An Anonymous Trace-and-Revoke Broadcast Encryption Scheme

Olivier Blazy[1], Sayantan Mukherjee[1], Huyen Nguyen[2], Duong Hieu Phan[4] and Damien Stehlé[2,3]

[1] XLIM, University of Limoges, CNRS, Limoges, France
[2] ENS de Lyon, Laboratoire LIP (U. Lyon, CNRS, ENSL, INRIA, UCBL), France
[3] Institut Universitaire de France.
[4] Telecom Paris, France

Abstract. Broadcast Encryption is a fundamental cryptographic primitive, that gives the ability to send a secure message to any chosen target set among registered users. In this work, we investigate broadcast encryption with anonymous revocation, in which ciphertexts do not reveal any information on which users have been revoked. We provide a scheme whose ciphertext size grows linearly with the number of revoked users. Moreover, our system also achieves traceability in the black-box confirmation model.

Technically, our contribution is threefold. First, we develop a generic transformation of linear functional encryption toward trace-and-revoke systems for 1-bit message space. It is inspired from the transformation by Agrawal *et al* (CCS'17) with the novelty of achieving anonymity. Our second contribution is to instantiate the underlying linear functional encryptions from standard assumptions. We propose a DDH-based construction which does no longer require discrete logarithm evaluation during the decryption and thus significantly improves the performance compared to the DDH-based construction of Agrawal *et al.* In the LWE-based setting, we tried to instantiate our construction by relying on the scheme from Wang *et al* (PKC'19) only to find an attack on this scheme. Our third contribution is to extend the 1-bit encryption from the generic transformation to n-bit encryption. By introducing matrix multiplication functional encryption, which essentially performs a fixed number of parallel calls on functional encryptions with the same randomness, we can prove the security of the final scheme with a tight reduction that does not depend on n, in contrast to employing the hybrid argument.

Keywords. Anonymity, Trace and Revoke, Functional Encryption.

1 Introduction

Trace-and-revoke systems, introduced in [23, 24] have been studied extensively in many works, including [4, 12, 15, 20, 26]. A trace-and-revoke system is a multi-recipient encryption scheme in which a content distributor can find malicious users and revoke their decryption capability. Note that a user might share its secret key with non-legitimate entity. In such a case, it should be possible to identify the user, so that it is revoked from further accessing new content. A traitor tracing system guarantees that if a coalition of users pool their secret keys to construct a pirate decoder box that can decrypt ciphertexts, then there is an efficient trace algorithm to find at least one guilty user provided the trace algorithm is given access to the decoder. Then the content distributor can use the revocation functionality to prohibit guilty users from accessing the data in future. A revocation system ensures that if a coalition of illegitimate users pools their secret keys, they still cannot decrypt the ciphertext. A natural question occurs if one can devise a protocol where a revoked user is not able to find out if it has been revoked. One may further request that, given a ciphertext, no legitimate user will get any information about the users who have been revoked.

Anonymity of receivers is important in numerous real-life applications and have been considered in multiple works, such as [7, 14, 17, 21, 22]. The standard notion of anonymity requires that the adversary cannot distinguish between ciphertexts of two targeted sets of its choice, even if it can corrupt any user in the intersection of these two sets or outside of the two sets. Unfortunately, it turned out to be extremely difficult to achieve this anonymity level in the general case without any restriction on the size of the target set. The state-of-the-art constructions by Barth *et al* [7] and Libert *et al* [22] start from a public-key encryption and result in schemes with ciphertext size which is N times larger, where N denotes the total number of users. Moreover, Kiayias and Samari [19] proved that ciphertext size will be linear in N in the general case.

For revoke systems, the efficiency is often negatively correlated to the upper bound on the number of revoked users. One of the most important applications of broadcast encryption is Pay-TV and it can typically be in the form of a revoke system: the service broadcasts to all users except revoked users who were detected as traitors or who unsubscribed from the system. The state-of-the-art revoke systems [4, 12, 23, 24] have compact ciphertext sizes that grow as $O(r)$ for r the bound of revoked users and which is not dependent in the number of users. None of these schemes is anonymous. An attempt was made to consider outsider adversaries, who can only corrupt users outside of the two targeted sets. In this limited setting, Fazio and Perera [17] showed that one can get key and ciphertext sizes that are sublinear in the number of users. We observe totally different situations for getting anonymity in broadcast encryption and in revoke systems: in broadcast encryption, optimal solutions exist [6, 9] but one cannot get the anonymity with sublinear ciphertext size in the total number of users; in revoke systems, no impossibility result has been settled and it does not exclude the possibility to get an anonymous schemes which is as efficient as non-anonymous ones, namely ciphertext size is $O(r)$, independent in the number of users. In this paper, we show that we can design anonymous schemes with $O(r)$ ciphertext size. Moreover, we also handle traceability to achieve anonymous trace-and-revoke systems.

1.1 Contributions

Our primary contribution is to develop the first symmetric-key trace-and-revoke scheme with traceability and anonymous revocation. We give two constructions of trace-and-revoke schemes, namely TR_0 and TR_1 from so-called linear functional encryptions. The former TR_0 is generically constructed from inner product functional encryption (IPFE) and encrypts single bit messages. Similarly, TR_1 is constructed from matrix multiplication functional encryption (MMFE) to support n-bit messages. Interestingly, unlike [4], our DDH instantiations do not require discrete-log evaluation for ciphertext decryption.

Our second contribution is to propose efficient constructions. We give an efficient construction of MMFE in the prime-order groups and prove that our MMFE construction is indeed tightly secure under the standard matDH assumption. This construction can be seen as tweaked Tomida's tightly secure IPFE for the symmetric-key settings [28]. However, we note that our security argument is somewhat different from Tomida's. On top of that, our tightly secure MMFE is more efficient than applying [28] naively.

Our third contribution is a cryptanalysis on the LWE-based IPFE construction of [29]. This justifies our choice of LWE-based IPFE to instantiate TR_0.

Anonymous Revocation. Before describing our results, we discuss the notion of anonymous revocation in trace-and-revoke schemes. The Enc algorithm of any trace-and-revoke scheme takes a message m and a revoked user set description $\mathcal{R}$ and computes a ciphertext that can only be decrypted by users outside $\mathcal{R}$. The anonymity property intuitively means that no information on $\mathcal{R}$ should be inferred from the ciphertext. A typical multi-challenge security model is defined by polynomially many challenge phases where the adversary adaptively produces $(m^{(t)}, \mathcal{R}_0^{(t)}, \mathcal{R}_1^{(t)})$ on the t-th phase and gets an encryption of $(m^{(t)}, \mathcal{R}_\beta^{(t)})$ for the same $\beta \leftarrow \{0,1\}$ for all the phases. However, this security model is quite strong and there are practical scenarios that do not require such stronger definition. For example, a typical trace-and-revoke scheme revokes more and more users over time. If a revoked user wants to get access to the system again, it has to contact the broadcaster, which can give the user a new key. In such a scenario, the revoked user set increases with time, such that $\mathcal{R}^{(t-1)} \subseteq \mathcal{R}^{(t)}$ for any timestamp $t > 1$. We model this scenario by introducing the restriction that, for any t, if the adversary produces the challenge $(m^{(t)}, \mathcal{R}_0^{(t)}, \mathcal{R}_1^{(t)})$, then $\mathcal{R}_0^{(t-1)} \subseteq \mathcal{R}_0^{(t)}$ and $\mathcal{R}_1^{(t-1)} \subseteq \mathcal{R}_1^{(t)}$, and call the resulting security property *multi-challenge monotonic anonymity* ($\mathsf{mIND\text{-}ID\text{-}CPA}$). Although this setting may suffice in many cases, this multi-challenge security model puts an additional restriction on the adversary that the challenge revocation sets must be related in a particular manner. This raises the following question: if we restrict ourselves to the single-challenge security model, can we get rid of such restriction on the monotonicity of challenge queries? Looking ahead, we formalize this security model in the appendix of the paper and show that our construction is secure in this model too.

1.2 Technical Overview

We start with a basic description of the trace-and-revoke scheme by Agrawal *et al* [4] (in the bounded collusion model). Each user id in this scheme is associated with a vector $\mathbf{x}_{\mathsf{id}}$ and, correspondingly, a set $\mathcal{R}$ is associated with $\mathbf{X}_{\mathcal{R}}$, the vector space spanned by $(\mathbf{x}_{\mathsf{id}})_{\mathsf{id}\in\mathcal{R}}$. Then, the predicate '$\mathsf{id} \notin \mathcal{R}$' can be emulated by testing if '$\langle \mathbf{x}_{\mathsf{id}}, \mathbf{v}_{\mathcal{R}} \rangle = 0$' for $\mathbf{v}_{\mathcal{R}}$ orthogonal to $\mathbf{X}_{\mathcal{R}}$. Using this relation, one encrypts a message m by encrypting $m \cdot \mathbf{v}_{\mathcal{R}}$ using an IPFE. An IPFE key for $\mathbf{x}_{\mathsf{id}}$ is used to evaluate $\mathsf{id} \notin \mathcal{R}$ in the encrypted domain. We now describe the decryption algorithm of [4] to clarify that this construction does not achieve anonymity of the revocation set. Decryption takes a ciphertext ct for $(m, \mathcal{R})$ and a secret key sk for id and runs IPFE decryption to obtain an intermediate $\mathsf{Res} = \langle \mathbf{x}_{\mathsf{id}}, m \cdot \mathbf{v}_{\mathcal{R}} \rangle$. The correctness then follows from the fact that decryption can compute $\langle \mathbf{x}_{\mathsf{id}}, \mathbf{v}_{\mathcal{R}} \rangle$ and divide Res by it to retrieve m. This is the reason why the description of $\mathcal{R}$ is provided as part of the ciphertext. Thus, the Agrawal *et al* scheme does not achieve revocation set hiding.

Our constructions build on [4], but avoid the above difficulty by exploiting the fact that if we consider the message to be single bit (i.e., $m \in \{0, 1\}$), we have the following four cases:

- $m = 0$, $\mathsf{id} \in \mathcal{R}$: The value of $\langle \mathbf{x}_{\mathsf{id}}, \mathbf{y}_{\mathcal{R}} \rangle = m \cdot \langle \mathbf{x}_{\mathsf{id}}, \mathbf{v}_{\mathcal{R}} \rangle$ is zero.
- $m = 1$, $\mathsf{id} \in \mathcal{R}$: Same as above where the value of $\langle \mathbf{x}_{\mathsf{id}}, \mathbf{y}_{\mathcal{R}} \rangle = m \cdot \langle \mathbf{x}_{\mathsf{id}}, \mathbf{v}_{\mathcal{R}} \rangle$ is zero; therefore, when $\mathsf{id} \in \mathcal{R}$, the message m is hidden.
- $m = 0$, $\mathsf{id} \notin \mathcal{R}$: The value of $\langle \mathbf{x}_{\mathsf{id}}, \mathbf{y}_{\mathcal{R}} \rangle = m \cdot \langle \mathbf{x}_{\mathsf{id}}, \mathbf{v}_{\mathcal{R}} \rangle$ is again zero.
- $m = 1$, $\mathsf{id} \notin \mathcal{R}$: The value of $\langle \mathbf{x}_{\mathsf{id}}, \mathbf{y}_{\mathcal{R}} \rangle = m \cdot \langle \mathbf{x}_{\mathsf{id}}, \mathbf{v}_{\mathcal{R}} \rangle$ is non-zero.

The above list of cases shows that a secret key for $\mathbf{x}_{\mathsf{id}}$ decrypts an IPFE ciphertext for $m \cdot \mathbf{v}_{\mathcal{R}}$ and retrieves $m \in \{0, 1\}$ correctly if $\mathsf{id} \notin \mathcal{R}$. Note that the decryption algorithm no longer requires the description of the revoked set $\mathcal{R}$. Based on this observation, our constructions translate $(m, \mathcal{R})$ into a vector $m \cdot \mathbf{v}_{\mathcal{R}}$ where $\mathbf{v}_{\mathcal{R}}$ is a random vector orthogonal to $\mathbf{X}_{\mathcal{R}}$ and id to a non-zero vector $\mathbf{x}_{\mathsf{id}}$. The monotonic anonymity (in the $\mathsf{mIND\text{-}ID\text{-}CPA}$ security model discussed above) then follows from the fact that the underlying IPFE hides the plaintext vector (here $m \cdot \mathbf{v}_{\mathcal{R}}$). For an n-bit message space, we can run independent and parallel executions of the IPFE that allow bit-by-bit retrieval of the message encrypted.[5] We propose a more efficient alternative, namely, matrix multiplication functional encryption (MMFE). Our generic transformation above ensures that any efficient instantiation of MMFE will result in efficient trace-and-revoke scheme. We discuss constructions of MMFE in both the group-based settings and in the lattice-based settings. We further show that our group-based construction of MMFE is tightly secure under standard assumptions. For lattice-based setting, we suggest to use [4] as we could mount a concrete attack on the state-of-the-art [29], rendering it insecure. Lastly, we note that tracing is performed in a similar fashion to [4].

An attack on the Wang et al *IPFE.* Here, we show that the IPFE construction by Wang *et al* can be broken for the parameters chosen in [29]. Our attack can be thwarted by increasing the parameters, but then the scheme does not enjoy great efficiency compared to the one from [4]. Here, we give the overview LWE-based IPFE from [29]. The dimension n of the LWE secrets is proportional to the security parameter λ, the parameters ℓ, m, p, q are polynomial in n. The master secret key is $\mathbf{Z}$, uniform over $\{0, \ldots, p-1\}^{\ell \times m}$. The public key is of the form $\mathsf{pk} = (\mathbf{A} \in \mathbb{Z}_q^{m \times n}, \mathbf{T} = \mathbf{Z}\mathbf{A} \in \mathbb{Z}_q^{\ell \times n})$. The secret key for the vector $\mathbf{x} \in \mathbb{Z}_p^{\ell}$ is $\mathsf{sk}_{\mathbf{x}} = \mathbf{x}^t \cdot \mathbf{Z}$. The ciphertext for a vector $\mathbf{y} \in \mathbb{Z}_p^{\ell}$ is of the form $(\mathbf{c}_0 \approx \mathbf{A}\mathbf{s}, \mathbf{c}_1 \approx \mathbf{T}\mathbf{s} + (q/p) \cdot \mathbf{y})$. The authors state that under the LWE assumption, this IPFE is adaptively secure for chosen message distributions, assuming that the secret key queries are linearly independent. We will give an algorithm that can recover the master key from the public key and ciphertexts (i.e., recover $\mathbf{z}$ from $\mathbf{X}^t$ and $\mathbf{X}^t\mathbf{z}$, where $\mathbf{z} \leftarrow \{0, \ldots, p-1\}^{\ell}$ and $\mathbf{X} \in \{0, \ldots, p-1\}^{\ell \times (\ell-1)}$ is chosen by the adversary). We remark that $\mathbf{z}$ belongs to a coset of the lattice orthogonal of $\mathbf{X}$ defined by $\mathbf{t}$. The crux of the attack is that for parameters as above, the minimum of this lattice is larger than $\|\mathbf{z}\|$. This means that we have a Bounded Distance Decoding problem instance in a lattice of dimension 1. Finally, we also explain why our attack does not extend to the schemes from [4, 5].

[5] In practice, we use this scheme to send 128-bit session keys or a stream: if an user is in the targeted set then it decrypts correctly and if the user is not in the targeted set then it gets all 0s (and therefore the equivalent of a trivial decryptor which generates 0 all the time).

Organization of the paper. In Section 2, we present some important definitions. In Section 3, we present black-box transformations to convert linear functional encryptions into trace-and-revoke systems with traceability and anonymity of revocation. Before we present group-based MMFE construction, in Section 4, we show an attack of a recent LWE-based IPFE construction [29]. Then, in Section 5, we present a construction of MMFE in the prime-order groups. We then give the definition of single-challenge anonymous security and give a proof in Appendix A.

2 Definitions and Preliminaries

For $a, b \in \mathbb{N}$ such that $a \leq b$, we often use $[a, b]$ to denote $\{a, \ldots, b\}$. Given a set of vectors S, we use $\mathsf{Matrix}(S)$ to denote the matrix whose each row is a distinct vector from S. For any two sets S and R, we define $S \Delta R = (S \setminus R) \cup (R \setminus S)$. For a dictionary $\mathsf{D} = (k, v_k)_k$, $\mathsf{D.vals}()$ gives the set $\{v_k : k \in \mathsf{D}\}$. For a vector space $\mathbf{V}$ over a field $\mathbb{K}$, the corresponding orthogonal space is denoted by $\mathbf{V}^{\perp}$. For a distribution D, we write $x \leftarrow D$ to say that x is sampled from D. The ppt abbreviation stands for probabilistic polynomial time. We denote $\mathcal{G}_{gen}(1^{\lambda}, p) \rightarrow (g, \mathbb{G})$ such that $\mathbb{G}$ is a cyclic group of prime order p and g generates $\mathbb{G}$. For $\mathbf{A} = (a_{ij}) \in \mathbb{Z}_p^{\beta \times \alpha}$ we denote $[\mathbf{A}] = (g^{a_{ij}}) \in \mathbb{Z}_p^{\beta \times \alpha}$. For $m, k \in \mathbb{N}$ for $m > k$, we use $\mathbf{M} \leftarrow \mathcal{D}_{m,k}$ to get a full rank matrix $\mathbf{M} \in \mathbb{Z}_p^{m \times k}$ where the first k rows are linearly independent.

2.1 Linear Functional Encryption

A functional encryption scheme [11] allows a user, having a secret key sk_f corresponding to a function f, to evaluate $f(z)$ securely given a ciphertext ct_z for a plaintext z. The inner product function, being one of the simplest functionalities, has received a tremendous amount of exposure [1–3, 5, 13, 28]. We here define a variant of inner product functional encryption (IPFE) in the symmetric-key setting and introduce matrix multiplication functional encryption (MMFE) as a generalization of IPFE to construct a trace-and-revoke scheme with anonymous revocation with larger message space.

2.2 Inner Product Functional Encryption.

We consider inner product functional encryption (IPFE) over $\mathbb{Z}_p$ in the symmetric-key settings [6] for a prime integer $p \geq 2$. Unlike existing IPFE definitions in [1,2,4,5], the $\mathcal{IPFE}.\mathsf{Dec}$ algorithm here retrieves an injective function of the inner product value. In particular, it may not be the inner product value itself. More precisely, the $\mathcal{IPFE}.\mathsf{Dec}$ algorithm takes as input a ciphertext ct that encrypts $\mathbf{y} \in \mathbb{Z}_p^{\ell}$ and a secret key $\mathsf{sk}_{\mathbf{x}}$ with respect to $\mathbf{x} \in \mathbb{Z}_p^{\ell}$, and outputs $f(\langle \mathbf{x}, \mathbf{y} \rangle)$.

Definition 1. *An inner product functional encryption (IPFE) over $\mathbb{Z}_p$ with respect to an injective map f is a tuple $\mathcal{IPFE} = (\mathcal{IPFE}.\mathsf{Setup}, \mathcal{IPFE}.\mathsf{KeyGen}, \mathcal{IPFE}.\mathsf{Enc}, \mathcal{IPFE}.\mathsf{Dec})$ of four ppt algorithms.*

- $\mathcal{IPFE}.\mathsf{Setup}(1^{\lambda}, 1^{\ell}, p)$ *takes as input the security parameter λ and the dimension of vectors ℓ. It outputs the public parameters pp and the master secret key msk. The public parameters pp contain the description of the injective function f.*
- $\mathcal{IPFE}.\mathsf{KeyGen}(\mathsf{pp}, \mathsf{msk}, \mathbf{x})$ *takes as input the public parameters pp, the master secret key msk and a vector $\mathbf{x} \in \mathbb{Z}_p^{\ell}$ and outputs a secret key $\mathsf{sk}_{\mathbf{x}}$.*
- $\mathcal{IPFE}.\mathsf{Enc}(\mathsf{pp}, \mathsf{msk}, \mathbf{y})$ *takes as input the public parameters pp, the master secret key msk and a vector $\mathbf{y} \in \mathbb{Z}_p^{\ell}$ and outputs a ciphertext ct.*
- $\mathcal{IPFE}.\mathsf{Dec}(\mathsf{pp}, \mathsf{sk}_{\mathbf{x}}, \mathsf{ct})$ *takes as input the public parameters pp, the secret key of a user $\mathsf{sk}_{\mathbf{x}}$ and a ciphertext $\mathsf{ct}_{\mathbf{y}}$, and outputs $f(\langle \mathbf{x}, \mathbf{y} \rangle)$.*

The correctness requirement is that, with overwhelming probability over the randomness used by the algorithms, for $(\mathsf{pp}, \mathsf{msk}) \leftarrow \mathcal{IPFE}.\mathsf{Setup}(1^{\lambda}, 1^{\ell}, p)$, for all $\mathbf{x}, \mathbf{y} \in \mathbb{Z}_p^{\ell}$, for $\mathsf{sk}_{\mathbf{x}} \leftarrow \mathcal{IPFE}.\mathsf{KeyGen}(\mathsf{pp}, \mathsf{msk}, \mathbf{x})$ and $\mathsf{ct} \leftarrow \mathcal{IPFE}.\mathsf{Enc}(\mathsf{pp}, \mathsf{msk}, \mathbf{y})$:

$$\mathcal{IPFE}.\mathsf{Dec}\,(\mathsf{pp}, \mathsf{sk}_{\mathbf{x}}, \mathsf{ct}) = f(\langle \mathbf{x}, \mathbf{y} \rangle).$$

[6] We define IPFE in the symmetric-key settings as a stepping stone to construct trace-and-revoke in the symmetric-key settings.

Security. The security (IND-CPA) of symmetric-key IPFE is modeled as the following security game played between a challenger and an adversary $\mathcal{A}$. This security model is reminiscent of that of [27].

- The challenger runs $(\mathsf{pp}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda, 1^\ell, p)$, keeps msk secret and gives the public parameters pp to the adversary $\mathcal{A}$. The challenger further samples $\beta \leftarrow \{0, 1\}$.
- Adversary $\mathcal{A}$ adaptively issues queries of one of the following two types:
  1. **Ciphertext query**: The adversary sends two vectors $\mathbf{y}^{(0)}, \mathbf{y}^{(1)} \in \mathbb{Z}_p^\ell$, and the challenger responds with $\mathsf{ct}^{(\beta)} \leftarrow \mathsf{Enc}(\mathsf{pp}, \mathsf{msk}, \mathbf{y}^{(\beta)})$.
  2. **Secret key query**: The adversary sends a vector $\mathbf{x} \in \mathbb{Z}_p^\ell$, and the challenger responds with $\mathsf{sk}_\mathbf{x} \leftarrow \mathsf{KeyGen}(\mathsf{pp}, \mathsf{msk}, \mathbf{x})$.

  These queries can be made under the restriction that for all ciphertext query $(\mathbf{y}^{(0)}, \mathbf{y}^{(1)})$ and all secret key query $\mathbf{x}$, we must have $f(\langle \mathbf{x}, \mathbf{y}^{(0)} \rangle) = f(\langle \mathbf{x}, \mathbf{y}^{(1)} \rangle)$.
- Finally, the adversary returns its guess $\beta' \in \{0, 1\}$ for the bit β chosen by the challenger. The adversary wins this game if $\beta = \beta'$.

The advantage of the adversary is defined as $\mathrm{Adv}_{\mathsf{FE},\mathcal{A}}^{\mathsf{IND\text{-}CPA}} = |\Pr[\beta = \beta'] - 1/2|$. A symmetric-key IPFE scheme $\mathcal{IPFE}$ is said secure if $\mathrm{Adv}_{\mathsf{FE},\mathcal{A}}^{\mathsf{IND\text{-}CPA}}$ is negligible for all ppt adversary $\mathcal{A}$.

2.3 Matrix Multiplication Functional Encryption.

We now define matrix multiplication functional encryption (MMFE) over $\mathbb{Z}_p$ for a prime integer $p \geq 2$. As the name suggests, MMFE decrypts a ciphertext for a matrix $\mathbf{M} \in \mathbb{Z}_p^{n \times \ell}$ with a key $\mathsf{sk}_\mathbf{x}$ made of $\mathbf{x} \in \mathbb{Z}_p^\ell$ revealing only $\mathbf{Mx}$ and nothing else. Due to its similarity with the definition of IPFE, MMFE can be achieved from available IPFE. In particular, one can use n-many instances of IPFE to encrypt n vectors $(\mathbf{y}_1, \dots, \mathbf{y}_n)$ independently and the Dec algorithm basically computes $\langle \mathbf{y}_i, \mathbf{x} \rangle$ for each $i \in [1, n]$ individually. However, such a trivial construction suffers from a degradation proportional to n. This gets worse in case of multi-challenge security which in fact we consider in this work.

We give a definition and propose a concrete construction with tight security in this paper. A related primitive was already introduced for predicate encryption to allow decryption based on subspace membership relation (decrypt if $\mathbf{Mx} = \mathbf{0}$) in [10]. Looking ahead, we present a symmetric-key MMFE definition here to construct symmetric-key trace-and-revoke scheme TR_1 for arbitrary n-bit messages in Section 3.2.

Definition 2. *A matrix multiplication functional encryption scheme $\mathcal{MMFE}$ over $\mathbb{Z}_p$ with respect to an injective function f is a tuple $\mathcal{MMFE} = (\mathcal{MMFE}.\mathsf{Setup}, \mathcal{MMFE}.\mathsf{KeyGen}, \mathcal{MMFE}.\mathsf{Enc}, \mathcal{MMFE}.\mathsf{Dec})$ of four ppt algorithms with the following specifications:*

- $\mathcal{MMFE}.\mathsf{Setup}(1^\lambda, 1^\ell, 1^n, p)$ *takes as input the security parameter λ and the dimensions (n, ℓ) of matrices. It outputs the public parameters* pp *and the master secret key* msk*. Similarly to $\mathcal{IPFE}$, the public parameters* pp *contain the description of an injective function f.*
- $\mathcal{MMFE}.\mathsf{KeyGen}(\mathsf{pp}, \mathsf{msk}, \mathbf{x})$ *takes as input the public parameters* pp*, the master secret key* msk *and a vector $\mathbf{x} \in \mathbb{Z}_p^\ell$ and outputs a secret key* $\mathsf{sk}_\mathbf{x}$.
- $\mathcal{MMFE}.\mathsf{Enc}(\mathsf{pp}, \mathsf{msk}, \mathbf{M})$ *takes as input the public parameters* pp*, the master secret key* msk *and a matrix $\mathbf{M} \in \mathbb{Z}_p^{n \times \ell}$ and outputs a ciphertext* ct.
- $\mathcal{MMFE}.\mathsf{Dec}(\mathsf{pp}, \mathsf{sk}_\mathbf{x}, \mathsf{ct})$ *takes as input the public parameters* pp*, the secret key of a user* $\mathsf{sk}_\mathbf{x}$ *and a ciphertext* ct*, and outputs $(f(\mathbf{M}_1\mathbf{x}), \dots, f(\mathbf{M}_n\mathbf{x}))$ where $\mathbf{M}_i$ is the i^{th} row of $\mathbf{M}$.*

The correctness requirement is that, with overwhelming probability over the randomness used by the algorithms, for $(\mathsf{pp}, \mathsf{msk}) \leftarrow \mathcal{MMFE}.\mathsf{Setup}(1^\lambda, 1^\ell, 1^n, p)$*, for all $\mathbf{x} \in \mathbb{Z}_p^\ell$ and $\mathbf{M} \in \mathbb{Z}_p^{n \times \ell}$, for* $\mathsf{sk}_\mathbf{x} \leftarrow \mathcal{MMFE}.\mathsf{KeyGen}(\mathsf{pp}, \mathsf{msk}, \mathbf{x})$ *and* $\mathsf{ct} \leftarrow \mathcal{MMFE}.\mathsf{Enc}(\mathsf{pp}, \mathsf{msk}, \mathbf{M})$*:*

$$\mathcal{MMFE}.\mathsf{Dec}\left(\mathsf{pp}, \mathsf{sk}_\mathbf{x}, \mathsf{ct}\right) = \left(f(\mathbf{M}_1\mathbf{x}), \dots, f(\mathbf{M}_n\mathbf{x})\right).$$

Security. Full security (IND-CPA) of symmetric-key matrix multiplication functional encryption is modeled as the following security game played between a challenger and an adversary $\mathcal{A}$.

- The challenger runs $(\mathsf{pp}, \mathsf{msk}) \leftarrow \mathsf{Setup}(1^\lambda, 1^\ell, 1^n, p)$, keeps msk secret and gives the public parameters pp to the adversary $\mathcal{A}$. The challenger further samples $\beta \leftarrow \{0,1\}$.
- Adversary $\mathcal{A}$ adaptively issues queries of one of the following two types:
  1. **Ciphertext query**: The adversary sends two matrices $\mathbf{M}^{(0)}, \mathbf{M}^{(1)} \in \mathbb{Z}_p^{n\times\ell}$, and the challenger responds with $\mathsf{ct}^{(\beta)} \leftarrow \mathsf{Enc}(\mathsf{pp}, \mathsf{msk}, \mathbf{M}^{(\beta)})$.
  2. **Secret key query**: The adversary sends a vector $\mathbf{x} \in \mathbb{Z}_p^\ell$, and the challenger responds with $\mathsf{sk}_\mathbf{x} \leftarrow \mathsf{KeyGen}(\mathsf{pp}, \mathsf{msk}, \mathbf{x})$.

  These queries can be made under the restriction that for all ciphertext query $(\mathbf{M}^{(0)}, \mathbf{M}^{(1)})$ and all secret key query $\mathbf{x}$, we must have $f(\mathbf{M}^{(0)}\mathbf{x}) = f(\mathbf{M}^{(1)}\mathbf{x})$.
- Finally, the adversary returns its guess $\beta' \in \{0,1\}$ for the bit β chosen by the challenger. The adversary wins this game if $\beta = \beta'$.

The advantage of the adversary is defined as $\mathrm{Adv}^{\mathsf{IND\text{-}CPA}}_{\mathcal{MMFE},\mathcal{A}} = |\Pr[\beta = \beta'] - 1/2|$. A symmetric-key inner matrix multiplication functional encryption scheme $\mathcal{MMFE}$ is said secure if $\mathrm{Adv}^{\mathsf{IND\text{-}CPA}}_{\mathcal{MMFE},\mathcal{A}}$ is negligible for all ppt adversary $\mathcal{A}$.

2.4 Trace-and-Revoke Systems

A symmetric key traitor tracing encryption scheme is a multi-recipient encryption system in which a broadcasting office has the master secret key for encryption and there are many users with decryption capabilities, each having its own secret key. Additionally, the encryption scheme provides a feature to let the broadcaster identify at least one user from a coalition $\mathcal{T}$ of malicious users (traitors) that built an unauthorized decryption device $\mathcal{D}$. The following is the blackbox confirmation model [8], in which an efficient tracing algorithm Trace is given oracle access to $\mathcal{D}$, which we denote by $\mathcal{O}^\mathcal{D}$. The oracle $\mathcal{O}^\mathcal{D}$ takes as input any message-ciphertext pair (m, C) and returns 1 if $\mathcal{D}(C) = m$ and 0 otherwise. Given as input a set $\mathcal{S}$ of suspected users containing $\mathcal{T}$, the Trace algorithm should disclose the identity of at least one user from the set $\mathcal{T}$. For security, a traitor coalition should not be able to design a useful box that escapes tracing, i.e., such that the Trace algorithm replies $\perp$ or frames an innocent user in $\mathcal{S} \setminus \mathcal{T}$.

Following [4], the probability of decryption of decoder $\mathcal{D}$, can be estimated by repeatedly querying the oracle $\mathcal{O}^\mathcal{D}$ with plaintext-ciphertext pairs. Therefore, we assume the decryption device $\mathcal{D}$ correctly decrypts a properly generated ciphertext with significant probability. The following is a description of $\mathcal{D}$, reproduced from [4] and modified for the symmetric-key setting. Let $\mathcal{R}$ be any set of revoked users, of size $\leq r$. Let the message m be sampled uniformly at random from the message space $\mathcal{M}$ and let $C_\mathcal{R}$ be the output of the encryption algorithm Enc using the master secret key msk and $\mathcal{R}$ as the set of revoked users. With $C_\mathcal{R}$ as input, the device $\mathcal{D}$ is assumed to output m with probability significantly more than $1/|\mathcal{M}|$:

$$\Pr_{\substack{m \leftarrow U(\mathcal{M}) \\ C_\mathcal{R} \leftarrow \mathtt{Enc}(\mathsf{msk}, \mathsf{pp}, \mathcal{R}, m)}} \left[\mathcal{O}^\mathcal{D}(C_\mathcal{R}, m) = 1\right] \geq \frac{1}{|\mathcal{M}|} + \frac{1}{\lambda^c}, \tag{1}$$

for some constant $c > 0$.

We let the identity space ID and the message space $\mathcal{M}$ be implicit arguments to the setup algorithm below. We let the secret key space $\mathcal{K}$, the ciphertext space $\mathcal{C}$ (along with ID and $\mathcal{M}$) and the descriptions of mathematical tools that are used be part of the public parameters output by the setup algorithm. We adapt the definition from [4] to the symmetric-key setting.

Definition 3. *A dynamic trace-and-revoke scheme* TR *in the black-box confirmation model is a tuple* $\mathsf{TR} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Trace})$ *of five* ppt *algorithms with the following specifications.*

- $\mathsf{Setup}(1^\lambda, 1^r, 1^t)$ *takes as input the security parameter* λ*, the bound* t *on the size of traitor coalitions and the bound* r *on the number of revoked users. It outputs* $(\mathsf{msk}, \mathsf{pp}, \mathsf{dir})$ *containing the master secret key* msk*, the public parameters* pp *and the initially empty user directory* dir*. Here, unlike [4],* dir *is kept secret.*
- $\mathsf{KeyGen}(\mathsf{pp}, \mathsf{msk}, \mathsf{dir}, \mathsf{id})$ *takes as input the public parameters* pp*, the master secret* msk*, the user directory* dir *and an identity* $\mathsf{id} \in \mathsf{ID}$ *of a user. It outputs the corresponding secret key* $\mathsf{sk}_{\mathsf{id}}$ *and some information* u_{id} *for the given identity* id*. It also updates* dir *to include* u_{id}*.*
- $\mathsf{Enc}(\mathsf{pp}, \mathsf{msk}, \mathsf{dir}, \mathcal{R}, m)$ *takes as input the public parameters* pp*, the master secret* msk*, the user directory* dir*, a set* $\mathcal{R}$ *of size* $\leq r$ *which contains the* u_{id} *of each revoked user in* dir*, and a plaintext message* $m \in \mathcal{M}$*. It outputs a ciphertext* $C_{\mathcal{R}} \in \mathcal{C}$*.*
- $\mathsf{Dec}(\mathsf{pp}, \mathsf{sk}_{\mathsf{id}}, C_{\mathcal{R}})$ *takes as input the public parameters* pp*, a secret key* $\mathsf{sk}_{\mathsf{id}}$ *of a user with identity* id *and a ciphertext* $C_{\mathcal{R}} \in \mathcal{C}$*. It outputs a plaintext* $m' \in \mathcal{M}$*.*
- $\mathsf{Trace}(\mathsf{pp}, \mathsf{msk}, \mathsf{dir}, \mathcal{R}, \mathcal{S}, \mathcal{O}^{\mathcal{D}})$ *is a tracing algorithm in the black-box confirmation model that takes as input the public parameters* pp*, the master secret key* msk*, the user directory* dir*, a set* $\mathcal{R}$ *of* $\leq r$ *revoked users, a set* $\mathcal{S}$ *of* $\leq t$ *suspect users, and has black-box access to the pirate decoder* $\mathcal{D}$ *through the oracle* $\mathcal{O}^{\mathcal{D}}$*. It outputs an identity* id *or* $\perp$*.*

The correctness requirement is that, with overwhelming probability over the randomness used by the algorithms, for $(\mathsf{pp}, \mathsf{msk}, \mathsf{dir}) \leftarrow \mathsf{Setup}(1^\lambda, 1^r, 1^t)$*, for any set* $\mathcal{R}$ *of* $\leq r$ *revoked users:*

$$\forall m \in \mathcal{M},\ \forall \mathsf{id} \in \mathsf{ID} \setminus \mathcal{R}:\ \mathsf{Dec}(\mathsf{pp}, \mathsf{sk}_{\mathsf{id}}, \mathsf{Enc}(\mathsf{pp}, \mathsf{msk}, \mathsf{dir}, \mathcal{R}, m)) = m.$$

In this work, we consider three security properties for a trace-and-revoke scheme: message hiding, revocation set hiding, and traceability.

2.4.1 Message Hiding. The $\mathsf{IND\text{-}CPA}$ security of a trace-and-revoke scheme TR is defined based on the following game. Informally speaking, neither a system outsider nor a revoked user must be able to get any information about the encrypted message.

- The challenger runs $\mathsf{Setup}(1^\lambda, 1^r, 1^t)$ and gives the produced public parameters pp to the adversary $\mathcal{A}$. The adversary may ask the challenger to add polynomially many users in the system (these user addition queries can be adaptive and take place at any time in the game). The challenger updates dir accordingly.
- The adversary can adaptively make up to r secret key queries and a single challenge ciphertext query, of the following form:
  - ∗ Given a key generation query id, the challenger provides the corresponding $\mathsf{sk}_{\mathsf{id}}$ to $\mathcal{A}$.
  - ∗ Given the challenge ciphertext query $(m_0, m_1, \mathcal{R})$ with $\mathcal{R} \subset \mathsf{ID}$ of size $\leq r$, the challenger samples $\beta \leftarrow \{0,1\}$ and provides $C^{(\beta)} \leftarrow \mathsf{Enc}(\mathsf{pp}, \mathsf{msk}, \mathsf{dir}, \mathcal{R}, m_\beta)$ to $\mathcal{A}$.

  These queries are subject to the restriction that every queried id belongs to $\mathcal{R}$.
- Finally, the adversary returns its guess $\beta' \in \{0,1\}$ for the bit β chosen by the challenger. The adversary wins this game if $\beta = \beta'$.

The advantage of the adversary $\mathcal{A}$ is defined as

$$\mathrm{Adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{TR},\mathcal{A}} = |\Pr[\beta = \beta'] - 1/2|.$$

A trace-and-revoke scheme TR is said to be $\mathsf{IND\text{-}CPA}$ secure if $\mathrm{Adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{TR},\mathcal{A}}$ is negligible for all ppt adversary $\mathcal{A}$.

2.4.2 Revocation Set Hiding. The anonymity of a trace-and-revoke scheme TR captures the idea of hiding the *revocation set* in the ciphertext: if t^{th} challenge ciphertext is created for one of the two adversarially chosen revoked sets $(\mathcal{R}_0^{(t)}, \mathcal{R}_1^{(t)})$ on the t^{th} challenge phase, then the adversary cannot distinguish if $\mathcal{R}_0^{(t)}$ or $\mathcal{R}_1^{(t)}$ was used for the encryption for all of t.

As we already have mentioned in the Introduction, we aim for a multi-challenge security settings that properly emulates the following scenario: A typical trace-and-revoke scheme traces and revokes more and more users over the time. In such a scenario, each new ciphertext is created for growing revoked user sets. We call this setting as *monotonic anonymity* security model ($\mathsf{mIND\text{-}ID\text{-}CPA}$) and define it as following.

- The challenger runs $\mathsf{Setup}(1^\lambda, 1^r, 1^t)$ and gives the produced public parameter pp to the adversary $\mathcal{A}$. The adversary may ask the challenger to add polynomially many users in the system (these user addition queries can be adaptive and take place at any time in the game). The challenger updates dir accordingly.
- The adversary can adaptively make up to $(r+t)$ secret key queries and polynomially many anonymity challenge queries, of the following form:
  - $*$ Given a key generation query id, the challenger provides the corresponding $\mathsf{sk}_{\mathsf{id}}$ to $\mathcal{A}$.
  - $*$ Given a challenge anonymity query $(m, \mathcal{R}_0, \mathcal{R}_1)$ with $\mathcal{R}_0, \mathcal{R}_1 \subset \mathsf{ID}$ of size $\leq r$, the challenger samples $\beta \leftarrow \{0,1\}$ and provides $C^{(\beta)} \leftarrow \mathsf{Enc}(\mathsf{pp}, \mathsf{msk}, \mathsf{dir}, \mathcal{R}_\beta, m)$ to $\mathcal{A}$.

  These queries are subject to the restriction that for every queried id, either $\mathsf{id} \in \mathcal{R}_0 \cap \mathcal{R}_1$ or $\mathsf{id} \in \mathsf{ID} \setminus (\mathcal{R}_0 \cup \mathcal{R}_1)$. Among all the key queries that have been made, at most t of them could be satisfying $\mathsf{id} \in \mathsf{ID} \setminus (\mathcal{R}_0 \cup \mathcal{R}_1)$ and at most r of them could be satisfying $\mathsf{id} \in \mathcal{R}_0 \cap \mathcal{R}_1$. The challenge anonymity queries also have a natural restriction that $\mathcal{R}_0^{(i)} \subseteq \mathcal{R}_0^{(j)}$ and $\mathcal{R}_1^{(i)} \subseteq \mathcal{R}_1^{(j)}$ for all $i \leq j$ where the t^{th} challenge anonymity query was made on $(m^{(t)}, \mathcal{R}_0^{(t)}, \mathcal{R}_1^{(t)})$.
- Finally, the adversary returns its guess $\beta' \in \{0,1\}$ for the bit β chosen by the challenger. The adversary wins this game if $\beta = \beta'$.

The advantage of the adversary $\mathcal{A}$ is defined as

$$\mathrm{Adv}_{\mathsf{TR},\mathcal{A}}^{\mathsf{mIND\text{-}ID\text{-}CPA}} = |\Pr[\beta = \beta'] - 1/2|.$$

A trace-and-revoke scheme TR is said to be $\mathsf{mIND\text{-}ID\text{-}CPA}$ secure if $\mathrm{Adv}_{\mathsf{TR},\mathcal{A}}^{\mathsf{mIND\text{-}ID\text{-}CPA}}$ is negligible for all ppt adversary $\mathcal{A}$.

2.4.3 Traceability. The notion of traceability considers a suspected set $\mathcal{S}$ of users who might have produced the pirate decoder $\mathcal{D}$. Then the tracing algorithm Trace outputs an $\mathsf{id} \in \mathcal{S} \setminus \mathcal{T}$ where $\mathcal{T}$ is the set of traitors who are already detected. This requirement is formalized using the following game, denoted by $\mathsf{AD\text{-}TT}$, between an adversary $\mathcal{A}$ and a challenger. We reproduce the security model from [4] for sake of completeness.[7]More precisely, the authors of [4] achieved *public-traceability*: for this purpose, the public-key Enc algorithm was used to construct so-called probe ciphertexts to query $\mathcal{O}^{\mathcal{D}}$ and identify a traitor. Our trace-and-revoke scheme relies on a symmetric key Enc algorithm, and hence tracing relies on the master secret key msk (in particular, tracing is not public).

- The challenger runs $\mathsf{Setup}(1^\lambda, 1^r, 1^t)$ and gives pp to $\mathcal{A}$. The adversary may ask the challenger to add polynomially many users in the system (these user addition queries can be adaptive and take place at any time in the game). The challenger updates dir accordingly.
- Adversary $\mathcal{A}$ makes adaptive traitor key queries on at most t distinct users. For every id queried, the challenger checks to find $\mathsf{u}_{\mathsf{id}} \leftarrow \mathsf{dir}[\mathsf{id}]$. If available, records id in $\mathcal{T}$ and returns $\mathsf{sk}_{\mathsf{id}}$. Otherwise, adds u_{id} to $\mathsf{dir}[\mathsf{id}]$, records id in $\mathcal{T}$ and returns $\mathsf{sk}_{\mathsf{id}} \leftarrow \mathsf{KeyGen}(\mathsf{pp}, \mathsf{msk}, \mathsf{id})$.
- Adversary $\mathcal{A}$ sends an adaptively chosen revocation set $\mathcal{R} \subset \mathsf{ID}$ of size $\leq r$ and gets back all the secret keys $\{\mathsf{sk}_{\mathsf{id}} \leftarrow \mathsf{KeyGen}(\mathsf{pp}, \mathsf{msk}, \mathsf{id})\}_{\mathsf{id} \in \mathcal{R}}$.
- Adversary $\mathcal{A}$ then produces a pirate decoder $\mathcal{D}$ and gives the challenger its access in terms of an oracle $\mathcal{O}^{\mathcal{D}}$. $\mathcal{A}$ also produces a suspect set $\mathcal{S}$ of size $\leq t$ containing $\mathcal{T}$ and sends it to the challenger.
- The challenger then runs $\mathsf{Trace}(\mathsf{pp}, \mathsf{msk}, \mathsf{dir}, \mathcal{R}, \mathcal{S}, \mathcal{O}^{\mathcal{D}})$. The adversary wins if both of the following hold:
  - $*$ Equation (1) is satisfied for the set of revoked users $\mathcal{R}$ chosen by the adversary (i.e., decoder $\mathcal{D}$ is useful),
  - $*$ the execution of Trace outputs $\perp$ or outputs an $\mathsf{id} \in \mathcal{S} \setminus \mathcal{T}$ with probability $\geq 1/\lambda^c$.

We define the tracing advantage $\mathrm{Adv}_{\mathsf{TR},\mathcal{A}}^{\mathsf{AD\text{-}TT}}$ as the probability of $\mathcal{A}$'s win. A trace-and-revoke scheme TR is said to be $\mathsf{AD\text{-}TT}$ secure if the advantage $\mathrm{Adv}_{\mathsf{TR},\mathcal{A}}^{\mathsf{AD\text{-}TT}}$ is negligible for all ppt adversary $\mathcal{A}$.

[7] Recently, a more general model of pirate, called *pirate distinguisher*, have been introduced and considered in [18,26]. However, as proven in [14], in the bit-encryption setting, such a notion of pirate distinguisher is equivalent to the pirate decoder. In this section, we consider bit-encryption and in the next section about multi-bit encryption, the tracing is reduced to the tracing in the bit-encryption sub schemes. Therefore, we keep using the definition from [4] (adapted to the symmetric-key setting).

2.5 Mathematical Tools and Hardness Assumptions

Prime-Order Groups. We assume $\mathcal{G}_{gen}$ to be the group generator that generates the prime order group description. Precisely, $\mathcal{G}_{gen}(1^\lambda, p) \to (g, \mathbb{G})$ such that $\mathbb{G}$ is a cyclic group of prime order p and g generates $\mathbb{G}$. We follow the notation of [16] to denote g^a by $[a]$ for any $a \in \mathbb{Z}_p$ and for $\mathbf{A} = (a_{ij}) \in \mathbb{Z}_p^{\beta \times \alpha}$ we denote

$$[\mathbf{A}] = \begin{pmatrix} g^{a_{11}} & \dots & g^{a_{1\alpha}} \\ \vdots & \ddots & \vdots \\ g^{a_{\beta 1}} & \dots & g^{a_{\beta\alpha}} \end{pmatrix} \in \mathbb{G}^{\beta \times \alpha}.$$

2.5.1 $\mathcal{D}_k$-matDH.

For all adversary $\mathcal{A}$, the advantage function is defined as following

$$\mathsf{Adv}_{\mathcal{A}}^{\mathcal{D}_k\text{-}\mathsf{matDH}}(\lambda) = |\Pr[\mathcal{A}([\mathbf{U}], [\mathbf{Ux}]) = 1] - \Pr[\mathcal{A}([\mathbf{U}], [\mathbf{z}]) = 1]|$$

where $\mathbf{U} \leftarrow \mathcal{D}_k$, $\mathbf{x} \leftarrow \mathbb{Z}_p^k$ and The $\mathcal{D}_k$-matDH assumption states that $\mathsf{Adv}_{\mathcal{A}}^{\mathcal{D}_k\text{-}\mathsf{matDH}}(\lambda)$ is negligible in λ for all ppt adversary $\mathcal{A}$.

2.5.2 n-$\mathcal{D}_k$-matDH.

For all adversary $\mathcal{A}$, the advantage function is defined as following

$$\mathsf{Adv}_{\mathcal{A}}^{n\text{-}\mathcal{D}_k\text{-}\mathsf{matDH}}(\lambda) = |\Pr[\mathcal{A}([\mathbf{U}], [\mathbf{UX}]) = 1] - \Pr[\mathcal{A}([\mathbf{U}], [\mathbf{Z}]) = 1]|$$

where $\mathbf{U} \leftarrow \mathcal{D}_k$, $\mathbf{X} \leftarrow \mathbb{Z}_p^{k \times n}$ and The n-fold $\mathcal{D}_k$-matDH assumption (i.e. n-$\mathcal{D}_k$-matDH) states that $\mathsf{Adv}_{\mathcal{A}}^{n\text{-}\mathcal{D}_k\text{-}\mathsf{matDH}}(\lambda)$ is negligible in Now, [16] showed that $\mathsf{Adv}^{n\text{-}\mathcal{D}_k\text{-}\mathsf{matDH}}(\lambda) \le \mathsf{Adv}^{\mathcal{D}_k\text{-}\mathsf{matDH}}(\lambda)$ for any fixed value n that is polynomial in λ.

2.5.3 $\mathcal{D}_{2k,k}$-matDH.

For all adversary $\mathcal{A}$, the advantage function is defined as following

$$\mathsf{Adv}_{\mathcal{A}}^{\mathcal{D}_{2k,k}\text{-}\mathsf{matDH}}(\lambda) = |\Pr[\mathcal{A}([\mathbf{V}], [\mathbf{Vy}]) = 1] - \Pr[\mathcal{A}([\mathbf{V}], [\mathbf{z}]) = 1]|$$

where $\mathbf{V} \leftarrow \mathcal{D}_{2k,k}$, $\mathbf{y} \leftarrow \mathbb{Z}_p^k$ and $\mathbf{z} \leftarrow \mathbb{Z}_p^{2k}$. The $\mathcal{D}_{2k,k}$-matDH assumption states that $\mathsf{Adv}_{\mathcal{A}}^{\mathcal{D}_{2k,k}\text{-}\mathsf{matDH}}(\lambda)$ is negligible in λ for all ppt adversary $\mathcal{A}$. [16] showed that given a $\mathcal{D}_k$-matDH problem instance, one can create a $\mathcal{D}_{2k,k}$-matDH problem instance with the degradation of k i.e.

$$\mathsf{Adv}^{\mathcal{D}_{2k,k}\text{-}\mathsf{matDH}}(\lambda) \le k \cdot \mathsf{Adv}^{\mathcal{D}_k\text{-}\mathsf{matDH}}(\lambda).$$

2.5.4 $\mathcal{D}_k$-matDH′.

For all adversary $\mathcal{A}$, the advantage function is defined as following

$$\mathsf{Adv}_{\mathcal{A}}^{\mathcal{D}_k\text{-}\mathsf{matDH}'}(\lambda) = |\Pr[\mathcal{A}([\mathbf{S}], [\mathbf{u}^\top \mathbf{S}]) = 1] - \Pr[\mathcal{A}([\mathbf{S}], [\mathbf{z}^\top]) = 1]|$$

where $\mathbf{S} \leftarrow \mathbb{Z}_p^{k \times m}$, $\mathbf{u} \leftarrow \mathbb{Z}_p^k$ and $\mathbf{z} \leftarrow \mathbb{Z}_p^m$ for a fixed value m that is polynomial in λ. The $\mathcal{D}_k$-matDH' assumption states that $\mathsf{Adv}_{\mathcal{A}}^{\mathcal{D}_k\text{-}\mathsf{matDH}'}(\lambda)$ is negligible in λ for all ppt adversary $\mathcal{A}$. [28] showed that given a $\mathcal{D}_k$-matDH' problem instance, one can create a m-fold $\mathcal{D}_k$-matDH problem instance without any degradation i.e. $\mathsf{Adv}^{\mathcal{D}_k\text{-}\mathsf{matDH}'}(\lambda) \le \mathsf{Adv}^{m\text{-}\mathcal{D}_k\text{-}\mathsf{matDH}}(\lambda)$. Due to relation between $\mathcal{D}_k$-matDH and m-$\mathcal{D}_k$-matDH mentioned above, $\mathsf{Adv}^{\mathcal{D}_k\text{-}\mathsf{matDH}'}(\lambda) \le \mathsf{Adv}^{\mathcal{D}_k\text{-}\mathsf{matDH}}(\lambda)$.

3 Trace-and-Revoke from Linear Functional Encryption

In this section, we construct a trace-and-revoke system from a linear functional encryption scheme that achieves traceability and anonymous revocation. This is achieved in two steps. First, a trace-and-revoke system for single-bit messages is constructed from inner product functional encryption. Then we extend such a trace-and-revoke system to support arbitrary fixed length strings.

We first define a generic transformation similar to the one of [4], which converts an $\mathsf{IND\text{-}CPA}$ secure inner product functional encryption scheme $\mathcal{IPFE}$ into a trace-and-revoke system TR_0 for the restricted message space $\mathcal{M} = \{0,1\}$ that enjoys anonymous revocation. Note that this transformation converts an $\mathsf{IND\text{-}CPA}$ secure IPFE in the bounded collusion model to a trace-and-revoke system TR_0 that supports an exponential number of users like [4]. Then we provide another generic transformation that converts an $\mathsf{IND\text{-}CPA}$ secure matrix multiplication functional encryption scheme (MMFE) into a trace-and-revoke system TR_1 for the message space $\mathcal{M} = \{0,1\}^n$ for n as large as $\mathsf{poly}(\lambda)$. This transformation also ensures that TR_1 achieves anonymous revocation along with supporting an exponential number of users.

As, our primary contribution in this paper, is to introduce trace-and-revoke schemes with anonymous revocation, our presentation mainly focuses on the construction and the anonymity security of TR_0 and TR_1. Nevertheless, in Section 3.1, we have provided a complete description of the TR_0 that includes an explicit description of the Trace function. For the sake of simplicity, we however have presented the general trace-and-revoke systems TR_1 in Section 3.2 without a Trace. Note that, TR_1 can use the Trace algorithm of TR_0.

3.1 Trace-and-Revoke for Single Bit Messages

We construct a trace-and-revoke scheme TR_0 following the specifications of Definition 3 for the message space $\mathcal{M} = \{0,1\}$. TR_0 relies on a user directory dir which contains the identities of all the users that have been assigned keys in the system. This user directory is initially empty. Unlike [4], we assume that dir can only be accessed by the central authority, which is the sender as well as the key generator. TR_0 relies on an inner product functional encryption scheme $\mathcal{IPFE}$ for the ℓ-dimensional vector space on $\mathbb{Z}_p$, where the value ℓ is a function of r and t. Recall that, in a typical trace-and-revoke scheme, the bound on the number of revoked users r and the bound on the number of suspected users (traitors) t are given as the system parameters. Our description of $\mathcal{IPFE}$ (simpler form of $\mathcal{MMFE}$ as noted in Section 2.1) comes with an injective map f whose description is included in the public parameters pp. To define the trace-and-revoke scheme TR_0, we define a special element in the range of the map $\mathit{elem}^* = f(0)$. Concretely, in case of a group-based construction of $\mathcal{IPFE}$, we take the exponentiation map $f : x \mapsto [x]$ and have $\mathit{elem}^* = [0]$. In case of a lattice-based construction, we take the identity map $f : x \mapsto x$ and have $\mathit{elem}^* = 0$.

1. $\mathsf{Setup}(1^\lambda, 1^r, 1^t)$. Upon input the security parameter λ, the bound t on the number of the suspected users, and the bound r on the number of revoked users, set $p = \lambda^{\omega(1)}$ and proceed as follows:
   (a) Let $(\mathsf{pp}, \mathsf{msk}) \leftarrow \mathcal{IPFE}.\mathsf{Setup}(1^\lambda, 1^\ell, p)$, where we set $\ell = 2r + t + 1$. The key space $\mathcal{K}$ and ciphertext space $\mathcal{C}$ are the $\mathcal{IPFE}$ key space and ciphertext space, respectively.
   (b) Create an empty directory dir.
   (c) Output the public parameter pp, master secret key msk and the (empty) user directory dir.
2. $\mathsf{KeyGen}(\mathsf{pp}, \mathsf{msk}, \mathsf{dir}, \mathsf{id})$. Upon input the public parameters pp, the master secret key msk, the user directory dir and a user identity $\mathsf{id} \in \mathsf{ID}$, proceed as follows:
   (a) Sample $\mathbf{x}_{\mathsf{id}} \leftarrow \mathbb{Z}_p^\ell$. The pair $\mathsf{u}_{\mathsf{id}} = (\mathsf{id}, \mathbf{x}_{\mathsf{id}})$ is then appended to dir.
   (b) Let $\mathsf{sk}_{\mathsf{id}} \leftarrow \mathcal{IPFE}.\mathsf{KeyGen}(\mathsf{pp}, \mathsf{msk}, \mathbf{x}_{\mathsf{id}})$.
   (c) Output $(\mathsf{sk}_{\mathsf{id}}, \mathbf{x}_{\mathsf{id}})$.
3. $\mathsf{Enc}(\mathsf{pp}, \mathsf{msk}, \mathsf{dir}, \mathcal{R}, m)$. Upon input the public parameters pp, the master secret key msk, the user directory dir, a set of revoked users $\mathcal{R}$ of size $\leq r$ and a plaintext message $m \in \mathcal{M} = \{0,1\}$, proceed as follows:
   (a) Sample $\mathbf{v}_{\mathcal{R}} \leftarrow \mathbf{X}_{\mathcal{R}}^{\perp}$ where $\mathbf{X}_{\mathcal{R}} = \{\mathbf{x}_{\mathsf{id}} \ : \ \mathsf{id} \in \mathcal{R}\}$.
   (b) Compute $\mathbf{y}_{\mathcal{R}} = m \cdot \mathbf{v}_{\mathcal{R}}$.
   (c) Output $C_{\mathcal{R}} = \mathcal{IPFE}.\mathsf{Enc}(\mathsf{pp}, \mathsf{msk}, \mathbf{y}_{\mathcal{R}})$.
4. $\mathsf{Dec}(\mathsf{pp}, (\mathsf{sk}_{\mathsf{id}}, \mathbf{x}_{\mathsf{id}}), C_{\mathcal{R}})$. Upon input the public parameters pp, the secret key $\mathsf{sk}_{\mathsf{id}}$ for user id and a ciphertext $C_{\mathcal{R}}$, proceed as follows:
   (a) Compute $\mathsf{Res} = \mathcal{IPFE}.\mathsf{Dec}(\mathsf{pp}, (\mathsf{sk}_{\mathsf{id}}, \mathbf{x}_{\mathsf{id}}), C_{\mathcal{R}})$.
   (b) If $\mathsf{Res} = \mathit{elem}^*$, then output 0. Otherwise output 1.
5. $\mathsf{Trace}(\mathsf{pp}, \mathsf{msk}, \mathsf{dir}, \mathcal{R}, \mathcal{S}, \mathcal{O}^{\mathcal{D}})$. Upon input the master secret key msk, the user directory dir, a revoked set of users $\mathcal{R}$, a suspect set of users $\mathcal{S}$ and given access to the oracle $\mathcal{O}^{\mathcal{D}}$, proceed as follows:

(a) Suppose the users in the suspect set $\mathcal{S}$ can distinguish between the messages $m = 0$ and $m' = 1$ except with negligible probability provided these users can access the oracle $\mathcal{O}^{\mathcal{D}}$.[8]
(b) Set $\mathcal{S}_1 = \{\mathsf{id}_1, \mathsf{id}_2, \ldots\} = \mathcal{S} \setminus \mathcal{R}$.
(c) Sample $\mathbf{v}_{\mathcal{R}} \leftarrow \mathbf{X}_{\mathcal{R}}^{\perp}$ where $\mathbf{X}_{\mathcal{R}} = \{\mathbf{x}_{\mathsf{id}} \ : \ \mathsf{id} \in \mathcal{R}\}$.
(d) For all $i = 1, 2, \ldots, t$,
   - If $i = 1$, set $\mathbf{v}_{\mathcal{S}_i} = \mathbf{0}$. If $\mathcal{S}_i = \emptyset$, set $\mathbf{v}_{\mathcal{S}_i} = (m' - m) \cdot \mathbf{v}_{\mathcal{R}}$.
   - Otherwise, sample $\mathbf{v}_{\mathcal{S}_i} \leftarrow \mathbf{X}_{\mathcal{R} \cup \mathcal{S}_i}^{\perp} \cap \left(\mathbf{X}_{\mathcal{S}_1 \setminus \mathcal{S}_i}^{\perp} + (m' - m) \cdot \mathbf{v}_{\mathcal{R}}\right)$ where $\mathbf{X}_{\mathcal{R} \cup \mathcal{S}_i} = \{\mathbf{x}_{\mathsf{id}} : \mathsf{id} \in \mathcal{R} \cup \mathcal{S}_i\}$ and $\mathbf{X}_{\mathcal{S}_1 \setminus \mathcal{S}_i} = \{\mathbf{x}_{\mathsf{id}} : \mathsf{id} \in \mathcal{S}_1 \setminus \mathcal{S}_i\}$.
   - Construct $\mathbf{y}_i = \mathbf{v}_{\mathcal{S}_i} + m \cdot \mathbf{v}_{\mathcal{R}}$;
   - Provide the oracle $\mathcal{O}^{\mathcal{D}}$ with $(C_{\mathcal{S}_i}, m)$ as input and get a binary value b_i as output. Suppose the probability of $b_i = 1$ is p_i.
   - The probe ciphertext is $C_{\mathcal{S}_i} = \mathcal{IPFE}.\mathsf{Enc}(\mathsf{pp}, \mathsf{msk}, \mathbf{y}_i)$; We note that, the decryption result of the probe ciphertext $C_{\mathcal{S}_i}$ is m if $\mathsf{id} \in \mathcal{S}_i$ and m' if $\mathsf{id} \in \mathcal{S} \setminus \mathcal{S}_i$.
   - If $i > 1$ and $|p_i - p_{i-1}|$ is non-negligible,
     - Output id_{i-1} as the traitor identity and abort;
     - If $\mathcal{S}_i = \phi$, output $\perp$ and abort. Otherwise, set $\mathcal{S}_{i+1} = \mathcal{S}_i \setminus \{\mathsf{id}_i\}$.

We first check the correctness of the scheme, whose proof is adapted from the correctness proof of [4]

Theorem 1. *Assume that $p = \lambda^{\omega(1)}$. Then, for every set $\mathcal{R}$ of revoked users of size $\leq r$, every $\mathsf{id} \notin \mathcal{R}$ and every $m \in \mathcal{M} = \{0, 1\}$, we have*

$$\mathsf{Dec}(\mathsf{pp}, (\mathsf{sk}_{\mathsf{id}}, \mathbf{x}_{\mathsf{id}}), \mathsf{Enc}(\mathsf{pp}, \mathsf{msk}, \mathsf{dir}, \mathcal{R}, m)) = m,$$

with probability $\geq 1 - \lambda^{-\omega(1)}$.

Proof. As $\mathbf{x}_{\mathsf{id}}$ is uniform in $\mathbb{Z}_p^{\ell}$, $p = \lambda^{\omega(1)}$ and $\ell > r$, we have that $\langle \mathbf{x}_{\mathsf{id}}, \mathbf{v}_{\mathcal{R}} \rangle \neq 0$, with overwhelming probability. The execution of $\mathsf{Dec}(\mathsf{pp}, (\mathsf{sk}_{\mathsf{id}}, \mathbf{x}_{\mathsf{id}}), C_{\mathcal{R}})$, with $C_{\mathcal{R}} = \mathsf{Enc}(\mathsf{pp}, \mathsf{msk}, \mathsf{dir}, \mathcal{R}, m)$, on Step (a) computes (with overwhelming probability):

$$\mathsf{Dec}(\mathsf{pp}, (\mathsf{sk}_{\mathsf{id}}, \mathbf{x}_{\mathsf{id}}), C_{\mathcal{R}}) = f(\langle \mathbf{x}_{\mathsf{id}}, \mathbf{y}_{\mathcal{R}} \rangle) = f(m \cdot \langle \mathbf{x}_{\mathsf{id}}, \mathbf{v}_{\mathcal{R}} \rangle)$$

by the correctness of $\mathcal{IPFE}$ where f is the deterministic function included in pp.

Now, observe that, if $m = 0$, then $f(\langle \mathbf{x}_{\mathsf{id}}, \mathbf{y}_{\mathcal{R}} \rangle) = f(0) = \mathit{elem}^*$. In this case, Dec outputs 0. On the other hand, if $m = 1$, then $f(\langle \mathbf{x}_{\mathsf{id}}, \mathbf{y}_{\mathcal{R}} \rangle) = f(\langle \mathbf{x}_{\mathsf{id}}, \mathbf{v}_{\mathcal{R}} \rangle) \neq \mathit{elem}^*$ (since $\langle \mathbf{x}_{\mathsf{id}}, \mathbf{v}_{\mathcal{R}} \rangle \neq 0$ and f is injective). In this case, Dec outputs 1. Thus, for both values of m, Dec retrieves the correct value of m with overwhelming probability. □

Now, we show that Step (a) of Trace is indeed successful, i.e., we can use $\mathcal{O}^{\mathcal{D}}$ (satisfying Equation (1) for $\mathcal{M} = \{0, 1\}$) to distinguish between $m = 0$ and $m' = 1$.

Theorem 2. *Let $\mathcal{R}$ be arbitrary of size $\leq r$ and assume Equation (1) holds for $\mathcal{O}^{\mathcal{D}}$ and $\mathcal{R}$. Then we have:*

$$\left| \Pr_{C \leftarrow \mathsf{Enc}(\mathsf{pp}, \mathsf{msk}, \mathsf{dir}, \mathcal{R}, 0)} [\mathcal{O}^{\mathcal{D}}(C, 0) = 1] - \Pr_{C \leftarrow \mathsf{Enc}(\mathsf{pp}, \mathsf{msk}, \mathsf{dir}, \mathcal{R}, 1)} [\mathcal{O}^{\mathcal{D}}(C, 0) = 1] \right| \geq \frac{2}{\lambda^c},$$

with probability $\geq 1 - \lambda^{-\omega(1)}$ and for some constant $c > 0$.

Proof. By Equation (1), we have

$$\Pr_{C \leftarrow \mathsf{Enc}(\mathsf{pp}, \mathsf{msk}, \mathsf{dir}, \mathcal{R}, 0)} [\mathcal{O}^{\mathcal{D}}(C, 0) = 1] \geq \frac{1}{2} + \frac{1}{\lambda^c}, \quad \text{and} \quad \Pr_{C \leftarrow \mathsf{Enc}(\mathsf{pp}, \mathsf{msk}, \mathsf{dir}, \mathcal{R}, 1)} [\mathcal{O}^{\mathcal{D}}(C, 1) = 1] \geq \frac{1}{2} + \frac{1}{\lambda^c}.$$

[8] Note that [4] used Hoeffding's inequality to ensure that one can efficiently find such distinguishable m and m'. In our case, it is simpler, as $\mathcal{M} = \{0, 1\}$.

The latter means that if $m' = 1$ is encrypted as C, then $\mathcal{O}^{\mathcal{D}}(C, 1)$ outputs 1 with probability non-negligibly better than a random choice. Taking the complement, we obtain that

$$\Pr_{C \leftarrow \mathsf{Enc}(\mathsf{pp}, \mathsf{msk}, \mathsf{dir}, \mathcal{R}, 1)} [\mathcal{O}^{\mathcal{D}}(C, 0) = 1] < \frac{1}{2} - \frac{1}{\lambda^c}.$$

The result the follows naturally. □

Security. We prove that the base scheme TR_0 enjoys message hiding, revocation set hiding and traceability.

Theorem 3. *If $\mathcal{IPFE}$ is an* $\mathsf{IND\text{-}CPA}$ *secure inner product functional encryption scheme allowing up to r key extraction queries, then* TR_0 *is* $\mathsf{IND\text{-}CPA}$ *secure.*

Proof. Let $\mathcal{A}_{\mathsf{TR}_0}$ be a ppt adversary that breaks the $\mathsf{IND\text{-}CPA}$ security of TR_0. We construct a ppt adversary $\mathcal{A}_{\mathcal{IPFE}}$ that breaks the $\mathsf{IND\text{-}CPA}$ security of the underlying $\mathcal{IPFE}$:

- It first obtains the public parameter pp output by the $\mathcal{IPFE}$ challenger (which runs the $\mathcal{IPFE}.\mathsf{Setup}(1^\lambda, 1^\ell)$ algorithm) and relays it to $\mathcal{A}_{\mathsf{TR}_0}$. On $\mathcal{A}_{\mathsf{TR}_0}$'s request, the adversary $\mathcal{A}_{\mathcal{IPFE}}$ creates dir with polynomially many $(\mathsf{id}, \mathbf{x}_{\mathsf{id}})$ pairs for $\mathbf{x}_{\mathsf{id}} \leftarrow \mathbb{Z}_p^\ell$. The $\mathcal{IPFE}$ challenger samples $\beta \leftarrow \{0, 1\}$.
- The adversary $\mathcal{A}_{\mathsf{TR}_0}$ can make multiple secret key queries on $\mathsf{id} \in \mathsf{ID}$ and multiple challenge ciphertext queries on $(m_0, m_1, \mathcal{R})$.
  - For every secret key query on id,
    - ∗ $\mathcal{A}_{\mathcal{IPFE}}$ retrieves $\mathbf{x}_{\mathsf{id}} = \mathsf{dir}[\mathsf{id}]$.
    - ∗ $\mathcal{A}_{\mathcal{IPFE}}$ then sends $\mathbf{x}_{\mathsf{id}}$ to the $\mathcal{IPFE}$ challenger. The latter returns $\mathsf{sk}_{\mathbf{x}_{\mathsf{id}}}$, which $\mathcal{A}_{\mathcal{IPFE}}$ forwards to $\mathcal{A}_{\mathsf{TR}_0}$ as $\mathsf{sk}_{\mathsf{id}}$.
  - For every challenge anonymity query on $(m_0, m_1, \mathcal{R})$,
    - ∗ It samples $\mathbf{v}_{\mathcal{R}} \leftarrow \mathbf{X}^\perp$ where $\mathbf{X} = \{\mathbf{x}_{\mathsf{id}} \in \mathbb{Z}_p^\ell \ : \ \mathsf{id} \in \mathcal{R}\}$.
    - ∗ It sends $\mathbf{y}_0 = m_0 \cdot \mathbf{v}_{\mathcal{R}}$ and $\mathbf{y}_1 = m_1 \cdot \mathbf{v}_{\mathcal{R}}$ to the $\mathcal{IPFE}$ challenger. The latter encrypts $\mathbf{y}_\beta$ as $\mathsf{ct}^{(\beta)} \leftarrow \mathcal{IPFE}.\mathsf{Enc}(\mathsf{pp}, \mathsf{msk}, \mathbf{y}_\beta)$ and outputs $\mathsf{ct}^{(\beta)}$.
    - ∗ It forwards the received ciphertext to $\mathcal{A}_{\mathsf{TR}_0}$ as its challenge $C^{(\beta)}$.
- Finally, the $\mathcal{A}_{\mathsf{TR}_0}$ adversary outputs its guess $\beta' \in \{0, 1\}$ and $\mathcal{A}_{\mathcal{IPFE}}$ also outputs β' as its own guess of β.

Note that adversary $\mathcal{A}_{\mathcal{IPFE}}$ behaves as an $\mathsf{IND\text{-}CPA}$ challenger in the view of $\mathcal{A}_{\mathsf{TR}_0}$. Further, it is a valid adversary against $\mathcal{IPFE}$ as $\langle \mathbf{y}_0, \mathbf{x}_{\mathsf{id}} \rangle = \langle \mathbf{y}_1, \mathbf{x}_{\mathsf{id}} \rangle = 0$ for every vector $\mathbf{x}_{\mathsf{id}}$ queried to the $\mathcal{IPFE}$ challenger (i.e., each $\mathsf{id} \in \mathcal{R}$). The advantage of $\mathcal{A}_{\mathcal{IPFE}}$ is exactly the same as the advantage of $\mathcal{A}_{\mathsf{TR}_0}$. □

Theorem 4. *If $\mathcal{IPFE}$ is an* $\mathsf{IND\text{-}CPA}$ *secure inner product functional encryption scheme allowing up to $(t+r)$ key extraction queries, then* TR_0 *is* $\mathsf{mIND\text{-}ID\text{-}CPA}$ *secure.*

Proof. Given an $\mathsf{mIND\text{-}ID\text{-}CPA}$ adversary $\mathcal{A}_{\mathsf{TR}_0}$, we produce $\mathcal{A}_{\mathcal{IPFE}}$ that breaks the $\mathsf{IND\text{-}CPA}$ security of $\mathcal{IPFE}$.

- $\mathcal{A}_{\mathcal{IPFE}}$ first obtains the public parameter pp output by the $\mathcal{IPFE}$ challenger (who runs the $\mathcal{IPFE}.\mathsf{Setup}(1^\lambda, 1^\ell)$ algorithm) and relays it to $\mathcal{A}_{\mathsf{TR}_0}$. The $\mathcal{IPFE}$ challenger, at this point, samples $\beta \leftarrow \{0, 1\}$. On $\mathcal{A}_{\mathsf{TR}_0}$'s request, $\mathcal{A}_{\mathcal{IPFE}}$ creates dir with polynomially many id without the corresponding $\mathbf{x}_{\mathsf{id}}$.
- Recall that, $\mathcal{A}_{\mathsf{TR}_0}$ can make multiple secret key queries on $\mathsf{id} \in \mathsf{ID}$ and multiple challenge ciphertext queries on $(m, \mathcal{R}_0, \mathcal{R}_1)$. To accommodate such queries, $\mathcal{A}_{\mathcal{IPFE}}$ first defines a set of vector $\mathcal{VS} = \{\mathbf{x}_1, \ldots, \mathbf{x}_{t+2r}\}$ where $\mathbf{x}_i \leftarrow \mathbb{Z}_p^\ell$. This set is used to answer to secret key queries.
  - For every secret key query on id,
    - ∗ If $\mathsf{id} \in \mathsf{dir}$, $\mathcal{A}_{\mathcal{IPFE}}$ sets $\mathbf{x} = \mathsf{dir}[\mathsf{id}]$.
    - ∗ Otherwise, $\mathcal{A}_{\mathcal{IPFE}}$ samples a vector $\mathbf{x} \leftarrow \mathcal{VS}$ and sets $\mathsf{dir}[\mathsf{id}] = \mathbf{x}$.
    - ∗ $\mathcal{A}_{\mathcal{IPFE}}$ then sends $\mathbf{x}$ to the $\mathcal{IPFE}$ challenger. The latter returns $\mathsf{sk}_{\mathbf{x}_{\mathsf{id}}}$, which $\mathcal{A}_{\mathcal{IPFE}}$ forwards to $\mathcal{A}_{\mathsf{TR}_0}$ as $\mathsf{sk}_{\mathsf{id}}$.
  - For every challenge anonymity query on $(m, \mathcal{R}_0, \mathcal{R}_1)$,
    - ∗ For every $\mathsf{id} \in \mathcal{R}_0 \cup \mathcal{R}_1$, if $\mathsf{id} \notin \mathsf{dir}$, $\mathcal{A}_{\mathcal{IPFE}}$ samples a vector $\mathbf{x} \leftarrow \mathcal{VS}$ without repetition and sets $\mathsf{dir}[\mathsf{id}] = \mathbf{x}$.

  - $\mathcal{A}_{\mathcal{IPFE}}$ defines $\hat{\mathcal{R}} = \mathcal{R}_0 \cap \mathcal{R}_1$.
  - Then $\mathcal{A}_{\mathcal{IPFE}}$ defines three matrices:
    1. $\mathbf{Z} = \mathsf{Matrix}(\mathcal{VS} \setminus Z)$ where $Z = \{\mathbf{x}_{\mathsf{id}} : \mathsf{id} \in \hat{\mathcal{R}}\}$.
    2. $\mathbf{X}_0 = \mathsf{Matrix}(X_0)$ where $X_0 = \{\mathbf{x}_{\mathsf{id}} : \mathsf{id} \in R_0\}$.
    3. $\mathbf{X}_1 = \mathsf{Matrix}(X_1)$ where $X_1 = \{\mathbf{x}_{\mathsf{id}} : \mathsf{id} \in \mathcal{R}_1\}$.
  - It samples $\begin{pmatrix} \mathbf{v}_{\mathcal{R}_0} \\ \mathbf{v}_{\mathcal{R}_1} \end{pmatrix} \leftarrow \mathbf{V}^{\perp}$ where
$$\mathbf{V} = \begin{pmatrix} \mathbf{Z} & -\mathbf{Z} \\ \mathbf{X}_0 & \mathbf{0} \\ \mathbf{0} & \mathbf{X}_1 \end{pmatrix}. \qquad (2)$$
  - It sends $\mathbf{y}_{\mathcal{R}_0} = m \cdot \mathbf{v}_{\mathcal{R}_0}$ and $\mathbf{y}_{\mathcal{R}_1} = m \cdot \mathbf{v}_{\mathcal{R}_1}$ to the $\mathcal{IPFE}$ challenger encrypts $\mathbf{y}_{\mathcal{R}_\beta}$ as $\mathsf{ct}^{(\beta)} \leftarrow \mathcal{IPFE}.\mathsf{Enc}(\mathsf{pp}, \mathsf{msk}, \mathbf{y}_{\mathcal{R}_\beta})$ and outputs $\mathsf{ct}^{(\beta)}$.
  - $\mathcal{A}_{\mathcal{IPFE}}$ then forwards the received ciphertext to $\mathcal{A}_{\mathsf{TR}_0}$ as its challenge $C^{(\beta)}$.
- At the end of the game, $\mathcal{A}_{\mathsf{TR}_0}$ returns β' as its guess of β which $\mathcal{A}_{\mathcal{IPFE}}$ forwards to the $\mathcal{IPFE}$ challenger as its answer.

From Equation (2), $\mathbf{Z}(\mathbf{v}_{\mathcal{R}_0} - \mathbf{v}_{\mathcal{R}_1}) = \mathbf{0}$, $\mathbf{X}_0\mathbf{v}_{\mathcal{R}_0} = \mathbf{0}$ and $\mathbf{X}_1\mathbf{v}_{\mathcal{R}_1} = \mathbf{0}$. As $\mathbf{y}_{\mathcal{R}_u} \in \mathrm{Span}(\mathbf{v}_{\mathcal{R}_u})$ for $u \in \{0,1\}$, $\mathbf{Z}(\mathbf{y}_{\mathcal{R}_1} - \mathbf{y}_{\mathcal{R}_0}) = \mathbf{0}$ and $\mathbf{X}_0\mathbf{y}_{\mathcal{R}_0} = \mathbf{X}_1\mathbf{y}_{\mathcal{R}_1} = \ \mathbf{0}$.

We now show that $\mathcal{A}_{\mathcal{IPFE}}$ is a valid challenger against $\mathcal{A}_{\mathsf{TR}_0}$ in the mIND-ID-CPA security model. For that we show, for every i^{th} key query on id_i and j^{th} challenge ciphertext query on $(m^{(j)}, \mathcal{R}_0^{(j)}, \mathcal{R}_1^{(j)})$ from $\mathcal{A}_{\mathsf{TR}_0}$, $\mathcal{A}_{\mathcal{IPFE}}$ can forward corresponding vectors to the $\mathcal{IPFE}$ challenger. Due to the natural restriction, note that, $\mathsf{id}_i \in (\mathcal{R}_0^{(j)} \cap \mathcal{R}_1^{(j)}) \sqcup (\mathsf{ID} \setminus (\mathcal{R}_0^{(j)} \cup \mathcal{R}_1^{(j)}))$ for all $i \in [1, t+r-1]$ and all $j \in [1, r]$.

For all queried id_i, if one of the following two holds.

- $\mathsf{id}_i \in (\mathcal{R}_0^{(j)} \cap \mathcal{R}_1^{(j)})$: This means, $\mathsf{id}_i \notin (\mathcal{R}_0^{(t)} \Delta \mathcal{R}_1^{(t)})$ for all $t \in [1, j-1]$ due to the natural restriction. The corresponding $\mathbf{x}_{\mathsf{id}_i} \in \mathcal{VS} \cap \hat{\mathcal{R}}^{(j)}$ and by our reduction, the $\mathbf{x}_{\mathsf{id}_i}$ vector is included in the definition of $\mathbf{X}_0^{(j)}$ and $\mathbf{X}_1^{(j)}$. From Equation (2) above, we see that $\mathbf{X}_0^{(j)}\mathbf{y}_{\mathcal{R}_0}^{(j)} = \mathbf{X}_1^{(j)}\mathbf{y}_{\mathcal{R}_1}^{(j)} = \mathbf{0}$. Thus, $\mathcal{A}_{\mathcal{IPFE}}$ can forward this to the $\mathcal{IPFE}$ challenger for key query.
- $\mathsf{id}_i \in \mathsf{ID} \setminus (\mathcal{R}_0^{(j)} \cup \mathcal{R}_1^{(j)})$: Observe that, the corresponding $\mathbf{x}_{\mathsf{id}_i} \in \mathcal{VS} \setminus \hat{\mathcal{R}}^{(j)}$ and such $\mathbf{x}_{\mathsf{id}_i}$ is included in the definition of $\mathbf{Z}^{(j)}$. From Equation (2) above, we see that $\mathbf{Z}^{(j)}\mathbf{y}_{\mathcal{R}_0}^{(j)} = \mathbf{Z}^{(j)}\mathbf{y}_{\mathcal{R}_1}^{(j)} \neq \mathbf{0}$ (w.h.p. as $\mathbf{y}_{\mathcal{R}_0}^{(j)}, \mathbf{y}_{\mathcal{R}_1}^{(j)}$ are sampled randomly). Thus, $\mathcal{A}_{\mathcal{IPFE}}$ can forward this to the $\mathcal{IPFE}$ challenger for key query.

Next, note that, for every query on id_i from $\mathcal{A}_{\mathsf{TR}_0}$, the adversary $\mathcal{A}_{\mathcal{IPFE}}$ returns a distinct random vector $\mathbf{x}_{\mathsf{id}_i}$ from $\mathcal{VS}$ that were sampled at the starting of the reduction. The crucial point here is $\mathcal{A}_{\mathcal{IPFE}}$ faces at most $(t+2r)$ many distinct identities id, hence $\mathcal{VS}$ is sufficient to assign the corresponding $\mathbf{x}_{\mathsf{id}}$. Moreover, $\mathcal{A}_{\mathsf{TR}_0}$ gets encryption of either $\mathbf{y}_{\mathcal{R}_0}$ or $\mathbf{y}_{\mathcal{R}_1}$ where both the vectors are randomly sampled. Thus, from the point of view of $\mathcal{A}_{\mathsf{TR}_0}$, $\mathbf{Z}\mathbf{y}_{\mathcal{R}_b}$ is a random vector. Thus, $\mathcal{A}_{\mathcal{IPFE}}$ behaves as a valid mIND-ID-CPA challenger to $\mathcal{A}_{\mathsf{TR}_0}$.

As we have seen above, for every $\mathbf{x}_{\mathsf{id}_i}$ and $(\mathbf{y}_{\mathcal{R}_0}^{(j)}, \mathbf{y}_{\mathcal{R}_1}^{(j)})$ the adversary $\mathcal{A}_{\mathcal{IPFE}}$ gives to the $\mathcal{IPFE}$ challenger, $\langle \mathbf{x}_{\mathsf{id}_i}, \mathbf{y}_{\mathcal{R}_0}^{(j)} \rangle = \langle \mathbf{x}_{\mathsf{id}_i}, \mathbf{y}_{\mathcal{R}_1}^{(j)} \rangle$ holds. Thus, $\mathcal{A}_{\mathcal{IPFE}}$ behaves as a valid IND-CPA adversary to the $\mathcal{IPFE}$ challenger.

If $\mathcal{A}_{\mathsf{TR}_0}$ can distinguish between any $\mathcal{R}_0^{(j)}$ and $\mathcal{R}_1^{(j)}$, $\mathcal{A}_{\mathcal{IPFE}}$ can distinguish between corresponding $\mathbf{y}_{\mathcal{R}_0}^{(j)}$ and $\mathbf{y}_{\mathcal{R}_1}^{(j)}$. Thus, the advantage of $\mathcal{A}_{\mathcal{IPFE}}$ is exactly the same as the advantage of $\mathcal{A}_{\mathsf{TR}_0}$. □

Theorem 5. *If $\mathcal{IPFE}$ is an* IND-CPA *secure inner product functional encryption scheme allowing $(r+t)$ queries, then* TR_0 *is* AD-TT *secure.*

Proof. Given an AD-TT adversary $\mathcal{A}_{\mathsf{TR}_0}$, we have to produce $\mathcal{A}_{\mathcal{IPFE}}$ that breaks the IND-CPA security of $\mathcal{IPFE}$. $\mathcal{A}_{\mathcal{IPFE}}$ first obtains the public parameter pp output by the $\mathcal{IPFE}$ challenger (who runs the $\mathcal{IPFE}.\mathsf{Setup}(1^{\lambda}, 1^{\ell})$ algorithm) and relays it to $\mathcal{A}_{\mathsf{TR}_0}$. On $\mathcal{A}_{\mathsf{TR}_0}$'s request, $\mathcal{A}_{\mathcal{IPFE}}$ creates dir with polynomially many id without the corresponding $\mathbf{x}_{\mathsf{id}}$. The $\mathcal{IPFE}$ challenger, being a symmetric key primitive, provides $\mathcal{A}_{\mathcal{IPFE}}$ polynomially many accesses to the encryption oracle $O_{\mathsf{ct}}(\cdot)$ and to the key generation oracle $O_{\mathsf{sk}}(\cdot)$.

$\mathcal{A}_{\mathsf{TR}_0}$ adaptively chooses $\mathsf{id} \in \mathsf{ID}$, $\mathcal{A}_{\mathcal{IPFE}}$ assigns a random $\mathbf{x}_{\mathsf{id}}$ to $\mathsf{dir}[\mathsf{id}]$ and makes query to O_{sk} on $\mathbf{x}_{\mathsf{id}}$. The response it gets is forwarded to $\mathcal{A}_{\mathsf{TR}_0}$ as the secret key $\mathsf{sk}_{\mathsf{id}}$. $\mathcal{A}$ can make at most t many such queries and these queries are collected as a set $\mathcal{T}$.

$\mathcal{A}_{\mathsf{TR}_0}$ then adaptively chooses $\mathcal{R} \subset \mathsf{ID}$ such that $|\mathcal{R}| \leq r$. For every $\mathsf{id} \in \mathcal{R}$, $\mathcal{A}_{\mathcal{IPFE}}$ assigns a $\mathbf{x}_{\mathsf{id}}$ and makes query to O_{sk} on $\mathbf{x}_{\mathsf{id}}$, the response it gets is forwarded to $\mathcal{A}_{\mathsf{TR}_0}$ as the secret key $\mathsf{sk}_{\mathsf{id}}$.

Finally, $\mathcal{A}$ produces a pirate decoder $\mathcal{O}^{\mathcal{D}}$ and a suspected list of traitors $\mathcal{S}$ that includes the traitor set $\mathcal{T}$ where $|\mathcal{S}| \leq t$. Next, $\mathcal{A}_{\mathcal{IPFE}}$ runs Trace on $\mathcal{S}$ and $\mathcal{R}$ given access to O_{ct} and $\mathcal{O}^{\mathcal{D}}$. Precisely, for all $i \in [1, |\mathcal{S}|]$, $\mathcal{A}_{\mathcal{IPFE}}$ computes $\mathbf{v}_{\mathcal{S}_i}$ and asks O_{ct} to get the so-called probe-ciphertext $C_{\mathcal{S}_i}$. Finally, Trace outputs either $\perp$ or some $\mathsf{id} \in \mathcal{S}$. More specifically, the winning condition of $\mathsf{AD\text{-}TT}$ security model tells that Trace outputs either $\perp$ or some $\mathsf{id} \in \mathcal{S} \setminus \mathcal{T}$ with probability $\geq 1/\lambda^c$ for some constant $c > 0$.

In case, Trace outputs $\perp$, $\mathcal{A}_{\mathcal{IPFE}}$ outputs a random bit. Otherwise, we assume id to be id_{i-1} for which Trace aborted on the i^{th} round for some $i < t$. Then, by the description of Trace, $|p_i - p_{i-1}|$ is non-negligible. At this point $\mathcal{A}_{\mathcal{IPFE}}$ retrieves $\mathbf{v}_{\mathcal{S}_{i-1}}$ and $\mathbf{v}_{\mathcal{S}_i}$ to define $\mathbf{y}_0 = \mathbf{v}_{\mathcal{S}_{i-1}} + m \cdot \mathbf{v}_{\mathcal{R}}$ and $\mathbf{y}_1 = \mathbf{v}_{\mathcal{S}_i} + m \cdot \mathbf{v}_{\mathcal{R}}$ and makes challenge ciphertext query to the $\mathcal{IPFE}$ challenger where $\mathcal{S}_i = \mathcal{S}_{i-1} \setminus \{\mathsf{id}_{i-1}\}$. The $\mathcal{IPFE}$ challenger responds with $C^{(\beta)} \leftarrow \mathsf{Enc}(\mathsf{pp}, \mathsf{msk}, \mathbf{y}_\beta)$ for $\beta \leftarrow \{0,1\}$. $\mathcal{A}_{\mathcal{IPFE}}$ runs $\beta' \leftarrow \mathcal{O}^{\mathcal{D}}(C^{(\beta)}, m)$ and outputs $(1 - \beta')$.

Here, we first show that $\mathcal{A}_{\mathcal{IPFE}}$ is a valid adversary in the $\mathsf{IND\text{-}CPA}$ security model. In other words, we show that for all secret key queries on $\mathsf{id} \in \mathcal{R} \cup \mathcal{T}$, $\langle \mathbf{y}_0, \mathbf{x}_{\mathsf{id}} \rangle = \langle \mathbf{y}_1, \mathbf{x}_{\mathsf{id}} \rangle$. This can be seen from the following:

1. $\mathsf{id} \in \mathcal{R}$: $\langle \mathbf{y}_0, \mathbf{x}_{\mathsf{id}} \rangle = \langle \mathbf{y}_1, \mathbf{x}_{\mathsf{id}} \rangle = 0$.
2. $\mathsf{id} \in \mathcal{T} \cap \mathcal{S}_{i-1}$: $\langle \mathbf{y}_0, \mathbf{x}_{\mathsf{id}} \rangle = \langle \mathbf{y}_1, \mathbf{x}_{\mathsf{id}} \rangle = m \cdot \langle \mathbf{v}_{\mathcal{R}}, \mathbf{x}_{\mathsf{id}} \rangle$.
3. $\mathsf{id} \in \mathcal{T} \cap (\mathcal{S}_1 \setminus \mathcal{S}_{i-1})$: $\langle \mathbf{y}_0, \mathbf{x}_{\mathsf{id}} \rangle = \langle \mathbf{y}_1, \mathbf{x}_{\mathsf{id}} \rangle = m' \cdot \langle \mathbf{v}_{\mathcal{R}}, \mathbf{x}_{\mathsf{id}} \rangle$.

Now, we show that $\mathcal{A}_{\mathcal{IPFE}}$ wins with probability given Trace didn't output $\perp$.

If $\beta = 0$. $C^{(\beta)}$ is an encryption of $\mathbf{y}_0$ that encoded $\mathbf{v}_{\mathcal{S}_{i-1}}$. The description of the Trace tells that $\langle \mathbf{v}_{\mathcal{S}_{i-1}}, \mathbf{x}_{\mathsf{id}_{i-1}} \rangle = 0$. Thus, given $\mathcal{O}^{\mathcal{D}}$ one views $C^{(\beta)}$ as an encryption of $\mathbf{y}_0 = m \cdot \mathbf{v}_{\mathcal{R}}$. In this case, $\mathcal{O}^{\mathcal{D}}(C^{(\beta)}, m)$ gives $\beta' = 1$ with very high probability.

If $\beta = 1$. $C^{(\beta)}$ is an encryption of $\mathbf{y}_1$ that encoded $\mathbf{v}_{\mathcal{S}_i}$. The description of the Trace tells that $\langle \mathbf{v}_{\mathcal{S}_i}, \mathbf{x}_{\mathsf{id}_{i-1}} \rangle = (m' - m) \cdot \langle \mathbf{v}_{\mathcal{R}}, \mathbf{x}_{\mathsf{id}_{i-1}} \rangle$. Thus, given $\mathcal{O}^{\mathcal{D}}$ one views $C^{(\beta)}$ as an encryption of $\mathbf{y}_1 = (m' - m) \cdot \mathbf{v}_{\mathcal{R}} + m \cdot \mathbf{v}_{\mathcal{R}} = m' \cdot \mathbf{v}_{\mathcal{R}}$. In this case, $\mathcal{O}^{\mathcal{D}}(C^{(\beta)}, m)$ gives $\beta' = 0$ with very high probability due to the so-called usefulness of $\mathcal{O}^{\mathcal{D}}$.

Thus, when $\mathcal{A}_{\mathsf{TR}_0}$ gives β', $\mathcal{A}_{\mathcal{IPFE}}$ just forwards $(1 - \beta')$ as its guess of β.

Now, we prove that the probability that Trace outputs $\perp$ is negligible. We mention, [4, Lemma 17] already have made this argument. However, for completeness, we overview the argument here. From Theorem 2, we see that $\mathcal{O}^{\mathcal{D}}$ distinguishes between $m = 0$ and $m' = 1$ with probability $\geq 2/\lambda^c$ for some constant $c > 0$. The description of Trace tells that $\left| \sum_{i \in [1,t]} (p_i - p_{i-1}) \right| \geq 2/\lambda^c$ that is non-negligible. Then, by triangle inequality, there exists an i such that $|p_i - p_{i-1}|$ is non-negligible. Thus, Trace outputs id_{i-1} with non-negligible probability and aborts. Therefore, the probability that Trace continues t many iterations and outputs $\perp$ is negligible. □

3.2 Efficient Trace-and-Revoke for Bit Strings

We present a trace-and-revoke scheme TR_1 for $\mathcal{M} = \{0,1\}^n$ that does not run parallel independent n executions of TR_0. However, we note that, we omit the description of Trace here as it follows from the Trace algorithm of TR_0. This scheme again assumes the existence of a user directory dir which is initialized to be empty, contains the identities of the users that have been assigned keys in the system. We assume that dir can only be modified by the central authority who is the sender as well as the key generator. Here, we assume existence of an efficient matrix multiplication functional encryption $\mathcal{MMFE}$ that encrypts matrices of $n \times \ell$ dimension. The intuitive idea here is that, we utilize n copies of inner product of ℓ dimensional vectors as a linear system of equations $\mathbf{Mx}$ where $\mathbf{M} \in \mathbb{Z}_p^{n \times \ell}$ and $\mathbf{x} \in \mathbb{Z}_p^{\ell}$. Each of the rows of $\mathbf{M}$ is used to encrypt each message bit.

1. $\mathsf{Setup}(1^\lambda, 1^n, 1^r, 1^t)$. Upon input the security parameter λ, the message bit-length n, the bound t on the number of the suspected users and the bound r on the number of revoked users, set $p = \lambda^{\omega(1)}$ and proceed as follows:
   (a) Let $(\mathsf{pp}, \mathsf{msk}) \leftarrow \mathcal{MMFE}.\mathsf{Setup}(1^\lambda, 1^\ell, 1^n, p)$, where we set $\ell = 2r + t + n + 1$.
   (b) Output the public parameter pp, master secret key msk and an empty user directory dir.
2. $\mathsf{KeyGen}(\mathsf{pp}, \mathsf{msk}, \mathsf{dir}, \mathsf{id})$. Upon input the public parameters pp, the master secret key msk, the user directory dir and a user identity $\mathsf{id} \in \mathsf{ID}$, proceed as follows:
   (a) Sample $\mathbf{x}_{\mathsf{id}} \leftarrow \mathbb{Z}_p^\ell$. The pair $\mathsf{u}_{\mathsf{id}} = (\mathsf{id}, \mathbf{x}_{\mathsf{id}})$ is then appended to the user directory dir.
   (b) Let $\mathsf{sk}_{\mathsf{id}} \leftarrow \mathcal{MMFE}.\mathsf{KeyGen}(\mathsf{pp}, \mathsf{msk}, \mathbf{x}_{\mathsf{id}}) \in \mathcal{MMFE}.\mathcal{K}$.
   (c) Output $(\mathsf{sk}_{\mathsf{id}}, \mathbf{x}_{\mathsf{id}})$.
3. $\mathsf{Enc}(\mathsf{pp}, \mathsf{msk}, \mathsf{dir}, \mathcal{R}, m)$. Upon input the public parameter pp, the master secret key msk, the user directory dir, a set of revoked users $\mathcal{R}$ of size $\leq r$ and a plaintext messages $m \in \mathcal{M} = \{0,1\}^n$, proceed as follows:
   (a) Sample $\mathbf{v}_{\mathcal{R},1}, \ldots, \mathbf{v}_{\mathcal{R},n} \leftarrow \mathbf{X}_{\mathcal{R}}^{\perp}$ where $\mathbf{X}_{\mathcal{R}} = \{\mathbf{x}_{\mathsf{id}} \in \mathbb{Z}_p^\ell \ : \ \mathsf{id} \in \mathcal{R}\}$.
   (b) Compute $\mathbf{y}_{\mathcal{R},i} = m_i \cdot \mathbf{v}_{\mathcal{R},i}$ for $i \in [1, n]$.
   (c) Define a matrix $\mathbf{M}_{\mathcal{R}} = (\mathbf{y}_{\mathcal{R},1}, \ldots, \mathbf{y}_{\mathcal{R},n})^\top$.
   (d) Output $C_{\mathcal{R}} = \mathcal{MMFE}.\mathsf{Enc}(\mathsf{pp}, \mathsf{msk}, \mathbf{M}_{\mathcal{R}})$.
4. $\mathsf{Dec}(\mathsf{pp}, (\mathbf{x}_{\mathsf{id}}, \mathsf{sk}_{\mathsf{id}}), C_{\mathcal{R}})$. Upon input the public parameters pp, the secret key $\mathsf{sk}_{\mathsf{id}}$ for user id and a ciphertext $C_{\mathcal{R}}$ considering the revoked set $\mathcal{R}$, proceed as follows:
   (a) Compute $\mathbf{t} = \mathcal{MMFE}.\mathsf{Dec}(\mathsf{pp}, (\mathbf{x}_{\mathsf{id}}, \mathsf{sk}_{\mathsf{id}}), C_{\mathcal{R}})$.
   (b) Output $m' = (m'_1, \ldots, m'_n) \in \{0,1\}^n$ where for all $i \in [1, n]$, $m'_i = 0$ if $t_i = \mathcal{elem}^*$; else $m'_i = 1$.

Correctness. The correctness basically follows from the correctness of TR_0 above. The main difference is that, functionally, Enc of TR_1 is some-what n many copies of Enc of TR_0. Thus, Dec must concatenate all the bits to get back the message. Therefore, TR_1 is correct if Dec of TR_1 retrieves all the bits m_i correctly. Now, if $\exists i \in [1, n]$, such that Dec of TR_1 didn't compute m_i correctly, this can be extended to an attack on the correctness of Dec of TR_0. This basically ensures the correctness of TR_1.

Security We prove that TR_1 enjoys message hiding and revocation set hiding.

Theorem 6. *If $\mathcal{MMFE}$ is an* $\mathsf{IND\text{-}CPA}$ *secure matrix multiplication functional encryption scheme, then* TR_1 *is* $\mathsf{IND\text{-}CPA}$ *secure.*

Proof Sketch. The proof is very similar to the proof of Theorem 3. However, the primary difference being the ciphertext generation on a challenge $(m_0, m_1, \mathcal{R})$. In particular, $\mathcal{A}_{\mathcal{MMFE}}$ finds solution of $\mathbf{X} \cdot \mathbf{V} = \mathbf{0}$ such that $\mathbf{V}$ is a full-rank matrix in $\mathbb{Z}_p^{\ell \times n}$. Precisely, $\mathbf{V} = \begin{pmatrix} \mathbf{v}_{\mathcal{R},1} \ldots \mathbf{v}_{\mathcal{R},n} \end{pmatrix}$. Then $\mathcal{A}_{\mathcal{MMFE}}$ constructs the challenge as $\mathbf{M}_0$ and $\mathbf{M}_1$ where $\mathbf{M}_b = (\mathbf{y}_{\mathcal{R},b,1}, \ldots, \mathbf{y}_{\mathcal{R},b,n})^\top$ such that $\mathbf{y}_{\mathcal{R},b,j} = m_{b,j} \cdot \mathbf{v}_{\mathcal{R},j}$. The rest follows naturally. □

Theorem 7. *If $\mathcal{MMFE}$ is an* $\mathsf{IND\text{-}CPA}$ *secure matrix-multiplication functional encryption scheme allowing at most $(t + r - 1)$ key extraction queries, then* TR_1 *is* $\mathsf{mIND\text{-}ID\text{-}CPA}$ *secure.*

Proof Sketch. The proof is very similar to the proof of Theorem 4. The difference is again how we handle ciphertext generation. For, ciphertext query on $(m, \mathcal{R})$, $\mathcal{A}_{\mathcal{MMFE}}$ finds solution of $\mathbf{X} \cdot \mathbf{V} = \mathbf{0}$ such that $\mathbf{V}$ is a full-rank matrix in $\mathbb{Z}_p^{\ell \times n}$. Precisely, $\mathbf{V} = \begin{pmatrix} \mathbf{v}_{\mathcal{R},1} \ldots \mathbf{v}_{\mathcal{R},n} \end{pmatrix}$. Then we construct the ciphertext query to the $\mathcal{MMFE}$ challenger as $\mathbf{M}$ where $\mathbf{M} = (\mathbf{y}_{\mathcal{R},1}, \ldots, \mathbf{y}_{\mathcal{R},n})^\top$ such that $\mathbf{y}_{\mathcal{R},j} = m_j \cdot \mathbf{v}_{\mathcal{R},j}$. For the challenge query on $(m, \mathcal{R}_0, \mathcal{R}_1)$, $\mathcal{A}_{\mathcal{MMFE}}$ finds non-trivial solution of the following equations where both $\mathbf{V}_{\mathcal{R}_0}, \mathbf{V}_{\mathcal{R}_1}$ are full-rank matrices from $\mathbb{Z}_p^{\ell \times n}$.

$$\begin{pmatrix} \mathbf{Z} & -\mathbf{Z} \\ \mathbf{X}_0 & \mathbf{0} \\ \mathbf{0} & \mathbf{X}_1 \end{pmatrix} \cdot \begin{pmatrix} \mathbf{V}_{\mathcal{R}_0} \\ \mathbf{V}_{\mathcal{R}_1} \end{pmatrix} = \begin{pmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix} \tag{3}$$

where $\mathbf{V}_{\mathcal{R}_b} = \begin{pmatrix} \mathbf{v}_{\mathcal{R}_b,1} \ldots \mathbf{v}_{\mathcal{R}_b,n} \end{pmatrix}$ for $b \in \{0,1\}$. Then $\mathcal{A}_{\mathcal{MMFE}}$ constructs the challenge as $\mathbf{M}_0$ and $\mathbf{M}_1$ where $\mathbf{M}_b = (\mathbf{y}_{\mathcal{R}_b,1}, \ldots, \mathbf{y}_{\mathcal{R}_b,n})^\top$ such that $\mathbf{y}_{\mathcal{R}_b,j} = m_j \cdot \mathbf{v}_{\mathcal{R}_b,j}$. The rest of the argument follows naturally. □

Construction TR_0 *and* TR_1. Note that, available IPFE schemes [4,5] suffice to construct of TR_0 and TR_1. In particular, withholding the public keys of available IPFE schemes, one can get symmetric-key IPFE schemes and use them to construct TR_0. Furthermore, TR_1 can be constructed from running n independent instances of any symmetric-key IPFE scheme. We in fact use this technique to construct TR_0 and TR_1 in the lattice-based settings withholding the public key of Agrawal *et al.*'s IPFE [4]. In the group-based settings, however, we can achieve more efficient constructions than naively hiding the public key of the public-key IPFE. In Section 5, we propose new constructions of symmetric-key IPFE and symmetric-key MMFE in the prime-order groups.

4 Cryptanalysis of the Wang *et al* IPFE Construction

As we mention above, the schemes from Section 3 can be instantiated with the LWE-based $\mathcal{IPFE}$ scheme from [4]. Note that the latter does not enjoy IND-CPA security, but it was showed to enjoy a weaker security property that still suffices for the trace-and-revoke scheme from [4]. That weaker security property restricts the number of key requests to be significantly smaller than the dimension of the vector space, and imposes that the vectors of the key queries are uniformly sampled. This relaxation of IND-CPA security also suffices for our adaptation from Section 3.

$\mathcal{IPFE}$ scheme from [29], note that the LWE-based $\mathcal{IPFE}$ scheme from [29] is also claimed to enjoy a security property that is stronger than IND-CPA security (which the authors leverage to obtain a decentralized Attribute-Based Encryption scheme). In fact, as we will show below, this scheme can be broken for the parameters suggested in [29]. Before showing an attack, we first recall some definitions.

Lattices. Given n linear independent vectors $\mathbf{b}_1, \ldots, \mathbf{b}_n \in \mathbb{R}^m$, the lattice generated by them is defined as

$$L(\mathbf{B}) := \{\mathbf{Bz} = \sum_{i\in[1,n]} z_i \mathbf{b}_i : \mathbf{z} \in \mathbb{Z}^n\}.$$

The rank of this lattice is n and its dimension is m.

We define the determinant of L as $\det(L) := \sqrt{\det(\mathbf{B}^t\mathbf{B})}$. For a rank-$n$ matrix $\mathbf{B} \in \mathbb{R}^{m\times n}$, there exist orthogonal matrices $\mathbf{U}, \mathbf{V}$ and a diagonal matrix $\mathbf{\Sigma} = \mathrm{Diag}(\sigma_1, \ldots, \sigma_n) \in \mathbb{R}^{m\times n}$ such that $\mathbf{B} = \mathbf{U\Sigma V}^t$ and $\sigma_1 \geq \cdots \geq \sigma_n > 0$. From this decomposition, we see that $\det(L(\mathbf{B})) = \prod_{i\in[1,n]} \|\sigma_{\mathbf{i}}\|$.

For $i \in [1, n]$, the i-th successive minimum $\lambda_i(L)$ is defined as

$$\lambda_i(L) := \inf\{r : \dim(\mathrm{Span}(L \cap \mathcal{B}(r))) \geq i\},$$

where $\mathcal{B}(r)$ denotes the closed zero-centered Euclidean ball of radius r.

Definition 4. *Let $m > n \geq 1$ be integers and $q \geq 2$ be prime. Let $\mathbf{X} \in \mathbb{Z}^{m\times n}$.*
*The **orthogonal lattice** $\Lambda^{\perp}(\mathbf{X})$ is the integral lattice whose vectors are orthogonal to the rows of $\mathbf{X}$, i.e.,*

$$\Lambda^{\perp}(\mathbf{X}) := \{\mathbf{u} \in \mathbb{Z}^m : \mathbf{X}^t\mathbf{u} = \mathbf{0}\}.$$

We note that if $\mathbf{X}$ has rank n (over the integers), then $\Lambda^{\perp}(\mathbf{X})$ has rank $(m - n)$.

Definition 5. *The bounded distance decoding problem BDD_γ is as follows: given a basis $\mathbf{B}$ of an n-rank lattice L, $\mathbf{t} \in \mathbb{R}^n$, and real $d \leq \frac{\lambda_1}{2}$ such that $dist(\mathbf{t}, L) \leq d$, find the unique $\mathbf{v} \in L$ closest to $\mathbf{t}$. Note that this is equivalent to finding $\mathbf{e} \in \mathbf{t} + L$ such that $\|\mathbf{e}\| \leq d$.*

We now describe here a simplified version of the security property that this scheme aims to achieve, and the corresponding simplified version of the scheme (this corresponds to setting $k = 1$ in the definition from [29]; our attack readily extends to $k \geq 1$). In the challenge phase, the adversary sends to the challenger descriptions of two distributions D_0 and D_1 over plaintext vectors. The challenger chooses $\beta \leftarrow \{0, 1\}$ and samples $\mathbf{y} \leftarrow D_\beta$; it encrypts it under the public key pk and the resulting ciphertext $\mathsf{Enc}_{\mathsf{pk}}(\mathbf{y})$ is given to the adversary. The adversary can adaptively make key queries $\mathbf{x}$, before or after the challenge phase. The security property, called adaptive security for chosen message distributions, requires that the adversary cannot guess β correctly, as long as the distributions D_0 and D_1 remain indistinguishable given the replies to the key queries.

We review their construction based on LWE.

- $\mathcal{IPFE}.\mathsf{Setup}(1^n, 1^\ell, p)$. Set integers $m, q = p^e$ for some integer e, and reals $\alpha, \alpha' \in (0, 1)$. Sample $\mathbf{A} \leftarrow \mathbb{Z}_q^{m\times n}$, $\mathbf{Z} \leftarrow \{0, \ldots, p-1\}^{\ell\times m}$,[9] compute $\mathbf{T} = \mathbf{ZA} \in \mathbb{Z}_q^{\ell\times n}$, define
$$\mathsf{msk} := \mathbf{Z} \text{ and } \mathsf{pk} := (\mathbf{A}, \mathbf{T}).$$
- $\mathcal{IPFE}.\mathsf{KeyGen}(\mathsf{msk}, \mathbf{x})$. Given $\mathbf{x} \in \mathbb{Z}_p^\ell$, set $\mathbf{z_x} = \mathbf{x}^t\mathbf{Z} \in \mathbb{Z}^m$ (interpreting each coordinate of $\mathbf{x}$ as an integer in $\{0, \ldots, p-1\}$), and output $\mathsf{sk}_\mathbf{x} = \mathbf{z_x}$.
- $\mathcal{IPFE}.\mathsf{Enc}(\mathsf{pk}, \mathbf{y})$. To encrypt a vector $\mathbf{y} \in \mathbb{Z}_p^\ell$, sample $\mathbf{s} \leftarrow \mathbb{Z}_q^n$, $\mathbf{e}_0 \leftarrow D_{\mathbb{Z}^m,\alpha q}$, $\mathbf{e}_1 \leftarrow D_{\mathbb{Z}^\ell,\alpha' q}$ and compute
$$\mathbf{c}_0 = \mathbf{As} + \mathbf{e}_0 \in \mathbb{Z}_q^m, \qquad \mathbf{c}_1 = \mathbf{Ts} + \mathbf{e}_1 + p^{e-1} \cdot \mathbf{y} \in \mathbb{Z}_q^\ell.$$
Then, return the ciphertext $C = (\mathbf{c}_0, \mathbf{c}_1)$.
- $\mathcal{IPFE}.\mathsf{Dec}(\mathsf{sk}, C)$. Given $C = (\mathbf{c}_0, \mathbf{c}_1)$ and secret key $\mathsf{sk}_\mathbf{x} = \mathbf{z_x}$, compute $\mu' = \langle \mathbf{x}, \mathbf{c}_1\rangle - \langle \mathbf{z_x}, \mathbf{c}_0\rangle \bmod q$, and output the value $\mu \in \mathbb{Z}_p$ that minimize $|\mu' - p^{e-1}\mu|$.

In [29], the dimensions n is proportional to the security parameter λ, the parameters $\ell, m, p, q, 1/\alpha, 1/\alpha'$ are polynomial in n, and e is a constant. In [29, Theorem 3.5], the authors state that under the LWE assumption, the above functional encryption for inner products is adaptively secure for chosen message distributions, assuming that the secret key queries corresponding are linearly independent.
Below, we describe a cryptanalysis of the scheme above with the specified parameters. We then explain why this attack does not apply to the schemes from [5] and [4].

We show that even for with challenge vectors rather than distributions, key queries allow to recover the master secret key msk. Concretely, we can recover $\mathbf{Z}$ from $\mathbf{X}^t$ and $\mathbf{X}^t\mathbf{Z}$, where $\mathbf{Z} \leftarrow \{0, \ldots, p-1\}^{\ell\times m}$ and $\mathbf{X} \in \{0, \ldots, p-1\}^{\ell\times(\ell-1)}$ is chosen by the adversary. We let our adversary sample $\mathbf{X} \leftarrow \{0, \ldots, p-1\}^{\ell\times(\ell-1)}$ (recall that the multiplication $\mathbf{X}^t\mathbf{Z}$ is over $\mathbb{Z}$). The fact that $\mathbf{X}$ has only $\ell - 1$ columns means that we can find distinct challenge plaintexts (which are elements of $\mathbb{Z}_p^\ell$) so that the columns of $\mathbf{X}$ are valid key queries.

It suffices to show how the adversary can recover the first column $\mathbf{z}$ of $\mathbf{Z}$ from $\mathbf{X}^t\mathbf{z}$, as it can proceed similarly for all columns of $\mathbf{Z}$. Given $\mathbf{t} = \mathbf{X}^t\mathbf{z}$ and $\mathbf{X}$, we know that $\mathbf{z}$ belongs to a coset of the lattice $\Lambda^\perp(\mathbf{X})$ defined by $\mathbf{t}$.

Let us now study the lattice $\Lambda^\perp(\mathbf{X})$. As $\mathbf{X} \leftarrow \{0, \ldots, p-1\}^{\ell\times(\ell-1)}$, its columns are expected to be linearly independent with overwhelming probability and $\det(\mathbf{X}\mathbb{Z}^{\ell-1})$ is expected to grow as $p^{\Omega(\ell)}$. These properties would be easier to prove if the entries of $\mathbf{X}$ were Gaussian with standard deviation p, but it can be experimentally checked that this behavior also holds for this distribution. We also expect the lattice $\mathbf{X}\mathbb{Z}^{\ell-1}$ to be primitive, i.e., that $\mathbf{X}^t\mathbb{Z}^\ell = \mathbb{Z}^{\ell-1}$. By [25, p. 30], we hence have that $\det(\Lambda^\perp(\mathbf{X})) = \det(\mathbf{X}\mathbb{Z}^{\ell-1})$. As $\mathbf{X}$ is full column-rank, we known that $\dim(\Lambda^\perp(\mathbf{X})) = 1$, and hence we expect that $\lambda_1(\Lambda^\perp(\mathbf{X})) = p^{\Omega(\ell)}$. Finally, note that the orthogonal lattice can be efficiently computed, by using a Hermite Normal Form algorithm.

Now, recall that we want to recover $\mathbf{z}$ from a known coset of $\Lambda^\perp(\mathbf{X})$. As $\|\mathbf{z}\| \leq \sqrt{\ell}p$, by the above analysis of $\Lambda^\perp(\mathbf{X})$, we expect to have
$$\|\mathbf{z}\| < \lambda_1(\Lambda^\perp(\mathbf{X}))/2.$$
This implies that $\mathbf{z}$ is uniquely determined from the coset. Moreover, this is a Bounded Distance Decoding problem instance in a lattice of dimension 1, which can be solved efficiently. Concretely, if $\Lambda^\perp(\mathbf{X}) = \mathbf{b}\mathbb{Z}$ and we are given $\mathbf{b}$ and $k\mathbf{b} + \mathbf{z}$, we can recover $k = \lfloor\langle k\mathbf{b} + \mathbf{z}, \mathbf{b}\rangle/\|\mathbf{b}\|^2\rceil$ and hence $\mathbf{z}$.

[9] In [29], the notation $\mathbb{Z}_p^{\ell\times m}$ is used instead of $\{0, \ldots, p-1\}^{\ell\times m}$. We stress that it should indeed be interpreted as $\{0, 1, \ldots, p-1\}^{\ell\times m}$. In particular, the operation $\mathbf{x}^t\mathbf{Z}$ in the $\mathcal{IPFE}.\mathsf{KeyGen}$ algorithm is over $\mathbb{Z}$ and not modulo p, as otherwise decryption correctness would not hold.

Remarks. The above discussion shows that the IPFE scheme of [29] is not secure with the specified parameters. We explain here why the above attack does not work for the [5] and [4] schemes. First, in the mod-p scheme from [5, Section 4.1], the authors take $\mathbf{z}$ from a discrete Gaussian distribution with a large standard deviation. With the parameters specified in [5], we then have that $\|\mathbf{z}\|$ is significantly larger than $\lambda_1(\Lambda^\perp(\mathbf{X}))$. This implies that there is a large amount of entropy left in $\mathbf{z}$ given $\mathbf{t} = \mathbf{X}^t\mathbf{z}$. Also, this attack does not work for the [5] scheme over $\mathbb{Z}$, because in that case, the matrix $\mathbf{X}$ and hence the lattice $\Lambda^\perp(\mathbf{X})$ are not random at all. Indeed, the kernel lattice is forced to be $(\mathbf{y}_0 - \mathbf{y}_1)\mathbb{Z}^\ell$, where $\mathbf{y}_0$ and $\mathbf{y}_1$ are the challenge vectors. By assumption on the scheme, these challenge vectors are small. Put differently, in that setting, if we first do $(\ell - 1)$ random queries, there does not exist $\mathbf{y}_0 - \mathbf{y}_1 \neq \mathbf{0}$ short anymore that allows us to create a non-trivial challenge phase. Finally, the attack does not work for the [4] scheme variant, because in that case, the matrix $\mathbf{X}$ has much fewer columns than rows. This increases the dimension of $\Lambda^\perp(\mathbf{X})$ enough to make $\lambda_1(\Lambda^\perp(\mathbf{X}))$ much smaller, and in particular smaller than $\|\mathbf{z}\|$.

5 Linear Functional Encryptions in Prime-Order Groups

As outlined in Section 3, our trace-and-revoke schemes are instantiated using different linear functional encryption schemes. In this section, we give a construction of $\mathcal{MMFE}$ in the symmetric-key setting. For $n = 1$, the $\mathcal{MMFE}$ construction reduces to $\mathcal{IPFE}$. Due to space restraint, we omit the description of $\mathcal{IPFE}$ and present the $\mathcal{MMFE}$ below. The point of interest being, the Dec in our $\mathcal{MMFE}$ (and in our $\mathcal{IPFE}$) does not compute the discrete log.

5.1 $\mathcal{MMFE}$ from $\mathcal{D}_k$-matDH

We propose a construction of matrix multiplication functional encryption ($\mathcal{MMFE}$) from $\mathcal{D}_k$-matDH. Since, the complete matrix $\mathbf{M} = (\mathbf{y}_1, \ldots, \mathbf{y}_n)^\top$ is available to Enc at once, our construction can reuse the randomness for all $\mathbf{y}_i \in \mathbb{Z}_p^\ell$. This also allows the proof to be tightly reduced to $\mathcal{D}_k$-matDH. For this, we require n matrices $\mathbf{W}_1, \ldots, \mathbf{W}_n$ unlike $\mathcal{IPFE}$ from $\mathcal{D}_k$-matDH that required only one. We emphasize that, similar to $\mathcal{IPFE}$ above, $\mathcal{MMFE}$ also does not need to evaluate discrete logarithm algorithm.

- $\mathsf{Setup}(1^\lambda, 1^\ell, 1^n, p)$. Run $(g, \mathbb{G}) \leftarrow \mathcal{G}_{gen}(1^\lambda, p)$. Sample $\mathbf{A} \leftarrow \mathcal{D}_k$ and $\mathbf{W}_1, \ldots,\ \mathbf{W}_n \leftarrow \mathbb{Z}_p^{\ell \times k\ell n}$. Define $\mathsf{msk} = (\mathbf{W}_1, \ldots, \mathbf{W}_n)$ and $\mathsf{pp} = ([1])$.
- $\mathsf{KeyGen}(\mathsf{pp}, \mathsf{msk}, \mathbf{x} \in \mathbb{Z}_p^\ell)$. Set $\mathsf{sk}_\mathbf{x} \leftarrow (\mathbf{x}^\top \mathbf{W}_1, \ldots, \mathbf{x}^\top \mathbf{W}_n, \mathbf{x})$.
- $\mathsf{Enc}(\mathsf{pp}, \mathsf{msk}, \mathbf{M} = (\mathbf{y}_1, \ldots, \mathbf{y}_n)^\top \in \mathbb{Z}_p^{n \times \ell})$ proceeds as follows to encrypt the given vectors $\mathbf{y}_1, \ldots, \mathbf{y}_n \in \mathbb{Z}_p^\ell$. Sample $\mathbf{s} \leftarrow \mathbb{Z}_p^{k\ell n}$. Set $\mathsf{ct}_\mathbf{M} \leftarrow ([\mathbf{s}], [\mathbf{y}_1 + \mathbf{W}_1\mathbf{s}], \ldots, [\mathbf{y}_n + \mathbf{W}_n\mathbf{s}])$.
- $\mathsf{Dec}(\mathsf{pp}, \mathsf{sk}_\mathbf{x}, \mathsf{ct}_\mathbf{M})$. Parse $\mathsf{ct}_\mathbf{M} = ([\mathbf{c}_0], [\mathbf{c}_1], \ldots, [\mathbf{c}_n])$. Return $\mathbf{t} = (t_1, \ldots, t_n)$ where $t_i = \left[\mathbf{x}^\top \mathbf{c}_i\right] \cdot [\mathsf{sk}_\mathbf{x} \cdot \mathbf{c}_0]^{-1}$.

The correctness is easy to verify.

We show a rough comparison of our scheme with [28] if their scheme was used for symmetric key settings directly. Section 5.1 shows that the symmetric key variant resulted from hiding the public key of [28] has bigger public parameters and bigger ciphertext i.e. contain more group elements than our scheme. On the other hand, our secret key contains more elements from $\mathbb{Z}_p$. Both the schemes are proven secure under same assumption $\mathcal{D}_k$-matDH with constant degradation. We further compare the result for the SXDH based instances which shows that their scheme outputs ciphertext that is 1.5 times bigger than us.

Security. Next, we argue the security of $\mathcal{MMFE}$ in the $\mathsf{IND\text{-}CPA}$ security model. Our construction is basically a modification of [28] for symmetric-key settings. This improves upon the performance in terms of ciphertext size and removes the usage of public parameters completely. Note that, this modification required us to argue the security proof in a different manner. Although the overall proof strategy stayed more-or-less the same, our proof presents a completely new proof for an essential lemma. We state the security theorem next and give the proof.

Table 1. Comparison of naive application of [28] with our construction in symmetric-key settings. The sizes of pp and ct are in number of group elements, whereas those of the sk column are in number of elements of $\mathbb{Z}_p$.

	$\lvert\mathsf{pp}\rvert$	$\lvert\mathsf{sk}\rvert$	$\lvert\mathsf{ct}\rvert$	Degradation	Assumption
[28]	$k^3(k+1)\ell^2 + k^2\ell^2$	$(k+1)k\ell$	$n((k+1)k\ell + \ell)$	4	$\mathcal{D}_k$-matDH
	$2\ell^2 + \ell^2$	2ℓ	$3n\ell$	4	SXDH
This work	1	$k\ell n^2$	$k\ell n + \ell n$	$k+1$	$\mathcal{D}_k$-matDH
	1	$n^2\ell$	$2n\ell$	2	SXDH

Theorem 8. *For any adversary $\mathcal{A}$ of the construction $\mathcal{MMFE}$ in the IND-CPA security model that makes at most q_{sk} secret key queries (for $q_{\mathsf{sk}} < \ell$) and q_{ct} challenge ciphertext queries in an interleaved manner, there exists adversary $\mathcal{C}$ such that,*

$$\mathrm{Adv}^{\mathsf{IND\text{-}CPA}}_{\mathcal{MMFE},\mathcal{A}}(\lambda) \leq (k+1) \cdot \mathrm{Adv}^{\mathcal{D}_k\text{-}\mathsf{matDH}}_{\mathcal{C}}(\lambda).$$

The proof is done by defining a hybrid argument of a sequence of games that begins with the real protocol (called $\mathbf{Game}_0$) and ends with a so-called final game (called $\mathbf{Game}_3$) where the adversary has no advantage at all. During the sequence, we use X_i to denote the event that the adversary has won $\mathbf{Game}_i$.

- $\mathbf{Game}_0$. This is the real game. All secret key queries on $\mathbf{x} \in \mathbb{Z}_p^{\ell}$ are responded as the real game. For all j^{th} (such that $j \in [1, q_{\mathsf{ct}}]$) ciphertext query on two matrices $\mathbf{M}_j^{(0)}, \mathbf{M}_j^{(1)} \in \mathbb{Z}_p^{n\times\ell}$, the challenge ciphertext returned is $\mathsf{ct}^{(\beta)} \leftarrow \mathcal{MMFE}.\mathsf{Enc}(\mathsf{pp}, \mathsf{msk}, \mathbf{M}_j^{(\beta)})$ for $\beta \leftarrow \{0,1\}$. More precisely, the j^{th} ciphertext query is responded as,
$$\mathbf{s}_j \leftarrow \mathbb{Z}_p^{k\ell n}, \quad [\mathbf{c}_{j,0}] = [\mathbf{s}_j], \quad [\mathbf{c}_{j,i}] = \left[\mathbf{y}_{j,i}^{(\beta)} + \mathbf{W}_i\mathbf{s}_j\right]$$
for $j \in [1, q_{\mathsf{ct}}]$. At the end, $\mathcal{A}$ outputs $\beta' \in \{0,1\}$ and wins the game if $\beta = \beta'$.
- $\mathbf{Game}_1$. The response of the challenge queries are defined as following. For j^{th} ciphertext query is made on $(\mathbf{M}_j^{(0)}, \mathbf{M}_j^{(1)})$ where $\mathbf{M}_j^{(b)} = (\mathbf{y}_{j,1}^{(b)}, \ldots, \mathbf{y}_{j,n}^{(b)})^\top$ for $b \in \{0,1\}$,
$$\mathbf{s}_j \leftarrow \mathbb{Z}_p^{k\ell n}, \quad [\mathbf{c}_{j,0}] = [\mathbf{s}_j], \quad [\mathbf{c}_{j,i}] = \left[\mathbf{y}_{j,i}^{(\beta)} + \mathbf{W}_i\mathbf{s}_j + \sum_{\iota\in[1,\phi_i(j)]} \mathbf{u}^\top\mathbf{v}_{j,i,\iota} \cdot \mathbf{z}_{\psi_i(\iota),i}\right]$$
where $\mathbf{u} \leftarrow \mathbb{Z}_p^k$, $\mathbf{z}_{j,i} = \mathbf{y}_{j,i}^{(1)} - \mathbf{y}_{j,i}^{(0)}$, $\phi_i(j) = \mathsf{Rank}(\mathbf{z}_{1,i}||\ldots||\mathbf{z}_{j,i})$, $\psi_i(j) = \mathsf{min}(\phi_i^{-1}(j))$, and $\mathbf{v}_{j,i,1}, \ldots, \mathbf{v}_{j,i,\ell} \leftarrow \mathbb{Z}_p^k$ where $i \in [1,n]$ and $j \in [1, q_{\mathsf{ct}}]$. In Lemma 1 we show that $|\Pr[X_1] - \Pr[X_0]| \leq \mathrm{Adv}^{\mathcal{D}_{2k,k}\text{-}\mathsf{matDH}}(\lambda)$.
- $\mathbf{Game}_2$. The response of the challenge queries are defined as following. For j^{th} ciphertext query is made on $(\mathbf{M}_j^{(0)}, \mathbf{M}_j^{(1)})$ where $\mathbf{M}_j^{(b)} = (\mathbf{y}_{j,1}^{(b)}, \ldots, \mathbf{y}_{j,n}^{(b)})^\top$ for $b \in \{0,1\}$,
$$\mathbf{s}_j \leftarrow \mathbb{Z}_p^{k\ell n}, \quad [\mathbf{c}_{j,0}] = [\mathbf{s}_j], \quad [\mathbf{c}_{j,i}] = \left[\mathbf{y}_{j,i}^{(\beta)} + \mathbf{W}_i\mathbf{s}_j + \sum_{\iota\in[1,\phi_i(j)]} v_{j,i,\iota} \cdot \mathbf{z}_{\psi_i(\iota),i}\right]$$
where $\mathbf{z}_{j,i} = \mathbf{y}_{j,i}^{(1)} - \mathbf{y}_{j,i}^{(0)}$, $\phi_i(j) = \mathsf{Rank}(\mathbf{z}_{1,i}||\ldots||\mathbf{z}_{j,i})$, $\psi_i(j) = \mathsf{min}(\phi_i^{-1}(j))$, and $v_{j,i,1}, \ldots, v_{j,i,\ell} \leftarrow \mathbb{Z}_p$ where $i \in [1,n]$ and $j \in [1, q_{\mathsf{ct}}]$. In Lemma 2 we show that $|\Pr[X_2] - \Pr[X_1]| \leq \mathrm{Adv}^{\mathcal{D}_k\text{-}\mathsf{matDH}'}(\lambda)$.
- $\mathbf{Game}_3$. Finally, we show that, the injected entropy is sufficient to hide β in the returned ciphertexts completely. This is because, for any $j \in [1, q_{\mathsf{ct}}]$ and $i \in [1,n]$, $\sum_{\iota\in[1,\phi_i(j)]} v_{j,i,\iota}\mathbf{z}_{\psi_i(\iota),i}$ is basically a random vector in the span of $\{\mathbf{z}_{\psi_i(\iota),i}\}_{\iota\in[1,\phi_i(j)]}$. Furthermore, by the definition of ϕ and ψ, $\{\mathbf{z}_{\psi_i(\iota),i}\}_{\iota\in[1,\phi_i(j)]}$ are

the basis and therefore each $\mathbf{z}_{j,i} \in \text{Span}(\{\mathbf{z}_{\psi_i(\iota),i}\}_{\iota\in[1,\phi_i(j)]})$. Then,

$$\begin{aligned}\mathbf{y}_{j,i}^{(\beta)} + \mathbf{W}_i\mathbf{s}_j + \sum_{\iota\in[1,\phi_i(j)]} v_{j,i,\iota}\mathbf{z}_{\psi_i(\iota),i} &= \beta\mathbf{z}_{j,i} + \mathbf{y}_{j,i}^{(0)} + \mathbf{W}_i\mathbf{s}_j + \sum_{\iota\in[1,\phi_i(j)]} v_{j,i,\iota}\mathbf{z}_{\psi_i(\iota),i} \\ &\equiv \mathbf{y}_{j,i}^{(0)} + \mathbf{W}_i\mathbf{s}_j + \sum_{\iota\in[1,\phi_i(j)]} v_{j,i,\iota}\mathbf{z}_{\psi_i(\iota),i}\end{aligned}$$

As the ciphertext distribution stays the same as in $\textbf{Game}_2$, $\Pr[X_3] = \Pr[X_2]$.
Now, notice that, $[\mathbf{c}_{j,i}] = \left[\mathbf{y}_{j,i}^{(\beta)} + \mathbf{W}_i\mathbf{s}_j + \sum_{\iota\in[1,\phi_i(j)]} v_{j,i,\iota}\mathbf{z}_{\psi_i(\iota),i}\right]$ hides β completely for all $j \in [1, q_{\mathsf{ct}}]$ and $i \in [1, n]$. Thus $\Pr[X_3] = 1/2$.
To summarise,

$$\begin{aligned}\text{Adv}_{\mathcal{MMFE},\mathcal{A}}^{\mathsf{IND\text{-}CPA}}(\lambda) &\leq |1/2 - \Pr[X_0]| \\ &= |\Pr[X_3] - \Pr[X_0]| \\ &\leq |\Pr[X_3] - \Pr[X_2]| + |\Pr[X_2] - \Pr[X_1]| + |\Pr[X_1] - \Pr[X_0]| \\ &\leq 0 + \text{Adv}^{\mathcal{D}_{2k,k}\text{-}\mathsf{matDH}}(\lambda) + \text{Adv}^{\mathcal{D}_k\text{-}\mathsf{matDH}'}(\lambda) \qquad . \\ &\leq k \cdot \text{Adv}^{\mathcal{D}_k\text{-}\mathsf{matDH}}(\lambda) + \text{Adv}^{\mathcal{D}_k\text{-}\mathsf{matDH}}(\lambda) \\ &\qquad\qquad\qquad\qquad\qquad \text{(due to } \textit{Section } 2.5.3) \\ &\leq (k+1) \cdot \text{Adv}^{\mathcal{D}_k\text{-}\mathsf{matDH}}(\lambda)\end{aligned}$$

Lemma 1. *For any efficient adversary $\mathcal{A}$ that makes at most q_{sk} secret key queries and at most q_{ct} ciphertext queries, there exists a algorithm $\mathcal{B}$ such that* $|\Pr[X_1] - \Pr[X_0]| \leq \text{Adv}_{\mathcal{B}}^{\mathcal{D}_{2k,k}\text{-}\mathsf{matDH}}(\lambda)$.

Proof. To simulate the game, we use a $\mathcal{D}_{2k,k}$-matDH (as described in Section 2.5.3) problem instance $([\mathbf{A}], [\mathbf{t}])$ where $\mathbf{t} = \begin{pmatrix}\overline{\mathbf{A}}\mathbf{w} \\ \underline{\mathbf{A}}\mathbf{w} + \boldsymbol{\delta}\end{pmatrix}$ for $\mathbf{w} \in \mathbb{Z}_p^k$ where $\boldsymbol{\delta} = \mathbf{0}$ or chosen uniformly random vector from $\mathbb{Z}_p^k$. In fact, we use random self-reducibility property to define $q_{\mathsf{ct}}n\ell$ many problem instances $([\mathbf{A}], [\mathbf{t}_{j,i,\iota}])$ for $j \in [1, q_{\mathsf{ct}}]$, $i \in [1, n]$ and $\iota \in [1, \ell]$. We use such problem instances to sample the $\mathbf{W}_1, \ldots, \mathbf{W}_n$. First, we set

$$\mathbf{s}_j = (\overline{\mathbf{t}}_{j,1,1}, \ldots, \overline{\mathbf{t}}_{j,1,\ell}, \ldots, \overline{\mathbf{t}}_{j,n,1}, \ldots, \overline{\mathbf{t}}_{j,n,\ell})^\top.$$

For all $i \in [1, n]$, we then sample

$$\mathbf{W}_i = \widetilde{\mathbf{W}}_i + \sum_{\iota\in[1,\phi_i(q)]} \mathbf{z}_{\psi_i(\iota),i}\mathbf{u}^\top\mathbf{T}\left[\mathbf{0}_{k\times k(\ell(i-1)+(\iota-1))}||\mathbf{I}_k||\mathbf{0}_{k\times k((\ell-\iota)+\ell(n-i))}\right]$$

where $\widetilde{\mathbf{W}}_i \leftarrow \mathbb{Z}_p^{\ell\times k\ell n}$, $\mathbf{u} \leftarrow \mathbb{Z}_p^k$, $\mathbf{T} = \underline{\mathbf{A}} \cdot (\overline{\mathbf{A}})^{-1}$ and $\mathbf{z}_{j,i} = \mathbf{y}_{j,i}^{(1)} - \mathbf{y}_{j,i}^{(0)}$ where $i \in [1, n]$ and $j \in [1, q_{\mathsf{ct}}]$ for j^{th} ciphertext query is made on $(\mathbf{M}_j^{(0)}, \mathbf{M}_j^{(1)})$ where $\mathbf{M}_j^b = (\mathbf{y}_{j,1}^b, \ldots, \mathbf{y}_{j,n}^b)^\top$ for $b \in \{0, 1\}$.

For all $j \in [1, q_{\mathsf{sk}}]$, the j^{th} secret key query on $\mathbf{x}_j$, we respond with $\mathsf{sk} = (\mathbf{x}_j^\top\widetilde{\mathbf{W}}_1, \ldots, \mathbf{x}_j^\top\widetilde{\mathbf{W}}_i)$. Given j^{th} ciphertext query on $(\mathbf{M}_j^{(0)}, \mathbf{M}_j^{(1)})$ for $j \in [1, q_{\mathsf{ct}}]$, we respond with $([\mathbf{c}_{j,0}], [\mathbf{c}_{j,1}], \ldots, [\mathbf{c}_{j,n}])$ where $\mathbf{c}_{j,0} = \mathbf{s}_j$ and for all $i \in [1, n]$, $\mathbf{c}_{j,i} = \mathbf{y}_{j,i}^b + \widetilde{\mathbf{W}}_i\mathbf{s}_j + \sum_{\iota\in[1,\phi_i(j)]} \mathbf{z}_{\psi_i(\iota),i}\mathbf{u}^\top\underline{\mathbf{t}}_{j,i,\iota}$ where $\phi_i(j) = \mathsf{Rank}(\mathbf{z}_{1,i}||\ldots||\mathbf{z}_{j,i})$ and $\psi_i(j) = \mathsf{min}(\phi_i^{-1}(j))$.

Firstly, observe that, the ciphertext generation uses $\mathbf{z}_{1,1}, \ldots, \mathbf{z}_{1,n}, \ldots, \mathbf{z}_{j,1}, \ldots, \mathbf{z}_{j,n}$ where $\mathbf{z}_{\iota,i} = \mathbf{y}_{\iota,i}^{(1)} - \mathbf{y}_{\iota,i}^{(0)}$ where $i \in [1, n]$ and $\iota \in [1, j]$ i.e. each j^{th} ciphertext is defined using already queried matrices $(\mathbf{M}_1^{(0)}, \mathbf{M}_1^{(1)})$, $\ldots, (\mathbf{M}_j^{(0)}, \mathbf{M}_j^{(1)})$. Moreover, $\mathbf{x}^\top\mathbf{W}_i = \mathbf{x}^\top\widetilde{\mathbf{W}}_i$ for all $i \in [1, n]$ as $\mathbf{x}^\top\mathbf{z}_{j,i} = 0$ for all $j \in [1, q]$ and therefore the secret keys are simulated properly. We now show that the ciphertexts are also simulated properly.

$$
\begin{aligned}
\mathbf{W}_i\mathbf{s}_j &= \widetilde{\mathbf{W}}_i\mathbf{s}_j + \sum_{\iota\in[1,\phi_i(q)]} \mathbf{z}_{\psi_i(\iota),i}\mathbf{u}^\top\mathbf{T}\left[\mathbf{0}_{k\times k(\ell(i-1)+(\iota-1))}||\mathbf{I}_k||\mathbf{0}_{k\times k((\ell-\iota)+\ell(n-i))}\right]\cdot \begin{bmatrix}\overline{\mathbf{t}}_{j,1,1}\\ \vdots \\ \overline{\mathbf{t}}_{j,1,\ell}\\ \vdots \\ \overline{\mathbf{t}}_{j,n,1}\\ \vdots \\ \overline{\mathbf{t}}_{j,n,\ell}\end{bmatrix}\\
&= \widetilde{\mathbf{W}}_i\mathbf{s}_j + \sum_{\iota\in[1,\phi_i(j)]} \mathbf{z}_{\psi_i(\iota),i}\mathbf{u}^\top\mathbf{T}\overline{\mathbf{t}}_{j,i,\iota}\\
&= \widetilde{\mathbf{W}}_i\mathbf{s}_j + \sum_{\iota\in[1,\phi_i(j)]} \mathbf{z}_{\psi_i(\iota),i}\mathbf{u}^\top\cdot(\underline{\mathbf{A}}(\overline{\mathbf{A}})^{-1})\cdot(\overline{\mathbf{A}}\mathbf{w}_{j,i,\iota})\\
&= \widetilde{\mathbf{W}}_i\mathbf{s}_j + \sum_{\iota\in[1,\phi_i(j)]} \mathbf{z}_{\psi_i(\iota),i}\mathbf{u}^\top\cdot(\underline{\mathbf{A}}\mathbf{w}_{j,i,\iota})\\
&\approx \widetilde{\mathbf{W}}_i\mathbf{s}_j + \sum_{\iota\in[1,\phi_i(j)]} \mathbf{z}_{\psi_i(\iota),i}\mathbf{u}^\top\underline{\mathbf{t}}_{j,i,\iota}
\end{aligned}
$$

Now, it is clear that if $\mathbf{t}\in\mathrm{Span}(\mathbf{A})$, the simulation is identical to $\mathbf{Game}_0$ as $\underline{\mathbf{t}}_{j,i,\iota}=\underline{\mathbf{A}}\mathbf{w}_{j,i,\iota}$ for $j\in[1,q_{\mathsf{ct}}]$, $i\in[1,n]$ and $\iota\in[1,\ell]$. Otherwise, the simulation is identical to $\mathbf{Game}_1$. □

Lemma 2. *For any efficient adversary $\mathcal{A}$ that makes at most q_{sk} secret key queries and at most q_{ct} ciphertext queries, there exists a* ppt *algorithm $\mathcal{B}$ such that* $|\Pr[X_2]-\Pr[X_1]|\le \mathrm{Adv}_{\mathcal{B}}^{\mathcal{D}_k\text{-}\mathsf{matDH}'}(\lambda)$.

Proof. Here, $\mathcal{B}$ gets an $\mathcal{D}_k$-matDH' problem instance $([\mathbf{T}],[\mathbf{v}^{(\delta)}])$ for $\delta\leftarrow\{0,1\}$ where $\mathbf{T}\in\mathbb{Z}_p^{k\times m}$, $\mathbf{v}^{(0)}=\mathbf{a}^\top\mathbf{T}$ and $\mathbf{v}^{(1)}\leftarrow\mathbb{Z}_p^{1\times m}$ (as described in Section 2.5.1) where $\mathbf{a}\leftarrow\mathbb{Z}_p^k$. Note that here we set $m=q_{\mathsf{ct}}n\ell$ and implicitly set $\mathbf{u}$ as $\mathbf{a}$ and set $\mathbf{T}=\left[\underline{\mathbf{t}}_{1,1,1}\cdots\underline{\mathbf{t}}_{q_{\mathsf{ct}},n,\ell}\right]$.

Given the problem instance, $\mathcal{B}$ chooses $\mathbf{W}_1,\ldots,\mathbf{W}_n\leftarrow\mathbb{Z}_p^{\ell\times k\ell n}$ to define msk. Since, $\mathcal{B}$ knows msk completely, it can respond to the secret key queries on its own. On j^{th} ciphertext query $(\mathbf{M}_j^{(0)},\mathbf{M}_j^{(1)})$, $\mathcal{B}$ samples $\mathbf{s}_j\leftarrow\mathbb{Z}_p^{k\ell n}$ and defines the ciphertext as following:

$$
\begin{aligned}
\mathbf{c}_{j,0} &= [\mathbf{s}_j]\\
\mathbf{c}_{j,i} &= \left[\mathbf{W}_i\mathbf{s}_j + \sum_{\iota\in[1,\phi_i(j)]}\mathbf{z}_{\psi_i(\iota),i}v_{j,i,\iota}^{(\delta)}+\mathbf{y}_{j,i}^{(\beta)}\right]
\end{aligned}
\tag{4}
$$

for all $i\in[1,n]$ where ϕ and ψ are defined as in the previous game. It is clear that the simulation is distributionally consistent. Precisely, if $\left[\mathbf{v}^{(0)}\right]$ is provided, the simulation is identical to $\mathbf{Game}_1$. Otherwise, the simulation is identical to $\mathbf{Game}_2$. If $\mathcal{A}$ distinguishes between $\mathbf{Game}_1$ and $\mathbf{Game}_2$, $\mathcal{B}$ can distinguish between $\left[\mathbf{v}^{(0)}\right]$ and $\left[\mathbf{v}^{(1)}\right]$. □

Acknowledgments. The authors thank Benoît Libert for interesting discussions. This work was supported in part by European Union Horizon 2020 Research and Innovation Program Grant 780701 and by BPI-France in the context of the national project RISQ (P141580).

References

1. Abdalla, M., Bourse, F., De Caro, A., Pointcheval, D.: Simple functional encryption schemes for inner products. In: Katz, J. (ed.) PKC 2015. LNCS, vol. 9020, pp. 733–751. Springer, Heidelberg (Mar / Apr 2015). https://doi.org/10.1007/978-3-662-46447-2_33

2. Abdalla, M., Catalano, D., Fiore, D., Gay, R., Ursu, B.: Multi-input functional encryption for inner products: Function-hiding realizations and constructions without pairings. In: Shacham, H., Boldyreva, A. (eds.) CRYPTO 2018, Part I. LNCS, vol. 10991, pp. 597–627. Springer, Heidelberg (Aug 2018). https://doi.org/10.1007/978-3-319-96884-1_20
3. Abdalla, M., Gay, R., Raykova, M., Wee, H.: Multi-input inner-product functional encryption from pairings. In: Coron, J.S., Nielsen, J.B. (eds.) EUROCRYPT 2017, Part I. LNCS, vol. 10210, pp. 601–626. Springer, Heidelberg (Apr / May 2017). https://doi.org/10.1007/978-3-319-56620-7_21
4. Agrawal, S., Bhattacherjee, S., Phan, D.H., Stehlé, D., Yamada, S.: Efficient public trace and revoke from standard assumptions: Extended abstract. In: Thuraisingham, B.M., Evans, D., Malkin, T., Xu, D. (eds.) ACM CCS 2017. pp. 2277–2293. ACM Press (Oct / Nov 2017). https://doi.org/10.1145/3133956.3134041
5. Agrawal, S., Libert, B., Stehlé, D.: Fully secure functional encryption for inner products, from standard assumptions. In: Robshaw, M., Katz, J. (eds.) CRYPTO 2016, Part III. LNCS, vol. 9816, pp. 333–362. Springer, Heidelberg (Aug 2016). https://doi.org/10.1007/978-3-662-53015-3_12
6. Agrawal, S., Yamada, S.: Optimal broadcast encryption from pairings and LWE. In: Canteaut, A., Ishai, Y. (eds.) EUROCRYPT 2020, Part I. LNCS, vol. 12105, pp. 13–43. Springer, Heidelberg (May 2020). https://doi.org/10.1007/978-3-030-45721-1_2
7. Barth, A., Boneh, D., Waters, B.: Privacy in encrypted content distribution using private broadcast encryption. In: Di Crescenzo, G., Rubin, A. (eds.) FC 2006. LNCS, vol. 4107, pp. 52–64. Springer, Heidelberg (Feb / Mar 2006)
8. Boneh, D., Franklin, M.K.: An efficient public key traitor tracing scheme. In: Wiener, M.J. (ed.) CRYPTO'99. LNCS, vol. 1666, pp. 338–353. Springer, Heidelberg (Aug 1999). https://doi.org/10.1007/3-540-48405-1_22
9. Boneh, D., Gentry, C., Waters, B.: Collusion resistant broadcast encryption with short ciphertexts and private keys. In: Shoup, V. (ed.) CRYPTO 2005. LNCS, vol. 3621, pp. 258–275. Springer, Heidelberg (Aug 2005). https://doi.org/10.1007/11535218_16
10. Boneh, D., Raghunathan, A., Segev, G.: Function-private subspace-membership encryption and its applications. In: Sako, K., Sarkar, P. (eds.) ASIACRYPT 2013, Part I. LNCS, vol. 8269, pp. 255–275. Springer, Heidelberg (Dec 2013). https://doi.org/10.1007/978-3-642-42033-7_14
11. Boneh, D., Sahai, A., Waters, B.: Functional encryption: Definitions and challenges. In: Ishai, Y. (ed.) TCC 2011. LNCS, vol. 6597, pp. 253–273. Springer, Heidelberg (Mar 2011). https://doi.org/10.1007/978-3-642-19571-6_16
12. Boneh, D., Waters, B.: A fully collusion resistant broadcast, trace, and revoke system. In: Juels, A., Wright, R.N., De Capitani di Vimercati, S. (eds.) ACM CCS 2006. pp. 211–220. ACM Press (Oct / Nov 2006). https://doi.org/10.1145/1180405.1180432
13. Castagnos, G., Laguillaumie, F., Tucker, I.: Practical fully secure unrestricted inner product functional encryption modulo p. In: Peyrin, T., Galbraith, S. (eds.) ASIACRYPT 2018, Part II. LNCS, vol. 11273, pp. 733–764. Springer, Heidelberg (Dec 2018). https://doi.org/10.1007/978-3-030-03329-3_25
14. Do, X.T., Phan, D.H., Yung, M.: A concise bounded anonymous broadcast yielding combinatorial trace-and-revoke schemes. In: Conti, M., Zhou, J., Casalicchio, E., Spognardi, A. (eds.) ACNS 20, Part II. LNCS, vol. 12147, pp. 145–164. Springer, Heidelberg (Oct 2020). https://doi.org/10.1007/978-3-030-57878-7_8
15. Dodis, Y., Fazio, N.: Public key trace and revoke scheme secure against adaptive chosen ciphertext attack. In: Desmedt, Y. (ed.) PKC 2003. LNCS, vol. 2567, pp. 100–115. Springer, Heidelberg (Jan 2003). https://doi.org/10.1007/3-540-36288-6_8
16. Escala, A., Herold, G., Kiltz, E., Ràfols, C., Villar, J.L.: An algebraic framework for Diffie-Hellman assumptions. Journal of Cryptology **30**(1), 242–288 (Jan 2017). https://doi.org/10.1007/s00145-015-9220-6
17. Fazio, N., Perera, I.M.: Outsider-anonymous broadcast encryption with sublinear ciphertexts. In: Fischlin, M., Buchmann, J., Manulis, M. (eds.) PKC 2012. LNCS, vol. 7293, pp. 225–242. Springer, Heidelberg (May 2012). https://doi.org/10.1007/978-3-642-30057-8_14
18. Goyal, R., Koppula, V., Waters, B.: Collusion resistant traitor tracing from learning with errors. In: Diakonikolas, I., Kempe, D., Henzinger, M. (eds.) 50th ACM STOC. pp. 660–670. ACM Press (Jun 2018). https://doi.org/10.1145/3188745.3188844
19. Kiayias, A., Samari, K.: Lower bounds for private broadcast encryption. In: Information Hiding. LNCS, vol. 7692, pp. 176–190. Springer (2012)
20. Kim, C.H., Hwang, Y.H., Lee, P.J.: An efficient public key trace and revoke scheme secure against adaptive chosen ciphertext attack. In: Laih, C.S. (ed.) ASIACRYPT 2003. LNCS, vol. 2894, pp. 359–373. Springer, Heidelberg (Nov / Dec 2003). https://doi.org/10.1007/978-3-540-40061-5_23
21. Li, J., Gong, J.: Improved anonymous broadcast encryptions - tight security and shorter ciphertext. In: Preneel, B., Vercauteren, F. (eds.) ACNS 18. LNCS, vol. 10892, pp. 497–515. Springer, Heidelberg (Jul 2018). https://doi.org/10.1007/978-3-319-93387-0_26

22. Libert, B., Paterson, K.G., Quaglia, E.A.: Anonymous broadcast encryption: Adaptive security and efficient constructions in the standard model. In: Fischlin, M., Buchmann, J., Manulis, M. (eds.) PKC 2012. LNCS, vol. 7293, pp. 206–224. Springer, Heidelberg (May 2012). https://doi.org/10.1007/978-3-642-30057-8_13
23. Naor, D., Naor, M., Lotspiech, J.: Revocation and tracing schemes for stateless receivers. In: Kilian, J. (ed.) CRYPTO 2001. LNCS, vol. 2139, pp. 41–62. Springer, Heidelberg (Aug 2001). https://doi.org/10.1007/3-540-44647-8_3
24. Naor, M., Pinkas, B.: Efficient trace and revoke schemes. In: Frankel, Y. (ed.) FC 2000. LNCS, vol. 1962, pp. 1–20. Springer, Heidelberg (Feb 2001)
25. Nguyen, P.: La géométrie des nombres en cryptologie. Ph.D. thesis, Université Paris 7 (1999)
26. Nishimaki, R., Wichs, D., Zhandry, M.: Anonymous traitor tracing: How to embed arbitrary information in a key. In: Fischlin, M., Coron, J.S. (eds.) EUROCRYPT 2016, Part II. LNCS, vol. 9666, pp. 388–419. Springer, Heidelberg (May 2016). https://doi.org/10.1007/978-3-662-49896-5_14
27. Shen, E., Shi, E., Waters, B.: Predicate privacy in encryption systems. In: Reingold, O. (ed.) TCC 2009. LNCS, vol. 5444, pp. 457–473. Springer, Heidelberg (Mar 2009). https://doi.org/10.1007/978-3-642-00457-5_27
28. Tomida, J.: Tightly secure inner product functional encryption: Multi-input and function-hiding constructions. In: Galbraith, S.D., Moriai, S. (eds.) ASIACRYPT 2019, Part III. LNCS, vol. 11923, pp. 459–488. Springer, Heidelberg (Dec 2019). https://doi.org/10.1007/978-3-030-34618-8_16
29. Wang, Z., Fan, X., Liu, F.H.: FE for inner products and its application to decentralized ABE. In: Lin, D., Sako, K. (eds.) PKC 2019, Part II. LNCS, vol. 11443, pp. 97–127. Springer, Heidelberg (Apr 2019). https://doi.org/10.1007/978-3-030-17259-6_4

A Single-Challenge Anonymous Security

In the Introduction, we informally discussed different practical scenarios involving the anonymity of revocation set. The anonymity security model in Section 2.4.2 is a multi-challenge security model and captures the security requirements of a typical broadcasting agency. However, the security definition is restrictive in principle as all the revoked sets in anonymity challenge queries are related. In this section, we first give a single-challenge security definition ($\mathsf{IND\text{-}ID\text{-}CPA}$) for revocation set hiding. Being a single-challenge security definition for symmetric-key settings, this new security definition ($\mathsf{IND\text{-}ID\text{-}CPA}$) for revocation set hiding supports multiple ciphertext queries along with multiple secret key queries and a single challenge anonymity query.

The positive side of $\mathsf{IND\text{-}ID\text{-}CPA}$ is that in the security proof, we no longer put any restriction on the revoked sets $\mathcal{R}$ across multiple ciphertext queries and challenge anonymity query. However, we still need to impose some new security restrictions on the adversary here in terms of post-challenge secret key queries. Precisely, we define $\mathsf{IND\text{-}ID^*\text{-}CPA}$ security that allows all pre-challenge queries (both key and ciphertext) and all post-challenge ciphertext queries (satisfying the natural restriction). However, for post-challenge secret key queries, $\mathsf{IND\text{-}ID^*\text{-}CPA}$ imposes a new restriction. In the literature, similar restriction has already been put like "outsider corruption" in [17]. However, unlike [17] we can still support "insider corruption" (i.e. post-challenge key queried on $\mathsf{id} \in \mathcal{R}_0 \cap \mathcal{R}_1$) completely and "outsider corruption" (i.e. post-challenge key queried on $\mathsf{id} \in \mathsf{ID} \setminus (\mathcal{R}_0 \cup \mathcal{R}_1)$) with some restriction. The restriction is a bit unusual in the sense, the adversary is not allowed to make post-challenge secret key queries on $\mathsf{id} \in \mathsf{ID} \setminus (\mathcal{R}_0 \cup \mathcal{R}_1)$ for an id that was a part of pre-challenge ciphertext query but was not queried for secret key in the pre-challenge query phase. But, this is all we require to argue our construction TR_0 is secure. Here, note that, TR_0 being a trace-and-revoke scheme with unbounded users, the set $\mathsf{ID} \setminus (\mathcal{R}_0 \cup \mathcal{R}_1)$ is sufficiently big and the adversary is restricted from making query on a small subset.

A.1 Security Definition

We first define the $\mathsf{IND\text{-}ID\text{-}CPA}$ security model and then weaken it to define $\mathsf{IND\text{-}ID^*\text{-}CPA}$ security. The $\mathsf{IND\text{-}ID\text{-}CPA}$ security of a trace-and-revoke scheme TR is defined based on the following game.

- The challenger runs $\mathsf{Setup}(1^\lambda, 1^r, 1^t)$ and gives the produced public parameter pp to the adversary $\mathcal{A}$. The adversary may ask the challenger to add polynomially many users in the system (these user addition queries can be adaptive and take place at any time in the game). The challenger updates dir accordingly.

- The adversary can adaptively make up to $(r+t)$ secret key queries, polynomially many ciphertext queries and a single anonymity challenge query, of the following form:
  - $*$ Given a key generation query id, the challenger provides the corresponding $\mathsf{sk_{id}}$ to $\mathcal{A}$.
  - $*$ Given a ciphertext query $(m, \mathcal{R})$ with $\mathcal{R} \subset \mathsf{ID}$ of size $\leq r$, the challenger provides $C \leftarrow \mathsf{Enc}(\mathsf{pp}, \mathsf{msk}, \mathsf{dir}, \mathcal{R}, m)$ to $\mathcal{A}$.
  - $*$ Given the challenge anonymity query $(m, \mathcal{R}_0, \mathcal{R}_1)$ with $\mathcal{R}_0, \mathcal{R}_1 \subset \mathsf{ID}$ of size $\leq r$, the challenger samples $\beta \leftarrow \{0, 1\}$ and provides $C^{(\beta)} \leftarrow \mathsf{Enc}(\mathsf{pp}, \mathsf{msk}, \mathsf{dir}, \mathcal{R}_\beta, m)$ to $\mathcal{A}$.

  These queries are subject to the restriction that for every queried id, either $\mathsf{id} \in \mathcal{R}_0 \cap \mathcal{R}_1$ or $\mathsf{id} \in \mathsf{ID} \setminus (\mathcal{R}_0 \cup \mathcal{R}_1)$. Among all the key queries that have been made, at most t of them could be satisfying $\mathsf{id} \in \mathsf{ID} \setminus (\mathcal{R}_0 \cup \mathcal{R}_1)$ and at most r of them could be satisfying $\mathsf{id} \in \mathcal{R}_0 \cap \mathcal{R}_1$.
- Finally, the adversary returns its guess $\beta' \in \{0, 1\}$ for the bit β chosen by the challenger. The adversary wins this game if $\beta = \beta'$.

The advantage of the adversary $\mathcal{A}$ is defined as $\mathrm{Adv}^{\mathsf{IND\text{-}ID\text{-}CPA}}_{\mathsf{TR},\mathcal{A}} = |\Pr[\beta = \beta'] - 1/2|$.

We then weaken the security model a small amount to define $\mathsf{IND\text{-}ID^*\text{-}CPA}$ security, which does not allow post-challenge secret key queries on $\mathsf{id} \in \mathsf{ID} \setminus (\mathcal{R}_0 \cup \mathcal{R}_1)$ for an id that was a part of pre-challenge ciphertext query but was not queried for secret key in the pre-challenge query phase. A trace-and-revoke scheme TR is said to be $\mathsf{IND\text{-}ID^*\text{-}CPA}$ secure if the advantage $\mathrm{Adv}^{\mathsf{IND\text{-}ID^*\text{-}CPA}}_{\mathsf{TR},\mathcal{A}}$ is negligible for all ppt adversary $\mathcal{A}$.

A.2 Security

Theorem 9. *If $\mathcal{IPFE}$ is an* $\mathsf{IND\text{-}CPA}$ *secure inner product functional encryption scheme allowing up to $(t + r - 1)$ key extraction queries, then* TR_0 *is* $\mathsf{IND\text{-}ID^*\text{-}CPA}$ *secure.*

Before we give the proof, we informally discuss the necessity of such unusual restriction of $\mathsf{IND\text{-}ID^*\text{-}CPA}$ security. Note that, in TR_0, for every id we assign a uniformly random vector $\mathbf{x}_{\mathsf{id}}$. However, being a symmetric-key trace-and-revoke, we define such an assignment on the fly when an id is referred for the first time. Thus, in the post-challenge phase, we can say for all id in pre-challenge ciphertext queries, a corresponding $\mathbf{x}_{\mathsf{id}}$ vector has already been assigned. With overwhelming probability, such $\mathbf{x}_{\mathsf{id}} \notin \mathsf{RowSpan}(\mathbf{Z})$ (see Equation (5).) This then creates a distributional problem while simulation. To avoid such scenario, we impose the restriction only on post-challenge "outsider corruption" queries not to include id for which $(\mathsf{id}, \mathbf{x}_{\mathsf{id}})$ relation has been fixed but has not been queried for key extraction. We now give a formal proof of the theorem.

Proof. Let $\mathcal{A}_{\mathsf{TR}_0}$ be a ppt adversary that breaks the $\mathsf{IND\text{-}ID^*\text{-}CPA}$ security of TR_0. Note that $\mathcal{A}_{\mathsf{TR}_0}$ is allowed to corrupt at most t legitimate users and the ciphertext is created considering at most r revoked users. We construct a ppt adversary $\mathcal{A}_{\mathcal{IPFE}}$ that breaks the $\mathsf{IND\text{-}CPA}$ security of the underlying $\mathcal{IPFE}$.

- It first obtains the public parameter pp output by the $\mathcal{IPFE}$ challenger (who runs the $\mathcal{IPFE}.\mathsf{Setup}(1^\lambda, 1^\ell)$ algorithm) and relays it to $\mathcal{A}_{\mathsf{TR}_0}$. On $\mathcal{A}_{\mathsf{TR}_0}$'s request, the adversary $\mathcal{A}_{\mathcal{IPFE}}$ creates dir with polynomially many $(\mathsf{id}, \mathbf{x}_{\mathsf{id}})$ pairs for $\mathbf{x}_{\mathsf{id}} \leftarrow \mathbb{Z}_p^\ell$. It then sets up two empty dictionaries $Q_{\mathsf{sk}} = \{\}$ and $Q_{\mathsf{ct}} = \{\}$. Informally speaking, Q_{sk} contains all id for which key query have been/could be made and Q_{ct} contains all id on which key query has not yet been made.
- When $\mathcal{A}_{\mathcal{IPFE}}$ receives a pre-challenge secret key query for $\mathsf{id} \in \mathsf{ID}$ from $\mathcal{A}_{\mathsf{TR}_0}$, it proceeds as follows:
  - $*$ If $\mathsf{id} \in Q_{\mathsf{ct}}$, it updates $Q_{\mathsf{sk}}[\mathsf{id}] = Q_{\mathsf{ct}}[\mathsf{id}]$ and removes the id entry from Q_{ct}.
  - $*$ If $\mathsf{id} \notin Q_{\mathsf{sk}}$, it samples $\mathbf{x}_{\mathsf{id}} \leftarrow \mathbb{Z}_p^\ell$ and sets $Q_{\mathsf{sk}}[\mathsf{id}] = \mathbf{x}_{\mathsf{id}}$.
  - $*$ If $\mathsf{id} \in Q_{\mathsf{sk}}$, it sets $\mathbf{x}_{\mathsf{id}} = Q_{\mathsf{sk}}[\mathsf{id}]$.
  - $*$ It then sends $\mathbf{x}_{\mathsf{id}}$ to the $\mathcal{IPFE}$ challenger. The latter returns $\mathsf{sk}_{\mathbf{x}_{\mathsf{id}}}$, which $\mathcal{A}_{\mathcal{IPFE}}$ forwards to $\mathcal{A}_{\mathsf{TR}_0}$ as $\mathsf{sk_{id}}$.
- When $\mathcal{A}_{\mathcal{IPFE}}$ receives a ciphertext query on $(m, \mathcal{R})$, it proceeds as follows:
  - $*$ For all $\mathsf{id} \in \mathcal{R} \setminus ((\mathcal{R} \cap Q_{\mathsf{sk}}.\mathsf{vals}()) \cup (\mathcal{R} \cap Q_{\mathsf{ct}}.\mathsf{vals}()))$, it samples $\mathbf{x}_{\mathsf{id}} \leftarrow \mathbb{Z}_p^\ell$ and then adds $Q_{\mathsf{ct}}[\mathsf{id}] = \mathbf{x}_{\mathsf{id}}$.
  - $*$ It samples $\mathbf{v}_{\mathcal{R}} \leftarrow \mathbf{X}^\perp$ where $\mathbf{X} = \{\mathbf{x}_{\mathsf{id}} \ : \ \mathsf{id} \in \mathcal{R}\} \subseteq Q_{\mathsf{sk}}.\mathsf{vals}() \cup Q_{\mathsf{ct}}.\mathsf{vals}()$.
  - $*$ It sends $\mathbf{y} = m \cdot \mathbf{v}_{\mathcal{R}}$ to the $\mathcal{IPFE}$ challenger. The latter returns $\mathsf{ct}_{\mathbf{y}}$, which $\mathcal{A}_{\mathcal{IPFE}}$ forwards to $\mathcal{A}_{\mathsf{TR}_0}$ as the ciphertext response $\mathsf{ct}_{\mathcal{R}}$.

- When $\mathcal{A}_{\mathcal{IPFE}}$ receives $\mathcal{A}_{\mathsf{TR}_0}$'s challenge query on $(m, \mathcal{R}_0, \mathcal{R}_1)$, it proceeds as follows:
  1. First, it sets $Q_{\mathsf{idR}} = \{\mathsf{id} : \mathsf{id} \in Q_{\mathsf{ct}} \setminus (\mathcal{R}_0 \cup \mathcal{R}_1)\}$.
  2. Then it defines $\hat{\mathcal{R}}_0 = \mathcal{R}_0 \setminus \mathcal{R}_1$, $\hat{\mathcal{R}} = \mathcal{R}_0 \cap \mathcal{R}_1$ and $\hat{\mathcal{R}}_1 = \mathcal{R}_1 \setminus \mathcal{R}_0$.
  3. For all $\mathsf{id} \in \hat{\mathcal{R}} \setminus Q_{\mathsf{sk}}$,
     - If $\mathsf{id} \notin Q_{\mathsf{ct}}$, it samples $\mathbf{x}_{\mathsf{id}} \leftarrow \mathbb{Z}_p^{\ell}$ and updates $Q_{\mathsf{sk}}[\mathsf{id}] = \mathbf{x}_{\mathsf{id}}$.
     - Otherwise, it updates $Q_{\mathsf{sk}}[\mathsf{id}] = Q_{\mathsf{ct}}[\mathsf{id}]$ and removes the id entry from Q_{ct}.
  4. Then it defines $\mathbf{Z} = \mathsf{Matrix}((Q_{\mathsf{sk}}.\mathsf{vals}() \setminus \hat{\mathcal{R}}) \sqcup T)$ where $T = \{\mathbf{x}_{\mathsf{id}} \leftarrow \mathbb{Z}_p^{\ell}\}$ is of size $t - |(Q_{\mathsf{sk}}.\mathsf{vals}() \setminus \hat{\mathcal{R}})|$.
  5. For all $\mathsf{id} \in (\hat{\mathcal{R}}_0 \setminus Q_{\mathsf{ct}}.\mathsf{vals}()) \cup (\hat{\mathcal{R}}_1 \setminus Q_{\mathsf{ct}}.\mathsf{vals}())$, it samples $\mathbf{x}_{\mathsf{id}} \leftarrow \mathbb{Z}_p^{\ell}$ and updates $Q_{\mathsf{sk}}[\mathsf{id}] = \mathbf{x}_{\mathsf{id}}$.
  6. It sets $\mathbf{X}_0 = \mathsf{Matrix}(\{\mathbf{x}_{\mathsf{id}} : \mathsf{id} \in \mathcal{R}_0 \cap (Q_{\mathsf{sk}}.\mathsf{vals}() \cup Q_{\mathsf{ct}}.\mathsf{vals}())\})$ and $\mathbf{X}_1 = \mathsf{Matrix}(\{\mathbf{x}_{\mathsf{id}} : \mathsf{id} \in \mathcal{R}_1 \cap (Q_{\mathsf{sk}}.\mathsf{vals}() \cup Q_{\mathsf{ct}}.\mathsf{vals}())\})$.
  7. It samples $\begin{pmatrix} \mathbf{v}_{\mathcal{R}_0} \\ \mathbf{v}_{\mathcal{R}_1} \end{pmatrix} \leftarrow \mathbf{V}^{\perp}$ for
     $$\mathbf{V} = \begin{pmatrix} \mathbf{Z} & -\mathbf{Z} \\ \mathbf{X}_0 & \mathbf{0} \\ \mathbf{0} & \mathbf{X}_1 \end{pmatrix}. \tag{5}$$
  8. It sends $\mathbf{y}_{\mathcal{R}_0} = m \cdot \mathbf{v}_{\mathcal{R}_0}$ and $\mathbf{y}_{\mathcal{R}_1} = m \cdot \mathbf{v}_{\mathcal{R}_1}$ to the $\mathcal{IPFE}$ challenger who samples $\beta \leftarrow \{0,1\}$ and encrypts $\mathbf{y}_{\mathcal{R}_\beta}$ as $\mathsf{ct}^{(\beta)} \leftarrow \mathcal{IPFE}.\mathsf{Enc}(\mathsf{pp}, \mathsf{msk}, \mathbf{y}_{\mathcal{R}_\beta})$ and outputs $\mathsf{ct}^{(\beta)}$.
  9. $\mathcal{A}_{\mathcal{IPFE}}$ then forwards the received ciphertext to $\mathcal{A}_{\mathsf{TR}_0}$ as its challenge $C^{(\beta)}$.
- $\mathcal{A}_{\mathsf{TR}_0}$ can make queries for secret key on $\mathsf{id} \in \mathsf{ID}$ and for ciphertext queries on $\mathcal{R}$.
  - ∗ For all post-challenge key queries on id, $\mathcal{A}_{\mathcal{IPFE}}$ does the following:
    1. If $\mathsf{id} \in \mathcal{R}_0 \cap \mathcal{R}_1$, it retrieves $\mathbf{x}_{\mathsf{id}} = Q_{\mathsf{sk}}[\mathsf{id}]$.
    2. If $\mathsf{id} \notin \mathcal{R}_0 \cup \mathcal{R}_1$, if $\mathsf{id} \notin Q_{\mathsf{sk}}$, it samples $\mathbf{x}_{\mathsf{id}} \leftarrow \mathsf{RowSpan}(\mathbf{Z})$ and sets $Q_{\mathsf{sk}}[\mathsf{id}] = \mathbf{x}_{\mathsf{id}}$.
    3. It then sends $\mathbf{x}_{\mathsf{id}}$ to the challenger who returns $\mathsf{sk}_{\mathbf{x}_{\mathsf{id}}}$ which $\mathcal{A}_{\mathcal{IPFE}}$ forwards to $\mathcal{A}_{\mathsf{TR}_0}$ as $\mathsf{sk}_{\mathsf{id}}$.
  - ∗ For all ciphertext queries on $(m, \mathcal{R})$, $\mathcal{A}_{\mathcal{IPFE}}$ does the following:
    1. For all $\mathsf{id} \in \mathcal{R} \setminus ((\mathcal{R} \cap Q_{\mathsf{sk}}.\mathsf{vals}()) \cup (\mathcal{R} \cap Q_{\mathsf{ct}}.\mathsf{vals}()))$, it samples $\mathbf{x}_{\mathsf{id}} \leftarrow \mathbb{Z}_p^{\ell}$ and then adds $Q_{\mathsf{ct}}[\mathsf{id}] = \mathbf{x}_{\mathsf{id}}$.
    2. It samples $\mathbf{v}_{\mathcal{R}} \leftarrow \mathbf{X}^{\perp}$ where $\mathbf{X} = \{\mathbf{x}_{\mathsf{id}} \ : \ \mathsf{id} \in \mathcal{R}\}$.
    3. Sends $\mathbf{y} = m \cdot \mathbf{v}_{\mathcal{R}}$ to the challenger who returns $\mathsf{ct}_{\mathbf{y}}$ which $\mathcal{A}_{\mathcal{IPFE}}$ forwards to $\mathcal{A}_{\mathsf{TR}_0}$ as the ciphertext response $\mathsf{ct}_{\mathcal{R}}$.
- Finally, adversary $\mathcal{A}_{\mathsf{TR}_0}$ outputs its guess $\beta' \in \{0,1\}$ and $\mathcal{A}_{\mathcal{IPFE}}$ also outputs β' as its own guess of β.

From Equation (5), $\mathbf{Z}(\mathbf{v}_{\mathcal{R}_0} - \mathbf{v}_{\mathcal{R}_1}) = \mathbf{0}$, $\mathbf{X}_0 \mathbf{v}_{\mathcal{R}_0} = \mathbf{0}$ and $\mathbf{X}_1 \mathbf{v}_{\mathcal{R}_1} = \mathbf{0}$. As $\mathbf{y}_{\mathcal{R}_j} \in \mathrm{Span}(\mathbf{v}_{\mathcal{R}_j})$ for $j \in \{0,1\}$, $\mathbf{Z}(\mathbf{y}_{\mathcal{R}_1} - \mathbf{y}_{\mathcal{R}_0}) = \mathbf{0}$ and $\mathbf{X}_0 \mathbf{y}_{\mathcal{R}_0} = \mathbf{X}_1 \mathbf{y}_{\mathcal{R}_1} = \mathbf{0}$. As a result, for any $\mathbf{x}_{\mathsf{id}} \in \mathsf{RowSpan}(\mathbf{Z})$, $\mathbf{x}_{\mathsf{id}}^{\top}(\mathbf{y}_{\mathcal{R}_1} - \mathbf{y}_{\mathcal{R}_0}) = 0$. Thus, $\mathcal{A}_{\mathcal{IPFE}}$ behaves as a valid adversary in the IND-ID^*-CPA security model.

Firstly, note that, $\mathcal{A}_{\mathsf{TR}_0}$ gets encryption of either $\mathbf{y}_{\mathcal{R}_0}$ or $\mathbf{y}_{\mathcal{R}_1}$ where both the vectors are randomly sampled. Thus, from the point of view of $\mathcal{A}_{\mathsf{TR}_0}$, $\mathbf{Z}\mathbf{y}_{\mathcal{R}_b}$ is a random vector. Then, we show that $\mathcal{A}_{\mathsf{TR}_0}$ sees $\mathbf{x}_{\mathsf{id}}$ purely random even though $\mathbf{x}_{\mathsf{id}}$ is sampled randomly from $\mathsf{RowSpan}(\mathbf{Z})$. This follows from the fact that $\mathcal{A}_{\mathsf{TR}_0}$ has access to all purely random vectors $\mathbf{x}_{\mathsf{id}} \leftarrow \mathbb{Z}_p^{\ell}$ for $\mathsf{id} \in \hat{\mathcal{R}}$. Thus, from the point of view of $\mathcal{A}_{\mathsf{TR}_0}$, it has access to $|\hat{\mathcal{R}}|$ basis vectors of $\mathbb{Z}_p^{\ell}$ and the space $\mathbb{Z}_p^{\ell}$ is left with entropy of $(\ell - |\hat{\mathcal{R}}|)$ basis vectors where $\ell - |\hat{\mathcal{R}}| > t$. As $\mathcal{A}_{\mathsf{TR}_0}$ gets at most t samples of $\mathbf{x}_{\mathsf{id}} \leftarrow \mathsf{RowSpan}(\mathbf{Z})$, it sees $\mathbf{x}_{\mathsf{id}}$ identically distributed to vectors chosen uniformly random from $\mathbb{Z}_p^{\ell}$. The ciphertext and the secret keys are already properly distributed since $\mathcal{A}_{\mathcal{IPFE}}$ has forwarded the reply of $\mathcal{IPFE}$ challenger. This shows that $\mathcal{A}_{\mathcal{IPFE}}$ behaves as a valid challenger in the IND-ID^*-CPA security model.

If $\mathcal{A}_{\mathsf{TR}_0}$ can distinguish between $\mathcal{R}_0$ and $\mathcal{R}_1$, $\mathcal{A}_{\mathcal{IPFE}}$ can distinguish between $\mathbf{y}_{\mathcal{R}_0}$ and $\mathbf{y}_{\mathcal{R}_1}$. Thus, the advantage of $\mathcal{A}_{\mathcal{IPFE}}$ is exactly the same as the advantage of $\mathcal{A}_{\mathsf{TR}_0}$. □