A Practical Post-Quantum Public-Key Cryptosystem Based on spLWE

Jung Hee Cheon, Kyoo Hyung Han, Jinsu Kim, Changmin Lee, and Yongha Son

Department of Mathematical Sciences, Seoul National University,
1 Gwanak-ro, Gwanak-gu, Seoul 151-747, Korea
{jhcheon,satanigh,nemokjs1,cocomi11,emsskk}@snu.ac.kr

Abstract. The Learning with Errors (LWE) problem has been widely used as a hardness assumption to construct public-key primitives. In this paper, we propose an efficient instantiation of a PKE scheme based on LWE with a sparse secret, named as spLWE. We first construct an IND-CPA PKE and convert it to an IND-CCA scheme in the quantum random oracle model by applying a modified Fujisaki-Okamoto conversion of Unruh. In order to guarantee the security of our base problem suggested in this paper, we provide a polynomial time reduction from LWE with a uniformly chosen secret to spLWE. We modify the previous attacks for LWE to exploit the sparsity of a secret key and derive more suitable parameters. We can finally estimate performance of our scheme supporting 256-bit messages: our implementation shows that our IND-CCA scheme takes 313 μ seconds and 302 μ seconds respectively for encryption and decryption with the parameters that have 128-quantum bit security.

Keywords: practical, post-quantum, IND-CCA, PKE, sparse secret, LWE, quantum random oracle model

1 Introduction

With advances in quantum computing, many people in various fields are working on making their information security systems resistant to quantum computing. The National Security Agency (NSA) has announced a plan to change its Suite B guidance [42], and the National Institute of Standards and Technology (NIST) is now beginning to prepare for the transition into quantum-resistant cryptography [41]. There have been also substantial support for post-quantum cryptography from national funding agencies including the PQCRYPTO projects [18] in Europe.

In that sense, lattice-based cryptography is a promising field to conduct practical quantum-resistant research. This is due to the seminal work of Ajtai [1] who proved a reduction from the worst-case to the average-case for some lattice problems. This means that certain problems are hard on average, as long as the related lattice problems are hard in all cases. This enables provably secure constructions unless all instances of related lattice problems are easy to solve. Another remarkable work in lattice-based cryptography is the introduction of Learning with Errors problem by Regev in [47]. This work shows that there exists a quantum reduction from some worst-case lattice problems (the shortest independent vectors problem, the shortest vector problem with a gap) to LWE. With a strong security guarantee, LWE makes versatile cryptographic constructions possible including fully homomorphic encryption, multi-linear map, etcetera. For more details, we refer to the recent survey [44].

In order to increase efficiency on lattice-based cryptographic schemes, ring structured problems such as Learning with Errors over the ring (RLWE) and NTRU [37], [32] have received much attentions. A major advantage of using a ring structure is that one can get a relatively smaller key size and faster speed. For that reason, a lot of works about cryptographic schemes with practical implementation have been proposed in RLWE and NTRU settings: public-key encryptions ([19], [49], [36]), signatures ([23], [22], [27]), key-exchanges ([12], [5], [51]). However, additional ring structures may give some advantages to attackers. As an example, some analyses using the ring structure have been proposed recently. In particular, some NTRU-based fully homomorphic encryptions proved valueless [39], [16] and some parameters of RLWE are confirmed

to be weak [30, 31]. Hence, there are growing concerns about the security gap for ring-structured cryptosystems.

On the other hand, it is reported that LWE-based signatures [22], [27], [17] achieve good performance without the use of RLWE, and studies of practicality of LWE-based key exchange protocols have been recently started in [11]. However, less attention has been paid to practical instantiations of LWE-based cryptosystems. In this sense, proposing of a practical LWE-based public-key cryptosystem and evaluation its performance would be an interesting topic in lattice-based cryptography. However, construction of public-key cryptosystem, which satisfies both high levels of security and efficiency, is a very non-trivial and hard task. it requires the right balance between security and efficiency to constitute a complete proposal, which considers the possibility of practical use.

Our first contribution is that we are suggesting a practical post-quantum public-key cryptosystem based on spLWE that is a variant of LWE with a sparse secret vector: Based on spLWE, we propose an IND-CPA PKE inspired from [43] and convert it into an IND-CCA version in the quantum random oracle model by applying the modified Fujisaki-Okamoto conversion of Unruh [52]. We identify its practicality from our implementation on a PC. The implementation result shows that our proposal enables relatively fast encryption and decryption that take about hundreds of microseconds.

Our second contribution is that we are providing the analysis for spLWE: We proved that spLWE can be reduced from LWE, which means that the hardness of spLWE can also be based on the worst-case lattice problems. We also extend all known LWE attacks to investigate concrete hardness of spLWE. As a result, we could derive concrete parameters based on those attacks. We would like to note that we exclude the parameters which have provable security from our reduction under the consideration about practicality. Our reduction serves to guarantee the hardness of spLWE, but is not tight enough to be useful in setting concrete parameters for our scheme.

1.1 Results and Techniques

We have suggested concrete parameters for both classical and quantum security, implementation results of our scheme and a comparison table with the previous LWE-based PKE [48] and RLWE-based PKE [37] in section 5.2. In 128-quantum bit security, the IND-CPA version of our encryption took about $314\mu s$ and the IND-CCA version of our encryption takes $313\mu s$ for 256-bit messages on Macbook Pro with CPU 2.6GHz Intel Core i5 without parallelization.

To achieve this result, we chose a variant of LWE with a sparse secret. In most LWE-based encryptions, it is necessary to compute $\boldsymbol{u}^T\mathbf{A}$ or $\boldsymbol{u}^T\mathbf{A}+\mathbf{e}$ for $\boldsymbol{u}\in\mathbb{Z}_q^m$, $\boldsymbol{e}\in\mathbb{Z}_q^n$ and $\mathbf{A}\in\mathbb{Z}_q^{m\times n}$. When the vector $\boldsymbol{u}$ has low hamming weight, real computation cost is similar to that of θ-length vector. Moreover, the cost can be reduced further when restricting the non-zero components by power of two.

Unfortunately, the use of sparse secret has one drawback. It requires relatively larger dimension than that of LWE to maintain security. This is a significant factor for the performance of LWE-based schemes. A important question then arises: How large dimension is needed to maintain security? We can observe that the problem of increase in dimension can be relieved by using a small modulus q. Since the security of LWE is proportional to the size of dimension and error rate, smaller modulus leads to larger error rate. We can choose a relatively small modulus q in spLWE case from Theorem 3: The decapsulation error completely depends on inner product of secret and error vectors. We were able to identify the effect concretely from the attacks in Section 4.2 and Appendix for spLWE by extending all known attacks of LWE, which can be improved by exploiting the sparsity of secret: The dimension of spLWE still remains below 520. We also provide a reduction from LWE to spLWE under certain parameters in Section 4.1. This implies that the hardness of spLWE can be also based on the worst-case lattice problems. It can be done by generalizing the reduction of [13] from LWE to the binary LWE.

Finally, we can prove IND-CCA security of our scheme in the random oracle model. More specifically, we applied the result of the recent paper [52] to construct our PKE, which gives a slight modification of the Fujisaki-Okamoto transform in a quantum adversary setting. The modification only needs simple operations such as hashing and XOR to convert a IND-CPA PKE into IND-CCA one, and hence converting overhead is expected to be small.

1.2 Related Works

Practical instantiations and implementation results about post-quantum primitives in lattice-based cryptography have been reported mostly in the RLWE case rather than in the LWE one (e.g. [36], [49], [19], etc). In particular, Peikert [43] presented efficient and practical lattice-based protocols for key transport and encryption on RLWE that are suitable for Internet standards and other open protocols. We also use the idea of KEM-based construction for improved efficiency. In our spLWE-based construction, the ciphertext size of an IND-CPA encryption scheme for ℓ-bit message is $(n \log q + 2\ell)$-bit. This is smaller than that of the known LWE-based PKEs [46, 48] which have $(n \log q + \ell \log q)$-bit ciphertext size.

In the case of LWE-based PKEs [46], [25], [45], [48], [40], there are a few works on efficiency improvement. Galbraith [24] proposed variants of LWE where the entries of the random matrix are chosen to be smaller than a modulus q or binary to reduce the size of a public-key. However, there was no complete proposal which inclues attacks and parameters for practical usage. Bai et al. [7] considered LWE with binary secret to reduce the size of their signature. However, the effect on parameter and speed of their scheme was not fully investigated.

2 Preliminaries

Notations. In this paper, we use upper-case bold letters to denote matrices, and lower-case bold letters for column vectors. For a distribution $\mathcal{D}$, $a \leftarrow \mathcal{D}$ denotes choosing an element according to the distribution of $\mathcal{D}$ and $\boldsymbol{a} \leftarrow \mathcal{D}^m$ means that each component of $\boldsymbol{a}$ is sampled independently from $\mathcal{D}$. In particular, if the x_i's are independent and each x_i follows a Bernoulli distribution with ρ for a vector $\boldsymbol{x} = (x_1, \ldots, x_n)$, then we say that a vector $\boldsymbol{x}$ follows $\mathrm{Ber}(n, \rho)$. For a given set $\mathcal{A}, \mathcal{U}(\mathcal{A})$ means a uniform distribution on the set $\mathcal{A}$ and $a \leftarrow \mathcal{A}$ denotes choosing an element according to the uniform distribution on $\mathcal{A}$. We denote by $\mathbb{Z}_q = \mathbb{Z}/q\mathbb{Z} = \{0, 1 \cdots, q-1\}$ and $\mathbb{T} = \mathbb{R}/\mathbb{Z}$ the additive group of real numbers modulo 1, and $\mathbb{T}_q$ the a subgroup of $\mathbb{T}$ having order q, consisting of $\{0, \frac{1}{q}, \cdots, \frac{q-1}{q}\}$. The $\langle\ ,\ \rangle$ means the inner product of two vectors and $[\boldsymbol{x}]_i$ means the its i-th component. A function $f(\lambda)$ is called *negligible* if $f(\lambda) = o(\lambda^{-c})$ for any $c > 0$, i.e., f decrease faster than any inverse polynomial.

2.1 Security Definitions

Definition 1 (γ-spread, [52]). *A PKE is γ-spread if for every public-key generated by* Keygen *algorithm and every message $\boldsymbol{m}$,*

$$\max_{\boldsymbol{y}} Pr[\boldsymbol{y} \leftarrow \mathsf{Enc}_{pk}(\boldsymbol{m})] \leq \frac{1}{2^{\gamma}}.$$

In particular, we say that a PKE is well-spread if $\gamma = \omega(\log(\lambda))$.

Definition 2 (One-way secure). *A PKE is One-Way secure if no (quantum) polynomial time algorithm (adversary) $\mathcal{A}$ can find a message $\boldsymbol{m}$ from $\mathsf{Enc}_{pk}(\boldsymbol{m})$, given only public-key except with probability at most negl(λ).*

2.2 Key Encapsulation Mechanism

A *key encapsulation mechanism* (in short, KEM) is a key exchange algorithm to transmit an ephemeral key to a receiver with the receiver's public key. It differs from encryption scheme where a sender can choose a message. The sender cannot intend to make a specific ephemeral key. A KEM with ciphertext space $\mathcal{C}$ and key space $\mathcal{K}$ consists of polynomial time algorithms Setup, Keygen, Encap(may be randomized), Decap(should be deterministic).

- Params outputs a public parameters.
- Keygen outputs a public encapsulation key pk and secret decapsulation key sk.
- Encap takes an encapsulation key pk and outputs a ciphertext $c \in \mathcal{C}$ and a key $k \in \mathcal{K}$.
- Decap takes a decapsulation key sk and a ciphertext c, and outputs some $k \in \mathcal{K} \cup \{\perp\}$, where $\perp$ denotes decapsulation failure.

2.3 Lattice and Lattice Reduction Algorithm

A *lattice* $L \subseteq \mathbb{R}^m$ is a set of integer linear combinations of a $\{\boldsymbol{b}_1, \cdots, \boldsymbol{b}_n\}$ which is a subset of independent column vectors in $\mathbb{R}^m$,

$$L = \{\sum_{i=1}^{n} a_i \boldsymbol{b}_i : a_i \in \mathbb{Z}\}$$

The set of vectors $\{\mathbf{b}_1, \ldots, \mathbf{b}_n\}$, and its matrix form $\mathbf{B}$ are called a basis, and basis matrix of L respectively. Two bases matrices $\mathbf{B}_1$ and $\mathbf{B}_2$ describe the same lattice, if and only if $\mathbf{B}_2 = \mathbf{B}_1\mathbf{U}$, where $\mathbf{U}$ is a unimodular matrix, i.e. $\det(U) = \pm 1$, $U \in \mathbb{Z}^{m \times m}$. Dimension of a lattice is defined as cardinality of a basis, i.e. $n = \dim(L)$. If $n = m$, we call lattice L to a full rank lattice. A sublattice is a subset $L' \subset L$ which is also a lattice. We define determinant (volume) of L by

$$\det(L) := \sqrt{\det(\boldsymbol{B}^T \boldsymbol{B})}$$

A length of the shortest vector in a lattice $L(\mathbf{B})$ is denoted by $\lambda_1(L(\mathbf{B}))$. More generally, the *i-th successive minima* $\lambda_i(L)$ is defined as the smallest radius r such that $\dim(\mathrm{span}(L \cap B(r))) \geq i$ where $B(r)$ is a n dimensional ball with radius r. There exist several bounds and estimations for the length of the shortest vector in a lattice.

- Minkowski's first theorem: $\lambda_1(L(\mathbf{B})) \leq \sqrt{n}(\det L(\mathbf{B}))^{1/n}$
- Gaussian heuristic: $\lambda_1(L(\mathbf{B})) \approx \sqrt{\frac{n}{2\pi e}}\det(L(\mathbf{B}))^{1/n}$ for random lattice L.

The *dual lattice* of L, denoted $\bar{L}$, is defined to be $\bar{L} = \{\boldsymbol{x} \in \mathbb{R}^n : \forall \boldsymbol{v} \in L, \langle \boldsymbol{x}, \boldsymbol{v} \rangle \in \mathbb{Z}\}$. We recall the Gram-Schmidt orthogonalization that is closely related with lattice basis reduction. The Gram-Schmidt algorithm computes orthogonal vectors $\{\mathbf{b}_1^*, \ldots, \mathbf{b}_m^*\}$ iteratively as follows:

$$\mathbf{b}_i^* = \mathbf{b}_i - \sum_{j<i} \mu_{i,j}\mathbf{b}_j^* \text{ where } \mu_{i,j} = \frac{\mathbf{b}_i \cdot \mathbf{b}_j^*}{\mathbf{b}_j^* \cdot \mathbf{b}_j^*}.$$

The goal of lattice (basis) reduction is to find a good basis for a given lattice. A basis is considered good, when the basis vectors are almost orthogonal and correspond approximately to the successive minima of the lattice. Performance of lattice reduction algorithms is evaluated by the *root Hermite factor* δ_0 defined by

$$\delta_0 = (||\boldsymbol{v}||/\det(L)^{1/n})^{1/n}$$

where $\boldsymbol{v}$ is the shortest vector of the reduced output basis.

2.4 Discrete Gaussian Distribution

For given $s > 0$, a *discrete Gaussian distribution* over a lattice L is defined as $D_{L,s}(x) = \rho_s(x)/\rho_s(L)$ for any $x \in L$, where

$$\rho_s(x) = \exp(-\pi\|x\|^2/s^2) \text{ and } \rho_s(L) := \sum_{x\in L} \rho_s(x).$$

We note that the standard deviation is $\sigma = s/\sqrt{2\pi}$. When $L = \mathbb{Z}$, we omit the subscript L. For a lattice L, the *smoothing parameter* $\eta_\epsilon(L)$ is defined by the smallest real number $s' > 0$ such that $\rho_{1/s'}(\bar{L} \setminus \{\mathbf{0}\}) \leq \epsilon$. We collect some useful lemmas related to a discrete Gaussian distribution and the smoothing parameter.

Lemma 1 ([9], Lemma 2.4). *For any real $s > 0$ and $T > 0$, and any vector $\boldsymbol{x} \in \mathbb{R}^n$, we have*

$$Pr[|\langle \boldsymbol{x}, D^n_{\mathbb{Z},s}\rangle| \geq T \cdot s\|\boldsymbol{x}\|] < 2\exp(-\pi \cdot T^2).$$

Lemma 2 ([46], Corollary 3.10). *Let L be an n-dimensional lattice, let $\boldsymbol{u}, \boldsymbol{z} \in \mathbb{R}^n$ be arbitrary vectors, and let r, α be positive real numbers. Assume that $(1/r^2 + (\|\boldsymbol{z}/\alpha\|)^2)^{-1/2} \geq \eta_\epsilon(\Lambda)$ for some $\epsilon < 1/2$. Then the distribution of $\langle \boldsymbol{z}, \boldsymbol{v}\rangle + e$ where $\boldsymbol{v} \leftarrow D_{L+\boldsymbol{u},r}$ and $e \leftarrow D_\alpha$ is within statistical distance 4ϵ of D_β for $\beta = \sqrt{(r\|\boldsymbol{z}\|)^2 + \alpha^2}$.*

Lemma 3 ([25], Lemma 3.1). *For any $\epsilon > 0$ and an n-dimensional lattice Λ with basis matrix $\boldsymbol{B}$, the smoothing parameter $\eta_\epsilon(\Lambda) \leq \|\tilde{\boldsymbol{B}}\| \ln(2n(1+1/\epsilon))/\pi$ where $\|\tilde{\boldsymbol{B}}\|$ denotes the length of the longest column vector of $\tilde{\boldsymbol{B}}$ which is the Gram-Schmidt orthogonalization of $\boldsymbol{B}$.*

2.5 Learning with Errors

For integers $n, q \geq 1$, a vector $\boldsymbol{s} \in \mathbb{Z}_q^n$, and a distribution ϕ on $\mathbb{R}$, let $A_{q,\boldsymbol{s},\phi}$ be the distribution of the pairs $(\boldsymbol{a}, b = \langle \boldsymbol{a}, \boldsymbol{s}\rangle + e) \in \mathbb{T}_q^n \times \mathbb{T}$, where $\boldsymbol{a} \leftarrow \mathbb{T}_q^n$ and $e \leftarrow \phi$.

Definition 3 (Learning with Errors (LWE)). *For integers $n, q \geq 1$, an error distribution ϕ over R, and a distribution $\mathcal{D}$ over $\mathbb{Z}_q^n$, $\mathsf{LWE}_{n,q,\phi}(\mathcal{D})$, is to distinguish (given arbitrarily many independent samples) the uniform distribution over $\mathbb{T}_q^n \times \mathbb{T}$ from $A_{q,\boldsymbol{s},\phi}$ with a fixed sample $\boldsymbol{s} \leftarrow \mathcal{D}$.*

We note that a search variant of LWE is the problem of recovering $\boldsymbol{s}$ from $(\mathbf{a}, b) = (\mathbf{a}, \langle \mathbf{a}, \mathbf{s}\rangle + e) \in \mathbb{T}_q^n \times \mathbb{T}$ sampled according to $A_{q,\boldsymbol{s},\phi}$, and these are also equivalently defined on $\mathbb{Z}_q^n \times \mathbb{Z}_q$ rather than $\mathbb{T}_q^n \times \mathbb{T}$ for discrete (Gaussian) error distributions over $\mathbb{Z}_q$. Let $\mathsf{LWE}_{n,m,q,\phi}(\mathcal{D})$ denotes the case when the number of samples are bounded by $m \in \mathbb{N}$. We simply denote $\mathsf{LWE}_{n,q,\phi}$ when the secret distribution $\mathcal{D}$ is $\mathcal{U}(\mathbb{Z}_q^n)$. In many cases, ϕ is a (discrete) Gaussian distribution so we simply denote by $\mathsf{LWE}_{n,m,q,s}$ instead of $\mathsf{LWE}_{n,m,q,\phi}$. We denote binLWE by the LWE problem whose secret vector is sampled from uniform distribution over $\{0,1\}^n$. For a set $X_{n,\rho,\theta}$ which consists of the vectors $\boldsymbol{s} \in \mathbb{Z}^n$ whose nonzero components are in $\{\pm 1, \pm 2, \pm 4, \cdots, \pm\rho\}$, and the number of nonzero components is θ, we write $\mathsf{spLWE}_{n,m,q,s,\rho,\theta}$ as the problem $\mathsf{LWE}_{n,m,q,s}(\mathcal{U}(X_{n,\rho,\theta}))$. We also consider a variant of LWE, $\mathsf{LWE}_{n,q,\leq\alpha}$, in which the amount of noise is some unknown $\beta \leq \alpha$ as in [13]. Similarly, $\mathsf{spLWE}_{n,q,\leq\alpha,\rho,\theta}$ can be defined by the same way.

The following lemma will be used to derive some parameters from the modified attacks in section 4 and appendix.

Lemma 4 ([48]). *Given $\mathsf{LWE}_{n,m,q,s}$ samples and a vector $\boldsymbol{v}$ of length $\|\boldsymbol{v}\|$ in the lattice $L = \{\boldsymbol{w} \in \mathbb{Z}_q^m : \boldsymbol{w}^T\boldsymbol{A} \equiv 0 \mod q\}$, the advantage of distinguishing $\langle \boldsymbol{v}, \boldsymbol{e}\rangle$ from uniform random is close to $\exp(-\pi(\|\boldsymbol{v}\|s/q)^2)$.*

We give some variants of LWE and some notion, which were introduced in [13] to show the reduction between binLWE and LWE.

Definition 4 ("first-is-errorless" LWE). *For integers $n, q \geq 1$ and an error distribution ϕ over $\mathbb{R}$, the "first-is-errorless" variant of the LWE problem is to distinguish between the following two scenarios. In the first, the first sample is uniform over $\mathbb{T}_q^n \times \mathbb{T}_q$ and the rest are uniform over $\mathbb{T}_q^n \times \mathbb{T}$. In the second, there is an unknown uniformly distributed $\boldsymbol{s} \in \{0, \ldots, q-1\}^n$, the first sample we get is from $A_{q,\boldsymbol{s},\{0\}}$ (where $\{0\}$ denotes the distribution that is deterministically zero) and the rest are from $A_{q,\boldsymbol{s},\phi}$.*

Definition 5 (extLWE problem). *For integers $n, m, q, t \geq 1$, a set $X \subseteq \mathbb{Z}^m$, and a distribution χ over $\frac{1}{q}\mathbb{Z}^m$, the extLWE$_{n,m,q,\chi,X}$ is as follows. The algorithm gets to choose $\boldsymbol{x} \in X$ and then receives a tuple $(\boldsymbol{A}, (\boldsymbol{b}_i)_{i\in[t]}, (\langle \boldsymbol{e}_i, \boldsymbol{x}\rangle)_{i\in[t]}) \in \mathbb{T}_q^{n\times m} \times (\mathbb{T}_q^m)^t \times (\frac{1}{q}\mathbb{Z})^t$. Its goal is to distinguish between the following two cases. In the first, $A \in \mathbb{T}_q^{n\times m}$ is chosen uniformly, $\boldsymbol{e}_i \in \frac{1}{q}\mathbb{Z}^m$ are chosen from χ, and $\boldsymbol{b}_i = \boldsymbol{A}^T\boldsymbol{s}_i + \boldsymbol{e}_i \mod 1$ where $\boldsymbol{s}_i \in \{0, \ldots, q-1\}^n$ are chosen uniformly. The second case is identical, except that the $\boldsymbol{b}_i$ are chosen uniformly in $\mathbb{T}_q^m$ independently of everything else.*

Definition 6 (Quality of a set). *A set $X \subset \mathbb{Z}^m$ is said of quality ξ if given any $\boldsymbol{x} \in X$, we can efficiently find a unimodular matrix $U \in \mathbb{Z}^{m\times m}$ such that if $U' \in \mathbb{Z}^{m\times(m-1)}$ is the matrix obtained from U by removing its leftmost column then all of the columns of U' are orthogonal to $\boldsymbol{z}$ and its largest singular value is at most ξ. It denoted by $Qual(X)$.*

We give a lemma to show a reduction to spLWE from the standard LWE in section 4.1.

Lemma 5. *The quality of a set $X \subseteq \{0, \pm 1, \pm 2, \ldots, \pm\rho\}^m$, $\rho = 2^l$ is bounded by $1 + \sqrt{\rho}$.*

Proof. Let $\mathbf{x} \in X$ and without loss of generality, we assume leftmost k components of $\mathbf{x}$ are nonzero, remainings are zero, and $|[\mathbf{x}]_i| \leq |[\mathbf{x}]_{i+1}|$ for nonzero components after reordering. We have $[\mathbf{x}]_{i+1} = \pm 2^{t_i}[\mathbf{x}]_i$ for some $t_i \leq l$. Now consider the upper bidiagonal matrix $\mathbf{U}$ whose diagonal is all 1 and whose diagonal above the main diagonal is $\mathbf{y} \in \mathbb{Z}^{m-1}$ such that $[\mathbf{x}]_{i+1} - [\mathbf{y}]_j[\mathbf{x}]_i = 0$ for $1 \leq j \leq k-1$, and rightmost $(m-k)$ components of $\mathbf{y}$ are 0. Since $[\mathbf{x}]_{i+1} = \pm 2^{t_i}[\mathbf{x}]_i$, it follows that $[\mathbf{y}]_j$ is 2^{t_j} or -2^{t_j}. Then $\mathbf{U}$ is clearly unimodular ($\det(\mathbf{U}) = 1$) and all the columns except the first one are orthogonal to $\mathbf{x}$. Moreover, by the triangle inequality, we can bound the norm (the largest singular value) of $\mathbf{U}$ by the sum of that of the diagonal 1 matrix and the off-diagonal matrix of which clearly have norm at most $\sqrt{\rho}$. □

3 Our spLWE-Based PKE

In this section, we introduce a public key encryption scheme whose security is based on spLWE, whose ciphertext size is smaller than those of the previous works [46, 48]. We use a noisy subset sum in our encryption algorithm which is proposed in the previous LWE-based encryption scheme [48], but our message encoding is differents: we first construct a key encapsulation mechanism based on LWE, and conceal a message with an ephemeral key shared by KEM.

We propose two versions of one encryption scheme based on the spLWE-based KEM, where one is IND-CPA secure and the other is an IND-CCA conversion of IND-CPA by the transformation proposed in [52]. We note that these different types of schemes can be applied to various circumstances.

3.1 Our Key Encapsulation Mechanism

We use a *reconciliation* technique in [43] which is the main tool to construct our spLWE-based KEM. In our KEM, the sender generates a random number $v \in \mathbb{Z}_{2q}$ for some even integer $q > 0$, and sends $\langle v \rangle_2$ where $\langle v \rangle_2 := [\lfloor \frac{2}{q} \cdot v \rfloor]_2 \in \mathbb{Z}_2$ to share $\lfloor v \rceil_2 := [\lfloor \frac{1}{q} \cdot v \rceil]_2 \in \mathbb{Z}_2$ securely. For all vectors $\boldsymbol{v} \in \mathbb{Z}_{2q}^k$, $\langle \boldsymbol{v} \rangle_2$ and $\lfloor \boldsymbol{v} \rceil_2$ are naturally defined by applying $\langle\rangle_2$ and $\lfloor\rceil_2$ component-wise, respectively. The receiver recovers $\lfloor v \rceil_2$ from $\langle v \rangle_2$ and sk using a special function named *rec*. The reconciliation function *rec* is defined as follows.

Definition 7. *For disjoint intervals* $I_0 := \left\{0, 1, \cdots, \lfloor \frac{q}{2} \rceil - 1\right\}$, $I_1 := \left\{-\lfloor \frac{q}{2} \rfloor, \cdots, -2, -1\right\}$ *and* $E = \left[-\frac{q}{4}, \frac{q}{4}\right) \cap \mathbb{Z}$, *we define*

$$rec : \mathbb{Z}_{2q} \times \mathbb{Z}_2 \to \mathbb{Z}_2 \text{ where } rec(w, b) := \begin{cases} 0 & \text{if } w \in I_b + E \mod 2q, \\ 1 & \text{otherwise.} \end{cases}$$

It is naturally extended to a vector-input function $\boldsymbol{rec} : \mathbb{Z}_{2q}^k \times \mathbb{Z}_2^k \to \mathbb{Z}_2^k$ *by applying rec componentwise.*

The following lemmas show that $\langle v \rangle_2$ reveals no information about $\lfloor v \rceil_2$, and *rec* decapsulates $\lfloor v \rceil_2$ correctly when it is provided with a proper approximation of v.

Lemma 6. *If* $v \in \mathbb{Z}_{2q}$ *is uniformly random, then* $\lfloor v \rceil_2$ *is uniformly random given* $\langle v \rangle_2$.

Proof. Suppose that $\langle v \rangle_2 = b \in \mathbb{Z}_2$. It implies that v is uniform over $I_b \cup (q + I_b)$. If $v \in I_b$, then $\lfloor v \rceil_2 = 0$, and if $v \in (q + I_b)$, then $\lfloor v \rceil_2 = 1$. Therefore $\lfloor v \rceil_2$ is uniformly random over $\{0, 1\}$ given $\langle v \rangle_2$. □

Lemma 7. *For* $v, w \in \mathbb{Z}_{2q}$, *if* $|v - w| < q/4$, *then* $rec(w, \langle v \rangle_2) = \lfloor v \rceil_2$.

Proof. Let $\langle v \rangle_2 = b \in \mathbb{Z}_2$, then $v \in I_b \cup (q + I_b)$. Then $\lfloor v \rceil_2 = 0$ if and only if $v \in I_b$. Since $(I_b + E) - E = I_b + (-\frac{q}{2}, \frac{q}{2})$ and $(q + I_b)$ are disjoint (mod $2q$), we know that $v \in I_b$ if and only if $w \in I_b + E$. □

The purpose of our KEM is sharing the ephemeral key from $\mathbf{u}^T\mathbf{A}\mathbf{s} + error$ and the reconciliation function between two parties as in [43]. Here, we describe our spLWE-based KEM for k-bit sharing as follows.

- KEM.Params(λ): generate a bit-length of shared key k, a bit-length of seed y and spLWE parameters $n, m, q, s, \rho, \theta, s', \rho', \theta'$ with λ-bit security. Publish all parameters by pp.

- KEM.Keygen(pp): sample $seed_A \leftarrow \{0,1\}^y$, $\mathbf{A} \leftarrow Gen(seed_A)$, $\mathbf{E} \leftarrow D_{\mathbb{Z},s}^{m \times k}$ and $\mathbf{S} \leftarrow \mathcal{U}(X_{n,\rho,\theta})^k$, and compute $\mathbf{B} = \mathbf{A}\mathbf{S} + \mathbf{E} \in \mathbb{Z}_q^{m \times k}$. For a secret key $\mathsf{sk} = \mathbf{S}$, publish a corresponding public key $\mathsf{pk} = (seed_A, \mathbf{B})$.

- KEM.Encap(pk,pp): sample $\boldsymbol{u} \leftarrow X_{m,\rho',\theta'}$, $(\boldsymbol{e}_1, \boldsymbol{e}_2) \leftarrow D_{\mathbb{Z},s'}^k \times D_{\mathbb{Z},s'}^n$ and $\boldsymbol{e}_3 \in \{0,1\}^k$. Let $\boldsymbol{v} = \boldsymbol{u}^T\mathbf{B} + \boldsymbol{e}_1 \in \mathbb{Z}_q^k$ and $\bar{\boldsymbol{v}} = 2\boldsymbol{v} + \boldsymbol{e}_3 \in \mathbb{Z}_{2q}^k$. Compute $\boldsymbol{c_1} = \langle \bar{\boldsymbol{v}} \rangle_2 \in \mathbb{Z}_2^k$ and $\boldsymbol{c_2} = \boldsymbol{u}^T\mathbf{A} + \boldsymbol{e_2} \in \mathbb{Z}_q^n$ from $\mathbf{A} \leftarrow Gen(seed_A)$. Send a ciphertext $\boldsymbol{c} = (\boldsymbol{c_1}, \boldsymbol{c_2}) \in \mathbb{Z}_2^k \times \mathbb{Z}_q^n$ to the receiver, and store an ephemeral secret key $\boldsymbol{\mu} = \lfloor \bar{\boldsymbol{v}} \rceil_2 \in \mathbb{Z}_2^k$.

- KEM.Decap($\boldsymbol{c}$, sk): If q is odd, compute $\boldsymbol{w} = 2\boldsymbol{c_2}^T\boldsymbol{S} \in \mathbb{Z}_q^k$, and ouput $\boldsymbol{\mu} = \boldsymbol{rec}(\boldsymbol{w}, \boldsymbol{c_1})$.

We would like to note that if q is even, the *doubling* process in the encapsulation phase, i.e. converting $\boldsymbol{v} = \boldsymbol{u}^T\mathbf{B} + \boldsymbol{e}_1$ to $\bar{\boldsymbol{v}} = 2\boldsymbol{v} + \boldsymbol{e}_3$, is not required.

3.2 Our KEM-Based Encryption Scheme

We now construct a public key encryption scheme based on the spLWE-based KEM in the previous section. When the message slot increases by one, the ciphertext spaces of our scheme grow only one or two bits, which is more efficient than the known LWE based encryption schemes [46], [48], where the growth is about $\log q$ bits.

PKE_1 (IND-CPA) : With a key exchange mechanism which shares a ℓ-bit length key, it is well-known that one can convert it into a public key encryption of the ℓ-bit length message having the same security as the key exchange mechanism. This conversion only includes XOR operations after generating an ephemeral key. Note that the ciphertext space is given as $\mathbb{Z}_q^n \times \mathbb{Z}_2^{2\ell}$, which is very efficient than $\mathbb{Z}_q^{n+\ell}$, ciphertext spaces of other LWE-based scehems.

PKE_1 is described as follows.

- $\mathrm{PKE}_1.\mathsf{Params}(\lambda)$: let ℓ be a message length, and run $\mathsf{KEM.Params}(\lambda)$ with $k = \ell$. Publish all parameters by pp.
- $\mathrm{PKE}_1.\mathsf{Keygen}(\mathsf{pp})$: output a key pair $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KEM.Keygen}(\mathsf{pp})$.
- $\mathrm{PKE}_1.\mathsf{Enc}(\mathsf{pk}, \boldsymbol{m}, \mathsf{pp})$: for $\boldsymbol{c}, \boldsymbol{\mu} \leftarrow \mathsf{KEM.Encap}(\mathsf{pk},\mathsf{pp})$, let $\boldsymbol{c'} = \boldsymbol{m} \oplus \boldsymbol{\mu}$ and output a ciphertext $(\boldsymbol{c}, \boldsymbol{c'})$.
- $\mathrm{PKE}_1.\mathsf{Dec}((\boldsymbol{c}, \boldsymbol{c'}), \mathsf{sk})$: for $\boldsymbol{\mu} = \mathsf{KEM.Decap}(\boldsymbol{c}, \mathsf{sk})$, output $\boldsymbol{m} = \boldsymbol{c'} \oplus \boldsymbol{\mu}$.

PKE_2 (IND-CCA) : We can apply the transformation suggested in [52], which can improve security of the existing public key encryption schemes. As a trade-off of security, this scheme requires a more complex construction than PKE_1, but note that this also use light operations such as XOR or hashing, which are not serious tasks for implementation.

We specially denote the encryption phase of PKE_1 by $\mathrm{PKE}_1.\mathsf{Enc}(\mathsf{pk}, \boldsymbol{m}, \mathsf{pp}; \boldsymbol{r})$ to emphasize that a random bit-string $\boldsymbol{r}$ is used for random sampling. Here, $\mathrm{PKE}_1.\mathsf{Enc}(\mathsf{pk}, \boldsymbol{m}, \mathsf{pp}; \boldsymbol{r})$ becomes deterministic.

It also requires quantumly secure hash functions $G : \{0,1\}^{k_1+\ell} \to \{0,1\}^*$, $H : \{0,1\}^{k_1} \to \{0,1\}^{k_2}$ and $H' : \{0,1\}^{k_1} \to \{0,1\}^{k_3}$, where k_i will be determined later. With these parameters, our scheme has a ciphertext space $\mathbb{Z}_q^n \times \mathbb{Z}_2^{k_1+k_2+k_3+\ell}$, which also gradually increases with the growth of message slot.

PKE_2 is described as follows.

- $\mathrm{PKE}_2.\mathsf{Params}(\lambda)$: let ℓ be a message length and $k_i > 0$ be integers such that hash functions $G : \{0,1\}^{k_1+\ell} \to \{0,1\}^*$, $H : \{0,1\}^{k_1} \to \{0,1\}^{k_2}$ and $H' : \{0,1\}^{k_1} \to \{0,1\}^{k_3}$ have λ-bit security. Let pp be an output of $\mathsf{KEM.Params}(\lambda)$ with $k = k_1$. Publish ℓ, pp and k_i.
- $\mathrm{PKE}_2.\mathsf{Keygen}(\mathsf{pp})$: output a key pair $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KEM.Keygen}(k_1)$.
- $\mathrm{PKE}_2.\mathsf{Enc}(\mathsf{pk}, \boldsymbol{m}, \mathsf{pp}, k_i)$: randomly choose $\boldsymbol{\omega} \leftarrow \{0,1\}^{k_1}$, and let $\boldsymbol{c_m} = H(\boldsymbol{\omega}) \oplus \boldsymbol{m}$. Compute $\boldsymbol{c}_h = H'(\boldsymbol{\omega})$ and $(\boldsymbol{c}, \boldsymbol{c'}) \leftarrow \mathrm{PKE}_1.\mathsf{Enc}(\mathsf{pk}, \boldsymbol{\omega}; G(\boldsymbol{\omega}||\boldsymbol{c_m}))$. Output a ciphertext $(\boldsymbol{c}, \boldsymbol{c'}, \boldsymbol{c}_h, \boldsymbol{c_m})$.
- $\mathrm{PKE}_2.\mathsf{Dec}((\boldsymbol{c}, \boldsymbol{c'}, \boldsymbol{c}_h, \boldsymbol{c_m}), \mathsf{sk}, \mathsf{pp}, k_i)$: compute $\boldsymbol{\omega} = \mathrm{PKE}_1.\mathsf{Dec}((\boldsymbol{c}, \boldsymbol{c'}), \mathsf{sk})$ and $\boldsymbol{m} = H(\boldsymbol{\omega}) \oplus \boldsymbol{c_m}$. Check whether $(\boldsymbol{c}, \boldsymbol{c'}) = \mathrm{PKE}_1.\mathsf{Enc}(\mathsf{pk}, \boldsymbol{\omega}; G(\boldsymbol{w}||\boldsymbol{c_m}))$ and $\boldsymbol{c}_h = H'(\boldsymbol{\omega})$. If so, output $\boldsymbol{m}$, otherwise output $\perp$.

3.3 Security

In this section, we show (IND-CPA, IND-CCA) security of our encryption scheme (PKE_1, PKE_2). Security of our encryption scheme is reduced to security of KEM and security of KEM comes from hardness of spLWE. Consequently, under the hardness of spLWE, PKE_1 can reach to IND-CPA security and PKE_2 achieves further quantumly IND-CCA security with the random oracle assumption. Here is a statement for security of KEM.

Theorem 1. *Assuming the hardness of* $\mathsf{spLWE}_{n,m,q,s,\rho,\theta}$, *and* $\mathsf{spLWE}_{n,m,q,s',\rho',\theta'}$, *our* KEM *is IND-CPA secure.*

Proof. (Sketch) By Lemma 3, one cannot extract any information about $\boldsymbol{\mu} = \lfloor \boldsymbol{v} \rceil_2$ with $\boldsymbol{c_1}$. Moreover, even if one can know some information of $\boldsymbol{v}$, the distribution of $(\boldsymbol{c_2}, \boldsymbol{v})$ can be regarded as LWE instances as :

$$(\boldsymbol{c_2}, \boldsymbol{v}) = (\boldsymbol{u}^T \cdot \mathbf{A} + \boldsymbol{e_2}, \boldsymbol{u}^T \cdot \mathbf{B} + \boldsymbol{e_1}) = (\mathbf{C}, \mathbf{C} \cdot \mathbf{S} + \boldsymbol{e'})$$

for $\mathbf{C} = \boldsymbol{u}^{\mathbf{T}} \cdot \mathbf{A} + \boldsymbol{e_2}$ and for some $\boldsymbol{e'}$. Thus, hardness of spLWE insures that the distribution of $(\boldsymbol{c_2}, \boldsymbol{v})$ is indistinguishable from the uniform distribution over $\mathbb{Z}_q^n \times \mathbb{Z}_q^k$. □

We refer [43] for the detailed IND-CPA game-based proof, where the only difference is that we assume the hardness of spLWE, not RLWE.

It is well-known in many cryptographic texts that PKE_1 has the same security level with KEM. Hence, security of PKE_1 has been demonstrated from the previous theorem. Moreover, the transformation of [52] gives quantumly IND-CCA security for PKE_2, when it is converted from an IND-CPA secure PKE with random oracle modeled hashes. When the aforementioned statements are put together, we can establish the following security theorem.

Theorem 2. *Assuming the hardness of* $\mathsf{spLWE}_{n,m,q,s,\rho,\theta}$, $\mathsf{spLWE}_{n,m,q,s',\rho',\theta'}$, PKE_1 *is IND-CPA secure, and* PKE_2 *is quantumly IND-CCA secure with further assumption that the function* G, H *and* H' *are modeled as random oracles.*

Proof. (Sketch) We only need to show that PKE_2 is IND-CCA secure. The transformation of [52] actually make an IND-CCA secure public key encryption from a public key encryption which is *well-spread* and *one-way*, and we briefly explain why (IND-CPA) PKE_1 is well-spread and one-way.

- Well-spreadness: Note that a ciphertext of PKE_1 is of the form

$$(\boldsymbol{c}_1, \boldsymbol{c}_2) = \left(\langle \mathbf{2}(\boldsymbol{u}^T B + \boldsymbol{e}_1) + \boldsymbol{e}_3 \rangle_2, \boldsymbol{u}^T A + \boldsymbol{e}_2\right),$$

  where $\boldsymbol{u} \leftarrow X_{m,\rho',\theta'}, (\boldsymbol{e}_1, \boldsymbol{e}_2) \leftarrow D^k_{\mathbb{Z},s'} \times D^n_{\mathbb{Z},s'}$. From hardness of spLWE, distributions of $\boldsymbol{u}^T B + \boldsymbol{e}_1 \in \mathbb{Z}_q^k$ and $\boldsymbol{u}^T A + \boldsymbol{e}_2 \in \mathbb{Z}_q^n$ are statistically close to uniform distributions over $\mathbb{Z}_q^k$ and $\mathbb{Z}_q^n$, and then PKE_1 is well-spread.
- One-wayness: With an oracle $\mathcal{O}$ finding $\boldsymbol{m}$ from $\mathrm{PKE}_1.\mathrm{Enc}(\mathsf{pk}, \boldsymbol{m})$ for any pk with probability ϵ, an adversary equipped with $\mathcal{O}$ wins the IND-CPA game for PKE_1 with bigger advantage than $\frac{\epsilon}{2}$: After given $\mathrm{PKE}_1.\mathrm{Enc}(\mathsf{pk}, \boldsymbol{m}_b)$, the adversary outputs the answer of $\mathcal{O}$. It can be easily shown that the advantage is bigger than $\frac{\epsilon}{2}$. □

3.4 Correctness

Similar to the security case, correctness of our (IND-CPA, IND-CCA) encryption scheme is dependent on that of our spLWE-based KEM. We remark that generally, one can obtain some correctness condition for all LWE variants by examining a bound of error term in the proof below. Here, we assume $s = s', \rho = \rho'$ and $\theta = \theta'$, which is used for our parameter instantiation.

Theorem 3. *Let* $n, m, \sigma, \rho, \theta$ *be parameters in* $\mathsf{spLWE}_{n,m,q,\sigma,\rho,\theta}$, *and* ℓ *be the shared key length in* KEM. *For a per-symbol error probability* γ, *the* KEM *decapsulates correctly if*

$$q \geq 8s\rho\sqrt{\frac{2\theta}{\pi}\ln(2/\gamma)}.$$

Proof. As shown in the description of KEM.Decap, the ephemeral key is decapsulated correctly if $|\bar{\boldsymbol{v}} - \boldsymbol{w}| < q/4$ by lemma 7. Since $\bar{\boldsymbol{v}} = 2\boldsymbol{u}^T\boldsymbol{A}\boldsymbol{S} + 2\boldsymbol{u}^T\boldsymbol{E} + 2\boldsymbol{e}_1 + \boldsymbol{e}_3$, and $\boldsymbol{w} = 2\boldsymbol{u}^T\boldsymbol{A}\boldsymbol{S} + 2\boldsymbol{e}_2\boldsymbol{S}$, it is rephrased by

$$|2\boldsymbol{u}^T \cdot \mathbf{E} - 2\boldsymbol{e_1} \cdot \mathbf{S} + 2\boldsymbol{e_2} + \boldsymbol{e_3}| < q/4,$$

which is equivalent to

$$|2\langle \boldsymbol{u}, [\mathbf{E}]^j \rangle + 2\langle -\boldsymbol{e_1}, [\mathbf{S}]^j \rangle + 2[\boldsymbol{e_2}]_j + [\boldsymbol{e_3}]_j| < q/4, 1 \le j \le \ell$$

where $\boldsymbol{u} \leftarrow X_{m,\rho',\theta'}, [\mathbf{S}]^j \leftarrow X_{n,\rho,\theta}, [\mathbf{E}]^j \leftarrow D^m_{\mathbb{Z},s}, \boldsymbol{e}_1 \leftarrow D^n_{\mathbb{Z},s'}, [\boldsymbol{e_2}]_j \leftarrow D_{\mathbb{Z},s'}, [\boldsymbol{e_3}]_j \leftarrow \{0,1\}$. For simplicity, we ignore the small term $2[\boldsymbol{e_2}]_j + [\boldsymbol{e_3}]_j$. (This is compensated in our final choice of parameters.) By applying lemma 1 to a $(m+n)$ dimensional vector $\boldsymbol{x} = (\boldsymbol{u}, [\boldsymbol{S}]^j)$ and the bound $Ts\|\boldsymbol{x}\| = q/8$, we came to have per-symbol error probability γ,

$$\gamma = 2\exp(-\pi(\frac{q}{8s\rho\sqrt{(2\theta)}})^2)$$

from $T = \frac{q}{8s\rho\sqrt{2\theta}}$. From the equation above, we get the bound on q as the statement.

4 The Hardness of spLWE

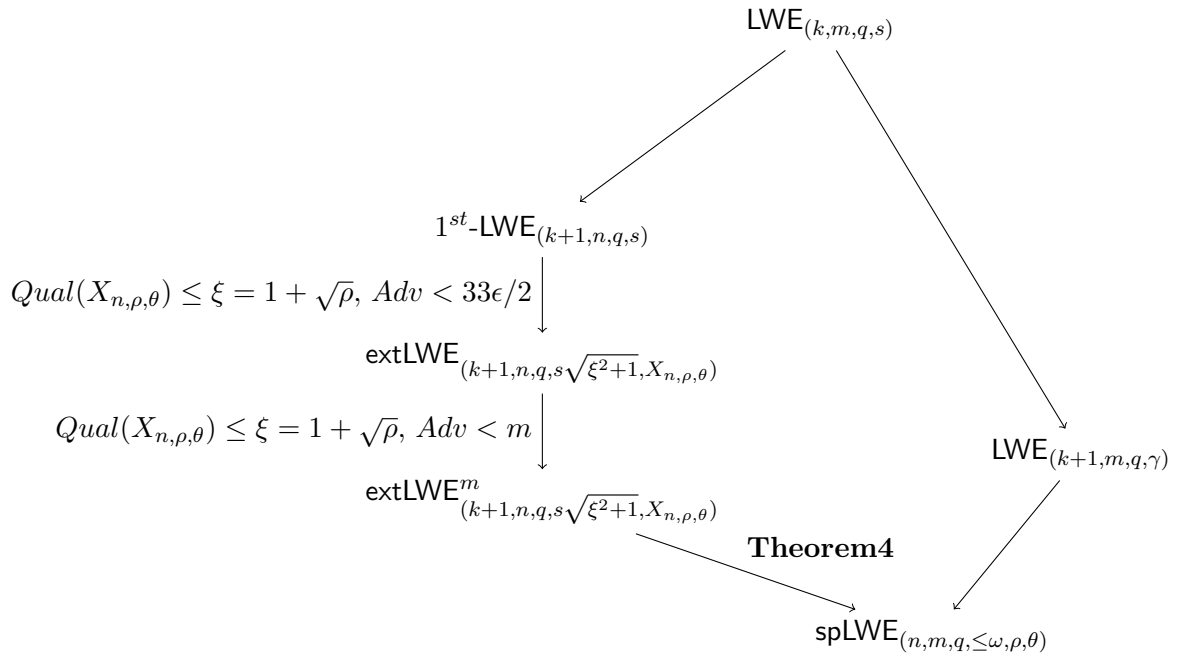
In this section, we show the hardness of spLWE via a security reduction and concrete attacks. First, we show spLWE is as hard as worst-case lattice problems to solve. For that, we provide a reduction from LWE to spLWE by generalizing the reduction [13]. Next, we also present modified attacks for spLWE, which exploit the sparsity of a secret from all known attacks for LWE and binLWE [8, 14].

4.1 A Reduction from LWE to spLWE

To show our reduction for spLWE, we need extLWE^m problem whose hardness was proved in [13]. They showed that for a set X of quality ξ, there exists a reduction from $\mathsf{LWE}_{k,m,q,s}$ to $\mathsf{extLWE}^m_{(k+1,n,q,\beta=\sqrt{s^2\xi^2+s^2},X)}$. (Here, $n \le m$) Based on a reduction from LWE to extLWE in [13], we prove a reduction of spLWE as shown in the diagram below. Here, ω, γ and s are constant satisfying

$$\omega = s\rho\sqrt{2\theta(2+2\sqrt{\rho}+\rho)},\ \gamma = \rho s\sqrt{\theta(2+2\sqrt{\rho}+\rho)},\ \beta \ge (\ln(2n(1+1/\epsilon))/\pi)^{1/2}/q.$$

Because $\mathsf{Qual}(X_{n,\rho,\theta}) < 1+\sqrt{\rho}$ by lemma 5, $\mathsf{extLWE}_{k+1,n,q,s\sqrt{(1+\sqrt{\rho})^2+1},X_{n,\rho,\theta}}$ is hard based on the hardness of $\mathsf{LWE}_{k,n,q,s}$. Following theorem shows that $\mathsf{spLWE}_{n,m,q,\le\omega,\rho,\theta}$ problem can be hard based on the hardness of $\mathsf{LWE}_{k,m,q,\gamma}$ and $\mathsf{extLWE}_{n,m,q,s\sqrt{(1+\sqrt{\rho})^2+1},X_{n,\rho,\theta}}$ for the $\omega, \gamma > 0$ as above. In particular, if $\log\left(\binom{n}{\theta}\cdot(2l+2)^\theta\right) \ge k\log q + 2\log(1/\delta)$, there is a reduction from $\mathsf{LWE}_{k,m,q,s}$ to $\mathsf{spLWE}_{n,m,q,\le\omega,\rho,\theta}$.



Theorem 4. *Let k, n, m, $\rho = 2^l$, θ, $q \in \mathbb{N}$, $\epsilon \in (0, 1/2)$, and $\delta, \omega, \beta, \gamma > 0$ such that*

$$\beta \geq \sqrt{2\ln(2n(1+1/\epsilon))/\pi}/q \text{ where } \beta = s\sqrt{(1+\sqrt{\rho})^2+1},$$

$$\omega = \rho\beta\sqrt{2\theta},\ \gamma = \rho\beta\sqrt{\theta},\ \log\left(\binom{n}{\theta}\cdot(2l+2)^\theta\right) \geq k\log q + 2\log(1/\delta).$$

There exist (two) reductions to $\mathsf{spLWE}_{n,m,q,\leq\omega,\rho,\theta}$ from $\mathsf{extLWE}^m_{k,n,q,\beta,X_{n,\rho,\theta}}$, $\mathsf{LWE}_{k,m,q,\gamma}$. An advantage of $\mathcal{A}$ for $\mathsf{spLWE}_{n,m,q,\leq\omega,\rho,\theta}(\mathcal{D})$ is bounded as follows:

$$Adv[\mathcal{A}] \leq 2Adv[\mathcal{C}_1] + Adv[\mathcal{C}_2] + 4m\epsilon + \delta$$

for the algorithms (distinguishers) of $\mathsf{extLWE}^m_{k,n,q,\beta,X_{n,\rho,\theta}}$, $\mathsf{LWE}_{k,m,q,\gamma}$, $\mathcal{C}_1$ and $\mathcal{C}_2$ respectively.

Proof. The proof follows by a sequence of distribution to use hybrid argument as stated in [13]. We take into account the following six distributions:

$H_0 := \{(\mathbf{A}, \mathbf{b} = \mathbf{A}^T\mathbf{x} + \mathbf{e}) \mid \mathbf{A} \leftarrow \mathbb{T}_q^{n\times m}, \mathbf{x} \leftarrow X_{n,\rho,\theta}, \mathbf{e} \leftarrow D^m_{\alpha'} \text{ for } \alpha' = \sqrt{\beta^2\|\mathbf{x}\|^2+\gamma^2} \leq \rho\beta\sqrt{2\theta} = \omega\}$.

$H_1 := \{(\mathbf{A}, \mathbf{A}^T\mathbf{x} - \mathbf{N}^T\mathbf{x} + \hat{\mathbf{e}} \mod 1) \mid \mathbf{A} \leftarrow \mathbb{T}_q^{n\times m}, \mathbf{x} \leftarrow X, \mathbf{N} \leftarrow D^{n\times m}_{\frac{1}{q}\mathbb{Z},\beta}, \hat{\mathbf{e}} \leftarrow D^m_\gamma\}$.

$H_2 := \{(q\mathbf{C}^T\mathbf{B} + \mathbf{N}, q\mathbf{B}^T\mathbf{C}\mathbf{x} + \hat{\mathbf{e}}) \mid \mathbf{B} \leftarrow \mathbb{T}_q^{k\times m}, \mathbf{C} \leftarrow \mathbb{T}_q^{k\times n},\ \mathbf{x} \leftarrow X, \mathbf{N} \leftarrow D^{n\times m}_{\frac{1}{q}\mathbb{Z},\beta}, \hat{\mathbf{e}} \leftarrow D^m_\gamma\}$.

$H_3 := \{(q\mathbf{C}^T\mathbf{B} + \mathbf{N}, \mathbf{B}^T\mathbf{s} + \hat{\mathbf{e}}) \mid \mathrm{s} \leftarrow \mathbb{Z}_q^k,\ \mathbf{B} \leftarrow \mathbb{T}_q^{k\times m}, \mathbf{C} \leftarrow \mathbb{T}_q^{k\times n},\ \mathbf{N} \leftarrow D^{n\times m}_{\frac{1}{q}\mathbb{Z},\beta}, \hat{\mathbf{e}} \leftarrow D^m_\gamma\}$.

$H_4 := \{(q\mathbf{C}^T\mathbf{B} + \mathbf{N}, \mathbf{u}) \mid \mathbf{u} \leftarrow \mathbb{T}^m,\ \mathbf{B} \leftarrow \mathbb{T}_q^{k\times m}, \mathbf{C} \leftarrow \mathbb{T}_q^{k\times n},\ \mathbf{N} \leftarrow D^{n\times m}_{\frac{1}{q}\mathbb{Z},\beta}\}$.

$H_5 := \{(\mathbf{A}, \mathbf{u}) \mid \mathbf{A} \leftarrow \mathbb{T}_q^{n\times m}, \mathbf{u} \leftarrow \mathbb{T}^m\}$.

Let $\mathcal{B}_i$ be the distinguisher for the distributions between H_i and H_{i+1} for $0 \leq i \leq 4$. There are some efficient transformations from the distributions $(\mathbf{C}, \mathbf{A}, \mathbf{N}^T\mathbf{z}), (\mathbf{C}, \hat{\mathbf{A}}, \mathbf{N}^T\mathbf{z})$ to H_1, H_2, from $(\mathbf{B}, \mathbf{B}^T\mathbf{s} + \hat{\mathbf{e}}), (\mathbf{B}, \mathbf{u})$ to H_3, H_4, and from $(\mathbf{C}, \hat{\mathbf{A}}), (\mathbf{C}, \mathbf{A})$ to H_4, H_5. In fact, the samples $(\mathbf{C}, \hat{\mathbf{A}}, \mathbf{N}^T\mathbf{z})$, $(\mathbf{B}, \mathbf{B}^T\mathbf{s} + \hat{\mathbf{e}})$, and $(\mathbf{C}, \hat{\mathbf{A}})$ are $\mathsf{extLWE}^m_{k,n,q,\beta,X}$, $\mathsf{LWE}_{k,m,q,\gamma}$ and $\mathsf{extLWE}^m_{k,n,q,\beta,\{0^n\}}$ samples respectively. The others are uniform distribution samples in the corresponding domain. It follows that $Adv[\mathcal{B}_1]$, $Adv[\mathcal{B}_3]$, $Adv[\mathcal{B}_4]$ are bound by the distinguishing advantages of $\mathsf{extLWE}^m_{k,n,q,\beta,X}$, $\mathsf{LWE}_{k,m,q,\gamma}$, $\mathsf{extLWE}^m_{k,n,q,\beta,\{0^n\}}$ respectively.

Since $\|\mathbf{x}\| \leq \rho\sqrt{\theta}$, and $\beta \geq \sqrt{2\ln(2n(1+1/\epsilon))/\pi}/q \geq \sqrt{2}\eta_\epsilon(\mathbb{Z}^n)/q$ from lemma 3, it follows that the statistical distance between $-\mathbf{N}^T\mathbf{x} + \hat{\mathbf{e}}$ and $D^m_{\alpha'}$ is at most $4m\epsilon$ by lemma 2. This gives $Adv[\mathcal{B}_0] \leq 4m\epsilon$. The last $Adv[\mathcal{B}_2]$ is bound by δ from the Leftover hash lemma. To sum up, $Adv[\mathcal{A}] \leq 2Adv[\mathcal{C}_1] + Adv[\mathcal{C}_2] + 4m\epsilon + \delta$ with trivial reduction to $\mathsf{extLWE}^m_{k,n,q,\beta,\{0^n\}}$ from $\mathsf{extLW}E^m_{k,n,q,\beta,X}$. □

4.2 Attacks for spLWE

There exist many attacks for LWE including a dual attack and primal attacks ([4], [20]). Here, we exclude a combinatorial BKW algorithm, the Arora and Ge algorithm and their variants, as they are not suitable in our case ([2], [6], [21], [33], [28]). Since the analysis of traditional dual attack is based on the (discrete) Gaussian error (and secret in the LWE normal form), these traditional attacks are not directly applicable to spLWE. Therefore, we modify those attacks to analyze concrete hardness of spLWE. We also consider random guess on a sparse secret vector $\mathbf{s}$ as in appendix.

Dual (distinguish) Attack Assume that we are given $(\mathbf{A}, \boldsymbol{b}) \in \mathbb{Z}_q^{m\times n} \times \mathbb{Z}_q^m$ and want to distinguish whether they are uniform random samples or spLWE samples. For a constant $c \in \mathbb{R}$ with $c \leq q$, consider a lattice $L_c(\mathbf{A})$ defined by

$$L_c(\mathbf{A}) = \left\{(\boldsymbol{x}, \boldsymbol{y}/c) \in \mathbb{Z}^m \times (\mathbb{Z}/c)^n : \boldsymbol{x}^T\mathbf{A} = \boldsymbol{y} \mod q\right\}.$$

If the samples $(\mathbf{A}, \boldsymbol{b})$ came from spLWE, for $(\boldsymbol{x}, \boldsymbol{y}) \in L_c(\mathbf{A})$, we have

$$\begin{aligned}\langle \boldsymbol{x}, \boldsymbol{b}\rangle &= \langle \boldsymbol{x}, \mathbf{A}\boldsymbol{s} + \boldsymbol{e}\rangle \\ &= \langle \boldsymbol{x}, \mathbf{A}\boldsymbol{s}\rangle + \langle \boldsymbol{x}, \boldsymbol{e}\rangle \\ &= c\langle \boldsymbol{y}, \boldsymbol{s}\rangle + \langle \boldsymbol{x}, \boldsymbol{e}\rangle \mod q\end{aligned}$$

For a sufficiently small vector $(\boldsymbol{v}, \boldsymbol{w}) \in L_c(\mathbf{A})$, the value $\langle \boldsymbol{v}, \boldsymbol{b}\rangle \mod q$ becomes small when the samples are spLWE ones, and $\langle \boldsymbol{v}, \boldsymbol{b}\rangle \mod q$ is uniformly distributed when $(\mathbf{A}, \boldsymbol{b})$ came from the uniform distribution. Hence, one can decide whether the samples came from spLWE distribution or uniform distribution from the size of $\langle \boldsymbol{v}, \boldsymbol{b}\rangle \mod q$ with some success probability. We now determine how small a vector $(\boldsymbol{v}, \boldsymbol{w})$ must be found as follows. First, we estimate the length of $(\boldsymbol{v}, \boldsymbol{w}) \in L_c(\mathbf{A})$. One can easily check that

$$\left(\begin{array}{c|c} I_m & 0 \\ \hline \frac{1}{c}\mathbf{A}^T & \frac{q}{c}I_n \end{array}\right)$$

is a basis matrix of $L_c(\mathbf{A})$. Hence, we can figure out $\dim(L_c(\mathbf{A})) = m + n$ and $\det(L_c(\mathbf{A})) = (q/c)^n$.

Therefore, a lattice reduction algorithm with a root Hermite factor δ_0 gives $(\boldsymbol{v}, \boldsymbol{w}) \in L_c(\mathbf{A})$, such that

$$||(\boldsymbol{v}, \boldsymbol{w})|| = \delta_0^{m+n}(q/c)^{\frac{n}{m+n}}, \tag{1}$$

and the length is minimized when $m = \sqrt{n(\log q - \log c)/\log \delta_0} - n$.

Next, we consider the distribution of $c\langle \boldsymbol{w}, \boldsymbol{s}\rangle + \langle \boldsymbol{v}, \boldsymbol{e}\rangle \mod q$. Here, we assume that the coefficients of sparse vector $\boldsymbol{s}$ are independently sampled by $(b_1d_1, b_2d_2, \ldots, b_nd_n)$ where $d_i \leftarrow \mathrm{Ber}(n, \theta/n)$, $b_i \leftarrow \{\pm 1, \pm 2, \pm 4, \ldots, \pm\rho\}$, and $\rho = 2^l$ for some $l \in \mathbb{Z}_{\geq 0}$. Since $c\langle \boldsymbol{w}, \boldsymbol{s}\rangle$ is the sum

of many independent random variables, asymptotically it follows a Gaussian distribution with mean 0, and variance $(c||\boldsymbol{w}||)^2 \cdot \frac{2\theta(4^{l+1}-1)}{3n(2l+2)}$. From that $\langle \boldsymbol{v}, \boldsymbol{e} \rangle$ follows a Gaussian distribution with mean 0, variance $(\sigma||\boldsymbol{v}||)^2$, and lemma 4, we have distinguishing advantage

$$\exp(-\pi(s'/q)^2) \text{ where } s' = \sqrt{2\pi}\sqrt{\sigma^2||\boldsymbol{v}||^2 + c^2\frac{2\theta(4^{l+1}-1)}{3n(2l+2)}||\boldsymbol{w}||^2}. \tag{2}$$

From above equations 1, 2 with distinguishing advantage ϵ, we need to find small δ_0 such that

$$\delta_0 = (c/q)^{\frac{-n}{(m+n)^2}}(\frac{q}{M}\sqrt{\ln(1/\epsilon)/\pi})^{1/(m+n)} \text{ where } M = \sqrt{2\pi}\sqrt{\sigma^2\frac{m}{m+n} + c^2\frac{2\theta(4^{l+1}-1)}{3n(2l+2)}\frac{n}{m+n}}$$

5 Parameter Selection and Implementation Result

5.1 Parameter Selection

To deduce some appropriate parameters, we assume that the best known classical and quantum SVP (sieving) algorithm in dimension k runs in time $2^{0.292k}$ and $2^{0.265k}$ respectively [10, 34]. The BKZ 2.0 lattice basis reduction algorithm gives the root Hermite factor $\delta_0 \approx (\frac{k}{2\pi e}(\pi k)^{1/k})^{1/2(k-1)}$ for block size k [15], and the iteration number of exact SVP solver is $\frac{n^3}{k^2}\log n$ [29].

We consider a direct CVP attack by sieving [35], modified dual (distinguish) and embedding attack. Moreover, since our secret key is a sparse vector, our attack can be improved if one can guess some components of secret to be zero. After that, we can apply the attack to a smaller dimensional spLWE instances. We denote the probability of the correct guessing t components from n components by $p_{n,t,\theta}$. It can be computed as $\binom{n-\theta}{t}/\binom{n}{t}$.

To sum up the previous sections, the parameters must satisfy the followings for the quantum security:

- $n\log q \cdot (2l+1)^\theta \cdot \binom{n}{\theta} > 2^{2\lambda}$ from bruteforce attack (grover algorithm), where $\binom{n}{\theta} = \frac{n!}{(\theta!)(n-\theta!)}$ (For classical security, 2λ becomes λ)
- Let $T(n, q, \theta, s, l)$ be a BKZ 2.0 running time to get root Hermite factor δ_0, which satisfies the following equation:

  $$\delta_0 = \max_{1<c<q, 1\le m\le n}\left\{(c/q)^{\frac{-n}{(m+n)^2}}(\frac{q}{M}\sqrt{\ln(1/\epsilon)/\pi})^{1/(m+n)}\right\}$$

  where

  $$M = \sqrt{2\pi}\cdot\sqrt{\sigma^2\frac{m}{m+n} + c^2\frac{2\theta(4^{l+1}-1)}{3n(2l+2)}\frac{n}{m+n}}.$$

  Taking into the probability $p_{n,t,\theta}$, our parameters should satisfy the following:

  $$\min_t\left\{\frac{1}{p_{n,t,\theta}}\cdot T(n-t, q, \theta, s, l)\right\} > 2^\lambda \text{ where } p_{n,t,\theta} = \binom{n-\theta}{t}/\binom{n}{t}$$

  .
- To prevent the direct CVP attack, n and θ should satisfy the following equation:

  $$\min_t\left\{\frac{1}{p_{n,t,\theta}}\cdot 2^{0.265(n-t)}\right\} > 2^\lambda$$

  For classical security, 0.265 becomes 0.292.
- For the correctness, $q \ge 8s\rho\sqrt{\frac{2\theta}{\pi}\ln(2/\gamma)}$ by the Lemma 7.
- The parameters k_1 and k_2 are a symmetric key length of XOR operations, and k_3 is a length of hash value. For λ-bit security, it is known that k_1 and k_2 should be λ (2λ) and k_3 should be 2λ (3λ) in classical (quantum) security model.

5.2 Implementation Result

We use C++ on a Linux-based system, with GCC compiler and apply the Eigen library (www.eigen.tuxfamily.org), which makes vector and matrix operations fast. To sample $\boldsymbol{u}$ efficiently in our encryption algorithm, we assume that there are only one non-zero element in each n/θ-size block. To follow the previous reduction and security proof, we need a sampling of discrete Gaussian distribution when we generate error vectors in key generation and encryption algorithm. We use *box-muller transformation* to generate discretized Gaussian distribution. In the case below, message space length is 32-byte and secret key is ternary vector. We used PC (Macbook Pro) with CPU 2.6GHz Intel Core i5 without parallelization.

Parameters					IND-CPA				IND-CCA		
λ	n	q	s	θ	Setup(ms)	Enc(μs)	Dec(μs)	Cptx(byte)	Enc(μs)	Dec(μs)	Cptx(byte)
72	300	382	5	27	9.8	96	41	401	116	130	435
96	400	441	5	36	16.3	167	62	513	181	182	548
128	565	477	5	42	29.3	273	102	700	291	282	733

Table 1: Implementation result in classical hardness with 256 bit message

Parameters					IND-CPA				IND-CCA		
λ	n	q	s	θ	Setup(ms)	Enc(μs)	Dec(μs)	Cptx(byte)	Enc(μs)	Dec(μs)	Cptx(byte)
72	300	410	5	31	9.8	96	41	401	108	130	435
96	400	477	5	42	16.0	163	56	514	186	191	548
128	565	520	5	50	129.5	314	106	770	313	302	804

Table 2: Implementation result in quantum hardness with 256 bit message

We also compare our implemetation with software implementation in [26], which implements LWE-based PKE [48] and Ring version PKE [37, 38]. Their implementation is executed on an Intel Core 2 Duo CPU running at 3.00 GHz PC. Parameters in each rows are secure in same security parameters.

Our scheme			[26]	LWE		RLWE	
(n, q, s, θ)	Enc	Dec	(n, q, s)	Enc	Dec	Enc	Dec
(150, 285, 5.0, 15)	0.027	0.011	(128, 2053, 6.77)	3.01	1.24	0.76	0.28
(300, 396, 5.0, 29)	0.063	0.019	(256, 4093, 8.87)	11.01	2.37	1.52	0.57
(400, 545, 5.0, 55)	0.109	0.026	(384, 4093, 8.35)	23.41	3.41	2.51	0.98
(560, 570, 5.0, 60)	0.223	0.04	(512, 4093, 8.0)	46.05	4.52	3.06	1.18

Table 3: Our scheme vs. LWE vs. RLWE: Time in milliseconds for encryption and decryption for a 16-byte plaintext.

The table above shows that our PKE scheme is about 20 times faster than RLWE-based PKE scheme in [37, 38]. The sparsity of secret vector make modulus size q smaller and complexity in encryption/decryption algorithm lower.

Acknowledgements. We thank Damien Stehlé for helpful discussions on the initial version of our reduction, and Duhyeong Kim for pointing out some typos in our reduction. We also thank Yongsoo Song and the ICISC reviewers for their useful comments. This work was supported by Samsung Research Funding Center of Samsung Electronics under Project Number SRFC-TB1403-00.

6 Appendix

6.1 Attacks for Search spLWE

Dual (search) Attack. In this section, we assume the Geometric Series Assumption (GSA) on q-ary lattices, introduced by Schnorr [50], and this will be used to estimate the length of last

vector of BKZ 2.0 reduced basis. Let $\mathbf{B} = \{\boldsymbol{b}_1, \cdots, \boldsymbol{b}_n\}$ be a basis for an n-dimensional lattice Λ, which is reduced by the BKZ 2.0 with root Hermite factor δ_0, then the GSA says:

$$\|\boldsymbol{b}_i^*\| = \beta^{i-1} \cdot \|\boldsymbol{b}_1^*\| \text{ for some constant } 0 < \beta \leq 1,$$

where $\{\boldsymbol{b}_1^*, \cdots, \boldsymbol{b}_n^*\}$ is the Gram-schmidt orthogonalization of $\{\boldsymbol{b}_1, \cdots, \boldsymbol{b}_n\}$. From $\|\boldsymbol{b}_1\| = \delta_0^n \cdot \det(\mathbf{B})^{1/n}$, we have:

$$\det(\mathbf{B}) = \prod_{i=1}^{n} \|\boldsymbol{b}_i^*\| = \prod_{i=1}^{n} \beta^{i-1} \cdot \|\boldsymbol{b}_1^*\| = \beta^{\frac{(n-1)n}{2}} \cdot \delta_0^{n^2} \cdot \det(\mathbf{B}).$$

From the equation above, it follows that $\beta = \delta_0^{-2n^2/(n-1)n}$. Since BKZ reduced basis satisfies $\boldsymbol{b}_i = \boldsymbol{b}_i^* + \sum_{j=0}^{i-1} \mu_{ij} \cdot \boldsymbol{b}_j^*$ with $|\mu_{ij}| \leq 1/2$, one can show that,

$$\|\boldsymbol{b}_i\| \leq ||\boldsymbol{b}_1|| \cdot \sqrt{\frac{1-\beta^{2i-2}}{4-4\beta^2} + \beta^{2i-2}}.$$

We now describe the dual attack against a small number of LWE instances $(\mathbf{A}, \mathbf{A}\boldsymbol{s} + \boldsymbol{e}) = (\mathbf{A}, \boldsymbol{b}) \in \mathbb{Z}^{m \times n} \times \mathbb{Z}^m$. For some constant $c \in \mathbb{N}$ with $c \leq q$, we consider a scaled lattice $\Lambda_c(\mathbf{A})$.

$$\Lambda_c(\mathbf{A}) = \{(\boldsymbol{x}, \boldsymbol{y}/c) \in \mathbb{Z}^m \times (\mathbb{Z}^n/c) : \boldsymbol{x}\mathbf{A} = \boldsymbol{y} \bmod q\}.$$

A dimension and determinant of the lattice $\Lambda_c(\mathbf{A})$ is $n+m$ and $(q/c)^n$ respectively. With the assumptions above, we can obtain vectors $\{(\boldsymbol{v}_i, \boldsymbol{w}_i)\}_{1 \leq i \leq n}$ in $\Lambda_c(\mathbf{A})$ such that,

$$\|(\boldsymbol{v}_i, \boldsymbol{w}_i)\| \leq \delta_0^{m+n} \cdot (q/c)^{\frac{n}{m+n}} \cdot \sqrt{\frac{1-\beta^{2i-2}}{4-4\beta^2} + \beta^{2i-2}} \approx \delta_0^{m+n} (q/c)^{\frac{n}{m+n}} \cdot \sqrt{\frac{1}{4-4\beta^2}}.$$

Clearly, the element $(\boldsymbol{v}_i, \boldsymbol{w}_i)$ in $\Lambda_c(\mathbf{A})$ satisfies

$$\boldsymbol{v}_i \cdot \boldsymbol{b} = \boldsymbol{v}_i \cdot (A \cdot \boldsymbol{s} + \boldsymbol{e}) = \langle c \cdot \boldsymbol{w}_i, \boldsymbol{s} \rangle + \langle \boldsymbol{v}_i, \boldsymbol{e} \rangle = \langle (\boldsymbol{v}_i, \boldsymbol{w}_i), (\boldsymbol{e}, c \cdot \boldsymbol{s}) \rangle \bmod q.$$

If, for $1 \leq i \leq n$, $(\boldsymbol{v}_i, \boldsymbol{w}_i)$ is short enough to satisfy $\|(\boldsymbol{v}_i, \boldsymbol{w}_i)\| \cdot \|(\boldsymbol{e}, c \cdot \boldsymbol{s})\| < q/2$, the above equation hold over $\mathbb{Z}$. Then we can recover $\boldsymbol{e}$ and $\boldsymbol{s}$ by solving the system of linear equations. Since, $\|(\boldsymbol{e}, c\boldsymbol{s})\| \approx \sqrt{n \cdot \sigma^2 + c^2 \cdot \|\boldsymbol{s}\|^2}$, condition for attack is following:

$$\delta_0^{n+m} \cdot (q/c)^{\frac{n}{m+n}} \cdot \sqrt{\frac{n \cdot \sigma^2 + c^2 \cdot \|\boldsymbol{s}\|^2}{4-4\beta^2}} < \frac{q}{2}$$

for constant $0 < c \leq q$. To find an optimized constant c, we assume $m = n$. In this case, the size is optimized with $c = \sqrt{n \cdot \sigma^2 / ||s||^2}$. Therefore, final condition to success attack is following:

$$2\delta_0^{4n} \cdot \sigma \cdot \|\boldsymbol{s}\| \cdot \sqrt{n} < q(1-\beta^2).$$

Modified Embedding Attack. One can reduce the LWE problem to unique-SVP problem via Kannan's embedding technique. First, we consider a column lattice

$$\Lambda_q(\mathbf{A}') = \{\boldsymbol{y} \in \mathbb{Z}^{m+1} : \boldsymbol{y} = \mathbf{A}'\boldsymbol{x} \bmod q\} \text{ for } \mathbf{A}' = \begin{pmatrix} 1 & 0 \\ -\boldsymbol{b} & \mathbf{A} \end{pmatrix}.$$

The vector $(1, \boldsymbol{e})^T$ is in lattice $\Lambda_q(\mathbf{A}')$ and its size is approximately $\sigma\sqrt{m}$. If this value is sufficiently smaller than $\lambda_2(\Lambda_q(\mathbf{A}'))$ ($\approx \sqrt{\frac{m}{2\pi e}} q^{(m-n)/m}$), one can find the vector $(1, \boldsymbol{e})^T$ via

some lattice reduction algorithms. In particular, the vector $(1, \boldsymbol{e})^T$ can be found with high probability with the BKZ algorithms in [3], if

$$\frac{\lambda_2(\Lambda_{m+1})}{\lambda_1(\Lambda_{m+1})} = \frac{\lambda_2(\Lambda_q(\mathbf{A}))}{||(1, \boldsymbol{e})||} \geq \tau \cdot \delta_0^m,$$

where $\tau \approx 0.4$. For spLWE case, we can obtain a much larger gap than that of the ordinary attack for LWE. We now consider a scaled lattice $\Lambda_c(\mathbf{B})$ generated by the following matrix:

$$\mathbf{B} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & c\mathbf{I}_n & 0 \\ -\boldsymbol{b} & \mathbf{A} & q\mathbf{I}_m \end{pmatrix}$$

for a constant $0 < c < 1$. The vector $(1, c\boldsymbol{s}, \boldsymbol{e})^T$ is in this lattice and its size is approximately $\sqrt{m \cdot \sigma^2 + c^2\|\boldsymbol{s}\|^2}$. Define a matrix $\mathbf{B}'$ as following,

$$\mathbf{B}' = \begin{pmatrix} c\mathbf{I}_n & 0 \\ \mathbf{A} & q\mathbf{I}_m \end{pmatrix}.$$

We have $\lambda_1(\Lambda_c(\boldsymbol{B})) = \sqrt{m \cdot \sigma^2 + c^2 \cdot ||\boldsymbol{s}||^2}$ and $\lambda_1(\Lambda_c(\mathbf{B}')) = \sqrt{\frac{n+m}{2\pi e}} \cdot \det(\Lambda_c(\mathbf{B}'))^{1/(n+m)} = \sqrt{\frac{n+m}{2\pi e}} \cdot (q^m c^n)^{1/n+m}$. Therefore, it is necessary to find the root Hermite factor δ_0 such that:

$$\sqrt{\frac{n+m}{2\pi e}} \cdot (q^m c^n)^{1/n+m} \geq 0.4 \cdot \delta_0^{n+m} \cdot \sqrt{m \cdot \sigma^2 + c^2\|\boldsymbol{s}\|^2} \tag{3}$$

$$\Leftrightarrow \sqrt{\frac{n+m}{2\pi e \cdot (m \cdot \sigma^2 + c^2\|\boldsymbol{s}\|^2)}} \cdot (q^m c^n)^{1/n+m} \geq 0.4 \cdot \delta_0^{n+m} \tag{4}$$

The left part of inequality above is maximized when $c = \sqrt{n\sigma^2}/\|\boldsymbol{s}\|$, so we have:

$$\sqrt{\frac{1}{2\pi e \cdot \sigma^2}} \left(q^m \cdot \left(\frac{\sigma\sqrt{n}}{\|\boldsymbol{s}\|} \right)^n \right)^{1/(n+m)} \geq 0.4 \cdot \delta_0^{n+m}$$

6.2 Improving Lattice Attacks for spLWE

A time complexity of all attacks suggested in this paper is heavily depend on the dimension of lattices used in the attacks. Therefore, if one can reduce the dimension of lattices, one can obtain a high advantage to solve the LWE problem. In this section, we introduce two techniques to improve lattice-based attacks for spLWE instances. The first thing is a method of ignoring some components of a sparse secret and the other is a method of trading between dimension and modulus, which has been introduced in [13]. For convenience, we denote $T(m)$ as the expected time of solving m-dimensional LWE.

Ignoring Components on Secret Vectors. Most entries of a secret vector $\boldsymbol{s}$ are zero. Therefore, by ignoring some components, one can reduce the dimension of LWE. More precisely, we delete k entries of secret vector $\boldsymbol{s}$ and its corresponding column of $\mathbf{A}$. For convenience, we denote it as $\boldsymbol{s}'$ and $\mathbf{A}'$, respectively. If the deleted components of $\boldsymbol{s}$ are zero, the following equation also hold:

$$\mathbf{A} \cdot \boldsymbol{s} + \boldsymbol{e} = \mathbf{A}' \cdot \boldsymbol{s}' + \boldsymbol{e} \bmod q.$$

The probability P_k that the selected k entries are zero is $\binom{n-\theta}{k}/\binom{n}{k}$. It implies that one can reduce the n-dimensional LWE to $(n-k)$-dimensional LWE with probability P_k. In other words, solving $1/P_k$ instances in $(n-k)$-dimensional LWE, one can expect to solve the n dimension LWE. Hence, in order to guarantee λ bits security, it gives:

$$T(n-k)/P_k \geq 2^\lambda. \tag{5}$$

Modulus Dimension Switching. In [13], they describe a modulus dimension switching technique for LWE instances. Using the corollary 3.4 in [13], for n, q, θ, w that divides n and $\epsilon \in (0, 1/2)$, one can reduce a $\mathsf{LWE}_{n,q,\leq\alpha}$ instances to $\mathsf{LWE}_{n/w,q^w,\leq\beta}$ instances, where β is a constant satisfying $\beta^2 \geq \alpha^2 + (4/\pi)\ln(2n(1+1/\epsilon)) \cdot \theta/q^2 \approx \alpha^2$. Along this reduction, a secret vector $\boldsymbol{s} = (s_1, s_2, \cdots, s_n)$ of $\mathsf{spLWE}_{n,q,\leq\alpha,\rho,\theta}$ is changed to $\boldsymbol{s}'' = (s_1 + qs_2 + \cdots + q^{w-1}s_w, \cdots, s_{n-w+1} + \cdots + q^{w-1}s_n)$ of $\mathsf{spLWE}_{n/w,q^w,\leq\beta,\rho',\theta'}$. Hence, if one can recover the $\boldsymbol{s}''$ by solving $\mathsf{LWE}_{n/w,q^w,\leq\beta,\rho',\theta'}$ instances, one can also reveal the vector $\boldsymbol{s}$. Let t be the number of a set $W = \{s_{wi} | s_{wi} \neq 0, 1 \leq i \leq n/w\}$ and P'_w be the probability of $t = 0$, i.e. P'_w is equal to $\frac{\binom{n-\theta}{n/w}}{\binom{n}{n/w}}$. When t is not zero, the expected size of $\|\boldsymbol{s}''\|$ is $\sqrt{tq^w}$. In that case, applying the attacks in section 4.2, 6.1 and 6.2 to converted n/w-dimensional LWE instances is not a good approach to obtain higher the advantage. Hence, we only consider the case $t = 0$. We can obtain the following conditions to get λ-bit security:

$$T(n/w)/P'_w \geq 2^\lambda. \tag{6}$$

By combining the ignoring k components with modulus dimension switching techniques, we can reach the final condition to obtain the λ-bit security:

$$T((n-k)/w)/(P_k P'_w) \geq 2^\lambda. \tag{7}$$

References

1. M. Ajtai. Generating hard instances of lattice problems. In *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, pages 99–108. ACM, 1996.
2. M. Albrecht, C. Cid, J.-C. Faugere, R. Fitzpatrick, and L. Perret. Algebraic algorithms for lwe problems. 2014.
3. 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.
4. 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.
5. E. Alkim, L. Ducas, T. Pöppelmann, and P. Schwabe. Post-quantum key exchange-a new hope. Technical report, Cryptology ePrint Archive, Report 2015/1092, 201 5. http://eprint. iacr. org, 2015.
6. 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.
7. S. Bai and S. D. Galbraith. An improved compression technique for signatures based on learning with errors. In *Cryptographers' Track at the RSA Conference*, pages 28–47. Springer, 2014.
8. 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.
9. W. Banaszczyk. Inequalities for convex bodies and polar reciprocal lattices inr n. *Discrete & Computational Geometry*, 13(2):217–231, 1995.
10. 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.
11. 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. 2016.
12. 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.
13. 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.
14. J. Buchmann, F. Göpfert, R. Player, and T. Wunderer. On the hardness of lwe with binary error: revisiting the hybrid lattice-reduction and meet-in-the-middle attack. In *International Conference on Cryptology in Africa*, pages 24–43. Springer, 2016.
15. Y. Chen. *Réduction de réseau et sécurité concrete du chiffrement completement homomorphe.* PhD thesis, ENS-Lyon, France, 2013.
16. J. 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. Cryptology ePrint Archive, Report 2016/139, 2016. `http://eprint.iacr.org/2016/139`.

17. Ö. Dagdelen, R. El Bansarkhani, F. Göpfert, T. Güneysu, T. Oder, T. Pöppelmann, A. H. Sánchez, and P. Schwabe. High-speed signatures from standard lattices. In *International Conference on Cryptology and Information Security in Latin America*, pages 84–103. Springer, 2014.
18. A. Daniel, B. Lejla, et al. Initial recommendations of long-term secure post-quantum systems. Technical report, http://pqcrypto.eu.org/docs/initial-recommendations.pdf, 2015.
19. R. De Clercq, S. S. Roy, F. Vercauteren, and I. Verbauwhede. Efficient software implementation of ring-lwe encryption. In *Proceedings of the 2015 Design, Automation & Test in Europe Conference & Exhibition*, pages 339–344. EDA Consortium, 2015.
20. L. De Meyer. Security of lwe-based cryptosystems.
21. 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.
22. L. Ducas, A. Durmus, T. Lepoint, and V. Lyubashevsky. Lattice signatures and bimodal gaussians. In *Advances in Cryptology–CRYPTO 2013*, pages 40–56. Springer, 2013.
23. R. El Bansarkhani and J. Buchmann. Improvement and efficient implementation of a lattice-based signature scheme. In *International Conference on Selected Areas in Cryptography*, pages 48–67. Springer, 2013.
24. S. D. Galbraith. Space-efficient variants of cryptosystems based on learning with errors. *url: https://www. math. auckland. ac. nz/˜ sgal018/compact-LWE. pdf*, 2013.
25. 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.
26. N. Göttert, T. Feller, M. Schneider, J. Buchmann, and S. Huss. On the design of hardware building blocks for modern lattice-based encryption schemes. In *International Workshop on Cryptographic Hardware and Embedded Systems*, pages 512–529. Springer, 2012.
27. T. Güneysu, V. Lyubashevsky, and T. Pöppelmann. Practical lattice-based cryptography: A signature scheme for embedded systems. In *International Workshop on Cryptographic Hardware and Embedded Systems*, pages 530–547. Springer, 2012.
28. Q. Guo, T. Johansson, and P. Stankovski. Coded-bkw: solving lwe using lattice codes. In *Annual Cryptology Conference*, pages 23–42. Springer, 2015.
29. G. Hanrot, X. Pujol, and D. Stehlé. Analyzing blockwise lattice algorithms using dynamical systems. In *Annual Cryptology Conference*, pages 447–464. Springer, 2011.
30. C. Hao, L. Kristin, and E. S. Katherine. Attacks on search rlwe. Cryptology ePrint Archive, Report 2015/971, 2015. `http://eprint.iacr.org/2015/971`.
31. C. Hao, L. Kristin, and E. S. Katherine. Vulnerable galois rlwe families and improved attacks. Cryptology ePrint Archive, Report 2016/193, 2016. `http://eprint.iacr.org/2016/193`.
32. B. Joe, editor. *Algorithmic Number Theory, Third International Symposium, ANTS-III, Portland, Oregon, USA, June 21-25, 1998, Proceedings*, volume 1423 of *Lecture Notes in Computer Science*. Springer, 1998.
33. 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.
34. T. Laarhoven. *Search problems in cryptography*. PhD thesis, PhD thesis, Eindhoven University of Technology, 2015. http://www. thijs. com/docs/phd-final. pdf. 8, 2015.
35. T. Laarhoven. Sieving for closest lattice vectors (with preprocessing). *arXiv preprint arXiv:1607.04789*, 2016.
36. Z. Liu, H. Seo, S. S. Roy, J. Großschädl, H. Kim, and I. Verbauwhede. Efficient ring-lwe encryption on 8-bit avr processors. In *International Workshop on Cryptographic Hardware and Embedded Systems*, pages 663–682. Springer, 2015.
37. 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.
38. V. Lyubashevsky, C. Peikert, and O. Regev. A toolkit for ring-lwe cryptography. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 35–54. Springer, 2013.
39. A. Martin, B. Shi, and D. Léo. A subfield lattice attack on overstretched ntru assumptions: Cryptanalysis of some fhe and graded encoding schemes. Cryptology ePrint Archive, Report 2016/127, 2016. `http://eprint.iacr.org/2016/127`.
40. D. Micciancio and C. Peikert. Trapdoors for lattices: Simpler, tighter, faster, smaller. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 700–718. Springer, 2012.
41. NIST. Technical report, http://www.nist.gov/itl/csd/ct/post-quantum-crypto-workshop-2015.cfm, 2015.
42. NSA. Cryptography today. Technical report, https://www.nsa.gov/ ia/programs/suiteb cryptography/, Also at: https://www.iad.gov/iad/programs/iad-initiatives/cnsa-suite.cfm, 2015.
43. C. Peikert. Lattice cryptography for the internet. In *International Workshop on Post-Quantum Cryptography*, pages 197–219. Springer, 2014.
44. C. Peikert et al. *Decade of Lattice Cryptography*. World Scientific, 2016.
45. C. Peikert and B. Waters. Lossy trapdoor functions and their applications. *SIAM Journal on Computing*, 40(6):1803–1844, 2011.
46. O. Regev. On lattices, learning with errors, random linear codes, and cryptography. In *STOC*, LNCS, pages 84–93, 2005.

47. O. Regev. On lattices, learning with errors, random linear codes, and cryptography. *Journal of the ACM (JACM)*, 56(6):34, 2009.
48. C. P. Richard Lindner. Better key sizes (and attacks) for lwe-based encryption. In A. Kiayias, editor, *CT-RSA*, volume 6558 of *LNCS*, pages 319–339. Springer, 2011.
49. S. S. Roy, F. Vercauteren, N. Mentens, D. D. Chen, and I. Verbauwhede. Compact ring-lwe cryptoprocessor. In *International Workshop on Cryptographic Hardware and Embedded Systems*, pages 371–391. Springer, 2014.
50. C. P. Schnorr. Lattice reduction by random sampling and birthday methods. In *Annual Symposium on Theoretical Aspects of Computer Science*, pages 145–156. Springer, 2003.
51. V. Singh. A practical key exchange for the internet using lattice cryptography. *IACR Cryptology ePrint Archive*, 2015:138, 2015.
52. E. E. Targhi and D. Unruh. Quantum security of the fujisaki-okamoto transform. Technical report, 2015.