Hardness of k-LWE and Applications in Traitor Tracing

San Ling[1], Duong Hieu Phan[2], Damien Stehlé[3], and Ron Steinfeld[4]

[1] Division of Mathematical Sciences,
School of Physical and Mathematical Sciences,
Nanyang Technological University, Singapore
[2] Laboratoire LAGA (CNRS, U. Paris 8, U. Paris 13), U. Paris 8, France
[3] Laboratoire LIP (U. Lyon, CNRS, ENSL, INRIA, UCBL), ENS de Lyon, France
[4] Faculty of Information Technology,
Monash University, Clayton, Australia

Abstract. We introduce the k-LWE *problem*, a Learning With Errors variant of the k-SIS problem. The Boneh-Freeman reduction from SIS to k-SIS suffers from an exponential loss in k. We improve and extend it to an LWE to k-LWE reduction with a polynomial loss in k, by relying on a new technique involving trapdoors for random integer kernel lattices. Based on this hardness result, we present the first algebraic construction of a traitor tracing scheme whose security relies on the worst-case hardness of standard lattice problems. The proposed LWE traitor tracing is almost as efficient as the LWE encryption. Further, it achieves public traceability, i.e., allows the authority to delegate the tracing capability to "untrusted" parties. To this aim, we introduce the notion of *projective sampling family* in which each sampling function is keyed and, with a projection of the key on a well chosen space, one can simulate the sampling function in a computationally indistinguishable way. The construction of a projective sampling family from k-LWE allows us to achieve public traceability, by publishing the projected keys of the users. We believe that the new lattice tools and the projective sampling family are quite general that they may have applications in other areas.

Keywords: Lattice-based cryptography, Traitor tracing, LWE.

1 Introduction

Since the pioneering work of Ajtai [3], there have been a number of proposals of cryptographic schemes with security provably relying on the worst-case hardness of standard lattice problems, such as the decision Gap Shortest Vector Problem with polynomial gap (see the surveys [30,40]). These schemes enjoy unmatched security guarantees: Security relies on *worst-case* hardness assumptions for problems expected to be *exponentially hard* to solve (with respect to the lattice dimension n), even with quantum computers. At the same time, they often enjoy great asymptotic efficiency, as the basic operations are matrix-vector multiplications in dimension $\widetilde{O}(n)$ over a ring of cardinality $\leq \mathcal{P}oly(n)$. A breakthrough result in that field was the introduction of the Learning With Errors problem (LWE) by Regev [38,39], who showed it to be at least as hard as worst-case lattice problems and exploited it to devise an elementary encryption scheme.

J.A. Garay and R. Gennaro (Eds.): CRYPTO 2014, Part I, LNCS 8616, pp. 315–334, 2014.

Gentry et al. showed in [19] that Regev's scheme may be adapted so that a master can generate a large number of secret keys for the same public key. As a result, the latter encryption scheme, called dual-Regev, can be naturally extended into a multi-receiver encryption scheme. In the present work, we build traitor tracing schemes from this dual-Regev LWE-based encryption scheme.

TRAITOR TRACING. A traitor tracing scheme is a multi-receiver encryption scheme where malicious receiver coalitions aiming at building pirate decryption devices are deterred by the existence of a tracing algorithm: Using the pirate decryption device, the tracing algorithm can recover at least one member of the malicious coalition. Such schemes are particularly well suited for fighting copyright infringement in the context of commercial content distribution (e.g., Pay-TV, subscription news websites, etc). Since their introduction by Chor et al. [15], much work has been devoted to devising efficient and secure traitor tracing schemes. The most desirable schemes are fully collusion resistant: they can deal with arbitrarily large malicious coalitions. But, unsurprisingly, the most efficient schemes are in the bounded collusion model where the number of malicious users is limited. The first non-trivial fully collusion resistant scheme was proposed by Boneh et al. [11]. However, its ciphertext size is still large ($\Omega(\sqrt{N})$, where N is the total number of users) and it relies on pairing groups of composite order. Very recently, Boneh and Zhandry [12] proposed a fully collusion resistant scheme with poly-log size parameters. It relies on indistinguishability obfuscation [18], whose security foundation remains to be studied, and whose practicality remains to be exhibited. In this paper, we focus on the bounded collusion model. The Boneh-Franklin scheme [7] is one of the earliest algebraic constructions but it can still be considered as the reference algebraic transformation from the standard ElGamal public key encryption into traitor tracing. This transformation induces a linear loss in efficiency, with respect to the maximum number of traitors. The known transformations from encryption to traitor tracing in the bounded collusion model present at least a linear loss in efficiency, either in the ciphertext size or in the private key size [7,31,23,41,6,10]. We refer to [21] for a detailed introduction to this rich topic.

OUR CONTRIBUTIONS. We describe the first algebraic construction of a public-key lattice-based traitor tracing scheme. It is semantically secure and enjoys public traceability. The security relies on the hardness of LWE, which is known to be at least as hard as standard worst-case lattice problems [39,33,13].

The scheme is the extension, described above, of the dual-Regev LWE-based encryption scheme from [19] to a multi-receiver encryption scheme, where each user has a different secret key. In the case of traitor tracing, several keys may be leaked to a traitor coalition. To show that we can trace the traitors, we extend the LWE problem and introduce the k-LWE problem, in which k hint vectors (the leaked keys) are given out.

Intuitively, k-LWE asks to distinguish between a random vector $\boldsymbol{t}$ close to a given lattice Λ and a random vector $\boldsymbol{t}$ close to the orthogonal subspace of the span of k given short vectors belonging to the dual Λ^* of that lattice. Even if we are given $(\boldsymbol{b}_i^*)_{i\leq k}$ small in Λ^*, computing the inner products $\langle \boldsymbol{b}_i^*, \boldsymbol{t}\rangle$ will not help in solving this problem, since they are small and distributed identically in both cases. The k-LWE problem can be interpreted as a dual of the k-SIS problem introduced by Boneh and Freeman [8], which intuitively requests to find a short vector in Λ^* that is linearly independent with the k given short vectors of Λ^*. Their reduction from SIS to k-SIS can be adapted to the LWE setup, but the hardness loss incurred by the reduction is gigantic. We propose a significantly sharper reduction from LWE_α to k-LWE_α. This improved reduction requires a new lattice technique: the equivalent for kernel lattices of Ajtai's simultaneous sampling of a random q-ary lattice with a short basis [4] (see also Lemma 2). We adapt the Micciancio-Peikert framework from [28] to sampling a Gaussian $X \in \mathbb{Z}^{m \times n}$ along with a short basis for the lattice $\ker(X) = \{\boldsymbol{b} \in \mathbb{Z}^m : \boldsymbol{b}^t X = \boldsymbol{0}\}$. Kernel lattices also play an important role in the re-randomization analysis of the recent lattice-based multilinear map scheme of Garg et al. [17], and we believe that our new trapdoor generation tool for such lattices is likely find additional applications in future. We also remark that our technique can be adapted to the SIS to k-SIS reduction. We thus solve the open question left by Boneh and Freeman of improving their reduction [8]: from an exponential loss in k to a polynomial loss in k. Consequently, their linearly homomorphic signatures and ordinary signature schemes enjoy much better efficiency/security trade-offs.

Our construction of a traitor tracing scheme from k-LWE can be seen as an additive and noisy variant of the (black-box) Boneh-Franklin traitor tracing scheme [7]. While the Boneh-Franklin scheme is transformed from the ElGamal encryption with a linear loss (in the maximum number of traitors) in efficiency, our scheme is almost as efficient as standard LWE-based encryption, as long as the maximum number of traitors is bounded below $n/(c \log n)$, where n is the LWE dimension determined by the security parameter, and c is a constant. The full functionality of black-box tracing in both the Boneh-Franklin scheme and ours are of high complexity as they both rely on the black-box confirmation: given a superset of the traitors, it is guaranteed to find at least one traitor and no innocent suspect is incriminated. Boneh and Franklin left the improvement of the black-box tracing as an interesting open problem. We show that in lattice setting, the black-box tracing can be accelerated by running the tracing procedure in parallel on untrusted machines. This is a direct consequence of the property of public traceability, i.e., the possibility of running tracing procedure on public information, that our scheme enjoys. We note that almost all traitor tracing systems require that the tracing key must be kept secret. Some schemes [14,37,9,12] achieve public traceability and some others achieve a stronger notion than public traceability, namely the non-repudation, but the setup in these schemes require some interactive protocol between the center and each user such as a secure 2-party computation protocol in [35], a commitment protocol in [36], an oblivious polynomial evaluation in [42,24,22].

To obtain public traceability and inspired from the notion of projective hash family [16], we introduce a new notion of *projective sampling family* in which each sampling function is keyed and, with a projection of the key on a well chosen space, one can simulate the sampling function in a computationally indistinguishable way. The construction of a set of projective sampling families from k-LWE allows us to publicly sample the tracing signals.

Independently, our new lattice tools may have applications in other areas. The k-LWE problem has a similar flavour to the Extended-LWE problem from [32]. It would be interesting to exhibit reductions between these problems. On a closely-related topic, it seems our sampling of a random Gaussian integer matrix X together with a short basis of $\ker(X)$ is compatible with the hardness proof of Extended-LWE from [13]. In particular, it should be possible to use it as an alternative to [13, Def 4.5] in the proof of [13, Le 4.7], to show that Extended-LWE remains hard with many hints independently sampled from discrete Gaussians.

REMARK. Due to lack of space, some background and the missing proofs of Sections 3 and 5 have been removed from this proceedings version. The full version is available on the webpages of the authors.

2 Preliminaries

If x is a real number, then $\lfloor x \rceil$ is the closest integer to x (with any deterministic rule in case x is half an odd integer). All vectors will be denoted in bold. By default, our vectors are column vectors. We let $\langle \cdot, \cdot \rangle$ denote the canonical inner product. For q prime, we let $\mathbb{Z}_q$ denote the field of integers modulo q. For two matrices A, B of compatible dimensions, we let $(A|B)$ and $(A\|B)$ respectively denote the horizontal and vertical concatenations of A and B. For $A \in \mathbb{Z}_q^{m \times n}$, we define $\mathrm{Im}(A) = \{A\boldsymbol{s} : \boldsymbol{s} \in \mathbb{Z}_q^n\} \subseteq \mathbb{Z}_q^m$. For $X \subseteq \mathbb{Z}_q^m$, we let $\mathrm{Span}(X)$ denote the set of all linear combinations of elements of X. We let $X^\perp$ denote the linear subspace $\{\boldsymbol{b} \in \mathbb{Z}_q^m : \forall \boldsymbol{c} \in X, \langle \boldsymbol{b}, \boldsymbol{c} \rangle = 0\}$. For a matrix $S \in \mathbb{R}^{m \times n}$, we let $\|S\|$ denote the norm of its longest column. If S is full column-rank, we let $\sigma_1(S) \geq \ldots \geq \sigma_n(S)$ denote its singular values. We let $\mathbb{T}$ denote the additive group $\mathbb{R}/\mathbb{Z}$.

If D_1 and D_2 are distributions over a countable set X, their statistical distance $\frac{1}{2}\sum_{x \in X} |D_1(x) - D_2(x)|$ will be denoted by $\Delta(D_1, D_2)$. The statistical distance is defined similarly if X is measurable. If X is of finite weight, we let $U(X)$ denote the uniform distribution over X. For any invertible $S \in \mathbb{R}^{m \times m}$ and $\boldsymbol{c} \in \mathbb{R}^m$, we define the function $\rho_{S,\boldsymbol{c}}(\boldsymbol{b}) = \exp(-\pi\|S^{-1}(\boldsymbol{b} - \boldsymbol{c})\|^2)$. For $S = sI_m$, we write $\rho_{s,\boldsymbol{c}}$, and we omit the subscripts S and $\boldsymbol{c}$ when $S = I_m$ and $\boldsymbol{c} = \boldsymbol{0}$. We let ν_α denote the one-dimensional Gaussian distribution with standard deviation α.

2.1 Euclidean Lattices and Discrete Gaussian Distributions

A lattice is a set of the form $\{\sum_{i \leq n} x_i \boldsymbol{b}_i : x_i \in \mathbb{Z}\}$ where the $\boldsymbol{b}_i$'s are linearly independent vectors in $\mathbb{R}^m$. In this situation, the $\boldsymbol{b}_i$'s are said to form a basis of the n-dimensional lattice. The n-th minimum $\lambda_n(L)$ of an n-dimensional lattice L is defined as the smallest r such that the n-dimensional closed hyperball of radius r centered in $\boldsymbol{0}$ contains n linearly independent vectors of L. The smoothing parameter of L is defined as $\eta_\varepsilon(L) = \min\{r > 0 : \rho_{1/r}(\widehat{L} \setminus \boldsymbol{0}) \leq \varepsilon\}$ for any $\varepsilon \in (0, 1)$, where $\widehat{L} = \{\boldsymbol{c} \in \mathrm{Span}(L) : \boldsymbol{c}^t \cdot L \subseteq \mathbb{Z}\}$ is the dual lattice of L. It was proved in [29, Le. 3.3] that $\eta_\varepsilon(L) \leq \sqrt{\ln(2n(1 + 1/\varepsilon))/\pi} \cdot \lambda_n(L)$ for all $\varepsilon \in (0, 1)$ and n-dimensional lattices L.

For a lattice $L \subseteq \mathbb{R}^m$, a vector $\boldsymbol{c} \in \mathbb{R}^m$ and an invertible $S \in \mathbb{R}^{m \times m}$, we define the Gaussian distribution of parameters L, $\boldsymbol{c}$ and S by $D_{L,S,\boldsymbol{c}}(\boldsymbol{b}) \sim \rho_{S,\boldsymbol{c}}(\boldsymbol{b}) = \exp(-\pi\|S^{-1}(\boldsymbol{b} - \boldsymbol{c})\|^2)$ for all $\boldsymbol{b} \in L$. When $S = \sigma \cdot I_m$, we simply write $D_{L,\sigma,\boldsymbol{c}}$.

Note that $D_{L,S,\boldsymbol{c}} = S^t \cdot D_{S^{-t}L,1,S^{-t}\boldsymbol{c}}$. Sometimes, for convenience, we use the notation $D_{L+\boldsymbol{c},S}$ as a shorthand for $\boldsymbol{c} + D_{L,S,-\boldsymbol{c}}$. Gentry et al. [19] gave an algorithm, referred to as GPV algorithm, to sample from $D_{L,S,\boldsymbol{c}}$ when given as input a basis $(\boldsymbol{b}_i)_i$ of L such that $\sqrt{\ln(2n+4)/\pi} \cdot \max_i \|S^{-t}\boldsymbol{b}_i\| \leq 1$.

We extensively use q-ary lattices. The q-ary lattice associated to $A \in \mathbb{Z}_q^{m\times n}$ is defined as $\Lambda^{\perp}(A) = \{\boldsymbol{x} \in \mathbb{Z}^m : \boldsymbol{x}^t \cdot A = \boldsymbol{0} \bmod q\}$. It has dimension m, and a basis can be computed in polynomial-time from A. For $\boldsymbol{u} \in \mathbb{Z}_q^m$, we define $\Lambda_{\boldsymbol{u}}^{\perp}(A)$ as the coset $\{\boldsymbol{x} \in \mathbb{Z}^m : \boldsymbol{x}^t \cdot A = \boldsymbol{u}^t \bmod q\}$ of $\Lambda^{\perp}(A)$.

2.2 Random Lattices

We consider the following random lattices, called q-ary Ajtai lattices. They are obtained by sampling $A \hookleftarrow U(\mathbb{Z}_q^{m\times n})$ and considering $\Lambda^{\perp}(A)$. The following lemma provides a probabilistic bound on the smoothing parameter of $\Lambda^{\perp}(A)$.

Lemma 1 (Adapted from [19, Le. 5.3]). *Let q be prime and m, n integers with $m \geq 2n$ and $\varepsilon > 0$, then $\eta_\varepsilon(\Lambda^{\perp}(A)) \leq 4q^{\frac{n}{m}}\sqrt{\log(2m(1+1/\varepsilon))/\pi}$, for all except a fraction $2^{-\Omega(n)}$ of $A \in \mathbb{Z}_q^{m\times n}$.*

It is possible to efficiently sample a close to uniform A along with a short basis of $\Lambda^{\perp}(A)$ (see [4,5,34,28]).

Lemma 2 (Adapted from [5, Th. 3.1]). *There exists a ppt algorithm that given n, m, $q \geq 2$ as inputs samples two matrices $A \in \mathbb{Z}_q^{m\times n}$ and $T \in \mathbb{Z}^{m\times m}$ such that: the distribution of A is within statistical distance $2^{-\Omega(n)}$ from $U(\mathbb{Z}_q^{m\times n})$; the rows of T form a basis of $\Lambda^{\perp}(A)$; each row of T has norm $\leq 3mq^{n/m}$.*

For $A \in \mathbb{Z}_q^{m\times n}$, $S \in \mathbb{R}^{m\times m}$ invertible, $\boldsymbol{c} \in \mathbb{R}^m$ and $\boldsymbol{u} \in \mathbb{Z}_q^n$, we define the distribution $D_{\Lambda_{\boldsymbol{u}}^{\perp}(A),S,\boldsymbol{c}}$ as $\bar{\boldsymbol{c}} + D_{\Lambda^{\perp}(A),S,-\bar{\boldsymbol{c}}+\boldsymbol{c}}$, where $\bar{\boldsymbol{c}}$ is any vector of $\mathbb{Z}^m$ such that $\bar{\boldsymbol{c}}^t \cdot A = \boldsymbol{u}^t \bmod q$. A sample $\boldsymbol{x}$ from $D_{\Lambda_{\boldsymbol{u}}^{\perp}(A),S}$ can be obtained using the GPV algorithm along with the short basis of $\Lambda^{\perp}(A)$ provided by Lemma 2. Boneh and Freeman [8] showed how to efficiently obtain the residual distribution of $(A, \boldsymbol{x})$ without relying on Lemma 2.

Theorem 1 (Adapted from [8, Th. 4.3]). *Let $n, m, q \geq 2$, $k \geq 0$ and $S \in \mathbb{R}^{m\times m}$ be such that $m \geq 2n$, q is prime with $q > \sigma_1(S) \cdot \sqrt{2\log(4m)}$, and $\sigma_m(S) = q^{\frac{n}{m}} \cdot \max(\Omega(\sqrt{n\log m}), 2\sigma_1(S)^{\frac{k}{m}})$. Let $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_k \in \mathbb{Z}_q^n$ and $\boldsymbol{c}_1, \ldots, \boldsymbol{c}_k \in \mathbb{R}^m$ be arbitrary. Then the residual distributions of the tuple $(A, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_k)$ obtained with the following two experiments are within statistical distance $2^{-\Omega(n)}$.*

$$\texttt{Exp}_0: \quad A \hookleftarrow U(\mathbb{Z}_q^{m\times n}); \qquad \forall i \leq k : \boldsymbol{x}_i \hookleftarrow D_{\Lambda_{\boldsymbol{u}_i}^{\perp}(A),S,\boldsymbol{c}_i}\,.$$

$$\texttt{Exp}_1: \forall i \leq k : \boldsymbol{x}_i \hookleftarrow D_{\mathbb{Z}^m,S,\boldsymbol{c}_i};\; A \hookleftarrow U\left(\mathbb{Z}_q^{m\times n} | \forall i \leq k : \boldsymbol{x}_i^t \cdot A = \boldsymbol{u}_i^t \bmod q\right).$$

This statement generalizes [8, Th. 4.3] in three ways. First, the latter corresponds to the special case corresponding to taking all the $\boldsymbol{u}_i$'s and $\boldsymbol{c}_i$'s equal to $\boldsymbol{0}$. This generalization does not add any extra complication in the proof of [8, Th. 4.3], but is important for our constructions. Second, the condition on m is less restrictive (the corresponding assumption in [8, Th. 4.3] is that $m \geq \max(2n \log q, 2k)$). To allow for such small values of m, we refine the bound on the smoothing parameter of the $\Lambda^{\perp}(A)$ lattice (namely, we use Lemma 1). Third, we allow for a non-spherical Gaussian distribution, which seems needed in our generalized Micciancio-Peikert trapdoor gadget used in the reduction from LWE to k-LWE in Section 3.2.

We also use the following result on the probability of the Gaussian vectors $\boldsymbol{x}_i$ from Theorem 1 being linearly independent over $\mathbb{Z}_q$.

Lemma 3 (Adapted from [8, Le. 4.5]). *With the notations and assumptions of Theorem 1, the k vectors $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k$ sampled in* $\mathtt{Exp}_0$ *and* $\mathtt{Exp}_1$ *are linearly independent over $\mathbb{Z}_q$, except with probability* $2^{-\Omega(n)}$.

2.3 Rényi Divergence

We use Rényi Divergence (RD) in our analysis, relying on techniques developed in [27,25,26]. For any two probability distributions P and Q such that the support of P is a subset of the support of Q over a countable domain X, we define the RD (of order 2) by $R(P\|Q) = \sum_{x \in X} \frac{P(x)^2}{Q(x)}$, with the convention that the fraction is zero when both numerator and denominator are zero. We recall that the RD between two offset discrete Gaussians is bounded as follows.

Lemma 4 ([25, Le. 4.2]). *For any n-dimensional lattice $L \subseteq \mathbb{R}^n$ and invertible matrix S, set $P = D_{L,S,\boldsymbol{w}}$ and $Q = D_{L,S,\boldsymbol{z}}$ for some fixed $\boldsymbol{w}, \boldsymbol{z} \in \mathbb{R}^n$. If $\boldsymbol{w}, \boldsymbol{z} \in L$, let $\varepsilon = 0$. Otherwise, fix $\varepsilon \in (0,1)$ and assume that $\sigma_n(S) \geq \eta_\varepsilon(L)$. Then $R(P\|Q) \leq \left(\frac{1+\varepsilon}{1-\varepsilon}\right)^2 \cdot \exp\left(2\pi\|\boldsymbol{w} - \boldsymbol{z}\|^2/\sigma_n(S)^2\right)$.*

We use this bound and the fact that the RD between the parameter distributions of two distinguishing problems can be used to relate their hardness, if they satisfy a certain public samplability property.

Lemma 5 ([26]). *Let Φ, Φ' denote two distributions, and $D_0(r)$ and $D_1(r)$ denote two distributions determined by some parameter r. Let P, P' be two decision problems defined as follows:*

- *P: Assess whether input x is sampled from distribution X_0 or X_1, where*
$$X_0 = \{x : r \hookleftarrow \Phi, x \hookleftarrow D_0(r)\},\ X_1 = \{x : r \hookleftarrow \Phi, x \hookleftarrow D_1(r)\}.$$
- *P': Assess whether input x is sampled from distribution X_0' or X_1', where*
$$X_0' = \{x : r \hookleftarrow \Phi', x \hookleftarrow D_0(r)\},\ X_1' = \{x : r \hookleftarrow \Phi', x \hookleftarrow D_1(r)\}.$$

Assume that $D_0(\cdot)$ and $D_1(\cdot)$ have the following public samplability *property: there exists a sampling algorithm S with run-time T_S such that for all r, b, given any sample x from $D_b(r)$ we have:*

- $\mathsf{S}(0, x)$ *outputs a sample distributed as $D_0(r)$ over the randomness of* S.
- $\mathsf{S}(1, x)$ *outputs a sample distributed as $D_1(r)$ over the randomness of* S.

If there exists a T-time distinguisher A for problem P with advantage ε, then , for every $\lambda > 0$, there exists an $O(\lambda\varepsilon^{-2} \cdot (T_S + T))$-time distinguisher A' for problem P' with advantage $\varepsilon' \geq \frac{\varepsilon^3}{8R(\Phi\|\Phi')} - O(2^{-\lambda})$.

2.4 Learning with Errors

Let $\boldsymbol{s} \in \mathbb{Z}_q^n$ and $\alpha > 0$. We define the distribution $A_{\boldsymbol{s},\alpha}$ as follows: Take $\boldsymbol{a} \hookleftarrow U(\mathbb{Z}_q^n)$ and $e \hookleftarrow \nu_\alpha$, and return $(\boldsymbol{a}, \frac{1}{q}\langle \boldsymbol{a}, \boldsymbol{s}\rangle + e) \in \mathbb{Z}_q^n \times \mathbb{T}$. The *Learning With Errors problem* LWE$_\alpha$, introduced by Regev in [38,39], consists in assessing whether an oracle produces samples from $U(\mathbb{Z}_q^n \times \mathbb{T})$ or $A_{\boldsymbol{s},\alpha}$ for some constant $\boldsymbol{s} \hookleftarrow U(\mathbb{Z}_q^n)$. Regev [39] showed that for $q \leq \mathcal{P}oly(n)$ prime and $\alpha \in (\frac{\sqrt{n}}{2q}, 1)$, LWE is (quantumly) not easier than standard worst-case lattice problems in dimension n with approximation factors $\mathcal{P}oly(n)/\alpha$. This hardness proof was partly dequantized in [33,13], and the requirements that q should be prime and $\mathcal{P}oly(n)$ were waived.

In this work, we consider a variant LWE where the number of oracle samples that the distinguisher requests is a priori bounded. If m denotes that bound, then we will refer to this restriction as LWE$_{\alpha,m}$. In this situation, the hardness assumption can be restated in terms of linear algebra over $\mathbb{Z}_q$: Given $A \hookleftarrow U(\mathbb{Z}_q^{m\times n})$, the goal is to distinguish between the distributions (over $\mathbb{T}^m$)

$$\frac{1}{q}U\left(\mathrm{Im}(A)\right) + \nu_\alpha^m \quad \text{and} \quad \frac{1}{q}U\left(\mathbb{Z}_q^m\right) + \nu_\alpha^m.$$

Under the assumption that $\alpha q \geq \Omega(\sqrt{n})$, the right hand side distribution is indeed within statistical distance $2^{-\Omega(n)}$ to $U(\mathbb{T}^m)$ (see, e.g., [29, Le. 4.1]). The hardness assumption states that by adding to them a small Gaussian noise, the linear spaces $\mathrm{Im}(A)$ and $\mathbb{Z}_q^m$ become computationally indistinguishable. This rephrasing in terms of linear algebra is helpful in the security proof of the traitor tracing scheme. Note that by a standard hybrid argument, distinguishing between the two distributions given one sample from either, and distinguishing between them given Q samples (from the same distribution), are computationally equivalent problems, up to a loss of a factor Q in the distinguishing advantage.

Finally, we will also use a variant of LWE where the noise distribution ν_α is replaced by $D_{q^{-1}\mathbb{Z},\alpha}$, and where $U(\mathbb{T})$ is replaced by $U(\mathbb{T}_q)$ with $\mathbb{T}_q$ being $q^{-1}\mathbb{Z}$ with addition mod 1. This variant, denoted by LWE$'$, was proved in [34] to be no easier than standard LWE (up to a constant factor increase in α).

3 New Lattice Tools

The security of our constructions relies on the hardness of a new variant of LWE, which may be seen as the dual of the k-SIS problem from [8].

Definition 1. *Let $k \leq m$, $S \in \mathbb{R}^{m\times m}$ invertible and $C = (\boldsymbol{c}_1\| \cdots \|\boldsymbol{c}_k) \in \mathbb{R}^{k\times m}$. The (k, S, C)-*LWE$_{\alpha,m}$ *problem (or (k, S)-LWE if $C = 0$) is as follows: Given $A \hookleftarrow U(\mathbb{Z}_q^{m\times n})$, $\boldsymbol{u} \hookleftarrow U(\mathbb{Z}_q^n)$ and $\boldsymbol{x}_i \hookleftarrow D_{\Lambda^{\perp}_{-\boldsymbol{u}}(A),S,\boldsymbol{c}_i}$ for $i \leq k$, the goal is to distinguish between the distributions (over $\mathbb{T}^{m+1}$)*

$$\frac{1}{q} \cdot U\Big(\mathrm{Im}\Big(\frac{\boldsymbol{u}^t}{A}\Big)\Big) + \nu_\alpha^{m+1} \quad \textit{and} \quad \frac{1}{q} \cdot U\Big(\mathrm{Span}_{i\leq k}\Big(\frac{1}{\boldsymbol{x}_i}\Big)^{\perp}\Big) + \nu_\alpha^{m+1}.$$

The classical LWE problem consists in distinguishing the left distribution from uniform, without the hint vectors $\boldsymbol{x}_i^+ = (1\|\boldsymbol{x}_i)$. These hint vectors correspond to the secret keys obtained by the malicious coalition in the traitor tracing scheme. Once these hint vectors are revealed, it becomes easy to distinguish the left distribution from the uniform distribution: take one of the vectors $\boldsymbol{x}_i^+$, get a challenge sample $\boldsymbol{y}$ and compute $\langle \boldsymbol{x}_i^+, \boldsymbol{y}\rangle \in \mathbb{T}$; if $\boldsymbol{y}$ is a sample from the left distribution, then the centered residue is expected to be of size $\approx \alpha \cdot (\sqrt{m}\sigma_1(S) + \|\boldsymbol{c}_i\|)$, which is $\ll 1$ for standard parameter settings; on the other hand, if $\boldsymbol{y}$ is sampled from the uniform distribution, then $\langle \boldsymbol{x}^+, \boldsymbol{y}\rangle$ should be uniform. The definition of (k, S)-LWE handles this issue by replacing $U(\mathbb{Z}_q^{m+1})$ by $U(\mathrm{Span}_{i\le k}(\boldsymbol{x}_i^+)^\perp)$.

Sampling $\boldsymbol{x}_i^+$ from $D_{\Lambda^\perp((\boldsymbol{u}^t\|A)),S,\boldsymbol{c}_i}$ may seem more natural than imposing that the first coordinate of each $\boldsymbol{x}_i^+$ is 1. Looking ahead, this constraint will prove convenient to ensure correctness of our cryptographic primitives. Theorem 3 below and its proof can be readily adapted to this hint distribution. They may also be adapted to improve the SIS to k-SIS reduction from [8]. Setting $C = 0$ is also more natural, but for technical reasons, our reduction from LWE to (k, S, C)-LWE works with unit vectors $\boldsymbol{c}_i$. However, we show that for small $\|\boldsymbol{c}_i\|$, there exist polynomial time reductions between (k, S, C)-LWE and (k, S)-LWE.

In the proof of the hardness of (k, S)-LWE problem, we rely on a gadget integral matrix G that has the following properties: its first rows have Gaussian distributions, it is unimodular and its inverse is small. Before going to this proof, we shall build such a gadget matrix by extending Ajtai's simultaneous sampling of a random q-ary lattice with a short basis [4] (see also Lemma 2) to kernel lattices. More precisely, we adapt the Micciancio-Peikert framework [28] to sampling a Gaussian $X \in \mathbb{Z}^{m\times n}$ along with a short basis for the lattice $\ker(X) = \{\boldsymbol{b} \in \mathbb{Z}^m : \boldsymbol{b}^t X = \boldsymbol{0}\}$.

3.1 Sampling a Gaussian X with a Small Basis of $\ker(X)$

The Micciancio-Peikert construction [28] relies on a *leftover hash lemma* stating that with overwhelming probability over $A \hookleftarrow U(\mathbb{Z}_q^{m\times n})$ and for a sufficiently large σ, the distribution of $A^t \cdot D_{\mathbb{Z}^m,\sigma} \bmod q$ is statistically close to $U(\mathbb{Z}_q^n)$. We use a similar result over the integers, starting from a Gaussian $X \in \mathbb{Z}^{m\times n}$ instead of a uniform $A \in \mathbb{Z}_q^{m\times n}$. The proof of the following lemma relies on [1], which improves over a similar result from [2]. The result would be neater with $\sigma_2 = \sigma_1$, but, unfortunately, we do not know how to achieve it. The impact of this drawback on our results and constructions is mostly cosmetic.

Lemma 6. *Let $m \ge n \ge 100$ and $\sigma_1, \sigma_2 > 0$ satisfying $\sigma_1 \ge \Omega(\sqrt{mn\log m})$, $m \ge \Omega(n\log(\sigma_1 n))$ and $\sigma_2 \ge \Omega(n^{5/2}\sqrt{m}\sigma_1^2\log^{3/2}(m\sigma_1))$. Let $X \hookleftarrow D_{\mathbb{Z},\sigma_1}^{m\times n}$. There exists a ppt algorithm that takes $n, m, \sigma_1, \sigma_2, X$ and $\boldsymbol{c} \in \mathbb{Z}^n$ as inputs and returns $\boldsymbol{x} \in \mathbb{Z}^n, \boldsymbol{r} \in \mathbb{Z}^m$ such that $\boldsymbol{x} = \boldsymbol{c} + X^t\boldsymbol{r}$ with $\|\boldsymbol{r}\| \le O(\sigma_2/\sigma_1)$, with probability $1 - 2^{-\Omega(n)}$, and*

$$\Delta\big((X, \boldsymbol{x}), D_{\mathbb{Z},\sigma_1}^{m\times n} \times D_{\mathbb{Z}^n,\sigma_2,\boldsymbol{c}}\big) \le 2^{-\Omega(n)}.$$

We now adapt the trapdoor construction from [28] to kernel lattices.

Theorem 2. *Let $n, m_1, \sigma_1, \sigma_2$ be as above, and $m_2 \geq m_1$ bounded as $n^{O(1)}$. There exists a ppt algorithm that given n, m_1, m_2 (in unary), σ_1 and σ_2, returns $X_1 \in \mathbb{Z}^{m_1 \times n}$, $X_2 \in \mathbb{Z}^{m_2 \times n}$, and $U \in \mathbb{Z}^{m \times m}$ with $m = m_1 + m_2$, such that:*

- *the distribution of (X_1, X_2) is within statistical distance $2^{-\Omega(n)}$ of $D_{\mathbb{Z},\sigma_1}^{m_1 \times n} \times (D_{\mathbb{Z}^{m_2},\sigma_2,\boldsymbol{\delta}_1} \times \cdots \times D_{\mathbb{Z}^{m_2},\sigma_2,\boldsymbol{\delta}_n})$, where $\boldsymbol{\delta}_i$ denotes the ith canonical unit vector in $\mathbb{Z}^{m_2}$ whose ith coordinate is 1 and whose remaining coordinates are 0.*
- *we have $|\det U| = 1$ and $U \cdot X = (I_n \| 0)$ with $X = (X_1 \| X_2)$,*
- *every row of U has norm $\leq O(\sqrt{nm_1}\sigma_2)$ with probability $\geq 1 - 2^{-\Omega(n)}$.*

The second statement implies that the last $m - n$ rows of U form a basis of the random lattice $\ker(X)$.

Proof. We first sample X_1 from $D_{\mathbb{Z},\sigma_1}^{m_1 \times n}$ using the GPV algorithm. We run m_2 times the algorithm from Lemma 6, on the input $n, m_1, \sigma_1, \sigma_2, X_1$ and $\boldsymbol{c}$ running through the columns of $C = [I_n | 0_{n \times (m_2 - n)}]$. This gives $X_2 \in \mathbb{Z}^{m_2 \times n}$ and $R \in \mathbb{Z}^{m_1 \times m_2}$ such that $X_2^t = [I_n | \mathbf{0}_{n \times (m_2 - n)}] + X_1^t \cdot R$. One can then see that $U \cdot X = [I_n \| \mathbf{0}]$, where

$$U = \left[\begin{array}{c|c} \mathbf{0} & I_{m_2} \\ \hline I_{m_1} & -(X_1|\mathbf{0}) \end{array}\right] \cdot \left[\begin{array}{c|c} I_{m_1} & \mathbf{0} \\ \hline -R^t & I_{m_2} \end{array}\right] = \left[\begin{array}{c|c} -R^t & I_{m_2} \\ \hline I_{m_1} + (X_1|\mathbf{0})R^t & -(X_1|\mathbf{0}) \end{array}\right], X = \left[\begin{array}{c} X_1 \\ \hline X_2 \end{array}\right].$$

The result then follows from Gaussian tail bounds (to bound the norms of the rows of X_1) and elementary computations. □

Our gadget matrix G is U^{-t}. In the following corollary, we summarize the properties we will use.

Corollary 1. *Let $n, m_1, m_2, m, \sigma_1, \sigma_2$ be as in Theorem 2. There exists a ppt algorithm that given n, m_1, m_2 (in unary), and σ_1, σ_2 as inputs, returns $G \in \mathbb{Z}^{m \times m}$ such that:*

- *the top $n \times m$ submatrix of G is within statistical distance $2^{-\Omega(n)}$ of $D_{\mathbb{Z},\sigma_1}^{n \times m_1} \times (D_{\mathbb{Z}^{m_2},\sigma_2,\boldsymbol{\delta}_1} \times \cdots \times D_{\mathbb{Z}^{m_2},\sigma_2,\boldsymbol{\delta}_n})^t$,*
- *we have $|\det G| = 1$ and $\|G^{-1}\| \leq O(\sqrt{nm_2}\sigma_2)$, with probability $1 - 2^{-\Omega(n)}$.*

3.2 Hardness of *k*-LWE

The following result shows that this LWE variant, with S a specific diagonal matrix, is no easier than LWE.

Theorem 3. *There exists $c > 0$ such that the following holds for $k = n/(c \log n)$. Let m, q, σ, σ' be such that $\sigma \geq \Omega(n)$, $\sigma' \geq \Omega(n^3\sigma^2/\log n)$, $q \geq \Omega(\sigma'\sqrt{\log m})$ is prime, and $m \geq \Omega(n \log q)$ (e.g., $\sigma = \Theta(n)$, $\sigma' = \Theta(n^5/\log n)$, $q = \Theta(n^5)$ and $m = \Theta(n \log n)$). Then there exists a probabilistic polynomial-time reduction from $\mathrm{LWE}_{m+1,\alpha}$ in dimension n to (k, S)-$\mathrm{LWE}_{m+2n,\alpha'}$ in dimension $4n$, with $\alpha' = \Omega(mn^{3/2}\sigma\sigma'\alpha)$ and $S = \left[\begin{array}{c|c} \sigma \cdot I_{m+n} & 0 \\ \hline 0 & \sigma' \cdot I_n \end{array}\right]$. More concretely, using a (k, S)-$\mathrm{LWE}_{m+2n,\alpha'}$ algorithm with run-time T and advantage ε, the reduction gives an $\mathrm{LWE}_{m+1,\alpha}$ algorithm with advantage $\varepsilon' \geq \frac{\varepsilon^3}{8R(\Phi\|\Phi')} - O(2^{-\lambda})$ and advantage $\varepsilon' = \Omega((\varepsilon - 2^{-\Omega(n/\log n)})^3) - O(2^{-n})$.*

The reduction takes an LWE instance and extends it to a related k-LWE instance for which the additional hint vectors $(\boldsymbol{x}_i)_{i\leq k}$ are known. The major difficulty in this extension is to restrain the noise increase, as a function of k.

The existing approach for this reduction (that we improve below) is the technique used in the SIS to k-SIS reduction from [8]. In the latter approach, the hint vectors are chosen independently from a small discrete Gaussian distribution, and then the LWE matrix A is extended to a larger matrix A' under the constraint that the hint vectors are in the q-ary lattice $\Lambda^{\perp}(A') = \{\boldsymbol{b} : \boldsymbol{b}^t A' = \boldsymbol{0} \bmod q\}$. Unfortunately, with this approach, the transformation from an LWE sample with respect to A, to a k-LWE sample with respect to A', involves a multiplication by the cofactor matrix $\det(G) \cdot G^{-1}$ over $\mathbb{Z}$ of a $k \times k$ full-rank submatrix G of the hint vectors matrix. Although the entries of G are small, the entries of its cofactor matrix are almost as large as $\det G$, which is exponential in k. This leads to an "exponential noise blowup," restraining the applicability range to $k \leq \widetilde{O}(1)$ if one wants to rely on the hardness of LWE with noise rate $1/\alpha \leq \mathcal{P}oly(n)$ (otherwise, LWE is not exponentially hard to solve). To restrain the noise increase for large k, we use the gadget of Corollary 1. Ignoring several technicalities, the core idea underlying our reduction is that the latter gadget allows us to sample a small matrix $\overline{X}_2$ with $\overline{X}_2^{-1}$ also small, which we can then use to transform the given LWE matrix $A^+ = (\boldsymbol{u}^t \| A) \in \mathbb{Z}_q^{(m+1)\times n}$ into a taller k-LWE matrix $A'^+ = T \cdot A^+$, using a transformation matrix T of the form

$$T = \left[\begin{array}{c} I_{m+1} \\ \hline -\overline{X}_2^{-1} X_1 \end{array}\right],$$

for some small independently sampled matrix $X_1 = [\boldsymbol{1}|\overline{X}_1]$. We can accordingly transform the given LWE sample vector $\boldsymbol{b} = A^+\boldsymbol{s} + \boldsymbol{e}$ for matrix A^+ into an LWE sample $\boldsymbol{b}' = T\boldsymbol{b} = A'^+\boldsymbol{s} + T\boldsymbol{e}$ for matrix A'^+ by multiplying the given sample by T. Since $[X_1|\overline{X}_2] \cdot T = 0$, it follows that $[X_1|X_2] \cdot A'^+ = 0$, so we can use k small rows of $[X_1|\overline{X}_2]$ as the k-LWE hints $\boldsymbol{x}_i^+$ for the new matrix A'^+, while, at same time, the smallness of T keeps the transformed noise $\boldsymbol{e}' = T\boldsymbol{e}$ small.

Proof. For a technical reason related to the non-zero centers $\boldsymbol{\delta}_i$ in the distribution of the hint vectors produced by our gadget from Corollary 1, we decompose our reduction from $\mathrm{LWE}_{m+1,\alpha}$ to (k, S)-LWE into two subreductions. The first subreduction (outlined above) reduces $\mathrm{LWE}_{m+1,\alpha}$ in dimension n to (k, S, C)-$\mathrm{LWE}_{m+2n,\alpha'}$ in dimension $4n$, where the ith row of C is the unit vector $\boldsymbol{c}_i = (0^{m+n}|\boldsymbol{\delta}_i) \in \mathbb{R}^{m+2n}$ for $i = 1, \ldots, k$. The second subreduction reduces (k, S, C)-$\mathrm{LWE}_{m+2n,\alpha'}$ in dimension $4n$ to (k, S)-$\mathrm{LWE}_{m+2n,\alpha'}$ in dimension $4n$. We first describe and analyze the first subreduction, and then explain the second subreduction.

Description of the First Subreduction. Let $(A^+, \boldsymbol{b})$ with $A^+ = (\boldsymbol{u}^t \| A)$ denote the given $\mathrm{LWE}_{\alpha,m+1}$ input instance, where $A^+ \hookleftarrow U(\mathbb{Z}_q^{(m+1)\times n})$, and $\boldsymbol{b} \in \mathbb{T}^{m+1}$ comes from either the "LWE distribution" $\frac{1}{q}U\left(\mathrm{Im}(A^+)\right) + \nu_\alpha^{m+1}$ or the "Uniform distribution" $\frac{1}{q}U\left(\mathbb{Z}_q^{m+1}\right) + \nu_\alpha^{m+1}$. The reduction maps $(A^+, \boldsymbol{b})$ to $(A', \boldsymbol{u}', X, \boldsymbol{b}')$ with $A' \in \mathbb{Z}_q^{(m+2n)\times 4n}$ and $\boldsymbol{u}' \in \mathbb{Z}_q^{4n}$ independent and uniform, $X \in \mathbb{Z}^{k\times(m+2n)}$ with its ith row $\boldsymbol{x}_i$ independently sampled from $D_{\Lambda^{\perp}_{-\boldsymbol{u}'}(A'),S}$ for $i \leq k$, and $\boldsymbol{b}' \in \mathbb{T}^{m+1+2n}$ coming from either the "k-LWE distribution" $\frac{1}{q}U\left(\mathrm{Im}(A'^{+})\right) + \nu_{\alpha}^{m+1+2n}$ if $\boldsymbol{b}$ is from the "LWE distribution," or the "k-Uniform distribution" $\frac{1}{q}U\left(\mathrm{Span}_{i \leq k}(\boldsymbol{x}_i^{+})^{\perp}\right)$ if $\boldsymbol{b}$ is from the "Uniform distribution." Here $A'^{+} = (\boldsymbol{u}'^{t} \| A')$, and $\boldsymbol{x}_i^{+}$ denotes the vector $(1 \| \boldsymbol{x}_i)$ for $i \leq k$. The reduction is as follows.

1. Sample gadget $\overline{X}_2 \in \mathbb{Z}^{2n \times 2n}$ using Corollary 1 (with parameters n, m_1, m_2, σ_1, σ_2 set to k, n, n, σ, σ' respectively), and sample $\overline{X}_1 \hookleftarrow D_{\mathbb{Z},\sigma}^{2n \times m}$. Define $T = \left[\frac{I_{m+1}}{-\overline{X}_2^{-1} \cdot (\mathbf{1}|\overline{X}_1)}\right] \in \mathbb{Z}^{(m+1+2n)\times(m+1)}$, where $\mathbf{1}$ is the all-1 vector. Let $X \in \mathbb{Z}^{k \times (m+2n)}$ denote the matrix made of the top k rows of $(\overline{X}_1|\overline{X}_2)$.
2. Sample $C^{+} \in \mathbb{Z}_q^{(m+1+2n)\times 3n}$ with independent columns uniform orthogonally to $\mathrm{Im}((\mathbf{1}|X))$ modulo q. Let $\boldsymbol{u}_C^{t} \in \mathbb{Z}_q^{3n}$ be the top row of C^{+}, and $C \in \mathbb{Z}_q^{(m+2n)\times 3n}$ denote its remaining $m+2n$ rows.
3. Compute $\Sigma = \alpha' \cdot I_{m+1+2n} - T \cdot T^{t}$ and $\sqrt{\Sigma}$ such that $\sqrt{\Sigma} \cdot \sqrt{\Sigma}^{t} = \Sigma$; if Σ is not positive definite, abort.
4. Compute $A'^{+} = (T \cdot A^{+}|C^{+})$ and $\boldsymbol{b}' = T\boldsymbol{b} + \frac{1}{q}C^{+} \cdot \boldsymbol{s}' + \sqrt{\Sigma}\boldsymbol{e}'$, with $\boldsymbol{s}' \hookleftarrow U(\mathbb{Z}_q^{3n})$ and $\boldsymbol{e}' \hookleftarrow \nu_1^{m+1+2n}$. Let $(\boldsymbol{u}')^{t} = (\boldsymbol{u} \| \boldsymbol{u}_C)^{t} \in \mathbb{Z}_q^{4n}$ be the top row of A'^{+}.
5. Return $(A', \boldsymbol{u}', X, \boldsymbol{b}')$.

Step 1 aims at building a transformation matrix T that sends A^{+} to the left n columns of A'^{+}. Two properties are required from this transformation. First, it must be a linear map with small coefficients, so that when we map the LWE right hand side to the k-LWE right hand side, the noise component does not blow up. Second, it must contain some vectors $(1\|\boldsymbol{x}_i)$ in its (left) kernel, with $\boldsymbol{x}_i$ normally distributed. These vectors are to be used as k-LWE hints. For this, we use the gadget of the previous subsection. This ensures that the $\boldsymbol{x}_i$'s are (almost) distributed as independent Gaussian samples from $D_{\mathbb{Z}^n,\sigma} \times D_{\mathbb{Z}^n,\sigma'}$, and that the matrix T is integral with small coefficients. We define $B \in \mathbb{Z}_q^{2n \times n}$ by $[A^{+}\|B] = TA^{+}$, so that we have:

$$[\mathbf{1}|\overline{X}_1|\overline{X}_2] \cdot \left[\frac{A^{+}}{B}\right] = [\mathbf{1}|\overline{X}_1|\overline{X}_2] \cdot \left[\frac{I_{m+1}}{-\overline{X}_2^{-1} \cdot (\mathbf{1}|\overline{X}_1)}\right] \cdot A^{+} = \mathbf{0} \bmod q.$$

This means each row of $\left(\overline{X}_1|\overline{X}_2\right)$ belongs to $\Lambda^{\perp}_{-\boldsymbol{u}}(A'')$, where $A'' = [A^{t}|B^{t}]^{t}$.

At this stage, it is tempting to define the k-LWE matrix as A'' and give away the k-LWE hint vectors $\boldsymbol{x}_i \in \Lambda^{\perp}_{-\boldsymbol{u}}(A'')$ making up the matrix X. However, this approach does not quite work: we have extended A by $2n$ rows, but we give only k hint vectors (we cannot output them all, as the bottom rows of $\overline{X}_2$ may not be normally distributed). This creates a difficulty for mapping "Uniform" to "k-Uniform" in the reduction. Step 2 circumvents the above difficulty by sampling extra column vectors $C^{+} \in \mathbb{Z}_q^{(m+1+2n)\times 3n}$ that are uniform in the subspace orthogonal to the hint vectors $\boldsymbol{x}_i^{+}$ modulo q. When the parameters are properly set, the columns of $[T|C^{+}]$ span the full subspace orthogonal to the $\boldsymbol{x}_i$'s mod q, with overwhelming probability. We finally set $A'^{+} = \left[\frac{A^{+}}{B}\middle|C^{+}\right]$.

It remains to see how to map "LWE" to "k-LWE." The main problem, when multiplying $\boldsymbol{b}$ by T, is that the LWE noise gets skewed. If its covariance matrix was of the form

$\alpha^2 \cdot I_{m+1}$, then it becomes $\alpha^2 T \cdot T^t$. To compensate for that, in Step 3, we add to $T \cdot \boldsymbol{b}$ an independent Gaussian noise with well-chosen covariance $\Sigma = \alpha'^2 \cdot I_{m+1+2n} - \alpha^2 T \cdot T^t$. We set α' large enough to ensure that this symmetric matrix is positive definite. This noise unskewing technique was adapted to discrete Gaussians and used in cryptography in [34].

Analysis of the First Subreduction. All steps of the reduction can be implemented in polynomial time. Its correctness follows from the following three lemmas. The proofs can be found in the full version.

Lemma 7. *The tuple $(A', \boldsymbol{u}', X)$ is within statistical distance $2^{-\Omega(n/\log n)}$ of the distribution in which $A' \in \mathbb{Z}_q^{(m+2n)\times 4n}$ and $\boldsymbol{u}' \in \mathbb{Z}_q^{4n}$ are independent and uniform, and the rows of $X \in \mathbb{Z}^{k\times(m+2n)}$ are from $D_{\Lambda^{\perp}_{-\boldsymbol{u}'}(A'),S,\boldsymbol{c}_i}$, where $\boldsymbol{c}_i = (0^{m+n}|\boldsymbol{\delta}_i) \in \mathbb{R}^{m+2n}$ and $\boldsymbol{\delta}_i$ denotes the ith canonical unit vector in $\mathbb{Z}^n$ for $i = 1, \ldots, k$.*

Next, we assume that (A'^+, X) is fixed and consider the distribution of $\boldsymbol{b}'$ in the two cases of the distribution of $\boldsymbol{b}$. First we consider the "LWE" to "k-LWE" distribution mapping.

Lemma 8. *The following holds with probability $1 - 2^{-\Omega(n/\log n)}$ over the choice of $\overline{X}_1$ and $\overline{X}_2$. If $\boldsymbol{b} \in \mathbb{T}^{m+1}$ is sampled from $\frac{1}{q}U(\mathrm{Im}A) + \nu_\alpha^{m+1}$, then $\boldsymbol{b}' \in \mathbb{T}^{m+1+2n}$ is within statistical distance $2^{-\Omega(n)}$ of $\frac{1}{q}U\left(\mathrm{Im}A'^+\right) + \nu_{\alpha'}^{m+1+2n}$.*

Finally, we consider the "Uniform" to "k-Uniform" distribution mapping.

Lemma 9. *The following holds with probability $1 - 2^{-\Omega(n/\log n)}$ over the choice of $\overline{X}_1$ and $\overline{X}_2$. If $\boldsymbol{b}$ is sampled from $\frac{1}{q}U\left(\mathbb{Z}_q^{m+1}\right) + \nu_\alpha^{m+1}$, then $\boldsymbol{b}'$ is within statistical distance $2^{-\Omega(n)}$ of $\frac{1}{q}U\left(\mathrm{Span}_{i\le k}(\boldsymbol{x}_i^+)^\perp\right) + \nu_{\alpha'}^{m+1+2n}$.*

Overall, we have described a reduction that maps the "LWE distribution" to the "k-LWE distribution," and the "Uniform distribution" to the "k-Uniform distribution," up to statistical distance $2^{-\Omega(n/\log n)}$.

Second Subreduction. It remains to reduce the (k, S, C)-LWE with non-zero centers for the hint distribution, to (k, S)-LWE with zero-centered hints. For this, we use Lemma 5 to obtain the following.

Lemma 10. *Let $m' = m + 2n$, $n' = 4n$, and assume that $\sigma_{m'}(S) \ge \omega(\sqrt{n})$. If there exists a distinguisher against (k, S)-$\mathrm{LWE}_{m',\alpha'}$ in dimension n' with run-time T and advantage ε, then there exists a distinguisher against (k, S, C)-$\mathrm{LWE}_{m',\alpha'}$ with run-time $T' = O(\mathcal{P}oly(m') \cdot (\varepsilon - 2^{-\Omega(n)})^{-2} \cdot T)$ and advantage $\varepsilon' = \Omega((\varepsilon - O(2^{-n}))^3 / R - O(2^{-n}))$, where $R = \exp(O(k \cdot (2^{-n} + \|C\|^2/\sigma_{m'}(S)^2)))$.*

The main idea of the proof of Lemma 10, given in the full version, is to apply Lemma 5 with P, P' being the (k, S)-LWE and (k, S, C)-LWE problems respectively, which have instances of the form $x = (r, \boldsymbol{y})$, where $r = (A, \boldsymbol{u}, \{\boldsymbol{x}_i\}_{i\le k})$ and the hints $\boldsymbol{x}_i$ for $i \le k$ sampled from either the zero-centered distribution $\hookleftarrow D_{\Lambda^{\perp}_{-\boldsymbol{u}}(A),S,\boldsymbol{0}}$ (distribution Φ of r, in (k, S)-LWE) or the non-zero center distribution $\hookleftarrow D_{\Lambda^{\perp}_{-\boldsymbol{u}}(A),S,\boldsymbol{c}_i}$

(distribution $\varPhi'$ of r, in (k, S, C)-LWE), and $\boldsymbol{y} \in \mathbb{T}^{m+1}$ is a sample from either the distribution

$$D_0(r) = \frac{1}{q} \cdot U\Big(\mathrm{Im}\Big(\frac{\boldsymbol{u}^t}{A}\Big)\Big) + \nu_\alpha^{m+1}$$

or the distribution

$$D_1(r) = \frac{1}{q} \cdot U\Big(\mathrm{Span}_{i \leq k}\Big(\frac{1}{\boldsymbol{x}_i}\Big)^{\perp}\Big) + \nu_\alpha^{m+1}.$$

Given $x = (r, \boldsymbol{y})$, is possible to efficiently sample $\boldsymbol{y}'$ from either $D_0(r)$ or $D_1(r)$, so the public-samplability property assumed by Lemma 5 is satisfied. This Lemma gives the desired reduction between (k, S)-LWE and (k, S, C)-LWE, as long as the RD $R(\varPhi \| \varPhi')$ between the distribution of r in the two problems is polynomially bounded. The latter reduces to obtaining a bound on the RD between a Gaussian distribution and a small offset thereof, which is given by Lemma 4.

In our application of Lemma 10, the (k, S, C)-LWE problem resulting from the first subreduction has $\|C\| = 1$, and $\sigma_{m'}(S) = \sigma$, so that $R = \exp(O(k \cdot (2^{-n} + 1/\sigma^2))) = O(1)$ using $\sigma = \varOmega(n)$ and $k \leq n$. This shows that the second subreduction is probabilistic polynomial time. □

Our technique can be applied to improve the Boneh-Freeman reduction from SIS to k-SIS, from an exponential loss in k to a polynomial loss in k. In fact, we map A to A'' in the same way (except that we do not use and add $\boldsymbol{u}$ on top of the matrix A) and then also use the top k rows of $(\overline{X}_1 | \overline{X}_2)$ as the k-SIS hints for the new matrix A''. Then, whenever the adversary can output a short vector $\boldsymbol{x_1} \| \boldsymbol{x_2}$ that is orthogonal to A'', we can also output a short vector $(\boldsymbol{x_1} - \boldsymbol{x_2} \cdot \overline{X}_2^{-1} \overline{X}_1)$ which is orthogonal to A. As the rows of $\overline{X}_1$ are distributed as independent Gaussian samples and the adversary is only given its first k rows, it can be shown that, if $\boldsymbol{x_1} \| \boldsymbol{x_2}$ is linearly independent from the k-SIS hints, then the vector $(\boldsymbol{x_1} - \boldsymbol{x_2} \cdot \overline{X}_2^{-1} \overline{X}_1)$ is null with a negligible probability. RD may also be used to reduce k-SIS with non-zero-centered hints (with small centers) to k-SIS with zero-centered hints.

4 A Lattice-Based Public-Key Traitor Tracing Scheme

In this section, we describe and analyze our basic traitor tracing scheme. First, we give the underlying multi-user public-key encryption scheme. We then explain how to implement black-box confirmation tracing.

4.1 A Multi-user Encryption Scheme

The scheme is designed for a given security parameter n, a number of users N and a maximum malicious coalition size t. It then involves several parameters q, m, α, S. These are set so that the scheme is correct (decryption works properly on honestly generated ciphertexts) and secure (semantically secure encryption and possibility to trace members of malicious coalitions). In particular, we define S

as $\mathrm{Diag}(\sigma, \dots, \sigma, \sigma', \dots, \sigma') \in \mathbb{R}^{m \times m}$ where $\sigma' > \sigma$ and their respective numbers of iterations are set so that (t, S)-$\mathrm{LWE}_{m+1,\alpha}$ is hard to solve.

`Setup`. The trusted authority generates a master key pair using the algorithm from Lemma 2. Let $(A, T) \in \mathbb{Z}_q^{m \times n} \times \mathbb{Z}^{m \times m}$ be the output. We additionally sample $\boldsymbol{u}$ uniformly in $\mathbb{Z}_q^n$. Matrix T will be part of the tracing key tk, whereas the public key is $pk = A^+$, with $A^+ = (\boldsymbol{u}^t \| A)$.

Each user $\mathcal{U}_i$ for $i \leq N$ obtains a secret key sk_i from the trusted authority, as follows. The authority executes the GPV algorithm using the basis of $\Lambda^\perp(A)$ consisting of the rows of T, and the standard deviation matrix S. The authority obtains a sample $\boldsymbol{x}_i$ from $D_{\Lambda^\perp_{-\boldsymbol{u}}(A), S}$. The standard deviations $\sigma' > \sigma$ may be chosen as small as $3mq^{n/m}\sqrt{(2m+4)/\pi}$. The user secret key is $\boldsymbol{x}_i^+ = (1 \| \boldsymbol{x}_i) \in \mathbb{Z}^{m+1}$. Using the Gaussian tail bound and the union bound, we have $\|\boldsymbol{x}_i\| \leq \sqrt{m}\sigma'$ for all $i \leq N$, with probability $\geq 1 - N \cdot 2^{-\Omega(m)}$.

The tracing key tk consists of the matrix T and all pairs $(\mathcal{U}_i, sk_i)$.

`Encrypt`. The encryption algorithm is exactly the 1-bit encryption scheme from [19, Se. 7.1], which we recall, for readability.[1] The plaintext and ciphertext domains are $\mathcal{P} = \{0, 1\}$ and $\mathcal{C} = \mathbb{Z}_q^{m+1}$ respectively, and:

$$\mathtt{Enc} : M \mapsto \left[\frac{\boldsymbol{u}^t}{A}\right] \cdot \boldsymbol{s} + \boldsymbol{e} + \left[\frac{M \cdot \lfloor q/2 \rfloor}{\boldsymbol{0}}\right], \text{ where } \boldsymbol{s} \hookleftarrow U(\mathbb{Z}_q^n) \text{ and } \boldsymbol{e} \hookleftarrow \lfloor \nu_{\alpha q} \rceil^{m+1}.$$

As explained in [19], this scheme is semantically secure under chosen plaintext attacks (IND-CPA), under the assumption that $\mathrm{LWE}_{m+1,\alpha}$ is hard to solve.

`Decrypt`. To decrypt a ciphertext $\boldsymbol{c} \in \mathbb{Z}_q^{m+1}$, user $\mathcal{U}_i$ uses its secret key $\boldsymbol{x}_i^+$ and evaluates the following function `Dec` from $\mathbb{Z}_q^{m+1}$ to $\{0, 1\}$: Map $\boldsymbol{c}$ to 0 if $\langle \boldsymbol{x}_i^+, \boldsymbol{c} \rangle \bmod q$ is closer to 0 than $\pm \lfloor q/2 \rfloor$.

If $\boldsymbol{c}$ is an honestly generated ciphertext of a plaintext $M \in \{0, 1\}$, we have $\langle \boldsymbol{x}_i^+, \boldsymbol{c} \rangle = \langle \boldsymbol{x}_i^+, \boldsymbol{e} \rangle + M \cdot \lfloor q/2 \rfloor \bmod q$, where $\boldsymbol{e} \hookleftarrow \lfloor \nu_{\alpha q} \rceil^{m+1}$. It can be shown that the latter has magnitude $\leq 2\sqrt{m}\alpha q \|\boldsymbol{x}_i^+\|$ with probability $1 - 2^{-\Omega(n)}$ over the randomness of $\boldsymbol{e}$. This is $\leq 3m\alpha q \sigma'$ for all i, with probability $\geq 1 - N \cdot 2^{-\Omega(n)}$. To ensure the correctness of the scheme, it suffices to set $q \geq 4m\alpha q \sigma'$. Note that other constraints will be added to enable tracing.

Theorem 4. *Let m, n, q, N be integers such that q is prime and $N \leq 2^{o(n)}$. Let $\alpha, \sigma, \sigma' > 0$ such that $\sigma' \geq \sigma \geq \Omega(mq^{n/m}\sqrt{\log m})$ and $\alpha \leq 1/(4m\sigma')$. Then the scheme described above is IND-CPA under the assumption that $\mathrm{LWE}_{m+1,\alpha}$ is hard. Further, the decryption algorithm is correct:*

$$\forall M \in \{0, 1\}, \forall i \leq N : \mathtt{Dec}\left(\mathtt{Enc}(M, pk), sk_i\right) = M$$

holds with probability $\geq 1 - 2^{-\Omega(n)}$ over the randomness used in `Setup` *and* `Enc`.

[1] As usual, the encryption algorithm may be used to encapsulate session keys which are then fed into an efficient data encapsulation mechanism to encrypt the data.

4.2 Tracing Traitors

We now present a black-box confirmation algorithm `Trace`.[2] It is given access to an oracle $\mathcal{O}^{\mathcal{D}}$ that provides black-box access to a decryption device $\mathcal{D}$. It takes as inputs the tracing key $tk = (T, (\mathcal{U}_i, \boldsymbol{x}_i^+)_{i \le N})$ and a set of suspect users $\{\mathcal{U}_{i_1}, \ldots, \mathcal{U}_{i_k}\}$ of cardinality $k \le t$, where t is the a priori bound on any coalition size. Wlog, we may consider that $k = t$ and $i_j = j$ for all $j \le k$.

Algorithm `Trace` gathers information about which keys have been used to build decoder $\mathcal{D}$, by feeding different carefully designed distributions to oracle $\mathcal{O}^{\mathcal{D}}$. We consider the following $t+1$ distributions $Tr_0, \ldots, Tr_t$ over $\mathcal{C} = \mathbb{Z}_q^{m+1}$:

$$Tr_i = U\left(\mathrm{Span}(\boldsymbol{x}_1^+, \ldots, \boldsymbol{x}_i^+)^{\perp}\right) + \lfloor \nu_{\alpha q} \rceil^{m+1}.$$

The first distribution Tr_0 is the uniform distribution, whereas the last distribution Tr_t is meant to be computationally indistinguishable from $\mathtt{Enc}(0)$. We define p_∞ as the probability $\Pr[\mathcal{O}^{\mathcal{D}}(\boldsymbol{c}, M) = 1]$ that the decoder can decrypt the ciphertexts, over the randomness of $M \hookleftarrow U(\{0,1\})$ and $\boldsymbol{c} \hookleftarrow \mathtt{Enc}(M)$. We define p_i as the probability the decoder decrypts the signals in Tr_i, for $i \in [0, t]$:

$$p_i = \Pr_{\substack{\boldsymbol{c} \hookleftarrow Tr_i \\ M \hookleftarrow U(\{0,1\})}} \left[\mathcal{O}^{\mathcal{D}}\left(\boldsymbol{c} + \begin{bmatrix} M \cdot \lfloor q/2 \rfloor \\ \mathbf{0} \end{bmatrix}, M\right) = 1\right].$$

A gap between p_{i-1} and p_i is meant to indicate that $\mathcal{U}_i$ is a traitor.

The confirmation and soundness properties are proved in the full version. We now concentrate on a new feature of our scheme: public traceability.

5 Projective Sampling and Public Traceability

We now modify the scheme of the Section 4 so that the tracing signals can be publicly sampled. For this purpose, we introduce the concept of projective sampling family.

5.1 Projective Sampling

Inspired from the notion of projective hash family [16], we propose the notion of projective sampling family in which each sampling function is keyed and, with a projected key, one can simulate the sampling function in a computationally indistinguishable way. Let X be a finite non-empty set. Let $F = (\mathtt{F}_k)_{k \in K}$ be a collection of sampling functions indexed by K, so that $\mathtt{F}_k$ is a sampling function over X, for every $k \in K$. We call $\mathtt{Sam} = (F, K, X)$ a sampling family. We now introduce the concept of projective sampling.

Definition 2 (Projective Sampling). *Let* $\mathtt{Sam} = (F, K, X)$ *be a sampling family. Let* J *be a finite, non-empty set, and let* $\pi : K \to J$ *be a (probabilistic) function. Let also*

[2] Note that in our context, minimal access is equivalent to standard access: since the plaintext domain is small, plaintext messages can be tested exhaustively.

$\mathtt{P} = (\mathtt{P}_j)_{j\in J}$ be a collection of sampling functions over X, and D be a distribution over K. Then $\mathtt{PSam} = (F, K, X, \mathtt{P}, J, \pi, D)$ is called a projective sampling family if, with overwhelming probability over the choice of $k, k' \hookleftarrow D$, and given the secret key k and its projected key $\pi(k)$, 1) the distributions obtained using $\mathtt{F}_k$ and $\mathtt{P}_{\pi(k)}$ are computationally indistinguishable, and 2) the distributions obtained using $\mathtt{F}_k$ and $\mathtt{P}_{\pi(k')}$ can be efficiently distinguished.

The first condition means that for $k \hookleftarrow D$, the value $\pi(k)$ "encodes" the sampling distribution of $\mathtt{F}_k$, so that when $\pi(k)$ is made public, the sampled signal $\mathtt{F}_k$ can be publicly simulated by $\mathtt{P}_{\pi(k)}$. The security requirement is very strong because the adversary is not only given the projected key, as in projective hashing, but also the secret key k. We require that sampling signals from the secret key and from its projected key are indistinguishable for the insiders who know the secret key. This is relevant for traitor tracing, as the traitors are system insiders and they possess secret data. The second condition (that we actually do not directly use in our cryptographic application) allows to prevent the trivial solution consisting in setting $\mathtt{P}_{\pi(k)}$ as an efficient sampling function that is independent of k: the simulation signal $\mathtt{P}_{\pi(k)}$ must be specific to k.[3]

5.2 Projective Sampling from k-LWE

We construct a set of projective sampling families $(\mathtt{PSam}_i)_{0\le i\le t}$. The parameters are almost identical to the parameters in the $\mathtt{Setup}$ of the multi-user scheme of Section 4. A further difference, required for simulation purposes in the security proof, is that $\sigma' > \sigma$ must be set $\widetilde{\Omega}(\sqrt{mn} + \pi q)$.

We let $A \hookleftarrow U(\mathbb{Z}_q^{m\times n})$ and $\boldsymbol{u} \hookleftarrow U(\mathbb{Z}_q^n)$ be public parameters. For each i, we define $K_i = (\mathbb{Z}_q^m)^i$ and D_i as the distribution on K_i that samples $k = (\boldsymbol{x}_j)_{j\le i}$ with $\boldsymbol{x}_j \hookleftarrow D_{\Lambda^\perp_{-\boldsymbol{u}}(A),\sigma}$ for all $j \le i$. The sampling function $\mathtt{F}_{i,k}$ is defined as $U(\mathrm{Span}_{j\le i}(\boldsymbol{x}_j^+)^\perp) + \lfloor \nu_{\alpha q} \rceil^{m+1}$. The projected key $\pi_i(k)$ is defined as follows:

- Sample $H \in \mathbb{Z}_q^{m\times(m-n)}$ uniformly, conditioned on $\mathrm{Im}(A) \subseteq \mathrm{Im}(H)$.
- For each $j \le i$, define $\boldsymbol{h}_j^t = -\boldsymbol{x}_j^t \cdot H$.
- Finally, set $J = \mathbb{Z}_q^{m\times(m-n)} \times (\mathbb{Z}_q^{m-n})^i$ and set $\pi_i(k) = (H, (\boldsymbol{h}_j)_{j\le i})$.

We now define the sampling $\mathtt{P}_{i,\pi_i(k)}$ with projected key $\pi_i(k) = (H, (\boldsymbol{h}_j)_{j\le i})$, as follows:

- Set $H_j = (\boldsymbol{h}_j^t \| H) \in \mathbb{Z}_q^{(m+1)\times(m-n)}$. We have $\boldsymbol{x}_j^{+t} \cdot H_j = \mathbf{0}$ and $\mathrm{Im}(A^+) \subseteq \mathrm{Im}(H_j)$.
- Set $\mathtt{P}_{i,\pi_i(k)} = U\left(\cap_{j\le i}\mathrm{Im}(H_j)\right) + \lfloor \nu_{\alpha q} \rceil^{m+1}$, with $\cap_{j\le 0}\mathrm{Im}(H_j) = \mathbb{Z}_q^{m+1}$ by convention. Note that $\cap_{j\le i}\mathrm{Im}(H_j) \subseteq \mathrm{Span}_{j\le i}(\boldsymbol{x}_j^+)^\perp$.

Theorem 5. *For each $i = 0, \ldots, t$, $\mathtt{PSam}_i$ is a projective sampling family. Concretely, under the (i, S)-LWE$_{\alpha,m}$ hardness assumptions, given the uniformly sampled public parameters $(A, \boldsymbol{u})$, the secret key $k = (\boldsymbol{x}_j)_{j\le i} \hookleftarrow D_i$ and its projected key $\pi_i(k) = (H, (\boldsymbol{h}_j)_{j\le i})$, the distributions $\mathtt{F}_{i,k}$ and $\mathtt{P}_{i,\pi_i(k)}$ are indistinguishable. Moreover, they are both indistinguishable from $U(\mathrm{Im}(A^+)) + \lfloor \nu_{\alpha q} \rceil^{m+1}$. Finally, with overwhelming*

[3] Another trivial situation occurs when $\pi(k) = k$: the projected key leaks the full information about the original key and one cannot safely publish the projected key.

probability, the distributions $\mathtt{F}_{i,k}$ *and* $\mathtt{P}_{i,\pi_i(k')}$ *can be efficiently distinguished, when* k' *is independently sampled from* D_i.

Proof. For the last statement, observe that with overwhelming probability, the secret key k' contains an $\boldsymbol{x}'_j \in \mathbb{Z}_q^m$ that does not belong to $\mathrm{Span}_{j\leq i}(\boldsymbol{x}_j)$ (by Lemma 3). In that case, taking the inner product of all $\boldsymbol{x}'_j$'s of k' with a sample from $\mathtt{P}_{i,\pi_i(k')}$ gives small residues modulo q, whereas one of the inner products of the $\boldsymbol{x}'_j$'s with a sample from with a sample from $\mathtt{F}_{i,k}$ will be uniform modulo q.

We now consider the first statement. From the hardness of (i, S)-$\mathrm{LWE}_{m,\alpha}$, given k, the distributions

$$\mathtt{F}_{i,k} = U(\mathrm{Span}_{j\leq i}(\boldsymbol{x}_j^+)^\perp) + \lfloor \nu_{\alpha q} \rceil^{m+1} \quad \text{and} \quad U(\mathrm{Im}(A^+)) + \lfloor \nu_{\alpha q} \rceil^{m+1}$$

are indistinguishable. Further, given $k = (\boldsymbol{x}_j)_{j\leq i}$, the projected key $\pi_i(k) = (H, (\boldsymbol{h}_j)_{j\leq i})$ can be sampled from D_i. Therefore, given both k and $\pi_i(k)$, the distributions $\mathtt{F}_{i,k}$ and $U(\mathrm{Im}(A^+)) + \lfloor \nu_{\alpha q} \rceil^{m+1}$ remain indistinguishable.

Now, we have $\mathrm{Im}(A^+) \subseteq \cap_{j\leq i}\mathrm{Im}(H_j) \subseteq (\mathrm{Span}_{j\leq i}(\boldsymbol{x}_j^+))^\perp$. Hence:

$$\begin{aligned} U(\mathrm{Im}(A^+)) + U(\cap_{j\leq i}\mathrm{Im}(H_j)) &= U(\cap_{j\leq i}\mathrm{Im}(H_j)), \\ U(\mathrm{Span}_{j\leq i}(\boldsymbol{x}_j^+)^\perp) + U(\cap_{j\leq i}\mathrm{Im}(H_j)) &= U(\mathrm{Span}_{j\leq i}(\boldsymbol{x}_j^+)^\perp). \end{aligned}$$

We note that given $\boldsymbol{h}_1, \ldots, \boldsymbol{h}_i$, one can efficiently sample from $U(\cap_{j\leq i}\mathrm{Im}(H_j))$. Therefore, under the hardness of (i, S)-$\mathrm{LWE}_{m,\alpha}$, this shows that $\mathtt{F}_{i,k}$, $\mathtt{P}_{i,\pi_i(k)}$ and $U(\mathrm{Im}(A^+)) + \lfloor \nu_{\alpha q} \rceil^{m+1}$ are indistinguishable. □

5.3 Public Traceability from Projective Sampling

In the scheme of Section 4, the tracing key $tk = (T, (\mathcal{U}_i, \boldsymbol{x}_i)_{i\leq N})$ must be kept secret, as it would reveal the secret keys of the users. The tracing signals are samples from $U(\mathrm{Span}_{j\leq i}(\boldsymbol{x}_j^+)^\perp) + \lfloor \nu_{\alpha q} \rceil^{m+1}$, which exactly matches $\mathtt{F}_{i,k}$. By publishing the projected key $\pi_i(k)$, anyone can use the projective sampling $\mathtt{P}_{i,\pi_i(k)}$: by Theorem 5, given $(k, \pi_i(k))$, $\mathtt{F}_{i,k}$ and $\mathtt{P}_{i,\pi_i(k)}$ are indistinguishable and they are both indistinguishable from the original sampling $U(\mathrm{Im}(A^+)) + \lfloor \nu_{\alpha q} \rceil^{m+1}$. We are thus almost done with public traceability.

However, a subtle point is that we have to use all the projective samplings $(\mathtt{P}_{i,\pi_i(k)})$ for transforming the secret tracing to the public tracing: all the projected keys $(\boldsymbol{h}_j)_{j\leq N}$ should be published. Because the keys k in $\mathtt{F}_{i,k}$ are not independent, it could occur that the adversary exploits a projected key $\pi_i(k)$ for distinguishing $\mathtt{P}_{i',\pi_{i'}(k')}$ from the original signals. To handle this, we prove that, given $(\boldsymbol{x}_j)_{j\leq i}$ and all the keys $(\boldsymbol{h}_j)_{j\leq N}$, the adversary cannot distinguish $\mathtt{P}_{i,\pi_i(k)}$ from the original signals. For this purpose, we exploit a technique from [20] to simulate $(\boldsymbol{h}_j)_{i<j\leq N}$ from the public information.

Theorem 6. *Set* $i \leq t$. *Under the* (i, S)-$\mathrm{LWE}_{\alpha,m}$ *and the* $\mathrm{LWE}'_{\alpha,m}$ *hardness assumptions, given the secret key* $k = (\boldsymbol{x}_j)_{j\leq i}$ *and the projected keys* $(H, (\boldsymbol{h}_j)_{j\leq N})$, *the following two distributions are indistinguishable*

$$\mathtt{P}_{i,\alpha(k)} = U(\cap_{j\leq i}\mathrm{Im}(H_j)) + \lfloor \nu_{\alpha q} \rceil^{m+1} \quad \textit{and} \quad U(\mathrm{Im}(A^+)) + \lfloor \nu_{\alpha q} \rceil^{m+1}.$$

Proof. Assume a ppt attacker is given $(\boldsymbol{x}_j)_{j\leq i}$ (with the $\boldsymbol{x}_j$'s independently sampled from $D_{A^{\perp}_{-\boldsymbol{u}}(A),\sigma}$) and all the projected keys $(\boldsymbol{h}_j)_{j\leq N}))$. We are to prove that, under the (i,S)-$\mathrm{LWE}_{\alpha,m}$ and $\mathrm{LWE}'_{\alpha,m}$ hardness assumptions, it cannot distinguish between the distributions (over $\mathbb{Z}_q^{m+1}$)

$$U(\mathrm{Im}(A^{+})) + \lfloor \nu_{\alpha q} \rceil^{m+1} \quad \text{and} \quad \mathtt{P}_{i,\pi_i(k)} = U(\cap_{j\leq i}\mathrm{Im}(H_j)) + \lfloor \nu_{\alpha q} \rceil^{m+1}.$$

We proceed by a sequence of games.

Game$_0$: This is the above distinguishing game. We let ε_0 denote the adversary's distinguishing advantage. The goal is to show that ε_0 is negligible.

Game$_1$: In this second game, we sample $\boldsymbol{x}_1,\ldots,\boldsymbol{x}_i$ from $D_{A^{\perp}_{-\boldsymbol{u}}(A),\sigma}$ as in **Game**$_0$, but the $\boldsymbol{x}_j$'s for $j > i$ are sampled uniformly in $\mathbb{Z}_q^n$, conditioned on $\boldsymbol{x}_j^t \cdot A = -\boldsymbol{u}^t$. The $\boldsymbol{h}_j$'s for $j > i$ are modified accordingly, but the rest is as in **Game**$_0$. We let ε_1 denote the adversary's distinguishing advantage.

The main point is that in **Game**$_1$, no secret information is required for sampling the projected keys $\boldsymbol{h}_j$'s for $j > i$. The proof of the following lemma may be found in the full version.

Lemma 11. *Under the* $\mathrm{LWE}'_{\alpha,m}$ *hardness assumption, the quantity* $|\varepsilon_1 - \varepsilon_0|$ *is negligible.*

We note that, in **Game**$_1$, the $\boldsymbol{h}_j$'s can be sampled publicly from the available data. Therefore, from Theorem 5, under the (i,S)-$\mathrm{LWE}_{\alpha,m}$ hardness assumptions, the advantage ε_1 is negligible. □

Semantic security of the updated scheme. We modify the public information of the scheme of Section 4, so that we can use the set of projective sampling families described above. For this aim, we simply add the projected key $(H, (\boldsymbol{h}_i)_{i\leq N})$ to the public key. The scheme becomes publicly traceable because the tracing signals can be sampled from the projected keys, as explained above. Finally, as the public key has been modified, we should prove that the knowledge of these projected keys provides no significant advantage for an adversary towards breaking the semantic security of the encryption scheme. Fortunately, the semantic security directly follows from Theorem 6, for the particular case of $i = 0$.

Acknowledgements. We thank M. Abdalla, D. Augot, R. Bhattacharrya, L. Ducas, V. Guleria, G. Hanrot, F. Laguillaumie, K. T. T. Nguyen, G. Quintin, O. Regev, H. Wang for helpful discussions. The authors were partly supported by the LaBaCry MERLION grant, the Australian Research Council Discovery Grant DP110100628, the ANR-09-VERSO-016 BEST and ANR-12-JS02-0004 ROMAnTIC Projects, the INRIA invited researcher scheme, the Singapore National Research Foundation Research Grant NRF-CRP2-2007-03, the Singapore MOE Tier 2 research grant MOE2013-T2-1-041, the LIA Formath Vietnam and the ERC Starting Grant ERC-2013-StG-335086-LATTAC.

References

1. Aggarwal, D., Regev, O.: A note on discrete gaussian combinations of lattice vectors (2013), Draft Available at, `http://arxiv.org/pdf/1308.2405v1.pdf`
2. Agrawal, S., Gentry, C., Halevi, S., Sahai, A.: Discrete gaussian leftover hash lemma over infinite domains. In: Sako, K., Sarkar, P. (eds.) ASIACRYPT 2013, Part I. LNCS, vol. 8269, pp. 97–116. Springer, Heidelberg (2013)
3. Ajtai, M.: Generating hard instances of lattice problems (extended abstract). In: Proc. of STOC, pp. 99–108. ACM (1996)
4. Ajtai, M.: Generating hard instances of the short basis problem. In: Wiedermann, J., Van Emde Boas, P., Nielsen, M. (eds.) ICALP 1999. LNCS, vol. 1644, pp. 1–9. Springer, Heidelberg (1999)
5. Alwen, J., Peikert, C.: Generating shorter bases for hard random lattices. Theor. Comput. Science 48(3), 535–553 (2011)
6. Billet, O., Phan, D.H.: Efficient Traitor Tracing from Collusion Secure Codes. In: Safavi-Naini, R. (ed.) ICITS 2008. LNCS, vol. 5155, pp. 171–182. Springer, Heidelberg (2008)
7. Boneh, D., Franklin, M.K.: An efficient public key traitor scheme (Extended abstract). In: Wiener, M. (ed.) CRYPTO 1999. LNCS, vol. 1666, pp. 338–353. Springer, Heidelberg (1999)
8. Boneh, D., Freeman, D.M.: Linearly homomorphic signatures over binary fields and new tools for lattice-based signatures. In: Catalano, D., Fazio, N., Gennaro, R., Nicolosi, A. (eds.) PKC 2011. LNCS, vol. 6571, pp. 1–16. Springer, Heidelberg (2011), Full version available at, `http://eprint.iacr.org/2010/453`
9. Boneh, D., Waters, B.: A fully collusion resistant broadcast, trace, and revoke system. In: Proc. of ACM CCS, pp. 211–220. ACM (2006)
10. Boneh, D., Naor, M.: Traitor tracing with constant size ciphertext. In: Ning, P., Syverson, P.F., Jha, S. (eds.) ACM CCS 2008, pp. 501–510. ACM Press (2008)
11. Boneh, D., Sahai, A., Waters, B.: Fully collusion resistant traitor tracing with short ciphertexts and private keys. In: Vaudenay, S. (ed.) EUROCRYPT 2006. LNCS, vol. 4004, pp. 573–592. Springer, Heidelberg (2006)
12. Boneh, D., Zhandry, M.: Multiparty key exchange, efficient traitor tracing, and more from indistinguishability obfuscation. Cryptology ePrint Archive, Report 2013/642 (2013), `http://eprint.iacr.org/`
13. Brakerski, Z., Langlois, A., Peikert, C., Regev, O., Stehlé, D.: Classical hardness of learning with errors. In: STOC, pp. 575–584. ACM (2013)
14. Chabanne, H., Phan, D.H., Pointcheval, D.: Public traceability in traitor tracing schemes. In: Cramer, R. (ed.) EUROCRYPT 2005. LNCS, vol. 3494, pp. 542–558. Springer, Heidelberg (2005)
15. Chor, B., Fiat, A., Naor, M.: Tracing traitors. In: Desmedt, Y.G. (ed.) CRYPTO 1994. LNCS, vol. 839, pp. 257–270. Springer, Heidelberg (1994)
16. Cramer, R., Shoup, V.: Universal hash proofs and a paradigm for adaptive chosen ciphertext secure public-key encryption. In: Knudsen, L.R. (ed.) EUROCRYPT 2002. LNCS, vol. 2332, pp. 45–64. Springer, Heidelberg (2002)
17. Garg, S., Gentry, C., Halevi, S.: Candidate multilinear maps from ideal lattices. In: Johansson, T., Nguyen, P.Q. (eds.) EUROCRYPT 2013. LNCS, vol. 7881, pp. 1–17. Springer, Heidelberg (2013)
18. Garg, S., Gentry, C., Halevi, S., Raykova, M., Sahai, A., Waters, B.: Candidate indistinguishability obfuscation and functional encryption for all circuits. In: Proc. of FOCS, pp. 40–49. IEEE Computer Society Press (2013)
19. Gentry, C., Peikert, C., Vaikuntanathan, V.: Trapdoors for hard lattices and new cryptographic constructions. In: Proc. of STOC, pp. 197–206. ACM (2008), Full version available at, `http://eprint.iacr.org/2007/432.pdf`

20. Gordon, S.D., Katz, J., Vaikuntanathan, V.: A group signature scheme from lattice assumptions. In: Abe, M. (ed.) ASIACRYPT 2010. LNCS, vol. 6477, pp. 395–412. Springer, Heidelberg (2010)
21. Kiayias, A., Pehlivanglu, S.: Encryption For Digital Content. Springer, Heidelberg (2010)
22. Kiayias, A., Yung, M.: Breaking and repairing asymmetric public-key traitor tracing. In: Digital Rights Management Workshop, pp. 32–50 (2002)
23. Kiayias, A., Yung, M.: Traitor tracing with constant transmission rate. In: Knudsen, L.R. (ed.) EUROCRYPT 2002. LNCS, vol. 2332, pp. 450–465. Springer, Heidelberg (2002)
24. Komaki, H., Watanabe, Y., Hanaoka, G., Imai, H.: Efficient asymmetric self-enforcement scheme with public traceability. In: Kim, K.-c. (ed.) PKC 2001. LNCS, vol. 1992, pp. 225–239. Springer, Heidelberg (2001)
25. Langlois, A., Stehlé, D., Steinfeld, R.: GGHLite: More efficient multilinear maps from ideal lattices. In: Nguyen, P.Q., Oswald, E. (eds.) EUROCRYPT 2014. LNCS, vol. 8441, pp. 239–256. Springer, Heidelberg (2014)
26. Langlois, A., Stehlé, D., Steinfeld, R.: Improved and simplified security proofs in lattice-based cryptography: using the Rényi divergence rather than the statistical distance (2014); Available on the webpages of the authors.
27. Lyubashevsky, V., Peikert, C., Regev, O.: On ideal lattices and learning with errors over rings. J. ACM 60(6), 43 (2013)
28. Micciancio, D., Peikert, C.: Trapdoors for lattices: Simpler, tighter, faster, smaller. In: Pointcheval, D., Johansson, T. (eds.) EUROCRYPT 2012. LNCS, vol. 7237, pp. 700–718. Springer, Heidelberg (2012)
29. Micciancio, D., Regev, O.: Worst-case to average-case reductions based on gaussian measures. SIAM J. Comput 37(1), 267–302 (2007)
30. Micciancio, D., Regev, O.: Lattice-based cryptography. In: Bernstein, D.J., Buchmann, J., Dahmen, E. (eds.) Post-Quantum Cryptography, pp. 147–191. Springer, Heidelberg (2009)
31. Naor, M., Pinkas, B.: Efficient trace and revoke schemes. In: Frankel, Y. (ed.) FC 2000. LNCS, vol. 1962, pp. 1–20. Springer, Heidelberg (2001)
32. O'Neill, A., Peikert, C., Waters, B.: Bi-deniable public-key encryption. In: Rogaway, P. (ed.) CRYPTO 2011. LNCS, vol. 6841, pp. 525–542. Springer, Heidelberg (2011)
33. Peikert, C.: Public-key cryptosystems from the worst-case shortest vector problem. In: Proc. of STOC, pp. 333–342. ACM (2009)
34. Peikert, C.: An efficient and parallel gaussian sampler for lattices. In: Rabin, T. (ed.) CRYPTO 2010. LNCS, vol. 6223, pp. 80–97. Springer, Heidelberg (2010)
35. Pfitzmann, B.: Trials of traced traitors. In: Anderson, R. (ed.) IH 1996. LNCS, vol. 1174, pp. 49–64. Springer, Heidelberg (1996)
36. Pfitzmann, B., Waidner, M.: Asymmetric fingerprinting for larger collusions. In: ACM CCS 1997, pp. 151–160. ACM Press (April 1997)
37. Phan, D.H., Safavi-Naini, R., Tonien, D.: Generic construction of hybrid public key traitor tracing with full-public-traceability. In: Bugliesi, M., Preneel, B., Sassone, V., Wegener, I. (eds.) ICALP 2006. LNCS, vol. 4052, pp. 264–275. Springer, Heidelberg (2006)
38. Regev, O.: On lattices, learning with errors, random linear codes, and cryptography. In: Proc. of STOC, pp. 84–93. ACM (2005)
39. Regev, O.: On lattices, learning with errors, random linear codes, and cryptography. J. ACM 56(6) (2009)
40. Regev, O.: The learning with errors problem. In: Invited survey in CCC 2010 (2010), http://www.cims.nyu.edu/~regev/
41. Sirvent, T.: Traitor tracing scheme with constant ciphertext rate against powerful pirates. In: Augot, D., Sendrier, N., Tillich, J.-P. (eds.) Workshop on Coding and Cryptography—WCC 2007, pp. 379–388 (April 2007)
42. Watanabe, Y., Hanaoka, G., Imai, H.: Efficient asymmetric public-key traitor tracing without trusted agents. In: Naccache, D. (ed.) CT-RSA 2001. LNCS, vol. 2020, pp. 392–407. Springer, Heidelberg (2001)