Disclaimer

These deliverables may be subject to final acceptance by the European Commission. The results of these deliverables reflect only the author's view and the Commission is not responsible for any use that may be made of the information it contains.

Statement for open documents

These documents and its content are the property of the FENTEC Consortium. The content of all or parts of these documents can be used and distributed provided that the FENTEC project and the document are properly referenced

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 780108. Any dissemination of results here presented reflects only the consortium view.

D4.7 Annual Report on Quantum-Safe Functional Encryption schemes Y2

Document Identification			
Status	Final	**Due Date**	31/12/2018
Version	1.0	**Submission Date**	17/12/2018

Related WP	WP4	**Document Reference**	D4.7
Related Deliverable(s)	D4.1	**Dissemination Level(*)**	PU
Lead Participant	ENS	**Lead Author**	Michel Abdalla
Contributors	ENS	**Reviewers**	Ward Beullens (KU Leuven) Svetla Nikova (KU Leuven) Hendrik Waldner (UEDIN)

Keywords:
Functional Encryption Schemes, Quantum-Safe, Web Analytics

This document is issued within the framework and for the purpose of the FENTEC project. This project has received funding from the European Union's Horizon2020 under Grant Agreement No. 780108. The opinions expressed and arguments employed herein do not necessarily reflect the official views of the European Commission.

Document Information

List of Contributors	
Name	Partner
Michel Abdalla	ENS

Document History			
Version	Date	Change editors	Changes
0.1	30/11/2018	Michel Abdalla (ENS)	ToC
0.2	11/12/2018	Michel Abdalla (ENS)	Version for reviewing
0.3	14/12/2018	Michel Abdalla (ENS)	Addressed reviewers comments
0.4	14/12/2018	Michel Abdalla (ENS)	Fixed minor typos
1	17/12/2018	Michel Abdalla (ENS)	Final version

Quality Control		
Role	Who (Partner short name)	Approval Date
Deliverable Leader	Michel Abdalla (ENS)	17/12/2018
Technical Manager	Michel Abdalla (ENS)	17/12/2018
Quality Manager	Diego Esteban (ATOS)	17/12/2018
Project Coordinator	Francisco Gala (ATOS)	17/12/2018

Table of Contents

List of Figures

List of Acronyms

Acronym	Description
DCR	Decisional Composite Residuosity
DDH	Decisional Diffie-Helmman
FE	Functional Encryption
LWE	Learning With Errors
MIFE	Multi-Input Functional Encryption
MQ	Multivariate Quadratic
PPT	Probabilistic Polynomial Time
ROM	Random-Oracle Model
WP	Work Package

Executive Summary

Most of the existing functional encryption schemes in use today are based on the presumed hardness of the discrete-log and the integer-factorization problems, which are known to be insecure with respect to quantum computers [23]. To prevent the collapse of the cryptographic protocols relying on these schemes, it is important to develop alternative solutions based on mathematical problems that are unrelated to factoring and discrete log and that may be impervious to attacks by quantum computers. Hence, one of the main goals of WP4 is to design quantum-safe functional encryption alternatives that use lattices as their source of computational hardness. In this deliverable, we describe our progress towards this goal.

More precisely, we describe a new multi-input functional encryption construction for the inner-product functionality developed by Abdalla et al. [3] in the context of the FENTEC project, which was the *first* such scheme based on lattice problems. Since their construction is generic and can be based on any single-input functional inner-product encryption satisfying some common structural properties, we describe two possible lattice instantiations based on the problem of Learning With Errors (LWE). In addition to being quantum-safe, another advantage of these schemes is that they also allow for the computation of inner products of arbitrary sizes.

1 Introduction

Most of the existing applications of public-key cryptography currently in use are based on the presumed hardness of the discrete-log and the integer-factorization problems. Unfortunately, it is well known that a technological breakthrough, such as the construction of a quantum computer, could call into question the difficulty of these problems, as demonstrated by Shor [23], and render all the existing protocols based on these problems completely insecure. A natural way of addressing this problem is to build provably secure protocols based on mathematical problems that are unrelated to factoring and discrete log and that could remain secure even in the presence of quantum computers. One of the most promising directions in this line of research is to use lattice problems as a source of computational hardness – in particular since they also offer features that other alternative public-key cryptosystems (such as MQ-based, code-based or hash-based schemes) cannot provide.

Despite great progress in the field over the last several years, efficiency still remains a very large obstacle for advanced lattice primitives. While constructions of identity-based encryption schemes, group signature schemes, functional encryption schemes, and even fully-homomorphic encryption schemes are known, the efficiency of their implementations remains an issue. It is safe to surmise that if the state of affairs remains as it is in the present, then despite all the theoretical efforts that went into their constructions, these schemes will never be used in practical applications.

Functional encryption. Functional encryption (FE) [10, 21] is a generalization of the notion of public-key encryption, which allows fine-grained access control over encrypted data. Besides the classical encryption and decryption procedures, functional encryption schemes consists of a key derivation algorithm, which allows the owner of a master secret key to derive keys with more restricted capabilities. These derived keys sk_f are called functional decryption keys and are associated with a function f. Using the key sk_f for the decryption of a ciphertext $\mathsf{Enc}(x)$ generates the output $f(x)$. During this decryption procedure no more information is revealed about the underlying plaintext than $f(x)$.

The standard security requirement for both FE and MIFE imposes that decryption keys should be collusion resistant. This means that a group of users, holding different decryption keys, should not be able to gain information about the encrypted messages, beyond the union of what they can individually learn. More precisely, an adversary that obtains the secret keys corresponding to functions $f_1, \ldots, f_n$ should not be able to decide which of the challenge messages x_0, x_1 was encrypted, as long as $f_i(x_0) = f_i(x_1)$ for all i. This models the idea that an individual's messages are still secure even if an arbitrary number of other users of the system collude against that user.

Several FE schemes for general functionalities have already been proposed [18, 11, 24, 19]. Unfortunately, they are far from being practical and their security relies on unstable assumptions, such as indistinguishable obfuscation or multilinear maps. In order to overcome the deficiency of these schemes, Abdalla et al. [1] focused on the construction of FE schemes for *specific functionalities* of practical interest. In particular, they proposed simple FE schemes for the inner-product functionality based on standard assumptions, such as the Decisional Diffie-Hellman (DDH) and the Learning-With-Errors (LWE) assumptions (see Section 2). Following their work, several other practical FE schemes for inner products [9, 15, 5] and their quadratic extensions [8] have been proposed.

Multi-input functional encryption. The basic notion of functional encryption considers functionalities where all the inputs are provided and encrypted by a single party. The more general case of multi-input functionalities is captured by the notion of multi-input functional encryption (MIFE, for short) [20]. Informally, this notion can be thought of as an FE scheme where n encryption slots are explicitly given, in the sense that a user who is assigned the i-th slot can, independently, create a ciphertext $\mathsf{Enc}(x_i)$ from his own plaintext x_i. Given ciphertexts $\mathsf{Enc}(x_1), \ldots, \mathsf{Enc}(x_n)$, one can use a secret key sk_f to retrieve $f(x_1, \ldots, x_n)$, similarly to the basic FE notion. This multi-input capability makes MIFE particularly well suited for many real life scenarios (such as data mining over encrypted data or multi-client delegation of computation) where the (encrypted) data may come from different and unrelated sources.

In the last few years, several multi-input functional encryption schemes have been constructed. The vast majority, however, are impractical and based on unstable assumptions, such as indistinguishable obfuscation or multilinear maps (e.g., [20, 7, 6, 12]).

The first practical construction of a MIFE scheme was proposed by Abdalla et al. in [4], by focusing on the inner-product functionality. Their construction, however, works over bilinear groups and cannot be instantiated with lattices. Their result was later extended by Chotard et al. in [14], which additionally considered the problem of decentralization.

1.1 Purpose of the Document

The primary goal of this deliverable is to describe our contributions to the design of practical quantum-safe functional encryption schemes within the FENTEC project. Towards this goal, we present a new MIFE construction by Abdalla, Catalano, Fiore, Gay, and Ursu [3], which overcomes the shortcomings of the original MIFE construction by Abdalla et al. in [4]. More precisely, the MIFE construction in [3] is generic, in the sense that it can transform any single-input FE that satisfies some structural properties into a multi-input FE, under the same assumption. In particular, by using previous known single-input FE schemes for the inner product functionality that are based on lattice problems, such as Learning With Errors (LWE), we obtain the first quantum-safe MIFE scheme for inner products.

1.2 Structure and Methodology

Section 2 first recalls some of the definitions and basic tools that are used in the remainder of the document, such as notations, complexity assumptions, and security definitions for multi-input functional encryption. Section 3 then describes our main contribution, which is the generic construction of multi-input functional inner-product encryption from a single-input functional inner-product encryption. Next, Section 4 describes two concrete quantum-safe single-input FE schemes that be used to instantiate the generic construction in Section 3, one by Agrawal et al. [5] and one by Abdalla et al. [2]. Finally, Section 5 concludes by discussing future research directions.

1.3 Relation to Deliverable 4.1

The scheme multi-input functional inner-product encryption was already described in Deliverable 4.1, since it is applicable to the web analytics use case considered in WP7. In comparison to that deliverable, the current deliverable provides more details about the actual construction and its possible instantiations.

2 Basic tools

In this section, we recall some of the definitions and basic tools that will be used in the remainder of the document.

2.1 Notation and conventions

We denote with $\lambda \in \mathbb{N}$ a security parameter. A *probabilistic polynomial time* (PPT) algorithm $\mathcal{A}$ is a randomized algorithm for which there exists a polynomial $p(\cdot)$ such that for every input x the running time of $\mathcal{A}(x)$ is bounded by $p(|x|)$. We say that a function $\varepsilon : \mathbb{N} \to \mathbb{R}^+$ is *negligible* if for every positive polynomial $p(\lambda)$ there exists $\lambda_0 \in \mathbb{N}$ such that for all $\lambda > \lambda_0$: $\varepsilon(\lambda) < 1/p(\lambda)$. If S is a set, $x \stackrel{R}{\leftarrow} S$ denotes the process of selecting x uniformly at random in S. If $\mathcal{A}$ is a probabilistic algorithm, $y \stackrel{R}{\leftarrow} \mathcal{A}(\cdot)$ denotes the process of running $\mathcal{A}$ on some appropriate input and assigning its output to y. For a positive integer n, we denote by $[n]$ the set $\{1, \ldots, n\}$. We denote vectors $\mathbf{x} = (x_i)$ and matrices $\mathbf{A} = (a_{i,j})$ in bold. For a set S (resp. vector $\mathbf{x}$) $|S|$ (resp. $|\mathbf{x}|$) denotes its cardinality (resp. number of entries). Also, given two vectors $\mathbf{x}$ and $\mathbf{x}'$ we denote by $\mathbf{x}\|\mathbf{x}'$ their concatenation. By $\equiv$, we denote the equality of statistical distributions, and for any $\varepsilon > 0$, we denote by $\approx_\varepsilon$ the ε-statistical difference of two distributions.

In the technical overview in Section 3.1, we use implicit representation of group elements as introduced in [17]. That is, if $\mathbb{G}$ is a group of order p and g a generator, then $\forall a \in \mathbb{Z}_p$, we note $[a] = g^a$. If $A \in \mathbb{Z}_p^{m\times n}$ is a matrix, then $[A] = (g^{a_{i,j}})_{1\le i\le m, 1\le j\le n}$.

2.2 Learning With Errors (LWE)

Since this report only considers quantum-safe schemes, we now recall the *Learning-With-Errors* (LWE) complexity assumption used in some of these schemes.

Definition 1 (Learning With Errors (LWE) assumption) *Let q, α, m be functions of a parameter n. For a secret $\mathbf{s} \in \mathbb{Z}_q^n$, the distribution $A_{q,\alpha,s}$ over $\mathbb{Z}_q^n \times \mathbb{Z}_q$ is obtained by sampling $\mathbf{a} \stackrel{R}{\leftarrow} \mathbb{Z}_q^n$ and an error $e \stackrel{R}{\leftarrow} \psi_{\mathbb{Z},\alpha,q}$ from an error distribution $\psi_{\mathbb{Z},\alpha,q}$, and returning $(\mathbf{a}, \langle \mathbf{a}, \mathbf{s}\rangle + e) \in \mathbb{Z}_q^{n+1}$. Let $U(\mathbb{Z}_q^{m\times(n+1)})$ denote the uniform distribution over $\mathbb{Z}_q^{m\times(n+1)}$. The Learning With Errors problem $\mathsf{LWE}_{q,\alpha,m}$ is as follows: For $\mathbf{s} \stackrel{R}{\leftarrow} \mathbb{Z}_q^n$, the goal is to distinguish between the distributions:*

$$\mathsf{D}_0(\mathbf{s}) := U(\mathbb{Z}_q^{m\times(n+1)}) \text{ and } \mathsf{D}_1(\mathbf{s}) := (A_{q,\alpha,\mathbf{s}})^m.$$

We say that a PPT algorithm $\mathcal{A}$ solves the $\mathsf{LWE}_{q,\alpha,m}$ problem if it distinguishes $\mathsf{D}_0(\mathbf{s})$ and $\mathsf{D}_1(\mathbf{s})$ (with non-negligible advantage over the random coins of $\mathcal{A}$ and the randomness of the samples) with non-negligible probability over the randomness of $\mathbf{s}$. The LWE assumption states that no such adversary exists.

2.3 Multi-Input Functional Encryption

We now recall the definitions of multi-input functional encryption [20] specialized to the private-key setting, as this is the one relevant for the constructions in this report.

Definition 2 (Multi-input Functional Encryption) *Let $\mathcal{F} = \{\mathcal{F}_n\}_{n\in\mathbb{N}}$ be an ensemble where each $\mathcal{F}_n$ is a family of n-ary functions. A function $f \in \mathcal{F}_n$ is defined as follows $f : \mathcal{X}_1 \times \ldots \times \mathcal{X}_n \rightarrow \mathcal{Y}$. A multi-input functional encryption scheme $\mathcal{MIFE}$ for $\mathcal{F}$ consists of the following algorithms:*

- $\mathsf{Setup}(1^\lambda, \mathcal{F}_n)$ *takes as input the security parameter λ and a description of $\mathcal{F}_n \in \mathcal{F}$, and outputs a master public key* pk[1] *and a master secret key* msk*. The master public key* pk *is assumed to be part of the input of all the remaining algorithms.*

- $\mathsf{Enc}(\mathsf{msk}, i, \mathbf{x}_i)$ *takes as input the master secret key* msk*, an index $i \in [n]$, and a message $\mathbf{x}_i \in \mathcal{X}_i$, and it outputs a ciphertext* ct*. Each ciphertext is assumed to be associated with an index i denoting for which slot this ciphertext can be used for. When $n = 1$, the input i is omitted.*

- $\mathsf{KeyGen}(\mathsf{msk}, f)$ *takes as input the master secret key* msk *and a function $f \in \mathcal{F}_n$, and it outputs a decryption key* sk_f*.*

- $\mathsf{Dec}(\mathsf{sk}_f, \mathsf{ct}_1, \ldots, \mathsf{ct}_n)$ *takes as input a decryption key* sk_f *for function f and n ciphertexts, and it outputs a value $y \in \mathcal{Y}$.*

Correctness. A scheme $\mathcal{MIFE}$ as defined above is correct if for all $n \in \mathbb{N}$, $f \in \mathcal{F}_n$ and all $\mathbf{x}_i \in \mathcal{X}_i$ for $1 \le i \le n$, we have

$$\Pr\left[\begin{array}{c}(\mathsf{pk}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda, \mathcal{F}_n); \quad \mathsf{sk}_f \leftarrow \mathsf{KeyGen}(\mathsf{msk}, f); \\ \mathsf{Dec}(\mathsf{sk}_f, \mathsf{Enc}(\mathsf{msk}, 1, \mathbf{x}_1), \ldots, \mathsf{Enc}(\mathsf{msk}, n, \mathbf{x}_n)) = f(\mathbf{x}_1, \ldots, \mathbf{x}_n)\end{array}\right] = 1,$$

where the probability is taken over the coins of Setup, KeyGen and Enc.

Security. In order to define the security of multi-input functional encryption schemes, we consider several security experiments depending on whether the adversary can ask one or many encryption queries and on whether it can has to choose the input on which it wishes to be challenged adaptively or at the very beginning of the experiment. These are denoted xx-AD-IND and xx-SEL-IND, where: xx $\in$\{one, many\}.

In the following, we first provide the definition of adaptive security under chosen-plaintext attacks (xx-AD-IND) followed by the definition of selective security under chosen-plaintext attacks (xx-SEL-IND).

one-AD-IND and many-AD-IND security experiments. For every multi-input functional encryption $\mathcal{MIFE}$ for $\mathcal{F}$, every stateful adversary $\mathcal{A}$, every security parameter $\lambda \in \mathbb{N}$, and every xx $\in$\{one, many\}, we define the following experiments for $\beta \in \{0, 1\}$:

Experiment $\textbf{xx-AD-IND}_\beta^{\mathcal{MIFE}}(1^\lambda, \mathcal{A})$:
$(\mathsf{pk}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda, \mathcal{F}_n)$ $\alpha \leftarrow \mathcal{A}^{\mathsf{KeyGen}(\mathsf{msk},\cdot),\mathsf{Enc}(\cdot,\cdot,\cdot)}(\mathsf{pk})$
Output: α

where Enc is an oracle that on input $(i, \mathbf{x}_i^0, \mathbf{x}_i^1)$ outputs $\mathsf{Enc}(\mathsf{msk}, i, \mathbf{x}_i^\beta)$. Also, $\mathcal{A}$ is restricted to only make queries f to $\mathsf{KeyGen}(\mathsf{msk}, \cdot)$ satisfying

$$f(\mathbf{x}_1^{j_1,0}, \ldots, \mathbf{x}_n^{j_n,0}) = f(\mathbf{x}_1^{j_1,1}, \ldots, \mathbf{x}_n^{j_n,1})$$

[1] In the private key setting, we think of pk as some public parameters common to all algorithms.

for all $j_1, \ldots, j_n \in [Q_1] \times \cdots \times [Q_n]$, where for all $i \in [n]$, Q_i denotes the number of encryption queries for input slot i. We denote by Q_f the number of key queries. Moreover, for all $i \in [n]$, $Q_i > 0$. When xx = one, $Q_i = 1$, for all $i \in [n]$.

Definition 3 (xx-AD-IND-secure MIFE) *For every xx $\in$ {one,many}, a private-key multi-input functional encryption $\mathcal{MIFE}$ for $\mathcal{F}$ is xx-AD-IND-secure if every PPT adversary $\mathcal{A}$ has advantage negligible in λ, where the advantage is defined as:*

$$\mathsf{Adv}^{\text{xx-AD-IND}}_{\mathcal{MIFE}}(\lambda, \mathcal{A}) = \left| \Pr\left[\textbf{xx-AD-IND}_0^{\mathcal{MIFE}}(1^\lambda, \mathcal{A}) = 1\right] - \Pr\left[\textbf{xx-AD-IND}_1^{\mathcal{MIFE}}(1^\lambda, \mathcal{A}) = 1\right] \right|$$

one-SEL-IND and many-SEL-IND security experiments. For every multi-input functional encryption $\mathcal{MIFE}$ for $\mathcal{F}$, every stateful adversary $\mathcal{A}$, every security parameter $\lambda \in \mathbb{N}$, and every xx $\in$ {one,many}, we define the following experiments for $\beta \in \{0, 1\}$:

Experiment $\textbf{xx-SEL-IND}_\beta^{\mathcal{MIFE}}(1^\lambda, \mathcal{A})$:

$\{\mathbf{x}_i^{j,b}\}_{i\in[n],j\in[Q_i],b\in\{0,1\}} \leftarrow \mathcal{A}(1^\lambda, \mathcal{F}_n)$
$(\mathsf{pk}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda, \mathcal{F}_n)$
$\mathsf{ct}_i^j := \mathsf{Enc}(\mathsf{msk}, \mathbf{x}_i^{j,\beta})$
$\alpha \leftarrow \mathcal{A}^{\mathsf{KeyGen}(\mathsf{msk},\cdot)}\left(\mathsf{pk}, \{\mathsf{ct}_i^j\}_{i\in[n],j\in[Q_i]}\right)$
Output: α

where $\mathcal{A}$ is restricted to only make queries f to $\mathsf{KeyGen}(\mathsf{msk}, \cdot)$ satisfying

$$f(\mathbf{x}_1^{j_1,0}, \ldots, \mathbf{x}_n^{j_n,0}) = f(\mathbf{x}_1^{j_1,1}, \ldots, \mathbf{x}_n^{j_n,1})$$

for all $j_1, \ldots, j_n \in [Q_1] \times \cdots \times [Q_n]$. When xx = one, we also require that $Q_i = 1$, for all $i \in [n]$.

Definition 4 (xx-SEL-IND-secure MIFE) *A $\mathcal{MIFE}$ for $\mathcal{F}$ is xx-SEL-IND-secure if every PPT adversary $\mathcal{A}$ has negligible advantage in λ, where the advantage is defined as:*

$$\mathsf{Adv}^{\text{xx-SEL-IND}}_{\mathcal{MIFE},\mathcal{A}}(\lambda) = \left| \Pr\left[\textbf{xx-SEL-IND}_0^{\mathcal{MIFE}}(1^\lambda, \mathcal{A}) = 1\right] - \Pr\left[\textbf{xx-SEL-IND}_1^{\mathcal{MIFE}}(1^\lambda, \mathcal{A}) = 1\right] \right|.$$

2.4 Inner-product functionality

In this report, we describe schemes that support the following two variants of the multi-input inner-product functionality:

Multi-Input Inner Products over $\mathbb{Z}_L$. This is a family of functions that is defined as $\mathcal{F}^m_{L,n} = \{f_{\mathbf{y}_1,\ldots,\mathbf{y}_n} : (\mathbb{Z}_L^m)^n \to \mathbb{Z}_L, \text{ for } \mathbf{y}_i \in \mathbb{Z}_L^m\}$ where

$$f_{\mathbf{y}_1,\ldots,\mathbf{y}_n}(\mathbf{x}_1, \ldots, \mathbf{x}_n) = \sum_{i=1}^n \langle \mathbf{x}_i, \mathbf{y}_i \rangle \bmod L.$$

Multi-Input Bounded-Norm Inner Products over $\mathbb{Z}$. This is defined as $\mathcal{F}^{m,X,Y}_n = \{f_{\mathbf{y}_1,\ldots,\mathbf{y}_n} : (\mathbb{Z}^m)^n \to \mathbb{Z}\}$ where $f_{\mathbf{y}_1,\ldots,\mathbf{y}_n}(\mathbf{x}_1, \ldots, \mathbf{x}_n)$ is the same as above except that the result is not reduced mod$L$, and vectors are required to satisfy the following bounds: $\|\mathbf{x}\|_\infty < X$, $\|\mathbf{y}\|_\infty < Y$.

3 Quantum-Safe Multi-Input Functional Encryption

In this section, we recall the multi-input functional encryption (MIFE) schemes proposed by Abdalla et al. in [3] for the inner-product functionality. The two constructions in [3] are generic, building a MIFE for inner-product functionality starting from any single-input FE ($\mathsf{Setup}, \mathsf{Enc}$, $\mathsf{KeyGen}, \mathsf{Dec}$) for the same functionality. While the first one addresses FE schemes that compute the inner-product functionality over a finite ring $\mathbb{Z}_L$ for some integer L, the second transformation addresses FE schemes for bounded-norm inner products. The two constructions are almost the same, and the only difference is that in the case of bounded-norm inner products, additional structural properties on the single-input FE are required. The main idea behind both constructions is to first build a simple MIFE scheme with unconditional one-time security and then use single-input FE in order to bootstrap the information-theoretic MIFE from one-time to many-time security.

Before proceeding with the actual description of the scheme, we provide a technical overview of the MIFE construction by Abdalla et al. [4] in Section 3.1.

3.1 Overview of the MIFE construction by Abdalla et al. [4]

To better understand the constructions in [3], let us first explain the basic idea behind the MIFE scheme by Abdalla et al. [4]. Informally, the latter builds upon a clever two-step decryption blueprint. The ciphertexts $\mathsf{ct}_1 = \mathsf{Enc}(\mathbf{x}_1), \ldots, \mathsf{ct}_n = \mathsf{Enc}(\mathbf{x}_n)$ (corresponding to slots $1, \ldots, n$) are all created using different instances of a single-input FE. Decryption is performed in two stages. One first decrypts each single ct_i separately using the secret key $\mathsf{sk}_{\mathbf{y}_i}$ of the underlying single-input FE, and then the outputs of these decryptions are added up to get the final result.

The main technical challenge of this approach is that the stage one of the above decryption algorithm leaks information on each partial inner product $\langle \mathbf{x}_i, \mathbf{y}_i \rangle$. To avoid this leakage, their idea is to let source i encrypt its plaintext vector $\mathbf{x}_i$ augmented with some fixed (random) value u_i, which is part of the secret key. Moreover, $\mathsf{sk}_{\mathbf{y}_i}$ are built by running the single-input FE key generation algorithm on input $\mathbf{y}_i || r$, i.e., the vector $\mathbf{y}_i$ augmented with fresh randomness r.

By these modifications, stage-one decryption then consists of using pairings to compute the values[2] $[\langle \mathbf{x}_i, \mathbf{y}_i \rangle + u_i r]_T$ for every slot i. From these quantities, the result $[\langle \mathbf{x}, \mathbf{y} \rangle]_T$ is obtained as

$$\prod_{i=1}^{n} [\langle \mathbf{x}_i, \mathbf{y}_i \rangle + u_i r]_T \cdot [-(\sum_{i=1}^{n} u_i) r]_T,$$

which can be easily computed if $[-(\sum_{i=1}^{n} u_i) r]_T$ is included in the secret key.

Intuitively, the scheme is secure as the quantities $[u_i r]_T$ are all pseudorandom (under the DDH assumption) and thus hide all the partial information $[\langle \mathbf{x}_i, \mathbf{y}_i \rangle + u_i r]_T$ may leak. Notice that, in order for this argument to go through, it is crucial that the quantities $[\langle \mathbf{x}_i, \mathbf{y}_i \rangle + u_i r]_T$ are all encoded in the exponent, and thus decoding is possible only for small norm exponents. Furthermore, this technique seems to inherently require pairings, as both u_i and r have to remain hidden while allowing to compute an encoding of their product at decryption time.

Abdalla et al. [3] overcome these difficulties via a new FE to MIFE transform, which manages to avoid leakage in a much simpler and efficient way. The transformation works in two steps. First,

[2]Here we implicitly adopt the bracket notation from [17] (see Section 2.1).

it considers a simplified scheme where only one ciphertext query is allowed and messages live in the ring $\mathbb{Z}_L$, for some integer L. In this setting, it builds the following multi-input scheme. For each slot i the (master) secret key for slot i consists of one random vector $\mathbf{u}_i \in \mathbb{Z}_L^m$. Encrypting $\mathbf{x}_i$ merely consists in computing $\mathbf{c}_i = \mathbf{x}_i + \mathbf{u}_i \bmod L$. The secret key for function $\mathbf{y} = (\mathbf{y}_1, \ldots, \mathbf{y}_n)$, is just $z_{\mathbf{y}} = \sum_{i=1}^{n} \langle \mathbf{u}_i, \mathbf{y}_i \rangle \bmod L$. To decrypt, one computes

$$\langle \mathbf{x}, \mathbf{y} \rangle \bmod L = \langle (\mathbf{c}_1, \ldots, \mathbf{c}_n), \mathbf{y} \rangle - z_{\mathbf{y}} \bmod L$$

Security comes from the fact that, if only one ciphertext query is allowed, the above can be seen as the functional encryption equivalent of the one-time pad.

Next, to guarantee security in the more challenging setting where many ciphertext queries are allowed, the scheme just adds a layer of (functional) encryption on top of the above one-time encryption. More specifically, it encrypts each $\mathbf{c}_i$ using a FE (supporting inner products) that is both linearly homomorphic and whose message space is compatible with L. So, given ciphertexts $\{\mathsf{ct}_i = \mathsf{Enc}(\mathbf{c}_i)\}$ and secret key $\mathsf{sk}_{\mathbf{y}} = (\{\mathsf{sk}_{\mathbf{y}_i}\}_i, z_{\mathbf{y}})$, one can first obtain $\{\langle \mathbf{c}_i, \mathbf{y}_i \rangle = \mathsf{Dec}(\mathsf{ct}_i, \mathsf{sk}_{\mathbf{y}_i})\}$, and then extract the result as $\langle \mathbf{x}, \mathbf{y} \rangle = \sum_{i=1}^{n} \langle \mathbf{c}_i, \mathbf{y}_i \rangle - \langle \mathbf{u}, \mathbf{y} \rangle$.

The transformation actually comes in two flavors: the first one addresses the case where the underlying FE computes inner products over some finite ring $\mathbb{Z}_L$; the second one instead considers FE schemes that compute bounded-norm inner products over the integers. In both cases the transformations are generic enough to be instantiated with known single-input FE schemes for inner products. Moreover, the proposed transform is security-preserving in the sense that, if the underlying FE achieves adaptive security, so does our resulting MIFE.

3.2 Information-Theoretic MIFE with One-Time Security

Figure 1 describes the multi-input scheme $\mathcal{MIFE}^{\mathsf{ot}}$ for the class $\mathcal{F}_{L,n}^m$. As shown in [3], this scheme can be easily shown to achieve unconditional one-time security (i.e., one-AD-IND security).

$\underline{\mathsf{Setup}^{\mathsf{ot}}(1^\lambda, \mathcal{F}_{L,n}^m)}$: For all $i \in [n]$, $\mathbf{u}_i \xleftarrow{R} \mathbb{Z}_L^m$ Return $\mathbf{u} = \{\mathbf{u}_i\}_{i \in [n]}$ $\underline{\mathsf{Enc}^{\mathsf{ot}}(\mathbf{u}, i, \mathbf{x}_i)}$: Return $\mathbf{x}_i + \mathbf{u}_i \bmod L$	$\underline{\mathsf{KeyGen}^{\mathsf{ot}}(\mathbf{u}, \mathbf{y}_1 \| \cdots \| \mathbf{y}_n)}$: Return $z := \sum_{i \in [n]} \langle \mathbf{u}_i, \mathbf{y}_i \rangle \bmod L$ $\underline{\mathsf{Dec}^{\mathsf{ot}}(z, \mathsf{ct}_1, \ldots, \mathsf{ct}_n)}$: Return $\sum_{i=1}^{n} \langle \mathsf{ct}_i, \mathbf{y}_i \rangle - z \bmod L$

Figure 1: Private-key, information theoretically secure, multi-input FE scheme $\mathcal{MIFE}^{\mathsf{ot}} = (\mathsf{Setup}^{\mathsf{ot}}, \mathsf{Enc}^{\mathsf{ot}}, \mathsf{KeyGen}^{\mathsf{ot}}, \mathsf{Dec}^{\mathsf{ot}})$ for the class $\mathcal{F}_{L,n}^m$ [3].

3.3 Multi-Input Inner Products over $\mathbb{Z}_L$

Figure 2 presents the multi-input scheme $\mathcal{MIFE}$ for the class $\mathcal{F}_{L,n}^m$ from [3]. The construction relies on the one-time scheme $\mathcal{MIFE}^{\mathsf{ot}}$ in Figure 1, and any single-input FE for the class $\mathcal{F}_{L,1}^m$.

Correctness. The correctness of $\mathcal{MIFE}$ follows from the correctness properties of the single-input scheme $\mathcal{FE}$ and the multi-input scheme $\mathcal{MIFE}^{\mathsf{ot}}$. More precisely, the correctness of the single-input scheme $\mathcal{FE}$ first implies that, for all input slots $i \in [n]$, $D_i = \langle \mathbf{w}_i, \mathbf{y}_i \rangle$ mod

$\underline{\mathsf{Setup}'(1^\lambda, \mathcal{F}^m_{L,n})}$:
$\mathbf{u} \leftarrow \mathsf{Setup}^{\mathsf{ot}}(1^\lambda, \mathcal{F}^m_{L,n})$, for all $i \in [n]$, $(\mathsf{pk}_i, \mathsf{msk}_i) \leftarrow \mathsf{Setup}(1^\lambda, \mathcal{F}^m_{L,1})$
$(\mathsf{pk}, \mathsf{msk}) := \big(\{\mathsf{pk}_i\}_{i\in[n]}, (\{\mathsf{msk}_i, \}_{i\in[n]}, \mathbf{u})\big)$
Return $(\mathsf{pk}, \mathsf{msk})$

$\underline{\mathsf{Enc}'(\mathsf{msk}, i, \mathbf{x}_i)}$:
$\mathbf{w}_i := \mathsf{Enc}^{\mathsf{ot}}(\mathbf{u}, i, \mathbf{x}_i)$
Return $\mathsf{Enc}(\mathsf{msk}_i, \mathbf{w}_i)$

$\underline{\mathsf{KeyGen}'(\mathsf{msk}, \mathbf{y}_1 \| \cdots \| \mathbf{y}_n)}$:
For all $i \in [n]$, $\mathsf{sk}_i \leftarrow \mathsf{KeyGen}(\mathsf{msk}_i, \mathbf{y}_i)$, $z := \mathsf{KeyGen}^{\mathsf{ot}}(\mathbf{u}, \mathbf{y}_1 \| \cdots \| \mathbf{y}_n)$
$\mathsf{sk}_{\mathbf{y}_1\|\cdots\|\mathbf{y}_n} := \big(\{\mathsf{sk}_i\}_{i\in[n]}, z\big)$
Return $\mathsf{sk}_{\mathbf{y}_1\|\cdots\|\mathbf{y}_n}$

$\underline{\mathsf{Dec}'\big((\{\mathsf{sk}_i\}_{i\in[n]}, z), \mathsf{ct}_1, \ldots, \mathsf{ct}_n\big)}$:
For all $i \in [n]$, $D_i \leftarrow \mathsf{Dec}(\mathsf{sk}_i, \mathsf{ct}_i)$
Return $\sum_{i\in[n]} D_i - z \bmod L$

Figure 2: Private-key multi-input FE scheme $\mathcal{MIFE} := (\mathsf{Setup}', \mathsf{Enc}', \mathsf{KeyGen}', \mathsf{Dec}')$ for the class $\mathcal{F}^m_{L,n}$ from a public-key single-input FE $\mathcal{FE} := (\mathsf{Setup}, \mathsf{Enc}, \mathsf{KeyGen}, \mathsf{Dec})$ for the class $\mathcal{F}^m_{L,1}$, and one-time multi-input FE $\mathcal{MIFE}^{\mathsf{ot}} = (\mathsf{Setup}^{\mathsf{ot}}, \mathsf{Enc}^{\mathsf{ot}}, \mathsf{KeyGen}^{\mathsf{ot}}, \mathsf{Dec}^{\mathsf{ot}})$ for the class $\mathcal{F}^m_{L,n}$ [3].

L. Next, the correctness of $\mathcal{MIFE}^{\mathsf{ot}}$ implies that $\sum_{i\in[n]} D_i - z = \mathsf{Dec}^{\mathsf{ot}}(z, \mathbf{w}_1, \ldots, \mathbf{w}_n) = \sum_{i\in[n]} \langle \mathbf{x}_i, \mathbf{y}_i \rangle \bmod L$.

Security. The security of $\mathcal{MIFE}$ follows from the following theorem, whose proof is given in [3]:

Theorem 1 *If the single-input FE, $\mathcal{FE}$ is many-AD-IND-secure, and the multi-input scheme $\mathcal{MIFE}^{\mathsf{ot}}$ is one-AD-IND-secure, then the multi-input FE, $\mathcal{MIFE}$, described in Figure 2, is many-AD-IND-secure.*

Instantiations. The construction in Figure 2 can be instantiated using the single-input LWE-based FE scheme of Agrawal, Libert, and Stehlé [5, Section 4.2] that is many-AD-IND-secure and allows for computing inner products over a finite ring. This results in a MIFE for inner products over $\mathbb{Z}_p$ for a prime p, based on the LWE assumption. As the scheme in [5], the resulting MIFE scheme has a stateful key generation. A stateless MIFE instantiation can be obtained from the transformation in the next section.

Another possible instantiation is to use the single-input LWE-based FE scheme of Abdalla et al. [1].

3.4 Multi-Input Inner Products over $\mathbb{Z}$

Figure 3 presents a multi-input scheme $\mathcal{MIFE}$ in [3] for the class $\mathcal{F}^{m,X,Y}_n$ from the one-time scheme $\mathcal{MIFE}^{\mathsf{ot}}$ of Figure 1, and a (single-input) scheme $\mathcal{FE}$ for the class $\mathcal{F}^{m,3X,Y}_1$. For the transformation to work, $\mathcal{FE}$ is required to satisfy two properties. The first one, called *two-step*

$\underline{\mathsf{Setup}'(1^\lambda, \mathcal{F}_n^{m,X,Y})\text{:}}$
$\mathbf{u} \leftarrow \mathsf{Setup}^{\mathsf{ot}}(1^\lambda, \mathcal{F}_{L,n}^{m})$, for all $i \in [n]$, $(\mathsf{pk}_i, \mathsf{msk}_i) \leftarrow \mathsf{Setup}^{\star}(1^\lambda, \mathcal{F}_1^{m,3X,Y}, 1^n)$
$(\mathsf{pk}, \mathsf{msk}) := \big(\{\mathsf{pk}_i\}_{i\in[n]}, (\{\mathsf{msk}_i, \}_{i\in[n]}, \mathbf{u})\big)$
Return $(\mathsf{pk}, \mathsf{msk})$

$\underline{\mathsf{Enc}'(\mathsf{msk}, i, \mathbf{x}_i)\text{:}}$
$\mathbf{w}_i := \mathsf{Enc}^{\mathsf{ot}}(\mathbf{u}, i, \mathbf{x}_i)$
Return $\mathsf{Enc}(\mathsf{msk}_i, \mathbf{w}_i)$

$\underline{\mathsf{KeyGen}'(\mathsf{msk}, \mathbf{y}_1 \| \cdots \| \mathbf{y}_n)\text{:}}$
For all $i \in [n]$, $\mathsf{sk}_i \leftarrow \mathsf{KeyGen}(\mathsf{msk}_i, \mathbf{y}_i)$, $z \leftarrow \mathsf{KeyGen}^{\mathsf{ot}}(\mathbf{u}, \mathbf{y}_1 \| \cdots \| \mathbf{y}_n)$
$\mathsf{sk}_{\mathbf{y}_1\|\cdots\|\mathbf{y}_n} := \big(\{\mathsf{sk}_i\}_{i\in[n]}, z\big)$
Return $\mathsf{sk}_{\mathbf{y}_1\|\cdots\|\mathbf{y}_n}$

$\underline{\mathsf{Dec}'\big((\{\mathsf{sk}_i\}_{i\in[n]}, z), \mathsf{ct}_1, \ldots, \mathsf{ct}_n)\text{:}}$
For all $i \in [n]$, $\mathcal{E}(\langle \mathbf{x}_i + \mathbf{u}_i, \mathbf{y}_i\rangle \bmod L, \mathsf{noise}_i) \leftarrow \mathsf{Dec}_1(\mathsf{sk}_i, \mathsf{ct}_i)$
Return $\mathsf{Dec}_2\big(\mathcal{E}(\langle \mathbf{x}_1 + \mathbf{u}_1, \mathbf{y}_1\rangle \bmod L, \mathsf{noise}_1) \circ \cdots \circ \mathcal{E}(\langle \mathbf{x}_n + \mathbf{u}_n, \mathbf{y}_n\rangle \bmod L, \mathsf{noise}_n) \circ \mathcal{E}(-z, 0)\big)$

Figure 3: Private-key multi-input FE scheme $\mathcal{MIFE} = (\mathsf{Setup}', \mathsf{Enc}', \mathsf{KeyGen}', \mathsf{Dec}')$ for the class $\mathcal{F}_n^{m,X,Y}$ from public-key single-input FE scheme $\mathcal{FE} = (\mathsf{Setup}, \mathsf{Enc}, \mathsf{KeyGen}, \mathsf{Dec})$ for the class $\mathcal{F}_1^{m,X,Y}$ and one-time multi-input FE $\mathcal{MIFE}^{\mathsf{ot}} = (\mathsf{Setup}^{\mathsf{ot}}, \mathsf{Enc}^{\mathsf{ot}}, \mathsf{KeyGen}^{\mathsf{ot}}, \mathsf{Dec}^{\mathsf{ot}})$ [3].

decryption, intuitively says that the $\mathcal{FE}$ decryption algorithm works in two steps: the first step uses the secret key to output an encoding of the result, while the second step returns the actual result $\langle \mathbf{x}, \mathbf{y}\rangle$ provided that the bounds $\|\mathbf{x}\|_\infty < X$, $\|\mathbf{y}\|_\infty < Y$ hold. The second property, called *linear encryption*, informally says that the $\mathcal{FE}$ encryption algorithm is additively homomorphic.

Correctness. As shown in [3], the correctness of the scheme $\mathcal{MIFE}$ follows from (i) the correctness and the two-step decryption property of the single-input FE scheme, and (ii) from the correctness of $\mathcal{MIFE}^{\mathsf{ot}}$ and the linear property of its decryption algorithm $\mathsf{Dec}^{\mathsf{ot}}$.

Security. As the following theorem from [3] shows, the security of the $\mathcal{MIFE}$ scheme in Figure 3 follows from the security of the underlying single-input FE scheme and that of the one-time scheme $\mathcal{MIFE}^{\mathsf{ot}}$.

Theorem 2 *Assume that the single-input FE is many-AD-IND-secure and the multi-input FE $\mathcal{MIFE}^{\mathsf{ot}}$ is one-AD-IND-secure. Then the multi-input FE $\mathcal{MIFE}$ in Figure 3 is many-AD-IND-secure.*

Instantiations. In [3], the authors show that the two additional properties are satisfied by the many-AD-IND secure FE schemes of Agrawal, Libert and Stehlé [5]. As a result, by instantiating the above construction with their LWE-based single-input FE scheme and recalled in Section 4.1, one can obtain a quantum-safe MIFE scheme for bounded-norm inner products based on LWE. In addition, the decryption algorithm of the resulting scheme also works efficiently for large outputs. This stands in contrast to the previous result [4], where decryption requires to extract discrete logarithms.

4 LWE Instantiations

In this section, we recall the description of two LWE-based (single-input) FE schemes which can be used to instantiate the MIFE schemes in Section 3.The first one is by Agrawal et al. [5, Section 4.1] and the second one is by Abdalla et al. [2].

4.1 Inner-product functional encryption from [5]

The many-AD-IND secure Inner-Product FE by Agrawal et al. [5, Section 4.1] is recalled in Fig. 4. The proof that it satisfies the two-step decryption and the linear encryption properties can be found in [3].

$\underline{\mathsf{Setup}(1^\lambda, \mathcal{F}_1^{m,X,Y})\text{:}}$
Let $N = N(\lambda)$, and set integers $M, q \geq 2$, real $\alpha \in (0,1)$, and distribution D over $\mathbb{Z}^{m\times M}$ as explained in [5]; set $K := m \cdot X \cdot Y$, $\mathbf{A} \xleftarrow{R} \mathbb{Z}_q^{M\times N}$, $\mathbf{Z} \xleftarrow{R} \mathsf{D}$, $\mathbf{U} := \mathbf{ZA} \in \mathbb{Z}_q^{m\times N}$, $\mathsf{pk} := (K, \mathbf{A}, \mathbf{U})$, $\mathsf{msk} := \mathbf{Z}$.
Return $(\mathsf{pk}, \mathsf{msk})$

$\underline{\mathsf{Enc}(\mathsf{pk}, \mathbf{x} \in \mathbb{Z}^m)\text{:}}$
$\mathbf{s} \xleftarrow{R} \mathbb{Z}_q^N$, $\mathbf{e}_0 \xleftarrow{R} D_{\mathbb{Z},\alpha q}^M$, $\mathbf{e}_1 \xleftarrow{R} D_{\mathbb{Z},\alpha q}^m$
$\mathbf{c}_0 := \mathbf{As} + \mathbf{e}_0 \in \mathbb{Z}_q^M$
$\mathbf{c}_1 := \mathbf{Us} + \mathbf{e}_1 + \mathbf{x} \cdot \lfloor \frac{q}{K} \rfloor \in \mathbb{Z}_q^m$
Return $\mathsf{ct}_{\mathbf{x}} := (\mathbf{c}_0, \mathbf{c}_1)$

$\underline{\mathsf{KeyGen}(\mathsf{msk}, \mathbf{y} \in \mathbb{Z}^m)\text{:}}$
Return $\mathsf{sk}_{\mathbf{y}} := \begin{pmatrix} \mathbf{Z}^\top \mathbf{y} \\ \mathbf{y} \end{pmatrix} \in \mathbb{Z}^{M+m}$

$\underline{\mathsf{Dec}(\mathsf{sk}_{\mathbf{y}}, \mathsf{ct}_{\mathbf{x}})\text{:}}$
$\mu' := \begin{pmatrix} \mathbf{c}_0 \\ \mathbf{c}_1 \end{pmatrix}^\top \mathsf{sk}_{\mathbf{y}} \mod q.$
Return $\mu \in \{-K+1, \dots, K-1\}$ that minimizes $\left|\lfloor \frac{q}{K} \rfloor \mu - \mu'\right|$.

Figure 4: Functional encryption scheme by Agrawal et al. [5] for the class $\mathcal{F}_1^{m,X,Y}$ based on the LWE assumption.

4.2 Inner-product functional encryption from [2]

The inner-product FE scheme by Abdalla, Bourse, De Caro, and Pointcheval in [2] is an extension of inner-product FE construction in [1]. It achieves adaptive security and has instantiations based on the ElGamal (plain DDH assumption) [16], Paillier/BCP (DCR assumption) [13], and Regev (LWE assumption) [22] encryption schemes.

Fig. 5 describes the instantiation based on the Regev encryption scheme [22]. The proof that it satisfies the two-step decryption and linear encryption properties is similar to the one for the [5] scheme given in [3].

$\underline{\mathsf{Setup}(1^\lambda, \mathcal{F}_1^{m,X,Y})}$:
Let n, m, p, q be integer parameters
Let σ a positive real parameter such that they verify the conditions required
Let $\mathbf{A} \xleftarrow{R} \mathbb{Z}_q^{m\times n}$ be a uniformly random matrix.
Set $\mathsf{p} = (\lambda, \ell, n, m, p, q, \mathbf{A})$
Sample $(\mathbf{s}_0, \mathbf{e}_0) \xleftarrow{R} \mathbb{Z}_q^n \times \chi_{\gamma_0}^m$
Sample $\mathbf{b}_0 \leftarrow \mathbf{A}\mathbf{s}_0 + \mathbf{e}_0 \in \mathbb{Z}_q^m$
For all $i \in [\ell]$, set $(t_i, \mathbf{s}_i, \mathbf{e}_i) \xleftarrow{R} \{0, \ldots, T\} \times \mathbb{Z}_q^n \times \chi_\sigma^m$
For all $i \in [\ell]$, set $\mathbf{b}_i \leftarrow \mathbf{A}(t_i \cdot \mathbf{s}_0 + \mathbf{s}_i) + \mathbf{e}_i \in \mathbb{Z}_q^m$
$\mathsf{msk} = (\mathbf{s}_i, t_i)_{i\in[\ell]}$
$\mathsf{pk} = (\mathbf{b}_0, \mathbf{b}_i)_{i\in[\ell]}$
Return $(\mathsf{pk}, \mathsf{msk})$

$\underline{\mathsf{Enc}(\mathsf{pk}, \mathbf{x} \in \mathcal{M}_x)}$:
Pick $\mathbf{r} \xleftarrow{R} \{0,1\}^m$
Set $\mathsf{ct}_0 \leftarrow \mathbf{A}^\intercal \mathbf{r} \in \mathbb{Z}_q^n$
Set $\mathsf{ct}_1 \leftarrow \mathbf{b}_0^\intercal \mathbf{r} \in \mathbb{Z}_q$
For all $i \in [\ell]$, $\mathsf{ct}_{2,i} \leftarrow \mathbf{b}_i^\intercal \mathbf{r} + t(x_i) \in \mathbb{Z}_q$, where $t(v) = v \cdot \lfloor q/p \rfloor \in \mathbb{Z}_q$.
Return $\mathsf{ct}_\mathbf{x} = (\mathsf{ct}_0, \mathsf{ct}_1, (\mathsf{ct}_{2,i})_{i\in[\ell]})$

$\underline{\mathsf{KeyGen}(\mathsf{msk}, \mathbf{y} \in \mathcal{M}_y)}$:
Set $\mathbf{s}_\mathbf{y} \leftarrow \sum_{i\in[\ell]} y_i \mathbf{s}_i \in \mathbb{Z}_q^n$
Set $t_\mathbf{y} \leftarrow \sum_{i\in[\ell]} y_i t_i \in \mathbb{Z}$
Return $\mathsf{sk}_\mathbf{y} = (\mathbf{s}_\mathbf{y}, t_\mathbf{y})$

$\underline{\mathsf{Dec}(\mathsf{sk}_\mathbf{y}, \mathsf{ct}_\mathbf{x})}$:
Set $\mathsf{ct}_{\langle \mathbf{x},\mathbf{y}\rangle} \leftarrow \sum_{i\in[\ell]} y_i \mathsf{ct}_{2,i} - t_\mathbf{y}\mathsf{ct}_1 - \mathsf{ct}_0^\intercal \mathsf{sk}_\mathbf{y} \in \mathbb{Z}_q$.
Return the plaintext m, where m is such that $d - t(m) \in \mathbb{Z}_q$ is closest to $0 \mod q$.

Figure 5: Functional encryption scheme by Abdalla et al. [2] for the class $\mathcal{F}_1^{m,X,Y}$ based on the LWE assumption.

According to [2], the message space is $\mathcal{M}_x = \{0, \ldots, M_x\} \subseteq \mathbb{Z}_p$ for some integer M_x and prime $p > \ell M_x M_y$. $\mathcal{T} = \{0, \ldots, T\}^\ell$, where T is set according to the security properties needed. T/M_x super-polynomial is needed for security against polynomially bounded adversaries, T/M_x exponential provides security against sub-exponentially bounded adversaries, where M_x is the biggest possible coordinate of any vector in $\mathcal{M}_x$.

In order for the proof of security to carry through, as well as the correctness, the following properties on the parameters have to be verified:

1. $m \geq (n + \ell + 2)\log q + 2\log\frac{1}{\epsilon} + \Omega(1)$;
2. $T = M_x \cdot \lambda^{\omega(1)}$;
3. $\sigma \geq (1 + T\sqrt{\ell})\sigma'$;
4. $\gamma_0 > \sqrt{\frac{\ln(2\ell(1+1/\epsilon))}{\pi}}$;

5. $\sigma' q > 2\sqrt{n}$;
6. $p > \ell M_x M_y$;
7. $\frac{q}{2p} > \sigma M_y^2 \ell \sqrt{2m\lambda}$.

5 Conclusion

In this document, we described the first specifications of quantum-safe functional encryption developed in the context of the FENTEC project. In particular, the new multi-input functional encryption construction for the inner-product functionality described in Section 3 was the first such scheme based on lattice problems and capable of handling inputs of arbitrary size. In addition to the quantum-safe instantiations in Section 4, we remark that other instantiations are also possible such as the LWE-based scheme by Abdalla et al. [1].

However, as stated in Deliverable 4.1, the use of a central authority in multi-input functional encryption schemes can make them not suitable for certain applications, such as the web analytics use case considered in WP7. Hence, an important open problem is to design a decentralized version of such schemes based on lattices. We currently have some preliminary results in this direction and we expect to be able to present in next corresponding deliverable for the second year of the project.

References

[1] Michel Abdalla, Florian Bourse, Angelo De Caro, and David Pointcheval. Simple functional encryption schemes for inner products. In Jonathan Katz, editor, *PKC 2015: 18th International Conference on Theory and Practice of Public Key Cryptography*, volume 9020 of *Lecture Notes in Computer Science*, pages 733–751, Gaithersburg, MD, USA, March 30 – April 1, 2015. Springer, Heidelberg, Germany. doi:10.1007/978-3-662-46447-2_33. (Pages 1, 8, 10, and 13.)

[2] Michel Abdalla, Florian Bourse, Angelo De Caro, and David Pointcheval. Better security for functional encryption for inner product evaluations. Cryptology ePrint Archive, Report 2016/011, 2016. http://eprint.iacr.org/2016/011. (Pages ii, iii, 2, 10, and 11.)

[3] Michel Abdalla, Dario Catalano, Dario Fiore, Romain Gay, and Bogdan Ursu. Multi-input functional encryption for inner products: Function-hiding realizations and constructions without pairings. In Hovav Shacham and Alexandra Boldyreva, editors, *Advances in Cryptology – CRYPTO 2018, Part I*, volume 10991 of *Lecture Notes in Computer Science*, pages 597–627, Santa Barbara, CA, USA, August 19–23, 2018. Springer, Heidelberg, Germany. doi:10.1007/978-3-319-96884-1_20. (Pages iii, v, 2, 6, 7, 8, 9, and 10.)

[4] Michel Abdalla, Romain Gay, Mariana Raykova, and Hoeteck Wee. Multi-input inner-product functional encryption from pairings. In Jean-Sébastien Coron and Jesper Buus Nielsen, editors, *Advances in Cryptology – EUROCRYPT 2017, Part I*, volume 10210 of *Lecture Notes in Computer Science*, pages 601–626, Paris, France, April 30 – May 4, 2017. Springer, Heidelberg, Germany. doi:10.1007/978-3-319-56620-7_21. (Pages ii, 2, 6, and 9.)

[5] Shweta Agrawal, Benoît Libert, and Damien Stehlé. Fully secure functional encryption for inner products, from standard assumptions. In Matthew Robshaw and Jonathan Katz, editors, *Advances in Cryptology – CRYPTO 2016, Part III*, volume 9816 of *Lecture Notes in Computer Science*, pages 333–362, Santa Barbara, CA, USA, August 14–18, 2016. Springer, Heidelberg, Germany. doi:10.1007/978-3-662-53015-3_12. (Pages ii, iii, 1, 2, 8, 9, and 10.)

[6] Prabhanjan Ananth and Abhishek Jain. Indistinguishability obfuscation from compact functional encryption. In Rosario Gennaro and Matthew J. B. Robshaw, editors, *Advances in Cryptology – CRYPTO 2015, Part I*, volume 9215 of *Lecture Notes in Computer Science*, pages 308–326, Santa Barbara, CA, USA, August 16–20, 2015. Springer, Heidelberg, Germany. doi:10.1007/978-3-662-47989-6_15. (Page 2.)

[7] Saikrishna Badrinarayanan, Divya Gupta, Abhishek Jain, and Amit Sahai. Multi-input functional encryption for unbounded arity functions. In Tetsu Iwata and Jung Hee Cheon, editors, *Advances in Cryptology – ASIACRYPT 2015, Part I*, volume 9452 of *Lecture Notes in Computer Science*, pages 27–51, Auckland, New Zealand, November 30 – December 3, 2015. Springer, Heidelberg, Germany. doi:10.1007/978-3-662-48797-6_2. (Page 2.)

[8] Carmen Elisabetta Zaira Baltico, Dario Catalano, Dario Fiore, and Romain Gay. Practical functional encryption for quadratic functions with applications to predicate encryption. In Jonathan Katz and Hovav Shacham, editors, *Advances in Cryptology – CRYPTO 2017, Part I*,

volume 10401 of *Lecture Notes in Computer Science*, pages 67–98, Santa Barbara, CA, USA, August 20–24, 2017. Springer, Heidelberg, Germany. doi:10.1007/978-3-319-63688-7_3. (Page 1.)

[9] Allison Bishop, Abhishek Jain, and Lucas Kowalczyk. Function-hiding inner product encryption. In Tetsu Iwata and Jung Hee Cheon, editors, *Advances in Cryptology – ASIACRYPT 2015, Part I*, volume 9452 of *Lecture Notes in Computer Science*, pages 470–491, Auckland, New Zealand, November 30 – December 3, 2015. Springer, Heidelberg, Germany. doi:10.1007/978-3-662-48797-6_20. (Page 1.)

[10] Dan Boneh, Amit Sahai, and Brent Waters. Functional encryption: Definitions and challenges. In Yuval Ishai, editor, *TCC 2011: 8th Theory of Cryptography Conference*, volume 6597 of *Lecture Notes in Computer Science*, pages 253–273, Providence, RI, USA, March 28–30, 2011. Springer, Heidelberg, Germany. doi:10.1007/978-3-642-19571-6_16. (Page 1.)

[11] Elette Boyle, Kai-Min Chung, and Rafael Pass. On extractability obfuscation. In Yehuda Lindell, editor, *TCC 2014: 11th Theory of Cryptography Conference*, volume 8349 of *Lecture Notes in Computer Science*, pages 52–73, San Diego, CA, USA, February 24–26, 2014. Springer, Heidelberg, Germany. doi:10.1007/978-3-642-54242-8_3. (Page 1.)

[12] Zvika Brakerski, Ilan Komargodski, and Gil Segev. Multi-input functional encryption in the private-key setting: Stronger security from weaker assumptions. *Journal of Cryptology*, 31(2):434–520, April 2018. doi:10.1007/s00145-017-9261-0. (Page 2.)

[13] Emmanuel Bresson, Dario Catalano, and David Pointcheval. A simple public-key cryptosystem with a double trapdoor decryption mechanism and its applications. In Chi-Sung Laih, editor, *Advances in Cryptology – ASIACRYPT 2003*, volume 2894 of *Lecture Notes in Computer Science*, pages 37–54, Taipei, Taiwan, November 30 – December 4, 2003. Springer, Heidelberg, Germany. doi:10.1007/978-3-540-40061-5_3. (Page 10.)

[14] Jérémy Chotard, Edouard Dufour Sans, Romain Gay, Duong Hieu Phan, and David Pointcheval. Decentralized multi-client functional encryption for inner product. In Thomas Peyrin and Steven Galbraith, editors, *Advances in Cryptology – ASIACRYPT 2018, Part II*, volume 11273 of *Lecture Notes in Computer Science*, pages 703–732, Brisbane, Queensland, Australia, December 2–6, 2018. Springer, Heidelberg, Germany. doi:10.1007/978-3-030-03329-3_24. (Page 2.)

[15] Pratish Datta, Ratna Dutta, and Sourav Mukhopadhyay. Functional encryption for inner product with full function privacy. In Chen-Mou Cheng, Kai-Min Chung, Giuseppe Persiano, and Bo-Yin Yang, editors, *PKC 2016: 19th International Conference on Theory and Practice of Public Key Cryptography, Part I*, volume 9614 of *Lecture Notes in Computer Science*, pages 164–195, Taipei, Taiwan, March 6–9, 2016. Springer, Heidelberg, Germany. doi:10.1007/978-3-662-49384-7_7. (Page 1.)

[16] Taher ElGamal. A public key cryptosystem and a signature scheme based on discrete logarithms. *IEEE Transactions on Information Theory*, 31:469–472, 1985. (Page 10.)

[17] Alex Escala, Gottfried Herold, Eike Kiltz, Carla Ràfols, and Jorge Villar. An algebraic framework for Diffie-Hellman assumptions. In Ran Canetti and Juan A. Garay, editors, *Advances in Cryptology – CRYPTO 2013, Part II*, volume 8043 of *Lecture Notes in Computer Science*, pages 129–147, Santa Barbara, CA, USA, August 18–22, 2013. Springer, Heidelberg, Germany. doi:10.1007/978-3-642-40084-1_8. (Pages 3 and 6.)

[18] Sanjam Garg, Craig Gentry, Shai Halevi, Mariana Raykova, Amit Sahai, and Brent Waters. Candidate indistinguishability obfuscation and functional encryption for all circuits. In *54th Annual Symposium on Foundations of Computer Science*, pages 40–49, Berkeley, CA, USA, October 26–29, 2013. IEEE Computer Society Press. doi:10.1109/FOCS.2013.13. (Page 1.)

[19] Sanjam Garg, Craig Gentry, Shai Halevi, and Mark Zhandry. Fully secure attribute based encryption from multilinear maps. Cryptology ePrint Archive, Report 2014/622, 2014. http://eprint.iacr.org/2014/622. (Page 1.)

[20] Shafi Goldwasser, S. Dov Gordon, Vipul Goyal, Abhishek Jain, Jonathan Katz, Feng-Hao Liu, Amit Sahai, Elaine Shi, and Hong-Sheng Zhou. Multi-input functional encryption. In Phong Q. Nguyen and Elisabeth Oswald, editors, *Advances in Cryptology – EUROCRYPT 2014*, volume 8441 of *Lecture Notes in Computer Science*, pages 578–602, Copenhagen, Denmark, May 11–15, 2014. Springer, Heidelberg, Germany. doi:10.1007/978-3-642-55220-5_32. (Pages 2 and 3.)

[21] Adam O'Neill. Definitional issues in functional encryption. Cryptology ePrint Archive, Report 2010/556, 2010. http://eprint.iacr.org/2010/556. (Page 1.)

[22] Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. In Harold N. Gabow and Ronald Fagin, editors, *37th Annual ACM Symposium on Theory of Computing*, pages 84–93, Baltimore, MA, USA, May 22–24, 2005. ACM Press. doi:10.1145/1060590.1060603. (Page 10.)

[23] Peter W. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM J. Comput.*, 26(5):1484–1509, 1997. (Pages v and 1.)

[24] Brent Waters. A punctured programming approach to adaptively secure functional encryption. In Rosario Gennaro and Matthew J. B. Robshaw, editors, *Advances in Cryptology – CRYPTO 2015, Part II*, volume 9216 of *Lecture Notes in Computer Science*, pages 678–697, Santa Barbara, CA, USA, August 16–20, 2015. Springer, Heidelberg, Germany. doi:10.1007/978-3-662-48000-7_33. (Page 1.)