Lizard: Cut off the Tail!
Practical Post-Quantum Public-Key Encryption from LWE and LWR

Jung Hee Cheon[1], Duhyeong Kim[1], Joohee Lee[1], and Yongsoo Song[1]

[1]Seoul National University (SNU), Republic of Korea
{jhcheon,doodoo1204,skfro6360,lucius05}@snu.ac.kr

Abstract

The Learning with Errors (LWE) is one of the most promising primitive for post-quantum cryptography due to its strong security reduction from the worst-case of NP-hard problems and its lightweight operations. The Public Key Encryption (PKE) scheme based on LWE has a simple and fast decryption, but its encryption is rather slow due to large parameter sizes for Leftover Hash Lemma or expensive Gaussian samplings.

In this paper, we propose a novel PKE without relying on either of them. For encryption, we first combine several LWE instances as in the previous LWE-based PKEs. However, the following step to re-randomize this combination before adding a message is different: remove several least significant bits of ciphertexts rather than inserting errors. We prove that our scheme is IND-CPA secure under the hardness of LWE and can be converted into an IND-CCA scheme in the quantum random oracle model.

Our approach accelerates encryption speed to a large extent and also reduces the size of ciphertexts. The proposed scheme is very competitive for all applications requiring both of fast encryption and decryption. In our single-core implementation in Macbook Pro, encryption and decryption of a 128-bit message for quantum 128-bit security take 7 and 6 microseconds that are 3.4 and 4.2 times faster than those of NTRU PKE, respectively. To achieve these results, we further take some advantage of sparse small secrets, under which the security of our scheme is also proved.

Keywords: Post-Quantum Cryptography, Public-Key Encryption, Learning with Rounding (LWR), Learning with Errors (LWE)

1 Introduction

Since the National Institute of Standards and Technology (NIST) launched the project to develop new quantum-resistant cryptography standards [2], post-quantum cryptography has gained a growing attention at this moment. Lattice based cryptography, one of the most attractive candidates of the post-quantum cryptography, has been studied actively over the last decade due to its distinctive advantages on strong security, fast implementations, and versatility in many applications. In particular, the Learning with Errors (LWE) problem [60] has very attractive features for many usages due to its rigorous reduction from the worst-case of the lattice problems that are NP-hard and regarded to be secure even after the advance of quantum computers.

The LWE problem was first introduced to construct a public key encryption by Regev [60] in 2005. Some well-known variants of Regev's scheme [37, 58] had a drawback requiring somewhat large parameters to be used in practice. It was improved by Lindner and Peikert [47] by taking a method to insert noises to a combination of LWE samples in the encryption stage, but the noise sampling from the discrete Gaussian distribution requires inefficient floating point operations of high bit precision [32]. Recently, several post-quantum key exchanges [27, 57, 20, 19, 9] and one more efficient PKE [24] with sparse small secrets have been proposed on the hardness assumptions of the LWE problem and its

ring variant. They enjoy fast performance in practice as well as its quantum-resistant security, but still suffer from some inefficiency due to Gaussian sampling. Several attempts were made to improve this [56, 19], but not satisfactory yet.

In this paper, we propose a novel LWE-based PKE scheme without Gaussian samplings in the encryption stage which is based on the hardness of *learning with rounding* (LWR) problem. The LWR problem, introduced by Banerjee, Peikert and Rosen [15], is a variant of the LWE problem. Instead of adding small errors, an instance of LWR problem is generated by the deterministic rounding process into a smaller modulus. It is shown in [15, 10] that the LWR problem is not easier than the LWE problem when modulus is somewhat large, but due to this constraints, LWR has been used only for special applications such as pseudorandom generator (PRG) [15]. Our key observation on other LWE-based schemes such as [47] is that a ciphertext generated by an ephemeral secret forms the restricted number of LWE instances of this ephemeral secret together with public information. This leads us to securely make the use of LWR assumption in encryption process relying on the recent result that LWR is hard under the hardness assumption of LWE when the number of samples is bounded [18]. The change of hardness assumption problem to LWR for encryption procedure not only reduces the parameters and ciphertext size, but also substitutes the expensive discrete Gaussian sampling by deterministic and efficient rounding.

Technical Details. Our public key encryption scheme consists of the algorithms KeyGen, Enc, and Dec. In the key generation phase, we choose the private key $\mathbf{s}$ and use it to generate several instances of LWE problem in modulo q. The public key is $(A,\ \mathbf{b} = A\mathbf{s} + \mathbf{e}) \pmod q$, where the error term $\mathbf{e}$ is sampled from the discrete Gaussian distribution. To encrypt a message $m \in \mathbb{Z}_t$, we first generate an ephemeral secret vector $\mathbf{r}$ and calculate $(\mathbf{r}^T A,\ \mathbf{r}^T\mathbf{b} + (q/t) \cdot m)$. Then, we rescale the vector into a lower modulus $p < q$ using the rounding function defined by

$$\mathbb{Z}_q^k \to \mathbb{Z}_p^k \quad , \quad \mathbf{x} \mapsto \lfloor (p/q) \cdot \mathbf{x} \rceil \, ,$$

where k is the number of components of the vector, and $\lfloor \cdot \rceil$ denotes the component-wise rounding of entries to the closest integers. Additionally, the second component of the public key can also be rescaled to reduce the size of public keys.

For efficiency, we take private keys and ephemeral secrets used in our encryption procedure from the set of signed binary vectors in $\{0, \pm 1\}^*$ with small Hamming weights. The Hamming weight of the ephemeral secret vectors has an effect on the error size after subset sum of public data, while the secret key size is related to the error caused by rounding into a smaller modulus p. Therefore, the sparsity of the private keys and ephemeral secrets takes an important role in efficiency of our scheme including parameter size, encryption speed, and ciphertext compression.

Security Analysis and Parameter Selection. Our scheme relies on the hardness of both LWE and LWR problems. By utilizing a reduction from LWE to LWR for some parameters, we show that our scheme is IND-CPA secure under the hardness of LWE. We also extend our proof to the scheme with sparse binary secrets.

For practical parameters, we survey all known attacks against LWE and derive the best attacks for our scheme. We exploit the sparsity of private key and ephemeral secrets to modify the previous attacks against LWE, that is of independent interest. We analyze the LWR problem by the same strategies used in the analyses for LWE. After analyzing both problems with the best known attacks, as following the methodology of security assessment in [19], we propose four parameter sets for our scheme. These parameter sets are determined to make the ciphertext size optimized and the number of samples of LWR restricted, simultaneously. We remark that we provide a *recommended* parameter set for the long-term security, which remains secure against all known quantum attacks.

Three Variants. Additionally, we describe several useful variants of our PKE. First, we propose a ring variant whose security is based on the hardness of ring-LWE and ring-LWR. The public key size

of our LWE-based PKE is compelled to be large because of the matrix structure in LWE and LWR instances. However, due to the compactness of ring structure, the public key size of our ring variant is considerably small compared to that of our original scheme (smaller than the size of one ciphertext.).

Second, we describe an IND-CCA version of our scheme that is secure in the *quantum* random oracle model, applying the Targhi-Unruh CCA conversion [66] on our IND-CPA scheme. Since the conversion is quite light, our IND-CCA scheme is still very efficient.

Finally, we extend our scheme to an additive homomorphic encryption scheme supporting some bounded number of additions. We provide a specific parameter of post-quantum 128-bit security for our scheme. As in [26], additive homomorphic encryptions have numerous applications. Our scheme could be a post-quantum secure alternative for additive homomorphic encryptions. The previous schemes [26, 54, 53] appeared to require large parameters [33] or insecure under the attack using a quantum computer [64].

Implementation and Comparison. We implement the proposed public-key encryption scheme and provide several parameter sets with different level of security as in [19]. We compare our scheme with NTRU [39] and RSA [61] in various (quantum) security levels. The experimental result shows that the size of ciphertexts is reduced more than a half and the encryption (resp. decryption) speed is less than 7 microseconds (μs) (resp. 6 μs) which is about 3 times (resp. 4 times) faster than 24 μs (resp. 25 μs) of NTRU scheme. The ring variant of our scheme takes 12 μs for encryption with reduced size of public information. All the implementations of our schemes were written in C++, and performed on Macbook Pro with an Intel core i5 running at 2.9 GHz processor. The implementation of our schemes will be uploaded at `https://github.com`.

Our scheme has stronger security guarantee than NTRU in the sense that LWE problem has a reduction from the standard lattice problems (GapSVP, SIVP) but NTRU does not have any reductions.[1] Our ring variant has a security level similar to NTRU. Recently, several attacks to exploit ring structure of NTRU have been proposed [4, 25, 17].

Applications in Practice. Our Public-key encryption scheme can be applied to various applications requiring public-key cryptosystems, and gives some advantages on efficiency. One immediate application is for Transport layer security (TLS) which provides secure communication and server-client authentication in the internet using public key cryptosystems such as RSA and Diffie-Hellman (DH) key exchange. Recently, there have been some attempts to replace RSA and DH by quantum-resistant schemes such as NTRU [3] and (ring-)LWE-based key exchange [19, 9].

Our PKE can be an appropriate alternative of RSA and NTRU on TLS protocol due to its quantum-resistant security and faster Enc / Dec speed. For 128-bit security, our scheme has similar ciphertext size with RSA, but our encryption and decryption speeds are about 5 times and 450 times faster than RSA, respectively.

Organization. The rest of the paper is organized as follows. In Section 2, we summarize some notations used in this paper, and introduce *Learning with Errors* (LWE) and *Learning with Rounding* (LWR). We describe our public-key encryption scheme based on both LWE and LWR in Section 3, and provide the concrete analysis and parameters of our scheme in Section 4. In Section 5, we propose some variants of our scheme including ring-based PKE. Finally, we provide implementation results, and compare the performance of our schemes with other encryption schemes in Section 6.

[1] Some NTRU-like encryption schemes [65, 41] with reduction from worst-case of lattice hard problems contain Gaussian sampling process in encryption procedures. Ours is the only choice that does have a reduction and does not consist of Gaussian sampling in encryption.

2 Preliminaries

2.1 Notation

All logarithms are base 2 unless otherwise indicated. For a positive integer q, we use $\mathbb{Z} \cap (-q/2, q/2]$ as a representative of $\mathbb{Z}_q$. For a real number r, $\lfloor r \rceil$ denotes the nearest integer to r, rounding upwards in case of a tie. We denote vectors in bold, *e.g.,* $\mathbf{a}$, and every vector in this paper is a column vector. The norm $\|\cdot\|$ is always 2-norm in this paper. We denote by $\langle \cdot, \cdot \rangle$ the usual dot product of two vectors and shortly write $\langle \cdot, \cdot \rangle_q = \langle \cdot, \cdot \rangle \pmod q$. We use $x \leftarrow D$ to denote the sampling x according to the distribution D. It denotes the uniform sampling when D is a finite set. For an integer $n \geq 1$, D^n denote the product of i.i.d. random variables $D_i \sim D$. We let λ denote the security parameter throughout the paper: all known valid attacks against the cryptographic scheme under scope should take $\Omega(2^\lambda)$ bit operations. For two matrices A and B with the same number of rows, $(A\|B)$ denotes their concatenation, *i.e.,* for $A \in \mathbb{Z}^{m \times n_1}$ and $B \in \mathbb{Z}^{m \times n_2}$, the $m \times (n_1 + n_2)$ matrix $C = (A \parallel B)$ is defined as $c_{ij} = \begin{cases} a_{i,j} & 1 \leq j \leq n_1 \\ b_{i,(j-n_1)} & n_1 < j \leq n_1 + n_2 \end{cases}$.

2.2 Distributions

For a positive integer q, we define $\mathcal{U}_q$ by the uniform distribution over $\mathbb{Z}_q$. For a real $\sigma > 0$, the discrete Gaussian distribution $\mathcal{D}G_\sigma$ is a probability distribution with support $\mathbb{Z}$ that assigns a probability proportional to $\exp(-\pi x^2/\sigma^2)$ to each $x \in \mathbb{Z}$. We will adapt the following simplified lemmas about tail bound on the discrete Gaussian distribution.

Lemma 1 ([13], Lemma 1.5)**.** *Let $c \geq 1$. Then for any real $\sigma > 0$ and any integer $n \geq 1$, we have*

$$\Pr\left[\|\mathbf{e}\| \geq c\sqrt{\frac{n}{2\pi}}\sigma : \mathbf{e} \leftarrow \mathcal{D}G_\sigma^n\right] \leq c^n \exp\left(\frac{n(1-c^2)}{2}\right).$$

Lemma 2 ([14], Lemma 2.4)**.** *For any real $\sigma > 0$ and $t > 0$, and any $\mathbf{x} \in \mathbb{R}^n$, we have*

$$\Pr[|\langle \mathbf{x}, \mathbf{e} \rangle| \geq t \cdot \sigma \cdot \|\mathbf{x}\| : \mathbf{e} \leftarrow \mathcal{D}G_\sigma^n] < 2\exp(-\pi t^2).$$

For an integer $0 \leq h \leq n$, the distribution $\mathcal{H}WT_n(h)$ samples a vector uniformly from $\{0, \pm 1\}^n$, under the condition that it has exactly h nonzero entries.

2.3 Learning with Errors

Since Regev [60] introduced the *learning with errors* (LWE) problem, a lot of cryptosystems based on this problem have been proposed relying on its versatility. For an n-dimensional vector $\mathbf{s} \in \mathbb{Z}^n$ and an error distribution χ over $\mathbb{Z}$, the LWE distribution $A_{n,q,\chi}^{\mathsf{LWE}}(\mathbf{s})$ over $\mathbb{Z}_q^n \times \mathbb{Z}_q$ is obtained by choosing a vector $\mathbf{a}$ uniformly and randomly from $\mathbb{Z}_q^n$ and an error e from χ, and outputting

$$(\mathbf{a}, b = \langle \mathbf{a}, \mathbf{s} \rangle + e) \in \mathbb{Z}_q^n \times \mathbb{Z}_q.$$

The search LWE problem is to find $\mathbf{s} \in \mathbb{Z}_q$ for given arbitrarily many independent samples $(\mathbf{a}_i, b_i)$ from $A_{n,q,\chi}^{\mathsf{LWE}}(\mathbf{s})$. The decision LWE, denoted by $\mathsf{LWE}_{n,q,\chi}(\mathcal{D})$, aims to distinguish the distribution $A_{n,q,\chi}^{\mathsf{LWE}}(\mathbf{s})$ from the uniform distribution over $\mathbb{Z}_q^n \times \mathbb{Z}_q$ with non-negligible advantage, for a fixed $\mathbf{s} \leftarrow \mathcal{D}$. When the number of samples are limited by m, we denote the problem by $\mathsf{LWE}_{n,m,q,\chi}(\mathcal{D})$.

In this paper, we only consider the discrete Gaussian $\chi = \mathcal{D}G_{\alpha q}$ as an error distribution where α is the error rate in $(0, 1)$, so α will substitute the distribution χ in description of LWE problem, say $\mathsf{LWE}_{n,m,q,a}(\mathcal{D})$. The LWE problem is self-reducible, so we usually omit the key distribution $\mathcal{D}$ when it is a uniform distribution over $\mathbb{Z}_q^n$.

The hardness of the decision LWE problem is guaranteed by the worst case hardness of the standard lattice problems: the decision version of the *shortest vector problem* (GapSVP), and the *shortest independent vectors problem* (SIVP). After Regev [60] presented the quantum reduction from those lattice problems to the LWE problem, Peikert et al. [55, 22] improved the reduction to a classical version. In this case, note that the reduction holds only for the GapSVP, not SIVP.

After the works on the connection between the LWE problem and some lattice problems, some variants of LWE, of which the secret distributions are modified from the uniform distribution, were proposed. In [22], Brakerski et al. proved that the LWE problem with binary secret is at least as hard as the original LWE problem. Following the approach of [22], Cheon et al. [24] proved the hardness of the LWE problem with sparse secret, *i.e.*, the number of non-zero components of the secret vector is a constant.

As a result of Theorem 4 in [24], the hardness of the LWE problem with *signed-binary secret of Hamming weight* h, $\mathsf{LWE}_{n,m,q,\beta}(\mathcal{HWT}_n(h))$, is guaranteed by the following theorem.

Theorem 1. *(Informal) If* $\log({}_nC_h)+h > k\log q$ *and* $\beta > \alpha\sqrt{10h}$*, the* $\mathsf{LWE}_{n,m,q,\beta}(\mathcal{HWT}_n(h))$ *problem is at least as hard as the* $\mathsf{LWE}_{k,m,q,\alpha}$ *problem.*

In [59, 58, 21], to pack a string of messages in a ciphertext, LWE with single secret was generalized to LWE with multiple secrets. An instance of multi-secret LWE is $(\mathbf{a}, \langle\mathbf{a},\mathbf{s}_1\rangle+\mathbf{e}_1, ..., \langle\mathbf{a},\mathbf{s}_k\rangle+\mathbf{e}_k)$ where $\mathbf{s}_1, ..., \mathbf{s}_k$ are secret vectors and $\mathbf{e}_1, ..., \mathbf{e}_k$ are independently chosen error vectors. Using the hybrid argument, multi-secret LWE is proved to be at least as hard as LWE with single secret. With this hardness guarantee, we use the LWE instances with a number of sparse signed-binary secrets in our encryption scheme in Section 3.

2.4 Learning with Rounding

The learning with rounding (LWR) problem was firstly introduced by Banerjee et al. [15] to improve the efficiency of pseudorandom generator (PRG) based on the LWE problem. Unlikely to the LWE problem, errors in the LWR problem are deterministic so that the problem is so-called a "derandomized" version of LWE problem. To hide secret information, the LWR problem uses a rounding by a modulus p instead of inserting errors. Then, the deterministic error is created by scaling down from $\mathbb{Z}_q$ to $\mathbb{Z}_p$.

For an n-dimensional vector $\mathbf{s}$ over $\mathbb{Z}_q$, the LWR distribution $A^{\mathsf{LWR}}_{n,q,p}(\mathbf{s})$ over $\mathbb{Z}_q^n \times \mathbb{Z}_p$ is obtained by choosing a vector $\mathbf{a}$ from $\mathbb{Z}_q^n$ uniform randomly, and returning

$$\left(\mathbf{a}, \left\lfloor \frac{p}{q} \cdot \langle\mathbf{a},\mathbf{s}\rangle_q \right\rceil\right) \in \mathbb{Z}_q^n \times \mathbb{Z}_p.$$

As in the LWE problem, $A^{\mathsf{LWR}}_{n,m,q,p}(\mathbf{s})$ denotes the distribution of m samples from $A^{\mathsf{LWR}}_{n,q,p}(\mathbf{s})$; that is contained in $\mathbb{Z}_q^{m\times n} \times \mathbb{Z}_p^m$. The search LWR problem are defined respectively as finding secret $\mathbf{s}$ just as same as the search version of LWE problem. In contrary, the decision $\mathsf{LWR}_{n,m,q,p}(\mathcal{D})$ problem aims to distinguish the distribution $A^{\mathsf{LWR}}_{n,q,p}(\mathbf{s})$ from the uniform distribution over $\mathbb{Z}_q^n \times \mathbb{Z}_p$ with m instances for a fixed $\mathbf{s} \leftarrow \mathcal{D}$.

In [15], Banerjee et al. proved that there is an efficient reduction from the LWE problem to the LWR problem for modulus q of super-polynomial size. Later, the follow-up works by Alwen et al. [10] and Bogdanov et al. [18] improved the reduction by eliminating the restriction on modulus size and adding a condition of the bound of the number of samples. In particular, the reduction by Bogdanov et al. works when $2mBp/q$ is a constant, where B is a bound of errors in the LWE problem, m is the number of samples in both problems, and p is the rounding modulus in the LWR problem. That is, the rounding modulus p is proportional to $1/m$ for fixed q and B. Since the reduction from LWE to LWR is independent of the secret distribution, the hardness of the $\mathsf{LWR}_{n,m,q,p}(\mathcal{HWT}_n(h))$ problem is obtained from that of the LWE problem with signed-binary secret of hamming weight h.

2.5 Ring variants of LWE and LWR

In [49], Lyubashevsky et al. deal with the LWE problem over rings, namely ring-LWE. For positive integers n and q, and an irreducible polynomial $g(x) \in \mathbb{Z}[x]$ of degree n, we define the ring $R = \mathbb{Z}[x]/(g(x))$ and its quotient ring modulo q, $R_q = \mathbb{Z}_q[x]/(g(x))$. The ring-LWE problem is to distinguish between the uniform distribution and the distribution of $(a, a \cdot s + e) \in R_q^2$ where a is uniform randomly chosen polynomial, e is chosen from a error distribution, and s is a secret polynomial.

Due to the efficiency and compactness of ring-LWE, many lattice-based cryptosystems are constructed as *ring*-LWE *based*, rather than LWE-based. Similarly to LWE, the ring-LWE problem over the ring R is at least as hard as GapSVP over the ideal lattices in R.

The ring variant of LWR is introduced in [15, 18] as an analogue of LWR. In the ring-LWR problem, the vectors chosen from $\mathbb{Z}_q^n$ are substituted by polynomials in R_q, *i.e.*, the ring-LWR instance for a secret polynomial $s \in R_q$ is

$$\left(a, \left\lfloor \frac{p}{q} \cdot a \cdot s \right\rceil\right) \in R_q \times R_p.$$

where $\lfloor (p/q) \cdot a \cdot s \rceil$ is obtained by applying the rounding function to each coefficient of $(p/q) \cdot a \cdot s$. The search and decision ring-LWR problems are defined the same way as the LWR problem, but over rings.

In [15], Banerjee et al. proved that decision ring-LWR is at least as hard as decision ring-LWE for sufficiently large modulus. Later, reduction from search ring-LWE to search ring-LWR was constructed in overall scope of the modulus [15] when the number of samples is bounded.

3 LWR-Based Public-key Encryption Scheme

In this section, we present a (probabilistic) LWR-based public-key encryption scheme. Note that the LWE hardness assumption is also used in our scheme even though the scheme is described as "LWR-*based*".

Our construction has several advantages: one is that we could compress the ciphertext size by scaling it down from $\mathbb{Z}_q$ to $\mathbb{Z}_p$ where p is the rounding modulus, and the other is that we speed up the encryption algorithm by eliminating the Gaussian sampling process.

3.1 The Construction

We present a public-key encryption scheme based on both LWE and LWR problems with *sparse signed-binary secrets*. The public key consists of m number of n dimensional LWE instances with a signed-binary secret distribution of Hamming weight h_s. On the other hand, encryptions of zero form $(n + \ell)$ instances of m dimensional LWR with signed binary secret of Hamming weight h_r. The scheme is described as follows:

- $\mathsf{Setup}(1^\lambda)$: Choose positive integers m, n, h_s, h_r, ℓ, t, p, and q satisfying $h_s < n$, $h_r < m$, and $t \mid p \mid q$. Fix an error parameter α with $0 < \alpha < 1$. Output $params \leftarrow (m, n, h_s, h_r, \ell, t, p, q, \alpha)$.
- $\mathsf{KeyGen}(params)$: Generate a random matrix $A \leftarrow \mathbb{Z}_q^{m \times n}$. Sample a signed binary secret matrix $S = (\mathbf{s}_1 \| \cdots \| \mathbf{s}_\ell)$ from $\mathcal{HWT}_n(h_s)^\ell$, and an error matrix $E = (\mathbf{e}_1 \| \cdots \| \mathbf{e}_\ell)$ from $\mathcal{D}_{\alpha q}^{m \times \ell}$. Let $B \leftarrow -AS + E \in \mathbb{Z}_q^{m \times \ell}$. Output the public key $\mathsf{pk} \leftarrow (A \| B) \in \mathbb{Z}_q^{m \times (n+\ell)}$ and the secret key $\mathsf{sk} \leftarrow S \in \{0, \pm 1\}^{n \times \ell}$.
- $\mathsf{Enc}_{\mathsf{pk}}(\mathbf{m})$: For a message $\mathbf{m} \in \mathbb{Z}_t^\ell$, choose a vector $\mathbf{r} \leftarrow \mathcal{HWT}_m(h_r)$. Compute the vectors $\mathbf{a}' \leftarrow A^T \mathbf{r}$ and $\mathbf{b}' \leftarrow B^T \mathbf{r}$, and output the vector

$$\mathbf{c} \leftarrow (\mathbf{a}, \mathbf{b}) \in \mathbb{Z}_p^{n+\ell}$$

where $\mathbf{a} \leftarrow \lfloor (p/q) \cdot \mathbf{a}' \rceil \in \mathbb{Z}_p^n$ and $\mathbf{b} \leftarrow (p/t) \cdot \mathbf{m} + \lfloor (p/q) \cdot \mathbf{b}' \rceil \in \mathbb{Z}_p^\ell$.

- $\mathsf{Dec}_{\mathsf{sk}}(\mathbf{c})$: For a ciphertext $\mathbf{c} = (\mathbf{a}, \mathbf{b}) \in \mathbb{Z}_p^{n+\ell}$, compute and output the vector

$$\mathbf{m}' \leftarrow \left\lfloor \frac{t}{p}(S^T\mathbf{a} + \mathbf{b}) \right\rceil \pmod t.$$

Remark 1. The size of the public key $\mathsf{pk} = (A\|B)$ can be compressed by using a pseudorandom generator $G : \mathbb{Z}_2^k \to \mathbb{Z}_q^{m\times n} (\simeq \mathbb{Z}_2^{mn\log q})$, where $k < mn\log q$. In the KeyGen phase, instead of choosing the matrix A uniform randomly, choose a vector $\mathbf{v} \in \mathbb{Z}_2^k$ uniform randomly and let $A = G(\mathbf{v})$. Then, we can set the *compressed* public key as $\mathsf{pk}' = (\mathbf{v}, B) \in \mathbb{Z}_2^k \times \mathbb{Z}_q^{m\times\ell}$.

3.2 Correctness and Security

We argue that the proposed encryption scheme is *IND-CPA secure* under the hardness assumptions of the LWE problem with sparse signed binary secrets. The following theorem gives an explicit proof of our argument on security.

Theorem 2 (Security)**.** *The public key encryption scheme* $(\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ *is IND-CPA secure under the hardness assumption of the* $\mathsf{LWE}_{n,m,q,\alpha}(\mathcal{HWT}_n(h_s))$ *problem and the* $\mathsf{LWR}_{m,n+\ell,q,p}(\mathcal{HWT}_m(h_r))$ *problem.*

Proof. It is enough to show that the distribution of $((A\|B), \mathbf{c})$ generated by $params \leftarrow \mathsf{Setup}(1^\lambda)$, $\mathsf{pk} = (A\|B) \leftarrow \mathsf{KeyGen}(params)$, and $\mathbf{c} \leftarrow \mathsf{Enc}_{\mathsf{pk}}(\mathbf{0})$ is computationally indistinguishable from the uniform distribution over $\mathbb{Z}_q^{m\times(n+\ell)} \times \mathbb{Z}_q^{n+\ell}$.

First, $\mathsf{pk} = (A\|B)$ is generated by sampling m instances of n dimensional LWE problem of error parameter α with ℓ independent secret vectors $\mathbf{s}_1, \ldots, \mathbf{s}_\ell \leftarrow \mathcal{HWT}_n(h_s)$. In addition, the multi-secrets LWE problem is no easier than the ordinary LWE problem as noted in Section 2.3. Hence, pk is computationally indistinguishable from the uniform over $\mathbb{Z}_q^{m\times(n+\ell)}$ under the $\mathsf{LWE}_{n,m,q,\alpha}(\mathcal{HWT}_n(h_s))$ assumption.

Now assume that pk is uniform random over $\mathbb{Z}_q^{m\times(n+\ell)}$. Then pk and $\mathbf{c} \leftarrow \mathsf{Enc}_{\mathsf{pk}}(\mathbf{0})$ together form $(n+\ell)$ instances of the m dimensional LWR problem with secret $\mathbf{r} \leftarrow \mathcal{HWT}_m(h_r)$, which is computationally indistinguishable from the uniform over $\mathbb{Z}_q^{m\times(n+\ell)} \times \mathbb{Z}_p^{(n+\ell)}$ under the $\mathsf{LWR}_{m,n+\ell,q,p}(\mathcal{HWT}_m(h_r))$ assumption.

□

Theorem 3 (Correctness)**.** *Let* $params \leftarrow \mathsf{Setup}(1^\lambda)$ *and* $(\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{KeyGen}(params)$. *Let* $\mathbf{c} \leftarrow \mathsf{Enc}_{\mathsf{pk}}(\mathbf{m})$ *be an encryption of a plaintext* $\mathbf{m} \in \mathbb{Z}_t^\ell$. *Then its decryption result* $\mathbf{m}' \leftarrow \mathsf{Dec}_{\mathsf{sk}}(\mathbf{c})$ *is equal to* $\mathbf{m}$ *with an overwhelming probability in the security parameter* λ *if*

$$\eta \cdot \alpha p \cdot \sqrt{h_r} + \frac{1}{2} + \xi \cdot h_s \le \frac{p}{2t}$$

for some positive numbers η *and* ξ *such that* $\eta^2 \ge \frac{\lambda}{\pi \log e}$ *and* $(e \cdot (1/2 - \xi))^{h_s} \le 2^{-\lambda}$.

Proof. Let $\mathbf{r} \in \{0, \pm 1\}^m$ be a vector sampled from $\mathcal{HWT}_m(h_r)$ in our encryption procedure, and let $\mathbf{c}' = (\mathbf{a}', \mathbf{b}') \leftarrow (A^T\mathbf{r}, B^T\mathbf{r}) \in \mathbb{Z}_q^{n+\ell}$. Let $\mathbf{u} \leftarrow \lfloor (p/q) \cdot \mathbf{a}' \rceil - (p/q) \cdot \mathbf{a}' \in (-1/2, 1/2]^n$ and $\mathbf{v} \leftarrow \lfloor (p/q) \cdot \mathbf{b}' \rceil - (p/q) \cdot \mathbf{b}' \in (-1/2, 1/2]^\ell$ be the negative fractional part of vectors $(p/q) \cdot \mathbf{a}'$ and $(p/q) \cdot \mathbf{b}'$, respectively. Then we have

$$S^T\mathbf{a} + \mathbf{b} = (p/t) \cdot \mathbf{m} + (p/q) \cdot (S^T\mathbf{a} + \mathbf{b}') + (S^T\mathbf{u} + \mathbf{v}).$$

Note that its i-th entry is

$$(p/t) \cdot m_i + (p/q) \cdot \langle \mathbf{e}_i, \mathbf{r} \rangle + \langle \mathbf{s}_i, \mathbf{u} \rangle + v_i \pmod{p},$$

for any $i = 1, \ldots, \ell$, and so the correctness of our scheme is guaranteed as long as the error term $(p/q) \cdot \langle \mathbf{e}_i, \mathbf{r} \rangle + \langle \mathbf{s}_i, \mathbf{u} \rangle + v_i$ is bounded by $(p/2t)$ with an overwhelming probability.

1. Entries of $\mathbf{e}_i$ are sampled from i.i.d. discrete Gaussian distribution of parameter σ, and $\mathbf{r} \leftarrow \mathcal{HWT}_m(h_r)$ has the constant norm $\|\mathbf{r}\| = \sqrt{h_r}$. From Lemma 2, the value $\langle \mathbf{e}_i, \mathbf{r} \rangle$ is bounded by $\eta \cdot \alpha q \cdot \sqrt{h_r}$ with a probability $\geq 1 - 2\exp(-\pi\eta^2)$. In this case, its scaled value $(p/q) \cdot \langle \mathbf{e}_i, \mathbf{r} \rangle$ is bounded by $\eta \cdot \alpha p \cdot \sqrt{h_r}$.

2. Trivially, each v_i is bounded by $1/2$.

3. We can heuristically assume that entries of $\mathbf{u}$ seem to be i.i.d. uniform random variables on the interval $(-1/2, 1/2]$, and so $\langle \mathbf{u}, \mathbf{s}_i \rangle$ seems to be the sum of h_s-number of uniform random variable on $(-1/2, 1/2]$. By using the Chernoff bound, the tail probability of sum of i.i.d. uniform random variables can be bounded by

$$\Pr[|\langle \mathbf{s}_i, \mathbf{u} \rangle| \geq \xi \cdot h_s] \leq 2 \left(e \cdot \left(\frac{1}{2} - \xi \right) \right)^{h_s}$$

for any $\xi > 0$.

Thus, putting these analysis together, the correctness of our scheme holds with an overwhelming probability if

$$\eta \cdot \alpha p \cdot \sqrt{h_r} + \frac{1}{2} + \xi \cdot h_s \leq \frac{p}{2t}$$

for some numbers $\eta > 0$ and $\xi > 0$ such that $\exp(-\pi\eta^2)$ and $(e \cdot (1/2 - \xi))^{h_s}$ are bounded by $2^{-\lambda}$. □

For a security parameter $\lambda = 128$ and $h_s = 64$, to make $\exp(-\pi\eta^2)$ and $(e \cdot (1/2 - \xi))^{h_s}$ both negligible in the security parameter $\lambda = 128$, we set $\eta = 5.3$ and $\xi = 0.41$.

4 Analysis and Parameter Selection

In this section, we analyze the parameter conditions to provide conservative security against both quantum and classical attacks. Our perspective is very rigorous, and the proposed parameters can be exploited in the real world for the long-term security.

In the following subsection, we would introduce and apply the known attacks against LWE and LWR problems with sparse small secrets, and analyze the required conditions of secure parameter. Since the number of problem samples for each problem is bounded, we exclude the BKW attacks [6, 5, 28, 42] and the linearization attacks [11]. This leaves us the attacks using lattice basis reduction algorithms [63, 23, 46, 52], which can be categorized as follows:

- We can convert the LWE problem to the Short Integer Solution (SIS) problem. The *distinguishing attack* analyzed in [50, 62] is this kind of attack which is extended to the *dual attack*.
- Regarding LWE as the Bounded-Distance Decoding (BDD) problem, we can reduce it to the unique-SVP (uSVP). The *embedding attack* analyzed in [48, 7] has this kind of attack strategy, which is extended to the *primal attack*.
- There are various techniques to adapt the above two strategies for the small secret variants of LWE, e.g. the *modulus switching* [29] and the Bai and Galbraith's rescaling technique for the embedding attack [12].

The main strategy of these attacks is to build an arbitrary lattice in which a short vector gives a hint to solve the LWE problem. Running a lattice basis reduction algorithm for the target lattice, we achieve a required short vector. We plugged the BKZ algorithm [63, 23] in the attacks, which is the most powerful lattice basis reduction algorithm and outputs sufficiently short basis of a lattice according to the time complexity. The BKZ algorithm consists of exact SVP calculations in smaller dimensions as a subroutine. There are several methods to do this by computing the Voronoi cell of the lattice, sieving [16, 43, 44, 45] or enumeration [36, 38].

Let Λ be a target lattice of dimension n. It is generally accepted that the norms of the shortest vectors in the output of BKZ is approximately

$$\|\mathbf{b}_1\| = \delta^n \cdot \det(\Lambda)^{1/n},$$

where δ converges to a constant rapidly as n grows. The constant δ, called *root Hermite factor*, is used to measure the security of lattice problems. In other words, the runtime of the BKZ algorithm to achieve a given root Hermite factor in large dimension (> 200) is determined heuristically by δ only. In analysis, once we obtain a condition of the root Hermite factor to make an attack successful, then we can calculate the attack complexity against LWE (LWR) from δ. Hence, we would analyze the conditions of δ for the several attack strategies.

4.1 Known Attacks

The LWE and LWR problems used in our scheme have small secrets and publish a limited number of instances. Hence, some methods which do not use the sparsity of secret vector such as distinguishing attack and embedding attack are not well-fitted in our cases. Instead, we mainly consider two attack strategies from the same stems, respectively: the dual attack and the primal attack. In the former approach, we try to find a short vector of a random lattice generated from matrix A, and use it to check if $\mathbf{b}$ is independently chosen from this short vector or not. In the latter approach, we use both A and $\mathbf{b}$ to construct a lattice containing a particular short vector, and try to solve uSVP. We further optimize these attack strategies for the short secret variants of LWE as in the Bai and Galbraith's embedding attack, and gives elegant formulas.

We first describe the attacks on the LWE problem and analyze their complexities. For the LWR problem, we first transform the LWR instances by simply multiplying (q/p) on the last components, and then apply the same attacks on the transformed instances that has a look of LWE instances in the attacker's view. We mention about this transformation more precisely after the following analysis on the known attacks.

4.1.1 Weighted Dual Attack

Let $(A, \mathbf{b} = A\mathbf{s} + \mathbf{e}) \in \mathbb{Z}_q^{m\times(n+1)}$ be given LWE samples of dimension n. In the original dual attack, an attacker constructs a lattice

$$\Lambda = \{(\mathbf{x}, \mathbf{y}) \in \mathbb{Z}^m \times \mathbb{Z}^n : \mathbf{x}^T A = \mathbf{y}^T \pmod q\}$$

that is the orthogonal lattice of the matrix $(-A\|I_n)$ modulo q. one can find a short vector $\mathbf{v} = (\mathbf{x}, \mathbf{y})$ in Λ using BKZ and then check if $\langle \mathbf{x}, \mathbf{b}\rangle \pmod q$ is small or not. If $(A, \mathbf{b})$ is an $\mathsf{LWE}_{n,m,q,\alpha}(\mathcal{HWT}_n(h_s))$ instance with secret $\mathbf{s}$ and $\langle \mathbf{x}, \mathbf{b}\rangle$ is less than q in $\mathbb{Z}$, then $\langle \mathbf{x}, \mathbf{b}\rangle = \langle \mathbf{y}, \mathbf{s}\rangle + \langle \mathbf{x}, \mathbf{e}\rangle$ behaves as a Gaussian, otherwise it is distributed uniformly. Hence, if the attacker can find and collect short vectors $\mathbf{v} = (\mathbf{x}, \mathbf{y})$ in Λ such that $\langle \mathbf{x}, \mathbf{b}\rangle < q$, then the attacker would solve the distinguish problem.

Our observation is that since the secret $\mathbf{s}$ is a sparse signed binary vector, the term $\langle \mathbf{y}, \mathbf{s}\rangle$ is somewhat smaller than $\langle \mathbf{x}, \mathbf{e}\rangle$. From this point, we optimize this attack when the variances of the

components in the secret vector $\mathbf{s}$ are significantly smaller than those of the error vector $\mathbf{e}$: We consider a weighted lattice

$$\Lambda' = \{(\mathbf{x}, \mathbf{y}') \in \mathbb{Z}^m \times (w^{-1} \cdot \mathbb{Z})^n : (\mathbf{x}, w \cdot \mathbf{y}) \in \Lambda\}$$

for some positive number $w > 0$. The optimal choice of w is

$$w = (\alpha q)\sqrt{\frac{n}{2\pi h_s}}$$

for reconciliation of variances $w^2 \cdot (h_s/n)$ and $(\alpha q)^2/(2\pi)$ of $w \cdot s_i$ and e_j, respectively.

Let $\hat{q} = q/w = \sqrt{2\pi h_s/n} \cdot \alpha^{-1}$. The lattice Λ' has the dimension $(m+n)$ and the volume $\hat{q}^n$. Hence, the BKZ algorithm outputs a short vector $\mathbf{v} = (\mathbf{x}, \mathbf{y}')$ of size $\|\mathbf{v}\| \approx \delta^{m+n} \cdot (\hat{q})^{\frac{n}{m+n}}$ which can be reduced down to $2^{2\sqrt{n \log \hat{q} \cdot \log \delta}}$ when $m + n = \sqrt{n \log \hat{q}/\log \delta}$. Then $\langle \mathbf{x}, \mathbf{b} \rangle = \langle \mathbf{y}, w \cdot \mathbf{s} \rangle + \langle \mathbf{x}, \mathbf{e} \rangle$ is distributed as a Gaussian centered around zero and of standard deviation $\sigma = \|\mathbf{v}\| \cdot (\alpha q/\sqrt{2\pi})$ by central limit theorem (CLT). If $\sqrt{2\pi}\sigma < q$, then $\langle \mathbf{x}, \mathbf{b} \rangle$ can be distinguished from the uniform distribution modulo q with advantage about $\frac{1}{23}$ [8]. Therefore, the $\mathsf{LWE}_{n,m,q,\alpha}(\mathcal{HWT}_n(h_s))$ problem is secure only if

$$\frac{n \log \hat{q}}{\log^2 \alpha} \geq \frac{1}{4 \log \delta},$$

where $\hat{q} = \sqrt{2\pi h_s/n} \cdot \alpha^{-1}$.

4.1.2 Weighted Primal Attack

The key idea of the primal attack is the reduction from LWE to unique-SVP over a special lattice generated by a LWE instance. If the gap between λ_1 and λ_2 of this lattice is large enough, an attacker may find the shortest vector using the BKZ algorithm.

For a given $\mathsf{LWE}_{n,m,q,\alpha}(\mathcal{HWT}_n(h_s))$ instance $(A, \mathbf{b} = A\mathbf{s} + \mathbf{e}) \in \mathbb{Z}_q^{m \times (n+1)}$, construct the lattice

$$\Lambda = \{\mathbf{v} \in \mathbb{Z}^{n+m+1} : (A \| I_m \| - \mathbf{b})\, \mathbf{v} = 0 \pmod q\}$$

with the unique shortest vector $(\mathbf{s}, \mathbf{e}, 1)$. Similarly to the case of dual attack, we consider the weighted lattice

$$\Lambda' = \{(\mathbf{x}, \mathbf{y}', z) \in \mathbb{Z}^n \times (w^{-1}\mathbb{Z})^m \times \mathbb{Z} : (\mathbf{x}, w \cdot \mathbf{y}, z) \in \Lambda\}.$$

for the constant $w = (\alpha q)\sqrt{(n/2\pi h_s)}$, which contains the unique shortest vector $(\mathbf{s}, w^{-1} \cdot \mathbf{e}, 1)$.

Let $\hat{q} = q/w = \sqrt{2\pi(h_s/n)} \cdot \alpha^{-1}$. Since the lattice Λ' has the dimension $n + m + 1$ and the volume $\hat{q}^m$, we get $\lambda_2(\Lambda') \approx \sqrt{\frac{m+n+1}{2\pi e}}\, \hat{q}^{\frac{m}{m+n+1}}$ by the Gaussian heuristic. The attacker succeeds to find the unique-SVP solution $(\mathbf{s}, w^{-1}\mathbf{e}, 1)$ if

$$\frac{\lambda_2(\Lambda')}{\lambda_1(\Lambda')} \approx \frac{\sqrt{\frac{m+n}{2\pi e}}\hat{q}^{\frac{m}{m+n}}}{\sqrt{m+n} \cdot \alpha\hat{q}} = \frac{\alpha^{-1}}{\sqrt{2\pi e} \cdot \hat{q}^{\frac{n}{m+n}}} \geq \tau \cdot \delta^{m+n}$$

for a constant $0 < \tau < 1$. To minimize the complexity, an attacker may choose $m + n = \sqrt{\frac{n \log \hat{q}}{\log \delta}}$ which yields $\hat{q}^{\frac{n}{m+n}} \cdot \delta^{m+n} = 2^{2\sqrt{n \log \hat{q} \cdot \log \delta}}$. Therefore, the $\mathsf{LWE}_{n,m,q,\alpha}(\mathcal{HWT}_n(h_s))$ problem is secure against the primal attack only if

$$\frac{n \log \hat{q}}{\log^2 \hat{\alpha}} \geq \frac{1}{4 \log \delta}$$

for $\hat{q} = \sqrt{2\pi(h_s/n)} \cdot \alpha^{-1}$ and $\hat{\alpha} = (\sqrt{2\pi e} \cdot \tau)\alpha$.

The constant τ is a constant that can be experimentally determined. Gama and Nguyen [35] and Albrecht et al. [7] estimated τ within the range [0.18, 0.48] for some special lattices. Observing those results, we assume a mild conjecture: $\tau \geq 0.242$. Then, the balanced dual attack is the best attack in our cases.

4.1.3 Weighted Dual and Primal attacks on LWR

Now we return to the LWR problem. For a given LWR instance $(A, \mathbf{b} = \lfloor (p/q) \cdot A\mathbf{r} \rceil) \in \mathbb{Z}_q^{m \times n} \times \mathbb{Z}_p^m$, we compute

$$\frac{q}{p} \cdot \mathbf{b} = \frac{q}{p} \cdot \left\lfloor \frac{p}{q} \cdot A\mathbf{r} \right\rceil = A\mathbf{r} + \mathbf{t}$$

where $\mathbf{t} \in (-q/2p, q/2p]^m$. We can heuristically assume that the rounding error $\mathbf{t}$ seems to be an uniform random variable on $(-q/2p, q/2p]^m$. Therefore, in the view of attacker, the transformed instance $(A, (q/p) \cdot \mathbf{b})$ is regarded as a LWE instance, and the attacks on LWE described above can be applied to $(A, (q/p) \cdot \mathbf{b})$.

Since the variance of uniform random variable on $(-q/2p, q/2p]$ is $(q^2/12p^2)$, the parameter conditions to make LWR secure against the attacks can be obtained by simply substituting α with $p^{-1}\sqrt{\pi/6}$. The following inequalities are the conditions for $\mathsf{LWR}_{m,n+\ell,q,p}(\mathcal{H}WT_m(h_r))$ to be secure against the weighted primal and dual attacks, respectively.

- Weighted dual attack:
$$\frac{m \log \hat{q}}{\log^2 \hat{p}} \geq \frac{1}{4 \log \delta}$$
for $\hat{p} = \sqrt{6/\pi} \cdot p$ and $\hat{q} = p\sqrt{12h_r/m}$.

- Weighted primal attack:
$$\frac{m \log \hat{q}}{\log^2 \hat{p}} \geq \frac{1}{4 \log \delta}$$
for $\hat{q} = p\sqrt{12(h_r/m)}$ and $\hat{p} = (\sqrt{3/\pi^2 e} \cdot \tau^{-1})p$.

4.2 Measuring BKZ Complexity

In this subsection, we investigate and fix the root Hermite factor such that the minimum attack complexities for given δ exceed 2^λ.

For the BKZ algorithm, we review the relations among the root Hermite factor δ, the block size b, and the time complexity T as follows.

- (pessimistic) T can be estimated as 2^{cb} (about $b2^{cb}$ CPU cycles) in our scheme as in [19] and [9], where c is some constant. This is an approximate lower bound of the complexity for a single SVP calculation using the sieve algorithm [16, 43, 44, 45].
- $\delta = ((\pi b)^{1/b} \cdot b/2\pi e)^{1/2(b-1)}$.

From this, if we fix the constant c, we can calculate T from a given δ.

According to the constant c, we consider the three cases, $c = c_C$ for classical security, $c = c_Q$ for quantum security, and $c = c_P$ for very pessimistic view, following analyses in [19]. On the classical view, the constant c has been studied for a long time, reaching $c_C = 0.292$ (See in [16, 44]). Quantum attacks make the constant c decline. The best known constant is achieved by applying Grover's quantum search algorithm to those sieve algorithms [45, 43], resulting in decrease of c to $c_Q = 0.265$. Since all the algorithms require building lists of $(4/3)^{b/2} = 2^{0.2075}$ vectors, we set $c_P = 0.2075$ as a pessimistic lower bound of the constant c.

Hence, in the each point of view, to make the BKZ-style attack infeasible for security parameter $\lambda = 128$, we should set the parameters such that the best attack is successful only when $\delta_C \leq 1.003922$, $\delta_Q \leq 1.00367$, and $\delta_P \leq 1.00309$, respectively. This can be shown in simple calculations visualized in Table 1.

c	T	b	δ
0.292	$2^{128.1}$	409	1.003922
0.265	2^{128}	450	1.00367
0.2075	2^{128}	573	1.00309

Table 1: Our views in BKZ complexity; estimated root Hermite factor δ for BKZ running time 2^{128} (in cycles).

4.3 Proposed Parameters

Some parameter sets derived from the security estimates against both classical and quantum attacks are described in Table 2. To satisfy the correctness condition in Theorem 3, we fix the message modulus $t = 2$, Hamming weights $h_s = h_r = 64$ achieving security guarantee against the exhaustive search of secrets for $\lambda = 128$. We provide several parameter sets with different level of security as in [19] assuring an overwhelming probability of decryption correctness.

	m	n	$\log q$	p	α^{-1}
Challenge	274	303	11	178	419
Classical	361	386	11	186	391
Recommended	383	414	11	184	400
Paranoid	470	504	12	188	389

Table 2: Suggested parameter sets for fixed $\lambda = 128$ and message bit length $\ell = 128$; m and n are dimensions of LWR and LWE, respectively. q is a large modulus in LWE and LWR, and p is a rounding modulus in LWR. α is an error rate in LWE.

Challenge Parameters. This parameter set can be a challenge for the state-of-the-art cryptanalysis. We set the root Hermite factor $\delta = 1.00488$ for setting this parameter set, and it takes more than 2^{128} time complexity to achieve the root Hermite factor with current best BKZ algorithm when the dimension of a target lattice is sufficiently high (> 200). However, we do not guarantee the security of this parameter.

Classical Parameters. This parameter set supplies the 128 bits of security against the classical attacks, but not enough against quantum attacks.

Recommended Parameters. It provides 128-bit security against all the known quantum attacks. We recommend to use this parameter for the long-term security.

Paranoid Parameters. This parameter set would remain secure and have 128 bis of security against quantum attacks even if a remarkable improvement towards solving SVP arises.

Note that the bit size of ciphertexts is very small, and it was not achievable in previous constructions based on the hardness of LWE. The following Table 3 shows the time complexity for solving $\mathsf{LWE}_{n,m,q,\alpha}(\mathcal{HWT}_n(h_s))$ and $\mathsf{LWR}_{m,n,q,p}(\mathcal{HWT}_m(h_r))$. We considered the best known attacks against these problems described in the previous subsection. In the table, the column labeled b denotes the required block size of the BKZ algorithm to achieve a root Hermite factor which draws the best known attack successful against $\mathsf{LWE}_{n,m,q,\alpha}(\mathcal{HWT}_n(h_s))$ and $\mathsf{LWR}_{m,n,q,p}(\mathcal{HWT}_m(h_r))$ respectively, and the

values in the columns labeled C, Q, and P shows the bit size of required time complexity in CPU cycles measured with the constants c_C, c_Q, and c_P, respectively.

Parameter	Problem	b	C	Q	P
Challenge	LWE	297	-	-	-
	LWR	297	-	-	-
Classical	LWE	418	**131**	119	95
	LWR	425	**133**	121	97
Recommended	LWE	456	142	**130**	103
	LWR	460	143	**131**	104
Paranoid	LWE	590	181	166	**132**
	LWR	595	183	167	**133**

Table 3: Attack Complexity of LWE and LWR for the best attack on the suggested parameter sets according to our analysis. The best classical attack takes 2^C operations, the best quantum attack takes 2^Q, and 2^P is the lower bound runtime that is the list size of vectors in the sieve algorithms. Numbers in bold type point to the security claim for the particular parameter set. For example, the Recommended set provides 128-bit post-quantum security.

5 Variants of Our Scheme

In this section, we propose some variants of our public-key encryption scheme introduced in Section 3; its ring variant, IND-CCA secure encryption scheme, (bounded) additive homomorphic encryption scheme, and symmetric encryption scheme.

5.1 Ring variant of our PKE

Our scheme has a natural analogue based on the harness of Ring-LWE and Ring-LWR problems. Although the security ground of ring variant of our scheme is weaker than that of original scheme based on LWE and LWR, the ring variant exploits better key sizes, message expansion rate, and Enc/Dec speed.

We bring some notations for the description of our ring-based encryption scheme. For an integer d, let $\Phi_d(X)$ be the d-th cyclotomic polynomial of degree $n = \phi(d)$. We write the cyclotomic ring and its residue ring modulo an integer q by $R = \mathbb{Z}[X]/(\Phi_d(X))$ and $R_q = \mathbb{Z}_q[X]/(\Phi_d(X))$. We identify the vectors of $\mathbb{Z}_q^n$ with the elements of R_q by $(a_0, ..., a_{n-1}) \mapsto \sum_{i=0}^{n-1} a_i X^i$. For the simplicity of ring operations, we choose a power-of-two degree in the following description.

- $\mathsf{Setup}^{ring}(1^\lambda)$: Choose positive integers t, p, and q satisfying $t|p|q$. Let $n \in \mathbb{Z}$ be a power of 2 and $\Phi(X) = X^n + 1$ be the $2n$-th cyclotomic polynomial. Choose h_s, h_r less than or equal to n. Fix an error parameter $0 < \alpha < 1$. Output $params \leftarrow (n, h_s, h_r, t, p, q, \alpha)$.

- $\mathsf{KeyGen}^{ring}(params)$: Generate a random polynomial $a \leftarrow R_q$. Sample a secret polynomial $s \leftarrow \mathcal{HWT}_n(h_s)$, and an error polynomial $e \leftarrow \mathcal{DG}_{\alpha q}^n$. Let $b = -a \cdot s + e \in R_q$. Output the public key $\mathsf{pk} \leftarrow (a, b) \in R_q^2$ and the secret key $\mathsf{sk} \leftarrow s \in R_2$.

- $\mathsf{Enc}_{\mathsf{pk}}^{ring}(\mathbf{m})$: For a message $m \in R_t = R/tR$, choose $r \leftarrow \mathcal{HWT}_m(h_r)$, and compute $a' \leftarrow a \cdot r$ and $b' \leftarrow b \cdot r$. Output the vector

$$\mathbf{c} \leftarrow (c_1, c_2) \in R_p^2,$$

where $c_1 \leftarrow \lfloor (p/q) \cdot a' \rceil \in R_p$ and $c_2 \leftarrow (p/t) \cdot m + \lfloor (p/q) \cdot b' \rceil \in R_p$.

- $\mathsf{Dec}_{\mathsf{sk}}^{ring}(\mathbf{c})$: For a ciphertext $\mathbf{c} = (c_1, c_2)$, compute and output the polynomial

$$m' \leftarrow \left\lfloor \frac{t}{p}(c_1 \cdot s + c_2) \right\rceil \pmod{t} \ \in R_t.$$

Note that all the polynomial multiplications with s or r required in key generation, encryption, and decryption phases can be done very efficiently by shifting and adding vectors.

5.1.1 Parameter Consideration

Since the best known attacks do not utilize the ring structure so far, we analyze the hardness of Ring-LWE as an LWE problem without ring structure as in the previous section. Setting $h_s = h_r = 64$, we can achieve our parameter set: we recommend to use the parameter

$$n = 512, \quad \log q = 12, \quad p = 128, \quad \alpha^{-1} = 957 \tag{1}$$

to resist all known quantum attacks for the security parameter $\lambda = 128$. For the Challenge and Classical parameter sets, since n should be a power of two, just use the same set as in the condition (1) with $q = 2^{11}$.

5.1.2 Hardness of Ring-LWR

There have been many progress in studying the hardness of the ring-LWR problem. Banergee et al. [15] proved that the decision version of the ring-LWR problem is harder than that of the ring-LWE problem for large modulus. Bogdanov et al. [18] extended the scope of the modulus, but the extension holds only for the search version of the ring-LWR problem. They stated that the search version of the ring-LWR problem is not easier than that of the ring-LWE problem when the number of samples is bounded with a flexible upper bound in Theorem 3 in [18].

5.2 IND-CCA Secure Encryption Scheme

Following the hybrid conversion technique [34], we can easily convert our IND-CPA secure encryption scheme into an encryption scheme that is IND-CCA secure in the Random Oracle Model (ROM). In this subsection, in particularly, we introduce an encryption scheme that is IND-CCA secure in the *Quantum Random Oracle Model* (QROM) following the conversion in [66].

We define three hash functions $G : \mathbb{Z}_t^\ell \to \{0,1\}^d$, $H : \{0,1\}^* \to R$, and $H' : \mathbb{Z}_t^\ell \to \mathbb{Z}_t^\ell$, where $\{0,1\}^d$ is a message space of the IND-CCA secure encryption scheme and R is a set of vectors $\mathbf{r} \in \{0, \pm 1\}^m$ of Hamming weight h_r. Our IND-CCA public-key encryption scheme is a hybrid encryption scheme of the IND-CPA public-key encryption scheme in Section 3 and the One-Time pad as a symmetric encryption scheme required for the conversion.

- $\mathsf{Setup}^{hy}(1^\lambda)$: Take $params = (m, n, h_s, h_r, \ell, t, p, q, \alpha) \leftarrow \mathsf{Setup}(1^\lambda)$. Choose hash functions $G : \mathbb{Z}_t^\ell \to \{0,1\}^d$, $H : \{0,1\}^* \to R$, and $H' : \mathbb{Z}_t^\ell \to \mathbb{Z}_t^\ell$.

- $\mathsf{KeyGen}^{hy}(params)$: Run and output the secret and public keys $\mathsf{sk} = S$, $\mathsf{pk} = (B \| A) \leftarrow \mathsf{KeyGen}(params)$.

- $\mathsf{Enc}^{hy}_{\mathsf{pk}}(m)$: For a message $m \in \{0,1\}^d$, choose $\delta \leftarrow \mathbb{Z}_t^\ell$ and compute

$$\begin{aligned} \mathbf{c}_1 &\leftarrow G(\delta) \oplus m, \\ \mathbf{v} &\leftarrow H(\delta \| \mathbf{c}_1), \\ \mathbf{c}_2 &\leftarrow ((p/t) \cdot \delta + \lfloor (p/q) \cdot B^T \mathbf{v} \rceil, \lfloor (p/q) \cdot A^T \mathbf{v} \rceil) \\ \mathbf{c}_3 &\leftarrow H'(\delta). \end{aligned}$$

  Then, output the ciphertext $\mathbf{c} = (\mathbf{c}_1, \mathbf{c}_2, \mathbf{c}_3) \in \{0,1\}^d \times \mathbb{Z}_p^{n+\ell} \times \mathbb{Z}_t^\ell$.

- $\mathsf{Dec}^{hy}_{\mathsf{sk}}(\mathbf{c})$: Compute $\delta' \leftarrow \mathsf{Dec}_{\mathsf{sk}}(\mathbf{c}_2)$ and $\mathbf{v}' \leftarrow H(\delta' \| \mathbf{c}_1)$. If $\mathbf{c}_2 = \mathsf{Enc}_{\mathsf{pk}}(\delta'; \mathbf{v}')$ and $\mathbf{c}_3 = H'(\delta')$, compute and output $M' \leftarrow G(\delta') \oplus \mathbf{c}_1$. Else, abort and output $\perp$.

Here, $\mathsf{Enc}_{\mathsf{pk}}(\delta'; \mathbf{v}')$ denotes the encryption of δ' with random vector $\mathbf{v}'$, *i.e.*, $\mathsf{Enc}_{\mathsf{pk}}(\delta'; \mathbf{v}') = ((p/t) \cdot \delta' + \lfloor (p/q) \cdot B^T \mathbf{v}' \rceil, \lfloor (p/q) \cdot A^T \mathbf{v}' \rceil)$.

Remark 2. Assuming the hardness of LWE and LWR, Our hybrid public-key encryption scheme $(\mathsf{Setup}^{hy}, \mathsf{KeyGen}^{hy}, \mathsf{Enc}^{hy}, \mathsf{Dec}^{hy})$ is IND-CCA secure by the Theorem 4 in [66] since the original scheme in Section 3 satisfies that the min-entropy of $\mathsf{Enc}_{\mathsf{pk}}(0)$ is bounded by $\omega(\log \lambda)$ for every pk with overwhelming probability.

5.3 Additive Homomorphic Encryption Scheme

Our scheme in Section 3 can be naturally seen as an additive homomorphic encryption supporting the bounded number of additions together with the following addition procedure:

- $\mathsf{Add}(\mathbf{c}_1, \cdots, \mathbf{c}_k)$: Output $\sum_{i=1}^k \mathbf{c}_i$ through componentwise modular p addition.

Corollary 1 (Correctness)**.** *When the inequality*

$$\eta \cdot \alpha p \cdot \sqrt{h_r} + \frac{1}{2} + \xi \cdot h_s \leq \frac{p}{2tk}$$

holds for some positive reals η and ξ such that $\exp(-\pi\eta^2)$ and $(e \cdot (1/2 - \xi))^{h_s}$ are negligible, our LWR*-based additive homomorphic encryption scheme supporting k additions works correctly.*

Proof. This is directly from Theorem 3, by the triangle inequality. □

5.3.1 Parameter Consideration

For our additive homomorphic encryption scheme, we set the message size to be 128 bits, and the maximum number of allowed additions among fresh ciphertexts to be 100. Setting $h_s = h_r = 64$, we achieve our recommended parameter set as follows to resist all known quantum attacks for $\lambda = 128$:

$$m = 728, \quad n = 748, \quad p = 19201, \quad \alpha^{-1} = 37565. \tag{2}$$

Yet there have been many proposals on the additive homomorphic encryption [54, 51, 40], and [26], some of them are too heavy due to the large parameters and the others does not provide quantum security. Hence, our scheme can be considered as a good candidate of quantum-secure additive homomorphic encryption scheme.

6 Implementation

We implemented our encryption schemes: Lizard and its ring variant called Ring-Lizard. We follow the recommended parameters from Table 2 and the condition (1) for the 128-bit security. The subsection 6.1 and 6.2 give comparisons of our post-quantum IND-CPA and IND-CCA schemes with RSA/NTRU and CHK+ [24], respectively.

All the implementations of our schemes in this paper were written in C++, and performed on Macbook Pro with an Intel dual-core i5 running at 2.9 GHz processor without parallelization. The performance of our schemes Lizard and Ring-Lizard in Table 5, 6, and 7 are reported as a mean value across 1000 measurements. The measurements of RSA and NTRU schemes, on a PC with Intel quad-core i5-6600 running at 3.3GHz processor, are taken from the website *ECRYPT Benchmarking of Cryptographic Systems* [1]. The implementation in [24] were performed on Macbook Pro with CPU 2.6GHz Intel Core i5 without parallelization.

Since the algorithms of Lizard and Ring-Lizard are very simple, the code sizes of the encryption and decryption algorithms are greatly small, about 33 lines and 25 lines of C++ code, respectively.

6.1 Post-Quantum 128-bit IND-CPA Security

Like the ElGamal encryption scheme [30], our IND-CPA secure encryptions themselves are useful primitives, and can be easily converted into an IND-CCA secure schemes using the conversion techniques such as Fujisaki-Okamoto conversion. Thus, we implemented Lizard and Ring-Lizard, and compare the results to those of NTRU and RSA. Our schemes would be set with the recommended parameters to achieve 128-bit post-quantum security, and the message sizes of Lizard and Ring-Lizard are 128 bits and 64 bytes, respectively.

We compare our performances with RSA-3072 that is known to achieve *c*lassical 128-bit security [3]. [2] For NTRU, we compare our scheme to the 192-bit set EES593EP1 since the 128-bit set EES439EP17 is attacked with 2^{112} complexity using a quantum computer [31]. The message size of RSA-3072 and NTRU in the implementations is 59 bytes.

We present the sizes of ciphertexts, public keys, and private keys in Table 4. The ciphertext size of Lizard is comparable to that of RSA-4096 and is 63 percents of that of NTRU EES593EP1. The key sizes of Lizard are large among them because of the lattice structure, but we can reduce the key sizes adopting the ring structure: the public key size of Ring-Lizard is comparable to those of NTRU and RSA. The private key size of Ring-Lizard is comparable to that of NTRU and is only 2 percent of that of RSA.

Encryption Scheme	Ciphertext (bytes)	Public Key (bytes)	Private Key (bytes)
(RSA-3072*)	(384)	(384)	(3072)
NTRU	816	820	87
Lizard	**510**	67424	9926
Ring-Lizard	935	**784**	**72**

Table 4: Parameter sizes of standalone PKE schemes in bytes. (*RSA-3072 provides only classical 128-bit security, and is insecure against attacks using a quantum computer.)

In Table 5, the performances of Lizard, Ring-Lizard, NTRU, and RSA are represented by the number of cycles per operation. Lizard has the best record in encryption speed among the four

[2] Note that Shor's algorithm [64] shows that integer factorizing is efficient on a quantum computer so that RSA can be defeated by constructing a large quantum computer.

Encryption Scheme	KeyGen (cycles)	Enc (cycles)	Dec (cycles)
(RSA-3072*)	(614,546,420)	(116,894)	(8,776,864)
NTRU	785,590	80,558	82078
Lizard	37,658,918	**22,928**	18,844
Ring-Lizard	**230,425**	37,366	**15,569**

Table 5: Performance of standalone PKE schemes; the numbers of cycles per each operation. (*RSA-3072 provides only classical 128-bit security, and is insecure against attacks using a quantum computer.)

encryption schemes, more than 3 and 5 times faster than NTRU and RSA, respectively. Note that, for Lizard, it takes only 7 microseconds for an encryption operation and 6 microseconds for a decryption as in Table 6. The key generation of Ring-Lizard only takes 230425 cycles and less than 0.1 ms, which is the fastest result among the four schemes. Decryption of Ring-Lizard also has the best performance, takes only 5 microseconds and is about 5 times faster than NTRU.

Encryption Scheme	KeyGen (ms)	Enc (ms)	Dec (ms)
Lizard	11.637	**0.007**	**0.006**
Ring-Lizard	**0.074**	0.012	0.005

Table 6: The running time of Lizard and Ring-Lizard for each operation in milliseconds.

6.2 Post-Quantum 128-bit IND-CCA Security

In this subsection, we consider Lizard to be plugged in Targhi-Unruh conversion for achieving provable post-quantum IND-CCA security and present the implementation results. Note that the NTRU encryption was proved to achieve IND-CCA security in ROM only, and would be defeated in QROM.

For the Targhi-Unruh CCA conversion in QROM, the message sizes of an IND-CPA secure PKE used in the conversion should be equal to the hash output size of the collision-resistant hash function H', and the message sizes of a resulting hybrid IND-CCA secure PKE should be equal to the output size of the one-way hash function G, respectively. Since the collision-resistant (resp. one-way) hash output size is required to be 3λ (resp. 2λ) to achieve λ-bit post-quantum security, the required message sizes of IND-CPA secure PKE and hybrid IND-CCA secure PKE would be 384 bits and 256 bits when $\lambda = 128$. Thus, for a fixed security parameter $\lambda = 128$, we implemented the IND-CPA secure Lizard of the message size 384 bits, and the IND-CCA secure conversion of Lizard of the message size 256 bits.

We compare our results to those of PKE schemes in [24], say CHK+, in Table 7, which also suggested a CPA-secure PKE scheme and converted it to obtain post-quantum 128-bit IND-CCA security by using the Targhi-Unruh conversion technique. Before conversion, the encryption speed of Lizard is more than 30 times faster than that of the IND-CPA secure CHK+, and is about 26 times faster when converted into IND-CCA secure schemes. The decryption speed of Lizard is more than 4 times faster than that of CHK+, and is about 8 times faster when converted into IND-CCA secure schemes. The ciphertext sizes in both CPA and CCA versions are comparable to those of CHK+.

IND-CPA Encryption	Enc (ms)	Dec (ms)	Cpxt (bytes)
Lizard	**0.010**	**0.025**	**745**
CHK+	0.314	0.106	770
IND-CCA Encryption	Enc (ms)	Dec (ms)	Cpxt (bytes)
CCA-Lizard	**0.012**	**0.036**	**778**
CCA-CHK+	0.313	0.302	804

Table 7: Performance comparison of Lizard and CHK+ [24] for both IND-CPA and IND-CCA versions; CCA-CHK+ is the Targhi-Unruh conversion of CHK+ which is secure in Quantum Random Oracle Model.

Acknowledgement. This work was supported by Samsung Research Funding Center of Samsung Electronics under Project Number SRFC-TB1403-03. The authors would like to thank Andrey Kim, Kyoohyung Han, Junbeom Shin, and Estsoft for valuable discussions.

References

[1] ebacs: Ecrypt benchmarking of cryptographic systems. `http://bench.cr.yp.to/results-encrypt.html`.

[2] Proposed submission requirements and evaluation criteria for the post-quantum cryptography standardization process. `http://csrc.nist.gov/groups/ST/post-quantum-crypto/documents/call-for-proposals-draft-aug-2016.pdf`.

[3] Recommendation for key management. `http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-57pt1r4.pdf`.

[4] M. Albrecht, S. Bai, and L. Ducas. *A Subfield Lattice Attack on Overstretched NTRU Assumptions*, pages 153–178. Springer Berlin Heidelberg, Berlin, Heidelberg, 2016.

[5] M. R. Albrecht, C. Cid, J.-C. Faugere, R. Fitzpatrick, and L. Perret. On the complexity of the bkw algorithm on lwe. *Designs, Codes and Cryptography*, 74(2):325–354, 2015.

[6] M. R. Albrecht, J.-C. Faugere, R. Fitzpatrick, and L. Perret. Lazy modulus switching for the bkw algorithm on lwe. In *International Workshop on Public Key Cryptography*, pages 429–445. Springer, 2014.

[7] M. R. Albrecht, R. Fitzpatrick, and F. Göpfert. On the efficacy of solving lwe by reduction to unique-svp. In *International Conference on Information Security and Cryptology*, pages 293–310. Springer, 2013.

[8] M. R. Albrecht, R. Player, and S. Scott. On the concrete hardness of learning with errors. *Journal of Mathematical Cryptology*, 9(3):169–203, 2015.

[9] E. Alkim, L. Ducas, T. Pöppelmann, and P. Schwabe. Post-quantum key exchange—a new hope. In *25th USENIX Security Symposium (USENIX Security 16)*, pages 327–343, Austin, TX, Aug. 2016. USENIX Association.

[10] J. Alwen, S. Krenn, K. Pietrzak, and D. Wichs. Learning with rounding, revisited. In *Advances in Cryptology–CRYPTO 2013*, pages 57–74. Springer, 2013.

[11] S. Arora and R. Ge. New algorithms for learning in presence of errors. In *International Colloquium on Automata, Languages, and Programming*, pages 403–415. Springer, 2011.

[12] S. Bai and S. D. Galbraith. Lattice decoding attacks on binary lwe. In *Australasian Conference on Information Security and Privacy*, pages 322–337. Springer, 2014.

[13] W. Banaszczyk. New bounds in some transference theorems in the geometry of numbers. *Mathematische Annalen*, 296(1):625–635, 1993.

[14] W. Banaszczyk. Inequalities for convex bodies and polar reciprocal lattices in r^n. *Discrete & Computational Geometry*, 13(2):217–231, 1995.

[15] A. Banerjee, C. Peikert, and A. Rosen. Pseudorandom functions and lattices. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 719–737. Springer, 2012.

[16] A. Becker, L. Ducas, N. Gama, and T. Laarhoven. New directions in nearest neighbor searching with applications to lattice sieving. In *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 10–24. SIAM, 2016.

[17] D. J. Bernstein. A subfield-logarithm attack against ideal lattices, February 2014. `https://blog.cr.yp.to/20140213-ideal.html`.

[18] A. Bogdanov, S. Guo, D. Masny, S. Richelson, and A. Rosen. On the hardness of learning with rounding over small modulus. In *Theory of Cryptography Conference*, pages 209–224. Springer, 2016.

[19] J. Bos, C. Costello, L. Ducas, I. Mironov, M. Naehrig, V. Nikolaenko, A. Raghunathan, and D. Stebila. Frodo: Take off the ring! practical, quantum-secure key exchange from lwe. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, CCS '16, pages 1006–1018, New York, NY, USA, 2016. ACM.

[20] J. W. Bos, C. Costello, M. Naehrig, and D. Stebila. Post-quantum key exchange for the tls protocol from the ring learning with errors problem. In *2015 IEEE Symposium on Security and Privacy*, pages 553–570. IEEE, 2015.

[21] Z. Brakerski, C. Gentry, and S. Halevi. Packed ciphertexts in lwe-based homomorphic encryption. In *Public-Key Cryptography–PKC 2013*, pages 1–13. Springer, 2013.

[22] Z. Brakerski, A. Langlois, C. Peikert, O. Regev, and D. Stehlé. Classical hardness of learning with errors. In *Proceedings of the forty-fifth annual ACM symposium on Theory of computing*, pages 575–584. ACM, 2013.

[23] Y. Chen and P. Q. Nguyen. Bkz 2.0: Better lattice security estimates. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 1–20. Springer, 2011.

[24] J. H. Cheon, K. Han, J. Kim, C. Lee, and Y. Son. Practical post-quantum public key cryptosystem based on lwe. In *the 19th Annual international Conference on Information Security and Cryptology*, 2016. Available at `https://eprint.iacr.org`.

[25] J. H. Cheon, J. Jeong, and C. Lee. An algorithm for ntru problems and cryptanalysis of the ggh multilinear map without a low-level encoding of zero. *LMS Journal of Computation and Mathematics*, 19(A):255266, Jan 2016.

[26] J. H. Cheon, H. T. Lee, and J. H. Seo. A new additive homomorphic encryption based on the co-acd problem. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, pages 287–298. ACM, 2014.

[27] J. Ding, X. Xie, and X. Lin. A simple provably secure key exchange scheme based on the learning with errors problem. *IACR Cryptology ePrint Archive*, 2012:688, 2012.

[28] A. Duc, F. Tramèr, and S. Vaudenay. Better algorithms for lwe and lwr. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 173–202. Springer, 2015.

[29] L. Ducas and D. Micciancio. Fhew: Bootstrapping homomorphic encryption in less than a second. In *Advances in Cryptology–EUROCRYPT 2015*, pages 617–640. Springer, 2015.

[30] T. ElGamal. A public key cryptosystem and a signature scheme based on discrete logarithms. In *Workshop on the Theory and Application of Cryptographic Techniques*, pages 10–18. Springer, 1984.

[31] S. Fluhrer. Quantum cryptanalysis of ntru. Cryptology ePrint Archive, Report 2015/676, 2015. `http://eprint.iacr.org/2015/676`.

[32] J. Folláth. Gaussian sampling in lattice based cryptography. *Tatra Mt. Math. Publ*, 60:1–23, 2014.

[33] P.-A. Fouque, M. S. Lee, T. Lepoint, and M. Tibouchi. Cryptanalysis of the co-acd assumption. In *Annual Cryptology Conference*, pages 561–580. Springer, 2015.

[34] E. Fujisaki and T. Okamoto. Secure integration of asymmetric and symmetric encryption schemes. *Journal of cryptology*, 26(1):80–101, 2013.

[35] N. Gama and P. Q. Nguyen. Predicting lattice reduction. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 31–51. Springer, 2008.

[36] N. Gama, P. Q. Nguyen, and O. Regev. Lattice enumeration using extreme pruning. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 257–278. Springer, 2010.

[37] C. Gentry, C. Peikert, and V. Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In *Proceedings of the fortieth annual ACM symposium on Theory of computing*, pages 197–206. ACM, 2008.

[38] G. Hanrot, X. Pujol, and D. Stehl. Terminating bkz. Cryptology ePrint Archive, Report 2011/198, 2011. `http://eprint.iacr.org/2011/198`.

[39] J. Hoffstein, J. Pipher, and J. H. Silverman. Ntru: A ring-based public key cryptosystem. In *International Algorithmic Number Theory Symposium*, pages 267–288. Springer, 1998.

[40] M. Joye and B. Libert. Efficient cryptosystems from 2 k-th power residue symbols. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 76–92. Springer, 2013.

[41] A. H. Karbasi and R. E. Atani. Iltru: An ntru-like public key cryptosystem over ideal lattices. In *The 7th International IEEE Symposium on Telecommunications (IST2014), Tehran, Iran*, 2014.

[42] P. Kirchner and P.-A. Fouque. An improved bkw algorithm for lwe with applications to cryptography and lattices. In *Annual Cryptology Conference*, pages 43–62. Springer, 2015.

[43] T. Laarhoven. *Search problems in cryptography*. PhD thesis, PhD thesis, Eindhoven University of Technology, 2015. `http://www.thijs.com/docs/phd-final.pdf`, 2015.

[44] T. Laarhoven. Sieving for shortest vectors in lattices using angular locality-sensitive hashing. In *Annual Cryptology Conference*, pages 3–22. Springer, 2015.

[45] T. Laarhoven, M. Mosca, and J. Van De Pol. Finding shortest lattice vectors faster using quantum search. *Designs, Codes and Cryptography*, 77(2-3):375–400, 2015.

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

[47] R. Lindner and C. Peikert. Better key sizes (and attacks) for lwe-based encryption. In *Cryptographers Track at the RSA Conference*, pages 319–339. Springer, 2011.

[48] M. Liu, X. Wang, G. Xu, and X. Zheng. Shortest lattice vectors in the presence of gaps. *IACR Cryptology ePrint Archive*, 2011:139, 2011.

[49] V. Lyubashevsky, C. Peikert, and O. Regev. On ideal lattices and learning with errors over rings. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 1–23. Springer, 2010.

[50] D. Micciancio and O. Regev. Lattice-based cryptography. In *Post-quantum cryptography*, pages 147–191. Springer, 2009.

[51] M. Naehrig, K. Lauter, and V. Vaikuntanathan. Can homomorphic encryption be practical? In *Proceedings of the 3rd ACM workshop on Cloud computing security workshop*, pages 113–124. ACM, 2011.

[52] P. Q. Nguên and D. Stehlé. Floating-point lll revisited. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 215–233. Springer, 2005.

[53] T. Okamoto and S. Uchiyama. A new public-key cryptosystem as secure as factoring. In *International Conference on the Theory and Applications of Cryptographic Techniques*, pages 308–318. Springer, 1998.

[54] P. Paillier. Public-key cryptosystems based on composite degree residuosity classes. In *International Conference on the Theory and Applications of Cryptographic Techniques*, pages 223–238. Springer, 1999.

[55] C. Peikert. Public-key cryptosystems from the worst-case shortest vector problem. In *Proceedings of the forty-first annual ACM symposium on Theory of computing*, pages 333–342. ACM, 2009.

[56] C. Peikert. An efficient and parallel gaussian sampler for lattices. In *Annual Cryptology Conference*, pages 80–97. Springer, 2010.

[57] C. Peikert. Lattice cryptography for the internet. In *International Workshop on Post-Quantum Cryptography*, pages 197–219. Springer, 2014.

[58] C. Peikert, V. Vaikuntanathan, and B. Waters. A framework for efficient and composable oblivious transfer. In *Annual International Cryptology Conference*, pages 554–571. Springer, 2008.

[59] C. Peikert and B. Waters. Lossy trapdoor functions and their applications. In *Proceedings of the fortieth annual ACM symposium on Theory of computing*, pages 187–196. ACM, 2008.

[60] O. Regev. On lattices, learning with errors, random linear codes, and cryptography. In *Proceedings of the Thirty-seventh Annual ACM Symposium on Theory of Computing*, STOC '05, pages 84–93, New York, NY, USA, 2005. ACM.

[61] R. L. Rivest, A. Shamir, and L. Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2):120–126, 1978.

[62] M. Rckert and M. Schneider. Estimating the security of lattice-based cryptosystems. Cryptology ePrint Archive, Report 2010/137, 2010. `http://eprint.iacr.org/2010/137`.

[63] C.-P. Schnorr and M. Euchner. Lattice basis reduction: improved practical algorithms and solving subset sum problems. *Mathematical programming*, 66(1-3):181–199, 1994.

[64] P. W. Shor. Algorithms for quantum computation: Discrete logarithms and factoring. In *Foundations of Computer Science, 1994 Proceedings., 35th Annual Symposium on*, pages 124–134. IEEE, 1994.

[65] D. Stehlé and R. Steinfeld. Making ntru as secure as worst-case problems over ideal lattices. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 27–47. Springer, 2011.

[66] E. E. Targhi and D. Unruh. Quantum security of the fujisaki-okamoto and oaep transforms. Cryptology ePrint Archive, Report 2015/1210, 2015. `http://eprint.iacr.org/2015/1210`.