VERSITA

TATRA MOUNTAINS
Mathematical Publications

DOI: 10.2478/v10127-012-0040-x
Tatra Mt. Math. Publ. **53** (2012), 81–102

A SELECTION OF RECENT LATTICE-BASED SIGNATURE AND ENCRYPTION SCHEMES

Rachid El Bansarkhani — Daniel Cabarcas — Po-Chun Kuo–
–Patrick Schmidt — Michael Schneider

ABSTRACT. It is known that the development of quantum computers will break the cryptographic schemes that are in use today. Since Shor's algorithm is able to solve the factoring and discrete logarithm problems, all cryptographic systems based on these two problems will get broken in the presence of large-scale quantum computers. Lattice-based schemes, however, are considered secure against attacks with these new machines. In this paper we present an overview of lattice-based cryptosystems, showing the most recent and the most promising candidates for encryption and signatures based on lattice problems. We explain the advantages and disadvantages of the cryptographic schemes. We also adjoin details about zero knowledge identification. With this work we try to give insight to one of the most promising candidates of future cryptography, for the time when potential quantum computers exist. We also point out drawbacks of these systems, which discloses directions for future work in lattice-based cryptography.

1. Introduction

Lattice-based cryptography gained a lot of interest in the past few years. It is a very vivid field of research with numerous publications at the top conferences in the cryptographic community. Multiple workshops and schools are organized in this area. New cryptographic schemes based on lattices are invented and, as a highlight, the first fully homomorphic encryption scheme has been developed based on the hardness of lattice problems.

As of today, the security of cryptosystems that are used in the wild is based on well-known number theoretic problems, namely the factorization problem and the problem of computing discrete logarithms. Algorithms for solving these two problems have been studied for decades. The fastest classical algorithms have runtime that is sub-exponential in the main security parameter. On quantum

2010 Mathematics Subject Classification: 68R01, 11C99.
Keywords: lattice-based cryptography, signatures, encryption.
This work was supported by CASED (www.cased.de).

computers, however, Shor's algorithm is able to solve both problems in polynomial runtime. Therefore, the construction of powerful quantum computers will threaten modern cryptography.

In contrast, the security of lattice-based cryptosystems is based on problems that, so far, cannot be attacked with quantum computers. Even on classical computers there is no algorithm for solving these problems which runs faster than exponential in the main security parameter. Lattice-based cryptography belongs to the field of *post-quantum cryptography*, which furthermore consists of cryptosystems based on problems in coding theory, cryptosystems based on multivariate quadratic equation systems and signature schemes based on hash functions.

Compared to the other candidates of the post-quantum era, lattice-based cryptography has one important advantage: The security of lattice schemes can be reduced to *worst-case* problems, whereas in other areas of cryptography security is based on *average-case* problems only. This property distinguishes lattice-based cryptography not only from the other post-quantum areas, but from all used candidates in cryptographic practice.

There are some more advantages of lattice-based cryptography. Most of the schemes in this area only require very few and easy operations in order to compute signatures or ciphertexts. The operations in use are products of matrices with vectors, sums of vectors, or multiplication of polynomials. These operations are faster than, for example, exponentiation, that is used in current classical systems like RSA or ElGamal.

The first lattice-based schemes date back to the mid 90s. The first secure scheme equipped with a hardness proof based on lattice problems was the hash function by Ajtai [Ajt96]. Its security was based on the *Short Integer Solution* problem (SIS). Hoffstein, Pipher and Silverman proposed NTRU encryption and signature scheme [HPS96]. Goldreich, Goldwasser and Halevi proposed the GGH encryption scheme [GGH97]. Of those early schemes only NTRU encryption and Ajtai's hash function remain secure. In subsequent years, Ajtai's original work was improved [GGH97], [MR07]. In 2005, Regev proposed a new lattice problem, namely the *Learning with Errors* problem (LWE). It allowed the construction of worst-case reductions for encryption schemes [Reg05]. Development of lattice-based cryptography until 2008 is well described in [BDS08]. After 2008, there were many new proposals, e.g., signature schemes [LM08], [Lyu09], [Lyu12], encryption schemes [SSTX09], [LP11], [SS11], fully homomorphic encryption schemes [Gen09], [BV11], [BGV11], and many more. We decided to give this overview since the current schemes are already very promising and allow a good overview of what is possible based on lattice problems. In addition, reviewing what has been done so far will reveal drawbacks that can be avoided in the future.

Our contribution

In this paper we present an overview of the most recent and most promising cryptographic schemes with hardness based on lattice problems. More exactly, we present the following schemes:

- the treeless signature scheme in [Lyu12],
- the trapdoor-signature scheme in [GPV08] using the trapdoor of [MP12],
- the LWE encryption scheme in [LP11],
- the provably secure NTRU encryption scheme in [SS11], and
- the identification scheme by zero-knowledge proofs in [KTX08].

We discuss their advantages and disadvantages and try to motivate why these schemes are valuable candidates for future cryptography.

Organization of the paper

The remainder of the paper is organized as follows. We present the required background on lattices and cryptography as well as our notation in Section 2. The description of the selected cryptographic signature and encryption schemes is shown in Section 3 and Section 4, respectively. Further schemes are detailed in Section 5. Finally we present a conclusion in Section 6.

2. Preliminaries

Originally, most lattice schemes are developed over integer lattices, i.e., the schemes mostly deal with matrices and vectors as elements. In order to have more efficient computations and save storage, most of the schemes were later transformed to the ring setting, namely dealing with polynomials over rings instead. Here we deal with the more efficient ring variants of the schemes. Only the trapdoor signature and the identification scheme are matrix-based.

We will use the polynomial rings $R = \mathbb{Z}[x]/\langle f(x)\rangle$ and $R_q = \mathbb{Z}_q[x]/\langle f(x)\rangle$ for a polynomial $f(x)$ that is monic and irreducible over $\mathbb{Z}$. An example choice for $f(x)$ is $f(x) = x^n + 1$ for n being a power of 2. This is the most commonly used polynomial ring in cryptography. Other possible choices are $f(x) = \sum_{i=0}^{n} x^i$ for $n + 1$ being prime.

Ring elements are denoted $\mathbf{p}$, whereas vectors of ring elements are written as $\hat{\mathbf{p}}$. $[m]$ denotes the set $\{1, \ldots, m\}$ and $a||b$ is the usual string concatenation of a and b.

For $\mathbf{x} \in R_q$ with $\mathbf{x} = x_0 + x_1 x + \cdots + x_{n-1}x^{n-1} \cong (x_0, \ldots, x_{n-1})^T$ let $\|\mathbf{x}\|_\infty := \max_{i=0,\ldots,n-1}(|x_i|)$ denote the ℓ_∞-norm, let $\|\mathbf{x}\|_1 := |x_0| + \cdots + |x_{n-1}|$

denote the ℓ_1-norm, and for $p > 1$ let $\|\mathbf{x}\|_p := \left(\sum_{i=0}^{n-1} |x_i|^p\right)^{1/p}$ denote the ℓ_p-norm. The ℓ_2-norm usually is referred to as Euclidean norm.

For choosing ("sampling") elements of sets or according to distributions, let $x \xleftarrow{\$} S$ denote the sampling of a uniformly random chosen element x from the set S, i.e., x is the output of a uniformly random choice of an element out of the set S, and let $x \leftarrow D$ denote the sampling of the element x according to the distribution D, i.e., the element x is chosen as an output of an algorithm which has distribution D as its output distribution for elements of an underlying set.

As needed in later sections, let $\mathbf{x} \leftarrow D^n_{\mathbf{v},\sigma}$ denote the sampling of $\mathbf{x}$ according to an n-dimensional discrete Gaussian distribution centered at $\mathbf{v}$ with standard deviation σ, and let

$$D^n_{\mathbf{v},\sigma}(\mathbf{x}) := \rho^n_{\mathbf{v},\sigma}(\mathbf{x}) \Big/ \sum_{\mathbf{z}\in\mathbb{Z}^n} \rho^n_{\mathbf{v},\sigma}(\mathbf{z})$$

with

$$\rho^n_{\mathbf{v},\sigma}(\mathbf{x}) := \exp\left(-(||\mathbf{x}-\mathbf{v}||/\sigma)^2/2\right) / (\sigma\sqrt{2\pi})^n$$

be the probability for this event. For brevity we write D^n_σ for $D^n_{\mathbf{0},\sigma}$.

Furthermore, we sample vectors from discrete Gaussian distributions over lattices. That is $\mathbf{x} \leftarrow D_{\mathcal{L},\sigma}$, where $\mathbf{x} \in \mathcal{L}$ with standard deviation σ.

2.1. The ring-LWE problem

Here we state the ring-LWE problem. The ring-LWE problem defined in [LPR10] is the adaption of the well-known LWE problem (for vectors or matrices, respectively) to polynomial rings.

Let $q \geq 2$ be an integer modulus, let $n > 1$ be the degree of the polynomial $f(x)$ defining R_q, and let χ be an error distribution (the distribution χ will be the discrete Gaussian error distribution in most cases).

In the general LWE setting, a vector $\mathbf{s} \in \mathbb{Z}_q^n$ (the secret) is given, and a vector $\mathbf{a} \in \mathbb{Z}_q^n$ is chosen uniformly at random. Furthermore, two things are computed: an error term e according to the ("random") error distribution χ, i.e., $e \leftarrow \chi$, and a pair $(\mathbf{a}, t)$ with $t = \langle \mathbf{a}|\mathbf{s}\rangle + e \mod q$.

In the search variant of the LWE problem, one has to find the vector $\mathbf{s}$ when given an arbitrary number of sample pairs $(\mathbf{a}_i, t_i)$. In the decision variant, one is asked to distinguish between arbitrary numbers of LWE sample pairs $(\mathbf{a}_i, t_i)$ and uniformly drawn samples $(\mathbf{a}_i, t_i)$ from $\mathbb{Z}_q^n \times \mathbb{Z}_q$.

The hardness of the (matrix) LWE problem is discussed in [Reg05], [Pei09]. Two practical attacks, the so-called distinguishing and decoding attacks on LWE--based cryptosystems, are described in [LP11].

2.2. The SIS problem over rings

The search variant of the *Small Integer Solution* problem ℓ_p-SIS$_{q,m,\beta,f}$ with ℓ_p-norm over rings and with parameters q, m, β, f is defined as follows: Given $n = \deg\big(f(x)\big)$, and m polynomials $\mathbf{g}_1, \ldots, \mathbf{g}_m$ chosen uniformly and independently from $R_q = \mathbb{Z}_q[x]/\langle f(x)\rangle$, find polynomials $\mathbf{e}_1, \ldots, \mathbf{e}_m \in \mathbb{Z}[x]$ not all zero such that $\sum_{i=1}^{m} \mathbf{e}_i \mathbf{g}_i = \mathbf{0} \in R_q$ and $\|\mathbf{e}\|_p \leq \beta$ with $\mathbf{e} = (\mathbf{e}_1^T, \ldots, \mathbf{e}_m^T)^T$. Informally speaking, in the SIS problem one is asked to find a small integral and non-zero element of the kernel of the function $g : \big(\mathbb{Z}[x]\big)^m \to R_q, (\mathbf{e}_1, \ldots, \mathbf{e}_m) \mapsto \sum_{i=0}^{m-1} \mathbf{g}_i \mathbf{e}_i$. Mostly when just writing SIS, we mean the search variant of SIS.

In the decisional variant of the SIS problem, one is given either the set $\big((\mathbf{g}_1, \ldots, \mathbf{g}_m), \mathbf{e}\big)$ uniformly chosen from $R_q^m \times R_q$ or $\big((\mathbf{g}_1, \ldots, \mathbf{g}_m), \mathbf{t}\big)$, where all $\mathbf{g}_i$ are uniformly chosen from R_q, but $\mathbf{t} = \sum_{i=0}^{m-1} \mathbf{g}_i \mathbf{e}_i$ for small integral elements $\mathbf{e}_i$, not all zero. The task is to distinguish both cases with non-negligible advantage, i.e., to state the correct case with probability significantly different from simply guessing.

In the work of [LM06] and [SSTX09], the authors show reduction proofs in the infinity and Euclidean norm from the shortest vector problem (SVP) to SIS over rings. An algorithm that solves SIS can be used to solve SVP for polynomial approximation factors in ideal lattices in the worst case.

3. Signature schemes

In this section we present the two recent, very efficient and most promising signature schemes based on hard lattice problems, i.e., the treeless signature scheme of [Lyu12] and the trapdoor-signature scheme of [GPV08] using the new trapdoor of [MP12].

3.1. Treeless signatures

The first provably secure signature scheme, whose security was based on ideal lattice problems, was introduced by Lyubashevsky and Micciancio in [LM08]. Unfortunately, this scheme was only able to produce one-time signatures; combined with a (hash) tree structure, one could obtain a signature scheme which allowed to sign a limited number of messages. To address and solve this limitation, Lyubashevsky removed the tree structure and developed several versions of an unbounded signature scheme, called "treeless signature scheme" (TSS), in a series of works [Lyu08b], [Lyu09], [Lyu12], which is provably secure in the Random Oracle Model.

The today's most efficient (ring) variant of the treeless signature scheme was proposed in [Lyu12] and is parametrized as follows:

- the lattice dimension n, being a power of 2, and the modulus q, which define the underlying ring $R_q = \mathbb{Z}_q[x]/\langle f(x)\rangle$ with $f(x) = x^n + 1$,
- a number γ of polynomials in the signing key and an integral bound d on their coefficients,
- an integral bound κ on the size of the coefficients of the polynomials output by the Random Oracle H,
- the (real) standard deviation σ of a discrete Gaussian distribution used in the signature algorithm, and
- a (real) smoothing parameter M.

Define the set of signing keys $S := \{\mathbf{f} \in R_q : ||\mathbf{f}||_\infty \leq d\}$ and the set $\{\mathbf{f} \in R_q : ||\mathbf{f}||_\infty \leq 1 \text{ and } ||\mathbf{f}||_1 \leq \kappa\}$ of outputs of the Random Oracle H for inputs from $\{0,1\}^*$, then the treeless signature scheme can be described as follows:

KeyGen(1^n): Sample $\hat{\mathbf{s}} = (\mathbf{s}_1, \ldots, \mathbf{s}_\gamma) \xleftarrow{\$} S^\gamma$ and $\hat{\mathbf{a}} = (\mathbf{a}_1, \ldots, \mathbf{a}_\gamma) \xleftarrow{\$} R_q^\gamma$. Output signing key $(\hat{\mathbf{a}}, \hat{\mathbf{s}})$ and verification key $(\hat{\mathbf{a}}, \mathbf{t})$ with $\mathbf{t} := \sum_{i=1}^{\gamma} \mathbf{a}_i \mathbf{s}_i$.

Sign($\mu, (\hat{\mathbf{a}}, \hat{\mathbf{s}})$): Sample $\mathbf{y}_1, \ldots, \mathbf{y}_\gamma \leftarrow D_\sigma^n$ and compute $\mathbf{c} = H\big(\sum_{i=1}^{\gamma} \mathbf{a}_i \mathbf{y}_i || \mu\big)$ and $\hat{\mathbf{z}} = (\mathbf{z}_1, \ldots, \mathbf{z}_\gamma)$ with $\mathbf{z}_i = \mathbf{s}_i \mathbf{c} + \mathbf{y}_i$. Output signature $(\hat{\mathbf{z}}, \mathbf{c})$ with probability

$$\min\left(1, \frac{D_\sigma^m(\overline{\mathbf{z}})}{M D_{\overline{\mathbf{c}},\sigma}^m(\overline{\mathbf{z}})}\right), \tag{1}$$

where $m = \gamma n$, and $\overline{\mathbf{z}} := \big(\mathbf{z}_1^T, \ldots, \mathbf{z}_\gamma^T\big)^T$ and $\overline{\mathbf{c}} := \big((\mathbf{s}_1\mathbf{c})^T, \ldots, (\mathbf{s}_\gamma\mathbf{c})^T\big)^T$ are m-dimensional vectors.

Verify($\mu, (\hat{\mathbf{z}}, \mathbf{c}), (\hat{\mathbf{a}}, \mathbf{t})$): Check if $\|\overline{\mathbf{z}}\| \leq 2\sigma\sqrt{m}$ and $\mathbf{c} = H\big(\sum_{i=1}^{\gamma} \mathbf{a}_i \mathbf{z}_i - \mathbf{t}\mathbf{c} || \mu\big)$ hold. If so, output 1 (accept), otherwise 0 (reject).

Concrete parameter choices for this scheme can be found in Table 1, where the security level for the instantiations is for $\delta = 1.007$ [Lyu12]. The parameter $M = \exp\big(12d\kappa\sqrt{m}/\sigma + (d\kappa\sqrt{m}/(2\sigma))^2\big)$ used in the output-condition (1) of the signing algorithm is needed to decouple the distribution of the signature $(\hat{\mathbf{z}}, \mathbf{c})$ from the distribution of the secret key $\hat{\mathbf{s}}$.

The security of the treeless signature scheme is based on the hardness of the ring equivalent of the ℓ_2-SIS$_{q,\gamma,\beta,f}$ search problem with $\beta = (4\sigma + 2d'\kappa)\sqrt{m}$ for $d' = (2\alpha+1)d + \alpha$ and for some positive integer α, and on the hardness of the decisional variant of the ring-SIS$_{q,\gamma,d,f}$ problem.

The sizes for a signing key, a verification key, and a signature are (in bits):

	Signing Key	Verification Key	Signature
TSS	$2\gamma n \cdot \lceil \log_2(q) \rceil$	$(\gamma+1)n \cdot \lceil \log_2(q) \rceil$	$(\gamma+1)n \cdot \lceil \log_2(q) \rceil$

TABLE 1. Treeless Signature Scheme: Parameters according to [Lyu12], Fig. 2, columns IV and V.

n	512	512
q	2^{24}	2^{31}
d	1	31
$m = 2n$	1024	1024
κ s.t. $2^\kappa \binom{n}{\kappa} \geq 2^{100}$	14	14
$\sigma = 12d\kappa\sqrt{m}$	5376	166656
$M = \exp\left(12d\kappa\sqrt{m}/\sigma + (d\kappa\sqrt{m}/(2\sigma))^2\right)$	2.72	2.72
approx. signature size (bits) $\approx m \log(12\sigma)$	16500	20500
approx. signing key size (bits) $\approx m \log(2d+1)$	2896	11585
approx. verification key size (bits) $\approx n \log q$	23170	32768

For practical parameters, i.e., $n = 512$, $q \approx n^4$ and $\gamma = 2$, the signing key as well as the verification key have a size of about 18 kilobytes and the signature is of about 10 kilobytes in size.

The number of required operations for key generation, signature creation and verification are:

	Gauss-samplings	Polynomial mult.	Polynomial add.
Key generation	0	γ	$\gamma - 1$
Signing	$k(\gamma + 2)$	$k\gamma$	$k(2\gamma - 1)$
Verification	0	γ	γ

The parameter k denotes the number of sampling rounds during signature creation (due to the output probability which can result in rejections of produced signatures $(\hat{\mathbf{z}}, \mathbf{c})$) and typically is $k \leq 7$.

The advantages of the ring variant of the treeless signature scheme are small key and signature sizes which are (up to some factor) quasi-linear in the security parameter n: The signature key is of size $2\gamma n \cdot \lceil \log_2(q) \rceil$, and the verification key and the signature are of size $(\gamma+1)n \cdot \lceil \log_2(q) \rceil$, i.e., all sizes being $\tilde{O}(n)$. A second advantage are the small number of operations that have to be performed to create the keys, or compute or verify a signature.

On the other hand, during the signing process we have a conditional output of the produced signature which results in a probabilistic signature creation due to the unknown rejection rate. Furthermore, the scheme itself is proven to be

secure in the Random Oracle Model, but not in the standard model. Thus, practical instantiations may lose the security properties provided by the authors. As a last point, there is no equivalence proof for the decisional and the search variant of the ring-SIS$_{q,\gamma,d,f}$ problem for d chosen according to [Lyu12] ("low-density SIS") since all equivalence relations are only shown for d being polynomial in the security parameter n. Thus, we cannot simply relate the security of the ring variant of the treeless signature scheme to a single problem like ring-SIS$_{q,\gamma,d,f}$, as is done for the matrix version of the scheme.

3.2. Trapdoor signatures

The signature scheme due to Gentry, Peikert and Vaikuntanathan [GPV08] consists mainly of sampling a preimage from a hash function featured with a trapdoor. The security of this construction is based on the hardness of SIS. In [MP12] Micciancio and Peikert provided a new trapdoor notion that improves all relevant bounds of the previous proposals [GPV08], [Ajt99], [AP09]. Similar to the constructions of [Ajt99], [AP09] they start with a uniform random matrix $\bar{\mathbf{A}}$ and extend it to a matrix $\mathbf{A} = [\bar{\mathbf{A}}|\mathbf{TG} - \bar{\mathbf{A}}\mathbf{R}]$ via deterministic transformations. The main idea behind this proposal is to use a primitive matrix $\mathbf{G}$ generating $\mathbb{Z}_q^n$ and for which one can easily sample preimages and find a basis $\mathbf{S}$ satisfying the congruence relation $\mathbf{G} \cdot \mathbf{S} \equiv \mathbf{0}$. Starting from the primitive vector $\mathbf{g}^T := (1, 2, 4, \ldots, 2^{k-1}) \in \mathbb{Z}_q^k$ where $k = \lceil \log_2 q \rceil$ one can find an associated basis $\mathbf{S_k}$ for the lattice $\Lambda_q^{\perp}(\mathbf{g}^T)$ which is defined by

$$\mathbf{S_k} = \begin{bmatrix} 2 & & & 0 \\ -1 & 2 & & \\ & \ddots & \ddots & \\ 0 & & -1 & 2 \end{bmatrix}.$$

From this vector $\mathbf{g}^T$ and the associated basis $\mathbf{S_k}$ one can easily create $\mathbf{S} \in \mathbb{Z}_q^{nk \times nk}$ and the parity check matrix $\mathbf{G} \in \mathbb{Z}_q^{n \times nk}$, respectively:

$$\mathbf{G} = \begin{bmatrix} \mathbf{g}^T & & & 0 \\ & \mathbf{g}^T & & \\ & & \ddots & \\ 0 & & & \mathbf{g}^T \end{bmatrix}, \qquad \mathbf{S} = \begin{bmatrix} \mathbf{S_k} & & & 0 \\ & \mathbf{S_k} & & \\ & & \ddots & \\ 0 & & & \mathbf{S_k} \end{bmatrix}.$$

In what follows we describe the preimage sampling algorithm for a syndrome $\mathbf{u}$ from the q-ary lattice $\Lambda_{\mathbf{u}}^{\perp}(\mathbf{G}) = \{\mathbf{x} \mid \mathbf{G} \cdot \mathbf{x} \equiv \mathbf{u} \mod q\}$ using the randomized nearest plane algorithm. Due to the niceness of the orthogonalized basis the algorithm reduces to a few steps where $a_0 = u$:

for $\mathsf{i} = 0, \ldots, \mathsf{k}-1$ do :

(1) $v_i \leftarrow D_{2\mathbb{Z}^n + a_i, s}$

(2) $a_{i+1} = \frac{a_i - v_i}{2}$

Output: $(v_0, \ldots, v_{k-1})^T$

The authors of [MP12] provide two different types of instantiations for the trapdoor generation algorithm, namely the statistical and computational one, where each of them uses a different distribution to sample the trapdoor $\mathbf{R}$.

In order to use the signature scheme of [GPV08] it is required to sample a preimage from a spherical discrete Gaussian for a given syndrome $\mathbf{u} \in \mathbb{Z}_q^n$ using the trapdoor $\mathbf{R}$. The Gaussian sampling algorithm mainly consists of two parts. The first part involves the trapdoor $\mathbf{R}$ which is used to transform a sample $\mathbf{x}$ from the primitive lattice $\Lambda_{\mathbf{u}}^{\perp}(\mathbf{G})$ with parameter

$$r \geq \| \mathbf{S} \| \cdot \sqrt{ln\big(2n(1+\frac{1}{\epsilon})\big)/\pi} \approx 9$$

to a sample $\mathbf{y} = [\begin{smallmatrix}\mathbf{R}\\ \mathbf{I}\end{smallmatrix}] \cdot \mathbf{x}$ of the lattice $\Lambda_{\mathbf{u}}^{\perp}(\mathbf{A})$. Sampling from a lattice Λ according to the discrete Gaussian distribution with parameter t means to use an appropriate sampling algorithm (e.g., rejection sampling) in order to get a sample $\mathbf{x} \in \Lambda$ from the distribution $D_{\Lambda,t}$ with probability $\rho_t(\mathbf{x})/\rho_t(\Lambda)$ where $\rho_t(\mathbf{x}) = e^{-\pi\|\mathbf{x}\|/t^2}$ denotes the standard n-dimensional Gaussian function. Due to the fact that $[\begin{smallmatrix}\mathbf{R}\\ \mathbf{I}\end{smallmatrix}]$ is not squared and the distribution of $\mathbf{y}$ with covariance $\mathbf{COV} = r^2 [\begin{smallmatrix}\mathbf{R}\\ \mathbf{I}\end{smallmatrix}] [\mathbf{R}^\top \ \mathbf{I}]$ is skewed it leaks information about the trapdoor. An attacker could collect some samples and reconstruct the covariance matrix. Therefore we need the second part to correct this flaw. This can be done by adding some perturbations from a properly chosen distribution. Using the convolution technique from [Pei10] we can choose a parameter s that is slightly larger than the largest eigenvalue of the covariance matrix $\mathbf{COV}$, and generate Gaussian pertubations $\mathbf{p} \in \mathbb{Z}^m$ having covariance $\Sigma_p = s^2\mathbf{I} - \mathbf{COV}$. The square root $\sqrt{\Sigma_p}$ can be computed via cholesky decomposition. In order to obtain a vector $\mathbf{b}$ that is from a spherical Gaussian with parameter s it is required to sample a preimage $\mathbf{y}$ for an adjusted syndrome $\mathbf{a} = \mathbf{u} - \mathbf{Ap}$ from $\Lambda_{\mathbf{u}}^{\perp}(\mathbf{A})$. Then $\mathbf{b} = \mathbf{p} + \mathbf{y}$ provides a spherical distributed sample satisfying $\mathbf{Ab} \equiv \mathbf{u} \mod q$.

The following signature scheme is the same as in [GPV08] except that the key generation algorithm of [MP12] is used. The main idea is to sample a preimage $\mathbf{x}$ for the hash value $H(\mu)$ of the message μ so that $\mathbf{A} \cdot \mathbf{x} \equiv H(\mu)$, where $H(\cdot)$ denotes a random oracle.

KeyGen(1^n): Sample $\bar{\mathbf{A}} \xleftarrow{\$} \mathbb{Z}_q^{n \times \bar{m}}$ and $\mathbf{R} \xleftarrow{\$} D$, where $\mathbf{R} \in \mathbb{Z}^{\bar{m} \times w}$, $w = \lceil \log_2(q) \rceil \cdot n$ and D a distribution, which depends on the instantiation. Output the signing key $\mathbf{R}$ and the verification key $\mathbf{A} = [\bar{\mathbf{A}} | \mathbf{TG} - \bar{\mathbf{A}}\mathbf{R}] \in \mathbb{Z}_q^{n \times m}$, where $\mathbf{G}$ is a primitive matrix and $\mathbf{T} \in \mathbb{Z}_q^{n \times m}$ is an invertible matrix.

Sign($\mu, \mathbf{R}$): Compute syndrome $\mathbf{u} = H(\mu)$, sample $\mathbf{p} \leftarrow D_{\mathbb{Z}^m, \sqrt{\Sigma_p}}$ and determine perturbed syndrome $\mathbf{v} = \mathbf{u} - \mathbf{A} \cdot \mathbf{p}$. Then sample $\mathbf{z} \leftarrow D_{\Lambda_{\mathbf{v}}^{\perp}(\mathbf{G}), r}$, where $r^2\mathbf{I}$ is the covariance matrix and

$$r \geq \| \mathbf{S} \| \cdot \sqrt{ln\left(2n\left(1 + \frac{1}{\epsilon}\right)\right) \Big/ \pi}.$$

Compute $\mathbf{x} = [\mathbf{p} + \begin{smallmatrix}\mathbf{R}\\ \mathbf{I}\end{smallmatrix} \mathbf{z}]$ and output the signature $(\mathbf{x}, s)$.

Verify($\mu, (\mathbf{x}, s), (H, \mathbf{A})$): Check whether $\mathbf{A} \cdot \mathbf{x} \equiv H(\mu) \mod q$ and $\|\mathbf{x}\| \leq s \cdot \sqrt{m}$. If so, output 1 (accept), otherwise 0 (reject).

This scheme has the following efficiency measures. The size of private and public key as well as of a signature are (in bits):

	Public Key	Private Key	Signature
Trapdoor [GPV08] [MP12]	nmk, $k = \lceil \log_2(q) \rceil$	$\bar{m}nk\big(1 + \lceil \log_2(4\sigma) \rceil\big)$, e.g., $\mathbf{R} \xleftarrow{\$} D_{\mathbb{Z}^{\bar{m} \times w}, \sigma}$	$m \cdot \lceil \big(1 + \log_2(s \cdot 4)\big) \rceil$, $s \approx (\sqrt{\bar{m}} + \sqrt{n}) \cdot \sigma \cdot r$

For key generation, signature creation and verification, the required operations are:

	Gauss-sampl.	Matrix-matrix mult.	Matrix-vector mult.	Vector add.	Cholesky decomp.
Key generation	1	2	0	0	1
Signing	2	0	3	2	0
Verification	0	0	1	0	0

4. Encryption schemes

In this section we present the two most promising lattice-based encryption schemes, namely the provably secure variant of the NTRU encryption scheme [SS11] as well as the ring-LWE based encryption system of [LP11].

4.1. LWE encryption

There are multiple encryption schemes whose security is based on the LWE problem. The development started with the work of R e g e v [Reg05], who presented a single-bit encryption scheme based on matrices. Followup schemes were

presented in [Mic10]. The work of [LPR10] presents the first ring-based variant, it means, a more efficient encryption scheme with hardness based on the ring-LWE problem.

In this section, we detail the scheme of Lindner and Peikert [LP11], which presents the most recent development in this line of research. The authors of [Mic10], [LP11] show the matrix variant and explain how to generate a polynomial (ring-based) variant of the scheme. Here we only present the polynomial variant due to the advantages in key size and runtime of this variant. This scheme has already been presented in [GFS12]. It also includes a practical implementation of the LWE scheme. The authors present a software implementation of the matrix and the ring variant of the LWE scheme as well as a hardware implementation on FPGA of the ring-based scheme.

Let χ_k and χ_e be error distributions over R for key generation and encryption. The LWE-Polynomial encryption is a tuple $(\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$, where

KeyGen(a): Choose $\mathbf{r}_1$, $\mathbf{r}_2 \leftarrow \chi_k$ and let $\mathbf{p} = \mathbf{r}_1 - \mathbf{a} \cdot \mathbf{r}_2$. Output public key $\mathbf{p}$ and secret key $\mathbf{r}_2$.

Enc($\mathbf{a}, \mathbf{p}, m \in \Sigma^n$): Choose $\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3 \leftarrow \chi_e$. Let $\bar{\mathbf{m}} = \mathsf{encode}(m) \in R_q$. The ciphertext is then $(\mathbf{c_1}, \mathbf{c_2}) \in R_q^2$ with $\mathbf{c_1} = \mathbf{a} \cdot \mathbf{e}_1 + \mathbf{e}_2$ and $\mathbf{c_2} = \mathbf{p} \cdot \mathbf{e}_1 + \mathbf{e}_3 + \bar{\mathbf{m}}$.

Dec($(\mathbf{c}_1, \mathbf{c}_2), \mathbf{r}_2$): Output $\mathsf{decode}(\mathbf{c}_1 \mathbf{r}_2 + \mathbf{c}_2)$.

The functions encode and decode compile messages from the message space to R_q (encode) and vice versa (decode). Both functions are detailed in [LP11] and [GFS12]. Choices for useful parameters can be found in [LPR10], [LP11] and [GFS12].

The decoding during decryption will fail with a certain probability. If $|\mathbf{e}_1 \cdot \mathbf{r}_1 + \mathbf{e}_2 \cdot \mathbf{r}_2 + \mathbf{e}_3|$ is bigger than the threshold $t = \lfloor q/4 \rfloor$, decoding will fail. The failure probability is depending on the error distributions χ_k and χ_e, which is upper bounded by a value δ. It is common to chose $\chi_k = \chi_e = \chi$ [LP11], [GFS12]. Different choices of the parameters n, q, c and s (s is the standard deviation of the Gaussian distribution χ) lead to different values of δ.

Following [GFS12], s should be chosen such that

$$s^2 = \frac{\sqrt{2}\pi}{c} \cdot \frac{t}{\sqrt{2n \cdot ln(2/\delta)}}.$$

The encryption system we present here is provably secure as long as the decision ring-LWE problem is hard. I.e., an attacker breaking the LWE-Polynomial encryption system is able to solve the ring-LWE problem instance, and thus is able to solve certain lattice problems in *all* lattices of a certain smaller dimension (the so-called *worst-case hardness*). So far, most researchers believe that LWE in polynomial rings is as hard as over the integers. Therefore, the restriction of the security of the scheme to ring-LWE is not a concern.

The size of private and public key as well as the size of a cipher are (in bits):

	Public Key	Private Key	Ciphertext
LWE $q \approx 5000$	$2n \cdot \lceil \log_2(q) \rceil$	$n \cdot \lceil \log_2(q) \rceil$	$2n \cdot \lceil \log_2(q) \rceil$

For key generation, computing a ciphertext, and decrypting it, the required operations are:

	Gauss-samplings	Polynomial mult.	Polynomial add.
Key generation	2	1	1
Encryption	3	2	3
Decryption	0	1	1

Here we left out the computation of encode and decode.

4.2. NTRU encrpytion

NTRUEncrypt is a lattice-based encryption scheme proposed in 1996 by Hoffstein, Pipher and Silverman [HPS96]. NTRU is a promising encryption scheme because of its efficiency and because it remains essentially unbroken after more than a decade. However, there exists no proof that breaking NTRU is as hard as the underlying lattice problem. Recently, Stehlé and Steinfeld [SS11] proposed a variant of NTRU and proved that breaking this variant is as hard as worst-case lattice problems. In this section we present this latest version of NTRU, which we refer to as NTRU-CPA.

The encryption scheme NTRU-CPA=(KeyGen,Enc,Dec) is specified by publicly known parameters as follows:

- the dimension $n > 8$, which must be a power of 2, determines the cyclotomic polynomial $f(x) = x^n + 1$ and the quotient ring $R := \mathbb{Z}[x]/\langle f(x) \rangle$,
- a prime $q > 5$ such that $q \equiv 1 \mod 2n$, determines the ciphertext space $R_q = R/qR$,
- a polynomial $\mathbf{p} \in R_q^\times$ with small coefficients (typically $\mathbf{p} = 2$, $\mathbf{p} = 3$ or $\mathbf{p} = x + 2$), determines the message space $R_p = R/\mathbf{p}R$,
- a distribution χ determines the ring-LWE noise,
- and a positive real σ determines the discrete Gaussian distribution $D_{\mathbb{Z}^n,\sigma}$ used for key generation.

NTRU-CPA key generation, encryption and decryption algorithms are defined as follows:

KeyGen(1^κ): Sample $\mathbf{f}'$ from $D_{\mathbb{Z}^n,\sigma}$, let $\mathbf{f} = \mathbf{pf}' + 1 \mod q$; if $\mathbf{f} \notin R_q^\times$ resample. Sample $\mathbf{g}$ from $D_{\mathbb{Z}^n,\sigma}$; if $\mathbf{g} \mod q \notin R_q^\times$ resample. Set secret key $sk := \mathbf{f}$ and public key $pk := \mathbf{h} := \mathbf{pg}/\mathbf{f} \in R_q$.

Enc(pk, m): Sample $\mathbf{s}, \mathbf{e} \in R_q$ from χ, and return ciphertext $\mathbf{c} := \mathbf{hs} + \mathbf{pe} + \mathbf{m} \in R_q$.

Dec($sk, \mathbf{c}$): Compute $\mathbf{c}' := \mathbf{f} \cdot \mathbf{c} \in R_q$ and return $\mathbf{c}' \mod \mathbf{p}$.

Parameter choices

Concrete parameters for NTRU-CPA can be selected as follows.

- Fix $\mathbf{p} = 2$ for simplicity and because it provides a useful message space.
- Set the distribution χ to be a discrete Gaussian distribution $D_{\mathbb{Z}^n,r}$,
- and choose $r = 8$ so that the discrete Gaussian approximates a continuous Gaussian well.
- Fix a value for n.
- Choose a prime q between $2^{21}r^2n^5\ln(n)$ and $2^{22}r^2n^6\ln(n)$ for correctness.
- Set $\sigma = 2n\sqrt{\ln(8nq)q}$.

For this choice of parameters, the public key is statistically close to uniform, thus an attack implies solving the ring-LWE problem. For the choice of n, the running time of the distinguishing attack as described in [MR08] is given by

$$\log_2(t_{adv}) = \frac{14.4n\log_2 q}{\log_2^2(c)} - 110, \tag{2}$$

where

$$c = \frac{q}{r}\sqrt{\ln(1/\epsilon)/\pi} \quad \text{and} \quad \epsilon$$

is the advantage of the adversary. Table 2 shows parameters as described above and the corresponding run time of an attack.

The NTRU-CPA scheme has the following efficiency measures. The size of private and public key as well as the size of a cipher in bits are:

	Public Key	Private Key	Ciphertext
NTRU	$n\lceil\log_2 q\rceil$	$n\lceil\log_2 q\rceil$	$n\lceil\log_2 q\rceil$

TABLE 2. Parameter values for NTRU-CPA and running time estimates for best known attacks. For given values of n, columns two through four show values for parameters q, σ and r that specify an instance of NTRU-CPA. For the given adversary advantage values, column six shows the estimated running time of a distinguishing attack.

Parameters				Advantage	Attack time [s]
n	$\log_2 q$	σ	r	$\log_2(1/\epsilon)$	$\log_2 t$
128	54.28	2.53×10^{11}	8	32	-76.67
256	59.47	3.20×10^{12}	8	32	-49.06
512	64.64	4.00×10^{13}	8	32	2.29
1024	69.79	4.96×10^{14}	8	32	98.24
2048	74.93	6.10×10^{15}	8	32	278.31

For key generation, computing a ciphertext, and decrypting it, the required operations are:

	Gauss-sampl.	Polyn.mult.	Polyn.add.	Polyn.inv.
Key generation	2	2	0	1
Encryption	2	2	2	0
Decryption	0	1	0	0

5. Further schemes

5.1. Zero knowledge

With strong security guarantees, lattice problems are also used to construct identification schemes, where zero-knowledge proofs are the main tool to construct secure identification protocols [FFS88]. The early study of zero-knowledge proofs based on lattice problems is due to Goldreich and Goldwasser [GG00], in which they showed that coGapSVP and coGapCVP are in the class of statistical zero-knowledge proofs (SZK). In 2003, Micciancio and Vadhan constructed a SZK protocol with efficient prover [MV03]. Moreover, Peikert and Vaikuntanathan proposed a non-interactive statistical zero-knowledge proof system with efficient prover based on the problems GapSIVP, GapCRP, and GapGSMP [PV08]. Later, Lyubashevsky [Lyu08a], as well as Kawachi, Tanaka, and Xagawa [KTX08], proposed concurrently-secure identification schemes based on lattice problems. Bendlin et al. [BD10] constructed

a scheme based on worst-case GapSVP, in which proofs of plaintext knowledge can be obtained using the LWE scheme by Regev [Reg05].

Here, we describe in more detail the scheme of Kawachi, Tanaka, and Xagawa [KTX08], which is essentially a variant of Stern's framework [Ste96] with the core string commitment technique replaced with one that is based on lattice problems. This scheme itself has several variants. For example, Xagawa and Tanaka rebuilt the scheme based on NTRU [XT09]; the variant of Cayrel et al., allows for a reduced soundness error [CLRS10]; the variant of Silva et al., incurs a lower communication cost and hence is suitable for practical use [SCL11]. We explain the basic variant which shows the construction and allows for simpler explanation than the followup versions.

First, we define the lattice-based hash function and string commitment, which will be used in zero-knowledge proof systems.

Lattice-based Hash: $f_{\mathbf{A}}(\mathbf{x}) = \mathbf{A}\mathbf{x} \bmod q$, where $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$, $q = q(n) = n^{O(1)}$, $m = m(n) > n \log q(n)$ and $\mathbf{x} \in \{0,1\}^m$.

This is a provably secure hash function [Ajt96], [GGH96] for suitably chosen m and q. That is, a collision in the hash function would imply solving the worst-case $\mathrm{GapSVP}^2_{\tilde{O}(n)}$ problem [MR07].

Lattice-based String Commitment $Com(s)$**:** Let $n, r, l \in \mathbb{Z}$ and $m = 2r$.
Step 1: $S \leftarrow pad(s)$, where the padding function *pad* could be according to the Merkle-Damgård construction.
Step 2: Cut S into $(S_0, \ldots, S_k)$, where $S_i \in \{0,1\}^{r-l}$.
Step 3: $H_0 = 0$ (or a fixed initialization vector).
Step 4: For $i = 0$ to k do $H_{i+1} \leftarrow f_{\mathbf{C}}(g(H_i)||S_i)$, where $g : \mathbb{Z}_q^n \to \{0,1\}^l$ is an efficiently invertible function and $f_{\mathbf{C}}$ is the lattice-based hash function for a uniformly random $\mathbf{C} \xleftarrow{\$} \mathbb{Z}_q^{n \times r}$.
Step 5: Output H_{k+1}.

Here, we note $Com : \{0,1\}^* \to \mathbb{Z}_q^n$; that is, the string commitment maps a string of *arbitrary* length to a vector over $\mathbb{Z}_q$.

With this, the identification scheme works as follows:

KeyGen(1^n)**:** For the security parameter n, choose $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$, a random vector $\mathbf{x} \in \{0,1\}^m$ such that $||\mathbf{x}||_1 = m/2$ (the hamming weight must be $m/2$) and $\mathbf{y} = f_{\mathbf{A}}(\mathbf{x})$. Output public key $(\mathbf{A}, \mathbf{y})$ and secret key $\mathbf{x}$.

Action$(\mathbf{A}, \mathbf{y})$**:** *Step Prover 1*: Choose a random permutation π over $[m]$ and a random vector $\mathbf{r} \in \mathbb{Z}_q^m$. Send commitments $\mathbf{c}_1, \mathbf{c}_2$ and $\mathbf{c}_3$:

- $\mathbf{c}_1 = Com\big(\pi, f_{\mathbf{A}}(\mathbf{r})\big)$,
- $\mathbf{c}_2 = Com\big(\pi(\mathbf{r})\big)$,
- $\mathbf{c}_3 = Com\big(\pi(\mathbf{x} + \mathbf{r})\big)$.

Step Verifier 1: Send a random challenge $Ch \in \{1, 2, 3\}$ to Prover.

Step Prover 2:

- If $Ch = 1$, send $\mathbf{s} = \pi(\mathbf{x})$ and $\mathbf{t} = \pi(\mathbf{r})$, which reveal $\mathbf{c}_2$ and $\mathbf{c}_3$, respectively.
- If $Ch = 2$, send $\phi = \pi$ and $\mathbf{u} = \mathbf{x} + \mathbf{r}$, which reveal $\mathbf{c}_1$ and $\mathbf{c}_3$, respectively.
- If $Ch = 3$, send $\phi = \pi$ and $\mathbf{v} = \mathbf{r}$, which reveal $\mathbf{c}_1$ and $\mathbf{c}_2$, respectively.

Step Verifier 2:

- If $Ch = 1$, accept if $\mathbf{c}_2 = Com(\mathbf{t})$, $\mathbf{c}_3 = Com(\mathbf{s}+\mathbf{t})$, and $||\mathbf{s}||_1 = m/2$.
- If $Ch = 2$, accept if $\mathbf{c}_1 = Com\big(\phi, f_{\mathbf{A}}(\mathbf{u}) - \mathbf{y}\big)$, $\mathbf{c}_3 = Com\big(\phi(\mathbf{u})\big)$.
- If $Ch = 3$, accept if $\mathbf{c}_1 = Com(\phi, \mathbf{Av})$, $\mathbf{c}_2 = Com\big(\phi(\mathbf{v})\big)$.

  Otherwise, reject.

It is easy to check that the soundness error of this scheme is 2/3.

Concerning the security of the scheme: Briefly speaking, breaking the identification scheme is to find a collision in the string commitment which implies a collision in the provably secure lattice hash function. Thus, the security of this scheme is based on the GapSVP problem, which is the decision version of the shortest vector problem. For details, we refer to the original paper [KTX08]. Finally, we summarize several state-of-the-art zero-knowledge identification schemes based on lattice problems in Table 3.

TABLE 3. Comparison of zero-knowledge identification schemes based on lattice problems for $n, m, q \in \mathbb{Z}$, the basis $\mathbf{A}, \mathbf{A_0}, \mathbf{A_1} \in \mathbb{Z}_q^{n\times m}$, $\mathbf{x} \in \mathbb{Z}_q^m$, $\mathbf{y} \in \mathbb{Z}_q^n$, and $\mathbf{a_h}, \mathbf{a_t}, \mathbf{x_h}, \mathbf{x_t} \in \mathbb{Z}_q[x]/(x^n - 1)$. Rounds means the number of rounds to reach soundness error $1/2^{16}$.

	PK	SK	Relation	γ in GapSVP^2_γ	Rounds
[MV03]	$\mathbf{A_0}, \mathbf{A_1}$	$\mathbf{x}$	$\mathbf{A_0x} = \mathbf{0}$ and $\mathbf{A_1x} = \mathbf{0}$	$\tilde{O}(n^{1.5})$	16
[PV08]	$\mathbf{A}, \mathbf{y}$	$\mathbf{x}$	$\mathbf{Ax} = \mathbf{y}$	$\tilde{O}(n^2)$	11
[KTX08]	$\mathbf{A}, \mathbf{y}$	$\mathbf{x}$	$\mathbf{Ax} = \mathbf{y}$ and hamming weight of $\mathbf{x}$ is $m/2$	$\tilde{O}(n)$	28
[CLRS10]	$\mathbf{A}, \mathbf{y}, Com$	$\mathbf{x}$	$\mathbf{Ax} = \mathbf{y}$	$\tilde{O}(n)$	17
[XT09]	$\mathbf{a_h}, \mathbf{a_t}, \mathbf{y}$	$\mathbf{x_h}$, $\mathbf{x_t}$	$\mathbf{a_h x_h} + \mathbf{a_t x_t} = \mathbf{y}$	Based on NTRU if $\mathbf{a_h} = -h$ and $\mathbf{a_t} = 1$	28

6. Conclusion

In this paper we presented the most recent, thus efficient and most promising lattice-based schemes, which are the treeless signature scheme and the trapdoor-signature scheme on the one hand and a provably secure version of NTRU and an LWE-based encryption scheme on the other hand. We provided a detailed description of the schemes and have taken a glance on the sizes of keys, ciphertexts or signatures, respectively. We furthermore reflected the number of operations needed for all schemes and summarized them in the appendix. As a roundup, we shed light on zero-knowledge identification schemes based on lattice problems. Of course, there are several disadvantages of the provided schemes which have to be considered. For instance, the key generation step of the trapdoor signature scheme involves the cholesky decomposition algorithm, which is very time consuming. Furthermore the storage sizes of the required matrices are very large compared to the other schemes.

Key generation of the NTRU-CPA requires sampling a discrete Gaussian distribution with large standard deviation. If done using standard techniques, such as rejection sampling, this sampling takes a very long time.

The treeless and the trapdoor signature schemes are both proven secure only in the Random Oracle Model. The signature generation of the treeless scheme fails with a certain error probability, which increases the signature generation time. The soundness error in the identification scheme is very large. This causes large communication costs, since the number of rounds has to be increased. The matrix-multiplications are slow as well. The identification scheme will be more efficient when instantiated with polynomials over rings instead of matrices. The LWE scheme is already quite efficient, both for storage and computations.

Acknowledgements. We are grateful to an anonymous referee for helpful comments. This work was supported in part by National Science Council under Grant NSC 99-2911-I-001-506 M i c h a e l S c h n e i d e r is supported by project BU 630/23-1 of the German Research Foundation (DFG). This work was supported by CASED (`www.cased.de`). We thank A n d r e a s H ü l s i n g for always nice discussions on the topic.

Appendix A. Overview tables

Here we collect the data from the single chapters to provide a full overview of the operations required inside the lattice-based schemes under supervision.

We omit the zero-knowledge identification scheme, since it is a more complex protocol than the other primitives "encryption" and "signature".

TABLE 4. Overview of the gathered data—sizes of keys, ciphertexts, and signatures in bits.

	Public Key	Private Key	Ciphertext	Signature
TSS	$(\gamma+1)n\lceil\log_2(q)\rceil$	$2\gamma n\lceil\log_2(q)\rceil$	–	$(\gamma+1)n\lceil\log_2(q)\rceil$
Trapdoor $k=\lceil\log_2(q)\rceil$	$nm\cdot k$	$n^2k\lceil\log_2(4n)\rceil$	–	$m\cdot\lceil\log_2(2s\sqrt{m})\rceil$ $s=2n^2k\sqrt{7}\cdot 4.5$
LWE $q\approx 5000$	$2n\cdot\lceil log_2(q)\rceil$	$n\cdot\lceil log_2(q)\rceil$	$2n\lceil log_2(q)\rceil$	–
NTRU-CPA	$n\lceil\log_2 q\rceil$	$n\lceil\log_2 q\rceil$	$n\lceil\log_2 q\rceil$	–

TABLE 5. Overview of the gathered data—operations required for computations.

		Gauss-sampl.	Polyn.mult.	Polyn.add.	Polyn.inv.
TSS	KeyGen	0	γ	$\gamma-1$	–
	Sign	$k(\gamma+2)$	$k\gamma$	$k(2\gamma-1)$	–
	Verify	0	γ	γ	–
LWE	KeyGen	2	1	1	–
	Enc	3	2	3	–
	Dec	0	1	1	–
NTRU-CPA	KeyGen	2	2	0	1
	Enc	2	2	2	0
	Dec	0	1	0	0

		Gauss-sampl.	Matrix mult.	Matrix-vector mult.	Vector add.	Cholesky decomp.
Trapdoor	KeyGen	1	2	0	0	1
	Sign	3	0	2	1	0
	Verify	0	0	1	0	0

REFERENCES

[Ajt96] AJTAI, M.: *Generating hard instances of lattice problems (extended abstract)*, in: Proc. of the 28th Annual ACM Symposium on the Theory of Comput.—STOC '96, Philadelphia, USA, 1996, ACM, New York, pp. 99–108.

[Ajt99] AJTAI, M.: *Generating hard instances of the short basis problem*, in: Automata, Languages and Programming, 26th Internat. Colloq.—ICALP '99 (J. Wiedermann et al., eds.), Prague, 1999, Lecture Notes in Comput. Sci., Vol. 1644, Springer, Berlin, 1999, pp. 1–9.

[AP09] ALWEN, J.—PEIKERT, CH.: *Generating shorter bases for hard random lattices*, in: 26th Internat. Symposium on Theoretical Aspects of Comput. Sci.—STACS '09 (S. Albers et al., eds.), Freiburg, Germany, 2009, LIPICS-Leibniz Internat. Proc. in Informatics, Vol. 3, Schloss Dagstuhl Leibniz Zentrum für Informatik, Wadern, 2009, pp. 75–86.

[BD10] BENDLIN, R.—DAMGÅRD, I.: *Threshold decryption and zero-knowledge proofs for lattice-based cryptosystems*, in: Theory of Cryptography, 7th Theory of Cryptography Conf.—TCC '10 (D. Micciancio, ed.), Zurich, Switzerland, 2010, Lecture Notes in Comput. Sci., Vol. 5978, Springer, Berlin, 2010, pp. 201–218.

[BDS08] BUCHMANN, J.—DAHMEN, E.—SCHNEIDER, M.: *Merkle tree traversal revisited*, in: Post-Guantum Cryptography, 2nd Internat. Workshop—PQCrypto '08 (J. Buchmann et al., eds.), Cincinnati, OH, USA, 2008, Lecture Notes in Comput. Sci., Vol. 5299, Springer, Berlin, 2008, pp. 63–78.

[BGV11] BRAKERSKI, Z.—GENTRY, C.—VAIKUNTANATHAN, V.: *Fully homomorphic encryption without bootstrapping*, Electronic Colloq. Comput. Complex. (ECCC) **18** (2011), p. 111.

[BV11] BRAKERSKI, Z.—VAIKUNTANATHAN, V.: *Fully homomorphic encryption from ring-LWE and security for key dependent messages*, in: Advances in Cryptology––CRYPTO '11, 31st Annual Cryptology Conf. (P. Rogaway, ed.), Santa Barbara, CA, USA, 2011, Lecture Notes in Comput. Sci., Vol. 6841, Springer, Berlin, 2011, pp. 505–524.

[CLRS10] CAYREL, P.-L.—LINDNER, R.—RÜCKERT, M.—SILVA, R.: *Improved zero-knowledge identification with lattices*, in: Provable Security, 4th Internat. Conf.––ProvSec '10 (S.H. Heng et al., eds.), Malacca, Malaysia, 2010, Lecture Notes in Comput. Sci., Vol. 6402, Springer, Berlin, 2010, pp. 1–17.

[FFS88] FEIGE, U.—FIAT, A.—SHAMIR, A.: *Zero-knowledge proofs of identity*, J. Cryptology **1** (1988), 77–94.

[Gen09] GENTRY, C.: *Fully homomorphic encryption using ideal lattices*, in: Proc. of the 41st Annual ACM Symposium on Theory of Comput.—STOC '09, (M. Mitzenmacher, ed.), Bethesda, MD, USA, 2009, ACM, New York, 2009, pp. 169–178.

[GFS12] GÖTTERT, N.—FELLER, T.—SCHNEIDER, M.—HUSS, S.A.—BUCHMANN, J.: *On the design of hardware building blocks for modern lattice-based encryption schemes*, in: Workshop on Cryptograph. Hardware and Embedded Syst.—CHES '12 (E. Prouff and P. Schaumont, eds.), Leuven, Belgium, Lecture Notes in Comput. Sci., Vol. 7428, Springer, Berlin, 2012, pp. 512–529.

[GG00] GOLDREICH, O.—GOLDWASSER, S.: *On the limits of nonapproximability of lattice problems*, J. Comput. System Sci. **60** (2000), 540–563.

[GGH96] GOLDREICH, O.—GOLDWASSER, S.—HALEVI, S.: *Public-key cryptosystems from lattice reduction problems*, Electronic Colloq. on Computational Complexity (ECCC) **3** (1996).

[GGH97] GOLDREICH, O.—GOLDWASSER, S.—HALEVI, S.: *Public-key cryptosystems from lattice reduction problems*, in: Advances in Cryptology—CRYPTO '97, 17th Annual Internat. Cryptology Conf. B. S. Kaliski, Jr., ed.), Santa Barbara, CA, USA, 1997, Lecture Notes in Comput. Sci., Vol. 1294, Springer, Berlin, 1997, pp. 112–131.

[GPV08] GENTRY, C.—PEIKERT, C.—VAIKUNTANATHAN, V.: *Trapdoors for hard lattices and new cryptographic constructions*, in: Proc. of the 40th Annual ACM Symposium on Theory of Comput.—STOC '08 Victoria, Canada, 2008, ACM, New York, pp. 197–206.

[HPS96] HOFFSTEIN, J.—PIPHER, J.—SILVERMAN, J. H.: *NTRU: a new high speed public key cryptosystem*, Preprint; presented at the rump session of Crypto '96, 1996.

[KTX08] KAWACHI, A.—TANAKA, K.—XAGAWA, K.: *Concurrently secure identification schemes based on the worst-case hardness of lattice problems*, in: Advances in Cryptology—ASIACRYPT '08, 14th Internat. Conf. on the Theory and Appl. of Cryptol. and Inform. Security (J. Pieprzyk, ed.), Melbourne, Australia, 2008, Lecture Notes in Comput. Sci., Vol. 5350, Springer, Berlin, 2008, pp. 372–389.

[LM06] LYUBASHEVSKY, V.—MICCIANCIO, D.: *Generalized compact knapsacks are collision resistant*, in: Automata, Languages and Programming, 33rd Internat. Colloq.—ICALP '06, Venice, Italy, 2006, Lecture Notes in Comput. Sci., Vol. 4052, Springer, Berlin, 2006, pp. 144–155.

[LM08] LYUBASHEVSKY, V.—MICCIANCIO, D.: *Asymptotically efficient lattice-based digital signatures*, in: Theory of Cryptography, 14th Theory of Cryptography Conf.—TCC '08, New York, USA, 2008, Lecture Notes in Comput. Sci., Vol. 4948, Springer, Berlin, 2008, pp. 37–54.

[LP11] LINDNER, R.—PEIKERT, C.: *Better key sizes (and attacks) for LWE-based encryption*, in: Topics in Cryptology—CT-RSA '11, The Cryptographers' Track at the RSA Conf. (A. Kiayias, ed.), San Francisco, CA, USA, 2011, Lecture Notes in Comput. Sci., Vol. 6558, Springer, Berlin, 2011, pp. 319–339.

[LPR10] LYUBASHEVSKY, V.—PEIKERT, C.—REGEV, O.: *On ideal lattices and learning with errors over rings*, in: Advances in Cryptology—EUROCRYPT '10, 29th Annual Internat. Conf. on the Theory and Appl. of Cryptogr. Tech. (H. Gilbert, ed.), French Riviera, 2010, Lecture Notes in Comput. Sci., Vol. 6110, Springer, Berlin, 2010, pp. 1–23.

[Lyu08a] LYUBASHEVSKY, V.: *Lattice-based identification schemes secure under active attacks*, in: Public key Cryptography—PKC '08, 11th Internat. Workshop on Practice and Theory in Public-Key Cryptography (R. Cramer, ed.), Barcelona, Spain, 2008, Lecture Notes in Comput. Sci., Vol. 4939, Springer, Berlin, 2008, pp. 162–179.

[Lyu08b] LYUBASHEVSKY, V.: *Towards practical lattice-based cryptography*, PhD Thesis, University of California, San Diego, 2008.

[Lyu09] LYUBASHEVSKY, V.: *Fiat-Shamir with Aborts: Applications to lattice and factoring-based signatures*, in: Advances in Cryptology—ASIACRYPT '09, 15th Internat. Conf. on the Theory and Appl. of Cryptology and Information Security

(M. Matsui, ed.), Tokyo, Japan, 2009, Lecture Notes in Comput. Sci., Vol. 5912, Springer, Berlin, 2009, pp. 598–616.

[Lyu12] LYUBASHEVSKY, V.: *Lattice signatures without trapdoors*, in: Advances in Cryptology—EUROCRYPT '12, 31st Annual Internat. Conf. on the Theory and Appl. of Cryptogr. Techniques (D. Pointcheval et al., eds.), Cambridge, UK, 2012, Lecture Notes in Comput. Sci., Vol. 7237, Springer, Berlin, 2012, pp. 738–755.

[Mic10] MICCIANCIO, D.: *Duality in lattice cryptography*, in: 13th Internat. Conf. on Practice and Theory in Public Key Cryptography—PKC '10, Paris, France, 2010 (invited talk).

[MP12] MICCIANCIO, D.—PEIKERT, CH.: *Trapdoors for lattices: simpler, tighter, faster, smaller*, in: Advances in Cryptology—EUROCRYPT '12, 31st Annual Internat. Conf. on the Theory and Appl. of Cryptogr. Techniques (D. Pointcheval et al., eds.), Cambridge, UK, 2012, Springer, Berlin, Vol. 7237, pp. 700–718.

[MR07] MICCIANCIO, D.—REGEV, O.: *Worst-case to average-case reductions based on Gaussian measures*, SIAM J. Comput. **37** (2007), 267–302.

[MR08] MICCIANCIO, D.—REGEV, O.: *Lattice-based cryptography*, in: Post-Quantum Cryptography—PQC '08, 2nd Internat. Workshop, Cincinnati, OH, USA, 2008, Lecture Notes in Comput. Sci., Vol. 5299, Springer, Berlin, 2008, pp. 147–191.

[MV03] MICCIANCIO, D.—VADHAN, S.: *Statistical zero-knowledge proofs with efficient provers: lattice problems and more*, in: Advances in Cryptology—CRYPTO '03, 23rd Annual Internat. Cryptology Conf. (D. Boneh, ed.), Santa Barbara, California, USA, 2003, Lecture Notes in Comput. Sci., Vol. 2729, Springer, Berlin, 2003, pp. 282–298.

[Pei09] CHRIS PEIKERT: *Public-key cryptosystems from the worst-case shortest vector problem: extended abstract*, in: Proc. of the 41st Annual ACM Symposium on Theory of Comput.—STOC '09 (M. Mitzenmacher, ed.), Bethesda, MD, USA, 2009, ACM, New York, 2009, pp. 333–342.

[Pei10] PEIKERT, CH.: *An efficient and parallel Gaussian sampler for lattices*, in: Advances in Cryptology—CRYPTO '10, 30th Annual Cryptology Conf. (T. Rabin, ed.), Santa Barbara, CA, USA, 2010, Lecture Notes in Comput. Sci., Vol. 6223, Springer, Berlin, 2010, pp. 80–97.

[PV08] PEIKERT, CH.—VAIKUNTANATHAN, V.: *Noninteractive statistical zero-knowledge proofs for lattice problems*, in: Advances in Cryptology—CRYPTO '08, 28th Annual Internat. Cryptology Conf., Santa Barbara, CA, USA, 2008, Lecture Notes in Comput. Sci., Vol. 5157, Springer, Berlin, 2008, pp. 536–553.

[Reg05] REGEV, O.: *On lattices, learning with errors, random linear codes, and cryptography*, in: Proc. of the 37th Annual ACM Symp. on Theory of Comput.—STOC '05, Baltimore, USA, 2005, ACM, New York, 2005, pp. 84–93.

[SCL11] SILVA, R.—CAYREL, P.-L.—LINDNER, R.: *Lattice-based zero-knowledge identification with low communication cost*, in: XI Simposio Brasileiro de Seguranca da Informacao e de Sistemas Computacionais—SBSEG '11, Brasil, 2011, pp. 95–107.

[SS11] STEHLÉ, D.—STEINFELD, R.: *Making NTRU as secure as worst-case problems over ideal lattices*, in: Advances in Cryptology—EUROCRYPT '11, 30th Annual Internat. Conf. on the Theory and Appl. of Cryptogr. Tech. (K. G. Paterson, ed.), Tallinn, Estonia, 2011, Lecture Notes in Comput. Sci., Vol. 6632, Springer, Berlin, 2011, pp. 27–47.

[SSTX09] STEHLÉ, D.—STEINFELD, R.—TANAKA, K.—XAGAWA, K.: *Efficient public key encryption based on ideal lattices*, in: Advances in Cryptology—ASIACRYPT '09, 15th Internat. Conf. on the Theory and Appl. of Cryptology and Inform. Security (M. Matsui, ed.), Tokyo, Japan, 2009, Lecture Notes in Comput. Sci., Vol. 5912, Springer, Berlin, 2009, pp. 617–635.

[Ste96] STERN, J.: *A new paradigm for public key identification*, IEEE Trans. Inform. Theory **42** (1996), 1757–1768.

[XT09] XAGAWA, K.—TANAKA, K.: *Zero-knowledge protocols for NTRU: application to identification and proof of plaintext knowledge*, in: The Provable Security––ProvSec '09 (J. Pieprzyk et al., eds.), Guangzhou, China, 2009, Lecture Notes in Comput. Sci., Vol. 5848, Springer, Berlin, 2009, pp. 198–213.

Received August 28, 2012

Cryptography and Computeralgebra Group
TU Darmstadt
Department of Computer Science
Hochschulstraße 10
D–64289 Darmstadt
GERMANY

E-mail: elbansarkhani@cdc.informatik.tu-darmstadt.de
cabarcas@cdc.informatik.tu-darmstadt.de
kbj@crypto.tw
pschmidt@cdc.informatik.tu-darmstadt.de
mischnei@cdc.informatik.tu-darmstadt.de