(Tightly) QCCA-Secure Key-Encapsulation Mechanism in the Quantum Random Oracle Model

Keita Xagawa and Takashi Yamakawa

NTT Secure Platform Laboratories
3-9-11, Midori-cho Musashino-shi, Tokyo 180-8585 Japan
{xagawa.keita, yamakawa.takashi}@lab.ntt.co.jp

Abstract. This paper shows the security against *quantum* chosen-ciphertext attacks (QCCA security) of the KEM in Saito, Yamakawa, and Xagawa (EUROCRYPT 2018) in the QROM. The proof is very similar to that for the CCA security in the QROM, easy to understand, and as tight as the original proof.

keywords: Tight security, quantum chosen-ciphertext security, post-quantum cryptography, KEM.

1 Introduction

Quantum Superposition Attacks: Quantum superposition attacks are worth being considered (if we can mount them in the future). Theoretically speaking, we already know quantum superposition attacks that break classically-secure cryptographic primitives: Kuwakado and Morii [KM12] presented a quantum chosen-plaintext attack against the Even-Monsour construction of a block cipher if the inner permutation is publicly available as quantum oracle, which employed Simon's algorithm neatly. Kaplan, Leurent, Leverrier, and Naya-Plasencia [KLLNP16] also studied quantum superposition attacks against several block ciphers and modes. (On the other hand, Anand, Targhi, Tabia, and Unruh [ATTU16] showed some modes are secure if the underlying block cipher is quantumly-secure PRF.) Boneh and Zhandry [BZ13] also gave an example of a block cipher that is secure against chosen-plaintext-and-ciphertext attacks but vulnerable against quantum chosen-ciphertext attacks.

Security of PKE/KEM against Quantum Chosen-Ciphertext Attacks: Boneh and Zhandry [BZ13] introduced the quantum-chosen-ciphertext (QCCA) security for public-key encryption (PKE), which is security against adversaries that make decryption queries in quantum superpositions.

Boneh and Zhandry [BZ13] showed that a PKE scheme obtained by applying the Canetti-Halevi-Katz conversion [BCHK07] to an ID-based encryption (IBE) scheme and one-time signature is IND-QCCA-secure if the underlying IBE scheme is selectively-secure against quantum chosen-ID queries and the underlying one-time signature scheme is (classically) strongly, existentially unforgeable against chosen-message attacks. They also showed that if there exists an IND-CCA-secure PKE, then there exists an ill-formed PKE that is IND-CCA-secure but not IND-QCCA-secure [BZ13].

As far as we know, this is the only known PKE scheme that is proven to be IND-QCCA secure (excluding the concurrent work by Zhandry [Zha18, 2018-08-14 ver.]).

1.1 Our Contribution

We show that a key encapsulation mechanism (KEM) in Saito, Xagawa, and Yamakawa [SXY18] is also IND-QCCA-secure in the quantum random oracle model (QROM) if the underlying deterministic PKE is perfectly correct and disjoint-simulatable.

Our idea is very simple: At the last step in the security proof of the IND-CCA security, the challenger should simulate the decapsulation oracle on a query of any ciphertext c except the challenge ciphertext c^*. Roughly speaking, we observe that, if this simulation is "history-free," i.e., if the simulation does not depend on previously made queries at all, this procedure can be quantumly simulated by implementing this procedure in the quantum way. For example, in the last step of the game hopping in [SXY18], the decapsulation oracle on input c returns $K = \mathsf{H}_q(c)$ if $c \neq c^*$, where H_1 is a random function chosen by the reduction algorithm. We observe this procedure is "history-free" and can be implemented quantumly. [1]

[1] Boneh et al. [BDF+11] defined the history-free property of reduction for signature scheme, but they gave no definition of the history-free property of reduction for encryption schemes.

1.2 Concurrent Works

Zhandry [Zha18, 2018-08-14 ver.] showed that the PKE scheme obtained by applying the Fujisaki-Okamoto conversion [FO13] to a PKE scheme PKE and a DEM scheme DEM is IND-qCCA-secure in the QROM, if PKE is OW-CPA-secure and well-spread, DEM is OT-secure [2]. Zhandry proposed recording and testing techniques to simulate the decryption oracles. We note that his security proof is non-tight unlike ours.

2 Preliminaries

2.1 Notation

A security parameter is denoted by κ. We use the standard O-notations: O, Θ, Ω, and ω. DPT and PPT stand for deterministic polynomial time and probabilistic polynomial time. A function $f(\kappa)$ is said to be *negligible* if $f(\kappa) = \kappa^{-\omega(1)}$. We denote a set of negligible functions by $\mathsf{negl}(\kappa)$. For two finite sets $\mathcal{X}$ and $\mathcal{Y}$, $\mathrm{Map}(\mathcal{X}, \mathcal{Y})$ denote a set of all functions whose domain is $\mathcal{X}$ and codomain is $\mathcal{Y}$.

For a distribution χ, we often write "$x \leftarrow \chi$," which indicates that we take a sample x from χ. For a finite set S, $U(S)$ denotes the uniform distribution over S. We often write "$x \leftarrow S$" instead of "$x \leftarrow U(S)$." For a set S and a deterministic algorithm A, $\mathsf{A}(S)$ denotes the set $\{\mathsf{A}(x) \mid x \in S\}$.

If inp is a string, then "out $\leftarrow$ A(inp)" denotes the output of algorithm A when run on input inp. If A is deterministic, then out is a fixed value and we write "out := A(inp)." We also use the notation "out := A(inp; r)" to make the randomness r explicit.

For the Boolean statement P, $\mathsf{boole}(P)$ denotes the bit that is 1 if P is true, and 0 otherwise. For example, $\mathsf{boole}(b' \overset{?}{=} b)$ is 1 if and only if $b' = b$.

2.2 Quantum Computation

We refer to [NC00] for basic of quantum computation.

Quantum Random Oracle Model. Roughly speaking, the quantum random oracle model (QROM) is an idealized model where a hash function is modeled as a publicly and quantumly accessible random oracle. See [BDF+11] for a more detailed description of the model.

Lemma. We review a useful lemma regarding the quantum oracles.

Lemma 2.1. *Let ℓ be an integer. Let $\mathsf{H}\colon \{0,1\}^\ell \times \mathcal{X} \to \mathcal{Y}$ and $\mathsf{H}'\colon \mathcal{X} \to \mathcal{Y}$ be two independent random oracles. If an unbounded time quantum adversary $\mathcal{A}$ makes a query to H at most q_H times, then we have*

$$\left|\Pr[\mathcal{A}^{\mathsf{H},\mathsf{H}(s,\cdot)}() \to 1 \mid s \leftarrow \{0,1\}^\ell] - \Pr[\mathcal{A}^{\mathsf{H},\mathsf{H}'}() \to 1]\right| \leq q_\mathsf{H} \cdot 2^{\frac{-\ell+1}{2}}$$

where all oracle accesses of $\mathcal{A}$ can be quantum.

Though this seems to be a folklore, Saito et al. [SXY18] and Jiang et al. [JZC+18] gave the proof.

Simulation of Random Oracle. In the original quantum random oracle model introduced by Boneh et al. [BDF+11], they do not allow a reduction algorithm to access a random oracle, so it has to simulate a random oracle by itself. In contrast, in this paper, we give a random oracle access to a reduction algorithm. We remark that this is just a convention and not a modification of the model since we can simulate a random oracle against quantum adversaries in several ways; 1) $2q$-wise independent hash function [Zha12], 2) quantumly-secure PRF [BDF+11], and 3) hash function modeled as quantum random oracle [KLS18]. In addition, Zhandry proposed a new technique to simulate the quantum random oracle, the compressed oracle technique [Zha18]. His new simulation of the quantum random oracle is perfect even for *unbounded* number of queries. In what follows, we use t_{RO} to denote a time needed to simulate a quantum random oracle.

[2] any efficient adversary cannot distinguish $E(k, m_0)$ from $E(k, m_1)$ even if it chooses m_0 and m_1 with $|m_0| = |m_1|$.

3 Definitions

3.1 Public-Key Encryption

The model for PKE schemes is summarized as follows:

Definition 3.1. *A PKE scheme* PKE *consists of the following triple of polynomial-time algorithms* (Gen, Enc, Dec).

- $\mathsf{Gen}(1^\kappa; r_g) \rightarrow (ek, dk)$*: a key-generation algorithm that on input* 1^κ*, where* κ *is the security parameter, outputs a pair of keys* (ek, dk)*. ek and dk are called the encryption key and decryption key, respectively.*
- $\mathsf{Enc}(ek, m; r_e) \rightarrow c$*: an encryption algorithm that takes as input encryption key ek and message* $m \in \mathcal{M}$ *and outputs ciphertext* $c \in C$*.*
- $\mathsf{Dec}(dk, c) \rightarrow m/\bot$*: a decryption algorithm that takes as input decryption key dk and ciphertext c and outputs message* $m \in \mathcal{M}$ *or a rejection symbol* $\bot \notin \mathcal{M}$*.*

Definition 3.2. *We say a PKE scheme* PKE *is deterministic if* Enc *is deterministic. DPKE stands for deterministic public key encryption.*

Definition 3.3 (Correctness). *We say* $\mathsf{PKE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *has* perfect correctness *if for any* (ek, dk) *generated by* Gen *and for any* $m \in \mathcal{M}$ *,we have that*

$$\Pr[\mathsf{Dec}(dk, c) = m \mid c \leftarrow \mathsf{Enc}(ek, m)] = 1.$$

We also review δ correctness in HHK17.

Definition 3.4 (δ-Correctness). *Let* $\delta = \delta(\kappa)$*. We say that* $\mathsf{PKE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *is* δ-correct *if*

$$\mathrm{Ex}_{(ek,dk)\leftarrow\mathsf{Gen}(1^\kappa)}\left[\max_{m\in\mathcal{M}} \Pr[\mathsf{Dec}(dk, c) = m \mid c \leftarrow \mathsf{Enc}(ek, m)]\right] \leq \delta(\kappa).$$

In particular, we just say perfeclty correct if $\delta = 0$*.*

We say that a key pair (ek, dk) is accurate if $\Pr[\mathsf{Dec}(dk, c) = m \mid c \leftarrow \mathsf{Enc}(ek, m)] = 1$ for any $m \in \mathcal{M}$. We observe that if PKE is deterministic, then δ-correctness implies that

$$\mathrm{Ex}_{(ek,dk)\leftarrow\mathsf{Gen}(1^\kappa)}[ek \text{ is inaccurate}] \leq \delta(\kappa).$$

In other words, if PKE is deterministic and δ-correct, then a key pair is accurate with probability $\geq 1 - \delta$.

Security Notions: We define onewayness under chosen-plaintext attacks (OW-CPA), indistinguishability under chosen-plaintext attacks (IND-CPA), and indistinguishability under chosen-ciphertext attacks (IND-CCA) for a PKE.

Definition 3.5 (Security notions for PKE). *Let* $\mathcal{D}_\mathcal{M}$ *be a distribution over the message space* $\mathcal{M}$*. For any adversary* $\mathcal{A}$*, we define its* OW-CPA*,* IND-CPA*, and* IND-CCA *advantages against a PKE scheme* $\mathsf{PKE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *as follows:*

$$\begin{aligned}
\mathrm{Adv}^{\text{ow-cpa}}_{\mathcal{A},\mathcal{D}_\mathcal{M},\mathsf{PKE}}(\kappa) &:= \Pr[\mathrm{Expt}^{\text{ow-cpa}}_{\mathsf{PKE},\mathcal{D}_\mathcal{M},\mathcal{A}}(\kappa) = 1],\\
\mathrm{Adv}^{\text{ind-cpa}}_{\mathsf{PKE},\mathcal{A}}(\kappa) &:= \left|2\Pr[\mathrm{Expt}^{\text{ind-cpa}}_{\mathsf{PKE},\mathcal{A}}(\kappa) = 1] - 1\right|,\\
\mathrm{Adv}^{\text{ind-cca}}_{\mathsf{PKE},\mathcal{A}}(\kappa) &:= \left|2\Pr[\mathrm{Expt}^{\text{ind-cca}}_{\mathsf{PKE},\mathcal{A}}(\kappa) = 1] - 1\right|,
\end{aligned}$$

where $\mathrm{Expt}^{\text{ow-cpa}}_{\mathsf{PKE},\mathcal{D}_\mathcal{M},\mathcal{A}}(\kappa)$*,* $\mathrm{Expt}^{\text{ind-cpa}}_{\mathsf{PKE},\mathcal{A}}(\kappa)$*, and* $\mathrm{Expt}^{\text{ind-cca}}_{\mathsf{PKE},\mathcal{A}}(\kappa)$ *are experiments described in Figure 1. For* GOAL-ATK $\in$ $\{$OW-CPA, IND-CPA, IND-CCA$\}$*, we say that* PKE *is* GOAL-ATK*-secure if* $\mathrm{Adv}^{\text{goal-atk}}_{\mathcal{A},\mathsf{PKE}}(\kappa)$ *is negligible for any PPT adversary* $\mathcal{A}$*. We omit* $\mathcal{D}_\mathcal{M}$ *from* OW-CPA *security if* $\mathcal{D}_\mathcal{M}$ *is the uniform distribution over* $\mathcal{M}$*.*

```
$\mathsf{Expt}^{\text{ow-cpa}}_{\mathsf{PKE},\mathcal{D}_\mathcal{M},\mathcal{A}}(\kappa)$
    $(ek, dk) \leftarrow \mathsf{Gen}(1^\kappa)$
    $m^* \leftarrow \mathcal{D}_\mathcal{M}$
    $c^* \leftarrow \mathsf{Enc}(ek, m^*)$
    $m' \leftarrow \mathcal{A}(ek, c^*)$
    return $\mathsf{boole}(m' \overset{?}{=} \mathsf{Dec}(dk, c^*))$
```

```
$\mathsf{Expt}^{\text{ind-cpa}}_{\mathsf{PKE},\mathcal{A}}(\kappa)$
    $b \leftarrow \{0, 1\}$
    $(ek, dk) \leftarrow \mathsf{Gen}(1^\kappa)$
    $(m_0, m_1, st) \leftarrow \mathcal{A}_1(ek)$
    $c^* \leftarrow \mathsf{Enc}(ek, m_b)$
    $b' \leftarrow \mathcal{A}_2(c^*, st)$
    return $\mathsf{boole}(b' \overset{?}{=} b)$
```

```
$\mathsf{Expt}^{\text{ind-cca}}_{\mathsf{PKE},\mathcal{A}}(\kappa)$
    $b \leftarrow \{0, 1\}$
    $(ek, dk) \leftarrow \mathsf{Gen}(1^\kappa)$
    $(m_0, m_1, st) \leftarrow \mathcal{A}_1^{\textsc{Dec}_\perp(\cdot)}(ek)$
    $c^* \leftarrow \mathsf{Enc}(ek, m_b)$
    $b' \leftarrow \mathcal{A}_2^{\textsc{Dec}_{c^*}(\cdot)}(c^*, st)$
    return $\mathsf{boole}(b' \overset{?}{=} b)$
```

```
$\textsc{Dec}_a(c)$
    if $c = a$, return $\perp$
    $m := \mathsf{Dec}(dk, c)$
    return $m$
```

Fig. 1: Games for PKE schemes

Disjoint Simulatability Saito et al. defined *disjoint simulatability* of DPKE [SXY18]. Intuitively speaking, a DPKE scheme is disjoint simulatable if there exists a simulator that is only given a public key and generates a “fake ciphertext” that is indistinguishable from a real ciphertext of a random message. Moreover, we require that a fake ciphertext falls in a valid ciphertext space with negligible probability. The formal definition is as follows.

Definition 3.6 (Disjoint simulatability [SXY18]). *Let $\mathcal{D}_\mathcal{M}$ denote an efficiently sampleable distribution on a set $\mathcal{M}$. A deterministic PKE scheme $\mathsf{PKE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ with plaintext and ciphertext spaces $\mathcal{M}$ and $\mathcal{C}$ is $\mathcal{D}_\mathcal{M}$-*disjoint simulatable *if there exists a PPT algorithm $\mathcal{S}$ that satisfies the following.*

- *(Statistical disjointness:)*

$$\mathsf{Disj}_{\mathsf{PKE},\mathcal{S}}(\kappa) := \max_{(ek,dk)\in\mathsf{Gen}(1^\kappa;\mathcal{R})} \Pr[c \in \mathsf{Enc}(ek, \mathcal{M}) \mid c \leftarrow \mathcal{S}(ek)]$$

  is negligible, where $\mathcal{R}$ denotes a randomness space for Gen.
- *(Ciphertext-indistinguishability:) For any PPT adversary $\mathcal{A}$,*

$$\mathsf{Adv}^{\text{ds-ind}}_{\mathsf{PKE},\mathcal{D}_\mathcal{M},\mathcal{A},\mathcal{S}}(\kappa) := \left| \begin{array}{l} \Pr\left[\mathcal{A}(ek, c^*) \to 1 \;\middle|\; \begin{array}{c} (ek, dk) \leftarrow \mathsf{Gen}(1^\kappa); m^* \leftarrow \mathcal{D}_\mathcal{M}; \\ c^* := \mathsf{Enc}(ek, m^*) \end{array}\right] \\ - \Pr\left[\mathcal{A}(ek, c^*) \to 1 \mid (ek, dk) \leftarrow \mathsf{Gen}(1^\kappa); c^* \leftarrow \mathcal{S}(ek)\right] \end{array} \right|$$

  is negligible.

IND-QCCA In [BZ13], Boneh and Zhandry showed that we cannot quantumize the challenge oracle; They showed that indistinguishability against fully-quantum chosen-plaintext attack (fqCPA) is impossible, in which an adversary guesses b by asking quantum challenges $\sum_{m_0,m_1,z} \phi_{m_0,m_1,z} \left|m_0, m_1, z\right\rangle$ and obtaining $\sum_{m_0,m_1,z} \phi_{m_0,m_1,z} \left|m_0, m_1, z \oplus \mathsf{Enc}(\right.$ They also show that indistinguishability against fully-quantum chosen-left-right-plaintext attack (fqlrCPA) is impossible, in which an adversary is allowed to ask quantum challenges $\sum_{m_0,m_1,z_0,z_1} \phi_{m_0,m_1,z_0,z_1} \left|m_0, m_1, z_0, z_1\right\rangle$ and obtain $\sum_{m_0,m_1,z_0,z_1} \phi_{m_0,m_1,z_0,z_1} \left|m_0, m_1, z_0 \oplus \mathsf{Enc}(k, m_b; r_0), z_1 \oplus \mathsf{Enc}(k, m_{1-b}; r_1)\right\rangle$.

Thus, we do not quantumize the challenge oracle, but quantumize decryption oracle.

We will need to define the result of $m \oplus \perp$, where $\perp \notin \mathcal{M}$. In order to do so, we encode $\perp$ as a bit string not in the message space.

Definition 3.7 (IND-qCCA for PKE [BZ13]). *For any adversary $\mathcal{A}$, we define its* IND-qCCA *advantages against a PKE scheme* $\mathsf{PKE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *as follows:*

$$\mathsf{Adv}^{\text{ind-qcca}}_{\mathsf{PKE},\mathcal{A}}(\kappa) := \left|2\Pr[\mathsf{Expt}^{\text{ind-qcca}}_{\mathsf{PKE},\mathcal{A}}(\kappa) = 1] - 1\right|,$$

where $\mathsf{Expt}^{\text{ind-qcca}}_{\mathsf{PKE},\mathcal{A}}(\kappa)$ is an experiment described in Figure 2. We say that PKE *is* IND-qCCA*-secure if $\mathsf{Adv}^{\text{ind-qcca}}_{\mathcal{A},\mathsf{PKE}}(\kappa)$ is negligible for any PPT adversary $\mathcal{A}$.*

```
$\mathsf{Expt}^{\text{ind-qcca}}_{\mathsf{PKE},\mathcal{A}}(\kappa)$
    $b \leftarrow \{0,1\}$
    $(ek, dk) \leftarrow \mathsf{Gen}(1^\kappa)$
    $(m_0, m_1, st) \leftarrow \mathcal{A}_1^{\textsc{qDec}_\perp(\cdot)}(ek)$
    $c^* \leftarrow \mathsf{Enc}(ek, m_b)$
    $b' \leftarrow \mathcal{A}_2^{\textsc{qDec}_{c^*}(\cdot)}(c^*, st)$
    return $\mathsf{boole}(b' \overset{?}{=} b)$

$\textsc{qDec}_a(|c, z\rangle)$
    $m := \mathsf{Dec}(dk, c)$
    if $c = a$, $m := \perp$
    return $|c, z \oplus m\rangle$
```

Fig. 2: More Games for PKE schemes

3.2 Key Encapsulation

The model for KEM schemes is summarized as follows:

Definition 3.8. *A KEM scheme* KEM *consists of the following triple of polynomial-time algorithms* (Gen, Encaps, Decaps)*:*

- $\mathsf{Gen}(1^\kappa; r_g) \to (ek, dk)$: *a key-generation algorithm that on input* 1^κ*, where* κ *is the security parameter, outputs a pair of keys* (ek, dk)*.* ek *and* dk *are called the encapsulation key and decapsulation key, respectively.*
- $\mathsf{Encaps}(ek; r_e) \to (c, K)$: *an encapsulation algorithm that takes as input encapsulation key* ek *and outputs ciphertext* $c \in \mathcal{C}$ *and key* $K \in \mathcal{K}$*.*
- $\mathsf{Decaps}(dk, c) \to K/\perp$: *a decapsulation algorithm that takes as input decapsulation key* dk *and ciphertext* c *and outputs key* K *or a rejection symbol* $\perp \notin \mathcal{K}$*.*

Definition 3.9 (Correctness). *We say* KEM = (Gen, Encaps, Decaps) *has* perfect correctness *if for any* (ek, dk) *generated by* Gen*, we have that*

$$\Pr[\mathsf{Decaps}(dk, c) = K : (c, K) \leftarrow \mathsf{Encaps}(ek)] = 1.$$

Security: We define indistinguishability under chosen-plaintext and chosen-ciphertext attacks (denoted by IND-CPA and IND-CCA) for KEM, respectively.

Definition 3.10. *For any adversary* $\mathcal{A}$*, we define its* IND-CPA *and* IND-CCA *advantages against a KEM scheme* KEM = (Gen, Encaps, Decaps) *as follows:*

$$\mathsf{Adv}^{\text{ind-cpa}}_{\mathsf{KEM},\mathcal{A}}(\kappa) := \left| 2\Pr[\mathsf{Expt}^{\text{ind-cpa}}_{\mathsf{KEM},\mathcal{A}}(\kappa) = 1] - 1 \right|,$$
$$\mathsf{Adv}^{\text{ind-cca}}_{\mathsf{KEM},\mathcal{A}}(\kappa) := \left| 2\Pr[\mathsf{Expt}^{\text{ind-cca}}_{\mathsf{KEM},\mathcal{A}}(\kappa) = 1] - 1 \right|,$$

where $\mathsf{Expt}^{\text{ind-cpa}}_{\mathsf{KEM},\mathcal{A}}(\kappa)$ *and* $\mathsf{Expt}^{\text{ind-cca}}_{\mathsf{KEM},\mathcal{A}}(\kappa)$ *are experiments described in Figure 3.*

For ATK ∈ {CPA, CCA}*, we say that* KEM *is* IND-ATK*-secure if* $\mathsf{Adv}^{\text{ind-atk}}_{\mathcal{A},\mathsf{PKE}}(\kappa)$ *is negligible for any PPT adversary* $\mathcal{A}$*.*

IND-qCCA We also define indistinguishability under *quantum chosen-ciphertext attacks* (denoted by IND-qCCA) for KEM by follwoing [BZ13].

Definition 3.11 (IND-qCCA **for KEM).** *For any adversary* $\mathcal{A}$*, we define its* IND-qCCA *advantage against a KEM scheme* KEM = (Gen, Encaps, Decaps) *as follows:*

$$\mathsf{Adv}^{\text{ind-qcca}}_{\mathsf{KEM},\mathcal{A}}(\kappa) := \left| 2\Pr[\mathsf{Expt}^{\text{ind-qcca}}_{\mathsf{KEM},\mathcal{A}}(\kappa) = 1] - 1 \right|,$$

where $\mathsf{Expt}^{\text{ind-qcca}}_{\mathsf{KEM},\mathcal{A}}(\kappa)$ *is an experiment described in Figure 4.*

We say that KEM *is* IND-qCCA*-secure if* $\mathsf{Adv}^{\text{ind-qcca}}_{\mathcal{A},\mathsf{PKE}}(\kappa)$ *is negligible for any PPT adversary* $\mathcal{A}$*.*

```
$\mathsf{Expt}^{\text{ind-cpa}}_{\mathsf{KEM},\mathcal{A}}(\kappa)$
$b \leftarrow \{0,1\}$
$(ek, dk) \leftarrow \mathsf{Gen}(1^\kappa)$
$(c^*, K_0^*) \leftarrow \mathsf{Encaps}(ek);$
$K_1^* \leftarrow \mathcal{K}$
$b' \leftarrow \mathcal{A}(ek, c^*, K_b^*)$
return $\mathsf{boole}(b' \overset{?}{=} b)$
```

```
$\mathsf{Expt}^{\text{ind-cca}}_{\mathsf{KEM},\mathcal{A}}(\kappa)$
$b \leftarrow \{0,1\}$
$(ek, dk) \leftarrow \mathsf{Gen}(1^\kappa)$
$(c^*, K_0^*) \leftarrow \mathsf{Encaps}(ek);$
$K_1^* \leftarrow \mathcal{K}$
$b' \leftarrow \mathcal{A}^{\textsc{Dec}_{c^*}(\cdot)}(ek, c^*, K_b^*)$
return $\mathsf{boole}(b' \overset{?}{=} b)$
```

```
$\textsc{Dec}_{c^*}(c)$
if $c = c^*$, return $\bot$
$K := \mathsf{Decaps}(dk, c)$
return $K$
```

Fig. 3: Games for KEM schemes

```
$\mathsf{Expt}^{\text{ind-qcca}}_{\mathsf{KEM},\mathcal{A}}(\kappa)$
$b \leftarrow \{0,1\}$
$(ek, dk) \leftarrow \mathsf{Gen}(1^\kappa)$
$(c^*, K_0^*) \leftarrow \mathsf{Encaps}(ek);$
$K_1^* \leftarrow \mathcal{K}$
$b' \leftarrow \mathcal{A}^{\textsc{qDec}_{c^*}(\cdot)}(ek, c^*, K_b^*)$
return $\mathsf{boole}(b' \overset{?}{=} b)$
```

```
$\textsc{qDec}_{c^*}(|c, z\rangle)$
$K := \mathsf{Decaps}(dk, c)$
if $c = c^*$, set $K := \bot$
return $|c, z \oplus K\rangle$
```

Fig. 4: More Games for KEM schemes

3.3 Data Encapsulation

The model for DEM schemes is summarized as follows:

Definition 3.12. *A DEM scheme* DEM *consists of the following triple of polynomial-time algorithms* (E, D)*:*

- $\mathsf{E}(K, m) \rightarrow d$*: an encapsulation algorithm that takes as input key* K *and data* M *and outputs ciphertext* d*.*
- $\mathsf{D}(K, d) \rightarrow m/\bot$*: a decapsulation algorithm that takes as input key* K *and ciphertext* c *and outputs data* m *or a rejection symbol* $\bot \notin \mathcal{M}$*.*

Definition 3.13 (Correctness). *We say* $\mathsf{DEM} = (\mathsf{E}, \mathsf{D})$ *has* perfect correctness *if for any* $K \in \mathcal{K}$*, we have that*

$$\Pr[\mathsf{D}(K, c) = m : d \leftarrow \mathsf{E}(K, m)] = 1.$$

Boneh and Zhandry [BZ13] defined the IND-qCCA security of DEM. They also showed that a combination of two quantumly-secure PRFs yields IND-qCCA-secure DEM. Thus, in the QROM, we easily have IND-qCCA-secure DEM using two quantum random oracles.

Soukharev, Jao, and Seshadri [SJS16] studied Encrypt-then-Mac construction in quantum setting and showed that $\mathsf{DEM} = \mathsf{EtM}[\mathsf{SKE}, \mathsf{MAC}]$ is IND-qCCA-secure if SKE is INDqCPA and MAC is SUF-qCMA, which is left as open problem in Boneh and Zhandry [BZ13].

3.4 Hybrid Encryption

It is obvious that IND-qCCA-secure KEM and IND-qCCA-secure DEM yield IND-qCCA-secure PKE. That is, the proof of Cramer and Shoup [CS03] goes through even for the quantum setting. We omit the detail.

4 IND-qCCA Security of SXY

We show that $\mathsf{KEM} := \mathsf{SXY}[\mathsf{PKE}_1, \mathsf{H}, \mathsf{H}']$ is IND-qCCA-secure if the underlying PKE_1 is a disjoint simulatable DPKE. Let $\mathsf{PKE}_1 = (\mathsf{Gen}_1, \mathsf{Enc}_1, \mathsf{Dec}_1)$ be a deterministic PKE scheme and let $\mathsf{H}\colon \mathcal{M} \rightarrow \mathcal{K}$ and $\mathsf{H}'\colon \{0,1\}^\ell \times \mathcal{C} \rightarrow \mathcal{K}$ be random oracles. We review the conversion SXY in Figure 5.

```
$\underline{\mathsf{Gen}(1^\kappa)}$
    $(ek', dk') \leftarrow \mathsf{Gen}_1(1^\kappa)$
    $s \leftarrow \{0,1\}^\ell$
    $dk \leftarrow (dk', ek', s)$
    return $(ek', dk)$
```

```
$\underline{\mathsf{Enc}(ek')}$
    $m \leftarrow \mathcal{D}_\mathcal{M}$
    $c := \mathsf{Enc}_1(ek', m)$
    $K := \mathsf{H}(m)$
    return $(K, c)$
```

```
$\underline{\overline{\mathsf{Dec}}(dk, c)\text{, where } dk = (dk', ek', s)}$
    $m := \mathsf{Dec}_1(dk', c)$
    if $m = \bot$, return $K := \mathsf{H}'(s, c)$
    if $c \neq \mathsf{Enc}_1(ek', m)$, return $K := \mathsf{H}'(s, c)$
    else return $K := \mathsf{H}(m)$
```

Fig. 5: $\mathsf{KEM} := \mathsf{SXY}[\mathsf{PKE}_1, \mathsf{H}, \mathsf{H}']$.

Theorem 4.1 (Security of SXY in the ROM (an adapted version of [HHK17, Theorem 3.6])). *Let PKE_1 be a perfectly correct DPKE scheme. For any IND-CCA adversary $\mathcal{A}$ against KEM issuing q_H and $q_{\mathsf{H}'}$ quantum random oracle queries to H and H' and $q_{\overline{\mathsf{Dec}}}$ decryption queries, there exists an OW-CPA adversary $\mathcal{B}$ against PKE_1, such that*

$$\mathsf{Adv}^{\text{ind-cca}}_{\mathsf{KEM},\mathcal{A}}(\kappa) \leq \mathsf{Adv}^{\text{ow-cpa}}_{\mathsf{PKE}_1,\mathcal{B}}(\kappa) + q_{\mathsf{H}'} \cdot 2^{-\ell}$$

and $\mathsf{Time}(\mathcal{B}) \approx \mathsf{Time}(\mathcal{A}) + q_\mathsf{H} \cdot \mathsf{Time}(\mathsf{Enc}_1) + (q_\mathsf{H} + q_{\mathsf{H}'} + q_{\overline{\mathsf{Dec}}}) \cdot t_\mathsf{CRO}$*, where t_CRO is the running time to simulate the classical random oracle.*

Theorem 4.2 (IND-CCA **Security of** SXY **in the QROM ([SXY18])).** *Let PKE_1 be a perfectly correct DPKE scheme that satisfies the $\mathcal{D}_\mathcal{M}$-disjoint simulatability with a simulator $\mathcal{S}$. For any IND-CCA quantum adversary $\mathcal{A}$ against KEM issuing q_H and $q_{\mathsf{H}'}$ quantum random oracle queries to H and H' and $q_{\overline{\mathsf{Dec}}}$ decryption queries, there exists an adversary $\mathcal{B}$ against the disjoint simulatability of PKE_1 such that*

$$\mathsf{Adv}^{\text{ind-cca}}_{\mathsf{KEM},\mathcal{A}}(\kappa) \leq \mathsf{Adv}^{\text{ds-ind}}_{\mathsf{PKE}_1,\mathcal{D}_\mathcal{M},\mathcal{S},\mathcal{B}}(\kappa) + \mathsf{Disj}_{\mathsf{PKE}_1,\mathcal{S}}(\kappa) + q_{\mathsf{H}'} \cdot 2^{\frac{-\ell+1}{2}}$$

and $\mathsf{Time}(\mathcal{B}) \approx \mathsf{Time}(\mathcal{A}) + q_\mathsf{H} \cdot \mathsf{Time}(\mathsf{Enc}_1) + (q_\mathsf{H} + q_{\mathsf{H}'} + q_{\overline{\mathsf{Dec}}}) \cdot t_\mathsf{RO}$.

Theorem 4.3 (IND-QCCA **security of** SXY **in the QROM).** *Let PKE_1 be a perfectly correct DPKE scheme that satisfies the $\mathcal{D}_\mathcal{M}$-disjoint simulatability with a simulator $\mathcal{S}$. For any IND-QCCA quantum adversary $\mathcal{A}$ against KEM issuing q_H and $q_{\mathsf{H}'}$ quantum random oracle queries to H and H' and $q_{\overline{\mathsf{Dec}}}$ decryption queries, there exists an adversary $\mathcal{B}$ against the disjoint simulatability of PKE_1 such that*

$$\mathsf{Adv}^{\text{ind-qcca}}_{\mathsf{KEM},\mathcal{A}}(\kappa) \leq \mathsf{Adv}^{\text{ds-ind}}_{\mathsf{PKE}_1,\mathcal{D}_\mathcal{M},\mathcal{S},\mathcal{B}}(\kappa) + \mathsf{Disj}_{\mathsf{PKE}_1,\mathcal{S}}(\kappa) + q_{\mathsf{H}'} \cdot 2^{\frac{-\ell+1}{2}}$$

and $\mathsf{Time}(\mathcal{B}) \approx \mathsf{Time}(\mathcal{A}) + q_\mathsf{H} \cdot \mathsf{Time}(\mathsf{Enc}_1) + (q_\mathsf{H} + q_{\mathsf{H}'} + q_{\overline{\mathsf{Dec}}}) \cdot t_\mathsf{RO}$.

The proof of Theorem 4.3 follows. We note that the proof is essentially equivalent to that of Theorem 4.2 except at the final game we require quantum simulation of decapsulation oracle.

Remark 4.1. We can relax the perfect correctness into the δ-correctness for some negligible $\delta = \delta(\kappa)$. Recall that if DPKE is δ-correct, then a key pair is accurate with probability $\geq 1 - \delta$. We can eliminate those inaccurate keys by introducing an additional game. The bound becomes

$$\mathsf{Adv}^{\text{ind-qcca}}_{\mathsf{KEM},\mathcal{A}}(\kappa) \leq \mathsf{Adv}^{\text{ds-ind}}_{\mathsf{PKE}_1,\mathcal{D}_\mathcal{M},\mathcal{S},\mathcal{B}}(\kappa) + \mathsf{Disj}_{\mathsf{PKE}_1,\mathcal{S}}(\kappa) + q_{\mathsf{H}'} \cdot 2^{\frac{-\ell+1}{2}} + \delta.$$

Security Proof. We use game-hopping proof. The overview of all games is given in Table 1.

Game_0*:* This is the original game, $\mathsf{Expt}^{\text{ind-qcca}}_{\mathsf{KEM},\mathcal{A}}(\kappa)$.

Game_1*:* This game is the same as Game_0 except that $\mathsf{H}'(s, c)$ in the decryption oracle is replaced with $\mathsf{H}_q(c)$ where $\mathsf{H}_q \colon \mathcal{C} \to \mathcal{K}$ is another random oracle. We remark that $\mathcal{A}$ is not given direct access to H_q.

$\mathsf{Game}_{1.5}$*:* This game is the same as Game_1 except that the random oracle $\mathsf{H}(\cdot)$ is simulated by $\mathsf{H}'_q(\mathsf{Enc}_1(ek, \cdot))$ where H'_q is yet another random oracle. We remark that a decryption oracle and generation of K_0^* also use $\mathsf{H}'_q(\mathsf{Enc}_1(ek, \cdot))$ as $\mathsf{H}(\cdot)$ and that $\mathcal{A}$ is not given direct access to H'_q.

Table 1: Summary of Games for the Proof of Theorem 4.3

					Decryption of		
Game	H	c^*	K_0^*	K_1^*	valid c	invalid c	justification
Game$_0$	$\mathsf{H}(\cdot)$	$\mathsf{Enc}_1(ek',m^*)$	$\mathsf{H}(m^*)$	random	$\mathsf{H}(m)$	$\mathsf{H}'(s,c)$	
Game$_1$	$\mathsf{H}(\cdot)$	$\mathsf{Enc}_1(ek',m^*)$	$\mathsf{H}(m^*)$	random	$\mathsf{H}(m)$	$\mathsf{H}_q(c)$	Lemma 2.1
Game$_{1.5}$	$\mathsf{H}'_q(\mathsf{Enc}_1(ek',\cdot))$	$\mathsf{Enc}_1(ek',m^*)$	$\mathsf{H}(m^*)$	random	$\mathsf{H}(m)$	$\mathsf{H}_q(c)$	Perfect correctness
Game$_2$	$\mathsf{H}_q(\mathsf{Enc}_1(ek',\cdot))$	$\mathsf{Enc}_1(ek',m^*)$	$\mathsf{H}(m^*)$	random	$\mathsf{H}(m)$	$\mathsf{H}_q(c)$	Conceptual
Game$_3$	$\mathsf{H}_q(\mathsf{Enc}_1(ek',\cdot))$	$\mathsf{Enc}_1(ek',m^*)$	$\mathsf{H}_q(c^*)$	random	$\mathsf{H}_q(c)$	$\mathsf{H}_q(c)$	Perfect correctness
Game$_4$	$\mathsf{H}_q(\mathsf{Enc}_1(ek',\cdot))$	$\mathcal{S}(ek')$	$\mathsf{H}_q(c^*)$	random	$\mathsf{H}_q(c)$	$\mathsf{H}_q(c)$	DS-IND

Game$_2$*:* This game is the same as Game$_{1.5}$ except that the random oracle $\mathsf{H}(\cdot)$ is simulated by $\mathsf{H}_q(\mathsf{Enc}_1(ek,\cdot))$ instead of $\mathsf{H}'_q(\mathsf{Enc}_1(ek,\cdot))$. We remark that the decryption oracle and generation of K_0^* also use $\mathsf{H}_q(\mathsf{Enc}_1(ek,\cdot))$ as $\mathsf{H}(\cdot)$.

Game$_3$*:* This game is the same as Game$_2$ except that K_0^* is set as $\mathsf{H}_q(c^*)$ and the decryption oracle always returns $\mathsf{H}_q(c)$ as long as $c \neq c^*$. We denote the modified decryption oracle by qDec'.

Game$_4$*:* This game is the same as Game$_3$ except that c^* is set as $\mathcal{S}(ek')$.

The above completes the descriptions of games. We clearly have

$$\mathsf{Adv}^{\text{ind-qcca}}_{\mathsf{KEM},\mathcal{A}}(\kappa) = |2\Pr[\mathsf{Game}_0 = 1] - 1|$$

by the definition. We upperbound this by the following lemmas.

Lemma 4.1. *We have*

$$|\Pr[\mathsf{Game}_0 = 1] - \Pr[\mathsf{Game}_1 = 1]| \leq q_{\mathsf{H}'} \cdot 2^{\frac{-\ell+1}{2}}.$$

Proof. This is obvious from Lemma 2.1. □

Lemma 4.2. *We have*

$$\Pr[\mathsf{Game}_1 = 1] = \Pr[\mathsf{Game}_{1.5} = 1].$$

Proof. Since we assume that PKE_1 has a perfect correctness, $\mathsf{Enc}_1(ek',\cdot)$ is injective. Therefore, if $\mathsf{H}'_q(\cdot)$ is a random function, then $\mathsf{H}'_q(\mathsf{Enc}_1(ek,\cdot))$ is also a random function. Remarking that access to H'_q is not given to $\mathcal{A}$, it causes no difference from the view of $\mathcal{A}$ if we replace $\mathsf{H}(\cdot)$ with $\mathsf{H}'_q(\mathsf{Enc}_1(ek,\cdot))$. □

Lemma 4.3. *We have*

$$\Pr[\mathsf{Game}_{1.5} = 1] = \Pr[\mathsf{Game}_2 = 1].$$

Proof. We call a ciphertext c valid if we have $\mathsf{Enc}_1(ek',\mathsf{Dec}_1(dk',c)) = c$ and invalid otherwise. We remark that H_q is used only for decrypting an invalid ciphertext c as $\mathsf{H}_q(c)$ in Game$_{1.5}$. This means that a value of $\mathsf{H}_q(c)$ for a valid c is not used at all in Game$_{1.5}$. On the other hand, any output of $\mathsf{Enc}_1(ek',\cdot)$ is valid due to the perfect correctness of PKE_1. Since H'_q is only used for evaluating an output of $\mathsf{Enc}(ek',\cdot)$, a value of $\mathsf{H}_q(c)$ for a valid c is not used at all in Game$_{1.5}$. Hence, it causes no difference from the view of $\mathcal{A}$ if we use the same random oracle H_q instead of two independent random oracles H_q and H'_q. □

Lemma 4.4. *We have*

$$\Pr[\mathsf{Game}_2 = 1] = \Pr[\mathsf{Game}_3 = 1].$$

Proof. Since we set $\mathsf{H}(\cdot) := \mathsf{H}_q(\mathsf{Enc}_1(ek',\cdot))$, for any valid c and $m := \mathsf{Dec}_1(dk',c)$, we have $\mathsf{H}(m) = \mathsf{H}_q(\mathsf{Enc}_1(ek',m)) = \mathsf{H}_q(c)$. Therefore, responses of the decryption oracle are unchanged. We also have $\mathsf{H}(m^*) = \mathsf{H}_q(c^*)$ for a similar reason. □

Lemma 4.5. *There exists an adversary $\mathcal{B}$ such that*

$$|\Pr[\mathsf{Game}_3 = 1] - \Pr[\mathsf{Game}_4 = 1]| \leq \mathsf{Adv}^{\text{ds-ind}}_{\mathsf{PKE}_1,\mathcal{D}_{\mathcal{M}},\mathcal{S},\mathcal{B}}(\kappa).$$

and $\mathsf{Time}(\mathcal{B}) \approx \mathsf{Time}(\mathcal{A}) + q_{\mathsf{H}} \cdot \mathsf{Time}(\mathsf{Enc}_1) + (q_{\mathsf{H}} + q_{\mathsf{H}'} + q_{\overline{\mathsf{Dec}}}) \cdot t_{\mathsf{RO}}$.

Proof. We construct an adversary $\mathcal{B}$, which is allowed to access two random oracles H_q and H', against the disjoint simulatability as follows [3].

$\mathcal{B}^{\mathsf{H}_q,\mathsf{H}'}(ek', c^*)$: It picks $b \leftarrow \{0,1\}$, sets $K_0^* := \mathsf{H}_q(c^*)$ and $K_1^* \leftarrow \mathcal{K}$, and invokes $b' \leftarrow \mathcal{A}^{\mathsf{H},\mathsf{H}',\textsc{qDec}'}(ek', c^*, K_b^*)$ where $\mathcal{A}'s$ oracles are simulated as follows.

- $\mathsf{H}(\cdot)$ is simulated by $\mathsf{H}_q(\mathsf{Enc}_1(ek', \cdot))$.
- H' can be simulated because $\mathcal{B}$ has access to an oracle H'.
- $\textsc{qDec}'(\cdot)$ is simulated by filtering c^* and forwarding to $\mathsf{H}_q(\cdot)$; that is, on input $\sum_{c,z} \phi_{c,z} |c, z\rangle$, $\mathcal{B}$ returns $\sum_{c \neq c^*,z} \phi_{c,z} |c, z \oplus \mathsf{H}_q(c)\rangle + \sum_z \phi_{c^*,z} |c^*, z \oplus \bot\rangle$.

Finally, $\mathcal{B}$ returns $\mathrm{boole}(b \overset{?}{=} b')$.

This completes the description of $\mathcal{B}$. It is easy to see that $\mathcal{B}$ perfectly simulates Game_3 if $c^* = \mathsf{Enc}_1(ek, m^*)$ and Game_4 if $c^* = \mathcal{S}(ek')$. Therefore, we have

$$|\Pr[\mathsf{Game}_3 = 1] - \Pr[\mathsf{Game}_4 = 1]| \leq \mathsf{Adv}^{\text{ds-ind}}_{\mathsf{PKE}_1,\mathcal{D}_{\mathcal{M}},\mathcal{S},\mathcal{B}}(\kappa)$$

as wanted. Since $\mathcal{B}$ invokes $\mathcal{A}$ once, H is simulated by one evaluation of Enc_1 plus one evaluation of a random oracle, and H' and $\textsc{qDec}'$ are simulated by one evaluation of random oracles, we have $\mathsf{Time}(\mathcal{B}) \approx \mathsf{Time}(\mathcal{A}) + q_{\mathsf{H}} \cdot \mathsf{Time}(\mathsf{Enc}_1) + (q_{\mathsf{H}} + q_{\mathsf{H}'} + q_{\overline{\mathsf{Dec}}}) \cdot t_{\mathsf{RO}}$. □

Lemma 4.6. *We have*

$$|2\Pr[\mathsf{Game}_4 = 1] - 1| \leq \mathsf{Disj}_{\mathsf{PKE}_1,\mathcal{S}}(\kappa).$$

Proof. Let Bad denote an event in which $c^* \in \mathsf{Enc}_1(ek', \mathcal{M})$ in Game_4. It is easy to see that we have

$$\Pr[\mathsf{Bad}] \leq \mathsf{Disj}_{\mathsf{PKE}_1,\mathcal{S}}(\kappa).$$

When Bad does not occur, i.e., $c^* \notin \mathsf{Enc}_1(ek', \mathcal{M})$, $\mathcal{A}$ obtains no information about $K_0^* = \mathsf{H}_q(c^*)$. This is because queries to H only reveal $\mathsf{H}_q(c)$ for $c \in \mathsf{Enc}_1(ek', \mathcal{M})$, and $\textsc{qDec}'(c)$ returns $\bot$ if $c = c^*$. Therefore, we have

$$\Pr[\mathsf{Game}_4 = 1 \mid \overline{\mathsf{Bad}}] = 1/2.$$

Combining the above, we have

$$\begin{aligned}
&|2\Pr[\mathsf{Game}_4 = 1] - 1| \\
&= \left|\Pr[\mathsf{Bad}] \cdot (2\Pr[\mathsf{Game}_4 = 1 \mid \mathsf{Bad}] - 1) + \Pr[\overline{\mathsf{Bad}}] \cdot (2\Pr[\mathsf{Game}_4 = 1 \mid \overline{\mathsf{Bad}}] - 1)\right| \\
&\leq \Pr[\mathsf{Bad}] + \left|2\Pr[\mathsf{Game}_4 = 1 \mid \overline{\mathsf{Bad}}] - 1\right| \\
&\leq \mathsf{Disj}_{\mathsf{PKE}_1,\mathcal{S}}(\kappa)
\end{aligned}$$

as we wanted. □

References

ATTU16. Mayuresh Vivekanand Anand, Ehsan Ebrahimi Targhi, Gelo Noel Tabia, and Dominique Unruh. Post-quantum security of the cbc, cfb, ofb, ctr, and XTS modes of operation. In Takagi [Tak16], pages 44–63. 1

BCHK07. Dan Boneh, Ran Canetti, Shai Halevi, and Jonathan Katz. Chosen-ciphertext security from identity-based encryption. *SIAM J. Comput.*, 36(5):1301–1328, 2007. 1

BDF$^+$11. Dan Boneh, Özgür Dagdelen, Marc Fischlin, Anja Lehmann, Christian Schaffner, and Mark Zhandry. Random oracles in a quantum world. pages 41–69, 2011. 1, 2

[3] We allow a reduction algorithm to access the random oracles. See subsection 2.2 for details.

BZ13. Dan Boneh and Mark Zhandry. Secure signatures and chosen ciphertext security in a quantum computing world. pages 361–379, 2013. 1, 4, 5, 6

CS03. Ronald Cramer and Victor Shoup. Design and analysis of practical public-key encryption schemes secure against adaptive chosen ciphertext attack. *SIAM J. Comput.*, 33(1):167–226, 2003. 6

FO13. Eiichiro Fujisaki and Tatsuaki Okamoto. Secure integration of asymmetric and symmetric encryption schemes. 26(1):80–101, January 2013. 2

HHK17. Dennis Hofheinz, Kathrin Hövelmanns, and Eike Kiltz. A modular analysis of the Fujisaki-Okamoto transformation. pages 341–371, 2017. 7

JZC$^+$18. Haodong Jiang, Zhenfeng Zhang, Long Chen, Hong Wang, and Zhi Ma. Ind-cca-secure key encapsulation mechanism in the quantum random oracle model, revisited. In Hovav Shacham and Alexandra Boldyreva, editors, *Advances in Cryptology - CRYPTO 2018 - 38th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2018, Proceedings, Part III*, volume 10993 of *Lecture Notes in Computer Science*, pages 96–125. Springer, 2018. 2

KLLNP16. Marc Kaplan, Gaëtan Leurent, Anthony Leverrier, and María Naya-Plasencia. Breaking symmetric cryptosystems using quantum period finding. pages 207–237, 2016. 1

KLS18. Eike Kiltz, Vadim Lyubashevsky, and Christian Schaffner. A concrete treatment of Fiat-Shamir signatures in the quantum random-oracle model. pages 552–586, 2018. 2

KM12. Hidenori Kuwakado and Masakatu Morii. Security on the quantum-type even-mansour cipher. In *Proceedings of the International Symposium on Information Theory and its Applications, ISITA 2012, Honolulu, HI, USA, October 28-31, 2012*, pages 312–316. IEEE, 2012. 1

NC00. Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, 2000. 2

SJS16. Vladimir Soukharev, David Jao, and Srinath Seshadri. Post-quantum security models for authenticated encryption. In Takagi [Tak16], pages 64–78. 6

SXY18. Tsunekazu Saito, Keita Xagawa, and Takashi Yamakawa. Tightly-secure key-encapsulation mechanism in the quantum random oracle model. pages 520–551, 2018. 1, 2, 4, 7

Tak16. Tsuyoshi Takagi, editor. *Post-Quantum Cryptography - 7th International Workshop, PQCrypto 2016, Fukuoka, Japan, February 24-26, 2016, Proceedings*, volume 9606 of *Lecture Notes in Computer Science*. Springer, 2016. 9, 10

Zha12. Mark Zhandry. Secure identity-based encryption in the quantum random oracle model. pages 758–775, 2012. 2

Zha18. Mark Zhandry. How to record quantum queries, and applications to quantum indifferentiability. *IACR Cryptology ePrint Archive*, 2018:276, 2018. 1, 2