IND-CCA-secure Key Encapsulation Mechanism in the Quantum Random Oracle Model, Revisited

Haodong Jiang [*,†] Zhenfeng Zhang [†,‡] **Long Chen** [†,‡] Hong Wang [*] Zhi Ma [*]

[*]Chinese State Key Laboratory of Mathematical Engineering and Advanced Computing

[†]Institute of Software, Chinese Academy of Sciences

[‡]University of Chinese Academy of Sciences

August 21, 2018

Overview

Background

Public Key Cryptography public key encryption (PKE), digital signatures (DS), and key encapsulation mechanism (KEM)

Background

Public Key Cryptography public key encryption (PKE), digital signatures (DS), and key encapsulation mechanism (KEM)

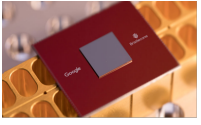
Current Deployment Diffie-Hellman key exchange, the RSA cryptosystem, and elliptic curve cryptosystems

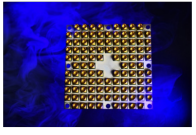
Background

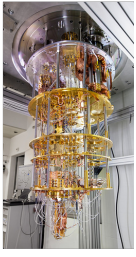
Public Key Cryptography public key encryption (PKE), digital signatures (DS), and key encapsulation mechanism (KEM)

Current Deployment Diffie-Hellman key exchange, the RSA cryptosystem, and elliptic curve cryptosystems

Shor's algorithm







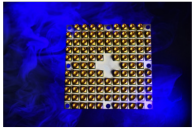
Rapid advance in quantum computing

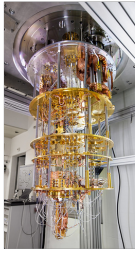
Background

Public Key Cryptography public key encryption (PKE), digital signatures (DS), and key encapsulation mechanism (KEM)

Current Deployment Diffie-Hellman key exchange, the RSA cryptosystem, and elliptic curve cryptosystems

Shor's algorithm





Rapid advance in quantum computing

NIST Post-Quantum Crypto (PQC) “Competition”

The SHIP HAS SAILED! – *Dustin Moody, NIST*

NIST Post-Quantum Crypto (PQC) "Competition"

The SHIP HAS SAILED! – *Dustin Moody, NIST*

- Feb 2016 – NIST report on PQC (NISTIR 8105)
- Dec 2016 – Submission requirements and evaluation criteria
- Nov 2017 – Deadline for Submissions
- Dec 2017 – Round-1-submissions
- Apr 2018 – The 1^{st} NIST PQC standardization conference

Key Encapsulation Mechanism (KEM)

- Among the 69 Round-1 submissions including PKE, DS and KEM, there are 35 proposals for IND-CCA-secure KEM constructions.

Key Encapsulation Mechanism (KEM)

- Among the 69 Round-1 submissions including PKE, DS and KEM, there are 35 proposals for IND-CCA-secure KEM constructions.

Generic transformation (**ROM**) [Den03,**HHK17**] (25/35)

CPA-secure PKE $\Rightarrow$ CCA-secure KEM

Key Encapsulation Mechanism (KEM)

- Among the 69 Round-1 submissions including PKE, DS and KEM, there are 35 proposals for IND-CCA-secure KEM constructions.

Generic transformation (**ROM**) [Den03,**HHK17**] (25/35)

CPA-secure PKE $\Rightarrow$ CCA-secure KEM

1. Fujisaki-Okamoto (FO) transformations: $\mathrm{FO}^{\not\perp}$, $\mathrm{FO}^{\perp}$, $\mathrm{FO}^{\not\perp}_{m}$, $\mathrm{FO}^{\perp}_{m}$, $\mathrm{QFO}^{\not\perp}_{m}$ and $\mathrm{QFO}^{\perp}_{m}$
2. Modular FO transformations: $\mathrm{U}^{\not\perp}$, $\mathrm{U}^{\perp}$, $\mathrm{U}^{\not\perp}_{m}$, $\mathrm{U}^{\perp}_{m}$, $\mathrm{QU}^{\not\perp}_{m}$ and $\mathrm{QU}^{\perp}_{m}$

Quantum random oracle model

- Generic constructions in the **ROM** have gathered renewed interest in post-quantum setting, where adversaries are equipped with a quantum computer.
- In the real world, quantum adversary can execute hash functions (the instantiation of **RO**) on an arbitrary superposition of inputs.
- Therefore, for fully evaluating the post-quantum security, the analysis in the quantum random oracle model (**QROM**), introduced by [BDF+11], is crucial.
- Accordingly, there has been an increased interest in analyzing post-quantum security of classical cryptosystems in the ROM, see [BDF+11, Zha12, DFG13, Son14, Unr15, TU16, HRS16, HHK17, Unr17, KLS18, SXY18].

Generic constructions in the QROM

Generally, QROM is quite difficult to deal with, since many proof techniques in the ROM including adaptive programmability or extractability have no analog in the QROM [BDF+11].

Generic constructions in the QROM

Generally, QROM is quite difficult to deal with, since many proof techniques in the ROM including adaptive programmability or extractability have no analog in the QROM [BDF+11].

- FO transformations: $\mathrm{FO}^{\not\perp}$, $\mathrm{FO}^{\perp}$, $\mathrm{FO}^{\not\perp}_{m}$, $\mathrm{FO}^{\perp}_{m}$, $\mathrm{QFO}^{\not\perp}_{m}$ and $\mathrm{QFO}^{\perp}_{m}$
- Modular FO transformations: $\mathrm{U}^{\not\perp}$, $\mathrm{U}^{\perp}$, $\mathrm{U}^{\not\perp}_{m}$, $\mathrm{U}^{\perp}_{m}$, $\mathrm{QU}^{\not\perp}_{m}$ and $\mathrm{QU}^{\perp}_{m}$

Generic constructions in the QROM

Generally, QROM is quite difficult to deal with, since many proof techniques in the ROM including adaptive programmability or extractability have no analog in the QROM [BDF+11].

- FO transformations: $\mathrm{FO}^{\not\perp}$, $\mathrm{FO}^{\perp}$, $\mathrm{FO}_m^{\not\perp}$, $\mathrm{FO}_m^{\perp}$, $\mathrm{QFO}_m^{\not\perp}$ and $\mathrm{QFO}_m^{\perp}$
- Modular FO transformations: $\mathrm{U}^{\not\perp}$, $\mathrm{U}^{\perp}$, $\mathrm{U}_m^{\not\perp}$, $\mathrm{U}_m^{\perp}$, $\mathrm{QU}_m^{\not\perp}$ and $\mathrm{QU}_m^{\perp}$

The QROM proofs in [HHK17]

1. require an additional length-preserving hash
2. suffer highly non-tight security reductions

Motivation

We revisit the security of FO transformations and modular FO transformations in the QROM with the goal of

1. removing the additional hash
2. making the QROM security reductions tighter

Our results

FO transformations from standard security assumptions

Transformation	Underlying security	Security bound	Additional hash	Perfectly correct?
$\mathrm{QFO}_m^{\not\perp}$ and $\mathrm{QFO}_m^{\perp}$ [HHK17]	OW-CPA	$q\sqrt{q^2\delta + q\sqrt{\epsilon}}$	Y	N
$\mathrm{FO}_m^{\not\perp}$ [SXY18]	IND-CPA	$q\sqrt{\epsilon}$	N	Y
$\mathrm{FO}^{\not\perp}$ and $\mathrm{FO}_m^{\not\perp}$ Our work	OW-CPA	$q\sqrt{\delta} + q\sqrt{\epsilon}$	N	N

Our results

Modular FO transformations from non-standard security assumptions

Transformation	Underlying security	Security bound	Additional hash	DPKE	Perfectly correct?
$\mathrm{QU}_m^{\perp}$ [HHK17]	OW-PCA	$q\sqrt{\epsilon}$	Y	N	N
$\mathrm{QU}_m^{\not\perp}$ [HHK17]	OW-PCA	$q\sqrt{\epsilon}$	Y	N	N
$\mathrm{U}_m^{\not\perp}$ [SXY18]	DS	ϵ	N	Y	Y
$\mathrm{U}^{\not\perp}$ Our work	OW-qPCA	$q\sqrt{\epsilon}$	N	N	N
$\mathrm{U}^{\perp}$ Our work	OW-qPVCA	$q\sqrt{\epsilon}$	N	N	N
$\mathrm{U}_m^{\not\perp}$ Our work	OW-CPA	$q\sqrt{\delta}+q\sqrt{\epsilon}$	N	Y	N
$\mathrm{U}_m^{\not\perp}$ Our work	DS	$q\sqrt{\delta}+\epsilon$	N	Y	N
$\mathrm{U}_m^{\perp}$ Our work	OW-VA	$q\sqrt{\delta}+q\sqrt{\epsilon}$	N	Y	N

List of NIST KEM submissions

List of KEM submissions based on (modular) FO transformations

Proposals	Transformations	Correctness error	DPKE?	QROM consideration?
CRYSTALS-Kyber	$\mathrm{FO}^{\not\perp}$	Y	N	Y
EMBLEM and R.EMBLEM	$\mathrm{QFO}^{\perp}$	Y	N	Y
FrodoKEM	$\mathrm{QFO}^{\not\perp}$	Y	N	Y
KINDI	$\mathrm{QFO}^{\not\perp}_{m}$	Y	N	Y
LAC	$\mathrm{FO}^{\not\perp}$	Y	N	N
Lepton	$\mathrm{QFO}^{\perp}$	Y	N	Y
LIMA	$\mathrm{FO}^{\perp}_{m}$	N	N	Y
Lizard	$\mathrm{QFO}^{\not\perp}$	Y	N	Y
NewHope	$\mathrm{QFO}^{\not\perp}$	Y	N	Y
NTRU-HRSS-KEM	$\mathrm{QFO}^{\perp}_{m}$	N	N	Y
Odd Manhattan	$\mathrm{U}^{\perp}_{m}$	N	N	N
OKCN-AKCN-CNKE	$\mathrm{QFO}^{\not\perp}$	Y	N	Y
Round2	$\mathrm{QFO}^{\not\perp}$	Y	N	Y

List of NIST KEM submissions

List of KEM submissions based on (modular) FO transformations

Proposals	Transformations	Correctness error	DPKE?	QROM consideration?
SABER	$\mathrm{FO}^{\not\perp}$	Y	N	Y
ThreeBears	$\mathrm{FO}_m^{\perp}$	Y	N	Y
Titanium	$\mathrm{QFO}^{\not\perp}$	Y	N	Y
BIG QUAKE	$\mathrm{QFO}^{\perp}$	N	N	Y
Classic McEliece	$\mathrm{U}^{\not\perp}$	N	Y	Y
DAGS	$\mathrm{QFO}_m^{\perp}$	N	N	Y
HQC	$\mathrm{QFO}^{\perp}$	Y	N	Y
LEDAkem	$\mathrm{U}_m^{\not\perp}$	Y	Y	N
LOCKER	$\mathrm{QFO}^{\perp}$	Y	N	Y
QC-MDPC	$\mathrm{QFO}_m^{\perp}$	Y	N	Y
RQC	$\mathrm{QFO}^{\perp}$	N	N	Y
SIKE	$\mathrm{FO}^{\not\perp}$	N	N	N

The application of our results

1. 16 KEM constructions including FrodoKEM etc., can be simplified by cutting off the additional hash and improved in performance with respect to speed and sizes.

The application of our results

1. 16 KEM constructions including FrodoKEM etc., can be simplified by cutting off the additional hash and improved in performance with respect to speed and sizes.
2. Provide a solid post-quantum security guarantee for LAC and SIKE without any additional ciphertext overhead.

The application of our results

1. 16 KEM constructions including FrodoKEM etc., can be simplified by cutting off the additional hash and improved in performance with respect to speed and sizes.
2. Provide a solid post-quantum security guarantee for LAC and SIKE without any additional ciphertext overhead.
3. Modular QROM security analyses not only provide post-quantum security guarantees for Odd Manhattan, Classic McEliece and LEDAkem, but also can help to obtain a variety of combined transformations with different requirements and properties.

Generic Construction $\mathrm{FO}^{\not\perp}$

Gen′

1 : $(pk, sk) \leftarrow Gen$
2 : $s \xleftarrow{\$} \mathcal{M}$
3 : $sk' := (sk, s)$
4 : **return** (pk, sk')

Encaps(pk)

1 : $m \xleftarrow{\$} \mathcal{M}$
2 : $c = Enc(pk, m; G(m))$
3 : $K := H(m, c)$
4 : **return** (K, c)

Decaps(sk', c)

1 : Parse $sk' = (sk, s)$
2 : $m' := Dec(sk, c)$
3 : **if** $Enc(pk, m'; G(m')) = c$
4 : **return** $K := H(m', c)$
5 : **else return**
6 : $K := H(s, c)$

Figure: IND-CCA-secure KEM-I=$\mathrm{FO}^{\not\perp}$[PKE,G,H]

Theorem 3.1

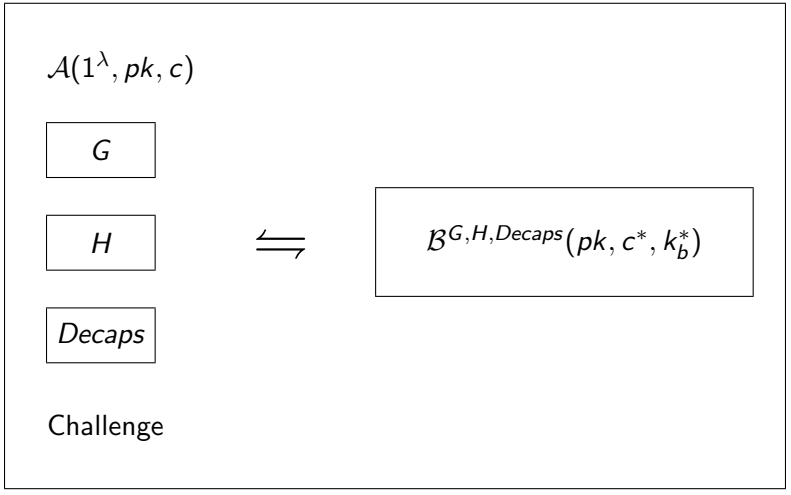
Theorem 3.1 (PKE OW-CPA $\overset{QROM}{\Rightarrow}$ KEM-I IND-CCA).

If PKE *is* δ*-correct, for any IND-CCA* $\mathcal{B}$ *against KEM-I, issuing at most* q_D *queries to the decapsulation oracle* $\textsc{Decaps}$*, at most* q_G *queries to the random oracle* G *and at most* q_H *queries to the random oracle* H*, there exists a OW-CPA adversary* $\mathcal{A}$ *against* PKE *such that*

$$\mathbf{Adv}^{\text{IND-CCA}}_{\text{KEM-I}}(\mathcal{B}) \leq 2q_H \frac{1}{\sqrt{|\mathcal{M}|}} + 4q_G\sqrt{\delta} + 2(q_G + q_H) \cdot \sqrt{\mathbf{Adv}^{\text{OW-CPA}}_{\text{PKE}}(\mathcal{A})}$$

and the running time of $\mathcal{A}$ *is about that of* $\mathcal{B}$*.*

Proof Skeleton of Theorem 3.1



Main Techniques

Removing the additional hash

- In the security proof of FO in the ROM, a RO-query list is used to simulate the decryption oracle.

Main Techniques

Removing the additional hash

- In the security proof of FO in the ROM, a RO-query list is used to simulate the decryption oracle.
- In the QROM, such a RO-query list does not exist due to the fact that there is no way to learn the actual content of adversarial RO queries.

Main Techniques

Removing the additional hash

- In the security proof of FO in the ROM, a RO-query list is used to simulate the decryption oracle.
- In the QROM, such a RO-query list does not exist due to the fact that there is no way to learn the actual content of adversarial RO queries.
- Targhi and Unruh [TU16] circumvented this issue by adding an additional length-preserving hash to the ciphertext.

Main Techniques

Removing the additional hash

- In the security proof of FO in the ROM, a RO-query list is used to simulate the decryption oracle.
- In the QROM, such a RO-query list does not exist due to the fact that there is no way to learn the actual content of adversarial RO queries.
- Targhi and Unruh [TU16] circumvented this issue by adding an additional length-preserving hash to the ciphertext.
- When considering the KEM version of FO, [HHK17] followed the Targhi-Unruh technique to simulate the decapsulation oracle.

Main Techniques

Removing the additional hash

We use a novel method to simulate the decapsulation oracle by associating the RO H (KDF) with a secret RO H' by

$$H = H' \circ g$$

such that

1. g is indistinguishable from an injective function.
2. $H'(\cdot) = \textsc{Decaps}(sk, \cdot)$

Main Techniques

Removing the additional hash

We use a novel method to simulate the decapsulation oracle by associating the RO H (KDF) with a secret RO H' by

$$H = H' \circ g$$

such that

1. g is indistinguishable from an injective function.
2. $H'(\cdot) = \textsc{Decaps}(sk, \cdot)$

In this way, we circumvent the decryption computation. Thereby, there is no need to read the content of adversarial RO queries!

Main Techniques

Tighten the security bound

- In [HHK17], OW-CPA PKE $\Rightarrow$ OW-PCA PKE$'$ $\Rightarrow$ IND-CCA KEM.
- Two instances of the OW2H lemma are required, and lead to quartic security loss.

Main Techniques

Tighten the security bound

- In [HHK17], OW-CPA PKE $\Rightarrow$ OW-PCA PKE$'$ $\Rightarrow$ IND-CCA KEM.
- Two instances of the OW2H lemma are required, and lead to quartic security loss.
- We choose to directly reduce OW-CPA PKE $\Rightarrow$ IND-CCA KEM.

Main Techniques

Tighten the security bound

- In [HHK17], OW-CPA PKE $\Rightarrow$ OW-PCA PKE$'$ $\Rightarrow$ IND-CCA KEM.
- Two instances of the OW2H lemma are required, and lead to quartic security loss.
- We choose to directly reduce OW-CPA PKE $\Rightarrow$ IND-CCA KEM.
- There will be an obstacle for simulator to keep guarantee the consistency of RO and the decapsulation oracle.

Main Techniques

Tighten the security bound

- In [HHK17], OW-CPA PKE $\Rightarrow$ OW-PCA PKE$'$ $\Rightarrow$ IND-CCA KEM.
- Two instances of the OW2H lemma are required, and lead to quartic security loss.
- We choose to directly reduce OW-CPA PKE $\Rightarrow$ IND-CCA KEM.
- There will be an obstacle for simulator to keep guarantee the consistency of RO and the decapsulation oracle.
- We overcome this by developing the OW2H lemma to the case with redundant oracle.

Conclusion

1. We present QROM security reductions for two widely used generic transformations without suffering any ciphertext overhead, with tighter security reduction.
2. Our results can directly apply to NIST Round-1 KEM submissions, and simplify the constructions.
3. Modular security reductions can help to obtain a variety of combined transformations with different requirements and properties.
4. The new technique for proving quantum security will likely be a common method of proving quantum security for certain types of schemes.

Open Problem

1. **Tightness:** Whether can one develop a novel proof technique to obtain a tight reduction in the QROM for $\mathrm{FO}^{\not\perp}$ and $\mathrm{FO}^{\not\perp}_m$ with the standard IND-CPA security assumption of the underlying PKE?
2. **Explicit Rejection:** How can we prove the QROM security of the transformations $\mathrm{FO}^{\not\perp}$ and $\mathrm{FO}^{\not\perp}_m$ with explicit rejection?

References

Den03 Alexander W. Dent, A designers guide to KEMs

BDF+11 Dan Boneh et al., Random oracles in a quantum world

Zha12 Mark Zhandry, Secure Identity-Based Encryption in the Quantum Random Oracle Model

DFG13 Özgür Dagdelen, Marc Fischlin, and Tommaso Gagliardoni, The FiatCShamir Transformation in a Quantum World

Unr15 Dominique Unruh, Non-Interactive Zero-Knowledge Proofs in the Quantum Random Oracle Model

TU16 Ehsan Ebrahimi Targhi and Dominique Unruh, Post-quantum security of the Fujisaki-Okamoto and OAEP transforms

References

Son14 Fang Song, A Note on Quantum Security for Post-Quantum Cryptography

HRS16 Andreas Hülsing, Joost Rijneveld and Fang Song, Mitigating Multi-Target Attacks in Hash-based Signatures

HHK17 Dennis Hofheinz, Kathrin Hövelmanns and Eike Kiltz, A modular analysis of the Fujisaki-Okamoto transformation

Unr17 Dominique Unruh, Post-quantum Security of Fiat-Shamir

KLS18 Eike Kiltz, Vadim Lyubashevsky and Christian Schaffner, A Concrete Treatment of Fiat-Shamir Signatures in the Quantum Random-Oracle Model

SXY18 Tsunekazu Saito, Keita Xagawa and Takashi Yamakawa, Tightly-secure key-encapsulation mechanism in the quantum random oracle model

Thanks for your attention!

Cryptographic Primitives

Definition 4.1 (Public-key encryption).

A public-key encryption scheme $\mathrm{PKE} = (Gen, Enc, Dec)$

- $Gen(1^\lambda) \to (pk, sk)$
- $Enc(pk, m; r) \to c$
- $Dec(sk, c) \to m$

Cryptographic Primitives

Definition 4.1 (Public-key encryption).

A public-key encryption scheme $\mathrm{PKE} = (Gen, Enc, Dec)$

- $Gen(1^\lambda) \to (pk, sk)$
- $Enc(pk, m; r) \to c$
- $Dec(sk, c) \to m$

Definition 4.2 (Key Encapsulation).

A key encapsulation mechanism KEM consists of three algorithms *Gen*, *Encaps* and *Decaps*.

- $Gen(1^\lambda) \to (pk, sk)$
- $Encaps(pk) \to (K, c)$
- $Decaps(sk, c) \to K$

Definition 4.3 (Correctness [HHK17]).

A PKE is δ-*correct* if $E[\max_{m \in \mathcal{M}} \Pr[Dec(sk, c) \neq m : c \leftarrow Enc(pk, m)]] \leq \delta$, where the expectation is taken over $(pk, sk) \leftarrow Gen$.

Definition 4.3 (Correctness [HHK17]).

A PKE is δ-*correct* if $E[\max_{m \in \mathcal{M}} \Pr[Dec(sk, c) \neq m : c \leftarrow Enc(pk, m)]] \leq \delta$, where the expectation is taken over $(pk, sk) \leftarrow Gen$.

Game OW-CPA

1 : $(pk, sk) \leftarrow Gen$

2 : $m^* \xleftarrow{\$} \mathcal{M}$

3 : $c^* \leftarrow Enc(pk, m^*)$

4 : $m' \leftarrow \mathcal{A}(pk, c^*)$

5 : **return** $m' =? m^*$

Figure: Game OW-CPA for PKE.

Cryptographic Primitives

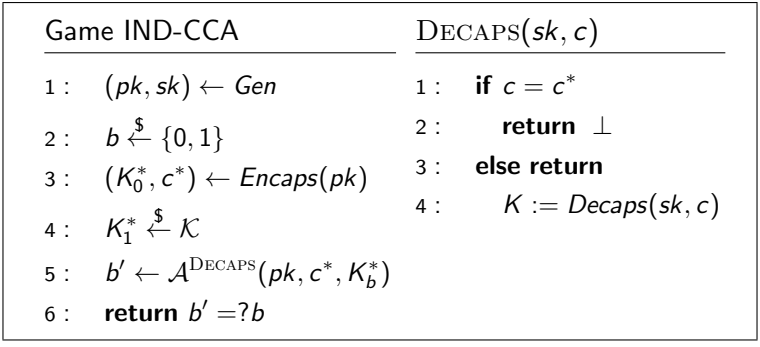


Figure: IND-CCA game for KEM.