Cryptanalysis of the Multilinear Map over the Integers

Jung Hee Cheon[1], Kyoohyung Han[1], Changmin Lee[1], Hansol Ryu[1], Damien Stehlé[2]

[1] Seoul National University (SNU), Republic of Korea
[2] ENS de Lyon, Laboratoire LIP (U. Lyon, CNRS, ENSL, INRIA, UCBL), France.

Abstract. We describe a polynomial-time cryptanalysis of the (approximate) multilinear map of Coron, Lepoint and Tibouchi (CLT). The attack relies on an adaptation of the so-called *zeroizing* attack against the Garg, Gentry and Halevi (GGH) candidate multilinear map. Zeroizing is much more devastating for CLT than for GGH. In the case of GGH, it allows to break generalizations of the Decision Linear and Subgroup Membership problems from pairing-based cryptography. For CLT, this leads to a total break: all quantities meant to be kept secret can be efficiently and publicly recovered.

Keywords: Multilinear maps, graded encoding schemes.[3]

1 Introduction

Cryptographic bilinear maps, made possible thanks to pairings over elliptic curves, have led to a bounty of exciting cryptographic applications. In 2002, Boneh and Silverberg [BS02] formalized the concept of cryptographic multilinear maps and provided two applications: a one-round key multi-party exchange protocol, and a very efficient broadcast encryption scheme. But these promising applications were only day-dreaming exercises, as no realization of such multilinear maps was known. This was changed about ten years later, as Garg, Gentry and Halevi proposed the first approximation to multilinear maps [GGH13a]. They introduced the concept of (approximate) graded encoding scheme as a variant of multilinear maps, and described a candidate construction relying on ideal lattices (which we will refer to as GGH in this work). Soon after, Coron, Lepoint and Tibouchi [CLT13] proposed another candidate construction of a graded encoding scheme, relying on a variant of the approximate greatest common divisor problem (CLT, for short).

The GGH and CLT constructions share similarities. Both are derived from a homomorphic encryption scheme (Gentry's scheme [Gen09] and the van Dijk *et al.* scheme [DGHV10], respectively). And both rely on some extra public data, called the zero-testing or extraction parameter, which allows to publicly decide whether the plaintext data hidden in a given encoding is zero, as long as the encoding is not the output of a too deep homomorphic evaluation circuit.

Graded encoding schemes serve as a basis to define presumably hard problems. These problems are then used as security foundations of cryptographic constructions. A major discrepancy between GGH and CLT is that some natural problems seem easy when instantiated with the GGH graded encoding scheme, and hard for CLT. Two such problems are subgroup membership (SubM) and decision linear (DLIN). Roughly speaking, SubM asks to distinguish between encodings of elements of a group and encodings of elements of a subgroup thereof. DLIN consists in determining whether a matrix of elements is singular, given as input encodings of those elements. Another similar discrepancy seems to exist between the asymmetric

[3] Date: November 5, 2014

variants of GGH and CLT: the External Decision Diffie-Hellman (XDH) problem seems hard for CLT but is easy for GGH. XDH is exactly DDH for one of the components of the asymmetric graded encoding scheme. These problems have been extensively used in the context of cryptographic bilinear maps [Sco02,BBS04,BGN05].

In the first public version of [GGH13a] (dated 29 Oct. 2012),[4] the GGH construction was thought to provide secure DLIN instantiation. It was soon realized that DLIN could be broken in polynomial-time. The attack consists in multiplying an encoding of some element m by an encoding of 0 and by the zero-testing parameter: this produces a small element (because the encoded value is $m \cdot 0 = 0$), which happens to be a multiple of m. This *zeroizing attack* (also called weak discrete logarithm attack) is dramatic for SubM, DLIN and XDH. Fortunately, it does not seem useful against other problems, such as Graded Decision Diffie Hellman (GDDH), the adaptation of DDH to the graded encoding scheme settnig. As no such attack was known for CLT, the presumed hardness of the CLT instantiations of SubM, DLIN and XDH was exploited as a security grounding for several cryptographic constructions [ABP14,Att14,BP13,BLMR13,GGHZ14a,GGHZ14b,GLW14,LMR14,Zha14].

Main result. We describe a zeroizing attack on the CLT graded encoding scheme. It runs in polynomial-time, and allows to publicly compute all the parameters of the CLT scheme that were supposed to be kept secret.

Impact of the attack. The CLT candidate construction should be considered broken, unless the low-level encodings of zero are not made public. At the moment, there does not remain any candidate multilinear map approximation for which any of SubM, DLIN and XDH is hard. Several recent cryptographic constructions cannot be realized anymore: this includes all constructions from [Att14,GGHZ14a,GGHZ14b,GLW14,Zha14], the GPAKE construction of [ABP14] for more than 3 users, one of the two constructions of password hashing of [BP13], the alternative key-homomorphic PRF construction from [BLMR13], and the use of the latter in [LMR14].

Our attack heavily relies on the fact that low-level encodings of zero are made publicly available. It is not applicable if these parameters are kept secret. They are used in applications to homomorphically re-randomize encodings, in order to "canonicalize" their distributions. Zeroizing attacks seem impossible to mount if those level-1 encodings of zero are not made public. This approach was used in [GGH$^+$13b] and [BR13], for example. It seems that this approach can be used to secure the constructions from [GLSW14] and [Zim14] as well.

Open problems. A natural line of research is to extend the range of applications of graded encoding schemes for which the encodings of zero are not needed.

Publishing encodings of zero as well as a zero-test parameter can lead to damaging consequences (total break of CLT, weakness of SubM, DLIN and XDH for GGH). An impossibility result would be fascinating.

Related work. A third construction of a variant of graded encoding schemes was recently proposed in [GGH14]. In that scheme, no encoding of zero is provided, as it would incur serious security issues (see [GGH14, Se. 4]).

Organization. In Section 2, we recall the CLT scheme and the zeroizing attack against GGH. In Section 3, we present our attack on CLT.

[4] It can be accessed from the IACR eprint server

2 Preliminaries

Notation. We use $a \leftarrow A$ to denote the operation of uniformly choosing an element a from a finite set A. We define $[n] = \{1, 2, \ldots, n\}$. We let $\mathbb{Z}_q$ denote the ring $\mathbb{Z}/(q\mathbb{Z})$. For pairwise coprime integers $p_1, p_2, \ldots, p_n$, we define $\mathsf{CRT}_{(p_1,p_2,\ldots,p_n)}(r_1, r_2, \ldots, r_n)$ (abbreviated as $\mathsf{CRT}_{(p_i)}(r_i)$) as the unique integer in $\left(-\frac{1}{2}\prod_{i=1}^{n} p_i, \frac{1}{2}\prod_{i=1}^{n} p_i\right]$ which is congruent to $r_i \bmod p_i$ for all $i \in [n]$.

We use lower-case bold letters to denote vectors whereas upper-case bold letters are used to denote matrices. For matrix $\boldsymbol{S}$, we denote by $\boldsymbol{S}^T$ the transpose of $\boldsymbol{S}$. We define $\|\boldsymbol{S}\|_\infty = \max_i \sum_{j\in[n]} |s_{ij}|$, where s_{ij} is the (i, j) component of $\boldsymbol{S}$. Finally we denote by $\mathsf{diag}(a_1, \ldots, a_n)$ the diagonal matrix with diagonal coefficients equal to $a_1, \ldots, a_n$.

2.1 A Candidate Multilinear Map over the Integers

First, we briefly recall the Coron *et al.* construction. We refer to the original paper [CLT13] for a complete description.

The scheme relies on the following parameters.

λ: the security parameter
κ: the multilinearity parameter
ρ: the bit length of the randomness used for encodings
α: the bit length of the message slots
η: the bit length of the secret primes p_i
n: the number of distinct secret primes
τ: the number of level-1 encodings of zero in public parameters
ℓ: the number of level-0 encodings in public parameters
ν: the bit length of the image of the multilinear map
β: the bit length of the entries of the zero-test matrix H

Coron *et al.* suggested to set the parameters so that the following conditions are met:

- $\rho = \Omega(\lambda)$: to avoid brute force attack (see also [LS14] for a constant factor improvement).
- $\alpha = \lambda$: so that the ring of messages $\mathbb{Z}_{g_1} \times \ldots \times \mathbb{Z}_{g_n}$ does not contain a small subring $\mathbb{Z}_{g_i}$.[5]
- $n = \Omega(\eta \cdot \lambda)$: to thwart lattice reduction attacks.
- $\ell \geq n \cdot \alpha + 2\lambda$: to be able to apply the leftover hash lemma from [CLT13, Le. 1].
- $\tau \geq n \cdot (\rho + \log_2(2n)) + 2\lambda$: to apply leftover hash lemma from [CLT13, Se. 4].
- $\beta = \Omega(\lambda)$: to avoid the so-called gcd attack.
- $\eta \geq \rho_\kappa + \alpha + 2\beta + \lambda + 8$, where ρ_κ is the maximum bit size of the random r_i's a level-κ encoding. When computing the product of κ level-1 encodings and an additional level-0 encoding, one obtains $\rho_\kappa = \kappa \cdot (2\alpha + 2\rho + \lambda + 2\log_2 n + 2) + \rho + \log_2 \ell + 1$.
- $\nu = \eta - \beta - \rho_f - \lambda - 3$: to ensure zero-test correctness.

[5] In fact, it seems that making the primes g_i public, equal, and $\Omega(\kappa)$ may not lead to any specific attack [CLT14].

Instance generation: $(\mathsf{params}, \boldsymbol{p}_{zt}) \leftarrow \mathsf{InstGen}(1^\lambda, 1^\kappa)$. Set the scheme parameters as explained above. For $i \in [n]$, generate η-bit primes p_i, α-bit primes g_i, and compute $x_0 = \prod_{i\in[n]} p_i$. Let $\Pi = (\pi_{ij}) \in \mathbb{Z}^{n\times n}$ with $\pi_{ij} \leftarrow (n2^\rho, (n+1)2^\rho) \cap \mathbb{Z}$ if $i = j$, otherwise $\pi_{ij} \leftarrow (-2^\rho, 2^\rho) \cap \mathbb{Z}$. For $i \in [n]$, generate $\boldsymbol{r}_i \in \mathbb{Z}^n$ by choosing randomly and independently in the half-open parallelepiped spanned by the columns of the matrix Π and denote by r_{ij} the j-th component of $\boldsymbol{r}_i$. Generate $H = (h_{ij}), A = (a_{ij}) \in \mathbb{Z}^{n\times n}$ such that H is invertible and $\|H^T\|_\infty \le 2^\beta$, $\|(H^{-1})^T\|_\infty \le 2^\beta$ and for $i \in [n]$, $j \in [\ell]$, $a_{ij} \leftarrow [0, g_i)$. Then define:

$$
\begin{aligned}
y &= \mathsf{CRT}_{(p_i)}\left(\frac{r_i g_i + 1}{z}\right), \text{ where } r_i \leftarrow (-2^\rho, 2^\rho) \cap \mathbb{Z} \text{ for } i \in [n],\\
x_j &= \mathsf{CRT}_{(p_i)}\left(\frac{r_{ij} g_i}{z}\right) \text{ for } j \in [\tau],\\
x'_j &= \mathsf{CRT}_{(p_i)}(x'_{ij}), \text{ where } x'_{ij} = r'_{ij} g_i + a_{ij} \text{ and } r'_{ij} \leftarrow (-2^\rho, 2^\rho) \cap \mathbb{Z} \text{ for } i \in [n], j \in [\ell],\\
(\boldsymbol{p}_{zt})_j &= \sum_{i=1}^{n} h_{ij} \cdot (z^\kappa \cdot g_i^{-1} \bmod p_i) \cdot \prod_{i'\neq i} p_{i'} \bmod x_0 \text{ for } j \in [n].
\end{aligned}
$$

Output $\mathsf{params} = (n, \eta, \alpha, \rho, \beta, \tau, \ell, \nu, y, \{x_j\}, \{x'_j\}, \{\Pi_j\}, s)$ and $\boldsymbol{p}_{zt}$.[6]

Re-randomizing level-1 encodings: $c' \leftarrow \mathsf{reRand}(\mathsf{params}, c)$. For $j \in [\tau], i \in [n]$, sample $b_j \leftarrow \{0,1\}$, $b'_i \leftarrow [0, 2^\mu) \cap \mathbb{Z}$, with $\mu = \rho + \alpha + \lambda$. Return $c' = [c + \sum_{j\in[\tau]} b_j \cdot x_j + \sum_{i\in[n]} b'_i \cdot \Pi_i]_{x_0}$. Note that this is the only procedure in the CLT multilinear map that uses the x_j's.[7]

Adding and multiplying encodings: $\mathsf{Add}(c_1, c_2) = [c_1 + c_2]_{x_0}$ and $\mathsf{Mul}(c_1, c_2) = [c_1 \cdot c_2]_{x_0}$.

Zero-testing: $\mathsf{isZero}(\mathsf{params}, \boldsymbol{p}_{zt}, u_\kappa) \stackrel{?}{=} 0/1$. Given a level-$\kappa$ encoding c, return 1 if $\|[\boldsymbol{p}_{zt} \cdot c]_{x_0}\|_\infty < x_0 \cdot 2^{-\nu}$, and return 0 otherwise.

In [GLW14, App. B.3], Gentry *et al.* described a variant of the above construction that aims at generalizing asymmetric cryptographic bilinear maps. Our attack can be adapted to that variant.

2.2 Zeroizing Attack on GGH

As a warm-up before describing the zeroizing attack on CLT, we recall the zeroizing attack on GGH.

Garg *et al.* constructed the first approximation to multilinear maps, by using ideal lattices [GGH13a]. They used the polynomial ring $R = \mathbb{Z}[x]/(x^n + 1)$ and a (prime) principal ideal $\mathcal{I} = \langle \boldsymbol{g} \rangle \subseteq R$, where $\boldsymbol{g}$ is a secret short element. They also chose an integer parameter q and another random secret $\boldsymbol{z} \in R_q = R/(qR)$. Then one can encode an element of $R/\mathcal{I}$, via division by $\boldsymbol{z}$ in R_q. More precisely, a level-i encoding of the coset $\boldsymbol{e} + \mathcal{I}$ is an element of the form $[\boldsymbol{c}/\boldsymbol{z}^i]_q$, where $\boldsymbol{c} \in \boldsymbol{e} + \mathcal{I}$ is short. By publishing a zero-testing parameter, any user can decide whether two elements encode the same coset or not.

The zero-testing parameter is $\boldsymbol{p}_{zt} = [\boldsymbol{h} \cdot \boldsymbol{z}^\kappa / \boldsymbol{g}]_q$, where $\boldsymbol{h}$ is appropriately small. For a given level-κ encoding $\boldsymbol{u} = [\boldsymbol{c}/\boldsymbol{z}^\kappa]_q$, the quantity $[\boldsymbol{u} \cdot \boldsymbol{p}_{zt}]_q = [\boldsymbol{h} \cdot \boldsymbol{c}/\boldsymbol{g}]_q$ is small if and only if $\boldsymbol{c} \in \mathcal{I}$, i.e., $\boldsymbol{u}$ is an encoding of zero.

[6] Coron *et al.* also described a variant where only one such $(\boldsymbol{p}_{zt})_j$ is given out, rather than n of them (see [CLT13, Section 6]). Our attack requires only one $(\boldsymbol{p}_{zt})_j$.

[7] This procedure can be adapted to higher levels $1 < k \le \kappa$ by publishing appropriate quantities in params.

The latter creates a weak point in the scheme, which enables to solve the Subgroup Membership (SubM) and the decision linear (DLIN) problems easily, by so-called "zeroizing" attack. It uses the property that an encoding of zero has small value when it is multiplied by the zero-testing parameter. In that case, the reduction modulo q is vacuous, and one can have equations over R (instead of R_q) and compute some fixed multiples of secrets. The attack procedure can be summarized as follows. We refer the reader to [GGH13a] for a more detailed description. It relies on the following public parameters:

- $\boldsymbol{y} = [\boldsymbol{a}/\boldsymbol{z}]_q$, a level-1 encoding of 1, namely $\boldsymbol{a} \in 1 + \mathcal{I}$ and $\boldsymbol{a}$ is small,
- $\boldsymbol{x}_j = [\boldsymbol{b}_j \boldsymbol{g}/\boldsymbol{z}]_q$, a level-1 encoding of 0, with $\boldsymbol{b}_j$ small,
- $\boldsymbol{p}_{zt} = [\boldsymbol{h}\boldsymbol{z}^\kappa/\boldsymbol{g}]_q$, $\boldsymbol{h} \in R$ appropriately small, the zero-testing parameter.

Step 1: Compute level-κ encodings of zero and get the equations in R by multiplying by the zero-testing parameter.

Let $\boldsymbol{u} = \boldsymbol{d}/\boldsymbol{z}^t$ be a level-t encoding of some message $\boldsymbol{d} \bmod \mathcal{I}$. Then

$$\begin{aligned}\boldsymbol{f} &= [\boldsymbol{u} \cdot \boldsymbol{x}_j \cdot \boldsymbol{p}_{zt} \cdot \boldsymbol{y}^{\kappa-t-1}]_q = \left[\frac{\boldsymbol{d}}{\boldsymbol{z}^t} \cdot \frac{\boldsymbol{b}_j \cdot \boldsymbol{g}}{\boldsymbol{z}} \cdot \frac{\boldsymbol{h}\boldsymbol{z}^\kappa}{\boldsymbol{g}} \cdot \frac{\boldsymbol{a}^{\kappa-t-1}}{\boldsymbol{z}^{\kappa-t-1}}\right]_q \\ &= \underbrace{\boldsymbol{d} \cdot \boldsymbol{b}_j \cdot \boldsymbol{h} \cdot \boldsymbol{a}^{\kappa-t-1}}_{\ll q}.\end{aligned}$$

Note that the last term in the above equation consists of only small elements, so that the equality holds without modulus reduction by q. Therefore we can obtain various multiples of $\boldsymbol{h}$ (in R) for various $\boldsymbol{u}$ and $\boldsymbol{x}_j$.

Step 2: From multiples of $\boldsymbol{h}$, compute a basis of $\langle \boldsymbol{h} \rangle$. Using a similar procedure, compute a basis of $\langle \boldsymbol{h} \cdot \boldsymbol{g} \rangle$, and hence a basis for $\mathcal{I}$ (by dividing $\langle \boldsymbol{h} \cdot \boldsymbol{g} \rangle$ by $\langle \boldsymbol{h} \rangle$).

SubM is as follows: Given level-1 encoding $\boldsymbol{u} = [\boldsymbol{d}/\boldsymbol{z}]_q$, assess whether $\boldsymbol{d} \in \langle \boldsymbol{g}_1 \rangle$, where $\boldsymbol{g} = \boldsymbol{g}_1 \cdot \boldsymbol{g}_2$ (note that in this context, $\mathcal{I}$ is not a prime ideal). Using the above method, we can get $\boldsymbol{f} = \boldsymbol{d} \cdot \Delta$ for some Δ (which is unrelated to $\boldsymbol{g}$). Taking the gcd of $\langle \boldsymbol{f} \rangle$ and $\mathcal{I}$, we easily solve the subgroup membership problem.

DLIN is as follows: Given level-t encodings $\boldsymbol{C} = (\boldsymbol{c}_{ij})_{i,j\in[N]}$ of messages $\boldsymbol{M} = (\boldsymbol{m}_{ij})_{i,j\in[N]}$ for some $t < \kappa$ and $N \cdot t > \kappa$,[8] assess whether the rank of $\boldsymbol{M}$ (over the field $R/\mathcal{I}$) is full or not. Using the above, we can compute $\boldsymbol{M} \cdot \Delta$ for some $\Delta \in R/\mathcal{I}$ which is unlikely to be 0. In that case, the matrices $\boldsymbol{M} \cdot \Delta$ and $\boldsymbol{M}$ have equal rank, and the problem is easy to solve.

3 A Zeroizing Attack on CLT

The first step of the attack is similar to that of the zeroizing attack of GGH. We compute many level-κ encodings of zero and multiply them by the zero-testing parameter. Then we get matrix equations over $\mathbb{Q}$ (not reduced modulo x_0). By adapting the latter to CLT, one would obtain samples from the ideal $\langle h_1, \ldots, h_n \rangle \subseteq \mathbb{Z}$. Most of the time, it is the whole $\mathbb{Z}$, and the samples do not contain any useful information. Instead, we form matrix equations by using several x_j's rather than a single one.

These equations share common terms. The second step of the attack is to remove some of these common terms by computing the ratio (over the rationals) between two such equations,

[8] If N is smaller than that, the problem is not interesting as it can always be solved efficiently using the zero-test parameter.

and to extract the ratios of the CRT components of the underlying plaintexts by computing the eigenvalues.

The third step consists in recovering the p_i's from these CRT components. Once the p_i's are obtained, recovering the other secret quantities is relatively straightforward.

Now we give full details of each step.

3.1 Constructing Matrix Equations over $\mathbb{Z}$

Let $t \leq \kappa - 1$. Let c be a level-t encoding of $(m_1^{(c)}, \ldots, m_n^{(c)})$, i.e., $c = c_i/z^t \bmod p_i$. Then we can compute the following quantities using the public parameters (for $j \in [\ell], k \in [\tau]$):

$$\begin{aligned} w_{jk} := \left[c \cdot x'_j x_k \cdot y^{\kappa-t-1} \cdot (\boldsymbol{p}_{zt})_1\right]_{x_0} &= \sum_{i=1}^{n} h_{i1} \cdot \left[c \cdot x'_j x_k y^{\kappa-t-1} z^\kappa g_i^{-1} \bmod p_i\right] \cdot \frac{x_0}{p_i} \mod x_0 \\ &= \sum_{i=1}^{n} h_{i1} c_i x'_{ij} r_{ik} (r_i g_i + 1)^{\kappa-t-1} \cdot \frac{x_0}{p_i} \mod x_0 \\ &= \sum_{i=1}^{n} x'_{ij} h'_i c_i r_{ik} \mod x_0, \end{aligned}$$

where $h'_i := h_{i1}(r_i g_i + 1)^{\kappa-t-1} x_0/p_i$ for $i \in [n]$.

Now, as c is a level-t encoding, then $x'_j \cdot (c \cdot x_k \cdot y^{\kappa-t-1})$ is a valid level-κ Diffie-Hellman product (i.e., a product of one level-0 encoding and κ level-1 encodings). Further, it is an encoding of 0, as x_k is an encoding of 0. By design, we have that $|w_{jk}|$ is much smaller than x_0 (this may be checked by a tedious computation, but this is exactly how the correctness requirement for the zero-test parameter is derived). As a result, the equation $w_{jk} = \sum_{i \in [n]} x'_{ij} h'_i c_i r_{ik}$ holds over the integers.

This equation can be rewritten as follows:

$$w_{jk} = (x'_{1j}, \ldots, x'_{nj}) \cdot \mathsf{diag}(c_1, \ldots, c_n) \cdot \mathsf{diag}(h'_1, \ldots, h'_n) \cdot (r_{1k}, \ldots, r_{nk})^T.$$

By letting the index pair (j, k) vary in $[n] \times [n]$, we obtain a matrix equation involving the following matrix $\boldsymbol{W}_c = (w_{jk})_{j,k} \in \mathbb{Z}^{n \times n}$.

$$\begin{aligned} \boldsymbol{W}_c &= \begin{pmatrix} x'_{11} & \cdots & x'_{n1} \\ & \ddots & \\ x'_{1n} & \cdots & x'_{nn} \end{pmatrix} \begin{pmatrix} c_1 & & 0 \\ & \ddots & \\ 0 & & c_n \end{pmatrix} \begin{pmatrix} h'_1 & & 0 \\ & \ddots & \\ 0 & & h'_n \end{pmatrix} \begin{pmatrix} r_{11} & \cdots & r_{1n} \\ & \ddots & \\ r_{n1} & \cdots & r_{nn} \end{pmatrix} \\ &= \quad \boldsymbol{X'} \qquad \mathsf{diag}(c_1, \ldots, c_n) \quad \mathsf{diag}(h'_1, \ldots, h'_n) \qquad \boldsymbol{R}. \end{aligned} \tag{1}$$

To build these equations, we need sufficiently many x'_j's and x_k's. Namely, we need $\ell \geq n$ and $\tau \geq n$. The design conditions on ℓ and τ ensure that this is the case.

Note that the only component in the right hand side of Equation (1) that depends on c is $\mathsf{diag}(c_1, \ldots, c_n)$: the matrices $\boldsymbol{X'}, \boldsymbol{R}$ and $\mathsf{diag}(h'_1, \ldots, h'_n)$ are independent of c.

3.2 Breaking into the CRT Decomposition

We now take $t = 0$, and instantiate Equation (1) with $c = x'_1$ and $c = x'_2$. We obtain, for $j \in \{1, 2\}$:

$$\boldsymbol{W}_j := \boldsymbol{X'} \cdot \mathsf{diag}(x'_{1j}, \ldots, x'_{nj}) \cdot \mathsf{diag}(h'_1, \ldots, h'_n) \cdot \boldsymbol{R}.$$

We can then compute (over $\mathbb{Q}$):

$$\boldsymbol{W}_1 \cdot \boldsymbol{W}_2^{-1} = \boldsymbol{X'} \cdot \mathsf{diag}\left(\frac{x'_{11}}{x'_{12}}, \ldots, \frac{x'_{n1}}{x'_{n2}}\right) \boldsymbol{X'}^{-1}.$$

In the latter, we need that $\boldsymbol{W}_2$ is invertible. Below, we will also need that $\boldsymbol{W}_1$ is invertible. We argue here that we may assume this is the case. We prove it for $\boldsymbol{W}_1$. Note first that the x'_{i1}'s and the h'_i's are all non-zero, with overwhelming probability. Note that by design, the matrix $(r_{ij})_{i\in[n],j\in[\tau]}$ has rank n (see [CLT13, Se. 4]). The same holds for the matrix $(x'_{ij})_{i\in[n],j\in[\ell]}$ (see [CLT13, Le. 1]). As we can compute the rank of a $\boldsymbol{W}_c \in \mathbb{Z}^{t\times t}$ obtained by using an $\boldsymbol{X'} \in \mathbb{Z}^{t\times n}$ and an $\boldsymbol{R} \in \mathbb{Z}^{n\times t}$ obtained by respectively using a t-subset of the x'_j's and a t-subset of the x_j's, without loss of generality we may assume that our $\boldsymbol{X'}, \boldsymbol{R} \in \mathbb{Z}^{n\times n}$ are non-singular. The cost of finding such a pair $(\boldsymbol{X'}, \boldsymbol{R})$ is bounded as $\widetilde{\mathcal{O}}((\tau+\ell)\cdot(\kappa^4\lambda^7 + n^\omega \log x_0)) = \widetilde{\mathcal{O}}(\kappa^{\omega+3}\lambda^{2\omega+6})$, with $\omega \leq 2.38$. Here we used the fact that the rank of a matrix $\boldsymbol{A} \in \mathbb{Z}^{n\times n}$ may be computed in time $\widetilde{\mathcal{O}}(n^\omega \log \|\boldsymbol{A}\|_\infty)$ (see [Sto09]). This dominates the overall cost of the attack.

As $\boldsymbol{X'}$ is non-singular, we obtain that the x'_{i1}/x'_{i2}'s are the eigenvalues (over $\mathbb{Q}$) of $\boldsymbol{W}_1 \cdot \boldsymbol{W}_2^{-1}$. These may be computed in polynomial-time from $\boldsymbol{W}_1 \cdot \boldsymbol{W}_2^{-1}$ (e.g., by factoring the characteristic polynomial). We hence obtain the x'_{i1}/x'_{i2}'s, for all $i \in [n]$, possibly in a permuted order. We write the fraction x'_{i1}/x'_{i2} as x''_{i1}/x''_{i2}, with co-prime x''_{i1} and x''_{i2}. At this stage, we have the (x''_{i1}, x''_{i2})'s at hand, for all $i \in [n]$. For each of these pairs, we compute:

$$\gcd(x''_{i1} \cdot x'_2 - x''_{i2} \cdot x'_1, x_0).$$

The prime p_i is a common factor of both $x''_{i1} \cdot x'_2 - x''_{i2} \cdot x'_1$ and x_0. As all the other factors of x_0 are huge, there is a negligible probability that the gcd is not exactly p_i: another p_j divides $x''_{i1} \cdot x'_2 - x''_{i2} \cdot x'_1$ if and only if $x'_{i1} \cdot x'_{2j} = x'_{i2} \cdot x'_{1j}$.

3.3 Disclosing all the Secret Quantities

We now know all the p_i's.

Let $j \in [\tau]$. We have $x_j/y = r_{ij}g_i/(r_i g_i + 1) \bmod p_i$. As the numerator and denominator are coprime and very small compared to p_i, they can be recovered by rational reconstruction. We hence obtain $r_{ij}g_i$ for all j. The gcd of the r_{ij}'s reveals g_i. As a result, we can recover all the r_{ij}'s and r_i's.

As $x_1 = r_{i1}g_i/z \bmod p_i$ and as the numerator is known, we can recover $z \bmod p_i$ for all i, and hence $z \bmod x_0$. The h_{ij}'s can then be recovered as well. So can the r'_{ij}'s and a_{ij}'s.

Acknowledgments. The authors thank Michel Abdalla Jean-Sébastien Coron, Tancrède Lepoint, Benoît Libert, Alon Rosen and Gilles Villard for helpful discussions. The last author was supported by the ERC Starting Grant ERC-2013-StG-335086-LATTAC.

References

[ABP14] Michel Abdalla, Fabrice Benhamouda, and David Pointcheval. Disjunctions for hash proof systems: New constructions and applications. *IACR Cryptology ePrint Archive*, 2014:483, 2014.

[Att14] N. Attrapadung. Fully secure and succinct attribute based encryption for circuits from multi-linear maps. *IACR Cryptology ePrint Archive*, 2014:772, 2014.

[BBS04] D. Boneh, X. Boyen, and H. Shacham. Short group signatures. In *Proc. of CRYPTO*, volume 3152 of *LNCS*, pages 41–55. Springer, 2004.

[BGN05] D. Boneh, E.-J. Goh, and K. Nissim. Evaluating 2-DNF formulas on ciphertexts. In *Proc. of TCC*, volume 3378 of *LNCS*, pages 325–341. Springer, 2005.

[BLMR13] Dan Boneh, Kevin Lewi, Hart Montgomery, and Ananth Raghunathan. Key homomorphic PRFs and their applications. In *Proc. of CRYPTO*, pages 410–428. Springer, 2013.

[BP13] F. Benhamouda and D. Pointcheval. Verifier-based password-authenticated key exchange: New models and constructions. *IACR Cryptology ePrint Archive*, 2013:833, 2013.

[BR13] Z. Brakerski and G. N. Rothblum. Obfuscating conjunctions. In *Proc. of CRYPTO*, volume 8043 of *LNCS*, pages 416–434. Springer, 2013.

[BS02] D. Boneh and A. Silverberg. Applications of multilinear forms to cryptography. *Contemporary Mathematics*, 324:71–90, 2002.

[CLT13] Jean-Sébastien Coron, Tancrede Lepoint, and Mehdi Tibouchi. Practical multilinear maps over the integers. In *Proc. of CRYPTO*, pages 476–493. Springer, 2013.

[CLT14] J.-S. Coron, T. Lepoint, and M. Tibouchi. Personal communication. 2014.

[DGHV10] M. van Dijk, C. Gentry, S. Halevi, and V. Vaikuntanathan. Fully homomorphic encryption over the integers. In *Proc. of EUROCRYPT*, volume 6110 of *LNCS*, pages 24–43. Springer, 2010.

[Gen09] C. Gentry. Fully homomorphic encryption using ideal lattices. In *Proc. of STOC*, pages 169–178. ACM, 2009.

[GGH13a] S. Garg, C. Gentry, and S. Halevi. Candidate multilinear maps from ideal lattices. In *Proc. of EUROCRYPT*, volume 7881 of *LNCS*, pages 1–17. Springer, 2013.

[GGH+13b] S. Garg, C. Gentry, S. Halevi, M. Raykova, A. Sahai, and B. Waters. Candidate indistinguishability obfuscation and functional encryption for all circuits. In *Proc. of FOCS*, pages 40–49. IEEE Computer Society Press, 2013.

[GGH14] C. Gentry, S. Gorbunov, and S. Halevi. Graded multilinear maps from lattices. *IACR Cryptology ePrint Archive*, 2014:645, 2014.

[GGHZ14a] S. Garg, C. Gentry, S. Halevi, and M. Zhandry. Fully secure attribute based encryption from multilinear maps. Cryptology ePrint Archive, Report 2014/622, 2014.

[GGHZ14b] S. Garg, C. Gentry, S. Halevi, and M. Zhandry. Fully secure functional encryption without obfuscation. Cryptology ePrint Archive, Report 2014/666, 2014.

[GLSW14] Craig Gentry, Allison B Lewko, Amit Sahai, and Brent Waters. Indistinguishability obfuscation from the multilinear subgroup elimination assumption. *IACR Cryptology ePrint Archive*, 2014:309, 2014.

[GLW14] Craig Gentry, Allison Bishop Lewko, and Brent Waters. Witness encryption from instance independent assumptions. *IACR Cryptology ePrint Archive*, 2014:273, 2014.

[LMR14] K. Lewi, H. W. Montgomery, and A. Raghunathan. Improved constructions of PRFs secure against related-key attacks. In *Proc. of ACNS*, volume 8479 of *LNCS*, pages 44–61. Springer, 2014.

[LS14] Hyung Tae Lee and Jae Hong Seo. Security analysis of multilinear maps over the integers. In *Proc. of CRYPTO*, pages 224–240. Springer, 2014.

[Sco02] M. Scott. Authenticated ID-based key exchange and remote log-in with simple token and PIN number. *IACR Cryptology ePrint Archive*, 2002:164, 2002.

[Sto09] A. Storjohann. Integer matrix rank certification. In *Proc. of ISSAC*, pages 333–340. ACM, 2009.

[Zha14] M. Zhandry. Adaptively secure broadcast encryption with small system parameters. *IACR Cryptology ePrint Archive*, 2014:757, 2014.

[Zim14] J. Zimmerman. How to obfuscate programs directly. *IACR Cryptology ePrint Archive*, 2014:776, 2014.