ECRYPT CSA

H2020-ICT-2014 – Project 645421

ECRYPT – CSA

ECRYPT – Coordination & Support Action

D2.1

Tools for Asymmetric Cryptanalysis – New Challenges and Research Directions

Due date of deliverable: 01. November 2015
Actual submission date: 30. October 2015

Start date of project: 1 March 2015 Duration: 3 years

Lead contractor: Ruhr Universität Bochum (RUB)

Revision 0.1

Project co-funded by the European Commission within the H2020 Programme		
Dissemination Level		
PU	Public	X
PP	Restricted to other programme participants (including the Commission services)	
RE	Restricted to a group specified by the consortium (including the Commission services)	
CO	Confidential, only for members of the consortium (including the Commission services)	

Tools for Asymmetric Cryptanalysis – New Challenges and Research Directions

Editor
Gottfried Herold
Alexander May

Contributors
Martin Albrecht,
Daniel Dadusch,
Léo Ducas,
Thomas Johansson,
Thijs Laarhoven,
Tancrède Lepoint.

30. October 2015

Revision 0.1

The work described in this report has in part been supported by the Commission of the European Communities through the H2020-ICT program under contract H2020-ICT-2014 no. 645421. The information in this document is provided as is, and no warranty is given or implied that the information is fit for any particular purpose. The user thereof uses the information at its sole risk and liability.

Contents

Executive Summary

Management Statement: ***Post-quantum crypto systems will be required in practice within the next two decades, the most promising candidates are based on* lattice problems*. Defining* secure parameters *is not possible without extensive cryptanalytic efforts in theory and practice, which requires also at least one more decade.***

Moreover, new cryptographic constructions that allow for additional functionalities – like e.g. fully homomorphic encryption, obfuscation and multilinear maps – require new cryptanalytic directions.

We identify the* major challenges and research directions *in these areas.

On the Necessity of Post-Quantum Crypto. After the invention of fast quantum algorithms to fully break current number theory-based, e.g. factoring and discrete logarithm, crypto systems in the mid-90s by Peter Shor, major efforts were invested by the scientific community to identify cryptographic schemes that remain secure even in the presence of quantum computers. Current predictions say that within the next two decades we will experience the development of quantum computers that are capable of breaking current number-theory based crypto systems.

Among the candidates for the so-called post-quantum era are currently systems based on the difficulty to solve multivariate polynomial equations, coding problems and lattice problems. Within the last decade, lattice problems were identified as the most promising candidate, since they admit a so-called average case to worst case reduction. This implies that average case problems are as hard as worst case instances. Stated more simply, either almost all problems are easy to solve or almost all problems are hard to solve. Since many problems on lattices are actually among the algorithmically hardest problems, it is widely believed that almost all instances are actually hard. This is a very intriguing property for cryptographic systems, where one wants to ensure that a cryptographic scheme is (almost) always hard to break.

So understanding the hardness of lattice problems has become one of the major challenges in cryptography. In fact, most topics of this report are directly or indirectly (via reductions) related to the hardness of lattice problems.

On the Importance of Cryptanalysis. Naturally, one wants to construct cryptographic systems that remain secure against *any* type of attacker. This is usually done via a reduction to a mathematically hard problem Π. Namely, one shows that any successful attacker A can be turned into an algorithm solving Π. In order to conclude that *no* efficient attacker exists, one has to guarantee that Π cannot be solved within a predetermined number of steps, say within 2^λ steps, where λ is called the security level. This in turn means that the parameters

for the instances of Π have to be set such that the fastest known algorithms run in time at least 2^λ.

Hence, for lattice problems one has to determine the running time to compute shortest/closest vectors as a function of the lattice dimension n. If this e.g. be done in 2^{cn} steps for some constant c, then one has to set $n \geq \frac{\lambda}{c}$ in order to achieve security level λ. Notice that we have conflicting goals in cryptography: whereas the security of systems grows with increasing parameter sizes, their efficiency suffers. Therefore, one wishes to set parameters as small as possible such that a predetermined level of security is fulfilled.

This in turn means that without an extensive cryptanalytic research one cannot determine the necessary key sizes of a cryptographic scheme.

On the Timeline for Establishing Stable Key Sizes via Cryptanalysis. For a good comparison it is instructive to look how such extensive cryptanalytic efforts have led to the well-established security of the factoring-based RSA crypto system against classical (not quantum) computers. At the time of its invention, RSA was considered to be secure for moduli of bit-sizes $n = 426$. In fact, in 1977 Ron Rivest (an RSA inventor) estimated that factoring such a number would require 40 quadrillion years. Eventually, this prediction was not quite accurate because RSA $-$ 426 was broken not even 20 years later in 1994.

This small anecdote illustrates that predictions should take cryptanalytic algorithmic progress into account. Today, the current record for factoring RSA numbers is $n = 768$ and the predictions for factoring this number quite accurately matched the real running times. This in turn allows us to design RSA crypto systems that will resist all classical attacks within the next decade – provided that there is no algorithmic breakthrough.

The reason that we can accurately predict key sizes for RSA in the long run is based on an extensive study for factoring on an *asymptotic* algorithmic level and as well in *practical experiments*. Similar research has to be performed on lattice-based and related problems such as Learning with Errors (LWE) before these problems can be used in widely deployed crypto systems in practice. Since it is extremely costly in practice to replace crypto systems, one has to establish the right hard problems with well-established (and long-term predictable) security guarantees.

The systems discussed in this report require still a major research effort before they can be safely deployed in practice.

Major Challenges and Research Directions

1. **Open problems for lattices** (for more details, see Section 1.3)
   (a) Freely available state-of-the-art BKZ implementation
   (b) Low memory lattice reduction
   (c) Narrow down the gap between heuristic and provable methods
   (d) Lower bounds for CVP/SVP
   (e) Generalize principal ideal lattice attacks with short generators
   (f) Attacks on general Ideal Lattices
   (g) How hard is approximating Shortest Vectors up to polynomial factors
2. **Open problems for Learning With Errors** (for more details, see Section 2.2)
   (a) Study low memory attacks
   (b) Develop asymptotic understanding
   (c) Optimize and unify current attack
   (d) Establish and break challenges of reasonable size
   (e) Establish (classically) tight security reduction to lattice problems
   (f) Study hardness of practical variants of LWE
3. **Open problems for Multilinear Maps** (for more details, see Section 3.2)
   (a) Construct new Multilinear Maps candidates avoiding zero-testing attacks
   (b) Prove Security of Constructions
   (c) Attacks for Indistinguishability Obfuscation

Chapter 1

SVP

The most fundamental computational problems about lattices are the Shortest Vector Problem (SVP) and the related Closest Vector Problem (CVP). For lattice-based schemes, we either have a reduction to a variant of SVP/CVP, or the relevant attacks on a scheme amount to solving SVP/CVP instances. So understanding the hardness of these problems is paramount to understanding the security of lattice-based cryptography.

Algorithms for SVP usually either solve the (harder) CVP problem, or can be modified to work for the variants of CVP that are relevant for cryptography. So we focus on algorithms for SVP as the central cornerstone of lattice-based cryptanalysis. The hardness of SVP depends mainly on the lattice dimension n. This parameter roughly plays the same role as the bit-size in RSA and is the parameter we can tune to achieve a desired level of security.

1.1 State of the Art

Algorithms solving SVP can be divided into two categories, depending on whether they provably solve all instances or whether they only work heuristically for average case instances. Usually, SVP-solvers are used in practice in block reduction algorithms that compute a block-reduced lattice basis [57]. Such a block-reduced basis defines an interpolation between a weakly reduced basis, e.g. an LLL-reduced basis[45] that corresponds to block-size $\beta = 2$, and a fully-reduced basis, i.e. a HKZ basis that corresponds to block-size $\beta = n$. Block reduction algorithms use SVP-solvers with dimension β as a subroutine. Block lattice algorithms approximate a shortest vector up to a factor of $O(\beta^{\frac{n}{\beta}})$.

1.1.1 Provable methods

Kannan's algorithm[40] from 1983 solves SVP in time $n^{O(n)}$, using only poly(n) memory. Kannan's method starts with a somewhat reduced lattice basis $b_1, \ldots, b_n \in \mathbb{R}^n$ (e.g. an LLL-reduced basis). A shortest vector v can then expressed as $v = \sum_{i=1}^{n} c_i b_i$ with c_i of bounded absolute value. One then enumerates over all possible values of c_i to search for a shortest vector. This defines a search tree for which some pruning strategies cut off subtrees that are (more) unlikely to contain the shortest vector. This pruned enumeration is currently the best strategy to find shortest vectors in practice.[28]

In 2001, Ajtai, Kumar and Sivakumar[4] improved Kannan's running time to single-exponential time $2^{O(n)}$, but their AKS algorithm requires also exponential memory $2^{O(n)}$.

This algorithm works by a process called sieving. The algorithm starts with a long list of lattice vectors within a somewhat large ball $\mathcal{B}$ around zero. In each iteration, one computes all pairs of lattice vectors that are close-by, thereby reducing the size of the lattice vectors. It is not hard to see that this process succeeds, whenever the number of vectors in the original ball $\mathcal{B}$ is large enough.

The best fully provable SVP sieve method is currently the algorithm of Aggarwal, Dadush, Regev and Stephens-Davidowitz[3] with running-time and space complexity 2^n.

1.1.2 Heuristic methods

The provable AKS method has many variants [52, 50, 43] that allow for a heuristic running time analysis with quite small constants c in the exponent 2^{cn}. The current record is [7] with $c = 0.292$. Unfortunately the memory consumption of these algorithms is usually in the same order of magnitude as the running time. This leads to memory issues for $n > 100$ in practice, since the lists do not fit into main memory.

1.2 Special Lattices

In cryptographic constructions, the lattices under consideration are often not arbitrary, but highly structured. For reasons of efficiency, implementors prefer structured lattices to both reduce key sizes and speed up computations, such as in Ring-LWE[48, 49], NTRU[38] or SWIFFT[47]. In some case, the additional structure in the lattices is even relevant for their core functionalities, e.g. for fully homomorphic encryption[32, 59] or some candidate multilinear maps[29].

In such applications, the lattice L under consideration may also be viewed as an ideal in some finite dimensional $\mathbb{Z}$-algebra R and is called an ideal lattice. A typical choice of R is $R = \mathbb{Z}[X]/(f)$, where f is either a cyclotomic polynomial of degree n or $f = X^n \pm 1$ (or both: $X^{2^n} + 1$ is the 2^{n+1}'th cyclotomic polynomial). The case $R = \mathbb{Z}_p[X]/(X^n \pm 1)$ can be obtained by adding (p) to L.

For such lattices, by a natural automorphism of R we can construct and consider whole orbits of vectors rather than single lattice vectors. This can save a factor of n for some SVP/CVP algorithms. Apart from that, it is not known how to solve SVP significantly faster for *general ideal* lattices than for general lattices (at least if n is prime and f is irreducible – otherwise, subfield attacks are an issue[8]).

For an even more special class of ideal lattices, better algorithms are known. These are lattices $L = (g)$ that are *principal* ideals with a *short generator*[1] g when R is the ring of integers in a number field (e.g. cyclotomic f). Such lattices are used, e.g. in [59, 29, 44]. In such a case, SVP can be solved by the following two steps: first, find *any* generator $L = (h)$ of L as a principal ideal. Second, construct a short generator $g = u \cdot h$ (for a unit $u \in R^*$) from h.

The first problem is known as the Principal Ideal Problem (PIP). It can be solved classically in subexponential time $2^{\widetilde{O}(n^{2/3})}$ [9, 10] or quantumly in polynomial time[27, 15, 11]. The second problem can be solved by bounded distance decoding on the log-unit lattice of R [15]. Due to the special geometry of number fields and the fact that we need to solve a BDD

[1] We need that g is actually a shortest vector. Even for general principal lattices, the shortest generator is typically not a shortest vector, but larger by a factor of $2^{\widetilde{\Theta}(\sqrt{n})}$ [25].

problem rather than CVP, this problem is actually not hard at all. It is possible to write down a good enough basis for the log-unit lattice and use Babai's algorithm[6]. The recent analysis of [25] shows that this gives a polynomial time algorithm if n is a prime power.

As a consequence, we have a quantum polynomial time attack against such SVP instances. Classically, we have a subexponential time attack $2^{\widetilde{O}(n^{2/3})}$ against SVP instances on principal ideals with short generators in cyclotomic number fields of prime-power index.

1.3 Open problems

(a) **Freely available state-of-the-art BKZ implementation:** Currently the standard in most experimental results is the BKZ lattice reduction that is implemented in Shoup's NTL-library [58]. The problem is that Shoup's implementation uses an SVP-subroutine with complexity $2^{O(n^2)}$. This makes the algorithm impractical for block-sizes greater than 30. While there exist more state-of-the-art algorithms like BKZ 2.0 [16] that include recent advances in pruning techniques, these algorithms are not publicly available.

Given the importance of lattice reduction, this situation is not satisfactory. Moreover, it led some researchers falsely conclude that SVP-problems are harder in practice than theoretically predicted.

(b) **Low memory lattice reduction:** Can we design an SVP algorithm with running time $2^{O(n)}$ and only poly(n) memory complexity? What is the best running time bound that we can achieve with some fixed memory constraints (in practice)?

(c) **Narrow down the gap between heuristic and provable methods:** Can we bring provable methods (2^n) closer to the heuristic bound ($2^{0.29n}$)? Or are there worst-case lattice instances that hinder us? And if yes, which structure do worst-case instances have? This would be interesting for constructions based on very hard lattices.

(d) **Lower bounds for CVP/SVP:** Current SVP/CVP algorithms offer running time $2^{O(n)}$ with a small constant in the exponent. Can we prove a matching lower bound of $2^{\Omega(n)}$ assuming SETH (Strong Exponential Time Hypothesis) or something similar.

(e) **Generalize Principle Ideal Lattice attacks with short generators:** Can we extend the attacks on principal ideal lattices with short generators to rings of integers of non-cyclotomic number fields?

(f) **Attacks on general Ideal Lattices:** What attacks are possible on general ideal lattices, possibly using the fact that the dimension is not prime?

(g) **How hard is approximating Shortest Vectors up to polynomial factors?:** Cryptographic lattice-based constructions are usually based on the hardness of Gap-SVP for polynomial approximation factors, e.g. in order to allow for decryption. We know that approximating SVP up to (roughly) constant factors is NP-hard [41], approximating to a factor of size (roughly) $\sqrt{n}$ is unlikely to be NP-hard [34], and approximating to (almost) exponential factors is efficient [45]. However, for the range of cryptographic keys the complexity status of Gap-SVP remains somewhat vague.

Chapter 2

Cryptanalysis of Learning with Errors

2.1 State of the Art

In the Learning Errors with Errors (LWE) computational problem, one is given a modulus $q \in \mathbb{N}$ and (an arbitrarily large) number m of samples $(\mathbf{a}_i, \langle \mathbf{a}_i, \mathbf{s} \rangle + e_i)$, where each $\mathbf{a}_i$ is chosen uniformly at random from $\mathbb{Z}_q^n$, $\langle \mathbf{a}_i, \mathbf{s} \rangle = \sum_{i=1}^{n} a_i s_i \pmod q$ and e_i is a discrete Gaussian error with standard deviation s. The goal is to recover the secret vector $s \in \mathbb{Z}_q^n$. An important special case of LWE is Learning Parity with Noise (LPN), where $q = 2$ and e_i is a Bernoulli variable.

In the decisional version of LWE one has to distinguish LWE samples from samples in which $\langle \mathbf{a}_i, \mathbf{s} \rangle + e_i$ is replaced by a random $b \in \mathbb{Z}_q$. It is well-known that for $q = \text{poly}(n)$ the search and decisional variant are polynomial-time equivalent. So most cryptanalytic attacks focus on the computational problem.

Notice that LWE instances are given by four parameters n, q, s and m, where an attacker might optimize m as a function of the others. So the running time of an LWE solver is a function of n, q and s, where for most practical applications both q and s are polynomial functions in n.

In the last decade, the LWE problem has been an extremely versatile source for constructing cryptographic primitives. Since solving LWE is at least as hard solving certain worst-case lattice problems (such as GapSVP and SIVP) [55, 53], it is nowadays considered as the *most important candidate for post-quantum cryptography.*

The algorithms for LWE solving split into the following 3 categories.

Combinatorial attacks: In 1999, Blum, Kalai and Wasserman (BKW[12]) showed that LPN can be solved in time $2^{O(\frac{n}{\log n})}$. The BKW algorithm can be easily generalized to the LWE setting, yielding a running time of $2^{O(n)}$, when $q, s = \text{poly}(n)$. Unfortunately, the space and the sample complexity – i.e. the number m of samples – of BKW is also $2^{\Theta(n)}$.

Lattice attacks: The LWE problem can be directly translated into a closest vector problem with running time $2^{O(n)}$. The benefit of this approach as compared to the previous BKW combinatorial attack is that the space complexity is only $O(n^2)$ elements and the sample complexity is $O(n)$.

Algebraic attacks: For sufficiently large s, the algorithm of Arora and Ge [5] runs in time, space and sample complexity $2^{O(n \log n)}$. However, for $s < \sqrt{n}$ the algorithm runs in sub-exponential time $2^{s^2 \log^2 n}$.

2.2 Open Problems

(a) **Study low memory attacks:** Currently, the asymptotically best attacks – BKW and lattices sieves – consume an outrageous amount of memory that cannot be handled in practice. Thus, it is unlikely that these attacks lead to real-world attacks. It seems plausible that in practice, one would take lattice attacks with BKZ-type algorithms that use enumeration techniques like Kannan's algorithm. This again stresses the necessity of freely public available block reduction algorithms for practically assessing LWE's hardness. On the theoretical side, one might consider to look for low-memory versions of BKW and lattice sieve algorithms.

(b) **Develop asymptotic understanding:** For a parameter selection with accurate security levels it is mandatory that we (fully) understand the LWE complexity as a function of the parameters n, q and s. Moreover, it would be useful for a fair comparison of algorithms to limit the sample and memory complexity to a reasonable size, since in practice (almost) exponential memory consumptions cannot be handled. It seems to be unclear in which cases the BKW algorithm (with limited sample size) outperforms lattice reduction techniques. To the best of our knowledge, the choice of the optimal algorithm heavily depends on the LWE parameters: The larger $\frac{q}{s}$, the better should be lattice reduction, since this fraction defines the gap in a closest vector problem.
It is absolutely necessary to develop accurate (and easy to handle) asymptotic formulas that tell us how LWE's complexity behaves as function of n, q, s, and that allow us to accurately extrapolate concrete cryptanalytic results to desired security levels.

(c) **Optimize and unify current attacks:** Current work often concentrates on optimizing single sub-procedures of the BKW algorithm [36, 26, 37, 42] or CVP enumeration techniques [46, 28]. However, we need a more flexible general framework that allows to analyze the effects of these local improvements, and allows to assess an optimized algorithm for concrete instances of LWE.

(d) **Establish and break challenges of reasonable size:** One should publish a webpage with BKW challenges for the cryptanalytic community, such as e.g. the RSA challenges [1] or the Darmstadt lattice challenges [2]. Such a page would encourage researchers to implement and test their algorithms on a large scale. So far, the experimental data for LWE is simply to thin too allow for an accurate extrapolation up to cryptographic security levels.

(e) **Establish (classically) tight security reduction to lattice problems:** While Regev's original quantum reduction from Gap-SVP is tight [55], Peikert's classical reduction [53] has a quadratic blow-up. This means that in order to guard against classical attacks, one has to double the bit-size of parameters. This makes LWE more inefficient in practical implementations. But is this quadratic blow-up inherent, or does there exist a tight classical reduction?

(f) **Study hardness of practical variants of LWE:** In practice, for efficiency reasons it is tempting to slightly tweak the originally proposed LWE instances, e.g. by choosing moduli q as powers of 2, take easy-to-sample noise distributions, and use variants of Ring-LWE. Hence, it is important to study these variants cryptanalytically, and to understand the extent to which these variants effect LWE's security.

Chapter 3

Cryptanalysis of Multilinear Maps

Many cryptographic primitives make use of cyclic groups G. Write group elements as g^x for some fixed, public generator $g \in G$. For security, we require at least that the discrete logarithm problem, i.e. computing x from g^x, is hard. Candidate groups, where this is believed to hold (against non-quantum adversaries) are subgroups of $\mathbb{F}_q^*$ or rational points of elliptic curves. For a suitably chosen elliptic curve G, the e.g. Weil or Tate pairings allow us to construct an efficiently computable (symmetric) pairing $e\colon G \times G \to G_T$, where $e(g^x, g^y) = g_T^{xy}$. Here, G_T is a subgroup of the multiplicative group of some finite field, generated by g_T. More generally, we can also consider asymmetric pairings $e\colon G_1 \times G_2 \to G_T$, where G_1, G_2 are related, but different elliptic curves. Viewing g^x as an encoding of x, such a pairing allows to multiply pairs of encodings. For security reasons, we typically ask that similar products of more than two factors cannot be efficiently computed.

Such pairings enable a plethora of applications such as efficient NIZK proofs[35], attribute-based encryption[56] and one-round 3-party key agreement[39].

An important question in that area is to extend bilinear maps to multilinear maps, i.e. maps $e\colon G^\kappa \to G_T$ for any (even fixed) $\kappa > 2$.

3.1 State of the Art

The first candidate construction for a κ-linear map is due to [29]. On a high level, this candidate constructs g^x as an encryption (or "encoding") of x for a specially designed homomorphic encryption scheme that allows to homomorphically add and multiply cipertexts. Homomorphic addition becomes the group operation, homomorphic multiplication becomes the pairing. Using such an underlying encryption scheme requires some trusted setup: indeed, this setup selects κ and a secret/public key pair, where the public key becomes part of the public parameters. Knowledge of the secret key allows to compute discrete logarithms.

Using an underlying encryption scheme has some drawbacks: since the underlying encryption scheme of [29] is randomized, this means that for a given x, there are many possibilities (i.e. ciphertexts) to encode x. In fact, by homomorphically adding encodings of 0, the group elements/ciphertexts may be randomized by the users. Depending on the application, security considerations mandate that we do so during computations. To enable this operation, sufficiently many encodings of zero are provided as part of the public parameters. Now, normally an encryption scheme is supposed not to reveal anything about the plaintexts. In particular, one could not tell whether two encodings encode the same group element. To overcome this,

some "handicapped" secret key, the so-called zero-testing parameters, is published as part of the public parameters. This zero-testing parameter allows to test group elements for equality, provided they have been obtained by multiplying at most κ initial group elements.

The construction due to [29] is based on ideal lattices. A similar construction using integers was proposed soon after by [23]. None of the existing constructions is truly practical. More precisely, an encoding/encryption of x in [29] has the form

$$\frac{x + rs}{z} \bmod q \quad \text{(in the ring of integers of some number field)},$$

where q is a public parameter, r is some small per-message randomness and s, z correspond to the secret key. Products of κ elements have the form $\frac{x+r's}{z^\kappa} \bmod q$. The zero-testing parameter has the form $p_{\text{zt}} = hz^\kappa/s \bmod q$ for somewhat small h. Zero-testing works by multiplying by p_{zt} and checking whether the result is small. The integer based scheme is conceptually similar, but is somewhat more complicated due to usage of the Chinese Remainder Theorem.

The exact security requirements for multilinear maps depend on the application. A typical requirement is that some DDH-like assumption holds. In any case, the minimal requirement is that discrete logarithms are hard to compute.

The main security issue is that in all those variants the zero-testing parameter, combined with the encodings of zero, helps cryptanalysis. Notably, it was realized already in [29] that it is possible to compute a value related to x (a so-called weak discrete logarithm) from encodings of x. The primary reason is that zero-testing is a linear operation: multiplying some encoding $c = \frac{x+rs}{z}$ by an encoding of zero and the zero-testing parameter results in a small element that depends linearly on $x + rs$. This zeroizing attack allows to break many natural assumptions for the ideal-lattice based scheme. [29] showed how to break DDH-like assumptions in the base group and based their applications on assumptions in the target group. Recently, [18] broke those very target group assumptions for the ideal-lattice based scheme.

For the construction based on the integers, [17] showed how to compute the secret key of the underlying scheme from the public parameters in polynomial time. This completely broke the scheme. Several fixes for the scheme were proposed[13, 31, 24], but all of those were subsequently broken as well[22, 19, 51].

Some other constructions were proposed, e.g.[20, 33], but are broken as well[54, 21].

So the current state of the art is that *all* schemes are broken one way or another. All these attacks make use of the linearity of zero-testing and encodings of zero (So far, attempts to fix the schemes by making zero-testing non-linear have been broken by taking derivatives [14], thereby reducing to the linear case).

Still, applications which do not need all those parameters as part of the public parameters might be unaffected by the attacks. Interestingly, this includes the candidate Indistinguishability Obfuscation constructions from multilinear maps [30], whose security remains unclear.

3.2 Open problems

(a) **Construct new Multilinear Map candidates avoiding zero-testing attacks:** Can we construct multilinear maps for $\kappa \geq 3$ where DDH-type assumptions hold? Can we find a way to design a zero-testing algorithm without breaking security? Are there fundamental limitations in the approaches taken so far?

(b) **Prove security of constructions:** Current candidate constructions came without security proof. Try to base security on some natural assumption.

(c) **Attacks for Indistinguishability Obfuscation:** Can we extend the existing attacks on multilinear maps to attacks on candidate Indistinguishability Obfuscation schemes?

Bibliography

[1] The RSA factoring challenge. `http://www.emc.com/emc-plus/rsa-labs/historical/the-rsa-factoring-challenge-faq.htm`.

[2] TU darmstadt lattice challenge. `https://www.latticechallenge.org/`.

[3] D. Aggarwal, D. Dadush, O. Regev, and N. Stephens-Davidowitz. Solving the shortest vector problem in 2^n time using discrete Gaussian sampling: Extended abstract. In R. A. Servedio and R. Rubinfeld, editors, *47th ACM STOC*, pages 733–742. ACM Press, June 2015.

[4] M. Ajtai, R. Kumar, and D. Sivakumar. A sieve algorithm for the shortest lattice vector problem. In *33rd ACM STOC*, pages 601–610. ACM Press, July 2001.

[5] S. Arora and R. Ge. New algorithms for learning in presence of errors. In L. Aceto, M. Henzinger, and J. Sgall, editors, *ICALP 2011, Part I*, volume 6755 of *LNCS*, pages 403–415. Springer, Heidelberg, July 2011.

[6] L. Babai. On lovász' lattice reduction and the nearest lattice point problem (shortened version). In *STACS 85, 2nd Symposium of Theoretical Aspects of Computer Science, Saarbrücken, Germany, January 3-5, 1985, Proceedings*, pages 13–20, 1985.

[7] A. Becker, L. Ducas, N. Gama, and T. Laarhoven. New directions in nearest neighbor searching with applications to lattice sieving. In *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms*, 2016. To appear.

[8] D. J. Bernstein. A subfield-logarithm attack against ideal lattices. Blogpost Feb 2014, retrieved 28.10.2015, `http://blog.cr.yp.to/20140213-ideal.html`.

[9] J.-F. Biasse. Subexponential time relations in the class group of large degree number fields. *Advances in Mathematics of Communications*, 8(4):407–425, 2014.

[10] J.-F. Biasse and C. Fieker. Subexponential class group and unit group computation in large degree number fields. *LMS Journal of Computation and Mathematics*, 17:385–403, 2014.

[11] J.-F. Biasse and F. Song. A polynomial time quantum algorithm for computing class groups and solving the principal ideal problem in arbitrary degree number fields. In *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms*, 2016. To appear.

[12] A. Blum, A. Kalai, and H. Wasserman. Noise-tolerant learning, the parity problem, and the statistical query model. In *32nd ACM STOC*, pages 435–440. ACM Press, May 2000.

[13] D. Boneh, D. J. Wu, and J. Zimmerman. Immunizing multilinear maps against zeroizing attacks. Cryptology ePrint Archive, Report 2014/930, 2014. `http://eprint.iacr.org/2014/930`.

[14] Z. Brakerski, C. Gentry, S. Halevi, T. Lepoint, A. Sahai, and M. Tibouchi. Cryptanalysis of the quadratic zero-testing of GGH. Cryptology ePrint Archive, Report 2015/845, 2015. `http://eprint.iacr.org/2015/845`.

[15] P. Campbell, M. Groves, and D. Shepherd. Soliloquy: A cautionary tale. ETSI 2nd Quantum-Safe Crypto Workshop. 2014, `http://docbox.etsi.org/Workshop/2014/201410_CRYPTO/S07_Systems_and_Attacks/S07_Groves_Annex.pdf`.

[16] Y. Chen and P. Q. Nguyen. BKZ 2.0: Better lattice security estimates. In D. H. Lee and X. Wang, editors, *ASIACRYPT 2011*, volume 7073 of *LNCS*, pages 1–20. Springer, Heidelberg, Dec. 2011.

[17] J. H. Cheon, K. Han, C. Lee, H. Ryu, and D. Stehlé. Cryptanalysis of the multilinear map over the integers. In E. Oswald and M. Fischlin, editors, *EUROCRYPT 2015, Part I*, volume 9056 of *LNCS*, pages 3–12. Springer, Heidelberg, Apr. 2015.

[18] J. H. Cheon and C. Lee. Cryptanalysis of the multilinear map on the ideal lattices. Cryptology ePrint Archive, Report 2015/461, 2015. `http://eprint.iacr.org/2015/461`.

[19] J. H. Cheon, C. Lee, and H. Ryu. Cryptanalysis of the new clt multilinear maps. Cryptology ePrint Archive, Report 2015/934, 2015. `http://eprint.iacr.org/`.

[20] G. Chunsheng. Ideal multilinear maps based on ideal lattices. Cryptology ePrint Archive, Report 2015/269, 2015. `http://eprint.iacr.org/2015/269`.

[21] J.-S. Coron. Cryptanalysis of ggh15 multilinear maps. Cryptology ePrint Archive, Report 2015/1037, 2015. `http://eprint.iacr.org/`.

[22] J.-S. Coron, C. Gentry, S. Halevi, T. Lepoint, H. K. Maji, E. Miles, M. Raykova, A. Sahai, and M. Tibouchi. Zeroizing without low-level zeroes: New MMAP attacks and their limitations. In R. Gennaro and M. J. B. Robshaw, editors, *CRYPTO 2015, Part I*, volume 9215 of *LNCS*, pages 247–266. Springer, Heidelberg, Aug. 2015.

[23] J.-S. Coron, T. Lepoint, and M. Tibouchi. Practical multilinear maps over the integers. In R. Canetti and J. A. Garay, editors, *CRYPTO 2013, Part I*, volume 8042 of *LNCS*, pages 476–493. Springer, Heidelberg, Aug. 2013.

[24] J.-S. Coron, T. Lepoint, and M. Tibouchi. New multilinear maps over the integers. In R. Gennaro and M. J. B. Robshaw, editors, *CRYPTO 2015, Part I*, volume 9215 of *LNCS*, pages 267–286. Springer, Heidelberg, Aug. 2015.

[25] R. Cramer, L. Ducas, C. Peikert, and O. Regev. Recovering short generators of principal ideals in cyclotomic rings. Cryptology ePrint Archive, Report 2015/313, 2015. `http://eprint.iacr.org/2015/313`.

[26] A. Duc, F. Tramèr, and S. Vaudenay. Better algorithms for LWE and LWR. In E. Oswald and M. Fischlin, editors, *EUROCRYPT 2015, Part I*, volume 9056 of *LNCS*, pages 173–202. Springer, Heidelberg, Apr. 2015.

[27] K. Eisenträger, S. Hallgren, A. Kitaev, and F. Song. A quantum algorithm for computing the unit group of an arbitrary degree number field. In D. B. Shmoys, editor, *46th ACM STOC*, pages 293–302. ACM Press, May / June 2014.

[28] N. Gama, P. Q. Nguyen, and O. Regev. Lattice enumeration using extreme pruning. In H. Gilbert, editor, *EUROCRYPT 2010*, volume 6110 of *LNCS*, pages 257–278. Springer, Heidelberg, May 2010.

[29] S. Garg, C. Gentry, and S. Halevi. Candidate multilinear maps from ideal lattices. In T. Johansson and P. Q. Nguyen, editors, *EUROCRYPT 2013*, volume 7881 of *LNCS*, pages 1–17. Springer, Heidelberg, May 2013.

[30] S. Garg, C. Gentry, S. Halevi, M. Raykova, A. Sahai, and B. Waters. Candidate indistinguishability obfuscation and functional encryption for all circuits. In *54th FOCS*, pages 40–49. IEEE Computer Society Press, Oct. 2013.

[31] S. Garg, C. Gentry, S. Halevi, and M. Zhandry. Fully secure attribute based encryption from multilinear maps. Cryptology ePrint Archive, Report 2014/622, 2014. `http://eprint.iacr.org/2014/622`.

[32] C. Gentry. Fully homomorphic encryption using ideal lattices. In M. Mitzenmacher, editor, *41st ACM STOC*, pages 169–178. ACM Press, May / June 2009.

[33] C. Gentry, S. Gorbunov, and S. Halevi. Graph-induced multilinear maps from lattices. In Y. Dodis and J. B. Nielsen, editors, *TCC 2015, Part II*, volume 9015 of *LNCS*, pages 498–527. Springer, Heidelberg, Mar. 2015.

[34] O. Goldreich and S. Goldwasser. On the limits of nonapproximability of lattice problems. *J. Comput. Syst. Sci.*, 60(3):540–563, 2000.

[35] J. Groth and A. Sahai. Efficient non-interactive proof systems for bilinear groups. In N. P. Smart, editor, *EUROCRYPT 2008*, volume 4965 of *LNCS*, pages 415–432. Springer, Heidelberg, Apr. 2008.

[36] Q. Guo, T. Johansson, and C. Löndahl. Solving LPN using covering codes. In P. Sarkar and T. Iwata, editors, *ASIACRYPT 2014, Part I*, volume 8873 of *LNCS*, pages 1–20. Springer, Heidelberg, Dec. 2014.

[37] Q. Guo, T. Johansson, and P. Stankovski. Coded-BKW: Solving LWE using lattice codes. In R. Gennaro and M. J. B. Robshaw, editors, *CRYPTO 2015, Part I*, volume 9215 of *LNCS*, pages 23–42. Springer, Heidelberg, Aug. 2015.

[38] J. Hoffstein, J. Pipher, and J. H. Silverman. NTRU: A ring-based public key cryptosystem. In *Algorithmic Number Theory, Third International Symposium, ANTS-III, Portland, Oregon, USA, June 21-25, 1998, Proceedings*, pages 267–288, 1998.

[39] A. Joux. A one round protocol for tripartite Diffie-Hellman. *Journal of Cryptology*, 17(4):263–276, Sept. 2004.

[40] R. Kannan. Improved algorithms for integer programming and related lattice problems. In *Proceedings of STOC*, pages 193–206, 1983.

[41] S. Khot. Hardness of approximating the shortest vector problem in high Lp norms. In *44th FOCS*, pages 290–297. IEEE Computer Society Press, Oct. 2003.

[42] P. Kirchner and P.-A. Fouque. An improved BKW algorithm for LWE with applications to cryptography and lattices. In R. Gennaro and M. J. B. Robshaw, editors, *CRYPTO 2015, Part I*, volume 9215 of *LNCS*, pages 43–62. Springer, Heidelberg, Aug. 2015.

[43] T. Laarhoven. Sieving for shortest vectors in lattices using angular locality-sensitive hashing. In R. Gennaro and M. J. B. Robshaw, editors, *CRYPTO 2015, Part I*, volume 9215 of *LNCS*, pages 3–22. Springer, Heidelberg, Aug. 2015.

[44] A. Langlois, D. Stehlé, and R. Steinfeld. GGHLite: More efficient multilinear maps from ideal lattices. In P. Q. Nguyen and E. Oswald, editors, *EUROCRYPT 2014*, volume 8441 of *LNCS*, pages 239–256. Springer, Heidelberg, May 2014.

[45] A. Lenstra, J. Lenstra, H.W., and L. Lovsz. Factoring polynomials with rational coefficients. *Mathematische Annalen*, 261(4):515–534, 1982.

[46] R. Lindner and C. Peikert. Better key sizes (and attacks) for LWE-based encryption. In A. Kiayias, editor, *CT-RSA 2011*, volume 6558 of *LNCS*, pages 319–339. Springer, Heidelberg, Feb. 2011.

[47] V. Lyubashevsky, D. Micciancio, C. Peikert, and A. Rosen. SWIFFT: A modest proposal for FFT hashing. In K. Nyberg, editor, *FSE 2008*, volume 5086 of *LNCS*, pages 54–72. Springer, Heidelberg, Feb. 2008.

[48] V. Lyubashevsky, C. Peikert, and O. Regev. On ideal lattices and learning with errors over rings. In H. Gilbert, editor, *EUROCRYPT 2010*, volume 6110 of *LNCS*, pages 1–23. Springer, Heidelberg, May 2010.

[49] V. Lyubashevsky, C. Peikert, and O. Regev. On ideal lattices and learning with errors over rings. Cryptology ePrint Archive, Report 2012/230, 2012. `http://eprint.iacr.org/2012/230`.

[50] D. Micciancio and P. Voulgaris. A deterministic single exponential time algorithm for most lattice problems based on voronoi cell computations. In L. J. Schulman, editor, *42nd ACM STOC*, pages 351–358. ACM Press, June 2010.

[51] B. Minaud and P.-A. Fouque. Cryptanalysis of the new multilinear map over the integers. Cryptology ePrint Archive, Report 2015/941, 2015. `http://eprint.iacr.org/`.

[52] P. Q. Nguyen and T. Vidick. Sieve algorithms for the shortest vector problem are practical. *J. Mathematical Cryptology*, 2(2):181–207, 2008.

[53] C. Peikert. Public-key cryptosystems from the worst-case shortest vector problem: extended abstract. In M. Mitzenmacher, editor, *41st ACM STOC*, pages 333–342. ACM Press, May / June 2009.

[54] A. Pellet-Mary and D. Stehle. Cryptanalysis of Gu's ideal multilinear map. Cryptology ePrint Archive, Report 2015/759, 2015. `http://eprint.iacr.org/2015/759`.

[55] O. Regev. On lattices, learning with errors, random linear codes, and cryptography. In H. N. Gabow and R. Fagin, editors, *37th ACM STOC*, pages 84–93. ACM Press, May 2005.

[56] A. Sahai and B. R. Waters. Fuzzy identity-based encryption. In R. Cramer, editor, *EUROCRYPT 2005*, volume 3494 of *LNCS*, pages 457–473. Springer, Heidelberg, May 2005.

[57] C. Schnorr. A hierarchy of polynomial time lattice basis reduction algorithms. *Theor. Comput. Sci.*, 53:201–224, 1987.

[58] V. Shoup. Ntl: A library for doing number theory, 2001. `www.shoup.net/ntl/`.

[59] N. P. Smart and F. Vercauteren. Fully homomorphic encryption with relatively small key and ciphertext sizes. In P. Q. Nguyen and D. Pointcheval, editors, *PKC 2010*, volume 6056 of *LNCS*, pages 420–443. Springer, Heidelberg, May 2010.