# Post-Quantum Cryptography: Lattice Identification Schemes

Rosemberg Silva[1], Pierre-Louis Cayrel[2], and Johannes Buchmann[2]

[1] State University of Campinas (UNICAMP)
Institute of Computing
P.O. Box 6176
13084-971 Campinas
Brazil
rasilva@ic.unicamp.br
[2] CASED – Center for Advanced Security Research Darmstadt,
Mornewegstrasse, 32
64293 Darmstadt
Germany
{pierre-louis.cayrel,johannes.buchmann}@cased.de

**Abstract.** This survey presents an overview and a comparative analysis of the state of art in post-quantum identification schemes based on lattices. Furthermore, we propose an adaptation of the HB family of identification in a lattice context. The aspects taken into account in such comparison are performance, security, communication costs, underlying hard-problem, completeness, soundness, and key sizes.

**Keywords**: survey, post-quantum cryptography, identification scheme, lattice, codes, Fiat-Shamir heuristic.

## 1 Introduction

One of the most common security goals consists in assuring identity authentication. This kind of operation can be used, for example, to provide access control. One can accomplish this security goal by means of the application of interactive zero-knowledge identification schemes. There are several constructions that rely on number theoretic problems as security basis, like discrete logarithm problems [17] or factoring of integers [5]. In the advent of quantum computers, these identification schemes will be broken with Shor's algorithm [22] for integer factorization, which was published in 1994. In particular, all constructions whose security relies on number theory (such as variants of the discrete logarithm problem or integer factorization) are vulnerable to this algorithm. If quantum computers will at one point exist, such schemes can be broken in polynomial time, whereas no quantum attacks are known for lattice-based, code-based, and multivariate cryptographic systems. On the other hand, even should such number-theoretic assumptions remain hard, it is not wise to rely on a single type of hard problems. Furthermore, as the capacity of current adversaries increases, so does the key size for classical constructions; it is possible that alternative post-quantum constructions may provide a better alternative in that sense.

### 1.1 Our Contribution

We present the state of the art in post-quantum identification schemes based on lattices, focusing in performance, security, communication costs, underlying hard problem, completeness, soundness, and key sizes. In addition, we propose to adapt HB and Véron's to obtain lattice-based schemes. We further discuss the suitability of signature schemes derived from such schemes through the application of Fiat-Shamir heuristics.

### 1.2 Organization of the document

This paper is divided as follows. In Section 2, we give general definitions regarding lattices, and identification schemes. Then, we describe and compare the lattice-based schemes in Section 3, both from security and performance perspectives. We give a summary of such schemes in Section 4. After that, an overall appreciation of the schemes is given in Section 5.

## 2 Preliminaries

In this section, we present some definitions that are employed to describe the identification schemes analyzed in this article.

### 2.1 Zero-Knowledge Proof of Knowledge

In cryptography, a zero-knowledge protocol is an interactive proof is a method by means of which one party (Prover) convinces another (Verifier) that a given statement is true, without revealing anything other than the veracity of the statement. The parties are considered honest if they follow the protocol. Otherwise, they are considered cheaters.

This kind of proof satisfies three properties:

- Completeness: an honest Prover is always able to convince an honest Verifier about the veracity of a true statement.
- Soundness: no cheating Prover is able to convince an honest Verifier that a false statement is true, except with some "small" probability.
- Zero-Knowledge: nothing but the truthfulness of the statement being proved is learned from the protocol execution.

There is a standard construction, proposed by Fiat and Shamir in [5], that converts identification schemes into signature schemes. In such construction, the entity Verifier is replaced by a source of pseudo-random bits extracted from both the message being signed and the commitment values computed by the signing entity, which corresponds to the Prover. Verifying the correctness of the signature consists in checking if it corresponds to a valid transcript of an execution of the identification protocol.

### 2.2 Lattices

**Definition 1.** *A lattice is a discrete subgroup of $\mathbb{R}^m$ with dimension $n \leq m$. In general, for cryptographic applications, it is restricted to $\mathbb{Z}^m$. It can be represented by a basis comprising $n$ linear independent vectors of $\mathbb{R}^m$.*

**Definition 2 (Ideal lattices).** *Let $f$ be some monic polynomial of degree $n$. Then, $L$ is an* ideal lattice *if it corresponds to an ideal $I$ in the ring $\mathbb{Z}[x]/\langle f\rangle$.*

The polynomial $f(x) = x^n - 1$ defines a particular class of ideal lattices known as cyclic lattices, whereas $f(x) = x^n + 1$ defines the anticyclic lattices. We also have the class of cyclotomic lattices resulting from all cyclotomic polynomials $f$. Currently, such class is the only one relevant for practical applications.

**Definition 3.** ***(Short Integer Solution - SIS)*** *Given $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$, a prime number $q$ and $L \in \mathbb{R}$ find a vector $\mathbf{x} \in \mathbb{Z}^m$ that satisfies the equation $\mathbf{Ax} = 0 \bmod q$ and has length restricted by $\|\mathbf{x}\| \leq L$.*

### 2.3 Learning Problems

The computational problems described in this subsection are related to error correcting codes and lattices [11]. Their hardness results from the noise added to the outcome of the dot product of a secret vector and a collection of randomly chosen vectors. Such noise is known to follow a specified distribution.

**Definition 4 (Learning Parity with Noise (LPN)).** *For an integer $n \geq 1$ and a real number $\epsilon \geq 0$, consider "learning parity with noise" the problem defined as follows: find an unknown $\mathbf{s} \in \mathbb{Z}_2^n$ given a list of "equations with errors" $\langle \mathbf{s}, \mathbf{a}_i \rangle \approx_\epsilon b_i \bmod 2$ where the $\mathbf{a}_i$'s are independently chosen from the uniform distribution on $\mathbb{Z}_2^n$, $\langle \mathbf{s}, \mathbf{a}_i \rangle = \sum_j s_j (\mathbf{a}_i)_j$ is the inner product modulo 2 of $\mathbf{s}$ and $\mathbf{a}_i$, and each equation is correct with probability $1 - \epsilon$. The goal is to find $\mathbf{s}$.*

**Definition 5 (Learning With Errors (LWE)).** *Let $p$ be a prime number and $\chi$ a probability distribution on $\mathbb{Z}_p$. Given a secret $\mathbf{s} \in \mathbb{Z}_p^n$, we denote by $\mathbf{A}_{\mathbf{s},\chi}$ the probability distribution on $\mathbb{Z}_p^n \times \mathbb{Z}_p$ obtained by choosing a vector $\mathbf{a} \in \mathbb{Z}_p^n$ uniformly at random, choosing $\mathbf{e} \in \mathbb{Z}_p$ according to $\chi$, and outputting $(\mathbf{a}, \langle \mathbf{a}, \mathbf{s} \rangle + \mathbf{e} \mod p)$. The problem of recovering $\mathbf{s}$ from this system of equations is named "learning with errors" and denoted by $LWE_{p,\chi}$. We say that an algorithm solves $LWE_{p,\chi}$ if, for any $\mathbf{s} \in \mathbb{Z}_p^n$, given samples from $A_{s,\chi}$ it outputs $\mathbf{s}$ with probability exponentially close to 1. We say that the algorithm is efficient if it runs in time polynomial in $n$.*

## 3 Identification Schemes

By making use of the concepts defined in the previous section, we now list a number of lattice-based identification schemes. They involve two entities, namely a prover and a verifier. The first has the goal of convincing the second about the authenticity of his identity by showing that he knows a secret value.

The algorithms that are used in the realization of the several security schemes described in this work rely upon a "Commitment function". This function can receive a variable number of input arguments, which are then concatenated and treated as a single string of bytes. The image obtained by such function follows a distribution close to uniform in $\mathbb{Z}_q$, where $q$ is an integer given as parameter in the realization of the "Commitment function".

### 3.1 Lyubashevsky's Identification Scheme

This is a 3-pass identification scheme whose security is based on the worst-case hardness of the shortest vector problem in all lattices, as initially presented at [15]. Its author also indicates that a more efficient version based on the hardness of the same problem in ideal lattices can be obtained. Such construction was later detailed in [16].

On the low end, this identification scheme has communication complexity of around 65000 bits and the length of the signatures produced by the corresponding signature scheme, obtained via Fiat-Shamir heuristic, is about 50000 bits, according to Table 1, reproduced from [16].

**Algorithm**

The algorithm depicted below corresponds to the identification scheme version described in [16], using ideal lattices. Differently from the scheme introduced in [15], the challenges sent by the verifier are not a single bit, but rather an element of a ring. This helps to obtain shorter signatures via Fiat-Shamir construction. Each step indicates the actor (either the prover or the verifier). For a list of concrete parameters and domain definition, one should refer to Tables 1 and 2, respectively.

– Private key: $\hat{\mathbf{s}} \overset{\$}{\leftarrow} D_s^m$
– Public key: $h \overset{\$}{\leftarrow} \mathcal{H}(R, D, m)$, $\mathbf{S} \leftarrow h(\hat{\mathbf{s}})$
1 Prover: compute $\hat{\mathbf{y}} \overset{\$}{\leftarrow} D_y^m$, $\mathbf{Y} \leftarrow h(\hat{\mathbf{y}})$ and send $\mathbf{Y}$ to the verifier.
2 Verifier: compute $c \overset{\$}{\leftarrow} D_c$ and send $c$ to the prover.
3 Prover: compute $\hat{\mathbf{z}} \leftarrow \hat{\mathbf{s}}c + \hat{\mathbf{y}}$. If $\hat{\mathbf{z}} \notin G^m$, then $\hat{\mathbf{z}} \leftarrow \perp$. Send $\hat{\mathbf{z}}$ to the verifier
4 Verifier: accept the prover if $\hat{\mathbf{z}} \in G^m$ and $h(\hat{\mathbf{z}}) = \mathbf{S}c + \mathbf{Y}$

**Interactive Proof Properties**

In spite of not being zero-knowledge, this identification scheme satisfies the *witness indistinguishability* property. That is, given two distinct private keys $\hat{\mathbf{s}}_1$ and $\hat{\mathbf{s}}_2$ associated with the same public key, $h(\hat{\mathbf{s}}_1) = h(\hat{\mathbf{s}}_2)$, it is not information-theoretically possible for the verifier to tell which private key was actually used in the algorithm execution. By a suitable choice of parameters, we can have all public keys associated with more than one private key, with overwhelming probability, as stated by the following lemma.

**Lemma 1.** *If $\hat{\mathbf{s}}$ is chosen uniformly at random from $D_s^m$, then, with probability $1 - 2^{-\Omega(n \log n)}$, there will be another $\hat{\mathbf{s}}'$ satisfying $h(\hat{\mathbf{s}}') = h(\hat{\mathbf{s}})$.*

For the parameters listed in Table 1, this probability is above $1 - 2^{-128}$. Besides, the witness-indistinguishability property is kept under parallel composition, and this fact enables to parallelize this scheme. The same cannot be done, in general, with zero-knowledge schemes.

As far as the *completeness* property is concerned, this scheme has a non-negligible error of $1 - 1/e$, as consequence of the lemma below.

**Lemma 2.** *Given that $\hat{\mathbf{y}} \in D_y^m$, for any $\hat{\mathbf{s}}$ such that $\|\hat{\mathbf{s}}\|_\infty \leq n$,*

$$Pr_{\hat{\mathbf{y}} \overset{\$}{\leftarrow} D_y^m} [\hat{\mathbf{s}} + \hat{\mathbf{y}} \in G^m] = \frac{1}{e} - o(1).$$

Therefore, in a given round of execution, an honest prover may be rejected for refusing to reveal $\hat{\mathbf{s}}c + \hat{\mathbf{y}}$ when such value does not fall in a safe region. Otherwise, he would reveal information about the private key.

Concerning the *soundness* property, this identification scheme has very small error, when compared to zero-knowledge constructions like Kawachi's. For the parameters listed in Table 1, the error is less than $2^{-80}$. It is determined by the size of the set $D_c$ from which the challenges are chosen.

**Cost**
The communication costs imposed by this scheme are determined by the three messages: commitment, challenge and answer. The first corresponds to an element in the ring $R$ and takes $n \log p$ bits. The challenge also occupies $n \log p$ bits, because it is given by an element belonging to $D_c$, which is a subset of $R$. The answer is represented by a vector in the $G^m$ space, and can have up to $mn \log p$ bits. The cost per round of execution, thus, equal to $(m+2)n \log p$.

In terms of computational complexity, the most demanding operations in each round are the multiplications involving elements of the ring $R$, which take $\tilde{O}(n)$. Besides, due to the completeness error value, $w(\log n)$ rounds are necessary in order to assure that the prover is indeed accepted.

**Security**
This identification scheme is secure against active attacks, i.e., an adversary is allowed to interact with the prover prior to impersonation. Breaking this identification scheme also implies in solving the approximate $\mathrm{SVP}_\gamma$ problem, with approximation factor given by $\tilde{O}(n^3)$ for every lattice corresponding to an ideal ring $\mathbb{Z}[x] / \langle f(x) \rangle$. Unfortunately, such choice makes the system parameters too large for practical use, as stated by its own author.

The theorem below applies the witness-indistinguishability property to establish the scheme security in the active attack model

**Theorem 1.** *If $h$ is any function in $\mathcal{H}(R, D, m)$ for the parameters defined in Table 1 and there exists a polynomial-time adversary who can break the identification scheme with probability $q'$ in the active attack model, then there exists a polynomial-time algorithm that finds collisions in the hash function $h$ in the domain $D$ with probability at least $\frac{q'}{4}(q' - \frac{1}{|D_c|}) - 2^{-\Omega(n \log n)}$.*

**Memory Requirements**
The algorithm that describes the scheme makes references to public and private keys, as well as hash function computations. These values are listed in Table 2 for a set of different instances, and represent the memory needed from prover and verifier in order to execute the scheme.

**Parameters**
In this section we list a set of safe parameters, showing the vector lengths that can be obtained with the state of art the algorithms and the length necessary to break this identification scheme. Tables 1 and 2 lists the values suggested by the author in [16].

### 3.2 Kawachi, Tanaka and Xagawa's Identification Scheme

This scheme has a 3-pass structure, is provably safe against concurrent attacks, and takes the hardness of the SIS lattice problem as security assumption. It follows a similar construction to Stern's code-based scheme [24], which is based on the hardness of the syndrome decoding problem on its turn. Its milder security assumption, when compared to Lyubashevksy's scheme, enables the use of smaller parameters. As consequence, it can reach a better performance in terms of communication costs, as shown in [12].

- Private key: $\mathbf{x} \xleftarrow{\$} \mathbb{F}_2^m$, with Hamming weight $m/2$
- Public key: $\mathbf{y} = \mathbf{A}\mathbf{x}$, where $\mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n \times n}$ is public.

| Parameter | Definition | Instances | | | |
|---|---|---|---|---|---|
| $n$ | integer that is a power of 2 | 512 | 512 | 512 | 1024 |
| $m$ | any integer | 4 | 5 | 8 | 8 |
| $\sigma$ | any integer | 127 | 2047 | 2047 | 2047 |
| $\kappa$ | integer such that $2^{\kappa}\binom{n}{k} \geq 2^{160}$ | 24 | 24 | 24 | 21 |
| $p$ | integer $\approx (2\sigma+1)^m 2^{-\frac{128}{n}}$ | $2^{31.7}$ | $2^{59.8}$ | $2^{95.8}$ | $2^{95.8}$ |
| Signature Size | $\approx mn\log(2mn\sigma\kappa)$ bits | 49000 | 72000 | 119000 | 246000 |
| Public Key Size | $\approx n\log p$ bits | 16000 | 31000 | 49000 | 98000 |
| Secret Key Size | $\approx mn\log(2\sigma+1)$ bits | 16000 | 31000 | 49000 | 98000 |
| Hash Function Size | $\approx mn\log p$ bits | 65000 | 153000 | 392000 | 786000 |
| Length of vector needed to break signature | | $2^{23.5}$ | $2^{27.9}$ | $2^{28.6}$ | $2^{29.5}$ |
| Length of the shortest vector that can be found | | $2^{25.5}$ | $2^{36.7}$ | $2^{47.6}$ | $2^{69.4}$ |

**Table 1.** Lyubashevsky's system parameters

| Domain | Definition |
|---|---|
| $R$ | ring $\mathbb{Z}_p[x] / \langle x^n+1 \rangle$ |
| $D$ | $\left\{g \in R : \|g\|_\infty \leq mn\sigma\kappa\right\}$ |
| $D_s$ | $\left\{g \in R : \|g\|_\infty \leq \sigma\right\}$ |
| $D_c$ | $\left\{g \in R : \|g\|_1 \leq \kappa\right\}$ |
| $D_y$ | $\left\{g \in R : \|g\|_\infty \leq mn\sigma\kappa\right\}$ |
| $G$ | $\left\{g \in R : \|g\|_\infty \leq mn\sigma\kappa - \sigma\kappa\right\}$ |

**Table 2.** Lyubashevsky's domain definitions

### Algorithm

```
1 Prover: choose a random permutation $\pi$ over $\{1,\dots,m\}$ and $\mathbf{r} \overset{\$}{\leftarrow} \mathbb{Z}_q^m$.
2 Prover: compute commitments $c_1, c_2$ and $c_3$ as
  2.1 $c_1 \leftarrow$ Commitment function$(\pi, \mathbf{Ar})$
  2.2 $c_2 \leftarrow$ Commitment function$(\pi(\mathbf{r}))$
  2.3 $c_3 \leftarrow$ Commitment function$(\pi(\mathbf{x}+\mathbf{r}))$
3 Verifier: compute $c \overset{\$}{\leftarrow} \{1,2,3\}$ and send $c$ to the prover.
4 Prover: reveal information to allow the verifier to check the commitment correctness
  4.1 If $c=1$, send $\pi(\mathbf{x})$ and $\pi(\mathbf{r})$ to the verifier.
  4.2 If $c=2$, send $\pi$ and $\mathbf{r}+\mathbf{x}$ to the verifier.
  4.3 If $c=3$, send $\pi$ and $\mathbf{r}$ to the verifier.
5 Verifier: check commitment correctness from information revealed by the prover, and accept
  prover in case of success.
  5.1 If $c=1$, verify that $c_2$ and $c_3$ can be computed, and that $\pi(\mathbf{x})$ is binary with Hamming
      weight $m/2$.
  5.2 If $c=2$, verify that $c_1$ and $c_3$ can be computed.
  5.3 If $c=3$, verify that $c_1$ and $c_2$ can be computed.
```

### Interactive Proof Properties

Kawachi's identification scheme constitutes an interactive proof of knowledge that a prover possesses a private key that represents a solution to an inhomogeneous SIS problem. Its statistically zero-knowledge property follows from the fact that the commitment function used is statistically-hiding and computationally binding string commitment scheme. Such scheme is built upon a hash function

$h$. The binding characteristic follows from the collision-resistance property of $h$, whereas the hiding characteristic is a consequence of the $\epsilon$-regularity of $h$.

The statistically hiding property is essential to assure that any computationally unbounded adversarial receiver cannot distinguish two commitment strings generated from two distinct strings. On its turn, the computationally binding implies that no polynomial-time adversarial sender can successfully change the committed string after sending the commitment and keep it consistent.

It has perfect *completeness*, which means that an honest prover will always be accepted by an honest verifier, regardless of the level of security desired.

Concerning its *soundness* property, on the other hand, there is an error of 2/3. Hence, in any given round, a cheating prover can make a successful impersonation with probability of up to 2/3. To circumvent this error, the scheme must be run a minimum number of times $r$, so that the overall probability of cheating success $(2/3)^r$ be lower than the level required by the specified application.

**Cost**

In spite of requiring a higher number of repetitions when compared to Lyubashevsky's scheme, as consequence of the soundness error, the scheme of Kawachi et al. still possesses lower communication costs. Its parameters are smaller due to the milder security assumptions, and so are the permutations and vectors exchanged between prover and verifier during the algorithm execution.

Kawachi did not provide concrete parameters. Instead, he only gave an asymptotic behavior. The numbers listed in Table 3 were extracted from Cayrel et al. [2] in order to establish a comparison with the CLRS scheme.

| Parameter | Value |
|---|---|
| $n$ | 512 |
| $m$ | 2048 |
| $q$ | 257 |
| Commitment length | 256 bits |
| Secret key length | 0.25 kBytes |
| Public key length | 0.06 kBytes |
| Rounds | 150 |
| Communication costs | 314.3 kBytes |

**Table 3.** Kawachi et al. system parameters

**Security**

This identification scheme is secure against concurrent attacks, i.e., an adversary is allowed to interact with a number of prover instances prior to impersonation. Each of those instances has the same secret key, but their random coins are independent and their own state is individually kept. As pointed out in subsection 3.1, the security assumptions used in Lyubashevky's identification scheme lead to parameters too big to be considered practical. Kawachi's construction addresses this issue by applying weaker security assumptions, i.e., the approximation factors used on the lattice problems upon which the system security relies are smaller. The scheme $S^{+}_{GL}$ is based on Stern's scheme and the GapSVP on general lattices with approximation factor $\tilde{O}(n)$ and euclidean norm. The $S^{+}_{C/IL}$, in its turn, is based on Stern's scheme and the SVP for ideal lattices with approximation factor $\tilde{O}(n)$ and infinite-norm.

**Memory Requirements**
The keys associated with this scheme are shorter than those required by Lyubashevsky's due to the milder security assumptions discussed above.

### 3.3 CLRS Identification Scheme

Cayrel et al. [2] proposed this lattice zero-knowledge identification scheme as an adaptation of a code-based construction due to Cayrel, Véron and El Yousfi [4]. It is concurrently secure under the hardness assumption of the SIS problem and the collision resistance of string commitment schemes.

Table 4 lists a set of parameters for a security level of 80 bits.

| Parameter | Value |
|---|---|
| $n$ | 512 |
| $m$ | 2048 |
| $q$ | 257 |
| Commitment length | 256 bits |
| Secret key length | 0.25 kBytes |
| Public key length | 0.06 kBytes |
| Rounds | 81 |
| Communication costs | 178.9 kBytes |

**Table 4.** CLRS system parameters

**Algorithm**
Figures 1 and 2 describe, respectively, the key pair generation process and the identification protocol for the CLRS scheme, as detailed in [2]. The private keys correspond to binary vectors whose Hamming weight is exactly half of their length. Deriving such values from the respective private key corresponds to solving worst-case instances of the SIS problem. The identification protocol corresponds to a zero-knowledge proof of knowledge that the prover possesses a private key associated with a given public key. The soundness error of approximately 1/2 allows performance gains when compared to other schemes directly derived from Stern's, like that described in Section 3.2. This gain is kept when the Fiat-Shamir transform is used in order to derive signature schemes, such as the lattice-based TRSS [3].

```
KeyGen:
  $\mathbf{x} \xleftarrow{\$} \{0,1\}^m$, s.t. $\mathrm{wt}(\mathbf{x}) = m/2$
  $\mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n \times m}$
  $\mathbf{y} \longleftarrow \mathbf{Ax} \bmod q$
  $\textsc{Com} \xleftarrow{\$} \mathcal{F}$, suitable family of commitment functions
  Output $(\mathrm{sk}, \mathrm{pk}) = (\mathbf{x}, (\mathbf{y}, \mathbf{A}, \textsc{Com}))$
```

**Fig. 1.** Key generation algorithm, parameters $n, m, q$ are public.

| **Prover** $\mathcal{P}(\mathrm{sk}, \mathrm{pk})$ |  | **Verifier** $\mathcal{V}(\mathrm{pk})$ |
|---|---|---|
|  | $(\mathrm{sk}, \mathrm{pk}) = (\mathbf{x}, (\mathbf{y}, \mathbf{A}, \textsc{Com})) \longleftarrow \textsc{KeyGen}$ |  |
| $\mathbf{u} \xleftarrow{\$} \mathbb{Z}_q^m, \sigma \xleftarrow{\$} S_m, \mathbf{z} \longleftarrow \mathbf{P}_\sigma \mathbf{x}$ |  |  |
| $\mathbf{r}_0 \xleftarrow{\$} \{0,1\}^n, \mathbf{r}_1 \xleftarrow{\$} \{0,1\}^n$ |  |  |
| $c_0 \longleftarrow \textsc{Com}(\sigma \,\|\, \mathbf{A}\mathbf{u}; \mathbf{r}_0)$ |  |  |
| $c_1 \longleftarrow \textsc{Com}(\mathbf{z} \,\|\, \mathbf{P}_\sigma \mathbf{u}; \mathbf{r}_1)$ | $\xrightarrow{c_0, c_1}$ |  |
|  | $\xleftarrow{\alpha}$ | $\alpha \xleftarrow{\$} \mathbb{Z}_q$ |
| $\beta \longleftarrow \mathbf{P}_\sigma(\mathbf{u} + \alpha \mathbf{x})$ | $\xrightarrow{\beta}$ |  |
|  | $\xleftarrow{\text{Challenge } b}$ | $b \xleftarrow{\$} \{0,1\}$ |
| **If** $b = 0$: | $\xrightarrow{\sigma, \mathbf{r}_0}$ | **Check** $c_0 \overset{?}{=} \textsc{Com}(\sigma \,\|\, \mathbf{A}\mathbf{P}_\sigma^{-1}\beta - \alpha \mathbf{y}; \mathbf{r}_0)$<br>$\sigma \overset{?}{\in} S_m$ |
| **Else**: | $\xrightarrow{\mathbf{z}, \mathbf{r}_1}$ | **Check** $c_1 \overset{?}{=} \textsc{Com}(\mathbf{z} \,\|\, \beta - \alpha \mathbf{z}; \mathbf{r}_1)$<br>$\mathbf{z} \overset{?}{\in} \{0,1\}^m, \mathrm{wt}(\mathbf{z}) \overset{?}{=} m/2$ |

**Fig. 2.** Identification protocol

### Interactive Proof Properties

The algorithm shown above constitutes a zero-knowledge interactive proof that the party called prover knows the secret key $\mathbf{x}$ that satisfies the relation $\mathbf{A}\mathbf{x} = \mathbf{y} \bmod q$ with the public key $\mathbf{y}$. The proofs for the zero-knowledge, completeness and soundness properties are given in [2]. The scheme is shown to have perfect completeness, but soundness error of $1/2$, approximately. This implies that in order to reach an overall soundness error $L$, $r$ rounds are necessary, so that the relation $(1/2)^r \leq L$ is satisfied. Thus, such schemes need a fewer number of rounds to reach the soundness goal, when compared to those of Kawachi et al. [12] and Xagawa et al. [26].

### Cost

The smaller soundness error per round of approximately $1/2$ also implies in a reduction in the communication costs, when compared to the scheme of Kawachi et al. [12], which has soundness error of $2/3$. For a security level of 80 bits, the costs are also smaller than those of Lyubashvsky's scheme, which has even smaller soundness error, but bigger parameters.

### Security

This identification scheme has its security based on the existence of a string commitment scheme and the intractability of the SIS problem. Therefore, the existence of an adversary that breaks the identification scheme implies that breaks at least one of the security assumptions: finding collisions in the commitment scheme, or solving the SIS problem. In particular, the problem of obtaining a private key associated with a given public key corresponds to some of the worst cases of SIS. The private keys can be seen as lattice vectors with very small norm.

### Memory Requirements

For a security level of 80 bits and the parameter set listed above, the space required to store the public and private keys are 0.25 kBytes and 0.06 Kbytes, respectively. For generic lattices, the space needed to store the basis is given by $n \times m \times \lceil \log q \rceil = 9$MBytes.

### 3.4 Xagawa and Tanaka's Identification Scheme

Xagawa and Tanaka [26] proposed zero-knowledge and proof-of-knowledge protocols for NTRU (short for N-th degree truncated polynomial ring) cryptosystem, using a statistically hiding and computationally binding commitment scheme. This protocol constitutes the first identification scheme based on NTRU and Stern's scheme [24]. It is passively secure under the assumption that it is hard to find two elements $\mathbf{x}_h$ and $\mathbf{x}_t$ of enumeration sets that satisfy the relation $\mathbf{a}_h \otimes \mathbf{x}_h + \mathbf{a}_t \otimes \mathbf{x}_t \equiv \mathbf{y}$, given that the triple $(\mathbf{a}_h, \mathbf{a}_t, \mathbf{y}) \in R_q^3$ is public, with $R_q = \mathbb{Z}_q\left[\alpha\right] / \left\langle \alpha^n - 1 \right\rangle$.

The set of parameters originally proposed for a security level of 80 bits is summarized in Tables 5 and 6.

#### Algorithm

```
– Private key: $\mathbf{x}_h$ and $\mathbf{x}_t$ such that $\mathbf{a}_h \otimes \mathbf{x}_h + \mathbf{a}_t \otimes \mathbf{x}_t \equiv \mathbf{y}$
– Public key: $(\mathbf{a}_h, \mathbf{a}_t, \mathbf{y}) \in R_q^3$.
1 Prover: choose random permutations $\pi_h$ and $\pi_t$ over $\{1, \ldots, n\}$, and random vectors $\mathbf{r}_h$ and $\mathbf{r}_h \in R_q$.
2 Prover: compute commitments $c_1, c_2$ and $c_3$ as
  2.1 $c_1 \leftarrow$ Commitment function$(\pi_h, \pi_t, \mathbf{a_h} \otimes \mathbf{r_h} + \mathbf{a_t} \otimes \mathbf{r_t})$
  2.2 $c_2 \leftarrow$ Commitment function$(\pi_h(\mathbf{r_h}), \pi_t(\mathbf{r_t}))$
  2.3 $c_3 \leftarrow$ Commitment function$(\pi_h(\mathbf{r_h} + \mathbf{x_h}), \pi_t(\mathbf{r_t} + \mathbf{x_t}))$
3 Verifier: compute $c \xleftarrow{\$} \{1, 2, 3\}$ and send $c$ to the prover.
4 Prover: reveal information to allow the verifier to check the commitment correctness
  4.1 If $c = 1$, send $\pi_h(\mathbf{x}_h)$, $\pi_t(\mathbf{x}_t)$, $\pi_h(\mathbf{r}_h)$ and $\pi_t(\mathbf{r}_t)$ to the verifier.
  4.2 If $c = 2$, send $\pi_h$, $\pi_t$, $\mathbf{r_h} + \mathbf{x_h}$ and $\mathbf{r_t} + \mathbf{x_t}$ to the verifier.
  4.3 If $c = 3$, send $\pi_h$, $\pi_t$, $\mathbf{r_h}$ and $\mathbf{r_t}$ to the verifier.
5 Verifier: check commitment correctness from information revealed by the prover, and accept prover in case of success.
  5.1 If $c = 1$, verify that $c_2$ and $c_3$ can be computed, and that $\pi(\mathbf{x_h})$ and $\pi(\mathbf{x_h})$ belong to enumeration sets.
  5.2 If $c = 2$, verify that $c_1$ and $c_3$ can be computed.
  5.3 If $c = 3$, verify that $c_1$ and $c_2$ can be computed.
```

#### Interactive Proof Properties

The algorithm shown above constitutes zero-knowledge interactive proof that the party called prover knows the secret values $\mathbf{x}_h$ and $\mathbf{x}_t$ that satisfy the relation $\mathbf{a}_h \otimes \mathbf{x}_h + \mathbf{a}_t \otimes \mathbf{x}_t \equiv \mathbf{y}$. A sketch of proof for the zero-knowledge, completeness and soundness properties is given in [26]. The scheme is shown to have perfect completeness, but soundness error of 2/3. This implies that in order to reach a security level $L$, $r$ rounds are necessary, so that the relation $(2/3)^r \leq L$ is satisfied.

| Parameter | Value |
|---|---|
| Security | 80 bits |
| Rounds | 150 |
| Queries allowed | $2^{60}$ |
| Commitment scheme | Halevi-Micali [7] |
| Communication costs | 716.5 kBytes |
| NTRU Set | ees677ep1 |

**Table 5.** Xagawa et al. system parameters

| Parameter | Value(bits) |
|---|---|
| Security level | 192 |
| Public key length | 7447 |
| Secret key length | 1354 |

**Table 6.** NTRU ees677ep1 set

**Cost**

Assuming the parameter set used by this scheme is NTRU-2008 ees677ep1 [8] and that the commitment scheme Halevi-Micali scheme [7] is applied in the construction, for a security level of 80 bits, the communication cost of this identification scheme is approximately 716.5 kB.

**Security**

This identification scheme has its security based on the existence of a string commitment scheme and the intractability of the NTRU decomposition problem. Therefore, the existence of an adversary that breaks the identification scheme implies that he is also able to break at least one of the security assumptions. One must recall that the security proofs and arguments for NTRU have been under dispute, including the hardness of the decomposition problem.

**Memory Requirements**

For a security level of 80 bits and the parameter set listed above, the space required to store the public and private keys are 7447 bits and 1354 bits, respectively.

### 3.5 LWE-Based Identification Scheme

Silva, Campello and Dahab [23] proposed an LWE-based zero-knowledge identification scheme. It constitutes the first identification scheme based on LWE and Véron's construction [25]. It is passively secure under the assumptions that (1) it is hard to find collisions in the underlying commitment function and (2) that it is computationally difficult to solve the LWE problem.

This scheme is comprised by two algorithms. The first one establishes a pair of keys, one private and one public, such that the private key corresponds to a solution to an instance of LWE problem which uses the public key as input parameter.

The second algorithm describes a sequence of message exchanges between the Prover and the Verifier. It serves as a proof of the fact that the Prover knows the solution to the LWE problem relating the public and private keys to which his identity is linked.

**Algorithms**

In order to obtain the pair of keys to be used in the proof of knowledge, the sequence of steps is followed in the first algorithm:

```
1 Choose the parameters $\mathbf{A} \xleftarrow{\$} \mathbb{F}_q^{n\times m},\ \mathbf{s} \xleftarrow{\$} \mathbb{F}_q^{m},\ e \xleftarrow{\chi} \mathbb{F}_q^{n}$.
2 $\mathbf{b} \leftarrow \mathbf{As} + \mathbf{e}$
3 Determine the Hamming weight $p \leftarrow \mathrm{wt}(\mathbf{e})$
4 Set $(\mathbf{A}, \mathbf{b}, p)$ is the public key
5 Set $(\mathbf{s}, \mathbf{e})$ is the private key
```

The mapping defined below is used in the interactive proof of knowledge as an isometry

**Definition 6 (Hamming isometry $\mathit{\Pi}_{\gamma,\mathit{\Sigma}}$).** *Let $\mathit{\Sigma}$ be a permutation of $\{1,\ldots,n\}$ and $\gamma = (\gamma_1,\ldots,\gamma_n) \in \mathbb{F}_q^n$ such that $\gamma_i \neq 0, \forall i$. We define the transformation $\mathit{\Pi}_{\gamma,\mathit{\Sigma}}$ as the mapping $\mathbb{F}_q^n \to \mathbb{F}_q^n$, taking $\mathbf{v}$ to $\gamma_{\mathit{\Sigma}(1)}v_{\mathit{\Sigma}(1)},\ldots,\gamma_{\mathit{\Sigma}(n)}v_{\mathit{\Sigma}(n)}$.*

The second algorithm, which corresponds to the actual proof of knowledge, is shown below.

```
1 Obtain the key pair $\{(\mathbf{A}, \mathbf{b}, p), (\mathbf{s}, \mathbf{e})\}$ with the previous algorithm.
2 Prover:
  2.1 $\mathbf{u} \xleftarrow{\$} \mathbb{F}_q^m, \gamma \xleftarrow{\$} \mathbb{F}_q^m$ with $\gamma_i \neq 0, \forall i \in \{1,\ldots,m\}$
  2.2 $\mathit{\Sigma} \xleftarrow{\$} S_n$
  2.3 Compute the commitments as
  2.4 $\mathbf{c_1} \leftarrow com(\mathit{\Pi}_{\gamma,\mathit{\Sigma}}; \mathbf{r_1})$
  2.5 $\mathbf{c_2} \leftarrow com(\mathit{\Pi}_{\gamma,\mathit{\Sigma}}(\mathbf{A}(\mathbf{u}+\mathbf{s})); \mathbf{r_2})$
  2.6 $\mathbf{c_3} \leftarrow com(\mathit{\Pi}_{\gamma,\mathit{\Sigma}}(\mathbf{Au}+\mathbf{b}); \mathbf{r_3})$
  2.7 Send the commitments to the Verifier
3 Verifier:
  3.1 $ch \xleftarrow{\$} \{1,2,3\}$.
  3.2 Send the challenge $ch$ to the Prover.
4 Prover:
  4.1 Open the commitments to the Verifier
  4.2 If $ch = 1$
  4.3 send $\mathbf{r_1}, \mathbf{r_2}, \mathbf{u}+\mathbf{s}$ and $\mathit{\Pi}_{\gamma,\mathit{\Sigma}}$
  4.4 else if $ch = 2$
  4.5 send $\mathbf{r_2}, \mathbf{r_3}, \mathit{\Pi}_{\gamma,\mathit{\Sigma}}(\mathbf{A}(\mathbf{u}+\mathbf{s}))$ and $\mathit{\Pi}_{\gamma,\mathit{\Sigma}}(\mathbf{e})$
  4.6 else if $ch = 3$
  4.7 send $\mathbf{r_1}, \mathbf{r_3}, \mathit{\Pi}_{\gamma,\mathit{\Sigma}}$ and $\mathbf{u}$
5 Verifier:
  5.1 Check the commitments
  5.2 If $ch = 1$
  5.3 check that $\mathbf{c_1}$ and $\mathbf{c_2}$ are correct.
  5.4 else if $ch = 2$
  5.5 check that $\mathbf{c_2}$ and $\mathbf{c_3}$ are correct;
  5.6 check that $\mathrm{wt}(\mathit{\Pi}_{\gamma,\mathit{\Sigma}}(\mathbf{e})) = p$.
  5.7 else if $ch = 3$
  5.8 check that $\mathbf{c_1}$ and $\mathbf{c_3}$ are correct.
```

**Interactive Proof Properties**

The algorithm shown above constitutes a zero-knowledge interactive proof that the party called Prover knows the secret key $(\mathbf{s}, \mathbf{e})$ that constitutes the solution to an LWE instance associated with the public key $(\mathbf{A}, \mathbf{b}, p)$. The instance is given by $\mathbf{b} \leftarrow \mathbf{A}\mathbf{s} + \mathbf{e}$, where $p$ is the Hamming weight of $\mathbf{e}$. The proofs for the zero-knowledge, completeness and soundness properties are given in [23]. The scheme is shown to have perfect completeness. Two different algorithms for the interactive proof are available in that work, with soundness errors of 2/3 and 1/2.

**Cost**

No parameters were given in the description of this identification scheme in [23]. However, taking into account recent surveys on LWE, such as [14] and [21], it is fair to assume that the communication costs will be considerably higher than those seen in the SIS-based schemes listed in the previous sections, when considering similar levels of security.

**Security**

This identification scheme has its security based on the existence of a string commitment scheme and the intractability of the LWE problem. Therefore, the existence of an adversary that breaks the identification scheme implies that he is also able to break at least one of the security assumptions.

### 3.6 HB$^+$ Identification Scheme

The HB shared-key identification protocol was proposed by Hopper and Blum [9], with proven security against passive attackers. Besides, this protocol has low computational costs that make it suitable for low-cost devices like RFID tags. An improvement made by Juels and Weis [10] named HB$^+$ achieved security against active attackers, provided that the protocol execution is sequential. They added a blinding factor in order to defend the prover from dishonest verifiers, which could adaptively choose challenges in order to extract the values of the shared keys. Katz, Shin and Smith [11] extended HB$^+$ security proofs to encompass concurrent execution as well. The LPN hardness is the security assumption on which both protocols are based. On its turn, LPN can be seen as a particular case of the hard lattice problem LWE [20].

**Algorithm**

*HB$^+$ Identification*

In this improved version of HB identification protocol, the prover and the verifier share two keys: $\mathbf{s}_1 \in \{0,1\}^k$ and $\mathbf{s}_2 \in \{0,1\}^\tau$, where $k$ determines the hardness of the underlying LPN problem and $\tau$ represents a statistical security parameter. Besides, the interactions are not required to be sequential. As proved in [11], security is preserved under concurrent executions of the basic step.

– Security parameters: $k$ for LPN hardness, $\tau$ for statistical security.
– Shared-keys: $\mathbf{s}_1 \in \{0,1\}^k$ and $\mathbf{s}_2 \in \{0,1\}^\tau$
– Ber$_\epsilon$: 1 with probability $\epsilon$; 0 with probability $1-\epsilon$
1 Prover: choose a blinding factor $\mathbf{b} \overset{\$}{\leftarrow} \{0,1\}^k$ and send it to the verifier.

```
2 Verifier: choose a random challenge $\mathbf{a} \xleftarrow{\$} \{0,1\}^\tau$ and send it to the prover.
3 Prover: compute the answer $z$
  3.1 $\nu \xleftarrow{\$} \{0,1\}$ according to the Bernoulli distribution $\mathrm{Ber}_\epsilon$
  3.2 $z \leftarrow \langle \mathbf{s}_1, \mathbf{b} \rangle \oplus \langle \mathbf{s}_2, \mathbf{a} \rangle \oplus \nu$
4 Prover: send the answer $z$ to the verifier
5 Verifier: accept, if $z$ matches $\langle \mathbf{s}_1, \mathbf{b} \rangle \oplus \langle \mathbf{s}_2, \mathbf{a} \rangle$.
```

### Proposal of an LWE-Extension for HB and HB$^+$

Li, Gong and Qin [13] explored connections between LPN and error-correcting codes in their proposal of the HB-CM identification protocol. Contrary to HB and HB$^+$, it is resilient to the man-in-the-middle attack devised by Ouafi, Overbeck and Vaudenay [18].

Conversely, in the algorithm below, we propose a straightforward extension of the HB/HB$^+$ protocols to use the hardness of the LWE problem as security assumption, instead of LPN. Given that reductions from worst-case lattice problems to solving LWE were demonstrated both in quantum form by Regev [20] and classical form by Peikert [19], and that an adversary that breaks the proposed algorithm can be used as a way of solving LWE, this gives us confidence that random instances of this protocol are hard to break.

For a fixed length of the shared keys, the communication costs of this algorithm increase by a factor of $\log p$ when compared to those of HB$^+$. On the other hand, we have a security gain as consequence of the discussion in the paragraph above, and also from the fact that the best algorithm know to solve LWE (BKW [1]) has higher computational cost than that from the best one which solves LPN (Fossorier et al. Algorithm [6]).

*LWE-based HB$^+$ Identification*

```
– Security parameters: $k$ for LWE hardness, $\tau$ for statistical security.
– Shared-keys: $\mathbf{s}_1 \in \mathbb{Z}_p^k$ and $\mathbf{s}_2 \in \mathbb{Z}_p^\tau$.
– Tolerance: indicates the threshold below which errors are accepted.
– $\chi$: error distribution.
1 Prover: choose a blinding factor $\mathbf{b} \xleftarrow{\$} \mathbb{Z}_p^k$ and send it to the verifier.
2 Verifier: choose a random challenge $\mathbf{a} \xleftarrow{\$} \mathbb{Z}_p^\tau$ and send it to the prover.
3 Prover: compute the answer $z$
  3.1 $\nu \xleftarrow{\$} \chi$
  3.2 $z \leftarrow \langle \mathbf{s}_1, \mathbf{b} \rangle + \langle \mathbf{s}_2, \mathbf{a} \rangle + \nu \bmod p$
4 Prover: send the answer $z$ to the verifier
5 Verifier: accept, if $|z - \langle \mathbf{s}_1, \mathbf{b} \rangle - \langle \mathbf{s}_2, \mathbf{a} \rangle \bmod p| \leq$ Tolerance.
```

## 4 Consolidation

In this section we summarize the strengths, weaknesses and characteristics of the lattice-based identification schemes described in Section 3.

| Scheme | Strength | Weakness | ZK/WI | Completeness Error | Soundness Error | Comm. Cost per round | Security |
|---|---|---|---|---|---|---|---|
| Lyubashevsky [16] | FS heuristic | Parameters | WI | $1 - 1/e$ | $< 2^{-80}$ | $\tilde{O}(n)$ | Active |
| Kawachi et al. [12] | Parameters | FS heuristic | ZK | None | $2/3$ | $\tilde{O}(n)$ | Concurrent |
| CLRS [2] | Soundness | FS heuristic | ZK | None | $\approx 1/2$ | $\tilde{O}(n)$ | Concurrent |
| Xagawa et al. [26] | Parameters | Security | ZK | None | $2/3$ | $\tilde{O}(n)$ | Active |
| HB+ [11] | Comm. Cost | Security | - | Depends on $\epsilon$ | $1/2$ | $O(n)$ | Concurrent |

**Table 7.** ID Schemes Comparison

## 5 Conclusions

This article showed post-quantum identification schemes based on lattices. Some constructions were zero-knowledge interactive proofs, with perfect completeness, but non-negligible soundness error. Therefore, in order to reach some security level, a minimum number of rounds of execution is necessary. When applying Fiat-Shamir heuristic to derive signature schemes, this implies in huge signatures.

## References

1. Avrim Blum, Adam Kalai, and Hal Wasserman. Noise-tolerant learning, the parity problem, and the statistical query model. *J. ACM*, 50(4):506–519, 2003.
2. Pierre-Louis Cayrel, Richard Lindner, Markus Rückert, and Rosemberg Silva. Improved zero-knowledge identification with lattices. *ProvSec 2010*, pages 1–17, 2010.
3. Pierre-Louis Cayrel, Richard Lindner, Markus Rückert, and Rosemberg Silva. A lattice-based threshold ring signature scheme. *LatinCrypt 2010*, pages 255–272, 2010.
4. Pierre-Louis Cayrel, Pascal Véron, and Sidi Mohamed El Yousfi Alaoui. Improved code-based identification scheme. *SAC 2010*, 2010. `http://arxiv.org/abs/1001.3017v1`.
5. Amos Fiat and Adi Shamir. How to prove yourself: Practical solutions to identification and signature problems. In Andrew M. Odlyzko, editor, *CRYPTO*, volume 263 of *Lecture Notes in Computer Science*, pages 186–194. Springer, 1986.
6. Marc P. C. Fossorier, Miodrag J. Mihaljevic, Hideki Imai, Yang Cui, and Kanta Matsuura. An algorithm for solving the lpn problem and its application to security evaluation of the hb protocols for rfid authentication. In Rana Barua and Tanja Lange, editors, *INDOCRYPT*, volume 4329 of *Lecture Notes in Computer Science*, pages 48–62. Springer, 2006.
7. Shai Halevi and Silvio Micali. Practical and provably-secure commitment schemes from collision-free hashing. In Neal Koblitz, editor, *CRYPTO*, volume 1109 of *Lecture Notes in Computer Science*, pages 201–215. Springer, 1996.
8. P. Hirschhorn, J. Hoffstein, N. Howgrave-Graham, and W. Whyte. Choosing NTRUEncrypt parameters in light of combined lattice reduction and MITM approaches. In *Applied cryptography and network security*, pages 437–455. Springer, 2009.
9. Nicholas J. Hopper and Manuel Blum. Secure human identification protocols. In Colin Boyd, editor, *ASIACRYPT*, volume 2248 of *Lecture Notes in Computer Science*, pages 52–66. Springer, 2001.
10. Ari Juels and Stephen A. Weis. Authenticating pervasive devices with human protocols. In Victor Shoup, editor, *CRYPTO*, volume 3621 of *Lecture Notes in Computer Science*, pages 293–308. Springer, 2005.
11. Jonathan Katz and Ji Sun Shin. Parallel and concurrent security of the hb and hb$^+$ protocols. In Serge Vaudenay, editor, *EUROCRYPT*, volume 4004 of *Lecture Notes in Computer Science*, pages 73–87. Springer, 2006.
12. Akinori Kawachi, Keisuke Tanaka, and Keita Xagawa. Concurrently secure identification schemes based on the worst-case hardness of lattice problems. In *ASIACRYPT '08: Proceedings of the 14th International Conference on the Theory and Application of Cryptology and Information Security*, pages 372–389, Berlin, Heidelberg, 2008. Springer-Verlag.
13. Zhijun Li, Guang Gong, and Zhiguang Qin. Secure and Efficient HB-CM Entity Authentication Protocol. 2009. `http://eprint.iacr.org/`.
14. Richard Lindner and Chris Peikert. Better key sizes (and attacks) for lwe-based encryption. In Aggelos Kiayias, editor, *CT-RSA*, volume 6558 of *Lecture Notes in Computer Science*, pages 319–339. Springer, 2011.
15. Vadim Lyubashevsky. Lattice-based identification schemes secure under active attacks. In Ronald Cramer, editor, *Public Key Cryptography*, volume 4939 of *Lecture Notes in Computer Science*, pages 162–179. Springer, 2008.
16. Vadim Lyubashevsky. Fiat-shamir with aborts: Applications to lattice and factoring-based signatures. In Mitsuru Matsui, editor, *ASIACRYPT*, volume 5912 of *Lecture Notes in Computer Science*, pages 598–616. Springer, 2009.
17. Tatsuaki Okamoto. Provably secure and practical identification schemes and corresponding signature schemes. In Ernest F. Brickell, editor, *CRYPTO*, volume 740 of *Lecture Notes in Computer Science*, pages 31–53. Springer, 1992.
18. Khaled Ouafi, Raphael Overbeck, and Serge Vaudenay. On the security of hb# against a man-in-the-middle attack. In Josef Pieprzyk, editor, *ASIACRYPT*, volume 5350 of *Lecture Notes in Computer Science*, pages 108–124. Springer, 2008.
19. Chris Peikert. Public-key cryptosystems from the worst-case shortest vector problem: extended abstract. In Michael Mitzenmacher, editor, *STOC*, pages 333–342. ACM, 2009.
20. Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. *J. ACM*, 56(6), 2009.
21. Markus Rückert and Michael Schneider. Estimating the security of lattice-based cryptosystems. *IACR Cryptology ePrint Archive*, 2010:137, 2010.
22. Peter W. Shor. Polynominal time algorithms for discrete logarithms and factoring on a quantum computer. In Leonard M. Adleman and Ming-Deh A. Huang, editors, *ANTS*, volume 877 of *Lecture Notes in Computer Science*, page 289. Springer, 1994.

23. Rosemberg Silva, Antonio Campello, and Ricardo Dahab. Lwe-based identification schemes. *CoRR*, abs/1109.0631, 2011.
24. Jacques Stern. A new identification scheme based on syndrome decoding. In Douglas R. Stinson, editor, *CRYPTO*, volume 773 of *Lecture Notes in Computer Science*, pages 13–21. Springer, 1993.
25. Pascal Véron. Improved identification schemes based on error-correcting codes. *Appl. Algebra Eng. Commun. Comput.*, 8(1):57–69, 1996.
26. Keita Xagawa and Keisuke Tanaka. Zero-knowledge protocols for ntru: Application to identification and proof of plaintext knowledge. In Josef Pieprzyk and Fangguo Zhang, editors, *ProvSec*, volume 5848 of *Lecture Notes in Computer Science*, pages 198–213. Springer, 2009.