integers $\mathbf{Z}$, and are less efficient than those of [ALS16]. Finally Benhamouda *et al.* [BBL17,Bou17] provided generic constructions from hash proof systems to both chosen plaintext and chosen ciphertext secure IPFE schemes. The resulting schemes are again restricted to the computation of inner products over the integers $\mathbf{Z}$ and the sizes of secret keys are larger than those of [ALS16] (see details at the end of the introduction).

These brilliant developments do however still suffer of practical drawbacks. Namely the computation of inner products modulo a prime $p$ are restricted, in that they require that the inner product $\langle \boldsymbol{y}, \boldsymbol{x} \rangle$ be small for decryption to be efficient (as is the case for the schemes of [ABDP15], [ABCP16], and the DDH based scheme of [ALS16]). To our knowledge, the only scheme that allows for decryption of inner products of any size modulo a prime $p$ is the LWE based scheme of [ALS16], which suffers of poor efficiency since the modulus should

[3] We note however that this application of linear FE modulo a prime $p$ can not be instantiated with our schemes, as we require $p$ to be at least a 112-bit prime, whereas this application typically calls for small values of $p$ (e.g. $p = 2$).

be exponentially large in the dimension of encrypted vectors while the size of ciphertexts is cubic in this dimension.

*Our Contributions.* In this paper we put forth IPFE schemes which resolve the aforementioned issue. Our constructions allow for inner products over the integers and modulo a prime integer $p$, and rely on novel cryptographic assumptions defined in Subsection 3.1. These are variants of the [CL15] assumption, which supposes the existence of a DDH *group with an easy* DL *subgroup*: a cyclic group $G = \langle g \rangle$ where the DDH assumption holds together with a subgroup $F = \langle f \rangle$ of $G$ where the discrete logarithm problem is easy. For ease of notation we will hereafter simply refer to this assumption as the DDH assumption.

The first assumption we introduce relies on a *hard subgroup membership* (HSM) problem (according to Gjøsteen's terminology [Gjø05]), in order to somewhat generalise Paillier's DCR assumption, which follows on a long line of assumptions of distinguishing powers in $\mathbf{Z}/N\mathbf{Z}$. Known attacks for these require computing the groups' order which reduces to factoring $N$. In the [CL15] framework, the group $G$ is cyclic of order $ps$ where $s$ is unknown and $\gcd(p, s) = 1$. We denote $G^p = \langle g_p \rangle$ the subgroup of $p-$th powers in $G$. In this setting one has $G = F \times G^p$. The assumption is that it is hard to distinguish the elements of $G^p$ in $G$.

We then define the DDH-f assumption, which is *weaker* than both the DDH assumption of [CL15], and the aforementioned HSM assumption. Denoting $\mathcal{D}$ a distribution statistically close to the uniform distribution modulo $ps$, this assumption states that it is hard to distinguish distributions $\{(g^x, g^y, g^{xy}), x, y \hookleftarrow \mathcal{D}\}$ (i.e. Diffie-Hellman triplets in $G$) and $\{(g^x, g^y, g^{xy} f^u), x, y \hookleftarrow \mathcal{D}, u \hookleftarrow \mathbf{Z}/p\mathbf{Z}\}$.

We prove that this assumption is actually *equivalent* to the semantic security of the generic CL homomorphic encryption scheme of [CL15], an Elgamal variant in $G$ where the messages are encoded in the exponent in the subgroup $F$. In fact, the DDH-f assumption is better suited to mask an element of $F$, thus providing clearer proofs.

These new assumptions allow us to construct generic, linearly homomorphic encryption schemes over $\mathbf{Z}/p\mathbf{Z}$ which are semantically secure under chosen plaintext attacks (ind-cpa), which we call HSM-CL and Modified CL (*cf.* Section 3.2). The reductions between their semantic security and the underlying assumptions are given in Fig. 1, where $A \to B$ indicates that assumption $B$ holds if assumption $A$ holds, i.e. $A$ is a stronger assumption than $B$.

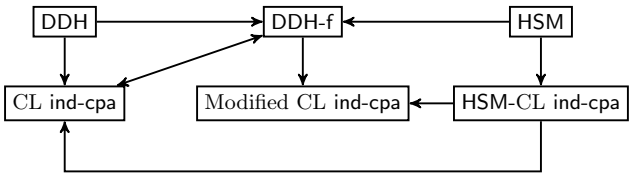


Fig. 1: Reductions between assumptions and ind-cpa security of CL variants

We then use the homomorphic properties of the above schemes to construct generic IPFE schemes over the integers and over $\mathbf{Z}/p\mathbf{Z}$, both from the weaker DDH-f assumption in Section 4, and from the HSM assumption in Section 5, somewhat generalising the scheme based on DCR of [ALS16]. Since the inner product is encoded in the exponent in the subgroup $F$, it can efficiently be recovered, whatever its size. We thereby present the first IPFE schemes which are both efficient and recover $\langle \boldsymbol{y}, \boldsymbol{x} \rangle \mod p$ whatever its size.

Our security proofs for the HSM based schemes follow a similar logic to those of [ALS16], analysing the entropy loss that occurs via queried keys, and demonstrating that there is enough residual entropy left for the challenge ciphertext to appear uniform to the adversary. However, significant difficulties occur for the schemes arising from the weaker DDH-f assumption. As in the DDH based scheme of [ALS16], we use a variant of Elgamal *à la* Cramer-Shoup. But unlike previous uses of this approach, the order of our group is unknown *and* may have small factors, so with constant probability an element may not be a generator. This calls for various subtleties: any element of the group can not be masked, however, if $p$ is large enough, elements of the subgroup $F$ of order $p$ can be.

Moreover, in order to handle private key queries, instead of computing the global distribution of the keys given this information, we carefully simplify the description of the adversary's view, since merely restricting the adversary's view modulo $p$ could potentially result in a loss of information.

We note that for our schemes over $\mathbf{Z}/p\mathbf{Z}$, vectors $\boldsymbol{x}_i$ from which keys are derived are in $\mathbf{Z}/p\mathbf{Z}$, whereas decryption keys are computed in $\mathbf{Z}$, so a lift of the $\boldsymbol{x}_i$ in $\mathbf{Z}$ must be done. Since lifting does not preserve linear dependencies, it is essential (as in [ALS16]) the key generation algorithm be stateful to lift vectors while maintaining linear dependencies. Without this restriction an adversary could learn a combination of the master key components which is singular modulo $p$ but invertible over $\mathbf{Z}$, thus revealing the whole master key.

To instantiate our generic constructions we use class groups of imaginary quadratic fields. Although the devastating attack from [CL09] eliminates a whole family of protocols built from such groups, this attack applies to schemes whose security is based on factoring a discriminant while here this factorisation is public. Moreover [CL15] showed that designing with care discrete logarithm based cryptosystems within such groups is still possible and allows for efficient and versatile protocols (*Encryption switching protocols* for instance, *cf.* [CIL17]). The problem of computing a discrete logarithm in class groups of imaginary quadratic fields has been extensively studied since the 80's, and the complexity of best known subexponential algorithms is[4] $\mathcal{O}(L_{1/2})$ (*cf.* [BJS10]) as opposed to $\mathcal{O}(L_{1/3})$ (*cf.* [Adl94]) for the discrete logarithm problem in finite field or factoring. In particular this implies that our keys can be chosen shorter and corroborates the above claim that the assumptions on which we rely are indeed weak.

[4] $L_\alpha$ is a shortcut to denote $L_{\alpha,c}(x) = \exp(c \log(x)^\alpha \log(\log(x))^{1-\alpha})$

In terms of efficiency, we show in Section 6 that for a security parameter of $\lambda = 112$ we outperform Paillier's variant of [ALS16] on all possible sizes by factors varying between 2 and 20.

*Relation to Hash Proof Systems.* Hash proof systems (HPS) were introduced in [CS02] as a generalisation of the techniques used in [CS98]. Consider a set of words $\mathcal{X}$, an NP language $\mathcal{L} \subset \mathcal{X}$ such that $\mathcal{L} = \{x \in \mathcal{X} \mid w : (x, w) \in \mathsf{R}\}$ where $\mathsf{R}$ is the relation defining the language, $\mathcal{L}$ is the language of true statements in $\mathcal{X}$, and for $(x, w) \in \mathsf{R}$, $w$ is a witness for $x \in \mathcal{L}$. A HPS defines a key generation algorithm $\mathsf{KeyGen}$ which outputs a secret hashing key $\mathsf{hk}$ and a public projection key $\mathsf{hp}$ such that $\mathsf{hk}$ defines a hash function $\mathcal{H}_{\mathsf{hk}} : \mathcal{X} \mapsto \Pi$, and $\mathsf{hp}$ allows for the (public) evaluation of the hash function on words $x \in \mathcal{L}$, i.e. $\mathcal{H}_{\mathsf{hp}}(x, w) = \mathcal{H}_{\mathsf{hk}}(x)$ for $(x, w) \in \mathsf{R}$. The *smoothness* property requires that for any $x \notin \mathcal{L}$, the value $\mathcal{H}_{\mathsf{hk}}(x)$ be uniformly distributed knowing $\mathsf{hp}$.

The $\mathsf{DDH}$ and $\mathsf{DCR}$ assumptions can be used to instantiate smooth HPS's where the languages $\mathcal{L} \subset \mathcal{X}$ define hard subset membership problems. Such HPS's, endowed with homomorphic properties over the key space, underly the IPFE schemes of [ALS16]. In fact Benhamouda, Bourse, and Lipmaa in [BBL17], and Bourse, in his thesis [Bou17], present a generic construction from a key homomorphic HPS (with a few other required properties) to an IPFE scheme in $\mathbf{Z}$ which is secure under chosen plaintext attacks. They instantiate it from $\mathsf{DDH}$ and from $\mathsf{DCR}$ but leave out $\mathsf{LWE}$ due to the complexity of the resulting scheme, as simpler constructions can be attained without using HPSs.

We note that though our constructions resemble the above – one can deduce new subset membership problems from the assumptions in Subsection 3.1 and associated HPS's – our proof techniques are very different to those of [Bou17]: so as to achieve adaptive security, their game challenger must *guess* the difference between challenge ciphertexts prior to generating the public/private key pair. If the hash key is *not* sampled uniformly at random from the key space (as in our constructions), then in order to maintain a level of security equivalent to that of the HPS the size of the secret keys increases substantially. Indeed, to encrypt vectors of dimension $\ell$ whose coordinates are bounded by $Y$, their proof techniques cause an additional $\ell \log(Y)$-bit term to appear in each coordinate of the secret key, whereas in our constructions over $\mathbf{Z}$, the bit length of the coordinates is independent of $\ell$. As a consequence, this approach leads to less efficient schemes.

Our goal has been to build *practical* IPFE schemes, therefore we avoid this genericity and the key blow up it entails, carefully evaluating the information leaked to the adversary by the public key, the secret key queries and by the challenge ciphertext, thus demonstrating that the challenge bit $\beta$ remains statistically hidden. This style of proof is closer to those of [ALS16], it allows us to obtain constructions for IPFE over $\mathbf{Z}$ that are substantially more efficient than those of [BBL17,Bou17], and constructions for IPFE modulo a prime $p$ that do not restrict the size of the inputs or of the resulting inner product, which are the most efficient such schemes to date.

## 2 Background

*Notations.* We denote sets by uppercase letters, vectors by bold lowercase letters, and the inner product of vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ is denoted $\langle \boldsymbol{x}, \boldsymbol{y} \rangle$. For a distribution $\mathcal{D}$, we write $d \hookleftarrow \mathcal{D}$ to refer to $d$ being sampled from $\mathcal{D}$. We overload the notation as $b \hookleftarrow B$ to say that $b$ is sampled uniformally at random in the set $B$. For an integer $x$, we denote its size by $|x|$, and by $[x]$ the set of integers $\{1, \ldots, x\}$. For any $\boldsymbol{c} \in \mathbf{R}^\ell$, real $\sigma > 0$, and $\ell$-dimensional lattice $\Lambda$, $\mathcal{D}_{\Lambda,\sigma,\boldsymbol{c}}$ will denote the usual discrete Gaussian distribution over $\Lambda$.

*Definition of Inner Product Functional Encryption.* This is a special case of functional encryption, as first formalised by Boneh, Sahai and Waters in [BSW11]. To start with, we provide the definition of a *functionality*.

**Definition 1 (Functionality).** *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 $\Sigma$ is an output space, which does not contain the special symbol $\perp$.*

In this article, we consider the inner product functionality, which means that decrypting the encryption of a vector $\mathbf{y}$ with a key associated to a vector $\mathbf{x}$ will reveal only $\langle \mathbf{x}, \mathbf{y} \rangle$. More precisely, we consider the function $F : (\mathbf{Z}/p\mathbf{Z})^\ell \times (\mathbf{Z}/p\mathbf{Z})^\ell \to \mathbf{Z}/p\mathbf{Z} \cup \{\perp\}$ such that $F(\mathbf{x}, \mathbf{y}) = \langle \mathbf{x}, \mathbf{y} \rangle$. The syntax of a functional encryption scheme is described below.

**Definition 2 (Functional encryption scheme).** *Let $\lambda$ be a positive integer. A functional encryption scheme for a functionality $F$ over $(\mathcal{K}, \mathcal{Y})$ is a tuple* $(\mathsf{Setup}, \mathsf{KeyDer}, \mathsf{Encrypt}, \mathsf{Decrypt})$ *of algorithms with the following specifications:*

- $\mathsf{Setup}$ *on input a security parameter $1^\lambda$, outputs a master key pair $(mpk, msk)$;*
- $\mathsf{KeyDer}$ *on input the master secret key $msk$ and a key $K \in \mathcal{K}$, outputs a key $sk_K$;*
- $\mathsf{Encrypt}$ *on input the master public key $mpk$ and a message $Y \in \mathcal{Y}$, outputs a ciphertext $C$;*
- $\mathsf{Decrypt}$ *takes as input the master public key $mpk$, a key $sk_K$ and a ciphertext $C$ and outputs $v \in \Sigma \cup \{\perp\}$.*

*For correctness, we require that for all $(mpk, msk) \leftarrow \mathsf{Setup}(1^\lambda)$, all keys $K \in \mathcal{K}$ and all messages $Y \in \mathcal{Y}$, if $sk_K \leftarrow \mathsf{KeyDer}(msk, K)$ and $C \leftarrow \mathsf{Encrypt}(mpk, Y)$, with overwhelming probability it holds that, if $v \leftarrow \mathsf{Decrypt}(mpk, sk_K, C)$ then $v = F(K, Y)$ whenever $F(K, Y) \neq \perp$.*

*Security.* We define below the security notion for functional encryption, which states that given the ciphertext of a message $Y$, the only information obtained from the secret key $sk_K$ is the evaluation of the function $f(K, Y)$. More precisely, no adversary can distinguish an encryption of $Y_0$ from an encryption of $Y_1$ even with the knowledge of secret keys $sk_K$ chosen adaptatively but satisfying $F(K, Y_0) = F(K, Y_1)$. The following definition is that of *adaptive* security, meaning that the adversary has access to the systems' public parameters,

and can perform a series of secret key requests *before* choosing $Y_0$ and $Y_1$. We consider an indistinguishability-based definition instead of the simulation-based security definition of Boneh, Sahai and Waters from [BSW11]. This adaptive indistinguishability notion is easier to handle, and it is also the strongest adaptive notion of security that can be achieved for numerous interesting functionalities. In particular, it has been demonstrated in [BSW11,AGVW13,BO13] that the strong simulation-based definition cannot be met in the standard model, while O'Neill showed in [O'N10] that indistinguishability-based security is equivalent to non-adaptive simulation-based security for a class of functions that includes the inner product. Moreover, De Caro *et al.* [DIJ+13] describe a method to transform an FE achieving an indistinguishability-based security notion into an FE attaining a certain simulation-based security.

**Definition 3 (Indistinguishability-based security).** *A functional encryption scheme* $\mathsf{FE} = (\mathsf{Setup}, \mathsf{KeyDer}, \mathsf{Encrypt}, \mathsf{Decrypt})$ *provides semantic security under chosen-plaintext attacks (*ind-fe-cpa*) if no PPT adversary* $\mathcal{A}$ *has non-negligible advantage* $\mathsf{Adv}_{\mathcal{A}}(\lambda)$ *as defined below, under the constraints that* $\mathcal{A}$*'s secret-key queries before and after its choice of challenge messages* $Y_0$ *and* $Y_1$ *satisfy* $F(K, Y_0) = F(K, Y_1)$ *for all* $K$ *in the set of key queries. The advantage is defined as follows:*

$$\mathsf{Adv}_{\mathcal{A}}(\lambda) = \Big| \Pr\big[\beta = \beta' : mpk, msk \leftarrow \mathsf{Setup}(1^\lambda), Y_0, Y_1 \leftarrow \mathcal{A}^{\mathsf{KeyDer}(msk,\cdot)}(mpk), \\ \beta \xleftarrow{\$} \{0,1\}, C^\star \leftarrow \mathsf{Encrypt}(mpk, Y_\beta), \beta' \leftarrow \mathcal{A}^{\mathsf{KeyDer}(msk,\cdot)}(C^\star)\big] - \frac{1}{2}\Big|.$$

*Backgound on Lattices.* We here recall some definitions and basic results about Gaussian distributions. These are useful for our security proofs, in which we need to evaluate the distribution of an inner product when one of the two vectors follow a Gaussian distribution. We also recall an important result from [GPV08] which explains the conditions for a Gaussian distribution over a lattice which is reduced modulo a sublattice to be close to a uniform distribution, which is also a crucial point of our proofs.

**Definition 4 (Gaussian Function).** *For any* $\sigma > 0$ *define the Gaussian function on* $\mathbf{R}^\ell$ *centred at* $\boldsymbol{c}$ *with parameter* $\sigma$*:*

$$\forall \boldsymbol{x} \in \mathbf{R}^\ell, \rho_{\sigma,\boldsymbol{c}}(\boldsymbol{x}) = \exp(-\pi ||\boldsymbol{x} - \boldsymbol{c}||^2/\sigma^2).$$

*If* $\sigma = 1$ *(resp.* $\boldsymbol{c} = \mathbf{0}$*), then the subscript* $\sigma$ *(resp.* $\boldsymbol{c}$*) is omitted.*

**Definition 5 (Discrete Gaussians).** *For any* $\boldsymbol{c} \in \mathbf{R}^\ell$*, real* $\sigma > 0$*, and* $\ell$*-dimensional lattice* $\Lambda$*, define the discrete Gaussian distribution over* $\Lambda$ *as:*

$$\forall \boldsymbol{x} \in \Lambda, \quad \mathcal{D}_{\Lambda,\sigma,\boldsymbol{c}}(\boldsymbol{x}) = \rho_{\sigma,\boldsymbol{c}}(\boldsymbol{x})/\rho_{\sigma,\boldsymbol{c}}(\Lambda),$$

*where* $\rho_{\sigma,\boldsymbol{c}}(\Lambda) = \sum_{\boldsymbol{x}\in\Lambda} \rho_{\sigma,\boldsymbol{c}}(\boldsymbol{x})$.

**Lemma 1.** *Let* $\boldsymbol{x} \in \mathbf{R}^\ell \setminus \{\mathbf{0}\}$, $\boldsymbol{c} \in \mathbf{R}^\ell$, $\sigma \in \mathbf{R}$ *with* $\sigma > 0$ *and* $\sigma' = \sigma/||\boldsymbol{x}||_2$, $c' = \frac{\langle \boldsymbol{c}, \boldsymbol{x} \rangle}{\langle \boldsymbol{x}, \boldsymbol{x} \rangle}$. *A random variable* $K$ *is distributed according to* $\mathcal{D}_{\mathbf{Z},\sigma',c'}$ *if and only if* $V := K\boldsymbol{x}$ *is distributed according to* $\mathcal{D}_{\boldsymbol{x}\mathbf{Z},\sigma,\boldsymbol{c}}$.

In dimension 1, Lemma 1 implies that if $x \in \mathbf{R}$, then $V = Kx$ is distributed according to $\mathcal{D}_{x\mathbf{Z},\sigma,c}$ if and only if $K$ is distributed according to $\mathcal{D}_{\mathbf{Z},\sigma/|x|,c/x}$. The following lemma allows to evaluate the distribution of the inner product resulting from a constant vector $\boldsymbol{x}$, and a vector with coordinates sampled from a Gaussian distribution over the lattice $\boldsymbol{x} \cdot \mathbf{Z}$.

**Lemma 2.** *Let* $\boldsymbol{x} \in \mathbf{R}^\ell$ *with* $\boldsymbol{x} \neq \mathbf{0}$, $\boldsymbol{c} \in \mathbf{R}^\ell$, $\sigma \in \mathbf{R}$ *with* $\sigma > 0$. *Let* $V$ *be a random variable distributed according to* $\mathcal{D}_{\boldsymbol{x}\cdot\mathbf{Z},\sigma,\boldsymbol{c}}$. *Then the random variable* $S$ *defined as* $S = \langle \boldsymbol{x}, V \rangle$ *is distributed according to* $\mathcal{D}_{||\boldsymbol{x}||_2^2 \cdot \mathbf{Z}, \sigma \cdot ||\boldsymbol{x}||_2, \langle \boldsymbol{c}, \boldsymbol{x} \rangle}$.

Proofs of Lemmas 1 and 2 are provided in Aux. Material I.

**Lemma 3 ([GPV08]).** *Let* $\Lambda'_0 \subset \Lambda_0 \subset \mathbf{R}^\ell$ *be two lattices with the same dimension. Let* $\epsilon \in (0, 1/2)$. *Then for any* $c \in \mathbf{R}^\ell$ *and any* $\sigma \geq \eta_\epsilon(\Lambda'_0)$, *the distribution* $D_{\Lambda_0,\sigma,c} \mod \Lambda'_0$ *is within statistical distance* $2\epsilon$ *from the uniform distribution over* $\Lambda_0/\Lambda'_0$. *The value* $\eta_\epsilon(\Lambda'_0)$ *is the* smoothing parameter *of the lattice* $\Lambda'_0$, *as defined in [MR04].*

## 3 Variants of CL: assumptions and ind-cpa schemes

In [CL15], Castagnos and Laguillaumie introduced the framework of a DDH *group with an easy* DL *subgroup*: a cyclic group $G$ where the DDH assumption holds together with a subgroup $F$ of $G$ where the discrete logarithm problem is easy. Within this framework, they designed a linearly homomorphic variant of Elgamal, described in Aux. Material II, and denoted CL throughout the rest of this paper. Moreover, they gave an instantiation using class groups of quadratic fields which allows the computation of linear operations modulo a prime $p$.

Their protocol is similar to the one of Bresson *et. al.* [BCP03] whose ind-cpa security relies on the DDH assumption in $(\mathbf{Z}/N^2\mathbf{Z})^\times$, where $N = pq$, using the arithmetic ideas of Paillier's encryption [Pai99]. Another encryption scheme based on Elgamal over $(\mathbf{Z}/N^2\mathbf{Z})^\times$ was proposed by Camenisch and Shoup in [CS03]. Its ind-cpa security relies on the Decision Composite Residuosity assumption (DCR), which consists in distinguishing the $N-$th powers in $(\mathbf{Z}/N^2\mathbf{Z})^\times$.

In the following subsection, we recall the framework of [CL15] and then generalise the DCR assumption to fit this framework of a DDH *group with an easy* DL *subgroup* with a hard subgroup membership problem (following [Gjø05]'s terminology). We also introduce a new DDH-like assumption which is weaker than DDH in $G$. Then, in Subsection 3.2, we give generic encryption schemes whose ind-cpa security are based on these assumptions. In particular we give a generalisation of the scheme of [CS03] in a DDH *group with an easy* DL *subgroup*, and a modification of CL *à la* Cramer-Shoup. Finally, in Subsection 3.3, we discuss the relations between these assumptions.

### 3.1 Algorithmic assumptions

To start with, we explicitly define the generator GenGroup used in the framework of a DDH group with an easy DL subgroup introduced in [CL15], with a few modifications as discussed below.

**Definition 6 (Generator for a DDH group with an easy DL subgroup).** *Let* GenGroup *be a pair of algorithms* (Gen, Solve)*. The* Gen *algorithm is a group generator which takes as inputs two parameters* $\lambda$ *and* $\mu$ *and outputs a tuple* $(p, \tilde{s}, g, f, g_p, G, F, G^p)$*. The set* $(G, \cdot)$ *is a cyclic group of order ps where* $s$ *is an integer,* $p$ *is a* $\mu$*-bit prime, and* $\gcd(p, s) = 1$*. The algorithm* Gen *only outputs an upper bound* $\tilde{s}$ *of* $s$*. The set* $G^p = \{x^p, x \in G\}$ *is the subgroup of order* $s$ *of* $G$*, and* $F$ *is the subgroup of order* $p$ *of* $G$*, so that* $G = F \times G^p$*. The algorithm* Gen *outputs* $f$*,* $g_p$ *and* $g = f \cdot g_p$ *which are respective generators of* $F$*,* $G^p$ *and* $G$*. Moreover, the* DL *problem is easy in* $F$*, which means that the* Solve *algorithm is a deterministic polynomial time algorithm that solves the discrete logarithm problem in* $F$*:*

$$\Pr\big[x = x^\star : (p, \tilde{s}, g, f, g_p, G, F, G^p) \leftarrow \mathsf{Gen}(1^\lambda, 1^\mu), x \hookleftarrow \mathbf{Z}_p, X = f^x, \\ x^\star \leftarrow \mathsf{Solve}(p, \tilde{s}, g, f, g_p, G, F, G^p, X)\big] = 1.$$

*Remark 1.* In practice the size of $s$ is chosen so that computing discrete logarithms in $G^p$ takes time $\mathcal{O}(2^\lambda)$.

We note that this definition differs slightly from the original definition of [CL15]. First, we impose $F$ to be of prime order $p$ as our agenda is to use the instantiation with class groups of quadratic fields in order to have $\mathbf{Z}/p\mathbf{Z}$ as the message space. This means that the generic constructions do not encompass the schemes built from Paillier where the message space is $\mathbf{Z}/N\mathbf{Z}$, with $N = pq$. If it is possible to use $N = pq$ as the order of $F$, the proofs have to rely on factoring assumptions to take care of the non-zero non-invertible elements of $\mathbf{Z}/N\mathbf{Z}$. As a consequence, this restriction simplifies the proofs, since an element of $\mathbf{Z}/p\mathbf{Z}$ is invertible if and only if it is non-zero.

Another modification is outputing the element $g_p$ that generates $G^p$ to define the HSM assumption below, and to set $g = f \cdot g_p$. In practice, the instantiation of [CL15] with class groups of quadratic fields already computes such an element $g_p$ and thus defines the generator $g$ of $G$. Note that this explicit definition of $g$ is only needed for the proof of Theorem 4 for the relation between the HSM and the DDH assumptions (*cf.* Def. 7, 8 and 9 of Aux. Material II respectively). A last modification is that Gen only outputs an upper bound $\tilde{s}$ of $s$ and not $n$. This is more accurate than the original definition as $n$ is not used in the applications and actually, the instantiation does not compute $n$ as it is a class number that requires sub-exponential time (with complexity $\mathcal{O}(L_{1/2})$) to be computed. This implies that in the following assumptions, exponents are sampled from distributions statistically close to uniform distributions. We discuss this in Remark 2.

Now, following Gjøsteen's terminology ([Gjø05]) we define a *hard subgroup membership* (HSM) problem, in order to somehow generalise Paillier's DCR as-

sumption: in Def. 6, one has $G = F \times G^p$. The assumption is that it is hard to distinguish the elements of $G^p$ in $G$.

**Definition 7 (HSM assumption).** *Let* $\mathsf{GenGroup} = (\mathsf{Gen}, \mathsf{Solve})$ *be a generator for* DDH *groups with an easy* DL *subgroup. Using the notations introduced in Def 6, the* HSM *assumption requires that the* HSM *problem is hard in* $G$ *even with access to the* Solve *algorithm. Let* $\mathcal{D}$ *(resp.* $\mathcal{D}_p$*) be a distribution over the integers such that the distribution* $\{g^x, x \hookleftarrow \mathcal{D}\}$ *(resp.* $\{g_p^x, x \hookleftarrow \mathcal{D}_p\}$*) is at distance less than* $2^{-\lambda}$ *from the uniform distribution in* $G$ *(resp. in* $G^p$*). Let* $\mathcal{A}$ *be an adversary for the* HSM *problem, its advantage is defined as:*

$$\mathsf{Adv}_{\mathcal{A}}^{\mathsf{HSM}}(\lambda,\mu) = \Big|2 \cdot \Pr\big[b = b^\star : (p,\tilde{s},g,f,g_p,G,F,G^p) \leftarrow \mathsf{Gen}(1^\lambda,1^\mu), \\ x \hookleftarrow \mathcal{D}, x' \hookleftarrow \mathcal{D}_p, b \hookleftarrow \{0,1\}, Z_0 = g^x, Z_1 = g_p^{x'}, \\ b^\star \leftarrow \mathcal{A}(p,\tilde{s},g,f,g_p,G,F,G^p,Z_b,\mathsf{Solve}(.))\big] - 1\Big|$$

*The* HSM *problem is said to be hard in* $G$ *if for all probabilistic polynomial time attacker* $\mathcal{A}$, $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{HSM}}(\lambda,\mu)$ *is negligible.*

*Remark 2.* In contrast to the traditional formulation of DDH or DCR, we can not sample uniformly elements in $G^p$ or $G$ either by a direct construction or using the generators and sampling uniformly exponents modulo the group order as the order $s$ (resp. $ps$) of $G^p$ (resp. of $G$) is unknown. As a result we use the upper bound $\tilde{s}$ of $s$ in order to instantiate the distributions $\mathcal{D}_p$ and $\mathcal{D}$ of the above definitions. Choosing distributions $\mathcal{D}$ and $\mathcal{D}_p$ with induced distributions statistically close to the uniform distributions in $G$ and $G^p$ allows for more flexibility in our upcoming proofs, which is of interest, since it is easy to see that the DDH and HSM assumptions do not depend on the choice of the distribution.

In practice, we will instantiate $\mathcal{D}_p$ and $\mathcal{D}$ thanks to the following lemma, whose proof is in Aux. Material III. We use folded gaussians as they provide better efficiency than folded uniforms, and allow us to apply Lemma 3 in our security proofs.

**Lemma 4.** *The distributions* $\mathcal{D}_p$ *and* $\mathcal{D}$ *can be implemented from the output of* Gen *as follows:*

1. *One can choose* $\mathcal{D}$ *to be the uniform distribution on* $\{0,\ldots,2^{\lambda-2}\cdot\tilde{s}\cdot p\}$.
2. *Alternatively, choosing* $\mathcal{D} = \mathcal{D}_{\mathbf{Z},\sigma}$ *with* $\sigma = \tilde{s}\cdot p\cdot\sqrt{\lambda}$ *allows for more efficient constructions as the sampled elements will tend to be smaller.*
3. *Likewise, one can choose* $\mathcal{D}_p = \mathcal{D}_{\mathbf{Z},\sigma'}$ *with* $\sigma' = \tilde{s}\cdot\sqrt{\lambda}$
4. *One can also, less efficiently, define* $\mathcal{D}_p = \mathcal{D}$.
5. *Conversely, one can also define* $\mathcal{D}$ *from* $\mathcal{D}_p$ *and the uniform distribution modulo* $p$*: the distribution* $\{g_p^x \cdot f^a, x \hookleftarrow \mathcal{D}_p, a \hookleftarrow \mathbf{Z}_p\}$ *is statistically close to the uniform distribution in* $G$.

Finally, we introduce a new assumption called DDH-f that we prove to be weaker than DDH. The security of our first IPFE relies on this assumption. Roughly speaking, it means that it is hard to distinguish the distributions

$$\{(g^x, g^y, g^{xy}), x, y \hookleftarrow \mathcal{D}\} \text{ and } \{(g^x, g^y, g^{xy} f^u), x, y \hookleftarrow \mathcal{D}, u \hookleftarrow \mathbf{Z}/p\mathbf{Z}\}.$$

In other words, as $g = f \cdot g_p$, we have on the left, a Diffie-Hellman triplet in $G$, and on the right, a triplet whose components in $G^p$ form a Diffie-Hellman triplet, and whose components in $F$ form a random triplet: $(f^x, f^y, f^{xy+u})$ (as noted in the previous remark, $\mathcal{D}$ induces distributions statistically close to the uniform in $G^p$ and $F$).

We will see in the next subsection that the security of the CL encryption scheme is actually *equivalent* to this assumption and that this assumption is *weaker* than the DDH assumption and the HSM assumption (see Theorem 4). As a side effect, using this assumption will simplify the forthcoming proofs as it is tightly related to the ind-cpa security of the underlying encryption scheme.

We note that DDH-f can be seen as an instance of the Extended-DDH (EDDH) problem as defined by Hemenway and Ostrovsky in [HO12]. They demonstrate that QR and DCR imply two different instantiations of EDDH, our implication from HSM to DDH-f somewhat generalises their proof since DDH-f is a more generic assumption than either of the hardness assumptions obtained from their reductions.

**Definition 8 (DDH-f assumption).** *Let* $\mathsf{GenGroup} = (\mathsf{Gen}, \mathsf{Solve})$ *be a generator for* DDH *groups with an easy* DL *subgroup. Using the notations of Def 6, the* DDH-f *assumption requires that the* DDH-f *problem is hard in* $G$ *even with access to the* Solve *algorithm. Let* $\mathcal{D}$ *be a distribution over the integers such that the distribution* $\{g^x, x \hookleftarrow \mathcal{D}\}$ *is at distance less than* $2^{-\lambda}$ *from the uniform distribution in* $G$*. Let* $\mathcal{A}$ *be an adversary for the* DDH-f *problem, its advantage is defined as:*

$$\mathsf{Adv}^{\mathsf{DDH\text{-}f}}_{\mathcal{A}}(\lambda, \mu) = \Big| 2 \cdot \Pr\big[b = b^\star : (p, \tilde{s}, g, f, g_p, G, F, G^p) \leftarrow \mathsf{Gen}(1^\lambda, 1^\mu),$$
$$x, y \hookleftarrow \mathcal{D}, u \hookleftarrow \mathbf{Z}/p\mathbf{Z}, X = g^x, Y = g^y, b \hookleftarrow \{0,1\}, Z_0 = g^{xy}, Z_1 = g^{xy} f^u,$$
$$b^\star \leftarrow \mathcal{A}(p, \tilde{s}, g, f, g_p, G, F, G^p, X, Y, Z_b, \mathsf{Solve}(.))\big] - 1 \Big|.$$

*The* DDH-f *problem is said to be hard in* $G$ *if for all probabilistic polynomial time attacker* $\mathcal{A}$*,* $\mathsf{Adv}^{\mathsf{DDH\text{-}f}}_{\mathcal{A}}(\lambda, \mu)$ *is negligible.*

### 3.2 Some variants of the CL generic encryption scheme

**The original Castagnos-Laguillaumie encryption scheme.** Castagnos and Laguillaumie put forth in [CL15] a generic construction for a linearly homomorphic encryption scheme over $\mathbf{Z}/p\mathbf{Z}$ based on a cyclic group with a subgroup of order $p$ where the DL problem is easy, as given by the GenGroup generator of

Def. 6. Its description is provided in Fig. 1 of Aux. Material II. They prove that this scheme is ind-cpa under the DDH assumption as defined in Def. 9 of Aux. Material II. We demonstrate below that we can be more precise and prove that the security of this scheme is equivalent to the DDH-f assumption of Def. 8: the key idea is to perform a one-time pad in $F$, instead of in the whole group $G$.

**Theorem 1.** *The* CL *encryption scheme is semantically secure under chosen plaintext attacks (*ind-cpa*) if and only if the* DDH-f *assumption holds.*

*Proof (sketch).* Suppose that the DDH-f assumption holds. Let us consider the ind-cpa game, with a public key, $h = g^x$, $x \hookleftarrow \mathcal{D}$, and a challenge ciphertext $(c_1, c_2) = (g^r, f^{m_\beta} h^r)$ with $r \hookleftarrow \mathcal{D}$ and $\beta \hookleftarrow \{0,1\}$, $m_0, m_1 \in \mathbf{Z}/p\mathbf{Z}$. We can replace $(h, c_1, h^r) = (g^x, g^r, g^{xr})$ by $(g^x, g^r, g^{xr} f^u) = (g^x, g^r, h^r f^u)$ with $u \hookleftarrow \mathbf{Z}/p\mathbf{Z}$. As a result $c_2 = h^r f^{u+m_\beta}$. For the adversary, the value of $r$ modulo $n$ is fixed by $c_1 = g^r$ as $g$ is a generator, so the value of $h^r$ is fixed. As a result from $c_2$ an unbounded adversary can infer $u + m_\beta \in \mathbf{Z}/p\mathbf{Z}$ but as $u$ is uniformly distributed in $\mathbf{Z}/p\mathbf{Z}$, he will have no information on $\beta$.

Conversely, we construct an ind-cpa adversary from a distinguisher for the DDH-f problem. Choose $m_0 \in \mathbf{Z}_p$ and $m_1 := m_0 + u$ with $u \hookleftarrow \mathbf{Z}/p\mathbf{Z}$. From the public key and the challenge ciphertext, construct the triplet

$$(h, c_1, c_2/f^{m_0}) = (g^x, g^r, g^{xr} f^{m_\beta - m_0}).$$

This gives a DH triplet if and only $\beta = 0$ and the exponent of $f$ is uniformly distributed in $\mathbf{Z}/p\mathbf{Z}$ if and only $\beta = 1$. As a result, one can use the output of a distinguisher for the DDH-f problem to win the ind-cpa game. □

**A linearly homomorphic encryption scheme from HSM.** As noted in the introduction of this section, the CL scheme was inspired by the scheme of [BCP03]. We here present a slight modification to this scheme so that it relies on the HSM assumption of Def. 7 in order to somewhat generalise the approach of the Camenisch and Shoup scheme of [CS03]. This ind-cpa scheme will be the basis of the functional encryption scheme for inner product of Section 5.

*Setting the parameters.* We use the output $(p, \tilde{s}, g, f, g_p, G, F, G^p)$ of the GenGroup generator of Def. 6. We ignore the generator $g$ (which is useless here). Following Lemma 4, Item 3, we require $\sigma' > \tilde{s}\sqrt{\lambda}$ to ensure that $\{g_p^r, r \hookleftarrow \mathcal{D}_{\mathbf{Z},\sigma'}\}$ is at distance less than $2^{-\lambda}$ from the uniform distribution in $G^p$. The plaintext space is $\mathbf{Z}/p\mathbf{Z}$, where $p$ is a $\mu$ bit prime, with $\mu \geq \lambda$. The scheme is depicted in Fig. 2a and the standard proof of the following theorem is provided in Aux. Material IV for completeness.

**Theorem 2.** *The scheme described in Fig. 2a is semantically secure under chosen plaintext attacks (*ind-cpa*) under the* HSM *assumption.*

**Algorithm** $\mathsf{KeyGen}(1^\lambda, 1^\mu)$

1. $(p, \tilde{s}, f, g_p, G, F, G^p) \leftarrow \mathsf{Gen}(1^\lambda, 1^\mu)$
2. Pick $x \hookleftarrow \mathcal{D}_{\mathbf{Z},\sigma'}$ and $h = g_p^x$
3. Set $pk = (\tilde{s}, g_p, f, p, h)$
4. Set $sk = x$
5. Return $(pk, sk)$

**Algorithm** $\mathsf{Encrypt}(pk, m)$

1. Pick $r \hookleftarrow \mathcal{D}_{\mathbf{Z},\sigma'}$
2. Return $(g_p^r, f^m h^r)$

**Algorithm** $\mathsf{Decrypt}(sk, (c_1, c_2))$

1. Compute $M = c_2/c_1^x$
2. Return $\mathsf{Solve}(M)$

(a) HSM-CL

**Algorithm** $\mathsf{KeyGen}(1^\lambda, 1^\mu)$

1. $(p, \tilde{s}, g, f, G, F) \leftarrow \mathsf{Gen}(1^\lambda, 1^\mu)$
2. Pick $x, y, \alpha \hookleftarrow \mathcal{D}_{\mathbf{Z},\sigma}$
3. Compute $h = g^\alpha$
4. Compute $\eta = g^x h^y$
5. Set $pk = (g, h, \eta)$
6. Set $sk = (x, y)$
7. Return $(pk, sk)$

**Algorithm** $\mathsf{Encrypt}(pk, m)$

1. Pick $r \hookleftarrow \mathcal{D}_{\mathbf{Z},\sigma}$
2. Return $(g^r, h^r, \eta^r f^m)$

**Algorithm** $\mathsf{Decrypt}(sk, (c_1, c_2, c_3))$

1. Compute $M = c_3/(c_1^x c_2^y)$
2. Return $\mathsf{Solve}(M)$

(b) Modified CL

Fig. 2: Description of our variants of the CL encryption

**Enhanced variant of the CL encryption scheme.** We here put forth an enhanced version of the CL homomorphic encryption scheme. We modify the original CL scheme by adding a key *à la* Cramer-Shoup (*cf.* [CS98]). The security of this scheme also relies on the DDH-f assumption. This ind-cpa scheme will be the basis of the functional encryption scheme for inner product of Section 4.

This modification to the CL encryption scheme incurs some challenges: let us consider the vanilla Elgamal scheme defined over a cyclic group of prime order $q$, generated by an element $g$. The modification leading to the Cramer-Shoup encryption scheme uses a second generator $h$ and creates a key $\eta = g^x h^y$ where $x, y \hookleftarrow \mathbf{Z}/q\mathbf{Z}$. Then $\eta^r$, with $r \hookleftarrow \mathbf{Z}/q\mathbf{Z}$ is used to mask the plaintext message. In the proof under the DDH assumption, one replaces the DH triplet $(h, g^r, h^r)$ built from the public key and the ciphertext by a random triplet and proves that the mask $\eta^r$ is then uniformly distributed and acts as a one-time pad for the plaintext, even with the knowledge of $\eta$. However, the triplet $(h, g^r, h^r)$ is indeed a DH triplet, because if $h$ is a generator, $h = g^\alpha$ with $\alpha \in (\mathbf{Z}/q\mathbf{Z})^*$. As a result, $\alpha$ is almost uniformly distributed in $\mathbf{Z}/q\mathbf{Z}$ (an element $\alpha \hookleftarrow \mathbf{Z}/q\mathbf{Z}$ is such that $\alpha \neq 0$ with overwhelming probability if $q$ is large). The same happens when considering a composite group order $N'$ where $N'$ is an RSA integer as in [Luc02], for instance, under the factoring assumption.

In our case, we use the GenGroup generator of Def. 6, i.e. a cyclic group $G$ of composite order $n = ps$ generated by $g$, where $s$ is unknown and may have some small factors. As a result, a random element $h = g^\alpha$, with $\alpha \hookleftarrow \mathcal{D}_{\mathbf{Z},\sigma}$

may not be a generator with constant probability. Consequently, the padding $\eta^r$ where $r \hookleftarrow \mathcal{D}_{\mathbf{Z},\sigma}$ and $\eta = g^x h^y$, with $x, y \hookleftarrow \mathcal{D}_{\mathbf{Z},\sigma}$ may not be uniformly distributed in $G$ knowing $\eta$. However, we can still adapt the proof: we only need $\eta^r$ to act as a one-time pad in the subgroup $F = \langle f \rangle$ of $G$ of order $p$, since our plaintext message $m \in \mathbf{Z}/p\mathbf{Z}$ is encoded as $f^m \in F$. Supposing that $p$ is a $\mu$-bit prime, with $\mu \geq \lambda$ is sufficient to prove this. As the exponents are taken close to uniform modulo $n$ and $n = p \cdot s$ with $\gcd(p, s) = 1$, they behave independently and close to uniform modulo $p$ and modulo $s$. As we are interested only in what happens modulo $p$, we can ignore the behavior modulo $s$ and get ind-cpa security under the DDH-f assumption. Note that the use of this assumption instead of the stronger DDH assumption greatly simplifies the proof.

*Setting the parameters.* We use the output $(p, \tilde{s}, g, f, g_p, G, F, G^p)$ of the generator GenGroup of Def. 6. As in the original CL scheme (*cf.* Aux. Material II), we ignore the group $G^p$ and its generator (which are useless here). Following Lemma 4, Item 2, we require $\sigma > p\tilde{s}\sqrt{\lambda}$ to ensure that $\{g^r, r \hookleftarrow \mathcal{D}_{\mathbf{Z},\sigma}\}$ is at distance less than $2^{-\lambda}$ from the uniform distribution in $G$. The plaintext space is $\mathbf{Z}/p\mathbf{Z}$, where $p$ is a $\mu$ bit prime, with $\mu \geq \lambda$. The scheme is depicted in Fig. 2b.

**Theorem 3.** *The scheme described in Fig. 2b is semantically secure under chosen plaintext attacks (*ind-cpa*) under the* DDH-f *assumption.*

The proof is provided in Aux. Material V for completeness.

### 3.3 Relations between the assumptions

One can establish direct reductions from the underlying problems of the DDH, DDH-f and HSM assumptions. However it is somewhat easier to use intermediate results on the ind-cpa security of the schemes defined in the previous subsection to demonstrate these reductions.

We proved in Theorem 1 that the original CL cryptosystem is ind-cpa if and only if the DDH-f assumption holds. In [CL15], it was proven that this scheme is ind-cpa under the DDH assumption. As a result, DDH-f is a weaker assumption than DDH. Furthermore, it is easy to see that if the HSM scheme of Fig. 2a is ind-cpa then the original CL cryptosystem is ind-cpa: from a public key $h = g_p^x$, $x \hookleftarrow \mathcal{D}_{\mathbf{Z},\sigma'}$ and a ciphertext $c = (c_1, c_2) = (g_p^r, f^m \cdot h^r)$, $r \hookleftarrow \mathcal{D}_{\mathbf{Z},\sigma'}$ for the HSM scheme, one can chose $a, b \hookleftarrow \mathbf{Z}/p\mathbf{Z}$ and construct $h' = h \cdot f^a$, and the ciphertext $c' = (c'_1, c'_2) = (c_1 \cdot f^b, c_2 \cdot f^{ab})$. According to Lemma 4, Item 5, $h'$ and $c'_1$ are statistically indistinguishable from the uniform distribution in $G$. Furthermore, $h' = g_p^x f^a = g^\alpha$ where $\alpha$ is defined modulo $n$ from the Chinese remainder theorem, such that $\alpha \equiv x \pmod{s}$ and $\alpha \equiv a \pmod{p}$. Likewise, $c'_1 = g_p^r f^b = g^\beta$ for some $\beta$ defined equivalently. Finally, one has $c'_2/f^m = g_p^{xr} f^{ab} = g_p^{\alpha\beta \mod s} f^{\alpha\beta \mod p} = g^{\alpha\beta}$. As a result, $(h', c'_1, c'_2/f^m)$ is a DH triplet in $G$, so $h', c'$ are a public key and a ciphertext for $m$ for the CL cryptosystem. As a result, an ind-cpa attacker against the cryptosystem based on HSM can be built from an ind-cpa attacker against CL. Now, if the HSM

assumption holds, from Theorem 2, the HSM scheme is ind-cpa, so the CL scheme is also ind-cpa and the DDH-f assumption holds.

We can sum up these results with the following theorem (see also Fig. 1).

**Theorem 4.** *The* DDH *assumption implies the* DDH-f *assumption. Furthermore, the* HSM *assumption implies the* DDH-f *assumption.*

## 4 Inner product FE relying on the DDH-f assumption

In this section, we build an IPFE scheme from the DDH-f assumption (*cf.* Def. 8). As proven in Theorem 4, this assumption is weaker than both the DDH and the HSM assumptions and yields simple proofs as it is suited to deal with the encoding of the message into a subgroup of prime order $p$. We use the formalism of a cyclic group with an easy DL subgroup. Our approach is based on the enhanced variant of the CL scheme, described in Fig. 2b. The resulting scheme over $\mathbf{Z}/p\mathbf{Z}$ can be viewed as an adaptation of the DDH scheme of [ALS16] to this setting, thereby removing the restriction on the size of the inner product.

The proof technique somewhat differs from the general approach of [ALS16]. We start from the ind-cpa proof of the enhanced variant of CL and then deal with the information provided by the key queries. Instead of computing the global distribution of the keys given this information, in order to make the proof go through, we have to carefully simplify the description of the adversary's view. A technical point is that even if we are only interested in what happens modulo $p$, as the plaintexts are defined in $(\mathbf{Z}/p\mathbf{Z})^\ell$, we cannot restrict the adversary's view modulo $p$: this could potentially result in a loss of information, as the key queries are defined in $\mathbf{Z}$.

We first present an FE scheme for inner products over $\mathbf{Z}$ (Section 4.1) and then consider a scheme for inner products over $\mathbf{Z}/p\mathbf{Z}$ (Section 4.2).

### 4.1 DDH-f-based FE for inner product over Z

*Setting the parameters.* As in the ind-cpa scheme of Fig. 2b, we use the output $(p, \tilde{s}, g, f, g_p, G, F, G^p)$ of the GenGroup generator of Def. 6. We ignore the group $G^p$ and its generator $g_p$ (which are useless here). We require that $p$ is a $\mu$-bit prime, with $\mu \geq \lambda$.

From Lemma 4, Item 2, choosing $\sigma > \tilde{s} \cdot p \cdot \sqrt{\lambda}$ suffices to ensure that the distribution $\{g^x, x \leftarrow \mathcal{D}_{\mathbf{Z},\sigma}\}$ is at distance less than $2^{-\lambda}$ from the uniform distribution in $G$, however for security we must take a larger $\sigma > \tilde{s} \cdot p^{3/2} \cdot \sqrt{2\lambda}$ (*cf.* proof of Theorem 5). The Encrypt algorithm operates on plaintext messages $\boldsymbol{y} \in \mathbf{Z}^\ell$ and the key derivation algorithm derives keys from vectors $\mathbf{x} \in \mathbf{Z}^\ell$. Norm bounds $X$ and $Y$ are chosen such that $X, Y < (p/2\ell)^{1/2}$ so as to ensure decryption correctness. Indeed key vectors $\mathbf{x}$ and message vectors $\mathbf{y}$ are assumed to be of bounded norm $||\mathbf{x}||_\infty \leq X$ and $||\mathbf{y}||_\infty \leq Y$, respectively. The decryption algorithm recovers $\langle \mathbf{x}, \mathbf{y} \rangle \mod p$ (using a centered modulus), which is exactly $\langle \mathbf{x}, \mathbf{y} \rangle$ over the integers, thanks to the Cauchy–Schwarz inequality and the norm bounds, since $X \cdot Y < p/2\ell$.

*Construction.* Fig. 3 depicts the functional encryption scheme for inner products in $\mathbf{Z}$ which is secure under the DDH-f assumption (*cf.* Def. 8).

**Algorithm** $\mathsf{Setup}(1^\lambda, 1^\mu, 1^\ell)$

1. $(p, \tilde{s}, g, f, G, F) \leftarrow \mathsf{Gen}(1^\lambda, 1^\mu)$
2. Pick $\alpha \hookleftarrow \mathcal{D}_{\mathbf{Z},\sigma}$
3. Compute $h = g^\alpha$
4. Pick $\boldsymbol{s}, \boldsymbol{t} \hookleftarrow \mathcal{D}_{\mathbf{Z}^\ell,\sigma}$
5. For $1 \leq i \leq \ell$ :
6. Compute $h_i = g^{s_i} h^{t_i}$
7. Return $msk = (\boldsymbol{s}, \boldsymbol{t})$ and $mpk = (\tilde{s}, g, h, f, p, \{h_i\}_{i\in[\ell]})$

**Algorithm** $\mathsf{KeyDer}(msk, \boldsymbol{x})$

1. Compute in $\mathbf{Z}$: $sk_{\mathbf{x}} = (s_{\mathbf{x}}, t_{\mathbf{x}}) = (\langle \mathbf{x}, \mathbf{s}\rangle, \langle \mathbf{x}, \mathbf{t}\rangle)$
2. Return $sk_{\boldsymbol{x}} = (s_{\mathbf{x}}, t_{\mathbf{x}})$

**Algorithm** $\mathsf{Encrypt}(mpk, \boldsymbol{y})$

1. Pick $r \hookleftarrow \mathcal{D}_{\mathbf{Z},\sigma}$
2. Set $C = g^r$ and $D = h^r$
3. For $1 \leq i \leq \ell$ :
4. Compute $E_i = f^{y_i} h_i^r$
5. Return $C_{\mathbf{y}} = (C, D, \{E_i\}_{i\in[\ell]})$

**Algorithm** $\mathsf{Decrypt}(mpk, C_{\mathbf{y}}, sk_{\boldsymbol{x}})$

1. Compute $C_{\mathbf{x}} = (\prod_{i\in[\ell]} E_i^{x_i})/(C^{s_{\mathbf{x}}} \cdot D^{t_{\mathbf{x}}})$
2. $\mathsf{sol} = \mathsf{Solve}(C_{\mathbf{x}})$
3. If $\mathsf{sol} \geq p/2$ :
4. Return $(\mathsf{sol} - p)$
5. Else:
6. Return $\mathsf{sol}$

Fig. 3: FE scheme for inner product over $\mathbf{Z}$ under the DDH-f assumption.

*Correctness.* We have

$$\begin{aligned}\prod_{i\in[\ell]} E_i^{x_i}/(C^{s_{\mathbf{x}}} \cdot D^{t_{\mathbf{x}}}) &= \prod_{i\in[\ell]} (f^{y_i}(g^{s_i} \cdot h^{t_i})^r)^{x_i}/((g^r)^{\langle \mathbf{x},\mathbf{s}\rangle} \cdot (h^r)^{\langle \mathbf{x},\mathbf{t}\rangle}) \\ &= (f^{\sum_{i=1}^\ell y_i x_i})(g^{r\sum_{i=1}^\ell s_i x_i})(h^{r\sum_{i=1}^\ell t_i x_i})/(g^{r\langle \mathbf{x},\mathbf{s}\rangle} \cdot h^{r\langle \mathbf{x},\mathbf{t}\rangle}) \\ &= f^{\langle x,y\rangle}.\end{aligned}$$

Applying the $\mathsf{Solve}$ algorithm to $C_{\mathbf{x}}$ yields $\langle \boldsymbol{x}, \boldsymbol{y}\rangle \mod p$, which, thanks to the norm bounds, is either $\langle \mathbf{x}, \mathbf{y}\rangle$ or $\langle \mathbf{x}, \mathbf{y}\rangle + p$. Since the absolute value of $\langle \mathbf{x}, \mathbf{y}\rangle$ is lower than $p/2$, if $\mathsf{sol} < p/2$ then $\langle \mathbf{x}, \mathbf{y}\rangle = \mathsf{sol}$ in $\mathbf{Z}$, otherwise $\langle \mathbf{x}, \mathbf{y}\rangle = (\mathsf{sol} - p)$.

**Theorem 5.** *Under the* DDH-f *assumption, the functional encryption scheme for inner products over* $\mathbf{Z}$ *of Fig. 3 provides full security (*ind-fe-cpa*).*

*Proof.* The proof proceeds as a sequence of games, starting in Game 0 with the real ind-fe-cpa game and ending in a game where the ciphertext statistically hides the random bit $\beta$ chosen by the challenger from the adversary's point of view. The beginning of the proof is similar to the proof of Theorem 3 on ind-cpa security. Then we take into account the fact that the adversary $\mathcal{A}$ has access to a key derivation oracle. For each Game i, we denote $S_i$ the event $\beta = \beta'$.
**Game 0 ⇒ Game 1:** In Game 1 the challenger, who has access to the master secret key $msk$, computes the ciphertext using $msk$ instead of $mpk$. The resulting

**Game 1**

1. $mpk, msk \leftarrow \mathsf{Setup}(1^\lambda, 1^\mu, 1^\ell)$
2. $\mathbf{y}_0, \mathbf{y}_1 \leftarrow \mathcal{A}^{\mathsf{KeyDer}(msk,\cdot)}(mpk)$
3. Pick $\beta \hookleftarrow \{0,1\}$
4. Pick $r \hookleftarrow \mathcal{D}_{\mathbf{Z},\sigma}$
5. Compute $C = g^r, D = h^r$
6. For $1 \leq i \leq \ell$:
7. Compute $E_i = f^{y_{\beta,i}} C^{s_i} D^{t_i}$
8. $C_{\mathbf{y}} = (C, D, \{E_i\}_{i\in[\ell]})$
9. $\beta' \leftarrow \mathcal{A}^{\mathsf{KeyDer}(msk,\cdot)}(C_{\mathbf{y}})$
10. Return $(\beta = \beta')$

**Game 2**

1. $mpk, msk \leftarrow \mathsf{Setup}(1^\lambda, 1^\mu, 1^\ell)$
2. $\mathbf{y}_0, \mathbf{y}_1 \leftarrow \mathcal{A}^{\mathsf{KeyDer}(msk,\cdot)}(mpk)$
3. Pick $\beta \hookleftarrow \{0,1\}$
4. Pick $r \hookleftarrow \mathcal{D}_{\mathbf{Z},\sigma}$ and $u \hookleftarrow \mathbf{Z}/p\mathbf{Z}$
5. Compute $C = g^r, D = h^r f^u$
6. For $1 \leq i \leq \ell$:
7. Compute $E_i = f^{y_{\beta,i}} C^{s_i} D^{t_i}$
8. $C_{\mathbf{y}} = (C, D, \{E_i\}_{i\in[\ell]})$
9. $\beta' \leftarrow \mathcal{A}^{\mathsf{KeyDer}(msk,\cdot)}(C_{\mathbf{y}})$
10. Return $(\beta = \beta')$

ciphertext has exactly the same distribution therefore:

$$\Pr[S_0] = \Pr[S_1].$$

**Game 1 ⇒ Game 2:** In Game 1, the tuple $(h = g^\alpha, C = g^r, D = h^r = g^{\alpha r})$, where $\alpha, r \hookleftarrow \mathcal{D}_{\mathbf{Z},\sigma}$, is a DH triplet since choosing $\sigma > p^{3/2} \cdot \tilde{s} \cdot \sqrt{2\lambda} > p \cdot \tilde{s} \cdot \sqrt{\lambda}$ ensures that the induced distribution is at distance less than $2^{-\lambda}$ of the uniform distribution in $G$. In Game 2, the challenger samples a random $u \hookleftarrow \mathbf{Z}/p\mathbf{Z}$ and computes $D = h^r f^u$. Both games are indistinguishable under the DDH-f assumption:

$$|\Pr[S_2] - \Pr[S_1]| = \mathsf{Adv}_{\mathcal{B}}^{\mathsf{DDH\text{-}f}}(\lambda, \mu).$$

Now in Game 2 the challenge ciphertext is:

$$(C = g^r,\ D = h^r f^u,\ \{E_i = f^{y_{\beta,i}} \cdot C^{s_i} \cdot D^{t_i} = f^{y_{\beta,i}+ut_i} h_i^r\}_{i\in[\ell]}).$$

**Lemma 5.** *In Game 2 the ciphertext $(C, D, E_1, \ldots, E_\ell) \in G^{\ell+2}$ statistically hides $\beta$ such that $|Pr[S_2] - 1/2| \leq 2^{-\lambda}$.*

Intuition. Following the proof methodology of [ALS16], we first delimit the information that is leaked in the challenge ciphertext by only considering the dimension in which both potential challenge ciphertexts differ. Indeed, denoting $\mathbf{z}_\beta = \mathbf{y}_\beta + u \cdot \boldsymbol{t} \mod p$, then projecting $\mathbf{z}_\beta$ onto the subspace generated by $\mathbf{y}_0 - \mathbf{y}_1$ encapsulates all the information revealed by the challenge ciphertext.
Next, we consider the distribution of the projection of the secret key component $\mathbf{t}$ on the subspace generated by $\mathbf{y}_0 - \mathbf{y}_1$, conditionally on the adversary's view (i.e. on the information leaked by private key queries and the public key). This amounts to a distribution over a one dimensional lattice $\Lambda_0$. We then reduce this distribution modulo a sub-lattice $\Lambda_0'$ such that $\Lambda_0/\Lambda_0' \simeq \mathbf{Z}/n\mathbf{Z}$, and using Lemma 3 we demonstrate that for an appropriate choice of the standard deviation $\sigma$ (which defines $\mathcal{D}_{\mathbf{Z}^\ell,\sigma}$, from which $\boldsymbol{t}$ is sampled), the projection of $\boldsymbol{t}$ on the subspace generated by $\mathbf{y}_0 - \mathbf{y}_1$ is statistically close to the uniform distribution over $\mathbf{Z}/n\mathbf{Z}$. As a result, $\langle \mathbf{y}, \boldsymbol{t} \rangle$ modulo $p$ is also close to the uniform distribution over $\mathbf{Z}/p\mathbf{Z}$, and thus $\mathbf{y}_\beta$ (and therefore $\beta$) is statistically hidden in $\mathbf{z}_\beta$.

*Proof (Lemma 5).* The ciphertext component $C = g^r$ information theoretically reveals $r \mod n$. Furthermore, $\forall i \in [\ell]$, $E_i$ information theoretically reveals $y_{\beta,i} + ut_i \mod p$ as the value of $h_i^r$ is fixed from $C$ and the public key. Therefore the challenge ciphertext information theoretically reveals $\boldsymbol{z}_\beta = \boldsymbol{y}_\beta + u \cdot \boldsymbol{t} \mod p$.

Throughout the rest of this proof we demonstrate that $\boldsymbol{y}_\beta$ is statistically hidden mod $p$, thanks to the distribution of $\boldsymbol{t}$ conditioned on $\mathcal{A}$'s view.

We denote $\boldsymbol{x}_i$ $\mathcal{A}$'s $i$th query to the key derivation oracle. It must hold that, for all $i$, $\langle \boldsymbol{x}_i, \boldsymbol{y}_0 \rangle = \langle \boldsymbol{x}_i, \boldsymbol{y}_1 \rangle$. Let $d \neq 0$ be the gcd of the coefficients of $\boldsymbol{y}_1 - \boldsymbol{y}_0$ and define

$$\boldsymbol{y} = (y_1, \ldots, y_\ell) = 1/d \cdot (\boldsymbol{y}_1 - \boldsymbol{y}_0) \in \mathbf{Z}^\ell.$$

It holds that $\langle \boldsymbol{x}_i, \boldsymbol{y} \rangle = 0$ over $\mathbf{Z}$ for all $i$. Therefore if we consider the lattice

$$\boldsymbol{y}^\perp = \{\boldsymbol{x} \in \mathbf{Z}^\ell : \langle \boldsymbol{x}, \boldsymbol{y} \rangle = 0\},$$

all the queries $\boldsymbol{x}_i$ must belong to that lattice. W.l.o.g., we assume the $n_0$ first coordinates of $\boldsymbol{y}$ are zero (for some $n_0$), and all remaining entries are non-zero. Further, the rows of the following matrix form a basis of $\boldsymbol{y}^\perp$:

$$\mathbf{X}_{\mathsf{top}} = \left[\begin{array}{c|ccccc} \mathbf{I}_{n_0} & & & & & \\ \hline & -y_{n_0+2} & y_{n_0+1} & & & \\ & & -y_{n_0+3} & y_{n_0+2} & & \\ & & & \ddots & \ddots & \\ & & & & -y_\ell & y_{\ell-1} \end{array}\right] \in \mathbf{Z}^{(\ell-1)\times\ell}.$$

We define the matrix:

$$\mathbf{X} = \left[\frac{\mathbf{X}_{\mathsf{top}}}{\boldsymbol{y}^T}\right] \in \mathbf{Z}^{\ell\times\ell}. \tag{1}$$

We claim that $\mathbf{X}$ is invertible mod $p$. The proof – detailed in Aux. Material VI – follows the same reasoning as [ALS16, Proof of Theorem 2].

Now since $\mathbf{X}$ is invertible over $\mathbf{Z}/p\mathbf{Z}$ and does not depend on $\beta \in \{0, 1\}$, it suffices to show that $\mathbf{X} \cdot \mathbf{z}_\beta \in (\mathbf{Z}/p\mathbf{Z})^\ell$ is statistically independent of $\beta$. Moreover by construction $\mathbf{X}_{\mathsf{top}} \cdot \boldsymbol{y}_0 = \mathbf{X}_{\mathsf{top}} \cdot \boldsymbol{y}_1$ (over the integers), so $\mathbf{X}_{\mathsf{top}} \cdot \mathbf{z}_\beta$ is clearly independent of $\beta$ and we only need to worry about the last row of $\mathbf{X} \cdot \mathbf{z}_\beta \mod p$, i.e. the information about $\beta$ leaked by the challenge ciphertext is contained in:

$$\langle \boldsymbol{y}, \boldsymbol{z}_\beta \rangle = \langle \boldsymbol{y}, \boldsymbol{y}_\beta \rangle + u \cdot \langle \boldsymbol{y}, \boldsymbol{t} \rangle \mod p. \tag{2}$$

We hereafter prove that, from $\mathcal{A}$'s perspective, $\langle \boldsymbol{y}, \boldsymbol{t} \rangle$ follows a distribution statistically close to the uniform distribution mod $p$, thus proving that $\beta$ is statistically hidden: since $u$ is sampled uniformly at random from $\mathbf{Z}/p\mathbf{Z}$, $u \neq 0 \mod p$ with all but negligible probability as $p$ is a $\mu$-bit prime, with $\mu \geq \lambda$. To this end, we analyse the information that the adversary gains on $\boldsymbol{t} \mod n$. From this, we will prove that the distribution of $\langle \boldsymbol{y}, \boldsymbol{t} \rangle$ is close to uniform mod $n$, and thus, mod $p$.

As in the proof of Theorem 3, the adversary learns $\boldsymbol{z} = \boldsymbol{s} + \alpha \boldsymbol{t} \mod n$ from the public key as $\forall i \in [\ell], h_i = g^{s_i} h^{t_i}$. Knowing $\boldsymbol{z}$, the joint distribution of $(\boldsymbol{s}, \boldsymbol{t}) \mod n$ is:

$$(\boldsymbol{z} - \alpha \boldsymbol{t} \mod n, \boldsymbol{t} \mod n) \text{ where } \boldsymbol{t} \hookleftarrow \mathcal{D}_{\mathbf{Z}^\ell, \sigma}.$$

As a result, knowing $\boldsymbol{z}$ does not give more information on $\boldsymbol{t}$ mod $n$ to $\mathcal{A}$.

One may assume that through its secret key queries, the information learned by $\mathcal{A}$ is completely determined by $\mathbf{X}_{\mathsf{top}} \cdot \boldsymbol{s}$ and $\mathbf{X}_{\mathsf{top}} \cdot \boldsymbol{t} \in \mathbf{Z}^{(\ell-1)}$, as all the queried vectors $\boldsymbol{x}_i$ can be obtained as linear combinations of the rows of $\mathbf{X}_{\mathsf{top}}$.

The value of $\mathbf{X}_{\mathsf{top}} \cdot \boldsymbol{s}$ does not give $\mathcal{A}$ more information on $\boldsymbol{t}$ mod $n$ than what he obtains from $\mathbf{X}_{\mathsf{top}} \cdot \boldsymbol{t}$. Indeed the remainder of the Euclidean division of $\mathbf{X}_{\mathsf{top}} \cdot \boldsymbol{s}$ by $n$ can be deduced from $\boldsymbol{z}$ and $\mathbf{X}_{\mathsf{top}} \cdot \boldsymbol{t}$; while the quotient is independent of $\boldsymbol{t}$ mod $n$ and $\mathbf{X}_{\mathsf{top}} \cdot \boldsymbol{t}$, as $\boldsymbol{s}$ and $\boldsymbol{t}$ are sampled independently and $\boldsymbol{z}$ only brings a relation modulo $n$. It is thus sufficient to analyse the distribution of $\boldsymbol{t}$ mod $n$ knowing $\mathbf{X}_{\mathsf{top}} \cdot \boldsymbol{t}$.

Let $\boldsymbol{t}_0 \in \mathbf{Z}^\ell$ be an arbitrary vector such that $\mathbf{X}_{\mathsf{top}} \cdot \boldsymbol{t}_0 = \mathbf{X}_{\mathsf{top}} \cdot \boldsymbol{t}$. Knowing $\mathbf{X}_{\mathsf{top}} \cdot \boldsymbol{t}$, the distribution of $\boldsymbol{t}$ is $\boldsymbol{t}_0 + \mathcal{D}_{\Lambda,\sigma,-\boldsymbol{t}_0}$ where $\Lambda = \{\mathbf{t} \in \mathbf{Z}^\ell : \mathbf{X}_{\mathsf{top}} \cdot \mathbf{t} = \mathbf{0}\}$. This lattice has dimension 1 and contains $\boldsymbol{y} \cdot \mathbf{Z}$. In fact, as $\gcd(y_1, \ldots, y_\ell) = 1$, one has $\boldsymbol{y} \cdot \mathbf{Z} = \Lambda$ (there exits $\boldsymbol{y}' \in \mathbf{Z}^\ell$ such that $\Lambda = \boldsymbol{y}' \cdot \mathbf{Z}$ and $\boldsymbol{y} = \alpha \boldsymbol{y}'$ so $\alpha$ must divide $\gcd(y_1, \ldots, y_\ell) = 1$). Therefore, applying Lemma 2, we see that conditioned on $\mathbf{X}_{\mathsf{top}} \cdot \boldsymbol{t}$, $\langle \boldsymbol{y}, \boldsymbol{t} \rangle$ is distributed according to

$$\langle \boldsymbol{y}, \boldsymbol{t}_0 \rangle + \mathcal{D}_{||\boldsymbol{y}||_2^2 \mathbf{Z}, ||\boldsymbol{y}||_2 \sigma, -\langle \boldsymbol{t}_0, \boldsymbol{y} \rangle}.$$

Now consider the distribution obtained by reducing $\mathcal{D}_{||\boldsymbol{y}||_2^2 \mathbf{Z}, ||\boldsymbol{y}||_2 \sigma, -\langle \boldsymbol{t}_0, \boldsymbol{y} \rangle}$ over $\Lambda_0 = ||\boldsymbol{y}||_2^2 \cdot \mathbf{Z}$ modulo the sublattice $\Lambda_0' = n \cdot ||\boldsymbol{y}||_2^2 \cdot \mathbf{Z}$. In order to apply Lemma 3 we need $||\boldsymbol{y}||_2 \cdot \sigma > \eta_\epsilon(\Lambda_0')$, which – applying a bound on the smoothing parameter from [MR07] for $\epsilon = 2^{-\lambda-1}$ – is guaranteed by choosing

$$||\boldsymbol{y}||_2 \cdot \sigma > \lambda_1(\Lambda_0') \cdot \sqrt{\lambda}.$$

Moreover since $\lambda_1(\Lambda_0') = n \cdot ||\boldsymbol{y}||_2^2$, we require

$$||\boldsymbol{y}||_2 \cdot \sigma > p \cdot \tilde{s} \cdot ||\boldsymbol{y}||_2^2 \cdot \sqrt{\lambda},$$

thus

$$\sigma > p \cdot \tilde{s} \cdot ||\boldsymbol{y}||_2 \cdot \sqrt{\lambda}.$$

Now from the norm bounds on $\boldsymbol{y}_0$ and $\boldsymbol{y}_1$ it holds that $||\boldsymbol{y}||_2 < \sqrt{2p}$, so choosing

$$\sigma > p^{3/2} \cdot \tilde{s} \cdot \sqrt{2\lambda}$$

suffices to ensure that from $\mathcal{A}'s$ view, $\langle \boldsymbol{y}, \boldsymbol{t} \rangle$ modulo $n$ is within distance $2^{-\lambda}$ from the uniform distribution over $\Lambda_0/\Lambda_0' \simeq \mathbf{Z}/n\mathbf{Z}$. As a result, $\langle \boldsymbol{y}, \boldsymbol{t} \rangle$ modulo $p$ is also close to the uniform distribution over $\mathbf{Z}/p\mathbf{Z}$.

We have therefore demonstrated that with overwhelming probability the term $\langle \boldsymbol{y}, \boldsymbol{y}_\beta \rangle$ in eq. (2) is statistically hidden modulo $p$ and $|Pr[S_2] - 1/2| \leq 2^{-\lambda}$.

Combining the different transition probabilities provides a bound for $\mathcal{A}$'s advantage, thus concluding the proof: $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{ind\text{-}fe\text{-}cpa}}(\lambda, \mu) \leq \mathsf{Adv}_{\mathcal{B}}^{\mathsf{DDH\text{-}f}}(\lambda, \mu) + 2^{-\lambda}$. □

### 4.2 DDH-f-based FE for inner product over $\mathbf{Z}/p\mathbf{Z}$

As in the LWE and Paillier-based IPFE modulo $p$ put forth in [ALS16], the main problem encountered here is that private key queries are performed over the integers. An adversary may therefore query keys for vectors that are linearly dependant over $(\mathbf{Z}/p\mathbf{Z})^\ell$ but independent over $\mathbf{Z}^\ell$. To solve this issue we require as in [ALS16] that the authority distributing private keys keeps track of all previously revealed private keys.

**Setting the parameters.** As in the previous construction, we use the output $(p, \tilde{s}, f, g_p, G, F, G^p)$ of the GenGroup generator of Def. 6, with $p$ a $\mu$ bit prime, and with $\mu \geq \lambda$. We sample the coordinates of the secret key from $\mathcal{D}_{\mathbf{Z}^\ell,\sigma}$. Choosing $\sigma > \tilde{s} \cdot p^\ell \cdot \sqrt{\lambda} \cdot (\sqrt{\ell})^{\ell-1}$ suffices for security to hold (*cf.* proof of Theorem 6), and ensures the distribution $\{g^x, x \leftarrow \mathcal{D}_{\mathbf{Z},\sigma}\}$ is at distance less than $2^{-\lambda}$ from the uniform distribution in $G$ (*cf.* Lemma 4, Item 2). The Encrypt algorithm encrypts plaintexts $\boldsymbol{y} \in (\mathbf{Z}/p\mathbf{Z})^\ell$ and the key derivation algorithm derives keys from vectors $\mathbf{x} \in (\mathbf{Z}/p\mathbf{Z})^\ell$.

**Construction.** Algorithms Setup and Encrypt proceed exactly as in the construction for inner products over $\mathbf{Z}$ under DDH-f (*cf.* Fig. 3). Algorithms KeyDer and Decrypt, which differ from those of the previous construction, are defined in Fig. 4. Again, correctness follows from the linearity of the inner product.

**Algorithm** $\mathsf{KeyDer}(msk, \boldsymbol{x}, st)$

Answering the $j^{th}$ key request $sk_{\boldsymbol{x}}$ where $\boldsymbol{x} \in (\mathbf{Z}/p\mathbf{Z})^\ell$. At any time the internal state $st$ contains at most $\ell$ tuples $(\boldsymbol{x}_i, \overline{\mathbf{x}}_i, z_{\mathbf{x}_i})$ where $(\overline{\mathbf{x}}_i, z_{\mathbf{x}_i})$ are previously queried secret keys and the $\boldsymbol{x}_i$'s are corresponding vectors.

1. If $\boldsymbol{x}$ is linearly independent of the $\boldsymbol{x}_i$'s modulo $p$ :
2. Set $\overline{\mathbf{x}} \in \{0, \ldots, p-1\}^\ell$ with $\overline{\mathbf{x}} = \boldsymbol{x} \mod p$
3. $z_{\mathbf{x}} = (s_{\mathbf{x}}, t_{\mathbf{x}}) = (\langle \overline{\mathbf{x}}, \boldsymbol{s} \rangle, \langle \overline{\mathbf{x}}, \mathbf{t} \rangle) \in \mathbf{Z} \times \mathbf{Z}$
4. $st = (st, (\boldsymbol{x}, \overline{\mathbf{x}}, z_{\mathbf{x}}))$
5. If $\exists \{k_i\}_{1 \leq i \leq j-1} \in \mathbf{Z}^{j-1}$ s.t. $\boldsymbol{x} = \sum_{i=1}^{j-1} k_i \boldsymbol{x}_i \in (\mathbf{Z}/p\mathbf{Z})^\ell$ then:
6. $\overline{\mathbf{x}} = \sum_{i=1}^{j-1} k_i \overline{\mathbf{x}}_i \in \mathbf{Z}^\ell$
7. $z_{\mathbf{x}} = (\sum_{i=1}^{j-1} k_i s_{\boldsymbol{x}_i}, \sum_{i=1}^{j-1} k_i t_{\boldsymbol{x}_i}) \in \mathbf{Z} \times \mathbf{Z}$
8. Return $sk_{\boldsymbol{x}} = (\overline{\mathbf{x}}, z_{\mathbf{x}})$

**Algorithm** $\mathsf{Decrypt}(mpk, C_{\mathbf{y}}, sk_{\boldsymbol{x}})$

1. Parse $(\overline{\mathbf{x}} = (\overline{x}_1, \ldots, \overline{x}_\ell);\ z_{\mathbf{x}} = (s_{\mathbf{x}}, t_{\mathbf{x}})) = sk_{\boldsymbol{x}}$
2. Compute $C_{\mathbf{x}} = (\prod_{i \in [\ell]} E_i^{\overline{x}_i}) / (C^{s_{\mathbf{x}}} \cdot D^{t_{\mathbf{x}}})$
3. Return $\mathsf{Solve}(C_{\boldsymbol{x}})$

Fig. 4: Stateful FE scheme for inner products over $\mathbf{Z}/p\mathbf{Z}$ from DDH-f.

**Theorem 6.** *Under the* DDH-f *assumption, the functional encryption scheme for inner products over* $\mathbf{Z}/p\mathbf{Z}$ *of Fig. 4 provides full security (*ind-fe-cpa*).*

*Proof.* The proof proceeds similarly to that of Theorem 5, only we must define the matrix $\mathbf{X}_{\mathsf{top}}$ differently, as we can no longer guarantee that it is invertible modulo $p$. So we here follow the same steps as in the previous proof up until the definition of Game 2. The only difference being that the adversary $\mathcal{A}$ queries the *stateful* key derivation algorithm. We denote Game $i'$ the variant of Game $i$ in which the key derivation algorithm is stateful. From the proof of Theorem 5, it holds that $|\Pr[S_2'] - \Pr[S_0']| = \mathsf{Adv}_{\mathcal{B}}^{\mathsf{DDH\text{-}f}}(\lambda, \mu)$.

As in the original Game 2, here in Game $2'$ the challenge ciphertext information theoretically reveals $\boldsymbol{z}_\beta = \boldsymbol{y}_\beta + u \cdot \boldsymbol{t} \mod p$

We define $\boldsymbol{y} = (y_1, \ldots, y_\ell) = \boldsymbol{y}_1 - \boldsymbol{y}_0 \in (\mathbf{Z}/p\mathbf{Z})^\ell$; and, assuming $\mathcal{A}$ has performed $j$ private key queries, for $1 \leq i \leq j$, we denote $\boldsymbol{x}_i \in (\mathbf{Z}/p\mathbf{Z})^\ell$ the vectors for which keys have been derived.

We want to demonstrate that from $\mathcal{A}$'s view, the bit $\beta$ is statistically hidden in Game $2'$. However we cannot use the same matrix $\mathbf{X}_{\mathsf{top}}$ as in the proof of Theorem 5; indeed, if we define $\boldsymbol{X}$ as in eq. (1) we cannot guarantee that $\boldsymbol{X}$ is invertible modulo $p$, since $\det(\boldsymbol{X}\boldsymbol{X}^T)$ could be a multiple of $p$. Therefore, so as to ensure that the queried vectors $\boldsymbol{x}_i$ do not in some way depend on $\beta$, we prove via induction that after the $j$ first private key queries (where $j \in \{0, \ldots, \ell - 1\}$), $\mathcal{A}$'s view remains statistically independent of $\beta$, thus proving that the challenge ciphertext in Game $2'$ statistically hides $\beta$ such that $|Pr[S_2'] - 1/2| \leq 2^{-\lambda}$. The induction proceeds on the value of $j$.

Recall that Game 2 and Game $2'$ are identical but for the key derivation algorithm. Therefore if the adversary can make no calls to its key derivation oracle, the indistinguishability of ciphertexts in Game $2'$ follows immediately from that in Game 2, demonstrated in the proof of Theorem 5, thus the induction hypothesis holds for $j = 0$. Now consider $j \in \{0, \ldots, \ell - 1\}$. From the induction hypothesis one may assume that at this point the state $st = \{(\boldsymbol{x}_i, \overline{\mathbf{x}}_i, z_{\mathbf{x}_i})\}_{i \in [j]}$ is independent of $\beta$. Indeed if $\mathcal{A}$'s view after $j - 1$ requests is independent of $\beta$ then the $j^{\text{th}}$ request performed by $\mathcal{A}$ must be so.

W.l.o.g. one may assume that the key requests $\boldsymbol{x}_i$ performed by the adversary are linearly independent. This implies that the $\overline{\mathbf{x}}_i$'s are linearly independent modulo $p$ and generate a subspace of

$$\boldsymbol{y}^{\perp p} = \{\boldsymbol{x} \in (\mathbf{Z}/p\mathbf{Z})^\ell : \langle \boldsymbol{x}, \boldsymbol{y} \rangle = 0 \mod p\}.$$

The set $\{\overline{\mathbf{x}}_i\}_{i \in [j]}$ can be extended to a basis $\{\overline{\mathbf{x}}_i\}_{1 \leq i \leq \ell - 1}$ of $\boldsymbol{y}^{\perp p}$. We define $\mathbf{X}_{\mathsf{top}} \in \mathbf{Z}^{(\ell-1)\times\ell}$ to be the matrix whose rows are the vectors $\overline{\mathbf{x}}_i$ for $i \in [\ell - 1]$. Let $\boldsymbol{x}' \in (\mathbf{Z}/p\mathbf{Z})^\ell$ be a vector chosen deterministically, $\boldsymbol{x}' \notin \boldsymbol{y}^{\perp p}$, such that the adversary $\mathcal{A}$ can also easily compute $\boldsymbol{x}'$. We define $\mathbf{x}_{\mathsf{bot}}$ to be the canonical lift of $\boldsymbol{x}'$ over $\mathbf{Z}$, and $\mathbf{X}$ as:

$$\mathbf{X} = \left[\frac{\mathbf{X}_{\mathsf{top}}}{\mathbf{x}_{\mathsf{bot}}^T}\right] \in \mathbf{Z}^{\ell \times \ell}.$$

The matrix $\mathbf{X}$ is invertible modulo $p$, statistically independent of $\beta$ by induction and by construction, and computable by $\mathcal{A}$, thus we need only prove that $\mathbf{X} \cdot \boldsymbol{z}_\beta$ is statistically independent of $\beta$. And since $\mathbf{X}_{\mathsf{top}} \cdot (\boldsymbol{y}_1 - \boldsymbol{y}_0) = 0 \mod p$, we need

only consider

$$\langle \mathbf{x}_{\mathsf{bot}}, \boldsymbol{z}_\beta \rangle = \langle \mathbf{x}_{\mathsf{bot}}, \boldsymbol{y}_\beta \rangle + u \cdot \langle \mathbf{x}_{\mathsf{bot}}, \boldsymbol{t} \rangle \mod p.$$

We hereafter prove that, from $\mathcal{A}$'s perspective, $\langle \mathbf{x}_{\mathsf{bot}}, \boldsymbol{t} \rangle$ follows a distribution statistically close to the uniform distribution modulo $p$, thus proving that $\beta$ is statistically hidden: since $u$ is sampled uniformly at random from $\mathbf{Z}/p\mathbf{Z}$, $u \neq 0 \mod p$ with all but negligible probability as $p$ is a $\mu$ bit prime, with $\mu \geq \lambda$. To this end, we analyse the information gained by $\mathcal{A}$ on $\boldsymbol{t} \mod n$. From this, we prove that $\boldsymbol{t} \mod p$ follows a distribution statistically close to the uniform distribution over $\boldsymbol{y} \cdot \mathbf{Z}/p\mathbf{Z}$, thus proving that $\langle \mathbf{x}_{\mathsf{bot}}, \boldsymbol{t} \rangle$ follows a distribution statistically close to uniform modulo $p$.

As in the proof of Theorem 3, the adversary learns $\boldsymbol{z} := \boldsymbol{s} + \alpha \boldsymbol{t}$ modulo $n$ from the public key as $\forall i \in [\ell], h_i = g^{s_i} h^{t_i}$. Knowing $\boldsymbol{z}$, the joint distribution of $(\boldsymbol{s}, \boldsymbol{t})$ modulo $n$ is $(\boldsymbol{z} - \alpha \boldsymbol{t} \mod n, \boldsymbol{t} \mod n)$ where $\boldsymbol{t} \hookleftarrow \mathcal{D}_{\mathbf{Z}^\ell, \sigma}$.

As a result, knowing $\boldsymbol{z}$ does not give $\mathcal{A}$ more information on $\boldsymbol{t} \mod n$. Then, as in the proof of Theorem 5, private key queries give the adversary the knowledge of $\mathbf{X}_{\mathsf{top}} \cdot \boldsymbol{s}$ and $\mathbf{X}_{\mathsf{top}} \cdot \boldsymbol{t}$ in $\mathbf{Z}^{\ell-1}$. The value of $\mathbf{X}_{\mathsf{top}} \cdot \boldsymbol{s}$ does not give the adversary more information on $\boldsymbol{t}$ modulo $n$ than what he obtains from $\mathbf{X}_{\mathsf{top}} \cdot \boldsymbol{t}$. It is thus sufficient to analyse the distribution of $\boldsymbol{t}$ modulo $n$ knowing $\mathbf{X}_{\mathsf{top}} \cdot \boldsymbol{t}$.

We define $\Lambda = \{\boldsymbol{x} \in \mathbf{Z}^\ell | \mathbf{X}_{\mathsf{top}} \cdot \boldsymbol{x} = \mathbf{0} \in \mathbf{Z}^\ell\}$. This one dimensional lattice can equivalently be defined as $\Lambda = \boldsymbol{y}' \cdot \mathbf{Z}$ where $\boldsymbol{y}' = \gamma \cdot \boldsymbol{y} \mod p$ for some $\gamma \in (\mathbf{Z}/p\mathbf{Z})^*$. One should note that all the coefficients of $\boldsymbol{y}'$ are co-prime (since $\boldsymbol{y}' / \gcd(y'_1, \ldots, y'_\ell) \in \Lambda$).

Let $\boldsymbol{t}_0 \in \mathbf{Z}^\ell$ be an arbitrary vector such that $\mathbf{X}_{\mathsf{top}} \cdot \boldsymbol{t}_0 = \mathbf{X}_{\mathsf{top}} \cdot \boldsymbol{t}$. Knowing $\mathbf{X}_{\mathsf{top}} \cdot \boldsymbol{t}$, the distribution of $\boldsymbol{t}$ is $\boldsymbol{t}_0 + \mathcal{D}_{\Lambda, \sigma, -\boldsymbol{t}_0}$. Now consider the distribution obtained by reducing the distribution $\mathcal{D}_{\Lambda, \sigma, -\boldsymbol{t}_0}$ over $\Lambda$ modulo the sublattice $\Lambda' := n \cdot \Lambda$. We first bound $||\boldsymbol{y}'||_2$ so as to bound $\lambda_1(\Lambda')$. We can then apply Lemma 3 by imposing a lower bound for $\sigma$.

Since $\Lambda = \boldsymbol{y}' \cdot \mathbf{Z}$, it holds that $||\boldsymbol{y}'||_2 = \det(\Lambda)$. We define $\Lambda_{\mathsf{top}}$ as the lattice generated by the rows of $\mathbf{X}_{\mathsf{top}}$, then applying results from [Mar03] and [Ngu91], one obtains that

$$||\boldsymbol{y}'||_2 = \det(\Lambda) \leq \det(\Lambda_{\mathsf{top}}).$$

We now apply Hadamard's bound, which tells us that, since the coordinates of each $\overline{\mathbf{x}}_i$ are smaller than $p$ and since we assumed all requested $\overline{\mathbf{x}}_i$'s are linearly independent,

$$\det(\Lambda_{\mathsf{top}}) \leq \prod_{i=1}^{\ell-1} ||\overline{\mathbf{x}}_i||_2 \leq (\sqrt{\ell} p)^{\ell-1}.$$

Therefore $||\boldsymbol{y}'||_2 \leq (\sqrt{\ell} p)^{\ell-1}$, this implies

$$\lambda_1(\Lambda') \leq n \cdot (\sqrt{\ell} p)^{\ell-1} < \tilde{s} \cdot p^\ell \cdot (\sqrt{\ell})^{\ell-1}.$$

From [MR07] we know that the smoothing parameter verifies

$$\eta_\epsilon(\Lambda') \leq \sqrt{\frac{\ln(2(1 + 1/\epsilon))}{\pi}} \cdot \lambda_1(\Lambda').$$

Thus for $\epsilon = 2^{-\lambda-1}$, we have $\eta_\epsilon(\Lambda') \leq \tilde{s} \cdot p^\ell \cdot \sqrt{\lambda} \cdot (\sqrt{\ell})^{\ell-1}$. Therefore setting

$$\sigma > \tilde{s} \cdot p^\ell \cdot \sqrt{\lambda} \cdot (\sqrt{\ell})^{\ell-1}$$

and applying Lemma 3 ensures that the distribution $\mathcal{D}_{\Lambda,\sigma,-\boldsymbol{t}_0} \mod \Lambda'$, and therefore that of $\boldsymbol{t} \mod n$ is within distance $2^{-\lambda}$ from the uniform distribution over $\Lambda/\Lambda' \simeq \boldsymbol{y}' \cdot \mathbf{Z}/n\mathbf{Z}$. This entails that $\boldsymbol{t} \mod p$ is within distance $2^{-\lambda}$ from the uniform distribution over $\boldsymbol{y}' \cdot \mathbf{Z}/p\mathbf{Z} \simeq \boldsymbol{y} \cdot \mathbf{Z}/p\mathbf{Z}$ since $\boldsymbol{y}' = \gamma \cdot \boldsymbol{y} \mod p$ for some $\gamma \in (\mathbf{Z}/p\mathbf{Z})^*$.

Since by construction $\langle \mathbf{x}_{\mathsf{bot}}, \boldsymbol{y} \rangle \neq 0 \mod p$, we get that $\langle \mathbf{x}_{\mathsf{bot}}, \boldsymbol{t} \rangle$ modulo $p$ is statistically close to the uniform distribution over $\mathbf{Z}/p\mathbf{Z}$. Moreover, with overwhelming probability $u \neq 0 \mod p$, so $u \cdot \langle \mathbf{x}_{\mathsf{bot}}, \boldsymbol{t} \rangle$ statistically hides $\langle \mathbf{x}_{\mathsf{bot}}, \boldsymbol{y}_\beta \rangle$ which implies that $\langle \mathbf{x}_{\mathsf{bot}}, \boldsymbol{z}_\beta \rangle$ does not carry significant information about $\beta$, thus concluding the proof. □

## 5 Inner product FE relying on the HSM assumption

We here build IPFE schemes from the HSM assumption and the ind-cpa scheme described in Fig. 2a, using the formalism of a cyclic group with an easy DL subgroup. Our approach is inspired by, and somewhat generalises, the approach of [ALS16] with Paillier's DCR assumption (an RSA integer $N$ plays the role of $p$ in this scheme so one should invoke the factoring assumption in our proof in order to encompass this construction). We first present an FE scheme for inner products over $\mathbf{Z}$ and then consider a scheme for inner products over $\mathbf{Z}/p\mathbf{Z}$.

### 5.1 HSM-based FE for inner product over Z

*Setting the parameters.* As in the ind-cpa scheme of Fig. 2a, we use the output $(p, \tilde{s}, g, f, g_p, G, F, G^p)$ of the GenGroup generator of Def. 6. We ignore the generator $g$ (which is useless here). We require that $p$ is a $\mu$ bit prime, with $\mu \geq \lambda$. The message space and decryption key space is $\mathbf{Z}^\ell$. As in Subsection 4.1 norm bounds $X, Y < (p/2\ell)^{1/2}$ are chosen to ensure decryption correctness. Key vectors $\mathbf{x}$ and message vectors $\mathbf{y}$ are assumed to have an infinite norm bounded by $X$ and $Y$ respectively. The decryption algorithm uses a centered modulus to recover $\langle \mathbf{x}, \mathbf{y} \rangle$ over $\mathbf{Z}$. To guarantee the scheme's security we sample the coordinates of the secret key $\boldsymbol{s} = (s_1, \ldots, s_\ell)^T \hookleftarrow \mathcal{D}_{\mathbf{Z}^\ell,\sigma}$ with discrete Gaussian entries of standard deviation $\sigma > \sqrt{2\lambda} \cdot p^{3/2} \cdot \tilde{s}$. Setting $\sigma' > \tilde{s}\sqrt{\lambda}$ ensures that $\{g_p^r, r \hookleftarrow \mathcal{D}_{\mathbf{Z}^\ell,\sigma'}\}$ is at distance less than $2^{-\lambda}$ from the uniform distribution in $G^p$.

*Construction.* Fig. 5 depicts our functional encryption for inner products over $\mathbf{Z}$ construction which relies on the HSM assumption. The proof of correctness is similar to that of the DDH-f construction.

**Algorithm** $\mathsf{Setup}(1^\lambda, 1^\mu, 1^\ell, X, Y)$

1. $(p, \tilde{s}, f, g_p, G, F, G^p) \leftarrow \mathsf{Gen}(1^\lambda, 1^\mu)$
2. $\boldsymbol{s} = (s_1, \ldots, s_\ell)^T \hookleftarrow \mathcal{D}_{\mathbf{Z}^\ell, \sigma}$
3. For $1 \leq i \leq \ell$ :
4. Compute $h_i = g_p^{s_i}$
5. Return $mpk = (\tilde{s}, g_p, f, p, \{h_i\}_{i\in[\ell]})$, $msk = \boldsymbol{s}$.

**Algorithm** $\mathsf{KeyDer}(msk, \mathbf{x})$
$\mathbf{x} = (x_1, \ldots, x_\ell)^T \in \mathbf{Z}^\ell$,

1. Compute $sk_{\boldsymbol{x}} = \langle \boldsymbol{s}, \boldsymbol{x} \rangle$ over $\mathbf{Z}$.
2. Return $sk_{\boldsymbol{x}}$

**Algorithm** $\mathsf{Encrypt}(mpk, \mathbf{y})$
$\mathbf{y} = (y_1, \ldots, y_\ell)^T \in \mathbf{Z}^\ell$,

1. Pick $r \hookleftarrow \mathcal{D}_{\mathbf{Z}, \sigma'}$
2. Compute $C_0 = g_p^r$
3. For $1 \leq i \leq \ell$ :
4. Compute $C_i = f^{y_i} \cdot h_i^r$
5. Return $C_{\mathbf{y}} = (C_0, C_1, \ldots, C_\ell)$

**Algorithm** $\mathsf{Decrypt}(mpk, C_{\mathbf{y}}, sk_{\boldsymbol{x}})$

1. Compute $C_{\mathbf{x}} = \left(\prod_{i\in[\ell]} C_i^{x_i}\right) \cdot C_0^{-sk_{\boldsymbol{x}}}$
2. $\mathsf{sol} \leftarrow \mathsf{Solve}(C_{\mathbf{x}})$
3. If $\mathsf{sol} \geq p/2$ :
4. Return $(\mathsf{sol} - p)$
5. Else return $\mathsf{sol}$

Fig. 5: FE scheme for inner product over $\mathbf{Z}$ from the HSM assumption.

**Theorem 7.** *Under the* HSM *assumption, the functional encryption scheme for inner products over* $\mathbf{Z}$ *depicted in Fig. 5 provides full security (*ind-fe-cpa*).*

*Proof.* The proof proceeds as a sequence of games, starting with the real ind-fe-cpa game (Game 0) and ending in a game where the ciphertext statistically hides the random bit $\beta$ chosen by the challenger from the adversary's point of view. The beginning of the proof is similar to the proof of Theorem 2 on ind-cpa security. Then we take into account the fact that the adversary $\mathcal{A}$ has access to a key derivation oracle. For each Game $i$, we denote $S_i$ the event $\beta = \beta'$.

**Game 1**

1. $mpk, msk \leftarrow \mathsf{Setup}(1^\lambda, 1^\mu, 1^\ell, X, Y)$
2. Parse $(s_1, \ldots, s_\ell)^T = msk$
3. Parse $(\tilde{s}, g_p, f, p, \{h_i\}_{i\in[\ell]}) = mpk$
4. $\mathbf{y}_0, \mathbf{y}_1 \leftarrow \mathcal{A}^{\mathsf{KeyDer}(msk,\cdot)}(mpk)$
5. Pick $\beta \hookleftarrow \{0, 1\}$
6. Pick $r \hookleftarrow \mathcal{D}_{\mathbf{Z}, \sigma'}$
7. Compute $C_0 = g_p^r \in G^p$
8. For $1 \leq i \leq \ell$ :
9. Compute $C_i = f^{y_{\beta,i}} \cdot C_0^{s_i}$
10. $C_{\mathbf{y}} = (C_0, C_1, \ldots, C_\ell)$
11. $\beta' \leftarrow \mathcal{A}^{\mathsf{KeyDer}(msk,\cdot)}(C_{\mathbf{y}})$
12. Return $(\beta = \beta')$

**Game 2**

1. $mpk, msk \leftarrow \mathsf{Setup}(1^\lambda, 1^\mu, 1^\ell, X, Y)$
2. Parse $(s_1, \ldots, s_\ell)^T = msk$
3. Parse $(\tilde{s}, g_p, f, p, \{h_i\}_{i\in[\ell]}) = mpk$
4. $\mathbf{y}_0, \mathbf{y}_1 \leftarrow \mathcal{A}^{\mathsf{KeyDer}(msk,\cdot)}(mpk)$
5. Pick $\beta \hookleftarrow \{0, 1\}$
6. Pick $r \hookleftarrow \mathcal{D}_{\mathbf{Z}, \sigma'}$ and $a \hookleftarrow \mathbf{Z}/p\mathbf{Z}$
7. Compute $C_0 = f^a \cdot g_p^r \in G$
8. For $1 \leq i \leq \ell$ :
9. Compute $C_i = f^{y_{\beta,i}} \cdot C_0^{s_i}$
10. $C_{\mathbf{y}} = (C_0, C_1, \ldots, C_\ell)$
11. $\beta' \leftarrow \mathcal{A}^{\mathsf{KeyDer}(msk,\cdot)}(C_{\mathbf{y}})$
12. Return $(\beta = \beta')$

**Game 0 $\Rightarrow$ Game 1:** In Game 1 the challenger uses the secret key $\boldsymbol{s} = (s_1, \ldots, s_\ell)$ to compute ciphertext elements $C_i = f^{y_{\beta,i}} \cdot (g_p^r)^{s_i} = f^{y_{\beta,i}} \cdot C_0^{s_i}$. This

change does not impact the distribution of the obtained ciphertext, therefore the adversary's success probability in both games is identical: $Pr[S_0] = Pr[S_1]$.
**Game 1 $\Rightarrow$ Game 2:** In Game 1, the distribution of $C_0$ is at distance less than $2^{-\lambda}$ of the uniform distribution in the subgroup $G^p$. Thus under the $\mathsf{HSM}$ assumption, we can, in Game 2, substitute $C_0$ by $g_p^r \cdot f^a \in G$, with $r \hookleftarrow \mathcal{D}_p, a \hookleftarrow \mathbf{Z}/p\mathbf{Z}$, which, as stated in Lemma 4, Item 5, is indeed at distance less than $2^{-\lambda}$ of the uniform distribution in $G$. Therefore, $|Pr[S_2] - Pr[S_1]| \leq \mathsf{Adv}_{\mathcal{B}}^{\mathsf{HSM}}(\lambda, \mu)$.
Now in Game 2 we have, for $a \hookleftarrow \mathbf{Z}/p\mathbf{Z}$ and $r \hookleftarrow \mathcal{D}_{\mathbf{Z},\sigma'}$:

$$\begin{cases} C_0 = f^a \cdot g_p^r \\ C_i = f^{y_{\beta,i} + a \cdot s_i} \cdot h_i^r \end{cases} . \tag{3}$$

**Lemma 6.** *In Game 2 the ciphertext $C_{\boldsymbol{y}} = (C_0, C_1, \ldots, C_\ell) \in G^{\ell+1}$ statistically hides $\beta$ such that $|Pr[S_2] - 1/2| \leq 2^{-\lambda}$.*

*Proof.* Let us begin with an overview of the proof. As in proof of Lemma 5, we first delimit the information that is leaked in the challenge ciphertext by considering the dimension in which both potential challenge ciphertexts differ. Indeed, we denote $\mathbf{z}_\beta = \mathbf{y}_\beta + a\boldsymbol{s} \mod p$, then projecting $\mathbf{z}_\beta$ onto the subspace generated by $\mathbf{y}_0 - \mathbf{y}_1$ encapsulates all the information revealed by the challenge ciphertext.
Next, we consider the distribution of the projection of the secret key $\mathbf{s}$ on the subspace generated by $\mathbf{y}_0 - \mathbf{y}_1$, conditionally on the adversary's view (i.e. from the information leaked by private key queries and the public key). This amounts to a distribution over a one dimensional lattice $\Lambda_0$. We then reduce this distribution modulo a sub-lattice $\Lambda_0'$ such that $\Lambda_0/\Lambda_0' \simeq \mathbf{Z}/p\mathbf{Z}$, and using Lemma 3 one gets that choosing $\sigma > \sqrt{2\lambda} \cdot \tilde{s} \cdot p^{3/2}$ suffices to ensure that the distribution of the projection of $\boldsymbol{s}$ on the subspace generated by $\mathbf{y}_0 - \mathbf{y}_1$ is within distance $2^{-\lambda}$ from the uniform distribution over $\mathbf{Z}/p\mathbf{Z}$, and thus $\mathbf{y}_\beta$ (and therefore $\beta$) is statistically hidden in $\mathbf{z}_\beta$.

We now provide the full proof that in Game 2 the ciphertext $C_{\mathbf{y}} = (C_0, C_1, \ldots, C_\ell) \in G^{\ell+1}$ statistically hides $\beta$ such that $|Pr[S_2] - 1/2| \leq 2^{-\lambda}$.
The proof follows the approach of [ALS16, Theorem 5]. Let us first consider the information leaked to $\mathcal{A}$ via private key queries. We denote $\mathbf{x}_i \in \mathbf{Z}^\ell$ the vectors corresponding to secret key queries made by $\mathcal{A}$. As $\mathcal{A}$ is a legitimate adversary, we have $\langle \mathbf{x}_i, \mathbf{y}_0 \rangle = \langle \mathbf{x}_i, \mathbf{y}_1 \rangle$ over $\mathbf{Z}$ for each secret key query $\mathbf{x}_i$.
Thus if we let $d \neq 0$ be the gcd of the coefficients of $\mathbf{y}_1 - \mathbf{y}_0$ and define $\mathbf{y} = (y_1, \ldots, y_\ell) = 1/d \cdot (\mathbf{y}_1 - \mathbf{y}_0) \in \mathbf{Z}^\ell$, it holds that all queried vectors $\mathbf{x}_i$ must belong to

$$\boldsymbol{y}^\perp = \{\mathbf{x} \in \mathbf{Z}^\ell : \langle \mathbf{x}, \mathbf{y} \rangle = 0\}.$$

We construct matrices $\mathbf{X}_{\mathsf{top}} \in \mathbf{Z}^{(\ell-1)\times\ell}$ and $\boldsymbol{X} \in \mathbf{Z}^{\ell\times\ell}$ exactly as in the proof of Theorem 5, such that the rows of $\mathbf{X}_{\mathsf{top}}$ form a basis of $\boldsymbol{y}^\perp$, $\boldsymbol{X}$ is invertible modulo $p$, and:

$$\mathbf{X} = \left[\frac{\mathbf{X}_{\mathsf{top}}}{\mathbf{y}^T}\right].$$

One may assume that through its secret key queries, the information learned by the adversary is completely determined by $\mathbf{X}_{\mathsf{top}} \cdot \boldsymbol{s} \in \mathbf{Z}^{(\ell-1)}$, as all the queried vectors $\mathbf{x}_i$ can be obtained as linear combinations of the rows of $\mathbf{X}_{\mathsf{top}}$.

Now let us consider the information leaked from the challenge ciphertext in Game 2. We recall that it is of the form:

$$\begin{cases} C_0 = f^a \cdot g_p^r \\ C_i = f^{y_{\beta,i}+a\cdot s_i} \cdot h_i^r \end{cases} \quad \text{for } a \hookleftarrow \mathbf{Z}/p\mathbf{Z} \text{ and } r \hookleftarrow \mathcal{D}_{\mathbf{Z},\sigma'}.$$

We denote:

$$\begin{aligned} \mathbf{z}_\beta &= (y_{\beta,1} + a \cdot s_1, \ldots, y_{\beta,\ell} + a \cdot s_\ell) \in (\mathbf{Z}/p\mathbf{Z})^\ell \\ &= \mathbf{y}_\beta + a \cdot \boldsymbol{s} \mod p. \end{aligned}$$

As in proofs of Theorems 5 and 6, information theoretically, the adversary can glean:

$$\mathbf{y}^T \cdot \mathbf{z}_\beta \mod p = \langle \mathbf{y}, \mathbf{y}_\beta \rangle + a \cdot \langle \mathbf{y}, \boldsymbol{s} \rangle \mod p. \tag{4}$$

From the public key and from private key queries, the information gained by the adversary amounts at most to:

- a subset of the coordinates of $\mathbf{X}_{\mathsf{top}} \cdot \boldsymbol{s} \in \mathbf{Z}^{\ell-1}$ (from private key queries).
- the knowledge that $h_i = g_p^{s_i}$ for $1 \leq i \leq \ell$ which information-theoretically reveals $s_i$ modulo $s$ (from the public key).

Let $\boldsymbol{s}_0$ denote an arbitrary vector satisfying the same equations as the secret key $\boldsymbol{s}$ from the view of the adversary, i.e.:

$$\mathbf{X}_{\mathsf{top}} \cdot \boldsymbol{s}_0 = \mathbf{X}_{\mathsf{top}} \cdot \boldsymbol{s} \in \mathbf{Z}^{\ell-1} \quad \wedge \quad \forall i \in [\ell], h_i = g_p^{s_i} = g_p^{s_{0,i}} \in G^p.$$

Denoting $\boldsymbol{t} = (t_1, \ldots, t_\ell) = \boldsymbol{s} - \boldsymbol{s}_0$ we can rewrite the above as:

$$\mathbf{X}_{\mathsf{top}} \cdot \mathbf{t} = \mathbf{0} \in \mathbf{Z}^{\ell-1} \quad \wedge \quad \forall i \in [\ell], t_i = 0 \mod s.$$

We define $\Lambda = \{\mathbf{t} \in \mathbf{Z}^\ell | \mathbf{X}_{\mathsf{top}} \cdot \mathbf{t} = \mathbf{0}, \mathbf{t} = \mathbf{0} \mod s\} \subset \mathbf{Z}^\ell$. Since the protocol samples $\boldsymbol{s} \hookleftarrow \mathcal{D}_{\mathbf{Z}^\ell,\sigma}$, from the adversary's view $\boldsymbol{s}$ is of the form $\boldsymbol{s}_0 + T$ where $T$ is a random variable with values in $\Lambda$. The random variable $T$ follows the same probability distribution as $\boldsymbol{s} - \boldsymbol{s}_0$ but taken over $\Lambda$, i.e.:

$$\begin{aligned} \forall \boldsymbol{t} \in \Lambda, \quad \Pr[T = \boldsymbol{t}] &= \mathcal{D}_{Z^\ell,\sigma,-\boldsymbol{s}_0}(\mathbf{t}) / \mathcal{D}_{Z^\ell,\sigma,-\boldsymbol{s}_0}(\Lambda) \\ &= \frac{\rho_{\sigma,-\boldsymbol{s}_0}(\mathbf{t})}{\rho_{\sigma,-\boldsymbol{s}_0}(\mathbf{Z}^\ell)} \times \frac{\rho_{\sigma,-\boldsymbol{s}_0}(\mathbf{Z}^\ell)}{\rho_{\sigma,-\boldsymbol{s}_0}(\Lambda)} \\ &= \mathcal{D}_{\Lambda,\sigma,-\boldsymbol{s}_0}(\mathbf{t}). \end{aligned}$$

Therefore, from the adversary's point of view, the distribution of $\boldsymbol{s} \in \mathbf{Z}^\ell$ is:

$$\boldsymbol{s}_0 + \mathcal{D}_{\Lambda,\sigma,-\boldsymbol{s}_0}.$$

Let us consider the lattice $\Lambda' = \{\mathbf{t} \in \mathbf{Z}^\ell : \mathbf{X}_{\mathsf{top}} \cdot \mathbf{t} = \mathbf{0}\}$. As in the proof of Theorem 5, this lattice has dimension 1 and $\Lambda' = \boldsymbol{y} \cdot \mathbf{Z}$. Moreover

$$\Lambda = \Lambda' \cap (s \cdot \mathbf{Z}^\ell) = (\mathbf{y} \cdot \mathbf{Z}) \cap (s \cdot \mathbf{Z}^\ell) = s \cdot \mathbf{y} \cdot \mathbf{Z},$$

since $\gcd(y_1, \ldots, y_\ell) = 1$ (for any $\alpha \in \mathbf{Z}$, for $s$ to divide all $\alpha \cdot y_i$, $s$ must divide $\alpha \cdot \gcd(y_1, \ldots, y_\ell) = \alpha$).

We now consider the distribution of $\langle \boldsymbol{s}, \boldsymbol{y} \rangle$, and then reduce it modulo $p$, so as to prove that, from the adversary's view (i.e. conditionally on the public key and queried keys), in eq. (4) the bit $\beta$ is statistically hidden. Let us denote $\Lambda_0 = s \cdot ||\mathbf{y}||_2^2 \cdot \mathbf{Z}$. It follows from Lemma 2 that the distribution of $\langle \boldsymbol{s}, \boldsymbol{y} \rangle$ is:

$$\langle \boldsymbol{s}_0, \mathbf{y} \rangle + \mathcal{D}_{\Lambda_0, ||\mathbf{y}||_2 \cdot \sigma, -c}$$

where $c = \langle \boldsymbol{s}_0, \mathbf{y} \rangle$ in $\mathbf{Z}$.
In order to prove that the above distribution is statistically close to the uniform distribution over $\mathbf{Z}/p\mathbf{Z}$, we consider the distribution obtained by reducing the distribution $\mathcal{D}_{\Lambda_0, ||\mathbf{y}||_2 \cdot \sigma, -c}$ over $\Lambda_0$ modulo the sublattice $\Lambda_0' = p\Lambda_0$. Since $\Lambda_0/\Lambda_0' \simeq \mathbf{Z}/p\mathbf{Z}$, demonstrating that $\langle \mathbf{y}, \boldsymbol{s} \rangle \mod p$ is within negligible statistical distance from the uniform distribution over $\Lambda_0/\Lambda_0'$ will conclude the proof.

From Lemma 3 it follows that to achieve the required smoothing parameter $\eta_\epsilon(\Lambda_0')$ one must impose a lower bound on the standard deviation, i.e. we need $||\mathbf{y}||_2 \cdot \sigma > \eta_\epsilon(\Lambda_0')$. If we set $\epsilon$ to be $2^{-\lambda-1}$, from [MR07] we know that

$$\eta_\epsilon(\Lambda_0') \leq \sqrt{\frac{\ln(2(1+1/\epsilon))}{\pi}} \cdot \lambda_1(\Lambda_0') < \sqrt{\lambda} \cdot \lambda_1(\Lambda_0').$$

Since $\lambda_1(\Lambda_0') = s \cdot ||\mathbf{y}||_2^2 \cdot p < \tilde{s} \cdot ||\mathbf{y}||_2^2 \cdot p$, we need

$$\sigma > ||\boldsymbol{y}||_2 \sqrt{\lambda} \cdot p \cdot \tilde{s}$$

(i.e. $||\mathbf{y}||_2 \cdot \sigma > \sqrt{\lambda} \cdot \lambda_1(\Lambda_0')$). Finally as $||\mathbf{y}||_2 < \sqrt{2p}$ (since $||\mathbf{y}_i||_\infty < \sqrt{p/(2\ell)}$ for $i \in \{0, 1\}$) choosing

$$\sigma > \sqrt{2\lambda} \cdot \tilde{s} \cdot p^{3/2}$$

suffices to ensure that $\langle \mathbf{y}, \boldsymbol{s} \rangle \mod p$ is within distance $2^{-\lambda}$ from the uniform distribution over $\Lambda_0/\Lambda_0' \simeq \mathbf{Z}/p\mathbf{Z}$.

Finally, since in eq. (4), $a \hookleftarrow \mathbf{Z}/p\mathbf{Z}$ is invertible modulo $p$ with all but negligible probability, the term $\langle \mathbf{y}, \mathbf{y}_\beta \rangle \mod p$ is statistically hidden, and $|Pr[S_2] - 1/2| \leq 2^{-\lambda}$.

□

Over all game transitions, after adding up the different probabilities, we find that $\mathcal{A}$'s advantage in the real game can be bounded as $|Pr[S_0] - 1/2| \leq \mathsf{Adv}_{\mathcal{B}}^{\mathsf{HSM}}(\lambda, \mu) + 2^{-\lambda}$ which is negligible if the HSM assumption holds in $G$. □

### 5.2 HSM-based FE for inner product over $\mathbf{Z}/p\mathbf{Z}$

As in the DDH-f based scheme for inner products over $\mathbf{Z}/p\mathbf{Z}$ of Section 4.2, the key generation algorithm is stateful to ensure the adversary cannot query keys for vectors that are linearly dependant over $(\mathbf{Z}/p\mathbf{Z})^\ell$ but independent over $\mathbf{Z}^\ell$.

*Setting the parameters.* As in the previous construction, we use the output $(p, \tilde{s}, f, g_p, G, F, G^p)$ of the GenGroup generator of Def. 6, with $p$ a $\mu$-bit prime, and with $\mu \geq \lambda$. The message space and vector space from which decryption keys are derived are now $(\mathbf{Z}/p\mathbf{Z})^\ell$. Given an encryption of $\boldsymbol{y} \in (\mathbf{Z}/p\mathbf{Z})^\ell$ and a decryption key for $\boldsymbol{x} \in (\mathbf{Z}/p\mathbf{Z})^\ell$, the decryption algorithm recovers $\langle \mathbf{x}, \mathbf{y} \rangle \in \mathbf{Z}/p\mathbf{Z}$. To guarantee the scheme's security we sample the coordinates of the secret key $\boldsymbol{s}$ from $\mathcal{D}_{\mathbf{Z}^\ell,\sigma}$ with discrete Gaussian entries of standard deviation $\sigma > \sqrt{\lambda} \cdot p \cdot \tilde{s} \cdot (\sqrt{\ell}p)^{\ell-1}$. We require $\sigma' > \tilde{s}\sqrt{\lambda}$ to ensure that $\{g_p^r, r \hookleftarrow \mathcal{D}_{\mathbf{Z}^\ell,\sigma'}\}$ is at distance less than $2^{-\lambda}$ from the uniform distribution in $G^p$.

*Construction.* The Setup and Encrypt algorithms proceed exactly as in Fig. 5, the only difference being that Encrypt operates on message vectors $\boldsymbol{y} \in (\mathbf{Z}/p\mathbf{Z})^\ell$ instead of $\boldsymbol{y} \in \mathbf{Z}^\ell$. In Fig. 6 we only define algorithms KeyDer and Decrypt, since they differ from those of the previous construction.

**Algorithm** KeyDer$(msk, \boldsymbol{x}, st)$

Answering the $j^{th}$ key request $sk_{\boldsymbol{x}}$ where $\boldsymbol{x} \in (\mathbf{Z}/p\mathbf{Z})^\ell$. At any time the internal state $st$ contains at most $\ell$ tuples $(\boldsymbol{x}_i, \overline{\mathbf{x}}_i, z_{\mathbf{x}_i})$ where $(\overline{\mathbf{x}}_i, z_{\mathbf{x}_i})$ are previously queried secret keys and the $\boldsymbol{x}_i$'s are corresponding vectors.

1. If $\boldsymbol{x}$ is linearly independent of the $\boldsymbol{x}_i$'s modulo $p$ :
2. Set $\overline{\mathbf{x}} \in \{0, \ldots, p-1\}^\ell$ with $\overline{\mathbf{x}} = \boldsymbol{x} \mod p$
3. $z_{\mathbf{x}} = \langle \boldsymbol{s}, \overline{\mathbf{x}} \rangle \in \mathbf{Z}$ ; $st = (st, (\boldsymbol{x}, \overline{\mathbf{x}}, z_{\mathbf{x}}))$
4. If $\exists \{k_i\}_{1 \leq i \leq j-1} \in \mathbf{Z}^{j-1}$ such that $\boldsymbol{x} = \sum_{i=1}^{j-1} k_i \boldsymbol{x}_i \in (\mathbf{Z}/p\mathbf{Z})^\ell$ then:
5. $\overline{\mathbf{x}} = \sum_{i=1}^{j-1} k_i \overline{\mathbf{x}}_i \in \mathbf{Z}^\ell$ ; $z_{\mathbf{x}} = \sum_{i=1}^{j-1} k_i z_{\mathbf{x}_i} \in \mathbf{Z}$
6. Return $sk_{\boldsymbol{x}} = (\overline{\mathbf{x}}, z_{\mathbf{x}})$

**Algorithm** Decrypt$(mpk, C_{\mathbf{y}}, sk_{\boldsymbol{x}})$

1. Parse $(\overline{\mathbf{x}} = (\overline{x}_1, \ldots, \overline{x}_\ell), z_{\mathbf{x}}) = sk_{\boldsymbol{x}}$
2. Compute $C_{\boldsymbol{x}} = \left( \prod_{i \in [\ell]} C_i^{\overline{x}_i} \right) \cdot (C_0^{-z_{\mathbf{x}}})$
3. Return Solve$(C_{\boldsymbol{x}})$

Fig. 6: Functional encryption scheme for inner products over $\mathbf{Z}/p\mathbf{Z}$ from HSM.

**Theorem 8.** *Under the* HSM *assumption the above stateful functional encryption scheme for inner products over* $\mathbf{Z}/p\mathbf{Z}$ *provides full security (*ind-fe-cpa*).*

The proof follows the same lines as the proof of the previous theorem and is adapted from the proofs of [ALS16].

The main issue is that we can no longer guarantee that $\boldsymbol{X}$ is invertible modulo $p$. We need to compute on-the-fly a basis for $\{\boldsymbol{x} \in (\mathbf{Z}/p\mathbf{Z})^\ell : \langle \boldsymbol{x}, \boldsymbol{y} \rangle = 0 \mod p\}$ to apply the same techniques as in Theorem 7. The analysis gives significantly larger standard deviations as mentioned above due a bad approximation of the determinant of a related matrix.

*Proof.* We here provide the proof that under the HSM assumption the stateful functional encryption scheme for inner products over $\mathbf{Z}/p\mathbf{Z}$ presented above provides full security (ind-fe-cpa).

The proof proceeds similarly to that in $\mathbf{Z}$ (*cf.* Theorem 7), starting with the real ind-fe-cpa game and ending in a game where the ciphertext statistically hides the random bit $\beta$ chosen by the challenger from the adversary's point of view.

Games 0 to 2 basically proceed identically to those of the proof of Theorem 7. The only difference is in the key derivation oracle that the adversary $\mathcal{A}$ has access to, which now executes the *stateful* key derivation algorithm. Thus we have a Game $2'$ for which:

$$|\Pr[S_2'] - \Pr[S_0]| \leq \mathsf{Adv}_{\mathcal{B}}^{\mathsf{HSM}}(\lambda, \mu).$$

Recall that $\mathcal{A}$ can query the key derivation oracle for any vector $\boldsymbol{x} \in (\mathbf{Z}/p\mathbf{Z})^\ell$ satisfying $\langle \boldsymbol{x}, \boldsymbol{y}_0 \rangle = \langle \boldsymbol{x}, \boldsymbol{y}_1 \rangle \in \mathbf{Z}/p\mathbf{Z}$. For each query, $\mathcal{A}$ is given a secret key $(\overline{\mathbf{x}}, z_{\mathbf{x}})$ as in the real scheme. And in Game $2'$ we have:

$$\begin{cases} C_0 = f^a \cdot g_p^r \\ C_i = f^{y_{\beta,i} + a \cdot s_i} \cdot h_i^r, \quad \forall i \in [\ell] \end{cases}$$

where $a \hookleftarrow \mathbf{Z}/p\mathbf{Z}$ and $r \hookleftarrow \mathcal{D}_{\mathbf{Z},\sigma'}$.

Therefore, the challenge ciphertext information-theoretically reveals:

$$\boldsymbol{z}_\beta = \boldsymbol{y}_\beta + a\boldsymbol{s} \mod p.$$

We define $\boldsymbol{y} = (y_1, \ldots, y_\ell) = \boldsymbol{y}_1 - \boldsymbol{y}_0 \in (\mathbf{Z}/p\mathbf{Z})^\ell$, and, assuming $\mathcal{A}$ has performed $j$ private key queries, for $1 \leq i \leq j$, we denote $\boldsymbol{x}_i \in (\mathbf{Z}/p\mathbf{Z})^\ell$ the vectors for which keys have been derived.

From here on, demonstrating that in Game $2'$ the challenge ciphertext statistically hides the bit $\beta$ is done as in proof of Theorem 6, we prove via induction that after the $j$ first private key queries, $\mathcal{A}$'s view remains statistically independent of $\beta$, thus proving that $|Pr[S_2'] - 1/2| \leq 2^{-\lambda}$. The induction proceeds on the value of $j$.

For $j = 0$ the adversary can make no private key queries. With this restriction games 2 and $2'$ are identical. It thus follows from the proof of Theorem 7 that for $j = 0$ the induction hypothesis holds, i.e. $\mathcal{A}$'s view is indeed statistically independent of $\beta$.

Consider $j \in \{0, \ldots, \ell - 1\}$. From the induction hypothesis one may assume that at this point the state $st = \{(\boldsymbol{x}_i, \overline{\mathbf{x}}_i, z_{\mathbf{x}_i}) \in (\mathbf{Z}/p\mathbf{Z})^\ell \times \mathbf{Z}^\ell \times \mathbf{Z}\}_{i \in [j]}$ is independent of $\beta$. Indeed if $\mathcal{A}$'s view after $j - 1$ requests is independent of $\beta$ then the $j^{\text{th}}$ request performed by $\mathcal{A}$ must be so. W.l.o.g. one may assume that the key requests $\boldsymbol{x}_i$ performed by the adversary are linearly independent (otherwise $\mathcal{A}$ does not gain any additional information from its request). This implies that the $\overline{\mathbf{x}}_i$'s are linearly independent modulo $p$ and generate a subspace of:

$$\boldsymbol{y}^{\perp p} = \{\boldsymbol{x} \in (\mathbf{Z}/p\mathbf{Z})^\ell : \langle \boldsymbol{x}, \boldsymbol{y} \rangle = 0 \mod p\}$$

Moreover the set $\{\overline{\mathbf{x}}_i\}_{i\in[j]}$ (generated during private key queries) can be extended to a basis $\{\overline{\mathbf{x}}_i\}_{i\in[\ell-1]}$ of $\boldsymbol{y}^{\perp_p}$. We define $\mathbf{X}_{\mathsf{top}} \in \mathbf{Z}^{(\ell-1)\times\ell}$ to be the matrix whose rows are the vectors $\overline{\mathbf{x}}_i$ for $i \in [\ell-1]$.

$$\mathbf{X}_{\mathsf{top}} = \begin{bmatrix} \overline{\mathbf{x}}_1^T \\ \overline{\mathbf{x}}_2^T \\ \vdots \\ \overline{\mathbf{x}}_{\ell-1}^T \end{bmatrix}$$

Let $\boldsymbol{x}' \in (\mathbf{Z}/p\mathbf{Z})^\ell$ be a vector such that $\boldsymbol{x}' \notin \boldsymbol{y}^{\perp_p}$. This vector $\boldsymbol{x}'$ is constructed deterministically from the set $\{\overline{\mathbf{x}}_i\}_{i\in[j]}$ and $\boldsymbol{y}$. We define $\mathbf{x}_{\mathsf{bot}}$ to be the canonical lift of $\boldsymbol{x}'$ over $\mathbf{Z}$, and $\mathbf{X}$ as:

$$\mathbf{X} = \left[\frac{\mathbf{X}_{\mathsf{top}}}{\mathbf{x}_{\mathsf{bot}}^T}\right].$$

The matrix $\mathbf{X}$ is built deterministically, invertible modulo $p$ by construction, and independent of $\beta$ by induction hypothesis and by construction. So $\mathbf{X}$ is known to $\mathcal{A}$ in the information theoretical sense. As in the proof of Theorem 7 we need only prove that $\mathbf{X}\cdot\boldsymbol{z}_\beta$ is statistically independent of $\beta$. And since $\mathbf{X}_{\mathsf{top}}\cdot(\boldsymbol{y}_1-\boldsymbol{y}_0)=0 \mod p$, we need only consider:

$$\langle \mathbf{x}_{\mathsf{bot}}, \boldsymbol{z}_\beta\rangle = \langle \mathbf{x}_{\mathsf{bot}}, \boldsymbol{y}_\beta\rangle + a\langle \mathbf{x}_{\mathsf{bot}}, \boldsymbol{s}\rangle \mod p. \tag{5}$$

As in the proof of Theorem 7, let $\boldsymbol{s}_0$ denote an arbitrary vector such that from the adversary's point of view, the distribution of $\boldsymbol{s}\in\mathbf{Z}^\ell$ is $\boldsymbol{s}_0 + \mathcal{D}_{\Lambda,\sigma,-\boldsymbol{s}_0}$ where

$$\Lambda = \{\boldsymbol{t}\in\mathbf{Z}^\ell : \mathbf{X}_{\mathsf{top}}\cdot\boldsymbol{t}=\mathbf{0}, \boldsymbol{t}=\mathbf{0} \mod s\}.$$

We also define $\Lambda' = \{\boldsymbol{t}\in\mathbf{Z}^\ell : \mathbf{X}_{\mathsf{top}}\cdot\boldsymbol{t}=\mathbf{0}\}$, clearly $\Lambda'$ is a one-dimensional lattice which can also be defined as $\Lambda' = \boldsymbol{y}'\cdot\mathbf{Z}$ for some $\boldsymbol{y}'\in\mathbf{Z}^\ell$. One should note that all the coefficients of $\boldsymbol{y}'$ are co-prime (since $\boldsymbol{y}'/\gcd(y'_1,\ldots,y'_\ell)\in\Lambda'$). Moreover, since $\mathbf{X}_{\mathsf{top}}$ is a basis of $\boldsymbol{y}^{\perp_p}$, we have $\Lambda' \mod p = \boldsymbol{y}\cdot\mathbf{Z}/p\mathbf{Z}$. As a result, there exists $\alpha\in(\mathbf{Z}/p\mathbf{Z})^*$ s.t. $\boldsymbol{y}' = \alpha\cdot\boldsymbol{y} \mod p$. Finally, $\Lambda = \Lambda'\cap(s\mathbf{Z})^\ell$ and, since the coefficients of $\boldsymbol{y}'$ are co-prime $\Lambda = s\cdot\boldsymbol{y}'\cdot\mathbf{Z} = s\cdot\Lambda'$.

Intuition. So as to prove that in eq. (5), the term $a\cdot\langle\mathbf{x}_{\mathsf{bot}},\boldsymbol{s}\rangle$ statistically hides $\beta$, we justify that the distribution $\mathcal{D}_{\Lambda,\sigma,-\boldsymbol{s}_0}$ reduced modulo the sub-lattice $p\Lambda$, (and therefore that of $\boldsymbol{s}$ reduced modulo $p$) is statistically close to the uniform distribution over $\boldsymbol{y}\cdot\mathbf{Z}/p\mathbf{Z}$. This is done by applying Lemma 3 and imposing a lower bound for $\sigma$. In order to do this we first need to bound $\lambda_1(p\Lambda)$ and therefore $||\boldsymbol{y}'||_2$. We thereby demonstrate that $\langle\mathbf{x}_{\mathsf{bot}},\boldsymbol{s}\rangle \mod p$ is statistically close to the uniform distribution over $\mathbf{Z}/p\mathbf{Z}$, and therefore, with overwhelming probability $\langle\mathbf{x}_{\mathsf{bot}},\boldsymbol{z}_\beta\rangle$ statistically hides $\beta$, thus concluding the proof.

Details. Since $\Lambda' = \boldsymbol{y}'\cdot\mathbf{Z}$, it holds that $||\boldsymbol{y}'||_2 = \det(\Lambda')$. If we define $\Lambda_{\mathsf{top}}$ as the lattice generated by the rows of $\mathbf{X}_{\mathsf{top}}$, then applying results from [Mar03] and [Ngu91, Theorem 2.8], one obtains that

$$||\boldsymbol{y}'||_2 = \det(\Lambda') \le \det(\Lambda_{\mathsf{top}}).$$

We now apply Hadamard's bound, which tells us that, since the coordinates of each $\overline{\mathbf{x}}_i$ are smaller than $p$ (since we assumed all requested $\overline{\mathbf{x}}_i$'s are independent), $\det(\Lambda_{\mathsf{top}}) \leq \prod_{i=1}^{\ell-1} ||\overline{\mathbf{x}}_i||_2 \leq (\sqrt{\ell}p)^{\ell-1}$. Therefore $||\boldsymbol{y}'||_2 \leq (\sqrt{\ell}p)^{\ell-1}$ and $s \cdot ||\boldsymbol{y}'||_2 < \tilde{s} \cdot (\sqrt{\ell}p)^{\ell-1}$, this implies

$$\lambda_1(p \cdot \Lambda) \leq p\tilde{s} \cdot (\sqrt{\ell}p)^{\ell-1}.$$

From [MR07] we know that the smoothing parameter verifies $\eta_\epsilon(p \cdot \Lambda) \leq \sqrt{\frac{\ln(2(1+1/\epsilon))}{\pi}} \cdot \lambda_1(p \cdot \Lambda)$. Thus for $\epsilon = 2^{-\lambda-1}$, we have $\eta_\epsilon(p \cdot \Lambda) \leq \sqrt{\lambda} \cdot p \cdot \tilde{s} \cdot (\sqrt{\ell}p)^{\ell-1}$. Therefore setting

$$\sigma \geq \sqrt{\lambda} \cdot p \cdot \tilde{s} \cdot (\sqrt{\ell}p)^{\ell-1}$$

and applying Lemma 3 ensures that the distribution $\mathcal{D}_{\Lambda,\sigma,-\boldsymbol{s}_0} \mod (p \cdot \Lambda)$ is within distance $2^{-\lambda}$ from the uniform distribution over $\Lambda/(p\Lambda)$ which is isomorphic to $\boldsymbol{y} \cdot \mathbf{Z}/p\mathbf{Z}$ because $\gcd(p, s) = 1$ and $\boldsymbol{y}' = \alpha \cdot \boldsymbol{y} \mod p$ for some $\alpha \in (\mathbf{Z}/p\mathbf{Z})^*$.

We have thus proven that $\boldsymbol{s}$ modulo $p$ is statistically close to the uniform distribution over $\boldsymbol{y} \cdot \mathbf{Z}/p\mathbf{Z}$ from the adversary's point of view. Since by construction $\langle \mathbf{x}_{\mathsf{bot}}, \boldsymbol{y} \rangle \neq 0 \mod p$, we get that $\langle \mathbf{x}_{\mathsf{bot}}, \boldsymbol{s} \rangle$ modulo $p$ is statistically close to the uniform distribution over $\mathbf{Z}/p\mathbf{Z}$. Moreover, in eq. 5, $\gcd(a, p) = 1$ with overwhelming probability, so $a \cdot \langle \mathbf{x}_{\mathsf{bot}}, \boldsymbol{s} \rangle$ statistically hides $\langle \mathbf{x}_{\mathsf{bot}}, \boldsymbol{y}_\beta \rangle$, which implies that $\langle \mathbf{x}_{\mathsf{bot}}, \boldsymbol{z}_\beta \rangle$ does not carry significant information about $\beta$, thus concluding the proof. □

## 6 Instantiation and efficiency considerations

We put forth two generic constructions of FE for the evaluation of inner products. Both schemes are based on variants of Elgamal in the same group and both sample their master secret keys from Gaussian distributions with the same standard deviation. As a result their asymptotic complexities are the same. The second scheme's security relies on a hard subgroup membership assumption (HSM) and this scheme appears to be the most efficient FE which evaluates inner product modulo a prime $p$. At the (small) expense of a single additional element in the keys and in the ciphertext, the first scheme's security relies on a weaker DDH-like assumption, which is also weaker than the DDH assumption in the group. We compare, in Table 1, an implementation of our HSM-based IPFE mod $p$ of Subsection 5.2 within the class group of an imaginary quadratic field and Paillier's variant of [ALS16]. This is the most relevant comparison since their DDH variant does not allow a full recovery of large inner products over $\mathbf{Z}/p\mathbf{Z}$, and, as detailed in the following paragraph, the LWE variant is far from being efficient, as ciphertexts are computed using arithmetic modulo $q = 2^\ell$ where $\ell$ is the dimension of the plaintext vectors.

*Comparison with the* LWE *based scheme of [ALS16].* Parameter choices for lattice-based cryptography are complex, indeed [ALS16] do not provide a concrete set of parameters. This being said, using [ALS16, Theorem 3], and setting

$\log p = \lambda$ as in Table 1, we give rough bit sizes for their LWE based FE scheme for computing inner products over $\mathbf{Z}/p\mathbf{Z}$. Basic elements are integers modulo $q$ of size $\ell$ since $q \approx 2^\ell$ for security to hold. The largest component in the master public key $mpk$ consists of $\lambda^2\ell^3$ elements, so $mpk$ is of size greater than $\lambda^2\ell^4$. The component $z_{\boldsymbol{x}}$ in secret keys is the product of a vector from $(\mathbf{Z}/p\mathbf{Z})^\ell$ with a matrix, which yields a secret key vector made up of $\lambda\ell^2$ inner products, where each inner product is of size $\ell\lambda$. Thus these keys are of size $\lambda^2\ell^3$. Finally ciphertexts consist of $\lambda\ell^2$ elements, and are thus of size greater than $\lambda\ell^3$. As a result, although it may be hard to compare the complexities in $\lambda$, for a fixed security level, the complexity in $\ell$ for all the parameters of the LWE based scheme is in $\ell^3$ or $\ell^4$ whereas we are linear in $\ell$ as one can see in Table 1. For example, for $\lambda = 128, \ell = 100$, their $sk_{\boldsymbol{x}}$ is of approximately $2^{34}$ bits vs. 13852 bits in our instantiation.

*Instantiation.* To instantiate the protocol of Section 5.2, we first need to define the algorithm GenGroup of Def. 6. To this end, we follow the lines of the construction from [CL15]. We start from a fundamental discriminant $\Delta_K = -p \cdot q$ with its class group $Cl(\Delta_K)$, where $q$ is a prime such that $p \cdot q \equiv -1 \pmod 4$ and $(p/q) = -1$. Then, we consider a non-maximal order of discriminant $\Delta_p = p^2 \cdot \Delta_K$ and its class group $Cl(\Delta_p)$. The order of $Cl(\Delta_p)$ is

$$h(\Delta_p) = p \cdot h(\Delta_K).$$

It is known (*cf.* [Coh00, p. 295]), that

$$h(\Delta_K) < \frac{1}{\pi} \log |\Delta_K| \sqrt{|\Delta_K|}$$

which is the bound we take for $\tilde{s}$ (note that a slightly better bound can be computed from the analytic class number formula, *cf.* [McC89]). In [CL15, Fig. 2] the authors show how to build a generator of a cyclic group of order $ps$ of the class group of discriminant $\Delta_p$ and a generator for the subgroup of order $p$ (in which the discrete logarithm problem is easy). We need to modify their generator of a DDH group with an easy DL subgroup, to make it output a generator $g_p$ of the subgroup of $p$-th powers. The computation of such an element is actually implicit in their generator: this is done by computing an ideal $\mathfrak{r}$ in the maximal order with norm a small prime $r$ such that $\left(\frac{\Delta_K}{r}\right) = 1$. Then the ideal $\mathfrak{r}^2$ is lifted into a class of $Cl(\Delta_p)$ which is then raised to the power $p$ to define $g_p$. A second modification is to output $\tilde{s}$ instead of their larger bound $B$ (since they sampled elements using a folded uniform distribution). We refer to [CL15] for a full description of the implementation. The manipulated objects are reduced ideals represented by two integers smaller than $\sqrt{p^3 q}$, and the arithmetic operations in class groups are very efficient, since the reduction and composition of quadratic forms have a quasi linear time complexity using fast arithmetic (see for instance [Coh00]).

The sole restriction on the size of the prime $p$ is that it must have at least $\lambda$ bits, where $\lambda$ is the security parameter. The size of $\Delta_K$, and thus of $q$, is chosen to thwart the best practical attack, which consists in computing discrete

logarithms in $Cl(\Delta_K)$ (or equivalently the class number $h(\Delta_K)$). An index-calculus method to solve the discrete logarithm problem in a class group of imaginary quadratic field of discriminant $\Delta_K$ was proposed in [Jac00]. It is conjectured in [BJS10] that a state of the art implementation of this algorithm has complexity $\mathcal{O}(L_{|\Delta_K|}[1/2, o(1)])$. They estimate that the discrete logarithm problem with a discriminant $\Delta_K$ of 1348 (resp. 1828 bits) is as hard as factoring a 2048 (resp. 3072 bits) RSA integer. This is our reference to estimate the bit size of the different elements in Table 1.

Table 1: Comparing our IPFE from HSM and the DCR scheme of [ALS16]

| | $\lambda = 112$ | | $\lambda = 128$ | |
|---|---|---|---|---|
| size | this work | DCR | this work | DCR |
| $(p, \tilde{s})$ | $(112, 684)$ | $(1024, 2046)$ | $(128, 924)$ | $(1536, 3070)$ |
| group element | 1572 | 4096 | 2084 | 6144 |
| secret key* $(z_{\boldsymbol{x}})$ | $112(\ell+1)+684$ | $2048(\ell+2)$ | $128(\ell+1)+924$ | $3072(\ell+2)$ |
| ciphertext | $1572(\ell+1)$ | $4096(\ell+1)$ | $2084(\ell+1)$ | $6144(\ell+1)$ |
| enc. expo. | 687 | 2046 | 928 | 3070 |
| dec. expo. | $112(\ell+1)+684$ | $2048(\ell+2)$ | $128(\ell+1)+924$ | $3072(\ell+2)$ |

* ignoring an additive term $(\ell \pm 1)\log(\sqrt{\ell})$

Note that in this case, the size of our group elements (reduced ideals in the class group of discriminant $p^3q$), are significantly smaller than those of the Paillier variant of [ALS16] (elements of $\mathbf{Z}/N^2\mathbf{Z}$). This is also the case for ciphertexts (which consist in both protocols of $\ell + 1$ group elements). We have the same situation with secret keys: to simplify the comparison we consider linearly independent queries (thus ignoring the vectors in $\mathbf{Z}^\ell$). As a result, we have, for our scheme, the inner product of a vector from $(\mathbf{Z}/p\mathbf{Z})^\ell$ with a vector sampled from a discrete Gaussian with standard deviation greater than $\sqrt{\lambda}p\tilde{s}(\sqrt{\ell}p)^{\ell-1}$ over $\mathbf{Z}^\ell$ vs. the inner product of a vector of $(\mathbf{Z}/N\mathbf{Z})^\ell$ with a vector sampled from a discrete Gaussian with standard deviation greater than $\sqrt{\lambda}(\sqrt{\ell}N)^{\ell+1}$ over $\mathbf{Z}^\ell$.

We note that our underlying message space $\mathbf{Z}/p\mathbf{Z}$ is much smaller than their message space $\mathbf{Z}/N\mathbf{Z}$. Using larger message spaces would be more favorable to their Paillier based scheme. But in practice, a 128 bits message space is large enough, if for instance, one needs to perform computations with double or quadruple precision. Our protocols are the most suited for such intermediate computations, since Paillier's construction from [ALS16] would add a large overhead cost, while their DDH construction could not decrypt the result.

In terms of timings, a fair comparison is difficult since to our knowledge, no library for the arithmetic of quadratic forms is as optimized as a standard library for the arithmetic of modular integers. Nevertheless, we note that the exponents involved in the (multi-)exponentiations (for encryption and decryption) are significantly smaller than those in [ALS16], and the group size is also smaller. Indeed, the encryption of Paillier's variant involves $(\ell + 1)$ exponentiations to

the power a $(|N|-2)$-bit integer modulo $N^2$, whereas our protocol involves one exponentiation to the power a $|\sigma'|$-bit integer in $Cl(p^3q)$, where $\sigma' > \tilde{s}\sqrt{\lambda}$ and $\ell$ (multi-)exponentiations whose maximum exponent size is also $|\sigma'|$. Decryptions involve respectively a multi-exponentiation whose maximum exponent size is lower than $\ell\sigma N = \ell\sqrt{\lambda}(\sqrt{\ell}N)^{\ell+1}N$ for [ALS16] and $\ell p\sigma = \ell p\sqrt{\lambda}p\tilde{s}(\sqrt{\ell}p)^{\ell-1}$ for our protocol.

We performed timings with Sage 8.1 on a standard laptop with a straightforward implementation. Using the settings of [CL15], the exponentiation in class groups uses a PARI/GP function (qfbnupow), which is far less optimised than the exponentiation in $\mathbf{Z}/N\mathbf{Z}$, implying a huge bias in favour of Paillier. Despite this bias, the efficiency improvement we expected from our protocols is reflected in practice, as showed in Table 2. We gain firstly from the fact that we can use smaller parameters for the same security level and secondly, because our security reductions allow to replace $N^\ell$ with $p^\ell$ in the derived keys. Thus the gain is not only in the constants and our scheme becomes more and more interesting as the security level and the dimension $\ell$ increase.

Table 2: Timings: our IPFE from HSM and *vs.* [ALS16]'s IPFE from DCR

| | $\lambda = 112, \ell = 10$ | | $\lambda = 128, \ell = 10$ | |
|---|---|---|---|---|
| | this work | [ALS16] | this work | [ALS16] |
| secret key bitsize | 1920 | 24592 | 2340 | 36876 |
| encryption time | 40ms | **27ms** | **78ms** | 85ms |
| decryption time | **110ms** | 301ms | **193ms** | 964ms |

For all parameters our dependency in $\ell$ is linear which allows to extrapolate timings for $\ell > 10$.

**Acknowledgements:** The authors would like to thank both Benoît Libert and Damien Stehlé for fruitful discussions. This work was supported by the French ANR ALAMBIC project (ANR-16-CE39-0006), and by ERC Starting Grant ERC-2013-StG-335086-LATTAC.

## References

ABCP16. M. Abdalla, F. Bourse, A. D. Caro, and D. Pointcheval. Better security for functional encryption for inner product evaluations. Cryptology ePrint Archive, Report 2016/011, 2016. http://eprint.iacr.org/2016/011.

ABDP15. M. Abdalla, F. Bourse, A. De Caro, and D. Pointcheval. Simple functional encryption schemes for inner products. In *PKC 2015*, *LNCS* 9020, p. 733–751. Springer, 2015.

ABP+17. S. Agrawal, S. Bhattacherjee, D. H. Phan, D. Stehlé, and S. Yamada. Efficient public trace and revoke from standard assumptions: Extended abstract. In *ACM CCS 17*, p. 2277–2293. ACM Press, 2017.

ABSV15. P. Ananth, Z. Brakerski, G. Segev, and V. Vaikuntanathan. From selective to adaptive security in functional encryption. In *CRYPTO 2015, Part II*, *LNCS* 9216, p. 657–677. Springer, 2015.

Adl94. L. M. Adleman. The function field sieve. In *Algorithmic Number Theory*, p. 108–121, Berlin, 1994. Springer Berlin Heidelberg.

AGVW13. S. Agrawal, S. Gorbunov, V. Vaikuntanathan, and H. Wee. Functional encryption: New perspectives and lower bounds. In *CRYPTO 2013, Part II*, *LNCS* 8043, p. 500–518. Springer, 2013.

ALS16. S. Agrawal, B. Libert, and D. Stehlé. Fully secure functional encryption for inner products, from standard assumptions. In *CRYPTO 2016, Part III*, *LNCS* 9816, p. 333–362. Springer, 2016.

BBL17. F. Benhamouda, F. Bourse, and H. Lipmaa. CCA-secure inner-product functional encryption from projective hash functions. In *PKC 2017, Part II*, *LNCS* 10175, p. 36–66. Springer, 2017.

BCP03. E. Bresson, D. Catalano, and D. Pointcheval. A simple public-key cryptosystem with a double trapdoor decryption mechanism and its applications. In *ASIACRYPT 2003*, *LNCS* 2894, p. 37–54. Springer, 2003.

BGJS16. S. Badrinarayanan, V. Goyal, A. Jain, and A. Sahai. Verifiable functional encryption. In *ASIACRYPT 2016, Part II*, *LNCS* 10032, p. 557–587. Springer, 2016.

BJS10. J.-F. Biasse, M. J. Jacobson, and A. K. Silvester. Security estimates for quadratic field based cryptosystems. In *ACISP 10*, *LNCS* 6168, p. 233–247. Springer, 2010.

BO13. M. Bellare and A. O'Neill. Semantically-secure functional encryption: Possibility results, impossibility results and the quest for a general definition. In *CANS 13*, *LNCS* 8257, p. 218–234. Springer, 2013.

Bou17. F. Bourse. *Functional Encryption for Inner-Product Evaluations.*. PhD thesis, PSL Research University, France, 2017.

BSW11. D. Boneh, A. Sahai, and B. Waters. Functional encryption: Definitions and challenges. In *TCC 2011*, *LNCS* 6597, p. 253–273. Springer, 2011.

CIL17. G. Castagnos, L. Imbert, and F. Laguillaumie. Encryption switching protocols revisited: Switching modulo p. In *CRYPTO 2017, Part I*, *LNCS* 10401, p. 255–287. Springer, 2017.

CL09. G. Castagnos and F. Laguillaumie. On the security of cryptosystems with quadratic decryption: The nicest cryptanalysis. In *EUROCRYPT 2009*, *LNCS* 5479, p. 260–277. Springer, 2009.

CL15. G. Castagnos and F. Laguillaumie. Linearly homomorphic encryption from DDH. In *CT-RSA 2015*, *LNCS* 9048, p. 487–505. Springer, 2015.

Coh00. H. Cohen. *A course in computational algebraic number theory*. Springer-Verlag, 2000.

CS98. R. Cramer and V. Shoup. A practical public key cryptosystem provably secure against adaptive chosen ciphertext attack. In *CRYPTO'98*, *LNCS* 1462, p. 13–25. Springer, 1998.

CS02. R. Cramer and V. Shoup. Universal hash proofs and a paradigm for adaptive chosen ciphertext secure public-key encryption. In *EUROCRYPT 2002*, *LNCS* 2332, p. 45–64. Springer, 2002.

CS03. J. Camenisch and V. Shoup. Practical verifiable encryption and decryption of discrete logarithms. In *CRYPTO 2003*, *LNCS* 2729, p. 126–144. Springer, 2003.

DIJ+13. 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 2013, Part II*, *LNCS* 8043, p. 519–535. Springer, 2013.

GGHZ16. S. Garg, C. Gentry, S. Halevi, and M. Zhandry. Functional encryption without obfuscation. In *TCC 2016-A, Part II*, *LNCS* 9563, p. 480–511. Springer, 2016.

Gjø05. K. Gjøsteen. Symmetric subgroup membership problems. In *PKC 2005*, *LNCS* 3386, p. 104–119. Springer, 2005.

GKP+13a. S. Goldwasser, Y. T. Kalai, R. A. Popa, V. Vaikuntanathan, and N. Zeldovich. How to run turing machines on encrypted data. In *CRYPTO 2013, Part II*, *LNCS* 8043, p. 536–553. 2013.

GKP+13b. S. Goldwasser, Y. T. Kalai, R. A. Popa, V. Vaikuntanathan, and N. Zeldovich. Reusable garbled circuits and succinct functional encryption. In *45th ACM STOC*, p. 555–564. ACM Press, 2013.

GPV08. C. Gentry, C. Peikert, and V. Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In *40th ACM STOC*, p. 197–206. ACM Press, 2008.

GVW12. S. Gorbunov, V. Vaikuntanathan, and H. Wee. Functional encryption with bounded collusions via multi-party computation. In *CRYPTO 2012*, *LNCS* 7417, p. 162–179. Springer, 2012.

HO12. B. Hemenway and R. Ostrovsky. Extended-DDH and lossy trapdoor functions. In *PKC 2012*, *LNCS* 7293, p. 627–643. Springer, 2012.

Jac00. M. J. Jacobson Jr. Computing discrete logarithms in quadratic orders. *Journal of Cryptology*, 13(4):473–492, 2000.

KSW08. J. Katz, A. Sahai, and B. Waters. Predicate encryption supporting disjunctions, polynomial equations, and inner products. In *EUROCRYPT 2008*, *LNCS* 4965, p. 146–162. Springer, 2008.

Luc02. S. Lucks. A variant of the Cramer-Shoup cryptosystem for groups of unknown order. In *ASIACRYPT 2002*, *LNCS* 2501, p. 27–45. Springer, 2002.

Mar03. J. Martinet. *Perfect Lattices in Euclidean Spaces*. Grundlehren der mathematischen Wissenschaften 327. Springer-Verlag Berlin 1 edition, 2003.

McC89. K. S. McCurley. Cryptographic key distribution and computation in class groups. In *Number Theory and Applications (Proc. NATO Advanced Study Inst. on Number Theory and Applications, Banff, 1988)*, 1989. Kluwer.

MR04. D. Micciancio and O. Regev. Worst-case to average-case reductions based on Gaussian measures. In *45th FOCS*, p. 372–381. IEEE Computer Society Press, 2004.

MR07. D. Micciancio and O. Regev. Worst-case to average-case reductions based on gaussian measures. *SIAM J. Comput.*, 37(1):267–302, 2007.

Ngu91. P. Q. Nguyen. *La Géométrie des Nombres en Cryptologie*. PhD thesis, École Normale Supérieure, 1991.

O'N10. A. O'Neill. Definitional issues in functional encryption. Cryptology ePrint Archive, Report 2010/556, 2010. http://eprint.iacr.org/2010/556.

Pai99. P. Paillier. Public-key cryptosystems based on composite degree residuosity classes. In *EUROCRYPT'99*, *LNCS* 1592, p. 223–238. Springer, 1999.

Sha84. A. Shamir. Identity-based cryptosystems and signature schemes. In *CRYPTO'84*, *LNCS* 196, p. 47–53. Springer, 1984.

SS10. A. Sahai and H. Seyalioglu. Worry-free encryption: functional encryption with public keys. In *ACM CCS 10*, p. 463–472. ACM Press, 2010.

SW05. A. Sahai and B. R. Waters. Fuzzy identity-based encryption. In *EUROCRYPT 2005*, *LNCS* 3494, p. 457–473. Springer, 2005.

Wat15. B. Waters. A punctured programming approach to adaptively secure functional encryption. In *CRYPTO 2015, Part II*, *LNCS* 9216, p. 678–697. Springer, 2015.

# Auxiliary Supporting Material

## I Proofs for background lemmas on Gaussian distributions

We here provide proofs for two of the lemmas on Gaussian distributions stated in Section 2. For ease of reading we here recall the lemmas before providing each of their respective proofs.

**Lemma 1.** *Let $\boldsymbol{x} \in \mathbf{R}^\ell \setminus \{\mathbf{0}\}$, $\boldsymbol{c} \in \mathbf{R}^\ell$, $\sigma \in \mathbf{R}$ with $\sigma > 0$ and $\sigma' = \sigma/||\boldsymbol{x}||_2$, $c' = \frac{\langle \boldsymbol{c}, \boldsymbol{x} \rangle}{\langle \boldsymbol{x}, \boldsymbol{x} \rangle}$. A random variable $K$ is distributed according to $\mathcal{D}_{\mathbf{Z},\sigma',c'}$ if and only if $V := K\boldsymbol{x}$ is distributed according to $\mathcal{D}_{\boldsymbol{x}\mathbf{Z},\sigma,\boldsymbol{c}}$.*

*Proof.* Let $k \in \mathbf{Z}$, and $\boldsymbol{v} := k\boldsymbol{x} \in \boldsymbol{x}\mathbf{Z}$, then

$$\Pr[V = \boldsymbol{v}] = \Pr[V = k\boldsymbol{x}] = \Pr[K = k] = \frac{\rho_{\sigma',c'}(k)}{\rho_{\sigma',c'}(\mathbf{Z})}.$$

As in the proof of [GPV08, Lemma 4.5], one can compute $\rho_{\sigma',c'}(k) = \rho_\sigma((k - c')||\boldsymbol{x}||_2) = \rho_\sigma((k - c')\boldsymbol{x}) = \rho_\sigma(\boldsymbol{v} - c'\boldsymbol{x})$. It holds that $\boldsymbol{u} := c'\boldsymbol{x}$ is the orthogonal projection of $\boldsymbol{c}$ on $\boldsymbol{x}\mathbf{R}$. By Pythagoras' Theorem,

$$||\boldsymbol{v} - \boldsymbol{c}||_2^2 = ||\boldsymbol{v} - \boldsymbol{u}||_2^2 + ||\boldsymbol{c} - \boldsymbol{u}||_2^2.$$

Thus $\rho_\sigma(\boldsymbol{v} - c'\boldsymbol{x}) = \rho_\sigma(||\boldsymbol{v} - \boldsymbol{u}||_2) = \rho_\sigma(||\boldsymbol{v} - \boldsymbol{c}||_2) \times C$ where $C = \exp(\frac{\pi||\boldsymbol{c} - \boldsymbol{u}||_2^2}{\sigma^2})$ is a constant. Therefore we have demonstrated that for $k \in \mathbf{Z}$, $\boldsymbol{v} = k\boldsymbol{x}$, $\rho_{\sigma',c'}(k) = \rho_{\sigma,\boldsymbol{c}}(\boldsymbol{v}) \times C$. And so:

$$\Pr[V = \boldsymbol{v}] = \frac{\rho_{\sigma,\boldsymbol{c}}(\boldsymbol{v}) \times C}{\sum_{z \in \mathbf{Z}} \rho_{\sigma,\boldsymbol{c}}(z\boldsymbol{x}) \times C} = \mathcal{D}_{\boldsymbol{x}\mathbf{Z},\sigma,\boldsymbol{c}}.$$

□

**Lemma 2.** *Let $\boldsymbol{x} \in \mathbf{R}^\ell$ with $\boldsymbol{x} \neq \mathbf{0}$, $\boldsymbol{c} \in \mathbf{R}^\ell$, $\sigma \in \mathbf{R}$ with $\sigma > 0$. Let $V$ be a random variable distributed according to $\mathcal{D}_{\boldsymbol{x}\cdot\mathbf{Z},\sigma,\boldsymbol{c}}$. Then the random variable $S$ defined as $S = \langle \boldsymbol{x}, V \rangle$ is distributed according to $\mathcal{D}_{||\boldsymbol{x}||_2^2 \cdot \mathbf{Z}, \sigma \cdot ||\boldsymbol{x}||_2, \langle \boldsymbol{c}, \boldsymbol{x} \rangle}$.*

*Proof.* As $V$ is distributed according to $\mathcal{D}_{\boldsymbol{x}\cdot\mathbf{Z},\sigma,\boldsymbol{c}}$, we have $V = K\boldsymbol{x}$ where $K$ is sampled from $\mathcal{D}_{\mathbf{Z},\sigma/||x||_2,c'}$ where $c' = \frac{\langle \boldsymbol{c}, \boldsymbol{x} \rangle}{\langle \boldsymbol{x}, \boldsymbol{x} \rangle}$ from the previous lemma. As a result, one can write $S = K\langle \boldsymbol{x}, \boldsymbol{x} \rangle$, and applying the previous lemma another time in dimension 1, we get that $S$ is distributed according to $\mathcal{D}_{||x||_2^2 \cdot \mathbf{Z}, \sigma \cdot ||\boldsymbol{x}||_2, \langle \boldsymbol{c}, \boldsymbol{x} \rangle}$. □

## II Description of the original CL protocol

From a DDH group with an easy DL subgroup, Castagnos and Laguillaumie proposed a generic framework to design a linearly homomorphic encryption scheme. An Elgamal type scheme is used in $G$, with plaintext message $m \in \mathbf{Z}/p\mathbf{Z}$ mapped to $f^m \in F$. The resulting scheme is linearly homomorphic. Thanks to the Solve algorithm, the decryption does not need a complex DL computation. We depict this scheme in Fig. 1. The Gen and Solve algorithms are those of Def. 6 except that we ignore the group $G^p$ and its generator (which are useless here). From Lemma 4, Item 2, choosing $\sigma > \tilde{s}p\sqrt{\lambda}$ suffices to ensure that the distribution $\{g^x, x \hookleftarrow \mathcal{D}_{\mathbf{Z},\sigma}\}$ is at distance less than $2^{-\lambda}$ from the uniform distribution in $G$. Note that the use of Gaussian sampling instead of uniform was suggested in [CIL17].

**Algorithm** $\mathsf{KeyGen}(1^\lambda)$

1. $(p, \tilde{s}, g, f, G, F) \leftarrow \mathsf{Gen}(1^\lambda, 1^\mu)$
2. Pick $x \hookleftarrow \mathcal{D}_{\mathbf{Z},\sigma}$ and set $h = g^x$
3. Set $pk = (p, \tilde{s}, g, h, f)$ and $sk = x$.
4. Return $(pk, sk)$

**Algorithm** $\mathsf{Decrypt}(1^\lambda, pk, sk, (c_1, c_2))$

1. Compute $M = c_2/c_1^x$
2. $m \leftarrow \mathsf{Solve}(p, g, f, G, F, M)$
3. Return $m$

**Algorithm** $\mathsf{Encrypt}(1^\lambda, pk, m)$

1. Pick $r \hookleftarrow \mathcal{D}_{\mathbf{Z},\sigma}$
2. Compute $c_1 = g^r$
3. Compute $c_2 = f^m h^r$
4. Return $(c_1, c_2)$

Fig. 1: The CL linearly homomorphic encryption scheme

We now recall the main assumption of [CL15].

**Definition 9 (DDH assumption [CL15])** *We say that* GenGroup *is the generator of a* DDH *group with easy* DL *subgroup $F$ if it holds that the* DDH *problem is hard in $G$ even with access to the* Solve *algorithm. More precisely, let $\mathcal{D}$ be a distribution over the integers such that the distribution over $G$ induced by $\{g^x; x \hookleftarrow \mathcal{D}\}$ is at distance less than $2^{-\lambda}$ from the uniform distribution in $G$. Let $\mathcal{A}$ be an adversary for the* DDH *problem, its advantage is defined as:*

$$\mathsf{Adv}^{\mathsf{DDH}}_{\mathcal{A}}(\lambda,\mu) = \Big|2 \cdot \Pr\big[b = b^\star : (p, \tilde{s}, g, f, g_p, G, F, G^p) \leftarrow \mathsf{Gen}(1^\lambda, 1^\mu), \\ x, y, z \hookleftarrow \mathcal{D}, X = g^x, Y = g^y, b \hookleftarrow \{0,1\}, Z_0 = g^z, Z_1 = g^{xy}, \\ b^\star \leftarrow \mathcal{A}(p, \tilde{s}, g, f, g_p, G, F, G^p, X, Y, Z_b, \mathsf{Solve}(.))\big] - 1\Big|$$

*The* DDH *problem is said to be hard in $G$ if for all probabilistic polynomial time attacker $\mathcal{A}$, $\mathsf{Adv}^{\mathsf{DDH}}_{\mathcal{A}}(\lambda,\mu)$ is negligible.*

## III Proof of Lemma 4: chosing distributions $\mathcal{D}$ and $\mathcal{D}_p$

The first item is a consequence of [CL15, Appendix C, Lemma 4]: it holds that the induced distribution on $G$ is at distance less than $(p \cdot s)/(2^\lambda \tilde{s} p) \leq 2^{-\lambda}$.

Item 2 follows from [CIL17, Appendix C, Lemma 1], Castagnos et al. demonstrate that the choice of $\mathcal{D} = \mathcal{D}_{\mathbf{Z},\sigma}$ with $\sigma > \tilde{s} \cdot p \cdot \sqrt{\lambda} > \tilde{s} \cdot p \cdot \sqrt{\ln(2(1+2^{\lambda+1}))/\pi}$ induces a distribution over $G$ at distance less than $2^{-\lambda}$ from the uniform distribution, and therefore trades a factor $2^{\lambda-1}$ for a factor $\sqrt{\lambda}$ compared to the previous choice. This also proves Item 3.

Since $G^p$ is a subgroup of $G$, $\mathcal{D}_p$ can be defined from $\mathcal{D}$ as in Item 4: the distribution $\{g_p^x, x \hookleftarrow \mathcal{D}\}$ is statistically close to the uniform distribution in $G^p$.

Item 5 follows from the fact that $G = F \times G^p$ and the following lemma.

**Lemma 7.** *Let $G = \langle g \rangle$ be a cyclic group of order $n = p \cdot s$ with $\gcd(p, s) = 1$, $G^p = \langle g_p \rangle$ the subgroup of $G$ of order $s$, and $F = \langle f \rangle$ the subgroup of $G$ of order $p$. Let $\mathcal{D}_p$ be a distribution over the integers such that $\{g_p^x, x \hookleftarrow \mathcal{D}_p\}$ is at statistical distance $\delta_p$ of the uniform distribution over $G^p$. Then the distribution induced by $\{g_p^x \cdot f^a, x \hookleftarrow \mathcal{D}_p, a \xleftarrow{\$} \mathbf{Z}/p\mathbf{Z}\}$ is also at statistical distance $\delta_p$ from the uniform distribution over $G$.*

*Proof.* As $\gcd(p, s) = 1$, one has $G = G^p \times F$. Let us denote $\psi = (\psi_1, \psi_2)$ the induced isomorphism from $G$ to $G^p \times F$. The probability that $\{g_p^x \cdot f^a, x \hookleftarrow \mathcal{D}_p, a \xleftarrow{\$} \mathbf{Z}/p\mathbf{Z}\}$ gives an element $h$ of $G$, is $\Pr[g_p^x = \psi_1(h)] \cdot \Pr[f^a = \psi_2(h)] = 1/p \Pr[g_p^x = \psi_1(h)]$. As a result, the statistical distance to the uniform distribution in $G$ is

$$\begin{aligned}\frac{1}{2} \sum_{h \in G} \left| \frac{1}{n} - \frac{1}{p} \cdot \Pr[g_p^x = \psi_1(h)] \right| &= \frac{1}{2} \cdot \frac{1}{p} \sum_{h \in G} \left| \frac{1}{s} - \Pr[g_p^x = \psi_1(h)] \right| \\ &= \frac{1}{2} \cdot \frac{1}{p} \cdot p \sum_{h_p \in G^p} \left| \frac{1}{s} - \Pr[g_p^x = h_p] \right| = \delta_p.\end{aligned}$$

□

## IV Proof of Theorem 2: the linearly homomorphic encryption scheme from HSM is ind-cpa

The proof proceeds as a sequence of games, starting with the real ind-cpa game and ending in a game where the ciphertext statistically hides the random bit $\beta$ chosen by the challenger. In Game $i$, we denote $S_i$ the event $\beta = \beta'$.

**Game 0 ⇒ Game 1:** In Game 1 the challenger creates the secret key $x$ from $\mathcal{D}_{\mathbf{Z},p\sigma'}$ instead of $\mathcal{D}_{\mathbf{Z},\sigma'}$. From Lemma 4, Item 4, $h$ is still at negligible distance of the uniform in $G^p$. Moreover, the challenger uses the secret key $x$ to compute the ciphertext element $c_2 = f^{m_\beta} g_p^{xr} = f^{m_\beta} c_1^x$. These two changes do not impact the distribution of the public key and of the ciphertext, therefore the adversary's success probability in both games is identical, $Pr[S_0] = Pr[S_1]$.

**Game 1**

1. $(p, \tilde{s}, f, g_p, G, F, G^p) \leftarrow \mathsf{Gen}(1^\lambda, 1^\mu)$
2. Pick $x \hookleftarrow \mathcal{D}_{\mathbf{Z},p\sigma'}$ and $h = g_p^x$
3. Set $pk = (\tilde{s}, g_p, f, p, h)$ and $sk = x$
4. $m_0, m_1 \leftarrow \mathcal{A}(pk)$
5. Pick $\beta \hookleftarrow \{0,1\}$
6. Pick $r \hookleftarrow \mathcal{D}_{\mathbf{Z},\sigma'}$
7. Compute $c_1 = g_p^r \in G^p$
8. Compute $c_2 = c_1^x \cdot f^{m_\beta}$
9. $\beta' \leftarrow \mathcal{A}(pk, c_1, c_2)$
10. Return $(\beta = \beta')$

**Game 2**

1. $(p, \tilde{s}, g, f, g_p, G, F, G^p) \leftarrow \mathsf{Gen}(1^\lambda, 1^\mu)$
2. Pick $x \hookleftarrow \mathcal{D}_{\mathbf{Z},p\sigma'}$ and $h = g_p^x$
3. Set $pk = (\tilde{s}, g_p, f, p, h)$ and $sk = x$
4. $m_0, m_1 \leftarrow \mathcal{A}(pk)$
5. Pick $\beta \hookleftarrow \{0,1\}$
6. Pick $r \hookleftarrow \mathcal{D}_{\mathbf{Z},\sigma'}$ and $u \hookleftarrow \mathbf{Z}/p\mathbf{Z}$
7. Compute $c_1 = f^u \cdot g_p^r \in G$
8. Compute $c_2 = c_1^x \cdot f^{m_\beta}$
9. $\beta' \leftarrow \mathcal{A}(pk, c_1, c_2)$
10. Return $(\beta = \beta')$

**Game 1 ⇒ Game 2:** In Game 1, the distribution of $c_1$ is at negligible distance of the uniform distribution in $G^p$. Now, in Game 2, the challenger samples a random $u \hookleftarrow \mathbf{Z}/p\mathbf{Z}$ and computes the ciphertext element $c_1 = f^u \cdot g_p^r \in G$ where $r \hookleftarrow \mathcal{D}_{\mathbf{Z},\sigma'}$. This gives an element at negligible distance of the uniform distribution in $G$ (*cf.* Lemma 4, Item 5). Both games are indistinguishable under the $\mathsf{HSM}$ assumption. Therefore, $|Pr[S_2] - Pr[S_1]| \leq \mathsf{Adv}_{\mathcal{B}}^{\mathsf{HSM}}(\lambda, \mu)$.

Now in Game 2 we have $c_1 = f^u \cdot g_p^r \in G$ which information theoretically reveals $u$ modulo $p$ and $r$ modulo $s$ by using the fact that $G = F \times G^p$. We also have $c_2 = c_1^x f^{m_\beta} = g_p^{rx} f^{m_\beta + ux} = h^r f^{m_\beta + ux}$. For the adversary the value of $h^r$ is fixed, so he can infer $m_\beta + ux \in \mathbf{Z}/p\mathbf{Z}$. Since $u$ is sampled uniformly at random from $\mathbf{Z}/p\mathbf{Z}$, $u \neq 0 \mod p$ with all but negligible probability as $p$ is a $\mu$ bit prime, with $\mu \geq \lambda$. Furthermore, as $x$ is sampled from $\mathcal{D}_{\mathbf{Z},p\sigma'}$ with $p\sigma' > p\tilde{s}\sqrt{\lambda}$, the distribution of $x$ modulo $n$ is at negligible distance of the uniform modulo $n$ (*cf.* Lemma 4, Item 2). In particular, as $n = ps$ with $\gcd(p, s) = 1$, $x$ modulo $p$ is at negligible distance of the uniform and is independent of $x$ modulo $s$. So even if an unbounded adversary can learn $x$ modulo $s$ from $h$, $x$ modulo $p$ remains at negligible distance of the uniform from his point of view and $m_\beta + ux$ perfectly hides $m_\beta \in \mathbf{Z}/p\mathbf{Z}$. Therefore: $|Pr[S_2] - 1/2| \leq 2^{-\lambda}$. Combining the probability equations, we conclude the proof with the following inequality:

$$\mathsf{Adv}_{\mathcal{A}}^{\Pi_{2a}}(\lambda, \mu) \leq \mathsf{Adv}_{\mathcal{B}}^{\mathsf{HSM}}(\lambda, \mu) + 2^{-\lambda}$$

# V Proof of Theorem 3: the linearly homomorphic encryption scheme from DDH-f is ind-cpa

The proof proceeds as a sequence of games, starting with the real $\mathsf{ind\text{-}cpa}$ game and ending in a game where the ciphertext statistically hides the random bit $\beta$ chosen by the challenger. In Game $i$, we denote $S_i$ the event $\beta = \beta'$.

**Game 0 ⇒ Game 1:** In Game 1 the challenger uses the secret key $x, y$ to compute the ciphertext element $c_3 = c_1^x c_2^y f^{m_\beta} = g^{rx} h^{ry} f^{m_\beta} = \eta^r f^{m_\beta}$. This change does not impact the distribution of the ciphertext, therefore the adversary's success probability in both games is identical, $Pr[S_0] = Pr[S_1]$.

**Game 1**

1. $(pk, sk) \leftarrow \mathsf{KeyGen}(1^\lambda, 1^\mu)$
2. $m_0, m_1 \leftarrow \mathcal{A}(pk)$
3. Pick $\beta \hookleftarrow \{0,1\}$
4. Pick $r \hookleftarrow \mathcal{D}_{\mathbf{Z},\sigma}$
5. Compute $c_1 = g^r$
6. Compute $c_2 = h^r$
7. Compute $c_3 = c_1^x c_2^y f^{m_\beta}$
8. $\beta' \leftarrow \mathcal{A}(pk, c_1, c_2, c_3)$
9. Return $(\beta = \beta')$

**Game 2**

1. $(pk, sk) \leftarrow \mathsf{KeyGen}(1^\lambda, 1^\mu)$
2. $m_0, m_1 \leftarrow \mathcal{A}(pk)$
3. Pick $\beta \hookleftarrow \{0,1\}$
4. Pick $r \hookleftarrow \mathcal{D}_{\mathbf{Z},\sigma}$ and $u \hookleftarrow \mathbf{Z}/p\mathbf{Z}$
5. Compute $c_1 = g^r$
6. Compute $c_2 = h^r f^u$
7. Compute $c_3 = c_1^x c_2^y f^{m_\beta}$
8. $\beta' \leftarrow \mathcal{A}(pk, c_1, c_2, c_3)$
9. Return $(\beta = \beta')$

**Game 1 $\Rightarrow$ Game 2:** In Game 1, $(h = g^\alpha, c_1 = g^r, c_2 = h^r = g^{\alpha r})$ with $\alpha, r \hookleftarrow \mathcal{D}_{\mathbf{Z},\sigma}$ is a DH triplet. Now, in Game 2, the challenger samples a random $u \hookleftarrow \mathbf{Z}/p\mathbf{Z}$ and computes $c_2 = h^r f^u$. Both games are indistinguishable under the DDH-f assumption (*cf.* Def. 8). Therefore, $|Pr[S_2] - Pr[S_1]| \leq \mathsf{Adv}_{\mathcal{B}}^{\mathsf{DDH\text{-}f}}(\lambda, \mu)$.

Now in Game 2 we have $c_1 = g^r$ which information theoretically reveals $r$ modulo $n$. Furthermore, $c_3 = c_1^x c_2^y f^{m_\beta} = \eta^r f^{m_\beta + uy}$. This information theoretically reveals $m_\beta + uy \in \mathbf{Z}/p\mathbf{Z}$ as the value of $\eta^r$ is fixed from $c_1$. Since $u$ is sampled uniformly at random from $\mathbf{Z}/p\mathbf{Z}$, $u \neq 0 \mod p$ with all but negligible probability as $p$ is a $\mu$ bit prime, with $\mu \geq \lambda$. As a result, we are interested in the distribution of $y$ modulo $p$ from the adversary's point of view.

The only information that $\mathcal{A}$ learns about $y$ comes from $c_3$ and $\eta = g^x h^y$. This means that from $\eta$ an unbounded adversary learns $z := x + \alpha y$ modulo $n$. Knowing $z$, the joint distribution of $(x, y)$ modulo $n$ is

$$(z - \alpha y \mod n, y \mod n) \text{ where } y \hookleftarrow \mathcal{D}_{\mathbf{Z},\sigma}.$$

As a result, knowing $z$, $y$ modulo $n$ is statistically close to the uniform distribution modulo $n$ due to the choice of $\mathcal{D}_{\mathbf{Z},\sigma}$. Consequently for the adversary $y$ modulo $p$ is also statistically close to the uniform distribution modulo $p$ and $m_\beta + uy \in \mathbf{Z}/p\mathbf{Z}$ perfectly hides $m_\beta \in \mathbf{Z}/p\mathbf{Z}$.

Therefore: $|Pr[S_2] - 1/2| \leq 2^{-\lambda}$. Combining the probability equations, we conclude the proof with the following inequality:

$$\mathsf{Adv}_{\mathcal{A}}^{\Pi_{2b}}(\lambda, \mu) \leq \mathsf{Adv}_{\mathcal{B}}^{\mathsf{DDH\text{-}f}}(\lambda, \mu) + 2^{-\lambda}$$

## VI Invertibility of X in Lemmas 5 and 6

We here prove that $\mathbf{X}$ is invertible modulo $p$. To see this consider the matrix:

$$\mathbf{X}\mathbf{X}^T = \begin{bmatrix} \mathbf{I}_{n_0} & & & \\ & y_{n_0+2}^2 + y_{n_0+1}^2 & -y_{n_0+1} \cdot y_{n_0+3} & \\ & -y_{n_0+1} \cdot y_{n_0+3} & y_{n_0+3}^2 + y_{n_0+2}^2 & -y_{n_0+2} \cdot y_{n_0+4} \\ & & \ddots & \ddots & \ddots \\ & & & & -y_\ell \cdot y_{\ell-2} \quad y_{\ell-1}^2 + y_\ell^2 \\ & & & & & ||\mathbf{y}||_2^2 \end{bmatrix}.$$

We first claim that:

$$\det \mathbf{X}\mathbf{X}^T = \left( \prod_{i=n_0+2}^{\ell-1} y_i^2 \right) \cdot ||\mathbf{y}||_2^4.$$

The proof proceeds by induction on $\ell$. Let us first introduce some notations. We denote $\widetilde{\mathbf{X}\mathbf{X}^T} \in \mathbf{Z}^{(\ell-n_0-1)\times(\ell-n_0-1)}$ the matrix:

$$\widetilde{\mathbf{X}\mathbf{X}^T} = \begin{bmatrix} y_{n_0+2}^2 + y_{n_0+1}^2 & -y_{n_0+1} \cdot y_{n_0+3} & & & \\ -y_{n_0+1} \cdot y_{n_0+3} & y_{n_0+3}^2 + y_{n_0+2}^2 & -y_{n_0+2} \cdot y_{n_0+4} & & \\ & \ddots & \ddots & \ddots & \\ & & & -y_\ell \cdot y_{\ell-2} & y_{\ell-1}^2 + y_\ell^2 \end{bmatrix}.$$

It holds that $\det \mathbf{X}\mathbf{X}^T = \det \widetilde{\mathbf{X}\mathbf{X}^T} \cdot ||\mathbf{y}||^2$. It thus suffices to prove that

$$\det \widetilde{\mathbf{X}\mathbf{X}^T} = \left( \prod_{i=n_0+2}^{\ell-1} y_i^2 \right) \cdot ||\mathbf{y}||^2.$$

For $\ell = n_0 + 2$, $\widetilde{\mathbf{X}\mathbf{X}^T} = y_{n_0+2}^2 + y_{n_0+1}^2$ and $\det \widetilde{\mathbf{X}\mathbf{X}^T} = ||\mathbf{y}||^2$.
For $\ell = n_0 + 3$,

$$\widetilde{\mathbf{X}\mathbf{X}^T} = \begin{bmatrix} y_{n_0+2}^2 + y_{n_0+1}^2 & -y_{n_0+1} \cdot y_{n_0+3} \\ -y_{n_0+1} \cdot y_{n_0+3} & y_{n_0+3}^2 + y_{n_0+2}^2 \end{bmatrix} \in \mathbf{Z}^{2\times 2}.$$

and

$$\begin{aligned} \det \widetilde{\mathbf{X}\mathbf{X}^T} &= (y_{n_0+2}^2 + y_{n_0+1}^2) \cdot (y_{n_0+3}^2 + y_{n_0+2}^2) - (y_{n_0+1} \cdot y_{n_0+3})^2 \\ &= y_{n_0+2}^2 \cdot \sum_{i=n_0+1}^{n_0+3} y_i^2 \\ &= y_{n_0+2}^2 \cdot \mathbf{y}^2. \end{aligned}$$

Thus the property holds for $\ell = n_0 + 2$ and $\ell = n_0 + 3$. Assume the property holds for $\ell = n_0 + k - 1$ and $\ell = n_0 + k$, for some $k \geq 3$. We prove that the property holds for $\ell = n_0 + k + 1$. We denote $\boldsymbol{A}_k \in \mathbf{Z}^{(k-1)\times(k-1)}$ the matrix considered for $\ell = n_0 + k$, i.e.:

$$\begin{bmatrix} y_{n_0+2}^2 + y_{n_0+1}^2 & -y_{n_0+1} \cdot y_{n_0+3} & & & \\ -y_{n_0+1} \cdot y_{n_0+3} & y_{n_0+3}^2 + y_{n_0+2}^2 & -y_{n_0+2} \cdot y_{n_0+4} & & \\ & \ddots & \ddots & \ddots & \\ & & & -y_{n_0+k-2} \cdot y_{n_0+k} & y_{n_0+k}^2 + y_{n_0+k-1}^2 \end{bmatrix}$$

Then we have:

$$\boldsymbol{A}_{k+1} = \left[\begin{array}{c|c} \boldsymbol{A}_k & \begin{matrix} \\ -y_{n_0+k+1} \cdot y_{n_0+k-1} \end{matrix} \\ \hline \begin{matrix} & -y_{n_0+k+1} \cdot y_{n_0+k-1} \end{matrix} & y_{n_0+k}^2 + y_{n_0+k+1}^2 \end{array}\right] \in \mathbf{Z}^{k\times k}.$$

If we define:

$$\boldsymbol{B}_k = \left[\begin{array}{c|c} \boldsymbol{A}_{k-1} & \begin{array}{c}\boldsymbol{0}\\ -y_{n_0+k}\cdot y_{n_0+k-2}\end{array} \\ \hline 0 & -y_{n_0+k-1}\cdot y_{n_0+k+1} \end{array}\right] \in \mathbf{Z}^{(k-1)\times(k-1)},$$

it is easy to see that $\det \boldsymbol{B}_k = -y_{n_0+k-1}\cdot y_{n_0+k+1}\cdot \det \boldsymbol{A}_{k-1}$.
Then we have:

$$\begin{aligned}
\det \boldsymbol{A}_{k+1} &= (y_{n_0+k}^2 + y_{n_0+k+1}^2)\cdot \det \boldsymbol{A}_k + (y_{n_0+k+1}\cdot y_{n_0+k-1})\cdot \det \boldsymbol{B}_k \\
&= (y_{n_0+k}^2 + y_{n_0+k+1}^2)\cdot \det \boldsymbol{A}_k - (y_{n_0+k-1}\cdot y_{n_0+k+1})^2\cdot \det \boldsymbol{A}_{k-1} \\
&= (y_{n_0+k}^2 + y_{n_0+k+1}^2)\cdot \left(\prod_{i=n_0+2}^{n_0+k-1} y_i^2\right)\cdot \sum_{i=n_0+1}^{n_0+k} y_i^2 \\
&\quad - (y_{n_0+k-1}\cdot y_{n_0+k+1})^2\cdot \left(\prod_{i=n_0+2}^{n_0+k-2} y_i^2\right)\cdot \sum_{i=n_0+1}^{n_0+k-1} y_i^2 \\
&= \left(\prod_{i=n_0+2}^{n_0+k} y_i^2 \cdot \sum_{i=n_0+1}^{n_0+k} y_i^2\right) + \left(y_{n_0+k+1}^2 \cdot \prod_{i=n_0+2}^{n_0+k-1} y_i^2 \cdot \sum_{i=n_0+1}^{n_0+k} y_i^2\right) \\
&\quad - \left(y_{n_0+k+1}^2 \cdot \prod_{i=n_0+2}^{n_0+k-1} y_i^2 \cdot \sum_{i=n_0+1}^{n_0+k-1} y_i^2\right) \\
&= \left(\prod_{i=n_0+2}^{n_0+k} y_i^2 \cdot \sum_{i=n_0+1}^{n_0+k} y_i^2\right) + \left(y_{n_0+k+1}^2 \cdot y_{n_0+k}^2 \cdot \prod_{i=n_0+2}^{n_0+k-1} y_i^2\right) \\
&= \left(\prod_{i=n_0+2}^{n_0+k} y_i^2 \cdot \sum_{i=n_0+1}^{n_0+k} y_i^2\right) + \left(y_{n_0+k+1}^2 \cdot \prod_{i=n_0+2}^{n_0+k} y_i^2\right) \\
&= \prod_{i=n_0+2}^{n_0+k} y_i^2 \cdot \sum_{i=n_0+1}^{n_0+k+1} y_i^2 \\
&= \prod_{i=n_0+2}^{\ell-1} y_i^2 \cdot ||\mathbf{y}||^2.
\end{aligned}$$

We reasonably assume that $\ell \geq 2$ and $p \geq 2$. Since $p$ is prime, and since for $n_0+1 \leq i \leq \ell$, each $y_i$ is non-zero and $y_i^2 < p$ due to the norm bound $||\mathbf{y}||_\infty < 2(p/2\ell)^{1/2} \leq \sqrt{p}$ (for $l \geq 2$), it holds that $\prod_{i=n_0+2}^{\ell-1} y_i^2 \neq 0 \mod p$. Moreover $||\mathbf{y}||_2 \leq \sqrt{\ell}||\mathbf{y}||_\infty < \sqrt{2p} \leq p$ (for $p \geq 2$), so $||\mathbf{y}||_2^4 \neq 0 \mod p$. This yields $\det \mathbf{X}\mathbf{X}^T \neq 0 \mod p$, i.e. $\mathbf{X}$ is invertible in $\mathbf{Z}_p$.