A Simple and Efficient CCA-Secure Lattice KEM in the Standard Model

A Simple and Efficient CCA-Secure Lattice KEM in the Standard Model

Xavier Boyen[1], Malika Izabachène[2], Qinyi Li[3] ⋆

[1] QUT, Brisbane, Australia, xb@boyen.org
[2] CEA LIST, Point Courrier 172, 91191 Gif-sur-Yvette Cedex, France
malika.izabachene@cea.fr
[3] Griffith University, Brisbane, Australia, qinyi.li@griffith.edu.au

Abstract. We present, to date, the most efficient public-key encapsulation mechanism from integer lattices in the standard model. Our construction achieves adaptive CCA security through a "direct" chosen-ciphertext security technique without relying on any generic transformation. The security of our construction is based on the standard learning-with-errors assumption. The efficiency of our construction is almost the same as the best known non-adaptive CCA-secure construction.

1 Introduction

Public-key encryption (PKE) is one of the most essential cryptographic primitives that provide data confidentiality. It is the *de facto* requirement that a PKE scheme should be CCA-secure, i.e., secure against adaptive chosen-ciphertext attack for internet applications. In general, the security definitions for PKE involve a game in which the adversary receives a challenge ciphertext ct^* and tries to extract non-trivial information about the underlying message $m*$. With the power of adaptive chosen-ciphertext attack, after receiving ct^*, the adversary can make decryption queries, i.e., adaptively formulating arbitrary ciphertexts that are different from but possibly related to the challenge ciphertext ct^*, and obtain the corresponding plaintexts. Such decryption queries could be helpful to the adversary. For instance, if the adversary can modify the challenge ciphertext ct^* to get a valid (i.e., properly decipherable) ciphertext $\mathsf{ct} \neq \mathsf{ct}^*$ such that m, the message that ct encrypts, is related to m^* in some known way, the adversary can learn something about m^*. This can be done by sending through ct as a decryption query and receiving back the m^*-related message m.

For efficiency reasons, the main use of public-key encryption is as a key encapsulation mechanism (KEM) for exchanging or delivering random session keys. For example, in the paradigm of "hybrid encryption" [13], a KEM scheme is used in conjunction with a data encapsulation mechanism (DEM) scheme which is a basically a symmetric-key cipher: the KEM encrypts a random session key

Acknowledgements: Second author acknowledges the support of the french Programme dInvestissement dAvenir under the national project RISQ.

⋆ Corresponding author

and the DEM uses the random session key to encrypt the actual message. It is known that if both KEM and DEM are CCA-secure, then the hybrid encryption is CCA-secure [13]. While any PKE scheme is also a KEM scheme (by sampling then encrypting a random symmetric keys), KEM schemes may be constructed in more efficient ways.

One of the most common ways of constructing a CCA-secure KEM is to apply the Fujisaki-Okamoto transformation [16] to a weakly secure PKE/KEM scheme (e.g., one with chosen-plaintext or CPA security). This approach has produced many practical KEMs that are CCA-secure in the random oracle model. Alas, the security argument from this approach is merely heuristic since random oracles are fictions that cannot be instantiated in the real world. There have been plentiful works on building efficient PKE/KEM schemes from various of number-theoretic assumptions, e.g., [13,9,19,17], without using random oracles. Since these schemes are insecure against quantum adversaries, it is desirable to find "quantum-safe" alternatives, e.g., lattice-based constructions.

There are three current approaches for constructing CCA-secure PKE/KEM from lattices in the standard model, i.e., without using random oracles. The first one is to apply the BCHK transformation [9] to tag-based encryption (TBE) or identity-based encryption (IBE). This approach introduces noticeable extra overheads in computation and ciphertext size to the underlying TBE/IBE scheme. The second one is to resort to lossy trapdoor functions (LTFs) [31], as in the construction from [10]. One of the main issue with this approach is that the known lattice-based lossy trapdoor functions (e.g., [31,8,5]) require relatively large parameters and rely on strong lattice assumptions (e.g., LWE with large (but still polynomial) modulus-to-noise ratio). For example, the recent simple construction of CCA-secure KEM scheme from [10] uses a stronger LWE assumption than the CCA-secure PKE/KEM obtained by applying BCHK transform to the identity/tag-based encryption schemes in [1,24]. The third approach is to instantiate the Naor-Yung paradigm [26,14,33] using a non-interactive zero-knowledge (NIZK) proof system from lattices, such as [30], but at the moment this approach is not even remotely practical.

In this paper, we propose a fourth approach, for constructing an efficient KEM in the standard model from standard lattice assumptions, without appealing to generic transformation, LTFs or NIZKs.

1.1 Our Approach

At a high level, we adopt the "direct chosen-ciphertext" approach from [11], previously used in pairing-based CCA-secure PKE constructions ([11,20]), and adapt it for use with an TBE/IBE from the learning-with-errors (LWE) problem, e.g., [1,24]. However, contrarily to pairing-based constructions, the ciphertexts in lattice-based constructions often contain noise terms. Varying the noise terms within a small interval results in slightly different ciphertexts which are still valid and properly decrypt to the same message. This is because those noise terms are removed by the error-correction mechanism of the decryption algorithms as

long as they are "small" enough. While those noise terms are crucial for security, they also impede the making of the ciphertexts non-malleable for CCA security.

Let's take the tag-based PKE scheme from [24] as an example. The ciphertext of the scheme is

$$\mathbf{c}_0^t = \mathbf{s}^t\mathbf{A} + \mathbf{e}_0^t \quad ; \quad \mathbf{c}_1^t = \mathbf{s}^t(\mathbf{A}_1 + H(\mathsf{id})\mathbf{G}) + \mathbf{e}_1^t$$

where matrices $\mathbf{A}$, $\mathbf{A}_1 \in \mathbb{Z}_q^{n\times m}$ are public keys, H is a full-rank difference encoding (FRD) which is either injective or with second pre-image resistance, and $\mathbf{G} \in \mathbb{Z}_q^{n\times m}$ is a "gadget" matrix. (We refer to [1] for the details of FRD and to [24] for the detail of $\mathbf{G}$.) Here we ignore how the actual message is hidden for simplicity. Notice that this ciphertext essentially contains many LWE samples and $\mathbf{e}_0$, $\mathbf{e}_1$ are the LWE noise terms. To prove security [1], the simulator embeds the challenge tag $H(\mathsf{id}^*)$ in $\mathbf{A}_1$ such that all ciphertexts with $H(\mathsf{id}) \neq H(\mathsf{id}^*)$ can be decrypted, except for the challenge ciphertext whose tag is $H(\mathsf{id}^*)$.

Drawing inspiration from [11,20,35], we would replace the identity id by $\mathbf{c}_0$ and obtain

$$\mathbf{c}_0^t = \mathbf{s}^t\mathbf{A} + \mathbf{e}_0^t \quad ; \quad \mathbf{c}_1^t = \mathbf{s}^t(\mathbf{A}_1 + H(\mathbf{c}_0)\mathbf{G}) + \mathbf{e}_1^t$$

The tag $H(\mathbf{c}_0)$ binds $\mathbf{s}$ and $\mathbf{e}_0$. Given a challenge ciphertext $(\mathbf{c}_0^*, \mathbf{c}_1^*)$, if the attacker modifies $\mathbf{c}_0^*$ or $\mathbf{c}_1^*$ by changing $\mathbf{s}$, the tag $H(\mathbf{c}_0^*)$ changes, and, by the same argument as in [1], decryption queries on such a modified ciphertext can be answered. However, this does not make the ciphertext non-malleable because adding a small error $\mathbf{e}$ to $\mathbf{e}_1$ results in a ciphertext that decrypts to the same message with high probability. In fact, a recent lattice-based public-key encryption scheme [15] does not appear to live up to its claim of CCA security, precisely for this reason.

Therefore, an additional mechanism is needed in order to prevent the adversary from modifying $\mathbf{e}_1$ while keeping the ciphertext valid. Our idea is simple. We add a hash of $\mathbf{e}_1$ to the ciphertext, where hashing is from the short integer solution (SIS) problem, i.e., $\mathbf{c}_2 = \mathbf{U}\mathbf{e}_1$ where $\mathbf{U} \in \mathbb{Z}_q^{n\times m}$ is a random matrix. We also input $\mathbf{c}_2$ to the FRD function. This gives a ciphertext

$$\mathbf{c}_0^t = \mathbf{s}^t\mathbf{A} + \mathbf{e}_0^t \quad ; \quad \mathbf{c}_1^t = \mathbf{s}^t(\mathbf{A}_1 + H(\mathbf{c}_0, \mathbf{c}_2)\mathbf{G}) + \mathbf{e}_1^t \quad ; \quad \mathbf{c}_2 = \mathbf{U}\mathbf{e}_1$$

This provides non-malleability, since modifying $\mathbf{e}_1$ into a "small" $\mathbf{e}_2'$ without changing $\mathbf{c}_2$ is infeasible, without also producing a solution to the SIS problem for the random matrix $\mathbf{U}$, given by $\mathbf{U}(\mathbf{e}_1 - \mathbf{e}_1') = \mathbf{0}$.

Lastly, we observe that it is enough to use $H(\mathbf{c}_0, \mathbf{c}_2)$, a hash value which can be as short as the security parameter, to make the ciphertext non-malleable, instead of $\mathbf{c}_2$. Therefore, the final ciphertext of our construction consists of three elements:

$$\mathbf{c}_0^t = \mathbf{s}^t\mathbf{A} + \mathbf{e}_0^t \quad ; \quad \mathbf{c}_1^t = \mathbf{s}^t(\mathbf{A}_1 + H(\mathbf{c}_0, \mathbf{U}\mathbf{e}_1)\mathbf{G}) + \mathbf{e}_1^t \quad ; \quad H(\mathbf{c}_0, \mathbf{U}\mathbf{e}_1)$$

which can be of the same size as the *non-adaptive* CCA1-secure PKE scheme proposed in [24].

The security rationale behind this simple idea is the duality between the lattice problems LWE and SIS [22,21]. LWE and SIS are "syntactically equivalent" in the sense that adding an SIS hash of the noise term of the given LWE samples is essentially equivalent to providing a redundant description of the LWE problem, and therefore would not make the LWE problem any significantly easier.

1.2 Comparison and Related Work

Our construction achieves *adaptive* CCA security (a.k.a. CCA2 security) while being almost as concise as the best-known *non-adaptive* CCA1-secure scheme without random oracles from [24] (dubbed "MP12" henceforward). Public key size is essentially the same except for one extra public matrix $\mathbf{U}$[4]. Our ciphertexts are as short as those of CCA1-secure MP12 under the same message encoding, hence shorter than any CCA2-secure ciphertext that could be obtained by upgrading MP12 through the BCHK transform (yielding "MP12-BCHK").

In terms of encapsulation (encryption) and decapsulation (decryption) speed, even though our construction incurs one extra matrix-vector multiplication $\mathbf{U}\mathbf{e}_1$, performance is very comparable to MP12-BCHK with commitments and message authentication codes, and better than the MP12-BCHK with one-time signatures (recall that the generic BCHK transform works in either of those ways). In terms of computational assumptions, we rely on almost the same LWE assumption as the one used in MP12. Crucially, the LWE noise-to-modulus ratio in both schemes is the same. The only difference is that to prove security, our construction requires a slightly larger number of LWE samples than needed by MP12, as these extra LWE samples are used to simulate $\mathbf{U}\mathbf{e}_1$. This, however, is a mostly academic distinction with limited bearing on security, as the hardness of the LWE problem is not sensitive to the number of samples.

Duong et al. [15] recently proposed a lattice-based PKE scheme with equality test. However, the scheme is not CCA-secure as the ciphertext is malleable, as we noted before. It would be interesting to work out if our technique is applicable to leverage their scheme to achieve CCA security without losing efficiency.

Zhang et al. [35] recently proposed another lattice-based PKE scheme with claimed CCA security. One similarity with ours is that they require the random session key to be encoded in the LWE secret; we elected to do the same to make our construction more compact. It should be noted, however, that the security proof of [35] seems incomplete, although no attack seems to have been found.

Table 1 below gives a comparison amongst efficient public-key encryption and encapsulation schemes from lattices in the standard model. To be generous, we assume that all schemes use the same efficient way of encoding the message into the normal-form LWE secret vectors, with the caveat that the resulting secrets post-encoding must retain sufficiently high entropy. (The original MP12 encodes messages into the LWE noise vectors, making its ciphertexts larger.) In

[4] A common practical implementation heuristic would be to expand $\mathbf{U}$ from a public seed using a secure pseudorandom number generator; for the security reduction $\mathbf{U}$ needs to be truly random.

Table 1, the pub, com and tag overheads affecting MP12-MAC, refer to public parameters, weak commitment and MAC tag borne by the "MAC" version of the BCHK transform. Likewise, vk and sig in MP12-SIG are the public key and signature in a strongly unforgeable one-time signature scheme incurred by the "SIG" version of the BCHK transform. Their size will generally be much larger than the security parameter, which we conventionally write λ. Both MP12 and our work have a ciphertext overhead component of size λ: in MP12 it is a random tag; in our case it is a second-preimage-resistant hash function. Given a modulus q and dimension n (suitably determined from the security parameter λ), the hardness of the LWE problem is mostly determined by the noise-to-modulus ratio parameter α. The smaller $\alpha > 0$ is, the easier the LWE problem becomes. As noted above, α is essentially the same across the relevant constructions.

Table 1: Comparison for PKE schemes

	param. α	$\lvert\mathsf{pk}\rvert$	$\lvert\mathsf{ct}\rvert$	Security	Type
MP12 [24]	$\tilde{O}(1/n)$	$2(n\log q)^2$	$3n\log q + \lambda$	CCA1	PKE
MP12-MAC [24,9]	$\tilde{O}(1/n)$	$2(n\log q)^2 + \lvert\mathsf{pub}\rvert$	$3n\log q + \lvert\mathsf{tag}\rvert + \lvert\mathsf{com}\rvert$	CCA	PKE
MP12-SIG [24,9]	$\tilde{O}(1/n)$	$2(n\log q)^2$	$3n\log q + \lvert\mathsf{sig}\rvert + \lvert\mathsf{vk}\rvert$	CCA	PKE
This work	$\tilde{O}(1/n)$	$3(n\log q)^2$	$3n\log q + \lambda$	CCA	KEM

2 Preliminaries

We denote the security parameter by λ. We use bold lowercase letters (e.g. $\mathbf{a}$) to denote vectors and bold capital letters (e.g. $\mathbf{A}$) to denote matrices. For a positive integer $q \geq 2$, let $\mathbb{Z}_q$ be the ring of integers modulo q. We denote the group of $n \times m$ matrices in $\mathbb{Z}_q$ by $\mathbb{Z}_q^{n\times m}$. Vectors are treated as column vectors. The transpose of a vector $\mathbf{a}$ is denoted by $\mathbf{a}^\top$. For $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ and $\mathbf{B} \in \mathbb{Z}_q^{n\times m'}$, let $[\mathbf{A}|\mathbf{B}] \in \mathbb{Z}_q^{n\times(m+m')}$ be the concatenation of $\mathbf{A}$ and $\mathbf{B}$. We denote by $\|\mathbf{x}\|$ the ℓ_2 norm of a vector $\mathbf{x}$. $\|\mathbf{X}\|$ denotes the ℓ_2 length of the longest column of $\mathbf{R}$. Let $\mathbf{x} = (x_1, x_2, ..., x_n)$ be s vector over $\mathbb{Z}_q^n$, $\|\mathbf{x}\| = \max(|x_1|, |x_2|, ..., |x_n|)$. Let X and Y be two random variables taking values in some finite set Ω. Their statistical distance, denoted $\Delta(X, Y)$, is $\Delta(X, Y) = \frac{1}{2}\sum_{s\in\Omega} |\Pr[X = s] - \Pr[Y = s]|$.

The following lemma will be useful in our security proofs.

Lemma 1 (Lemma 1 of [34]). *Let X_1, X_2, B be events defined in some probability distribution, and suppose that $X_1 \wedge \neg B \Leftrightarrow X_2 \wedge \neg B$. Then $|\Pr[X_1] - \Pr[X_2]| \leq \Pr[B]$.*

2.1 Lattices, Discrete Gaussians, and Trapdoors

Definition 1 (Random Integer Lattice). *For a positive integer q (later to be prime), a matrix $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ and a vector $\mathbf{u} \in \mathbb{Z}_q^n$, define the m-dimensional*

full-rank integer lattices

$$\Lambda(\mathbf{A}) = \{\mathbf{y} \in \mathbb{Z}^m \ s.t. \ \exists \ \mathbf{x} \in \mathbb{Z}_q^n \ where \ \mathbf{x}^t\mathbf{A} = \mathbf{y} \pmod q\}$$

$$\Lambda^{\perp}(\mathbf{A}) := \{\mathbf{y} \in \mathbb{Z}^m \ s.t. \ \mathbf{A}\mathbf{y} = \mathbf{0} \pmod q\}$$

In fact, up to a scaling factor of q, $\Lambda(\mathbf{A})$ and $\Lambda^{\perp}(\mathbf{A})$ are dual to each other.

Definition 2. *Let $m \in \mathbb{Z}_{>0}$ be a positive integer and $\Lambda \subset \mathbb{Z}^m$. For any real vector $\mathbf{c} \in \mathbb{R}^m$ and positive parameter $\sigma \in \mathbb{R}_{>0}$, $\forall \mathbf{y} \in \Lambda$, the discrete Gaussian distribution over Λ with center $\mathbf{c}$ and parameter σ is denoted by $D_{\Lambda,\sigma,\mathbf{c}} = \rho_{\sigma,\mathbf{c}}(\mathbf{y})/\rho_{\sigma,\mathbf{c}}(\Lambda)$ where $\rho_{\sigma,\mathbf{c}}(\mathbf{x}) = \exp\left(-\pi\|\mathbf{x}-\mathbf{c}\|^2/\sigma^2\right)$ is the Gaussian function and $\rho_{\sigma,\mathbf{c}}(\Lambda) = \sum_{\mathbf{x}\in\Lambda}\rho_{\sigma,\mathbf{c}}(\mathbf{x})$. For notational convenience, $\rho_{\sigma,\mathbf{0}}$ and $D_{\Lambda,\sigma,\mathbf{0}}$ are abbreviated as ρ_σ and $D_{\Lambda,\sigma}$.*

Lemma 2 (special case of Lemma 4.4 of [25]). *For $\mathbf{x} \leftarrow D_{\mathbb{Z}^m,s}$, $\Pr[\|\mathbf{x}\| > s\sqrt{m}] < 1 - 2^{-\Omega(m)}$.*

Lemma 3 (Proposition 5.1 of [18]). *Let $q \geq 2$. For all but a $2q^{-n}$ fraction of all $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ and for any $s \geq \omega(\sqrt{\log n})$, the distribution of $\mathbf{A}\mathbf{e} \bmod q$ is statistically close to uniform over $\mathbb{Z}_q^n$, where $\mathbf{e} \sim D_{\mathbb{Z}^m,s}$.*

We will use the super-increasing vector $\mathbf{g}^t = (1, 2, 4, \ldots, 2^{k-1})$, for $k = \lceil \log_2 q \rceil$ and extend it to form a "gadget" matrix $\mathbf{G} = diag(\mathbf{g}^t, \ldots, \mathbf{g}^t) \in \mathbb{Z}_q^{n\times nk}$ as in [24]. Here we use a base 2 but other choices of base can be used. We formulate the following lemma which is directly derived from the Theorem 4.1 and Theorem 5.4 and of [24].

Lemma 4. *Let $w = n\lceil \log q \rceil$. Let $\mathbf{F} = [\mathbf{A}|\mathbf{A}\mathbf{R} + \mathbf{H}\mathbf{G}]$ where $\mathbf{R} \in \mathbb{Z}^{m\times w}$, $\mathbf{H} \in \mathbb{Z}_q^{n\times n}$ is invertible in $\mathbb{Z}_q$, and $\mathbf{G} \in \mathbb{Z}_q^{n\times w}$ is the gadget matrix. Given $\mathbf{b}^t = \mathbf{s}^t\mathbf{F} + \mathbf{e}^t$ where $\mathbf{e}^t = [\mathbf{e}_0^t|\mathbf{e}_1^t]$, there exists a p.p.t algorithm $\mathsf{Invert}(\mathbf{R}, \mathbf{F}, \mathbf{b})$ that outputs $\mathbf{s}$ and $\mathbf{e}$ when $\|\mathbf{e}_1^t - \mathbf{e}_0^t\mathbf{R}\|_\infty < q/4$.*

2.2 Hardness Assumptions

Definition 3 (Short-Integer-Solution Problem). *Let λ be the security parameter, $n = n(\lambda)$, $m = m(\lambda)$, $q = q(\lambda)$ and $\beta = \beta(\lambda)$. The advantage of an algorithm $\mathcal{A}$ that solves the problem $\mathsf{SIS}_{n,m,q,\beta}$, denoted by $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{SIS}_{n,m,q,\beta}}(\lambda)$, is defined as $\Pr[\mathcal{A}(\mathbf{A}, \beta) \rightarrow \mathbf{e} \neq \mathbf{0} \ : \ \mathbf{A}\mathbf{e} = 0 \bmod q \ \wedge \ \|\mathbf{e}\| \leq \beta]$ where $\mathbf{A} \leftarrow \mathbb{Z}_q^{n\times m}$. We say $\mathsf{SIS}_{n,m,q,\beta}$ is hard if for all p.p.t algorithms $\mathcal{A}$, $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{SIS}_{n,m,q,\beta}}(\lambda) \leq \mathsf{negl}(\lambda)$.*

A series of works show that solving the $\mathsf{SIS}_{n,m,q,\beta}$ problem is as hard as approximating classic lattice problems, e.g., GapSVP and SIVP, for some approximation factor $\gamma = \beta \cdot \mathsf{poly}(n)$. We refer to [29] and the references therein for further details.

Definition 4 (Learning-With-Errors Problem). *Let λ be the security parameter, $n = n(\lambda)$, $m = m(\lambda)$, $q = q(\lambda)$ and an error distribution $\chi = \chi(n)$ over $\mathbb{Z}_q$. The advantage of a p.p.t adversary $\mathcal{A}$ for the learning with errors problem $\mathsf{LWE}_{n,m,q,\chi}$, denoted by $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{LWE}_{n,m,q,\chi}}(\lambda)$, is defined as*

$$\left|\Pr[\mathcal{A}(\mathbf{A}, \mathbf{s}^t\mathbf{A} + \mathbf{e}^t) = 1] - \Pr[\mathcal{A}(\mathbf{A}, \mathbf{b}^t) = 1]\right|$$

where $\mathbf{A} \leftarrow \mathbb{Z}_q^{n\times m}$, $\mathbf{s} \leftarrow \mathbb{Z}_q^n$, $\mathbf{e} \leftarrow \chi^m$. The $\mathsf{LWE}_{n,m,q,\chi}$ problem is hard if $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{LWE}_{n,m,q,\chi}}(\lambda) \leq \mathsf{negl}(\lambda)$ for all p.p.t adversary $\mathcal{A}$.

Regev [32] shows that, for $\alpha q > \sqrt{n}$, LWE is as hard as approximating some traditional worst-case lattice problems, e.g., SIVP. We refer to [32,29] for details.

We prove the CCA security of our KEM scheme based on a variant of the LWE problem with a uniformly distributed SIS "hint", that we call the $\mathsf{SISnLWE}$ problem ("n" stands for "normal form"). It can also be seen as another variant of the Extended LWE problem [27,4,12,3,6] which has been studied in several different contexts. In Section 4.1, we prove that $\mathsf{SISnLWE}$ problem is hard as the LWE problem.

Definition 5 (Normal-form LWE with SIS hint). *Let λ be the security parameter, $n = n(\lambda)$, $m = m(\lambda)$, $q = q(\lambda)$ and an error distribution $\chi = \chi(n)$ over $\mathbb{Z}_q$. The advantage of a p.p.t. adversary $\mathcal{A}$ for the learning with errors problem $\mathsf{SISnLWE}_{n,m,q,\chi}$, denoted by $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{SISnLWE}_{n,m,q,\chi}}(\lambda)$, is defined as*

$$\left|\Pr[\mathcal{A}(\mathbf{A}, \mathbf{s}^t\mathbf{A} + \mathbf{e}^t, \mathbf{Ze}, \mathbf{Z}) = 1] - \Pr[\mathcal{A}(\mathbf{A}, \mathbf{b}^t, \mathbf{Ze}, \mathbf{Z}) = 1]\right|$$

where $\mathbf{A}, \mathbf{Z} \leftarrow \mathbb{Z}_q^{n\times m}$, $\mathbf{s} \leftarrow \chi^n$, $\mathbf{e} \leftarrow \chi^m$. The $\mathsf{SISnLWE}_{n,m,q,\chi}$ problem is hard if $\mathsf{Adv}_{\mathcal{A}}^{\mathsf{SISnLWE}_{n,m,q,\chi}}(\lambda) \leq \mathsf{negl}(\lambda)$ for all p.p.t adversary $\mathcal{A}$.

We note that unlike the Extended LWE problems studied in [4,3,6] where $\mathbf{Z}$ is of low-norm, here $\mathbf{Z}$ is chosen uniformly at random which makes arguing the hardness of this new $\mathsf{SISnLWE}$ problem a lot easier.

Definition 6 (Second Pre-Image Collision Resistance). *Let λ be the security parameter. Let $H : \{0,1\}^* \to \{0,1\}^\ell$ be a hash function. We say that H is second-pre-image resistant if for all p.p.t algorithms $\mathcal{A}$, the advantage,*

$$\mathsf{Adv}_{\mathcal{A}}^{\mathsf{coll}}(\lambda) = \Pr[\mathcal{A}(H, x) \to x' \neq x \ : H(x) = H(x')]$$

where $x \leftarrow \{0,1\}^*$ and $x' \in \{0,1\}^*$ is negligible in λ.

We note that the notion of second pre-image collision resistance (or just second pre-image resistance) is weaker than the notion of collision resistance.

2.3 Public-Key Encapsulation

A public-key encapsulation (KEM) scheme $\Pi = (\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$ with key space $\mathcal{K}_\lambda$ consists of three polynomial-time algorithms. The key generation algorithm $\mathsf{KeyGen}(1^\lambda)$ generates a public key pk and a private key sk. The randomised

key encapsulation algorithm $\mathsf{Enc}(\mathsf{pk})$ generates a session key $K \in \mathcal{K}_\lambda$ and a ciphertext ct. The decapsulation algorithm $\mathsf{Dec}(\mathsf{pk}, \mathsf{sk}, \mathsf{ct})$ returns the session key K or the error symbol $\perp$. The correctness of a KEM scheme requires that for all $\lambda \in \mathbb{N}$, and all $(K, \mathsf{ct}) \leftarrow \mathsf{Enc}(\mathsf{pk})$,

$$\Pr[\mathsf{Dec}(\mathsf{pk}, \mathsf{sk}, \mathsf{ct}) = K] \geq 1 - \mathsf{negl}(\lambda)$$

where the probability is taken over the random coins of KeyGen and Enc.

We recall the adaptive chosen-ciphertext security of KEM. The IND-CCA security of a KEM scheme Π with session key space $\mathcal{K}_\lambda$ is defined by the following security game. The challenger $\mathcal{C}$ runs $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KeyGen}(1^\lambda)$, chooses a random coin $\mu \leftarrow_\$ \{0,1\}$, samples $K_0^* \leftarrow_\$ \mathcal{K}_\lambda$, and computes $(K_1^*, \mathsf{ct}^*) \leftarrow \mathsf{Enc}(\mathsf{pk})$. Then $\mathcal{C}$ passes $(\mathsf{pk}, K_\mu^*, \mathsf{ct}^*)$ through to the adversary. The adversary launches adaptive chosen-ciphertext attacks: It repeatedly chooses any ciphertext $\mathsf{ct} \neq \mathsf{ct}^*$ and sends it over to $\mathcal{C}$ and $\mathcal{C}$ returns $\mathsf{Dec}(\mathsf{pk}, \mathsf{sk}, \mathsf{ct})$. Finally, $\mathcal{A}$ outputs μ' and wins if $\mu' = \mu$. We define $\mathcal{A}$'s advantage in the above security game as

$$\mathsf{Adv}^{\mathsf{ind-cca}}_{\mathcal{A},\Pi}(\lambda) = |\Pr[\mu' = \mu] - 1/2|.$$

We say Π is IND-CCA-secure if $\mathsf{Adv}^{\mathsf{ind-cca}}_{\mathcal{A},\Pi}(\lambda)$ is negligible in λ.

3 CCA-Secure KEM

We use the $\mathsf{SISnLWE}_{n,m,q,D_{\mathbb{Z},\alpha q}}$ problem for security where q is prime, $D_{\mathbb{Z},\alpha q}$ denotes the discrete Gaussian distribution with parameter αq, and $m \geq 2n\lceil \log q \rceil$. The following specifications are shared across instances of our scheme.

1. A full-rank difference encoding [1] $\mathsf{FRD} : \mathbb{Z}_q^n \to \mathsf{GL}(n, q)$ where $\mathsf{GL}(n, q)$ denotes the set of modulo-q invertible matrices in $\mathbb{Z}_q^{n \times n}$.
2. $\mathbf{G} \in \mathbb{Z}_q^{n \times w}$ for $w = n\lceil \log q \rceil$ is the gadget matrix originally defined in [24].
3. The LWE error rate α such that $1/\alpha = 8 \cdot O(w) \cdot \omega(\sqrt{\log n})$.
4. A second-pre-image resistant hash function $H : \{0,1\}^* \to \{0,1\}^\lambda \setminus \{0^\lambda\}$ where 0^λ is the length-λ string with all 0's. W.l.o.g, we assume there is an efficient injective encoding that encodes the outputs of H to elements in $\mathbb{Z}_q^n$.

- $\mathsf{KeyGen}(1^\lambda)$ The key generation algorithm does:
  1. Sample $\mathbf{A} \leftarrow \mathbb{Z}_q^{n \times m}$, $\mathbf{R} \leftarrow D^{m \times w}_{\mathbb{Z}, \omega(\sqrt{\log n})}$ and set $\mathbf{A}_1 \leftarrow \mathbf{AR}$.
  2. Sample $\mathbf{U} \leftarrow \mathbb{Z}_q^{n \times w}$.
  3. Return the public key $\mathsf{pk} = (\mathbf{A}, \mathbf{A}_1, \mathbf{U})$
- $\mathsf{Enc}(\mathsf{pk})$ The key encapsulation algorithm does:
  1. Sample a session key $\mathbf{k} \leftarrow \{0,1\}^n$, $\bar{\mathbf{s}} \leftarrow \mathbb{Z}_q^n$; Set $\mathbf{s} \leftarrow \mathbf{k}\lfloor q/2 \rfloor + \bar{\mathbf{s}}$.
  2. Sample $\mathbf{e}_0 \leftarrow D^m_{\mathbb{Z},\alpha q}$, $\mathbf{e}_1 \leftarrow D^w_{\mathbb{Z},s}$ where $s^2 = (\|\mathbf{e}_0\|^2 + m(\alpha q)^2) \cdot \omega(\sqrt{\log n})^2$.
  3. Compute $\mathbf{c}_0^t \leftarrow \mathbf{s}^t\mathbf{A} + \mathbf{e}_0^t$, and $\mathbf{c}_2 \leftarrow \mathbf{U}\mathbf{e}_1$.
  4. Compute $\mathbf{c}_1^t \leftarrow \mathbf{s}^t(\mathbf{A}_1 + \mathsf{FRD}(\mathbf{t})\mathbf{G}) + \mathbf{e}_1^t$ where $\mathbf{t} = H(\mathbf{c}_0, \mathbf{c}_2)$ is encoded as an element in $\mathbb{Z}_q^n$ before being sent to $\mathsf{FRD}(\cdot)$.
  5. Return the ciphertext $\mathsf{ct} = (\mathbf{c}_0, \mathbf{c}_1, \mathbf{t})$ and session key $\mathbf{k}$.

– $\mathsf{Dec}(\mathsf{pk}, \mathsf{sk}, \mathsf{ct})$ The decapsulation algorithm does:
  1. Parse $\mathsf{ct} = (\mathbf{c}_0, \mathbf{c}_1, \mathbf{t})$; Output $\perp$ if ct doesn't parse.
  2. Call $\mathsf{Invert}(\mathbf{R}, \mathbf{F}, [\mathbf{c}_0^t | \mathbf{c}_1^t])$ where $\mathbf{F} = [\mathbf{A} | \mathbf{A}_1 + \mathsf{FRD}(\mathbf{t})\mathbf{G}]$ to get $\mathbf{e}_0$ and $\mathbf{e}_1$.
  3. If $\|\mathbf{e}_0\| > \alpha q\sqrt{m}$ or $\|\mathbf{e}_1\| > \alpha q\sqrt{2mw} \cdot \omega(\sqrt{\log n})$, output $\perp$.
  4. If $H(\mathbf{c}_0, \mathbf{U}\mathbf{e}_1) \neq \mathbf{t}$, output $\perp$.
  5. Let $\mathbf{s}[i]$ be the i-th coordinate of $\mathbf{s}$; Set $\mathbf{k}[i] \leftarrow 0$ if $\mathbf{s}[i]$ is closer to 0 or $\mathbf{k}[i] \leftarrow 1$ if $\mathbf{s}[i]$ is closer to $q/2$.
  6. Output $\mathbf{k}$ if $\|\mathbf{s} - \mathbf{k}\| \leq \alpha q\sqrt{n}$; Otherwise output $\perp$.

The ciphertext consists of three elements, i.e., $\mathbf{c}_0 \in \mathbb{Z}_q^m$, $\mathbf{c}_1 \in \mathbb{Z}_q^w$, and $\mathbf{t} \in \{0,1\}^\lambda$.

Decapsulation correctness. It is sufficient to show that for a correctly generated ciphertext, the algorithm $\mathsf{Invert}(\mathbf{R}, \mathbf{F}, [\mathbf{c}_0^t | \mathbf{c}_1^t])$ will output $\mathbf{s}$ with overwhelming probability. Let $\mathbf{e}^t = [\mathbf{e}_0^t | \mathbf{e}_1^t]$. By Lemma 2, we have $\|\mathbf{e}_1\| \leq s\sqrt{w}$, $\|\mathbf{e}_0\| \leq \alpha q\sqrt{m}$, and $\|\mathbf{R}\| \leq \sqrt{m} \cdot \omega(\sqrt{\log n})$ except with negligible probability. By Lemma 4,

$$\begin{aligned}
\left\|\mathbf{e}_1^t - \mathbf{e}_0^t\mathbf{R}\right\|_\infty &\leq \left\|\mathbf{e}_1^t - \mathbf{e}_0^t\mathbf{R}\right\| \leq \left\|\mathbf{e}_1^t\right\| + \left\|\mathbf{e}_0^t\mathbf{R}\right\| \\
&\leq 2\alpha q \cdot O(\sqrt{w}) \cdot \omega(\sqrt{\log n}) \cdot \sqrt{3w} \\
&\leq 2\alpha q \cdot O(w) \cdot \omega(\sqrt{\log n}) \\
&\leq q/4
\end{aligned}$$

holds with overwhelming probability. So $\mathsf{Invert}(\mathbf{R}, \mathbf{F}, [\mathbf{c}_0^t | \mathbf{c}_1^t])$ correctly recovers $\mathbf{e}$ and $\mathbf{s}$ with overwhelming probability. Finally, since $\bar{\mathbf{s}} \sim D_{\mathbb{Z},\alpha q}^n$, we have $\|\bar{\mathbf{s}}\|_\infty \leq q/4$ (i.e., $|\bar{\mathbf{s}}[i]| < q/4$ for all i) with overwhelming probability. So, $\mathbf{k}$ can be recovered from $\mathbf{s} = \mathbf{k}\lfloor q/2 \rfloor + \bar{\mathbf{s}}$ with overwhelming probability.

4 Security Analysis

The security of our KEM scheme is based on a variant of LWE problem we call $\mathsf{SISnLWE}$ (normal-form LWE with a SIS hint). In this section, we first show that $\mathsf{SISnLWE}$ is as hard as the standard LWE problem. Then we give the security proof of the KEM scheme.

4.1 The SISnLWE Problem

Let q, n, m, $m' \geq 2$ be integers where $m = O(n \log q)$ and $m' = m + n$. Let χ be a noise distribution on $\mathbb{Z}_q$. The $\mathsf{SISnLWE}_{n,m,q,\chi}$ problem gives as challenge

$$(\mathbf{A}, \mathbf{b}^t, \mathbf{z} = \mathbf{Z}\mathbf{e} \bmod q, \mathbf{Z})$$

where $\mathbf{A} \leftarrow \mathbb{Z}_q^{n \times m}$, $\mathbf{b} \in \mathbb{Z}_q^m$, $\mathbf{Z} \leftarrow \mathbb{Z}_q^{n \times m}$ and $\mathbf{e} \leftarrow \chi^m$. It asks to tell if there exists a vector $\mathbf{s} \sim \chi^n$ such that $\mathbf{b}^t = \mathbf{s}^t\mathbf{A} + \mathbf{e}^t \pmod q$ or if $\mathbf{b} \in \mathbb{Z}_q^n$ is random.

Instead of reducing the problem to standard LWE problem directly, we reduce the $\mathsf{SISnLWE}$ problem to the equivalent knapsack form of LWE, defined by Micciancio and Mol [23]. The Knapsack LWE problem, $\mathsf{KLWE}_{m'-n,m',q,\chi}$, gives

$(\mathbf{B}, \mathbf{c})$ where $\mathbf{B} \leftarrow \mathbb{Z}_q^{(m'-n)\times m'}$, $\mathbf{c} \in \mathbb{Z}_q^{m'-n}$ and asks to determine if $\mathbf{c} = \mathbf{Bx} \pmod q$ for some $\mathbf{x} \leftarrow \chi^{m'}$ or if $\mathbf{c}$ is uniformly random. The hardness of the knapsack-form LWE problem is in turn implied by the hardness of the standard LWE problem. This is shown by Micciancio and Mol (Lemma 10, [23]). We also note that the knapsack-form LWE problem has been used to prove the hardness of a version of the Extended LWE problem in [4].

We first define an intermediate problem $\mathsf{SISLWE}_{n,m,q,\chi}$ (LWE with a SIS hint): Given $(\mathbf{A}, \mathbf{b}^t, \mathbf{Ze}, \mathbf{Z})$ where $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$, $\mathbf{b} \in \mathbb{Z}_q^m$, $\mathbf{Z} \in \mathbb{Z}_q^{n\times m}$ and $\mathbf{e} \leftarrow \chi^m$, decide if $\mathbf{b}^t = \mathbf{s}^t\mathbf{A} + \mathbf{e}^t$ for some $\mathbf{s} \leftarrow \mathbb{Z}_q^n$ or if $\mathbf{b}$ is random. Note the only difference between the $\mathsf{SISnLWE}_{n,m,q,\chi}$ problem and the $\mathsf{SISLWE}_{n,m,q,\chi}$ problem is that in the former one $\mathbf{s}$ are sampled from the noise distribution χ^n and the in the latter one $\mathbf{s} \leftarrow \mathbb{Z}_q^n$.

Lemma 5. $\mathsf{SISLWE}_{n,q,m',\chi}$ *is no easier than* $\mathsf{KLWE}_{m'-n,q,m',\chi}$.

Proof. We show how to turn a $\mathsf{KLWE}_{m'-n,q,m',\chi}$ problem instance $(\mathbf{B}, \mathbf{c}) \in \mathbb{Z}_q^{(m'-n)\times m'} \times \mathbb{Z}_q^{m'-n}$ to an $\mathsf{SISLWE}_{n,m',q,\chi}$ problem instance. Let $\mathbf{B} = \begin{bmatrix}\mathbf{H}\\ \mathbf{Z}\end{bmatrix}$ and $\mathbf{c} = \begin{bmatrix}\mathbf{h}\\ \mathbf{z}\end{bmatrix}$ where $\mathbf{H} \in \mathbb{Z}_q^{(m'-2n)\times m'}$, $\mathbf{Z} \in \mathbb{Z}_q^{n\times m'}$, $\mathbf{h} \in \mathbb{Z}_q^{m'-2n}$, and $\mathbf{z} \in \mathbb{Z}_q^n$. The transformation directly follows from the proof of Lemma 4.9, [23]. Since $\mathbf{H}$ is random, the columns of $\mathbf{H}$ generates the set $\mathbb{Z}_q^{m'-2n}$ except with all but negligible probability. Using linear algebra, we first find a matrix $\mathbf{A'} \in \mathbb{Z}_q^{n\times m'}$ such that the rows of $\mathbf{A'}$ generates the set $S = \{\mathbf{x} \in \mathbb{Z}_q^{m'} : \mathbf{Hx} = \mathbf{0} \pmod q\}$. We randomise $\mathbf{A'}$ by left-multiplying it by a unimodular matrix $\mathbf{U} \in \mathbb{Z}_q^{n\times n}$ to get $\mathbf{A} \in \mathbb{Z}_q^{n\times m'}$. We set $\mathbf{b}^t = \tilde{\mathbf{s}}^t\mathbf{A} + \mathbf{r}^t$ where $\tilde{\mathbf{s}} \leftarrow \mathbb{Z}_q^n$ and $\mathbf{r} \in \mathbb{Z}_q^{m'}$ be an arbitrary solution to $\mathbf{Hr} = \mathbf{h} \pmod q$. Finally, $(\mathbf{A}, \mathbf{b}, \mathbf{z}, \mathbf{Z})$ is returned as an $\mathsf{SISLWE}_{n,m',q,\chi}$ challenge.

We analyse the transformation. If $\mathbf{Hx} = \mathbf{h}$ and $\mathbf{Zx} = \mathbf{z}$, we can write $\mathbf{r} = \mathbf{r'} + \mathbf{x}$ where $\mathbf{Hr'} = \mathbf{0} \pmod q$. As the row of $\mathbf{A}$ generates S, we have $\mathbf{r'} = \mathbf{A}^t\mathbf{v}$ for some $\mathbf{v} \in \mathbb{Z}_q^n$. This shows that $\mathbf{b}^t = \mathbf{s}^t\mathbf{A} + \mathbf{x}^t$ for some random $\mathbf{s}$ $(= \tilde{\mathbf{s}} + \mathbf{v})$ and therefore $(\mathbf{A}, \mathbf{b}^t, \mathbf{z} = \mathbf{Zx}, \mathbf{Z})$ is distributed as SISLWE samples. On the other hand, if $\mathbf{c}$ is random, $\mathbf{b}^t = \tilde{s}^t\mathbf{A} + \mathbf{r}^t$ is uniformly random on $\mathbb{Z}_q^n$. So, in the tuple $(\mathbf{A}, \mathbf{b}^t, \mathbf{z}, \mathbf{Z})$, $\mathbf{b}$ is uniformly random and there exists an (unknown) $\mathbf{e} \sim \chi^{m'}$ such that $\mathbf{z} = \mathbf{Ze}$. □

Lemma 6. $\mathsf{SISnLWE}_{n,q,m,\chi}$ *is no easier than* $\mathsf{SISLWE}_{n,m',q,\chi}$.

Proof. We essentially apply the proof of [7], Lemma 2. Let $(\bar{\mathbf{A}}, \bar{\mathbf{b}}, \bar{\mathbf{z}}, \bar{\mathbf{Z}})$ be an instance of $\mathsf{SISLWE}_{n,m',q,\chi}$ where $m' = n + m$. Without loss of generality, let $\bar{\mathbf{A}} = [\bar{\mathbf{A}}_1|\bar{\mathbf{A}}_2]$, $\bar{\mathbf{b}}^t = [\bar{\mathbf{b}}_1^t|\bar{\mathbf{b}}_2^t]$ where $\bar{\mathbf{A}}_1 \in \mathbb{Z}_q^{n\times n}$ is invertible over $\mathbb{Z}_q$ (recall the random matrices $\bar{\mathbf{A}}$ and $\bar{\mathbf{Z}}$ have $m' = O(n \log q)$ columns so with overwhelming probability they have n linearly independent columns), $\bar{\mathbf{b}}_1 \in \mathbb{Z}_q^n$. We first set

$$\mathbf{A} \leftarrow -\bar{\mathbf{A}}_1^{-1}\bar{\mathbf{A}}_2 \quad \text{and} \quad \mathbf{b}^t \leftarrow \bar{\mathbf{b}}_2^t + \bar{\mathbf{b}}_1^t\mathbf{A}$$

Let $\bar{\mathbf{Z}} = [\bar{\mathbf{Z}}_1|\bar{\mathbf{Z}}_2]$ where $\bar{\mathbf{Z}}_1 \in \mathbb{Z}_q^{n\times n}$ is invertible modulus q. We set

$$\mathbf{z} = \mathbf{S}\bar{\mathbf{z}} \quad \text{and} \quad \mathbf{Z} = \mathbf{S}\bar{\mathbf{Z}}_2$$

where $\mathbf{S} \in \mathbb{Z}_q^{n\times n}$ such that $\mathbf{S}\bar{\mathbf{Z}}_1 = \mathbf{0} \pmod q$ ($\mathbf{S}$ can be constructed by left-multiplying a random $\mathbb{Z}_q^{n\times n}$-matrix to the solution of $\mathbf{X}\bar{\mathbf{Z}}_1 = \mathbf{0} \bmod q$). Then $(\mathbf{A}, \mathbf{b}, \mathbf{z}, \mathbf{Z})$ is returned as an $\mathsf{SISnLWE}_{n,m,q,\chi}$ problem instance. We analyse the transformation. If there exists $\mathbf{s} \leftarrow \mathbb{Z}_q^n, \mathbf{x} \leftarrow \chi^{m'}$ where $\mathbf{x}^t = [\mathbf{x}_1^t|\mathbf{x}_2^t]$ and $\mathbf{x}_1 \leftarrow \chi^n$ such that

$$\bar{\mathbf{b}}^t = \mathbf{s}^t[\bar{\mathbf{A}}_1|\bar{\mathbf{A}}_2] + [\mathbf{x}_1^t|\mathbf{x}_2^t] \quad \text{and} \quad \bar{\mathbf{z}} = \bar{\mathbf{Z}}_1\mathbf{x}_1 + \bar{\mathbf{Z}}_2\mathbf{x_2}.$$

The transformation gives us

$$\mathbf{b}^t = \mathbf{x}_1^t\mathbf{A} + \mathbf{x}_2^t \quad \text{and} \quad \mathbf{z} = \mathbf{Z}\mathbf{x}_2$$

where $\mathbf{Z}$ is uniformly random (because $\bar{\mathbf{Z}}$ is uniformly random). So $(\mathbf{A}, \mathbf{b}^t, \mathbf{z}, \mathbf{Z})$ has pseudorandom distribution. If $\bar{\mathbf{b}}$ and $\bar{\mathbf{z}}$ are uniformly random, then $\mathbf{b}$ and $\mathbf{z}$ are uniformly random. So $(\mathbf{A}, \mathbf{b}^t, \mathbf{z}, \mathbf{Z})$ has random distribution. □

4.2 Security Games

In order to prove security, we proceed by games. For $i = 0, 1, 2, 3$, we denote by G_i the i-th security game and S_i the event that the adversary $\mathcal{A}$ wins the security game G_i, e.g., by outputting $\mu' = \mu$.

G_0: The first game is the real IND-CCA security game. Let $\mathbf{k}_0, \mathbf{k}_1 \leftarrow \{0,1\}^n$ and let $\mathbf{s} = \mathbf{k}_1 \cdot \lfloor q/2 \rfloor + \bar{\mathbf{s}}$. The adversary $\mathcal{A}$ gets $\mathsf{pk} = (\mathbf{A}, \mathbf{A}_1, \mathbf{U})$, the hash function H, and a challenge ciphertext $\mathsf{ct}^* = (\mathbf{c}_0^*, \mathbf{c}_1^*, \mathbf{t}^*)$, where

$$\mathbf{c}_0^* = \mathbf{s}^t\mathbf{A} + \mathbf{e}_0^t, \quad \mathbf{c}_1^* = \mathbf{s}^t(\mathbf{A}_1 + \mathsf{FRD}(\mathbf{t}^*)\mathbf{G}) + \mathbf{e}_1^t, \quad \mathbf{t}^* = H(\mathbf{c}_0^*, \mathbf{U}\mathbf{e}_1)$$

and a session key $\mathbf{k}_\mu$ where $\mu \leftarrow \{0,1\}$. The simulator $\mathcal{B}$ implements the decapsulation oracle by following the real decapsulation algorithm. $\mathcal{A}$ eventually outputs a bit μ' and it wins if $\mu' = \mu$.

G_1: Game G_1 is identical to G_0 except that the decapsulation oracle rejects any ciphertext $\mathsf{ct} = (\mathbf{c}_0, \mathbf{c}_1, \mathbf{t})$ where $\mathbf{c}_0 \neq \mathbf{c}_0^*$ but $\mathbf{t} = \mathbf{t}^*$.

G_2: Game G_2 is identical to G_1 except that the decapsulation oracle rejects any ciphertext $\mathsf{ct} = (\mathbf{c}_0, \mathbf{c}_1, \mathbf{t})$ where $\mathbf{c}_0 = \mathbf{c}_0^*$ and $\mathbf{c}_1 \neq \mathbf{c}_1^*$.

G_3: Game G_3 is identical to G_2 except that the public key and the challenge ciphertext are simulated as follows.

1. Choose a hash function $H : \{0,1\}^* \to \{0,1\}^\lambda \setminus \{0^\lambda\}$.
2. Sample $\mathbf{A} \leftarrow \mathbb{Z}_q^{n\times m}$;
3. Sample $\mathbf{Z} \leftarrow \mathbb{Z}_q^{n\times w}$, $\mathbf{R}^t \leftarrow D_{\mathbb{Z},\omega(\sqrt{\log n})}^{w\times m}$. If $\mathbf{R}$ has rank $< w$, the simulator aborts the simulation and exits without proceeding to the next step.
4. Using linear algebra to find $(\mathbf{R}^t)^{-1} \in \mathbb{Z}^{m\times w}$ such that $\mathbf{R}^t \cdot (\mathbf{R}^t)^{-1} = \mathbf{I}_w$; Set $\mathbf{U} \leftarrow \mathbf{Z}(\mathbf{R}^t)^{-1}$.
5. Sample $\mathbf{e}_0 \leftarrow D_{\mathbb{Z},\alpha q}^m$, $\mathbf{v} \leftarrow D_{\mathbb{Z},\alpha q\sqrt{m}\cdot\omega(\sqrt{\log n})}^w$; Set $\mathbf{e}_1^t \leftarrow \mathbf{e}_0^t\mathbf{R} + \mathbf{v}^t$.

6. Sample the session key $\mathbf{k}_1 \leftarrow \{0,1\}^n$, $\bar{\mathbf{s}} \leftarrow D^n_{\mathbb{Z},\alpha q}$; Set $\mathbf{s} \leftarrow \mathbf{k}_1 \cdot \lfloor q/2 \rfloor + \bar{\mathbf{s}}$.
7. Compute $\mathbf{c}_0^{*t} \leftarrow \mathbf{s}^t\mathbf{A} + \mathbf{e}_0^t$ and $\mathbf{c}_2^* \leftarrow \mathbf{Z}\mathbf{e}_0 + \mathbf{U}\mathbf{v}$ (which equals to $\mathbf{U}\mathbf{e}_1$).
8. Set $\mathbf{A}_1 \leftarrow \mathbf{A}\mathbf{R} - \mathsf{FRD}(\mathbf{t}^*)\mathbf{G}$ and $\mathbf{c}_1^{*t} \leftarrow \mathbf{c}_0^{*t}\mathbf{R} + \mathbf{v}^t$ where $\mathbf{t}^* \leftarrow H(\mathbf{c}_0^*, \mathbf{c}_2^*)$.
9. Return $\mathsf{pk} = (\mathbf{A}, \mathbf{A}_1, \mathbf{U})$, public parameter H, $\mathsf{sk} = \mathbf{R}$, and $\mathsf{ct}^* = (\mathbf{c}_0^*, \mathbf{c}_1^*, \mathbf{t}^*)$

G_4**:** Game G_4 is the identical to G_3 except that both session keys $\mathbf{k}_0^*$ and $\mathbf{k}_1^*$ are sampled uniformly at random and, in particular, independently of the challenge ciphertext. We note that in this game, the adversary has no advantage in winning the game.

4.3 Security Proofs

Theorem 1. *Under the assumptions that H be second-pre-image resistant, the problem $\mathsf{SIS}_{n,w,q,\beta}$ be hard, and the problem $\mathsf{SISnLWE}_{n,m,q,D_{\mathbb{Z},\alpha q}}$ be hard, the KEM scheme presented in section 3 is IND-CCA secure. In particular, we have*

$$\mathsf{Adv}^{\mathsf{ind-cca}}_{\mathcal{A},\Pi}(\lambda) \leq \mathsf{Adv}^{\mathsf{coll}}_{\mathcal{B}_1,\mathcal{H}}(\lambda) + \mathsf{Adv}^{\mathsf{SIS}_{n,q,\beta}}_{\mathcal{B}_2}(\lambda) + \mathsf{Adv}^{\mathsf{SISnLWE}_{n,m,q,D_{\mathbb{Z},\alpha q}}}_{\mathcal{B}_3}(\lambda) + \mathsf{negl}(\lambda)$$

for some algorithms $\mathcal{B}_1$ $\mathcal{B}_2$ and $\mathcal{B}_3$, and $\mathsf{negl}(\lambda)$ is negligible in λ.

Proof. We establish the theorem by showing that the neighbour games are indistinguishable (either computationally or statistically) based on our assumptions.

Lemma 7. *G_0 and G_1 are computationally indistinguishable if $H : \{0,1\} \to \{0,1\}^\lambda \setminus \{0^\lambda\}$ is second-pre-image collision resistant. In particular,*

$$|\Pr[S_0] - \Pr[S_1]| \leq \mathsf{Adv}^{\mathsf{coll}}_{\mathcal{B}_1,\mathcal{H}}(\lambda) \tag{1}$$

for some algorithm $\mathcal{B}_1$.

Proof. First of all, we have, by definition,

$$\Pr[S_0] = \Pr[\mu = \mu'] = 1/2 \cdot \mathsf{Adv}^{\mathsf{ind-cca}}_{\mathcal{A},\Pi}(\lambda) + 1/2 \tag{2}$$

Let E_1 be the event that the adversary $\mathcal{A}$ issues a valid ciphertext (i.e., which can be decrypted properly) $\mathsf{ct} = (\mathbf{c}_0, \mathbf{c}_1, \mathbf{t})$ where $\mathbf{c}_0 \neq \mathbf{c}_0^*$ but $\mathbf{t} = \mathbf{t}^*$. We note that G_0 is identical to G_1 unless E_1 happens. So by Lemma 1 we have $|\Pr[S_0] - \Pr[S_1]| \leq \Pr[E_1]$. One the other hand, it is readily seen that E_1 implies a collision of H under a given pre-image $\mathbf{c}_0^*$. Therefore, $|\Pr[S_0] - \Pr[S_1]| \leq \mathsf{Adv}^{\mathsf{coll}}_{\mathcal{B},\mathcal{H}}(\lambda)$.

Lemma 8. *G_1 and G_2 are computationally indistinguishable if $\mathsf{SIS}_{n,w,q,\beta}$ problem is hard. In particular, for some algorithm $\mathcal{B}_2$*

$$|\Pr[S_1] - \Pr[S_2]| \leq \mathsf{Adv}^{\mathsf{SIS}_{n,q,\beta}}_{\mathcal{B}_2}(\lambda) \tag{3}$$

where $\beta = 2s\sqrt{w}$ for $s \leq \sqrt{6w} \cdot \alpha q \cdot \omega(\sqrt{\log n})$.

Proof. Recall that in G_2, the simulator $\mathcal{B}$ runs the real $\mathsf{KeyGen}(1^\lambda)$ to generate $\mathsf{pk} = (\mathbf{A}, \mathbf{A}_1, \mathbf{U})$ and private key $\mathbf{R}$ as well as the public hash function H. It also computes a real challenge ciphertext $\mathsf{ct}^* = (\mathbf{c}_0^*, \mathbf{c}_1^*, \mathbf{t}^*)$ where

$$\mathbf{c}_0^{*t} = \mathbf{s}_1^t \mathbf{A} + \mathbf{e}_0^t, \quad \mathbf{c}_1^{*t} = \mathbf{s}_1^t (\mathbf{A}_1 + \mathsf{FRD}(\mathbf{t}^*)\mathbf{G}) + \mathbf{e}_1^t, \quad \mathbf{t}^* = H(\mathbf{c}_0^*, \mathbf{U}\mathbf{e}_1)$$

Let E_2 be the event that the adversary issues a *valid* ciphertext (i.e., which can be decrypted properly to a valid session key) $\mathsf{ct} = (\mathbf{c}_0, \mathbf{c}_1, \mathbf{t})$ where $\mathbf{c}_0 = \mathbf{c}_0^*$ (thus $\mathbf{t} = \mathbf{t}^*$) and $\mathbf{c}_1 \neq \mathbf{c}_1^*$. Since G_1 and G_2 are identical unless E_2 happens, by Lemma 1 we have $|\Pr[S_1] - \Pr[S_2]| \leq \Pr[E_2]$.

Next we show $\Pr[E_2]$ is bounded by the probability of successfully solving the SIS problem. A SIS adversary $\mathcal{B}_2$ receives its challenge, a random matrix $\mathbf{U} \in \mathbb{Z}_q^{n \times w}$. It generates $\mathbf{A}$, $\mathbf{A}_1$, H exactly as in the algorithm KeyGen, and publishes $\mathsf{pk} = (\mathbf{A}, \mathbf{A}_1, \mathbf{U}, H)$. It is easy to see that pk has the correct distribution. $\mathcal{B}_2$ simulates the security game G_1, interrupting the simulation whenever E_2 happens and using the corresponding query to solve its SIS instance.

Assume the decapsulation query is $\mathsf{ct} = (\mathbf{c}_0^*, \mathbf{c}_1, \mathbf{t}^*)$ where $\mathbf{c}_1^t = \tilde{\mathbf{s}}^t(\mathbf{A}_1 + \mathsf{FRD}(\mathbf{t}^*)\mathbf{G}) + \tilde{\mathbf{e}}_1^t$. We must have $\mathbf{U}\tilde{\mathbf{e}} = \mathbf{U}\mathbf{e}_1$. $\mathcal{B}$ uses its trapdoor $\mathbf{R}$ to recover $\tilde{\mathbf{e}}_1$, and outputs $\mathbf{e} \leftarrow \tilde{\mathbf{e}}_1 - \mathbf{e}_1$. Now we argue that $\mathbf{e}$ is indeed a correct solution to the SIS problem instance. First, we must have $\tilde{\mathbf{e}}_1 \neq \mathbf{e}_1$ with overwhelming probability. Otherwise, since $\mathbf{c}_0 = \mathbf{c}_0^*$ and $\mathbf{t} = \mathbf{t}^*$, we have to have $\mathbf{c}_1 = \mathbf{c}_1^*$, and thus the decapsulation query is the challenge ciphertext. This shows that $\mathbf{e} \neq \mathbf{0}$. Second, since ct is a valid ciphertext, $\|\bar{\mathbf{e}}_1\| \leq s\sqrt{w}$ by construction. To bound the norm of $\mathbf{e}$, we have

$$\begin{aligned} \|\mathbf{e}\| \leq 2s\sqrt{w} &= 2\sqrt{(\|\mathbf{e}_0\|^2 + m(\alpha q)^2) \cdot \omega(\sqrt{\log n})^2} \cdot \sqrt{w} \\ &\leq \sqrt{6w} \cdot \alpha q \cdot \omega(\sqrt{\log n}) \end{aligned}$$

as required. This shows that $\Pr[E_2] \leq \mathsf{Adv}_{\mathcal{B}_2}^{\mathsf{SIS}_{n,q,\beta}}(\lambda)$.

Lemma 9. *There exist a negligible function* $\mathsf{negl}(\lambda)$ *such that*

$$|\Pr[S_2] - \Pr[S_3]| \leq \mathsf{negl}(\lambda) \tag{4}$$

Proof. First of all, let Abort be the event that the simulation aborts due to the fact that the matrix $\mathbf{R}$'s rows (i.e. $\mathbf{R}^t$'s columns) are not all linearly independent. Recall that $m \geq 2w = 2n \log q$ which means $\mathbf{R}$ has (at least) $m = 2w$ independent samples from $D_{\mathbb{Z}^w, \omega(\sqrt{\log n})}$. As shown in [2], the matrix $\mathbf{R}$ has rank w, i.e., the event Abort happens with a negligible probability.

In the following, we show that the public key and the challenge ciphertext simulated in G_3 is indistinguishable from those output in G_2 from the adversary's view, conditioned on that Abort did not happen.

For pk in G_3, we can see that $\mathbf{A}$ and H are correctly distributed. Since $\mathbf{R}^t \leftarrow D_{\mathbb{Z}, \omega(\sqrt{\log n})}^{w \times m}$ (i.e., the coordinates of $\mathbf{R}$ are independent and each of them has distribution $D_{\mathbb{Z}, \omega(\sqrt{\log n})}$), by Lemma 3, $\mathbf{A}_1$ generated in G_3 (i.e., $\mathbf{A}\mathbf{R} - \mathsf{FRD}(\mathbf{t}^*)\mathbf{G}$) and G_2 (i.e., $\mathbf{A}\mathbf{R}$) are statistically close (both are statistically close

to the uniform distribution over $\mathbb{Z}_q^{n\times m}$). Moreover, by the fact that $\mathbf{Z} \in \mathbb{Z}_q^{n\times m}$ is uniformly random, we have $\mathbf{U} = \mathbf{Z}(\mathbf{R}^t)^{-1}$ is also uniformly random as in G_2.

Now we look at the challenge ciphertext. First of all, $\mathbf{c}_0^*$ is correctly distributed. Second, recall $\mathbf{e}_1^t \leftarrow \mathbf{e}_0^t\mathbf{R} + \mathbf{v}^t$. By adapting Theorem 3.1 of [28] and Corollary 3.10 of [32], conditioned on $\mathbf{A}_1$ and $\mathbf{U}$, $\mathbf{e}_1$ has a distribution that is statistically close to $D^w_{\mathbb{Z},s}$, where $s^2 = (\|\mathbf{e}_0\|^2 + m(\alpha q)^2) \cdot \omega(\sqrt{\log n})^2$. So, $\mathbf{e}_1$ has the required distribution except with negligible probability. Moreover,

$$\begin{aligned}\mathbf{c}_1^{*t} &= \mathbf{c}_0^{*t}\mathbf{R} + \mathbf{v}^t = (\mathbf{s}^t\mathbf{A} + \mathbf{e}_0^t)\mathbf{R} + \mathbf{v}^t \\ &= \mathbf{s}^t(\mathbf{A}_1 + \mathsf{FRD}(\mathbf{t}^*)\mathbf{G}) + (\mathbf{e}_0^t\mathbf{R} + \mathbf{v}^t) \\ &= \mathbf{s}^t(\mathbf{A}_1 + \mathsf{FRD}(\mathbf{t}^*)\mathbf{G}) + \mathbf{e}_1^t\end{aligned}$$

and

$$\mathbf{t}^* = H(\mathbf{c}_0^*, \mathbf{Z}\mathbf{e}_0 + \mathbf{U}\mathbf{v}) = H(\mathbf{c}_0^*, \mathbf{U}\mathbf{R}^t\mathbf{e}_0 + \mathbf{U}\mathbf{v}) = H(\mathbf{c}_0^*, \mathbf{U}\mathbf{e}_1)$$

which shows that $\mathbf{c}_1^*$ and $\mathbf{t}^*$ are also correctly distributed.

Finally, notice that the simulator in G_3 cannot decapsulate any ciphertext $\mathsf{ct} = (\mathbf{c}_0, \mathbf{c}_1, \mathbf{t})$ where $\mathbf{c}_0 = \mathbf{c}_0^*$ and $\mathbf{t} = \mathbf{t}^*$. However, such a decapsulation query has already been excluded in G_2. Summing up, G_3 is distributed the same as G_2 except with negligible statistical error $\mathsf{negl}(\lambda)$. So, we have equation (4).

Lemma 10. *G_3 and G_4 are computationally indistinguishable if $\mathsf{SISnLWE}_{n,m,q,D_{\mathbb{Z},\alpha q}}$ is hard. In particular,*

$$|\Pr[S_3] - \Pr[S_4]| \leq \mathsf{Adv}^{\mathsf{SISnLWE}_{n,m,q,D_{\mathbb{Z},\alpha q}}}_{\mathcal{B}_3}(\lambda) \tag{5}$$

where $\chi = D_{\mathbb{Z},\alpha q}$ for some algorithm $\mathcal{B}_3$. Moreover the adversary has no advantage in G_4, i.e.,

$$\Pr[S_4] = 1/2 \tag{6}$$

Proof. We show a simulator that simulates $\mathcal{S}$ either G_3 or G_4 from an instance of $\mathsf{SISnLWE}_{n,m,q,D_{\mathbb{Z},\alpha q}}$ problem. $\mathcal{S}$ receives its challenge $(\mathbf{A}, \mathbf{b}^t, \mathbf{z} = \mathbf{Z}\mathbf{e}_0, \mathbf{Z})$ for some $\mathbf{e}_0 \sim \chi^m$ and needs to decide if there are vectors $\bar{\mathbf{s}} \leftarrow D^n_{\mathbb{Z},\alpha q}$ such that $\mathbf{b}^t = \bar{\mathbf{s}}^t\mathbf{A} + \mathbf{e}_0^t$. $\mathcal{S}$ does the following to prepare pk, H and ct^*.

1. Set a hash function $H : \{0,1\}^* \to \{0,1\}^\lambda \setminus \{0^\lambda\}$.
2. Set $\mathbf{A}$ from the challenge.
3. Sample $\mathbf{R}^t \leftarrow D^{w\times m}_{\mathbb{Z},\omega(\sqrt{\log n})}$. If the columns of $\mathbf{R}$ are not linearly independent, the simulator aborts the simulation and exits without going to the next step.
4. Use linear algebra to find $(\mathbf{R}^t)^{-1} \in \mathbb{Z}^{m\times w}$ such that $\mathbf{R}^t \cdot (\mathbf{R}^t)^{-1} = \mathbf{I}_w$; Set $\mathbf{U} \leftarrow \mathbf{Z}(\mathbf{R}^t)^{-1} \in \mathbb{Z}_q^{n\times w}$.
5. Sample $\mathbf{k}_0, \mathbf{k}_1 \leftarrow \{0,1\}^n$; Set $\mathbf{c}_0^{*t} \leftarrow (\mathbf{k}_1\lfloor q/2 \rfloor)^t\mathbf{A} + \mathbf{b}^t$.
6. Sample $\mathbf{v} \leftarrow D^w_{\mathbb{Z},\alpha q\sqrt{m}\cdot\omega(\sqrt{\log n})}$; Set $\mathbf{c}_1^{*t} \leftarrow \mathbf{c}_0^{*t}\mathbf{R} + \mathbf{v}^t$.
7. Set $\mathbf{t}^* = H(\mathbf{c}_0^*, \mathbf{z} + \mathbf{U}\mathbf{v})$ and $\mathbf{A}_1 \leftarrow \mathbf{A}\mathbf{R} - \mathsf{FRD}(\mathbf{t}^*)\mathbf{G}$ where $\mathbf{z} \in \mathbb{Z}_q^n$ is from the challenge.

8. Return $\mathsf{pk} = (\mathbf{A}, \mathbf{A}_1, \mathbf{U}, H)$, $\mathsf{ct}^* = (\mathbf{c}_0^*, \mathbf{c}_1^*, \mathbf{t}^*)$, and $\mathbf{k}_\mu$ for $\mu \leftarrow \{0,1\}$
9. Keep $\mathbf{R}$ for further use (i.e., implementing the decapsulation oracle).

Firstly, we can see that both games abort with the same probability. Conditioned on that both games do not abort, it is easy to see that pk has the correct distribution. Secondly, the trapdoor $\mathbf{R}$ enables answering all kinds of decapsulation queries except the ones that have already been excluded in G_1 and G_2.

We argue that depending on the $\mathsf{SISnLWE}_{n,m,q,D_{\mathbb{Z},\alpha q}}$ challenge, the simulator $\mathcal{S}$ either simulates G_4 or G_5. In the first case where $\mathbf{b}$ and $\mathbf{z}$ from the $\mathsf{SISnLWE}_{n,m,q,D_{\mathbb{Z},\alpha q}}$ challenge is random, $\mathbf{c}_0^*$ statistically hides $\mathbf{k}_1$ meaning that the challenge session $\mathbf{k}_\mu^*$ is anyway independent from ct^*. Therefore, $\mathcal{S}$ is simulating G_4. In the other case where $\mathbf{b}^t = \bar{\mathbf{s}}^t\mathbf{A} + \mathbf{e}_0^t$, the challenge ciphertext follows the correct distribution of G_3. To see this, we have

$$\mathbf{c}_0^* = (\mathbf{k}_1\lfloor q/2 \rfloor)^t\mathbf{A} + \mathbf{b}^t = (\mathbf{k}_1\lfloor q/2 \rfloor + \bar{\mathbf{s}})^t\mathbf{A} + \mathbf{e}_0^t = \mathbf{s}^t\mathbf{A} + \mathbf{e}_0^t$$

and

$$\mathbf{t}^* = H(\mathbf{c}_0^*, \mathbf{z} + \mathbf{U}\mathbf{v}) = H(\mathbf{c}_0^*, \mathbf{Z}\mathbf{e}_0 + \mathbf{U}\mathbf{v}) = H(\mathbf{c}_0^*, \mathbf{U}\mathbf{e}_1)$$

and

$$\begin{aligned}\mathbf{c}_1^{*t} &= \mathbf{c}_0^{*t}\mathbf{R} + \mathbf{v}^t = (\mathbf{s}^t\mathbf{A} + \mathbf{e}_0^t)\mathbf{R} + \mathbf{v}^t \\ &= \mathbf{s}^t(\mathbf{A}_1 + \mathsf{FRD}(\mathbf{t}^*)\mathbf{G}) + (\mathbf{e}_0^t\mathbf{R} + \mathbf{v}^t) \\ &= \mathbf{s}^t(\mathbf{A}_1 + \mathsf{FRD}(\mathbf{t}^*)\mathbf{G}) + \mathbf{e}_1^t\end{aligned}$$

This shows that $\mathcal{S}$ is simulating G_3. So we have an efficient distinguisher of G_3 and G_4 which in turn leads to an efficient algorithm that solves the problem $\mathsf{SISLWE}_{n,m,q,\chi,D_{\mathbb{Z},\alpha q}}$. Equation (5) follows.

Finally, we note that in G_4, both $\mathbf{k}_0$ and $\mathbf{k}_1$ are independent of the challenge ciphertext ct^* and the adversary has no advantage in winning the security game. So, we have equation (6)

We conclude the proof by combing inequalities (1), (2), (3), (4), (5), and (6). □

References

1. Shweta Agrawal, Dan Boneh, and Xavier Boyen. Efficient lattice (h)ibe in the standard model. In Henri Gilbert, editor, *Advances in Cryptology - EUROCRYPT 2010*, volume 6110 of *Lecture Notes in Computer Science*, pages 553–572. Springer Berlin Heidelberg, 2010.
2. Shweta Agrawal, Dan Boneh, and Xavier Boyen. Lattice basis delegation in fixed dimension and shorter-ciphertext hierarchical ibe. In Tal Rabin, editor, *Advances in Cryptology - CRYPTO 2010*, volume 6223 of *Lecture Notes in Computer Science*, pages 98–115. Springer Berlin Heidelberg, 2010.

3. Shweta Agrawal, Benoît Libert, and Damien Stehlé. Fully secure functional encryption for inner products, from standard assumptions. In *Annual International Cryptology Conference*, pages 333–362. Springer, 2016.
4. Jacob Alperin-Sheriff and Chris Peikert. Circular and kdm security for identity-based encryption. In *PKC 2012*, pages 334–352. Springer, 2012.
5. Joël Alwen, Stephan Krenn, Krzysztof Pietrzak, and Daniel Wichs. Learning with rounding, revisited. In *Advances in Cryptology–CRYPTO 2013*, pages 57–74. Springer, 2013.
6. Daniel Apon, Xiong Fan, and Feng-Hao Liu. Deniable attribute based encryption for branching programs from lwe. In *Theory of Cryptography Conference*, pages 299–329. Springer, 2016.
7. Benny Applebaum, David Cash, Chris Peikert, and Amit Sahai. Fast cryptographic primitives and circular-secure encryption based on hard learning problems. In Shai Halevi, editor, *Advances in Cryptology - CRYPTO 2009*, volume 5677 of *Lecture Notes in Computer Science*, pages 595–618. Springer Berlin Heidelberg, 2009.
8. Mihir Bellare, Eike Kiltz, Chris Peikert, and Brent Waters. Identity-based (lossy) trapdoor functions and applications. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 228–245. Springer, 2012.
9. Dan Boneh, Ran Canetti, Shai Halevi, and Jonathan Katz. Chosen-ciphertext security from identity-based encryption. *SIAM J. Comput.*, 36(5):1301–1328, December 2006.
10. Xavier Boyen and Qinyi Li. Direct cca-secure kem and deterministic pke from plain lwe. In *International Conference on Post-Quantum Cryptography*, pages 116–130. Springer, 2019.
11. Xavier Boyen, Qixiang Mei, and Brent Waters. Direct chosen ciphertext security from identity-based techniques. In *Proceedings of the 12th ACM conference on Computer and communications security*, pages 320–329. ACM, 2005.
12. Zvika Brakerski, Adeline Langlois, Chris Peikert, Oded Regev, and Damien Stehlé. Classical hardness of learning with errors. In *Proceedings of the Forty-fifth Annual ACM Symposium on Theory of Computing*, STOC '13, pages 575–584, New York, NY, USA, 2013. ACM.
13. Ronald Cramer and Victor Shoup. Design and analysis of practical public-key encryption schemes secure against adaptive chosen ciphertext attack. *SIAM Journal on Computing*, 33(1):167–226, 2003.
14. Danny Dolev, Cynthia Dwork, and Moni Naor. Nonmalleable cryptography. *SIAM review*, 45(4):727–784, 2003.
15. Dung Hoang Duong, Kazuhide Fukushima, Shinsaku Kiyomoto, Partha Sarathi Roy, and Willy Susilo. A lattice-based public key encryption with equality test in standard model. In *Australasian Conference on Information Security and Privacy*, pages 138–155. Springer, 2019.
16. Eiichiro Fujisaki and Tatsuaki Okamoto. Secure integration of asymmetric and symmetric encryption schemes. *Journal of cryptology*, 26(1):80–101, 2013.
17. Romain Gay, Dennis Hofheinz, Eike Kiltz, and Hoeteck Wee. Tightly cca-secure encryption without pairings. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 1–27. Springer, 2016.
18. Craig Gentry, Chris Peikert, and Vinod Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In *Proceedings of the 40th annual ACM symposium on Theory of computing*, STOC '08, pages 197–206, New York, NY, USA, 2008. ACM.

19. Dennis Hofheinz and Eike Kiltz. Practical chosen ciphertext secure encryption from factoring. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 313–332. Springer, 2009.
20. Junzuo Lai, Robert H Deng, Shengli Liu, and Weidong Kou. Efficient cca-secure pke from identity-based techniques. In *Cryptographers Track at the RSA Conference*, pages 132–147. Springer, 2010.
21. Richard Lindner and Chris Peikert. Better key sizes (and attacks) for lwe-based encryption. In *Cryptographers Track at the RSA Conference*, pages 319–339. Springer, 2011.
22. Daniele Micciancio. Duality in lattice cryptography. In *Public-Key Cryptography*. Invited Talk, 2010.
23. Daniele Micciancio and Petros Mol. Pseudorandom knapsacks and the sample complexity of lwe search-to-decision reductions. In *Annual Cryptology Conference*, pages 465–484. Springer, 2011.
24. Daniele Micciancio and Chris Peikert. Trapdoors for lattices: Simpler, tighter, faster, smaller. In David Pointcheval and Thomas Johansson, editors, *Advances in Cryptology – EUROCRYPT 2012*, volume 7237 of *Lecture Notes in Computer Science*, pages 700–718. Springer Berlin Heidelberg, 2012.
25. Daniele Micciancio and Oded Regev. Worst-case to average-case reductions based on gaussian measures. *SIAM J. Comput.*, 37(1):267–302, April 2007.
26. Moni Naor and Moti Yung. Public-key cryptosystems provably secure against chosen ciphertext attacks. In *Proceedings of the twenty-second annual ACM symposium on Theory of computing*, pages 427–437, 1990.
27. Adam ONeill, Chris Peikert, and Brent Waters. Bi-deniable public-key encryption. In *Annual Cryptology Conference*, pages 525–542. Springer, 2011.
28. Chris Peikert. An efficient and parallel gaussian sampler for lattices. In *Annual Cryptology Conference*, pages 80–97. Springer, 2010.
29. Chris Peikert et al. A decade of lattice cryptography. *Foundations and Trends® in Theoretical Computer Science*, 10(4):283–424, 2016.
30. Chris Peikert and Sina Shiehian. Noninteractive zero knowledge for np from (plain) learning with errors. In *Annual International Cryptology Conference*, pages 89–114. Springer, 2019.
31. Chris Peikert, Vinod Vaikuntanathan, and Brent Waters. A framework for efficient and composable oblivious transfer. In *Advances in Cryptology – CRYPTO 2008*, pages 554–571. Springer, 2008.
32. Oded 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.
33. Amit Sahai. Non-malleable non-interactive zero knowledge and adaptive chosen-ciphertext security. In *40th Annual Symposium on Foundations of Computer Science (Cat. No. 99CB37039)*, pages 543–553. IEEE, 1999.
34. Victor Shoup. Sequences of games: a tool for taming complexity in security proofs. Cryptology ePrint Archive, Report 2004/332, 2004. `https://eprint.iacr.org/2004/332`.
35. Jiang Zhang, Yu Yu, Shuqin Fan, and Zhenfeng Zhang. Improved lattice-based cca2-secure pke in the standard model. Cryptology ePrint Archive, Report 2019/149, 2019. `https://eprint.iacr.org/2019/149`.