# Post-Quantum Anonymity of Kyber

Varun Maram
Applied Cryptography Group
ETH Zurich


Joint work with Keita Xagawa

[Full version of paper: *https://eprint.iacr.org/2022/1696.pdf*]

# Post-Quantum IND-CCA Security and Anonymity of Kyber

Varun Maram
Applied Cryptography Group
ETH Zurich


Joint work with Keita Xagawa

[Full version of paper: *https://eprint.iacr.org/2022/1696.pdf*]

# Background

## PQC Standardization Process: Third Round Candidate Announcement

**NIST is announcing the third round finalists of the NIST Post-Quantum Cryptography Standardization Process. More details are included in NISTIR 8309.**

July 22, 2020

It has been almost a year and a half since the second round of the NIST PQC Standardization Process began. After careful consideration, NIST would like to announce the candidates that will be moving on to the third round.

| Third Round Finalists | Alternate Candidates |
|---|---|
| Public-Key Encryption/KEMs | Public-Key Encryption/KEMs |
| Classic McEliece | BIKE |
| CRYSTALS-KYBER | FrodoKEM |
| NTRU | HQC |
| SABER | NTRU Prime |
| | SIKE |


ORGANIZATIONS

Information Technology Laboratory

Computer Security Division

**Cryptographic Technology Group**

# Background

## PQC Standardization Process: Third Round Candidate Announcement

**NIST is announcing the third round finalists of the NIST Post-Quantum Cryptography Standardization Process. More details are included in NISTIR 8309.**

July 22, 2020

It has been almost a year and a half since the second round of the NIST PQC Standardization Process began. After careful consideration, NIST would like to announce the candidates that will be moving on to the third round.

| Third Round Finalists | Alternate Candidates |
|---|---|
| Public-Key Encryption/KEMs | Public-Key Encryption/KEMs |
| Classic McEliece | BIKE |
| CRYSTALS-KYBER | FrodoKEM |
| NTRU | HQC |
| SABER | NTRU Prime |
| | SIKE |

ORGANIZATIONS

Information Technology Laboratory

Computer Security Division

**Cryptographic Technology Group**

**4.A.2 Security Definition for Encryption/Key-Establishment**

NIST intends to standardize one or more schemes that enable "semantically secure" encryption or key encapsulation with respect to adaptive chosen ciphertext attack, for general use. This property is generally denoted *IND-CCA2 security* in academic literature.

# Background

## Anonymous, Robust Post-quantum Public Key Encryption

Paul Grubbs[1], Varun Maram[2(✉)], and Kenneth G. Paterson[2]

[1] University of Michigan, Ann Arbor, USA
paulgrub@umich.edu
[2] Department of Computer Science, ETH Zurich, Zurich, Switzerland
{vmaram,kenny.paterson}@inf.ethz.ch [Eurocrypt'22]

## Anonymity of NIST PQC Round 3 KEMs

Keita Xagawa[(✉)]

NTT Social Informatics Laboratories, Tokyo, Japan
keita.xagawa.zv@hco.ntt.co.jp [Eurocrypt'22]

# Background

**Anonymous, Robust Post-quantum Public Key Encryption**

Paul Grubbs[1], Varun Maram[2(✉)], and Kenneth G. Paterson[2]

[1] University of Michigan, Ann Arbor, USA
paulgrub@umich.edu
[2] Department of Computer Science, ETH Zurich, Zurich, Switzerland
{vmaram,kenny.paterson}@inf.ethz.ch [Eurocrypt'22]

**Anonymity of NIST PQC Round 3 KEMs**

Keita Xagawa[(✉)]

NTT Social Informatics Laboratories, Tokyo, Japan
keita.xagawa.zv@hco.ntt.co.jp [Eurocrypt'22]

IND-CCA security not sufficient for some modern applications.

# Background

**Anonymous, Robust Post-quantum Public Key Encryption**

Paul Grubbs[1], Varun Maram[2(✉)], and Kenneth G. Paterson[2]

[1] University of Michigan, Ann Arbor, USA
paulgrub@umich.edu
[2] Department of Computer Science, ETH Zurich, Zurich, Switzerland
{vmaram,kenny.paterson}@inf.ethz.ch [Eurocrypt'22]

**Anonymity of NIST PQC Round 3 KEMs**

Keita Xagawa[(✉)]

NTT Social Informatics Laboratories, Tokyo, Japan
keita.xagawa.zv@hco.ntt.co.jp [Eurocrypt'22]

IND-CCA security not sufficient for some modern applications.

E.g., applications like  and

require *anonymity*.

# Background

**Anonymous, Robust Post-quantum Public Key Encryption**

Paul Grubbs[1], Varun Maram[2(✉)], and Kenneth G. Paterson[2]

[1] University of Michigan, Ann Arbor, USA
paulgrub@umich.edu
[2] Department of Computer Science, ETH Zurich, Zurich, Switzerland
{vmaram,kenny.paterson}@inf.ethz.ch [Eurocrypt'22]

**Anonymity of NIST PQC Round 3 KEMs**

Keita Xagawa[(✉)]

NTT Social Informatics Laboratories, Tokyo, Japan
keita.xagawa.zv@hco.ntt.co.jp [Eurocrypt'22]

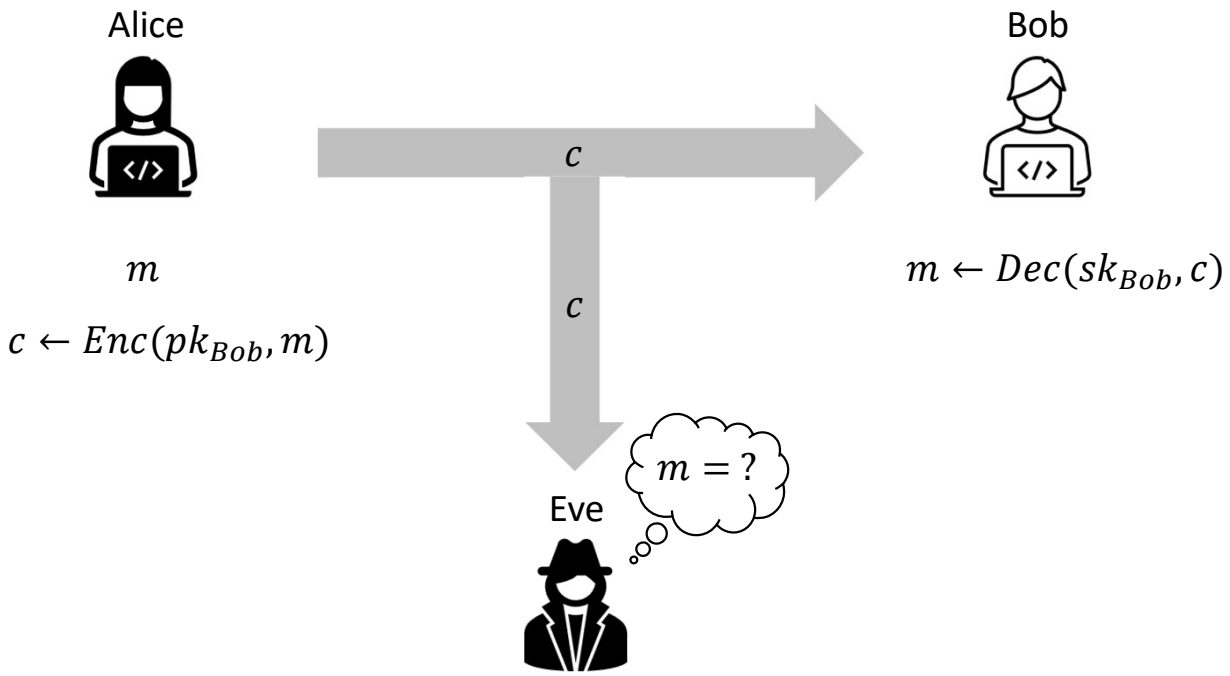
IND-CCA security not sufficient for some modern applications.

E.g., applications like  and

require *anonymity*.

# IND-CCA Security

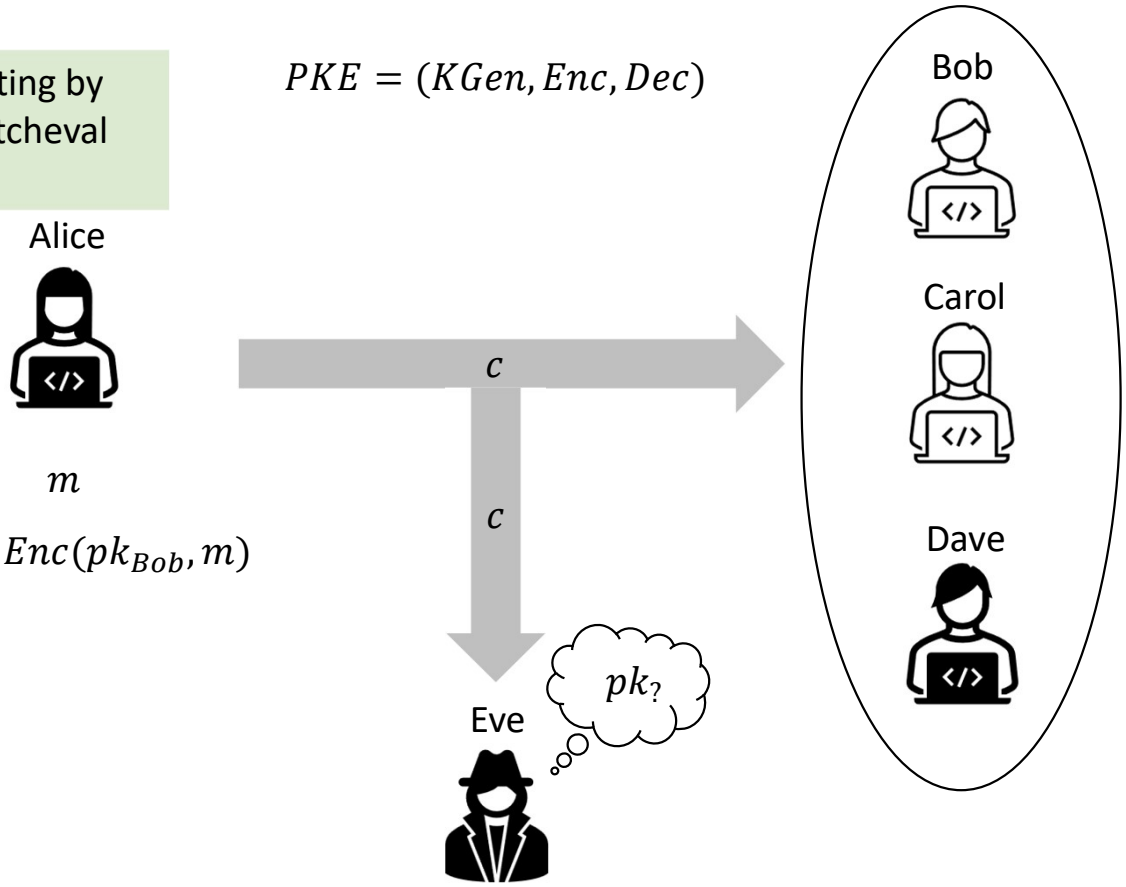
$$PKE = (KGen, Enc, Dec)$$



# Anonymity (ANO-CCA Security)

Formalized in a public-key setting by [Bellare-Boldyreva-Desai-Pointcheval @Asiacrypt'01].

$PKE = (KGen, Enc, Dec)$



# Background

**Anonymous, Robust Post-quantum Public Key Encryption**

Paul Grubbs[1], Varun Maram[2(✉)], and Kenneth G. Paterson[2]

[1] University of Michigan, Ann Arbor, USA
paulgrub@umich.edu
[2] Department of Computer Science, ETH Zurich, Zurich, Switzerland
{vmaram,kenny.paterson}@inf.ethz.ch [Eurocrypt'22]

**Anonymity of NIST PQC Round 3 KEMs**

Keita Xagawa[(✉)]

NTT Social Informatics Laboratories, Tokyo, Japan
keita.xagawa.zv@hco.ntt.co.jp [Eurocrypt'22]

IND-CCA security not sufficient for some modern applications.

E.g., applications like  and

require *anonymity*.

# Background

**Anonymous, Robust Post-quantum Public Key Encryption**

Paul Grubbs[1], Varun Maram[2(✉)], and Kenneth G. Paterson[2]

[1] University of Michigan, Ann Arbor, USA
paulgrub@umich.edu
[2] Department of Computer Science, ETH Zurich, Zurich, Switzerland
{vmaram,kenny.paterson}@inf.ethz.ch [Eurocrypt'22]

**Anonymity of NIST PQC Round 3 KEMs**

Keita Xagawa[(✉)]

NTT Social Informatics Laboratories, Tokyo, Japan
keita.xagawa.zv@hco.ntt.co.jp [Eurocrypt'22]

| | ANO |
|---|---|
| Classic McEliece | |
| NTRU | |
| Kyber | |
| Saber | |

# Fujisaki-Okamoto Transformation

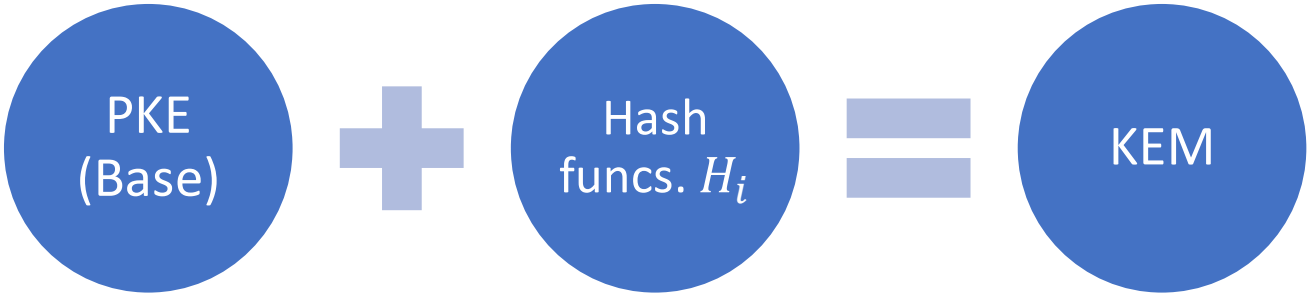

| | ANO |
|---|---|
| Classic McEliece | |
| NTRU | |
| Kyber | |
| Saber | |

# Fujisaki-Okamoto Transformation



| | ANO |
|---|---|
| Classic McEliece | |
| NTRU | |
| Kyber | |
| Saber | |

# Fujisaki-Okamoto Transformation




| | ANO |
|---|---|
| Classic McEliece | |
| NTRU | |
| Kyber | |
| Saber | |

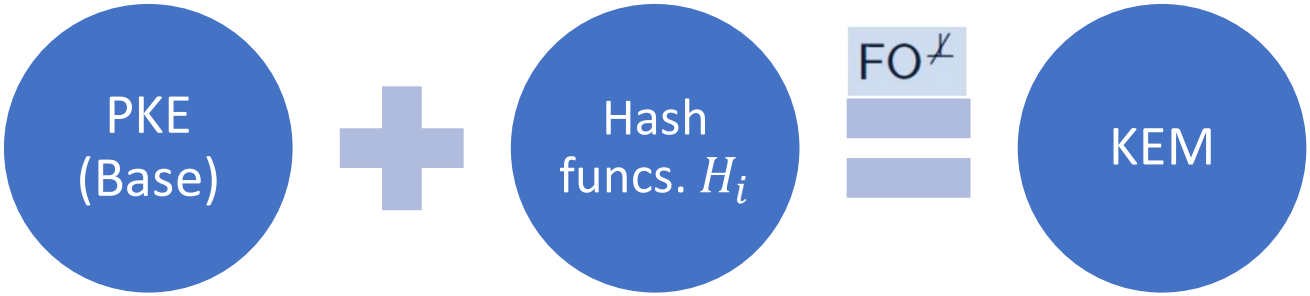
# Fujisaki-Okamoto Transformation

| $\mathsf{KGen}'$ | $\mathsf{Encap}(\mathsf{pk})$ | $\mathsf{Decap}(\mathsf{sk}', c)$ |
|---|---|---|
| $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KGen}$ | $m \leftarrow\!\!\$\, \mathcal{M}$ | Parse $\mathsf{sk}' = (\mathsf{sk}, s)$ |
| $s \leftarrow\!\!\$\, \mathcal{M}$ | $r \leftarrow G_r(m)$ | $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| $\mathsf{sk}' \leftarrow (\mathsf{sk}, s)$ | $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | $r' \leftarrow G_r(m')$ |
| **return** $(\mathsf{pk}, \mathsf{sk}')$ | $\overline{k} \leftarrow G_k(m, c)$ | $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | **return** $(c, \overline{k})$ | **if** $c' = c$ **then** |
| | | **return** $G_k(m', c)$ |
| | | **else return** $G_k(s, c)$ |

$\mathsf{FO}^{\not\perp}$ [Hofheinz-Hövelmanns-Kiltz @TCC'17]

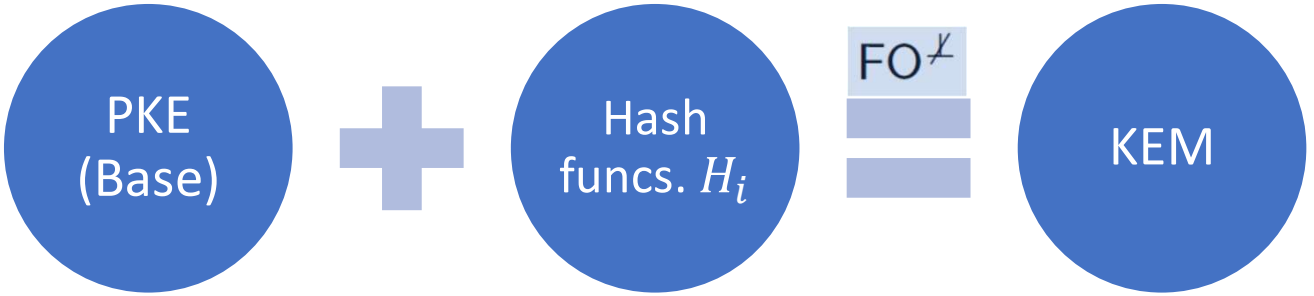
| | ANO |
|---|---|
| Classic McEliece | |
| NTRU | |
| Kyber | |
| Saber | |

# Fujisaki-Okamoto Transformation




| | ANO |
|---|---|
| Classic McEliece | |
| NTRU | |
| Kyber | |
| Saber | |

# Fujisaki-Okamoto Transformation




{OW/IND}-CPA security

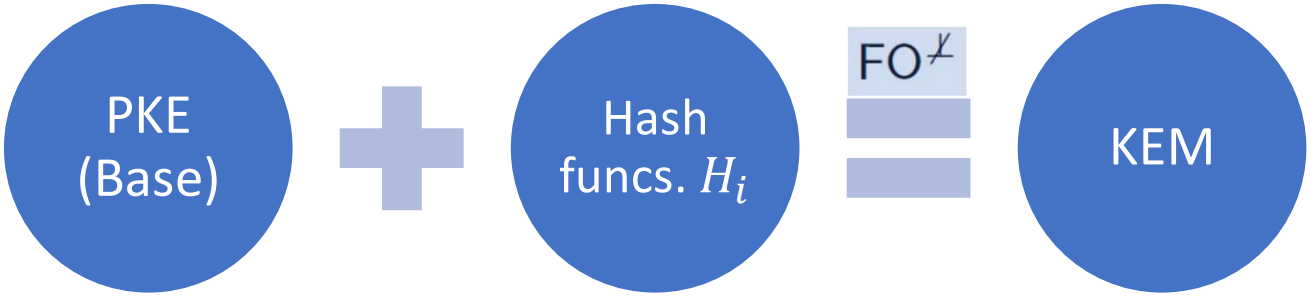
Quantum Random Oracles

IND-CCA security

| | ANO |
|---|---|
| Classic McEliece | |
| NTRU | |
| Kyber | |
| Saber | |

Shown in [Jiang-Zhang-Chen-Wang-Ma@Crypto'18]

# Fujisaki-Okamoto Transformation




{OW/IND}-CPA security + "mild" properties

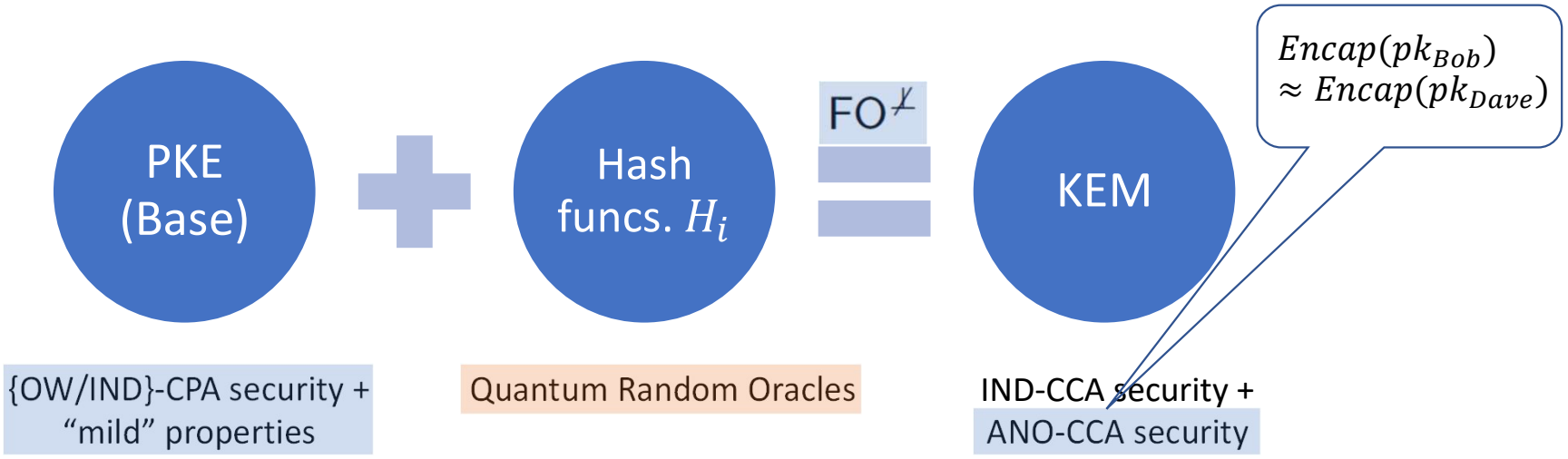
Quantum Random Oracles

IND-CCA security + ANO-CCA security

| | ANO |
|---|---|
| Classic McEliece | |
| NTRU | |
| Kyber | |
| Saber | |

Shown in [Grubbs-Maram-Paterson@Eurocrypt'22] and [Xagawa@Eurocrypt'22]

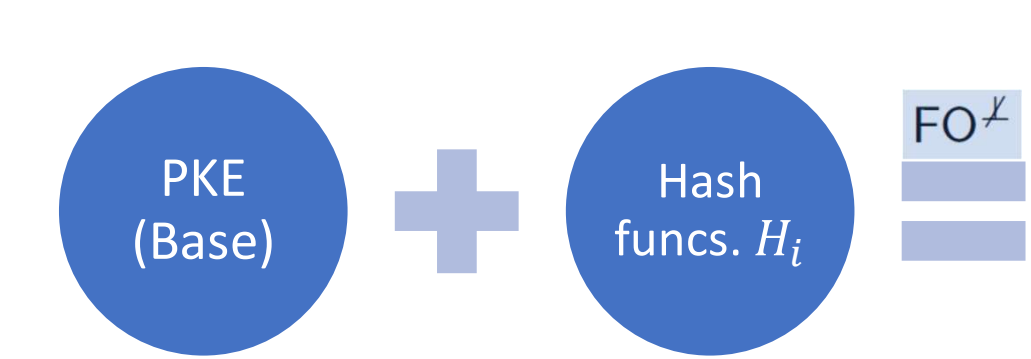
# Fujisaki-Okamoto Transformation




| | ANO |
|---|---|
| Classic McEliece | |
| NTRU | |
| Kyber | |
| Saber | |

Shown in [Grubbs-Maram-Paterson@Eurocrypt'22] and [Xagawa@Eurocrypt'22]

# Fujisaki-Okamoto Transformation




{OW/IND}-CPA security + "mild" properties

Quantum Random Oracles

IND-CCA security + ANO-CCA security

$Encap(pk_{Bob})$
$\approx Encap(pk_{Dave})$

Composing with an appropriate DEM results in ANO-CCA secure hybrid PKE.

| | ANO |
|---|---|
| Classic McEliece | |
| NTRU | |
| Kyber | |
| Saber | |

Shown in [Grubbs-Maram-Paterson@Eurocrypt'22] and [Xagawa@Eurocrypt'22]

# Fujisaki-Okamoto Transformation

| $\mathsf{KGen}'$ | $\mathsf{Encap}(\mathsf{pk})$ | $\mathsf{Decap}(\mathsf{sk}', c)$ |
|---|---|---|
| $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KGen}$ | $m \leftarrow\!\!\$\, \mathcal{M}$ | Parse $\mathsf{sk}' = (\mathsf{sk}, s)$ |
| $s \leftarrow\!\!\$\, \mathcal{M}$ | $r \leftarrow G_r(m)$ | $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| $\mathsf{sk}' \leftarrow (\mathsf{sk}, s)$ | $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | $r' \leftarrow G_r(m')$ |
| **return** $(\mathsf{pk}, \mathsf{sk}')$ | $\overline{k} \leftarrow G_k(m, c)$ | $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | **return** $(c, \overline{k})$ | **if** $c' = c$ **then** |
| | | **return** $G_k(m', c)$ |
| | | **else return** $G_k(s, c)$ |

$\mathsf{FO}^{\not\perp}$ [Hofheinz-Hövelmanns-Kiltz @TCC'17]

| | ANO |
|---|---|
| Classic McEliece | |
| NTRU | |
| Kyber | |
| Saber | |

# Fujisaki-Okamoto Transformation

| $\mathsf{KGen}'$ | $\mathsf{Encap}(\mathsf{pk})$ | $\mathsf{Decap}(\mathsf{sk}', c)$ |
|---|---|---|
| $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KGen}$ | $m \leftarrow\!\!\$\ \mathcal{M}$ | Parse $\mathsf{sk}' = (\mathsf{sk}, s)$ |
| $s \leftarrow\!\!\$\ \mathcal{M}$ | $r \leftarrow G_r(m)$ | $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| $\mathsf{sk}' \leftarrow (\mathsf{sk}, s)$ | $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | $r' \leftarrow G_r(m')$ |
| **return** $(\mathsf{pk}, \mathsf{sk}')$ | $\overline{k} \leftarrow G_k(m, c)$ | $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | **return** $(c, \overline{k})$ | **if** $c' = c$ **then** |
| | | **return** $G_k(m', c)$ |
| | | **else return** $G_k(s, c)$ |

$\mathsf{FO}^{\not\perp}$ [Hofheinz-Hövelmanns-Kiltz @TCC'17]

Uses $\mathsf{FO}^{\not\perp}$ transform, but with an additional plaintext confirmation hash.

| | ANO |
|---|---|
| Classic McEliece | Y |
| NTRU | |
| Kyber | |
| Saber | |

Shown in [Xagawa@Eurocrypt'22]

# Fujisaki-Okamoto Transformation

"key ← hash$(m, c)$"

| KGen′ | | Encap(pk) | | Decap(sk′, $c$) | |
|---|---|---|---|---|---|
| 1 : | (pk, sk) ← KGen | 1 : | $m \leftarrow_{\$} \mathcal{M}$ | 1 : | Parse sk′ = (sk, $s$) |
| 2 : | $s \leftarrow_{\$} \mathcal{M}$ | 2 : | $r \leftarrow G_r(m)$ | 2 : | $m' \leftarrow$ Dec(sk, $c$) |
| 3 : | sk′ ← (sk, $s$) | 3 : | $c \leftarrow$ Enc(pk, $m; r$) | 3 : | $r' \leftarrow G_r(m')$ |
| 4 : | **return** (pk, sk′) | 4 : | $\overline{k} \leftarrow G_k(m, c)$ | 4 : | $c' \leftarrow$ Enc(pk, $m'; r'$) |
| | | 5 : | **return** $(c, \overline{k})$ | 5 : | **if** $c' = c$ **then** |
| | | | | 6 : | **return** $G_k(m', c)$ |
| | | | | 7 : | **else return** $G_k(s, c)$ |

$FO^{\not\perp}$ [Hofheinz-Hövelmanns-Kiltz @TCC'17]

| | ANO |
|---|---|
| Classic McEliece | Y |
| NTRU | |
| Kyber | |
| Saber | |

# Fujisaki-Okamoto Transformation

"key ← hash($m$)"

| KGen′ | Encap(pk) | Decap(sk′, $c$) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse sk′ = (sk, $s$) |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow$ Dec(sk, $c$) |
| 3 : sk′ ← (sk, $s$) | 3 : $c \leftarrow$ Enc(pk, $m; r$) | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow$ Enc(pk, $m'; r'$) |
| | 5 : **return** $(c, \overline{k})$ | 5 : **if** $c' = c$ **then** |
| | | 6 : **return** $G_k(m')$ |
| | | 7 : **else return** $G_k(s, c)$ |

$\mathsf{FO}_m^{\not\perp}$ [Hofheinz-Hövelmanns-Kiltz @TCC'17]

Uses $\mathsf{FO}_m^{\not\perp}$ transform, starting with a deterministic base PKE scheme.

| | ANO |
|---|---|
| Classic McEliece | Y |
| NTRU | Y |
| Kyber | |
| Saber | |

Shown in [Xagawa@Eurocrypt'22]

# Fujisaki-Okamoto Transformation

"key ← hash$(m, c)$"

| KGen′ | Encap(pk) | Decap(sk′, $c$) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse sk′ = (sk, $s$) |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow$ Dec(sk, $c$) |
| 3 : sk′ ← (sk, $s$) | 3 : $c \leftarrow$ Enc(pk, $m; r$) | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m, c)$ | 4 : $c' \leftarrow$ Enc(pk, $m'; r'$) |
| | 5 : **return** $(c, \overline{k})$ | 5 : **if** $c' = c$ **then** |
| | | 6 : **return** $G_k(m', c)$ |
| | | 7 : **else return** $G_k(s, c)$ |

FO$^{\not\perp}$ [Hofheinz-Hövelmanns-Kiltz @TCC'17]

| | ANO |
|---|---|
| Classic McEliece | Y |
| NTRU | Y |
| Kyber | |
| Saber | |

# Fujisaki-Okamoto Transformation

"key ← hash$(m, c)$"

| KGen′ | Encap(pk) | Decap(sk′, $c$) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse sk′ = (sk, $s$) |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow$ Dec(sk, $c$) |
| 3 : sk′ ← (sk, $s$) | 3 : $c \leftarrow$ Enc(pk, $m$; $r$) | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m, c)$ | 4 : $c' \leftarrow$ Enc(pk, $m'$; $r'$) |
| | 5 : **return** $(c, \overline{k})$ | 5 : **if** $c' = c$ **then** |
| | | 6 : **return** $G_k(m', c)$ |
| | | 7 : **else return** $G_k(s, c)$ |

FO$^{\not\perp}$ [Hofheinz-Hövelmanns-Kiltz @TCC'17]

"key ← hash(hash$(m)$, hash$(c)$)"

| KGen′ | Encap(pk) | Decap(sk′, $c$) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse sk′ = (sk, pk, $h$, $s$) |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $m \leftarrow H(m)$ | 2 : $m' \leftarrow$ Dec(sk, $c$) |
| 3 : pk′ ← (pk, $H$(pk)) | 3 : $h \leftarrow H$(pk) | 3 : $(\overline{k'}, r') \leftarrow G_{kr}(m', h)$ |
| 4 : sk′ ← (sk, pk′, $s$) | 4 : $(\overline{k}, r) \leftarrow G_{kr}(m, h)$ | 4 : $c' \leftarrow$ Enc(pk, $m'$; $r'$) |
| 5 : **return** (pk, sk′) | 5 : $c \leftarrow$ Enc(pk, $m$; $r$) | 5 : **if** $c' = c$ **then** |
| | 6 : $k \leftarrow H'(\overline{k}, H(c))$ | 6 : **return** $H'(\overline{k'}, H(c))$ |
| | 7 : **return** $(c, k)$ | 7 : **else return** $H'(s, H(c))$ |

Kyber, Saber

| | ANO |
|---|---|
| Classic McEliece | Y |
| NTRU | Y |
| Kyber | |
| Saber | |

As observed in [Grubbs-Maram-Paterson@Eurocrypt'22] and [Xagawa@Eurocrypt'22]

# Fujisaki-Okamoto Transformation

"key ← hash$(m, c)$"

| KGen′ | Encap(pk) | Decap(sk′, $c$) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse sk′ = (sk, $s$) |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow$ Dec(sk, $c$) |
| 3 : sk′ ← (sk, $s$) | 3 : $c \leftarrow$ Enc(pk, $m$; $r$) | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m, c)$ | 4 : $c' \leftarrow$ Enc(pk, $m'$; $r'$) |
| | 5 : **return** $(c, \overline{k})$ | 5 : **if** $c' = c$ **then** |
| | | 6 : **return** $G_k(m', c)$ |
| | | 7 : **else return** $G_k(s, c)$ |

FO$^{\not\perp}$ [Hofheinz-Hövelmanns-Kiltz @TCC'17]

"key ← hash(hash$(m)$, hash$(c)$)"

| KGen′ | Encap(pk) | Decap(sk′, $c$) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse sk′ = (sk, pk, $h$, $s$) |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $m \leftarrow H(m)$ | 2 : $m' \leftarrow$ Dec(sk, $c$) |
| 3 : pk′ ← (pk, $H$(pk)) | 3 : $h \leftarrow H$(pk) | 3 : $(\overline{k'}, r') \leftarrow G_{kr}(m', h)$ |
| 4 : sk′ ← (sk, pk′, $s$) | 4 : $(\overline{k}, r) \leftarrow G_{kr}(m, h)$ | 4 : $c' \leftarrow$ Enc(pk, $m'$; $r'$) |
| 5 : **return** (pk, sk′) | 5 : $c \leftarrow$ Enc(pk, $m$; $r$) | 5 : **if** $c' = c$ **then** |
| | 6 : $k \leftarrow H'(\overline{k}, H(c))$ | 6 : **return** $H'(\overline{k'}, H(c))$ |
| | 7 : **return** $(c, k)$ | 7 : **else return** $H'(s, H(c))$ |

Kyber, Saber

As observed in [Grubbs-Maram-Paterson@Eurocrypt'22] and [Xagawa@Eurocrypt'22]

This nested hash "$H(c)$" not only acts as a barrier towards establishing anonymity …

| | ANO |
|---|---|
| Classic McEliece | Y |
| NTRU | Y |
| Kyber | ? |
| Saber | ? |

# Fujisaki-Okamoto Transformation

"key ← hash$(m, c)$"

| KGen′ | Encap(pk) | Decap(sk′, $c$) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse sk′ = (sk, $s$) |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow$ Dec(sk, $c$) |
| 3 : sk′ ← (sk, $s$) | 3 : $c \leftarrow$ Enc(pk, $m$; $r$) | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m, c)$ | 4 : $c' \leftarrow$ Enc(pk, $m'$; $r'$) |
| | 5 : **return** $(c, \overline{k})$ | 5 : **if** $c' = c$ **then** |
| | | 6 : **return** $G_k(m', c)$ |
| | | 7 : **else return** $G_k(s, c)$ |

FO$^{\not\perp}$ [Hofheinz-Hövelmanns-Kiltz @TCC'17]

"key ← hash(hash$(m)$, hash$(c)$)"

| KGen′ | Encap(pk) | Decap(sk′, $c$) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse sk′ = (sk, pk, $h$, $s$) |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $m \leftarrow H(m)$ | 2 : $m' \leftarrow$ Dec(sk, $c$) |
| 3 : pk′ ← (pk, $H$(pk)) | 3 : $h \leftarrow H$(pk) | 3 : $(\overline{k'}, r') \leftarrow G_{kr}(m', h)$ |
| 4 : sk′ ← (sk, pk′, $s$) | 4 : $(\overline{k}, r) \leftarrow G_{kr}(m, h)$ | 4 : $c' \leftarrow$ Enc(pk, $m'$; $r'$) |
| 5 : **return** (pk, sk′) | 5 : $c \leftarrow$ Enc(pk, $m$; $r$) | 5 : **if** $c' = c$ **then** |
| | 6 : $k \leftarrow H'(\overline{k}, H(c))$ | 6 : **return** $H'(\overline{k'}, H(c))$ |
| | 7 : **return** $(c, k)$ | 7 : **else return** $H'(s, H(c))$ |

Kyber, Saber

| | ANO | IND |
|---|---|---|
| Classic McEliece | Y | Y |
| NTRU | Y | Y |
| Kyber | ? | ? |
| Saber | ? | ? |

As observed in [Grubbs-Maram-Paterson@Eurocrypt'22] and [Xagawa@Eurocrypt'22]

This nested hash "$H(c)$" not only acts as a barrier towards establishing anonymity …

… but also makes prior IND-CCA security analysis of FO$^{\not\perp}$ in the QROM inapplicable!

# NIST PQC Updates

## NIST Announces First Four Quantum-Resistant Cryptographic Algorithms

**Federal agency reveals the first group of winners from its six-year competition.**

July 05, 2022

**For general encryption,** used when we access secure websites, NIST has selected the CRYSTALS-Kyber algorithm. Among its advantages are comparatively small encryption keys that two parties can exchange easily, as well as its speed of operation.

# NIST PQC Updates

Kyber decisions, part 2: FO transform 692 views

**Peter Schwabe**

to pqc-forum, aut...@pq-crystals.org

Dear all,

This is the second mail about possible tweaks to Kyber as part of the standardization. Kyber as specified in round-3 (and also previous rounds) uses a tweaked Fujisaki-Okamoto transform to build a CCA-secure KEM from a CPA-secure PKE. Specifically, Kyber hashes the hash of the public key into the random coins and the shared key and Kyber hashes the hash of the ciphertext into the shared key. The reasons for those tweaks are the following:

* Hashing the (hash of the) public key into the final key makes the KEM "contributory", i.e., the shared key depends on inputs from both parties;

* hashing the (hash of the) public key into the random coins gives protection against multi-target attacks exploiting decryption failures; and

* hashing the (hash of the) ciphertext into the shared key ensures that this shared key depends on the full transcript.

Through the course of the NIST PQC project, multiple papers considered the FO transform and also the tweaked version used in Kyber. The question for standardization is if the results of these papers should be incorporated into Kyber, or not:

1.) https://eprint.iacr.org/2021/1351 shows that as a protection against multi-target failure attacks it is not necessary to make random coins dependent on the full public key. It is sufficient to hash in a prefix of the public key, if that prefix has sufficiently high min-entropy. We are not aware of any formal definition of a KEM being "contributory", but intuitively also for this property using such a prefix would be sufficient. Using prefix(pk) instead of H(pk) would require fewer Keccak permutations in Kyber and thus speed up encapsulation. Should the Kyber standard use prefix(pk) rather than H(pk)?

2.) Hashing the (hash of the) ciphertext into the final shared key does not help at all with any formal security property or with proofs. On the contrary, hashing the hash of the ciphertext into the final key complicates QROM proofs as pointed out in https://eprint.iacr.org/2021/708.pdf. Removing this tweak simplifies proofs and speeds up encapsulation. Note that decapsulation will still need to compute a hash over the full ciphertext for implicit rejection and, to avoid timing side channels, needs to do so in every decapsulation, not just after a decryption failure. So, there won't be any performance gain in decapsulation. Should the Kyber standard drop hashing the hash of the ciphertext into the shared key?

The obvious disadvantage with both possible changes is that such changes at such a late stage require very careful evaluation. We may have missed some non-standard property that Kyber achieves with these tweaks, but does not without. Also, the modifications are not completely orthogonal to potential modifications of symmetric crypto (see the previous mail), because they require changes to the hashing inside the FO transform with possible consequences for domain separation.

Again, we're looking forward to hear what everybody thinks!

All the best,

The Kyber team

# NIST PQC Updates

Kyber decisions, part 2: FO transform 692 views

**Peter Schwabe**

to pqc-forum, aut...@pq-crystals.org

Dear all,

This is the second mail about possible tweaks to Kyber as part of the standardization. Kyber as specified in round-3 (and also previous rounds) uses a tweaked Fujisaki-Okamoto transform to build a CCA-secure KEM from a CPA-secure PKE. Specifically, Kyber hashes the hash of the public key into the random coins and the shared key and Kyber hashes the hash of the ciphertext into the shared key. The reasons for those tweaks are the following:

* Hashing the (hash of the) public key into the final key makes the KEM "contributory", i.e., the shared key depends on inputs from both parties;

* hashing the (hash of the) public key into the random coins gives protection against multi-target attacks exploiting decryption failures; and

* hashing the (hash of the) ciphertext into the shared key ensures that this shared key depends on the full transcript.

Through the course of the NIST PQC project, multiple papers considered the FO transform and also the tweaked version used in Kyber. The question for standardization is if the results of these papers should be incorporated into Kyber, or not:

1.) https://eprint.iacr.org/2021/1351 shows that as a protection against multi-target failure attacks it is not necessary to make random coins dependent on the full public key. It is sufficient to hash in a prefix of the public key, if that prefix has sufficiently high min-entropy. We are not aware of any formal definition of a KEM being "contributory", but intuitively also for this property using such a prefix would be sufficient. Using prefix(pk) instead of H(pk) would require fewer Keccak permutations in Kyber and thus speed up encapsulation. Should the Kyber standard use prefix(pk) rather than H(pk)?

2.) Hashing the (hash of the) ciphertext into the final shared key does not help at all with any formal security property or with proofs. On the contrary, hashing the hash of the ciphertext into the final key complicates QROM proofs as pointed out in https://eprint.iacr.org/2021/708.pdf. Removing this tweak simplifies proofs and speeds up encapsulation. Note that decapsulation will still need to compute a hash over the full ciphertext for implicit rejection and, to avoid timing side channels, needs to do so in every decapsulation, not just after a decryption failure. So, there won't be any performance gain in decapsulation. Should the Kyber standard drop hashing the hash of the ciphertext into the shared key?

The obvious disadvantage with both possible changes is that such changes at such a late stage require very careful evaluation. We may have missed some non-standard property that Kyber achieves with these tweaks, but does not without. Also, the modifications are not completely orthogonal to potential modifications of symmetric crypto (see the previous mail), because they require changes to the hashing inside the FO transform with possible consequences for domain separation.

Again, we're looking forward to hear what everybody thinks!

All the best,

The Kyber team

# Our Contributions

| | ANO | IND |
|---|---|---|
| Classic McEliece | Y | Y |
| NTRU | Y | Y |
| Kyber | ? | ? |
| Saber | ? | ? |

# Our Contributions

| | ANO | IND |
|---|---|---|
| Classic McEliece | Y | Y |
| NTRU | Y | Y |
| Kyber | ? | Y |
| Saber | ? | ? |

Provided concrete and tight proof of IND-CCA security for Kyber in the QROM:

# Our Contributions

| | ANO | IND |
|---|---|---|
| Classic McEliece | Y | Y |
| NTRU | Y | Y |
| Kyber | ? | Y |
| Saber | ? | ? |

Provided concrete and tight proof of IND-CCA security for Kyber in the QROM:

$$\mathrm{Adv}^{\mathrm{IND-CCA}}_{\mathrm{Kyber}} \leq \mathrm{Adv}^{\mathrm{IN}}_{\mathrm{FO}^{\not\perp}_m} + \mathrm{Adv}^{\mathrm{CR}}_{H}$$

Collision-resistance of nested hash "$H(c)$", when modelled as a QRO.

# Our Contributions

Established ANO-CCA security of Kyber and associated "KEM-DEM" hybrid PKE schemes in the QROM.

| | ANO | IND |
|---|---|---|
| Classic McEliece | Y | Y |
| NTRU | Y | Y |
| Kyber | Y | Y |
| Saber | ? | ? |

Provided concrete and tight proof of IND-CCA security for Kyber in the QROM:

$$\mathrm{Adv}^{\mathrm{IND-CCA}}_{\mathrm{Kyber}} \leq \mathrm{Adv}^{\mathrm{IND-CCA}}_{\mathrm{FO}^{\not\perp}_m} + \mathrm{Adv}^{\mathrm{CR}}_{H}$$

# Our Contributions

| | ANO | IND |
|---|---|---|
| Classic McEliece | Y | Y |
| NTRU | Y | Y |
| Kyber | Y | Y |
| Saber | Y | Y |

Established ANO-CCA security of Kyber and associated "KEM-DEM" hybrid PKE schemes in the QROM.

Provided concrete and tight proof of IND-CCA security for Kyber in the QROM:

$$\mathrm{Adv}^{\mathrm{IND-CCA}}_{\mathrm{Kyber}} \leq \mathrm{Adv}^{\mathrm{IND-CCA}}_{\mathrm{FO}^{\not\perp}_m} + \mathrm{Adv}^{\mathrm{CR}}_{H}$$

Our above IND-CCA and ANO-CCA security analyses also extends to Saber in the QROM.

# Technical Overview

| KGen′ | Encap(pk) | Decap(sk, $c$) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 2 : **return** (pk, sk) | 2 : $r \leftarrow G_r(m)$ | 2 : $r' \leftarrow G_r(m')$ |
| | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $\overline{k'} \leftarrow G_k(m')$ |
| | 5 : **return** $(c, \overline{k})$ | 5 : **if** $c' = c$ **then** |
| | | 6 : **return** $\overline{k'}$ |
| | | 7 : **else return** $\perp$ |

$\mathsf{FO}_m^{\perp}$ [Hofheinz-Hövelmanns-Kiltz@TCC'17]

$\mathsf{FO}_m^{\not\perp}$ outputs $G_k(s, c)$.

# Technical Overview

| KGen′ | Encap(pk) | Decap(sk, $c$) |
|---|---|---|
| 1 : $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KGen}$ | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 2 : **return** $(\mathsf{pk}, \mathsf{sk})$ | 2 : $r \leftarrow G_r(m)$ | 2 : $r' \leftarrow G_r(m')$ |
| | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $\overline{k'} \leftarrow G_k(m')$ |
| | 5 : **return** $(c, \overline{k})$ | 5 : **if** $c' = c$ **then** |
| | | 6 : **return** $\overline{k'}$ |
| | | 7 : **else return** $\perp$ |

$\mathsf{FO}_m^{\perp}$ [Hofheinz-Hövelmanns-Kiltz@TCC'17]

# Technical Overview

| KGen′ | Encap(pk) | Decap(sk, $c$) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 2 : **return** (pk, sk) | 2 : $r \leftarrow G_r(m)$ | 2 : $r' \leftarrow G_r(m')$ |
| | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $\overline{k'} \leftarrow G_k(m')$ |
| | 5 : **return** $(c, \overline{k})$ | 5 : **if** $c' = c$ **then** |
| | | 6 : **return** $\overline{k'}$ |
| | | 7 : **else return** $\perp$ |

$\mathsf{FO}_m^{\perp}$ [Hofheinz-Hövelmanns-Kiltz@TCC'17]

| KGen′ | Encap(pk) | Decap(sk′, $c$) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse $\mathsf{sk}' = (\mathsf{sk}, s)$ |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : $\mathsf{sk}' \leftarrow (\mathsf{sk}, s)$ | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : $k \leftarrow H'(\overline{k}, H(c))$ | 5 : $\overline{k'} \leftarrow G_k(m')$ |
| | 6 : **return** $(c, k)$ | 6 : **if** $c' = c$ **then** |
| | | 7 : **return** $H'(\overline{k'}, H(c))$ |
| | | 8 : **else return** $H'(s, H(c))$ |

Kyber (simplified)

Ignoring initial hashes "$H(m)$" and "$H(pk)$" in Encap.

# Technical Overview

| KGen′ | Encap(pk) | Decap(sk, $c$) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_\$ \mathcal{M}$ | 1 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 2 : **return** (pk, sk) | 2 : $r \leftarrow G_r(m)$ | 2 : $r' \leftarrow G_r(m')$ |
| | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $\overline{k'} \leftarrow G_k(m')$ |
| | 5 : **return** $(c, \overline{k})$ | 5 : **if** $c' = c$ **then** |
| | | 6 : **return** $\overline{k'}$ |
| | | 7 : **else return** $\perp$ |

$\mathsf{FO}_m^{\perp}$ [Hofheinz-Hövelmanns-Kiltz@TCC'17]

| KGen′ | Encap(pk) | Decap(sk′, $c$) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_\$ \mathcal{M}$ | 1 : Parse $\mathsf{sk}' = (\mathsf{sk}, s)$ |
| 2 : $s \leftarrow_\$ \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : $\mathsf{sk}' \leftarrow (\mathsf{sk}, s)$ | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : $k \leftarrow H'(\overline{k}, H(c))$ | 5 : $\overline{k'} \leftarrow G_k(m')$ |
| | 6 : **return** $(c, k)$ | 6 : **if** $c' = c$ **then** |
| | | 7 : **return** $H'(\overline{k'}, H(c))$ |
| | | 8 : **else return** $H'(s, H(c))$ |

Kyber (simplified)

# Technical Overview

| KGen′ | Encap(pk) | Decap(sk, $c$) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_\$ \mathcal{M}$ | 1 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 2 : **return** (pk, sk) | 2 : $r \leftarrow G_r(m)$ | 2 : $r' \leftarrow G_r(m')$ |
| | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $\overline{k'} \leftarrow G_k(m')$ |
| | 5 : **return** $(c, \overline{k})$ | 5 : **if** $c' = c$ **then** |
| | | 6 : **return** $\overline{k'}$ |
| | | 7 : **else return** $\perp$ |

$\mathsf{FO}_m^\perp$ [Hofheinz-Hövelmanns-Kiltz@TCC'17]

| KGen′ | Encap(pk) | Decap(sk′, $c$) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_\$ \mathcal{M}$ | 1 : Parse $\mathsf{sk}' = (\mathsf{sk}, s)$ |
| 2 : $s \leftarrow_\$ \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : $\mathsf{sk}' \leftarrow (\mathsf{sk}, s)$ | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : $k \leftarrow H'(\overline{k}, H(c))$ | 5 : $\overline{k'} \leftarrow G_k(m')$ |
| | 6 : **return** $(c, k)$ | 6 : **if** $c' = c$ **then** |
| | | 7 : **return** $H'(\overline{k'}, H(c))$ |
| | | 8 : **else return** $H'(s, H(c))$ |

Kyber (simplified)

# Technical Overview

| KGen′ | Encap(pk) | Decap(sk, $c$) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_\$ \mathcal{M}$ | 1 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 2 : **return** (pk, sk) | 2 : $r \leftarrow G_r(m)$ | 2 : $r' \leftarrow G_r(m')$ |
| | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $\overline{k'} \leftarrow G_k(m')$ |
| | 5 : **return** $(c, \overline{k})$ | 5 : **if** $c' = c$ **then** |
| | | 6 : **return** $\overline{k'}$ |
| | | 7 : **else return** $\perp$ |

$\mathsf{FO}_m^{\perp}$ [Hofheinz-Hövelmanns-Kiltz@TCC'17]

| KGen′ | Encap(pk) | Decap(sk′, $c$) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_\$ \mathcal{M}$ | 1 : Parse $\mathsf{sk}' = (\mathsf{sk}, s)$ |
| 2 : $s \leftarrow_\$ \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : $\mathsf{sk}' \leftarrow (\mathsf{sk}, s)$ | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : $k \leftarrow H'(\overline{k}, H(c))$ | 5 : $\overline{k'} \leftarrow G_k(m')$ |
| | 6 : **return** $(c, k)$ | 6 : **if** $c' = c$ **then** |
| | | 7 : **return** $H'(\overline{k'}, H(c))$ |
| | | 8 : **else return** $H'(s, H(c))$ |

Kyber (simplified)

$(pk, sk)$

$(pk, sk')$


$s \leftarrow_\$ M$

$sk' \leftarrow (sk, s)$

# Technical Overview

| KGen′ | Encap(pk) | Decap(sk, $c$) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 2 : **return** (pk, sk) | 2 : $r \leftarrow G_r(m)$ | 2 : $r' \leftarrow G_r(m')$ |
| | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $\overline{k'} \leftarrow G_k(m')$ |
| | 5 : **return** $(c, \overline{k})$ | 5 : **if** $c' = c$ **then** |
| | | 6 : **return** $\overline{k'}$ |
| | | 7 : **else return** $\perp$ |

$\mathsf{FO}_m^{\perp}$ [Hofheinz-Hövelmanns-Kiltz@TCC'17]

| KGen′ | Encap(pk) | Decap(sk′, $c$) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse $\mathsf{sk}' = (\mathsf{sk}, s)$ |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : $\mathsf{sk}' \leftarrow (\mathsf{sk}, s)$ | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : $k \leftarrow H'(\overline{k}, H(c))$ | 5 : $\overline{k'} \leftarrow G_k(m')$ |
| | 6 : **return** $(c, k)$ | 6 : **if** $c' = c$ **then** |
| | | 7 : **return** $H'(\overline{k'}, H(c))$ |
| | | 8 : **else return** $H'(s, H(c))$ |

Kyber (simplified)

$(c, \overline{k})$

$H, H'$

$k \leftarrow H'(\overline{k}, H(c))$

$(c, k)$

# Technical Overview

| KGen′ | Encap(pk) | Decap(sk, $c$) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 2 : **return** (pk, sk) | 2 : $r \leftarrow G_r(m)$ | 2 : $r' \leftarrow G_r(m')$ |
| | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $\overline{k'} \leftarrow G_k(m')$ |
| | 5 : **return** $(c, \overline{k})$ | 5 : **if** $c' = c$ **then** |
| | | 6 : **return** $\overline{k'}$ |
| | | 7 : **else return** $\perp$ |

$\mathsf{FO}_m^{\perp}$ [Hofheinz-Hövelmanns-Kiltz@TCC'17]

| KGen′ | Encap(pk) | Decap(sk′, $c$) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse $\mathsf{sk}' = (\mathsf{sk}, s)$ |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : $\mathsf{sk}' \leftarrow (\mathsf{sk}, s)$ | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : $k \leftarrow H'(\overline{k}, H(c))$ | 5 : $\overline{k'} \leftarrow G_k(m')$ |
| | 6 : **return** $(c, k)$ | 6 : **if** $c' = c$ **then** |
| | | 7 : **return** $H'(\overline{k'}, H(c))$ |
| | | 8 : **else return** $H'(s, H(c))$ |

Kyber (simplified)

$\overline{k'}$

$\perp$

$H'(\overline{k'}, H(c))$

$H'(s, H(c))$

# Technical Overview

| KGen′ | Encap(pk) | Decap(sk, $c$) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 2 : **return** (pk, sk) | 2 : $r \leftarrow G_r(m)$ | 2 : $r' \leftarrow G_r(m')$ |
| | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $\overline{k'} \leftarrow G_k(m')$ |
| | 5 : **return** $(c, \overline{k})$ | 5 : **if** $c' = c$ **then** |
| | | 6 : **return** $\overline{k'}$ |
| | | 7 : **else return** $\perp$ |

$\mathsf{FO}_m^{\perp}$ [Hofheinz-Hövelmanns-Kiltz@TCC'17]

| KGen′ | Encap(pk) | Decap(sk′, $c$) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse sk′ = (sk, $s$) |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, $s$) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : $k \leftarrow H'(\overline{k}, H(c))$ | 5 : $\overline{k'} \leftarrow G_k(m')$ |
| | 6 : **return** $(c, k)$ | 6 : **if** $c' = c$ **then** |
| | | 7 : **return** $H'(\overline{k'}, H(c))$ |
| | | 8 : **else return** $H'(s, H(c))$ |

Kyber (simplified)

$\overline{k'}$

$\perp$

$H'(\overline{k'}, H(c))$

$H'(s, H(c))$


IND-CCA security of $\mathsf{FO}_m^{\perp}$ KEMs in the QROM $\Rightarrow$ IND-CCA security of Kyber in the QROM

# Technical Overview

| KGen′ | Encap(pk) | Decap(sk, $c$) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 2 : **return** (pk, sk) | 2 : $r \leftarrow G_r(m)$ | 2 : $r' \leftarrow G_r(m')$ |
| | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $\overline{k'} \leftarrow G_k(m')$ |
| | 5 : **return** $(c, \overline{k})$ | 5 : **if** $c' = c$ **then** |
| | | 6 : **return** $\overline{k'}$ |
| | | 7 : **else return** $\perp$ |

$\mathsf{FO}_m^{\perp}$ [Hofheinz-Hövelmanns-Kiltz@TCC'17]

| KGen′ | Encap(pk) | Decap(sk′, $c$) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse $\mathsf{sk}' = (\mathsf{sk}, s)$ |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : $\mathsf{sk}' \leftarrow (\mathsf{sk}, s)$ | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : $k \leftarrow H'(\overline{k}, H(c))$ | 5 : $\overline{k'} \leftarrow G_k(m')$ |
| | 6 : **return** $(c, k)$ | 6 : **if** $c' = c$ **then** |
| | | 7 : **return** $H'(\overline{k'}, H(c))$ |
| | | 8 : **else return** $H'(s, H(c))$ |

Kyber (simplified)

$\overline{k'}$

$\perp$

$H'(\overline{k'}, H(c))$

$H'(s, H(c))$


IND-CCA security of $\mathsf{FO}_m^{\perp}$ KEMs in the QROM $\Rightarrow$ IND-CCA security of Kyber in the QROM

# Technical Overview

IND-CCA security of $\mathsf{FO}_m^{\perp}$ KEMs in the QROM $\Rightarrow$ IND-CCA security of Kyber in the QROM

# Technical Overview

IND-CCA security of $\mathsf{FO}_m^{\perp}$ KEMs in the QROM $\Rightarrow$ IND-CCA security of Kyber in the QROM

Non-tight proofs, compared to $\mathsf{FO}_m^{\not\perp}$.

**Tighter Proofs of CCA Security in the Quantum Random Oracle Model**

Nina

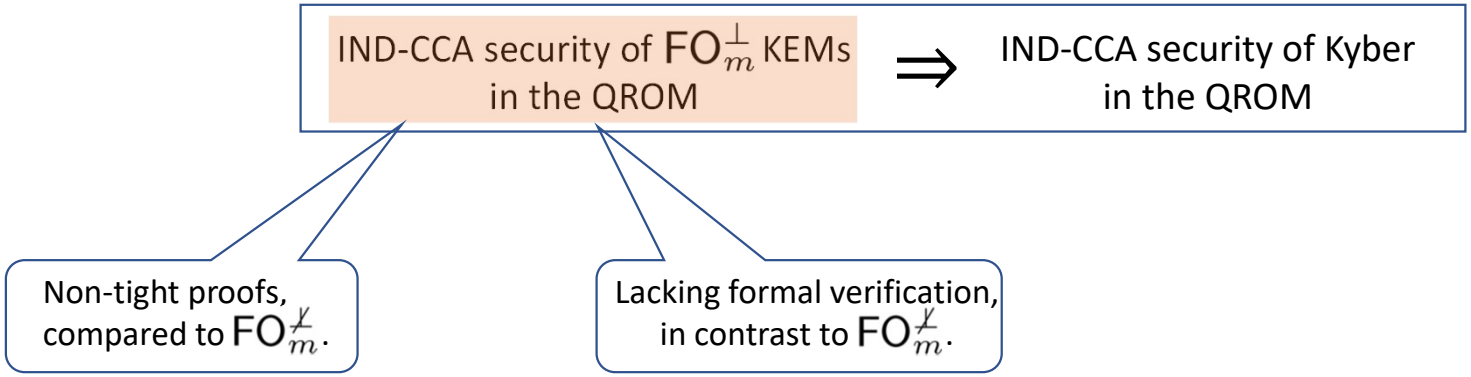
**Measure-Rewind-Measure: Tighter Quantum Random Oracle Model Proofs for One-Way to Hiding and CCA Security**

Veronika Kuchta[1], Amin Sakzad[1(✉)], Damien Stehlé[2,3], Ron Steinfeld[1(✉)] and Shi-Feng Sun[1,4]

[4] Ein

[1] Faculty of Information Technology, Monash University, Melbourne, Australia
{amin.sakzad,ron.steinfeld}@monash.edu
[2] Univ. Lyon, EnsL, UCBL, CNRS, Inria, LIP, 69342 Lyon Cedex 07, France
[3] Institut Universitaire de France, Paris, France
[4] Data61, CSIRO, Canberra, Australia [Eurocrypt'20]

# Technical Overview




**Tighter Proofs of CCA Security in the Quantum Random Oracle Model**

**Measure-Rewind-Measure: Tighter Quantum Random Oracle Model Proofs for One-Way to Hiding and CCA Security**

Nina

Veronika Kuchta[1], Amin Sakzad[1(✉)], Damien Stehlé[2,3], Ron Steinfeld[1(✉)] and Shi-Feng Sun[1,4]

[4] Ein

[1] Faculty of Information Technology, Monash University, Melbourne, Australia
{amin.sakzad,ron.steinfeld}@monash.edu
[2] Univ. Lyon, EnsL, UCBL, CNRS, Inria, LIP, 69342 Lyon Cedex 07, France
[3] Institut Universitaire de France, Paris, France
[4] Data61, CSIRO, Canberra, Australia

[Eurocrypt'20]

**Post-Quantum Verification of Fujisaki-Okamoto**

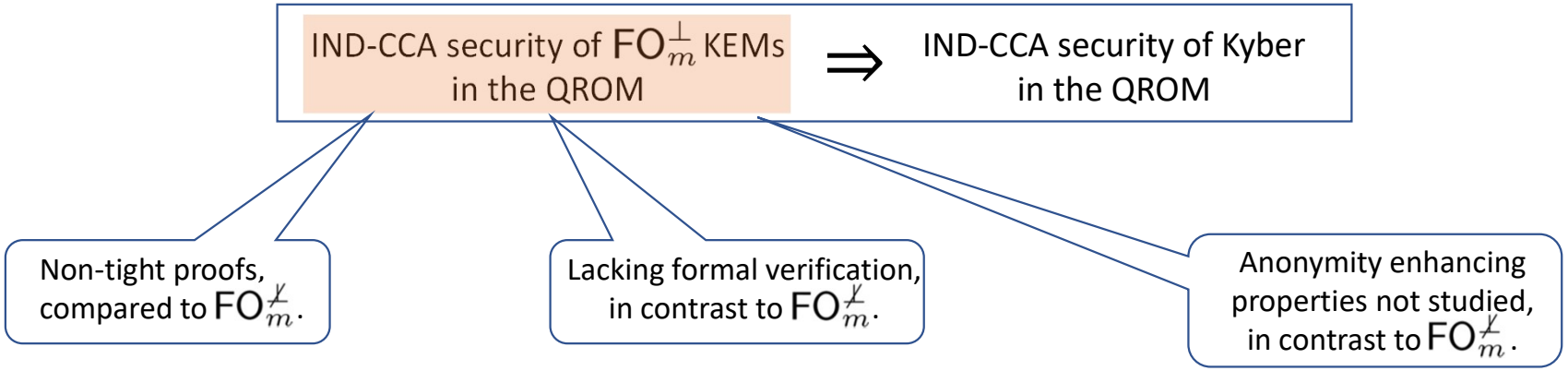
Dominique Unruh(✉)

University of Tartu, Tartu, Estonia
unruh@ut.ee

**Abstract.** We present a computer-verified formalization of the post-quantum security proof of the Fujisaki-Okamoto transform (as analyzed by Hövelmanns, Kiltz, Schäge, and Unruh, PKC 2020). The formalization is done in quantum relational Hoare logic and checked in the `qrhl-tool` (Unruh, POPL 2019).

[Asiacrypt'20]

# Technical Overview




**Tighter Proofs of CCA Security in the Quantum Random Oracle Model**

**Measure-Rewind-Measure: Tighter Quantum Random Oracle Model Proofs for One-Way to Hiding and CCA Security**

Nina

Veronika Kuchta[1], Amin Sakzad[1(✉)], Damien Stehlé[2,3], Ron Steinfeld[1(✉)] and Shi-Feng Sun[1,4]

[4] Ein

[1] Faculty of Information Technology, Monash University, Melbourne, Australia
{amin.sakzad,ron.steinfeld}@monash.edu
[2] Univ. Lyon, EnsL, UCBL, CNRS, Inria, LIP, 69342 Lyon Cedex 07, France
[3] Institut Universitaire de France, Paris, France
[4] Data61, CSIRO, Canberra, Australia [Eurocrypt'20]

**Post-Quantum Verification of Fujisaki-Okamoto**

Dominique Unruh(✉)

University of Tartu, Tartu, Estonia
unruh@ut.ee

**Abstract.** We present a computer-verified formalization of the post-quantum security proof of the Fujisaki-Okamoto transform (as analyzed by Hövelmanns, Kiltz, Schäge, and Unruh, PKC 2020). The formalization is done in quantum relational Hoare logic and checked in the `qrhl-tool` (Unruh, POPL 2019). [Asiacrypt'20]

**Anonymous, Robust Post-quantum Public Key Encryption**

**Anonymity of NIST PQC Round 3 KEMs**

Keita Xagawa(✉)

NTT Social Informatics Laboratories, Tokyo, Japan
keita.xagawa.zv@hco.ntt.co.jp [Eurocrypt'22]

# Technical Overview

IND-CCA security of $\mathsf{FO}_m^{\perp}$ KEMs in the QROM $\Rightarrow$ IND-CCA security of Kyber in the QROM

IND-CCA security of $\mathsf{FO}_m^{\not\perp}$ KEMs in the QROM $\Rightarrow$ ?

# Technical Overview

| KGen′ | Encap(pk) | Decap(sk, $c$) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 2 : **return** (pk, sk) | 2 : $r \leftarrow G_r(m)$ | 2 : $r' \leftarrow G_r(m')$ |
| | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : **if** $c' = c$ **then** |
| | 5 : **return** $(c, \overline{k})$ | 5 : **return** $G_k(m')$ |
| | | 6 : **else return** $\perp$ |

$\mathsf{FO}_m^{\perp}$ [Hofheinz-Hövelmanns-Kiltz@TCC'17]

| KGen′ | Encap(pk) | Decap(sk′, $c$) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse $\mathsf{sk}' = (\mathsf{sk}, s)$ |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : $\mathsf{sk}' \leftarrow (\mathsf{sk}, s)$ | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : $k \leftarrow H'(\overline{k}, H(c))$ | 5 : $\overline{k'} \leftarrow G_k(m')$ |
| | 6 : **return** $(c, k)$ | 6 : **if** $c' = c$ **then** |
| | | 7 : **return** $H'(\overline{k'}, H(c))$ |
| | | 8 : **else return** $H'(s, H(c))$ |

Kyber (simplified)


IND-CCA security of $\mathsf{FO}_m^{\not\perp}$ KEMs in the QROM $\Rightarrow$ ?

# Technical Overview

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_\$ \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_\$ \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : return (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : return $(c, \overline{k})$ | 5 : if $c' = c$ then |
| | | 6 : return $G_k(m')$ |
| | | 7 : else return $G_k(s, c)$ |

$\mathsf{FO}_m^{\not\perp}$ [Hofheinz-Hövelmanns-Kiltz@TCC'17]

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_\$ \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_\$ \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : return (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : $k \leftarrow H'(\overline{k}, H(c))$ | 5 : $\overline{k'} \leftarrow G_k(m')$ |
| | 6 : return $(c, k)$ | 6 : if $c' = c$ then |
| | | 7 : return $H'(\overline{k'}, H(c))$ |
| | | 8 : else return $H'(s, H(c))$ |

Kyber (simplified)


IND-CCA security of $\mathsf{FO}_m^{\not\perp}$ KEMs in the QROM $\Rightarrow$ ?

# Technical Overview

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_\$ \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_\$ \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : **return** $(c, \overline{k})$ | 5 : **if** $c' = c$ **then** |
| | | 6 : **return** $G_k(m')$ |
| | | 7 : **else return** $G_k(s, c)$ |

$\mathsf{FO}_m^{\not\perp}$ [Hofheinz-Hövelmanns-Kiltz@TCC'17]

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_\$ \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_\$ \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : $k \leftarrow H'(\overline{k}, H(c))$ | 5 : $\overline{k'} \leftarrow G_k(m')$ |
| | 6 : **return** $(c, k)$ | 6 : **if** $c' = c$ **then** |
| | | 7 : **return** $H'(\overline{k'}, H(c))$ |
| | | 8 : **else return** $H'(s, H(c))$ |

Kyber (simplified)

$\overline{k}'$

?


IND-CCA security of $\mathsf{FO}_m^{\not\perp}$ KEMs in the QROM $\Rightarrow$ ?

# Technical Overview

| KGen′ | Encap(pk) | Decap(sk′, $c$) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_\$ \mathcal{M}$ | 1 : Parse sk′ = (sk, $s$) |
| 2 : $s \leftarrow_\$ \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow$ Dec(sk, $c$) |
| 3 : sk′ ← (sk, $s$) | 3 : $c \leftarrow$ Enc(pk, $m$; $r$) | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow$ Enc(pk, $m'$; $r'$) |
| | 5 : **return** $(c, \overline{k})$ | 5 : **if** $c' = c$ **then** |
| | | 6 : **return** $G_k(m')$ |
| | | 7 : **else return** $G_k(s, c)$ |

$\mathsf{FO}_m^{\not\perp}$ [Hofheinz-Hövelmanns-Kiltz@TCC'17]

| KGen′ | Encap(pk) | Decap(sk′, $c$) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_\$ \mathcal{M}$ | 1 : Parse sk′ = (sk, $s$) |
| 2 : $s \leftarrow_\$ \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow$ Dec(sk, $c$) |
| 3 : sk′ ← (sk, $s$) | 3 : $c \leftarrow$ Enc(pk, $m$; $r$) | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow$ Enc(pk, $m'$; $r'$) |
| | 5 : $k \leftarrow H'(\overline{k}, H(c))$ | 5 : $\overline{k'} \leftarrow G_k(m')$ |
| | 6 : **return** $(c, k)$ | 6 : **if** $c' = c$ **then** |
| | | 7 : **return** $H'(\overline{k'}, H(c))$ |
| | | 8 : **else return** $H'(s, H(c))$ |

Kyber (simplified)

$\overline{k}' = G_k(m')$?
or
$\overline{k}' = G_k(s, c)$?

$\overline{k}'$

?


IND-CCA security of $\mathsf{FO}_m^{\not\perp}$ KEMs in the QROM $\Rightarrow$ ?

# Technical Overview

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_\$ \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_\$ \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : **return** $(c, \overline{k})$ | 5 : **if** $c' = c$ **then** |
| | | 6 : **return** $G_k(m')$ |
| | | 7 : **else return** $G_k(s, c)$ |

$\mathsf{FO}_m^{\not\perp}$ [Hofheinz-Hövelmanns-Kiltz@TCC'17]

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_\$ \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_\$ \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : $k \leftarrow H'(\overline{k}, H(c))$ | 5 : $\overline{k'} \leftarrow G_k(m')$ |
| | 6 : **return** $(c, k)$ | 6 : **if** $c' = c$ **then** |
| | | 7 : **return** $H'(\overline{k'}, H(c))$ |
| | | 8 : **else return** $H'(s, H(c))$ |

Kyber (simplified)


IND-CCA security of $\mathsf{FO}_m^{\not\perp}$ KEMs in the QROM $\Rightarrow$ ?

# Technical Overview

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : return (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : return $(c, \overline{k})$ | 5 : if $c' = c$ then |
| | | 6 : return $G_k(m')$ |
| | | 7 : else return $\overline{H}(H(c))$ |

$\mathsf{FO}_m^{\not\perp}$ [Maram-Xagawa @PKC'23]

modified

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : return (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : $k \leftarrow H'(\overline{k}, H(c))$ | 5 : $\overline{k'} \leftarrow G_k(m')$ |
| | 6 : return $(c, k)$ | 6 : if $c' = c$ then |
| | | 7 : return $H'(\overline{k'}, H(c))$ |
| | | 8 : else return $H'(\overline{H}(H(c)), H(c))$ |

Kyber (simplified)

modified


IND-CCA security of $\mathsf{FO}_m^{\not\perp}$ KEMs in the QROM $\Rightarrow$ ?

# Technical Overview

| KGen′ | Encap(pk) | Decap(sk′, $c$) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse sk′ = (sk, $s$) |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow$ Dec(sk, $c$) |
| 3 : sk′ ← (sk, $s$) | 3 : $c \leftarrow$ Enc(pk, $m$; $r$) | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow$ Enc(pk, $m'$; $r'$) |
| | 5 : **return** $(c, \overline{k})$ | 5 : **if** $c' = c$ **then** |
| | | 6 : **return** $G_k(m')$ |
| | | 7 : **else return** $\overline{H}(H(c))$ |

$\mathsf{FO}_m^{\not\perp}$ [Maram-Xagawa @PKC'23]

modified

| KGen′ | Encap(pk) | Decap(sk′, $c$) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse sk′ = (sk, $s$) |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow$ Dec(sk, $c$) |
| 3 : sk′ ← (sk, $s$) | 3 : $c \leftarrow$ Enc(pk, $m$; $r$) | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow$ Enc(pk, $m'$; $r'$) |
| | 5 : $k \leftarrow H'(\overline{k}, H(c))$ | 5 : $\overline{k'} \leftarrow G_k(m')$ |
| | 6 : **return** $(c, k)$ | 6 : **if** $c' = c$ **then** |
| | | 7 : **return** $H'(\overline{k'}, H(c))$ |
| | | 8 : **else return** $H'(\overline{H}(H(c)), H(c))$ |

Kyber (simplified)

modified

$\overline{k}'$

$H'(\overline{k}', H(c))$


IND-CCA security of $\mathsf{FO}_m^{\not\perp}$ KEMs in the QROM $\Rightarrow$ ?

# Technical Overview

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : **return** $(c, \overline{k})$ | 5 : **if** $c' = c$ **then** |
| | | 6 : **return** $G_k(m')$ |
| | | 7 : **else return** $\overline{H}(H(c))$ |

$\mathsf{FO}_m^{\not\perp}$ modified [Maram-Xagawa @PKC'23]

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : $k \leftarrow H'(\overline{k}, H(c))$ | 5 : $\overline{k'} \leftarrow G_k(m')$ |
| | 6 : **return** $(c, k)$ | 6 : **if** $c' = c$ **then** |
| | | 7 : **return** $H'(\overline{k'}, H(c))$ |
| | | 8 : **else return** $H'(\overline{H}(H(c)), H(c))$ |

Kyber (simplified) modified

$\overline{k}' = G_k(m')$

$\overline{k}'$

$H'(\overline{k}', H(c))$


IND-CCA security of $\mathsf{FO}_m^{\not\perp}$ KEMs in the QROM $\Rightarrow$ ?

# Technical Overview

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : **return** $(c, \overline{k})$ | 5 : **if** $c' = c$ **then** |
| | | 6 : **return** $G_k(m')$ |
| | | 7 : **else return** $\overline{H}(H(c))$ |

$\mathsf{FO}_m^{\not\perp}$ [Maram-Xagawa @PKC'23]

modified

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : $k \leftarrow H'(\overline{k}, H(c))$ | 5 : $\overline{k'} \leftarrow G_k(m')$ |
| | 6 : **return** $(c, k)$ | 6 : **if** $c' = c$ **then** |
| | | 7 : **return** $H'(\overline{k'}, H(c))$ |
| | | 8 : **else return** $H'(\overline{H}(H(c)), H(c))$ |

Kyber (simplified)

modified

$\overline{k}' = \overline{H}(H(c))$

$\overline{k}'$

$H'(\overline{k}', H(c))$


IND-CCA security of $\mathsf{FO}_m^{\not\perp}$ KEMs in the QROM $\Rightarrow$ ?

# Technical Overview

| KGen′ | Encap(pk) | Decap(sk′, $c$) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse sk′ = (sk, $s$) |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow$ Dec(sk, $c$) |
| 3 : sk′ ← (sk, $s$) | 3 : $c \leftarrow$ Enc(pk, $m$; $r$) | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow$ Enc(pk, $m'$; $r'$) |
| | 5 : **return** $(c, \overline{k})$ | 5 : **if** $c' = c$ **then** |
| | | 6 : **return** $G_k(m')$ |
| | | 7 : **else return** $\overline{H}(H(c))$ |

$\mathsf{FO}_m^{\not\perp}$ [Maram-Xagawa @PKC'23]

modified

| KGen′ | Encap(pk) | Decap(sk′, $c$) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse sk′ = (sk, $s$) |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow$ Dec(sk, $c$) |
| 3 : sk′ ← (sk, $s$) | 3 : $c \leftarrow$ Enc(pk, $m$; $r$) | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow$ Enc(pk, $m'$; $r'$) |
| | 5 : $k \leftarrow H'(\overline{k}, H(c))$ | 5 : $\overline{k'} \leftarrow G_k(m')$ |
| | 6 : **return** $(c, k)$ | 6 : **if** $c' = c$ **then** |
| | | 7 : **return** $H'(\overline{k'}, H(c))$ |
| | | 8 : **else return** $H'(\overline{H}(H(c)), H(c))$ |

Kyber (simplified)

modified

$\overline{k}'$

$H'(\overline{k}', H(c))$


IND-CCA security of $\mathsf{FO}_m^{\not\perp}$ KEMs in the QROM $\Rightarrow$ ?

# Technical Overview

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : **return** $(c, \overline{k})$ | 5 : **if** $c' = c$ **then** |
| | | 6 : **return** $G_k(m')$ |
| | | 7 : **else return** $\overline{H}(H(c))$ |

$\mathsf{FO}_m^{\not\perp}$ [Maram-Xagawa @PKC'23]

modified

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : $k \leftarrow H'(\overline{k}, H(c))$ | 5 : $\overline{k'} \leftarrow G_k(m')$ |
| | 6 : **return** $(c, k)$ | 6 : **if** $c' = c$ **then** |
| | | 7 : **return** $H'(\overline{k'}, H(c))$ |
| | | 8 : **else return** $H'(\overline{H}(H(c)), H(c))$ |

Kyber (simplified)

modified

$\approx$

$\overline{k}'$

$H'(\overline{k}', H(c))$


$\approx$


IND-CCA security of both KEMs computationally equivalent.

IND-CCA security of $\mathsf{FO}_m^{\not\perp}$ KEMs in the QROM $\Rightarrow$ ?

IND-CCA security of both KEMs computationally equivalent.

# Technical Overview

| KGen′ | Encap(pk) | Decap(sk′, $c$) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse sk′ = (sk, $s$) |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow$ Dec(sk, $c$) |
| 3 : sk′ ← (sk, $s$) | 3 : $c \leftarrow$ Enc(pk, $m$; $r$) | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow$ Enc(pk, $m'$; $r'$) |
| | 5 : **return** $(c, \overline{k})$ | 5 : **if** $c' = c$ **then** |
| | | 6 : **return** $G_k(m')$ |
| | | 7 : **else return** $\overline{H}(H(c))$ |

$\mathsf{FO}_m^{\not\perp}$ modified [Maram-Xagawa @PKC'23]

| KGen′ | Encap(pk) | Decap(sk′, $c$) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse sk′ = (sk, $s$) |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow$ Dec(sk, $c$) |
| 3 : sk′ ← (sk, $s$) | 3 : $c \leftarrow$ Enc(pk, $m$; $r$) | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow$ Enc(pk, $m'$; $r'$) |
| | 5 : $k \leftarrow H'(\overline{k}, H(c))$ | 5 : $\overline{k'} \leftarrow G_k(m')$ |
| | 6 : **return** $(c, k)$ | 6 : **if** $c' = c$ **then** |
| | | 7 : **return** $H'(\overline{k'}, H(c))$ |
| | | 8 : **else return** $H'(\overline{H}(H(c)), H(c))$ |

Kyber (simplified) modified

≈

$\overline{k}'$

$H'(\overline{k}', H(c))$

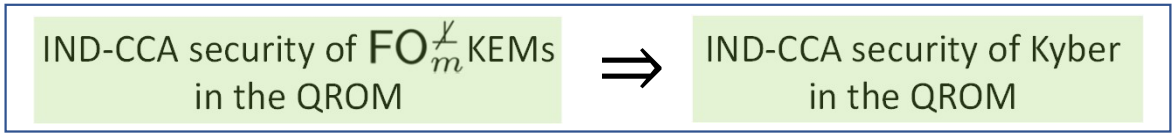
≈

IND-CCA security of both KEMs computationally equivalent.

IND-CCA security of $\mathsf{FO}_m^{\not\perp}$ KEMs in the QROM ⇒ IND-CCA security of Kyber in the QROM

IND-CCA security of both KEMs computationally equivalent.

# Technical Overview



# Technical Overview

IND-CCA security of $\mathsf{FO}_m^{\not\perp}$ KEMs in the QROM $\Rightarrow$ IND-CCA security of Kyber in the QROM

As shown in [Xagawa@Eurocrypt'22].

ANO-CCA security of $\mathsf{FO}_m^{\not\perp}$ KEMs in the QROM $\Rightarrow$ ANO-CCA security of Kyber in the QROM

# Technical Overview

IND-CCA security of $\mathsf{FO}_m^{\not\perp}$ KEMs in the QROM $\Rightarrow$ IND-CCA security of Kyber in the QROM

ANO-CCA security of $\mathsf{FO}_m^{\not\perp}$ KEMs in the QROM

ANO-CCA security of Kyber in the QROM

# Technical Overview

IND-CCA security of $\mathsf{FO}_m^{\not\perp}$ KEMs in the QROM $\Rightarrow$ IND-CCA security of Kyber in the QROM

SPR-CCA security of $\mathsf{FO}_m^{\not\perp}$ KEMs in the QROM

As shown in [Xagawa@Eurocrypt'22].

ANO-CCA security of $\mathsf{FO}_m^{\not\perp}$ KEMs in the QROM

ANO-CCA security of Kyber in the QROM

# Technical Overview

$(c, k) \leftarrow Encap(pk_{Bob})$, then $k$ indistinguishable from a random key.

IND-CCA security of $\mathsf{FO}_m^{\not\perp}$ KEMs in the QROM $\Rightarrow$ IND-CCA security of Kyber in the QROM

$(c, k) \leftarrow Encap(pk_{Bob})$, then $k$ and $c$ indistinguishable from a random key and random ciphertext, i.e., strong pseudorandomness.

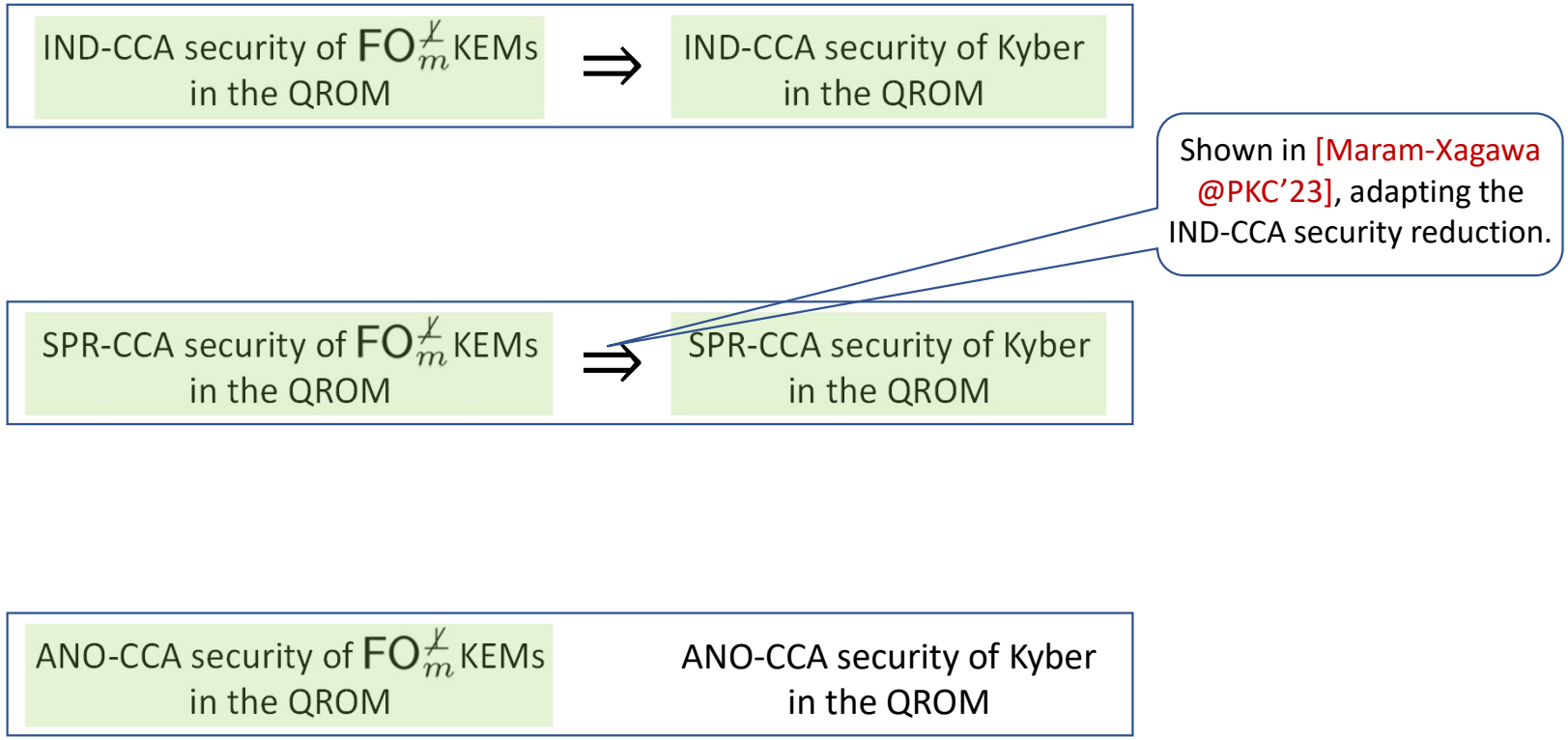
SPR-CCA security of $\mathsf{FO}_m^{\not\perp}$ KEMs in the QROM

As shown in [Xagawa@Eurocrypt'22].

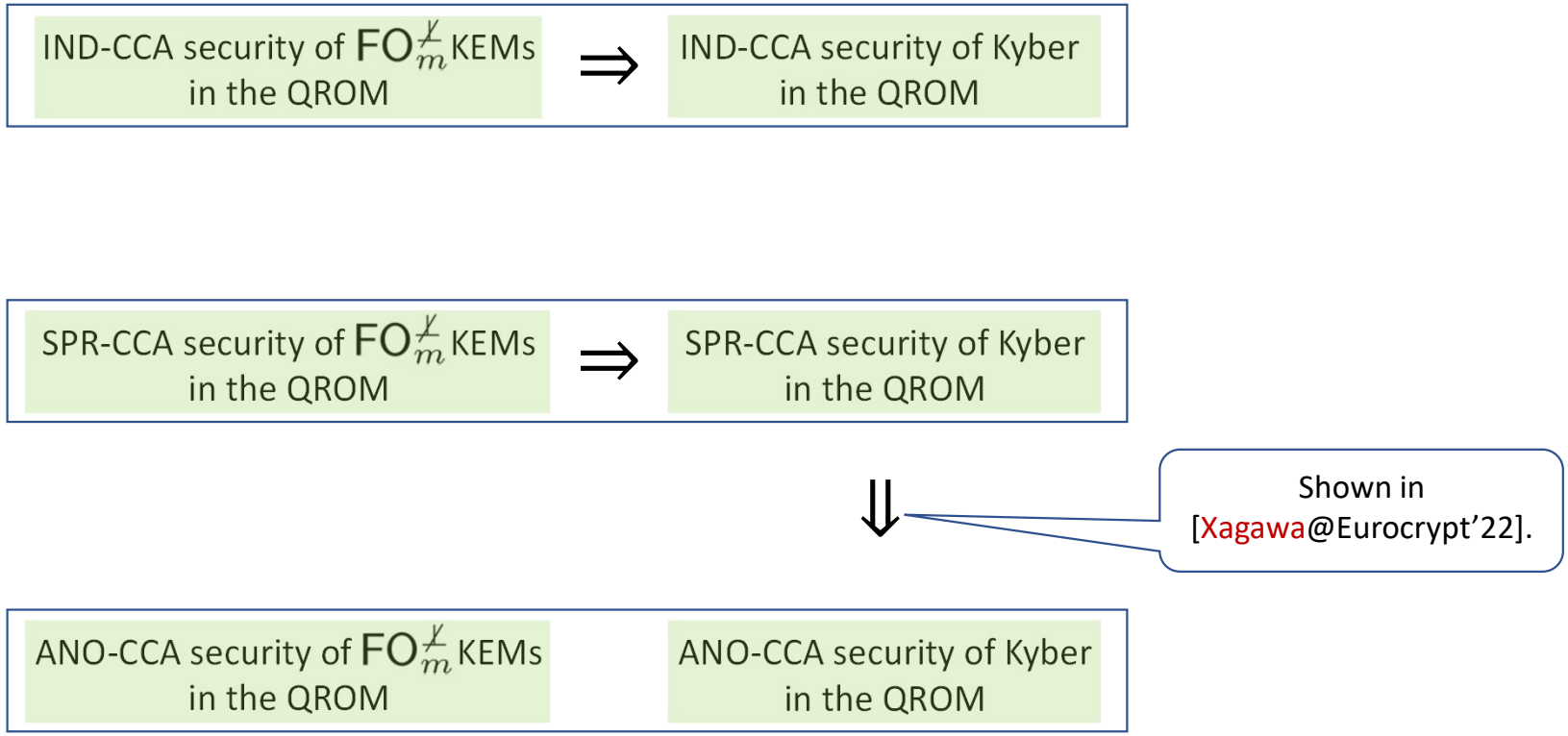
ANO-CCA security of $\mathsf{FO}_m^{\not\perp}$ KEMs in the QROM

ANO-CCA security of Kyber in the QROM

# Technical Overview



# Technical Overview



# Discussion

$$\mathrm{Adv}^{\mathrm{IND-CCA}}_{\mathrm{Kyber}} \leq \mathrm{Adv}^{\mathrm{IND-C}}_{\mathsf{FO}^{\not\perp}_m} + \mathrm{Adv}^{\mathrm{CR}}_{H}$$

# Discussion

$$\mathrm{Adv}^{\mathrm{IND-CCA}}_{\mathrm{Kyber}} \leq \mathrm{Adv}^{\mathrm{IND-CCA}}_{\mathsf{FO}^{\not\perp}_m} + \mathrm{Adv}^{\mathrm{CR}}_{H}$$


# Discussion

$$\mathrm{Adv}^{\mathrm{IND-CCA}}_{\mathrm{Kyber}} \leq \mathrm{Adv}^{\mathrm{IND-CCA}}_{\mathsf{FO}^{\not\perp}_m} + \mathrm{Adv}^{\mathrm{CR}}_{H}$$


Collision-resistance of nested hash "$H(c)$", when modelled as a QRO.

# Discussion

$$\mathrm{Adv}^{\mathrm{IND-CCA}}_{\mathrm{Kyber}} \leq \mathrm{Adv}^{\mathrm{IND}}_{\mathrm{FO}^{\not\perp}_m} + \mathrm{Adv}^{\mathrm{CR}}_{H}$$


Collision-resistance of nested hash "$H(c)$", when modelled as a QRO.

$\mathrm{FO}^{\not\perp}_m$ modified

$\approx$

IND-CCA security of both KEMs computationally equivalent.

# Discussion

$$\mathrm{Adv}^{\mathrm{IND-C}}_{\mathrm{Kyber}} \leq \mathrm{Adv}^{\mathrm{IND-CCA}}_{\mathrm{FO}^{\not\perp}_m} + \mathrm{Adv}^{\mathrm{CR}}_{H} + \cdots$$

Collision-resistance of nested hash "$H(c)$", when modelled as a QRO.

$\mathrm{FO}^{\not\perp}_m$ modified

≈

IND-CCA security of both KEMs computationally equivalent.

≈

Kyber modified

IND-CCA security of both KEMs computationally equivalent.

# Discussion

[Maram-Xagawa @PKC'23]

$$\mathrm{Adv}^{\mathrm{IND-CCA}}_{\mathrm{Kyber}} \leq \mathrm{Adv}^{\mathrm{IND-C}}_{\mathsf{FO}^{\not\perp}_m} + k_1 \frac{q^3}{2^{256}} + k_2 \frac{q}{2^{128}}$$

Bound on number of QRO queries by the IND-CCA adversary.

Collision-resistance of QROs, as proven in [Zhandry@QIC'15]

# Discussion

[Maram-Xagawa @PKC'23]

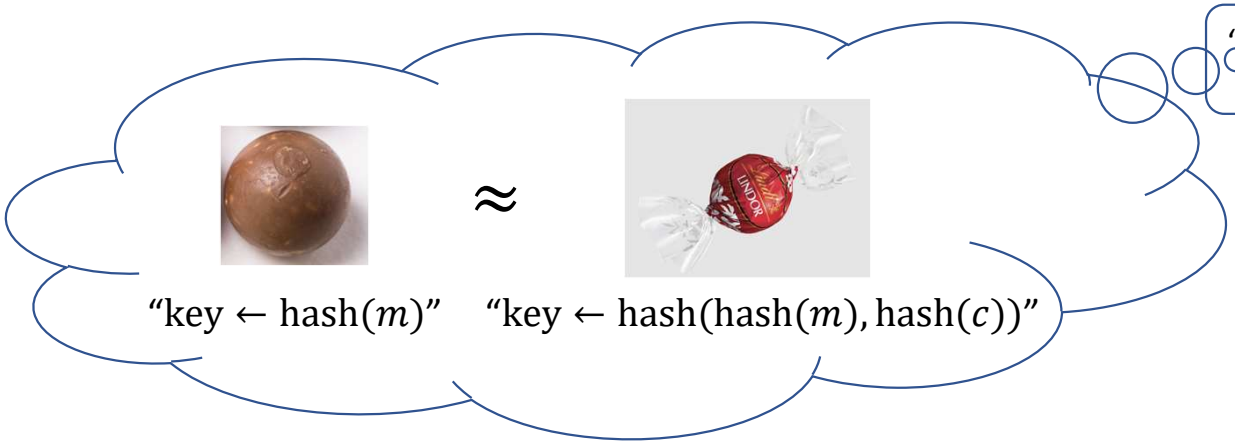
$$\mathrm{Adv}^{\mathrm{IND-C}}_{\mathrm{Kyber}} \leq \mathrm{Adv}^{\mathrm{IND-C}}_{\mathrm{FO}^{\not\perp}_m} + k_1 \frac{q^3}{2^{256}} + k_2 \frac{q}{2^{128}}$$

Bound on number of QRO queries by the IND-CCA adversary.

[Zhandry @Crypto'19]

$$\mathrm{Adv}^{\mathrm{IND}}_{\mathrm{Kyber}} \leq \mathrm{Adv}^{\mathrm{IND-CCA}}_{\mathrm{FO}^{\not\perp}_m} + k_1 \frac{q^2}{2^{128}}$$

"Indifferentiability loss".



# Discussion

[Maram-Xagawa @PKC'23]

$$\mathrm{Adv}^{\mathrm{IND-C}}_{\mathrm{Kyber}} \leq \mathrm{Adv}^{\mathrm{IND-CCA}}_{\mathrm{FO}^{\not\perp}_m} + k_1 \frac{q^3}{2^{256}} + k_2 \frac{q}{2^{128}}$$

Bound on number of QRO queries by the IND-CCA adversary.

[Zhandry @Crypto'19]

$$\mathrm{Adv}^{\mathrm{IND-C}}_{\mathrm{Kyber}} \leq \mathrm{Adv}^{\mathrm{IND-C}}_{\mathrm{FO}^{\not\perp}_m} + k_1 \frac{q^2}{2^{128}}$$

"Indifferentiability loss".

[Chen-Lu-Jia-Li @Inscrypt'22]

$$\mathrm{Adv}^{\mathrm{IND-C}}_{\mathrm{Kyber}} \leq \mathrm{Adv}^{\mathrm{IND-CCA}}_{\mathrm{FO}^{\not\perp}_m} + k_1 \sqrt{\frac{q^3}{2^{128}}} + k_2 \frac{q^3}{2^{256}}$$

Loss incurred w.r.t. simulating random invertible permutations in the QROM.

# Discussion

[Maram-Xagawa @PKC'23]

$$\mathrm{Adv}^{\mathrm{IND-CCA}}_{\mathrm{Kyber}} \leq \mathrm{Adv}^{\mathrm{IND-CCA}}_{\mathrm{FO}^{\not\perp}_m} + k_1 \frac{q^3}{2^{256}} + k_2 \frac{q}{2^{128}}$$

Bound on number of QRO queries by the IND-CCA adversary.

[Zhandry @Crypto'19]

$$\mathrm{Adv}^{\mathrm{IND-}}_{\mathrm{Kyber}} \leq \mathrm{Adv}^{\mathrm{IND-CCA}}_{\mathrm{FO}^{\not\perp}_m} + k_1 \frac{q^2}{2^{128}}$$

"Indifferentiability loss".

Collision-resistance of QROs.

[Chen-Lu-Jia-Li @Inscrypt'22]

$$\mathrm{Adv}^{\mathrm{IND-C}}_{\mathrm{Kyber}} \leq \mathrm{Adv}^{\mathrm{IND-CCA}}_{\mathrm{FO}^{\not\perp}_m} + k_1 \sqrt{\frac{q^3}{2^{128}}} + k_2 \frac{q^3}{2^{256}}$$

Loss incurred w.r.t. simulating random invertible permutations in the QROM.

# Discussion

[Maram-Xagawa @PKC'23]

$$\mathrm{Adv}^{\mathrm{IND-CCA}}_{\mathrm{Kyber}} \leq \mathrm{Adv}^{\mathrm{IND-CCA}}_{\mathrm{FO}^{\not\perp}_m} + k_1 \frac{q^3}{2^{256}} + k_2 \frac{q}{2^{128}}$$

Bound on number of QRO queries by the IND-CCA adversary.

$q \leq 2^{86}$

[Zhandry @Crypto'19]

$$\mathrm{Adv}^{\mathrm{IND-C}}_{\mathrm{Kyber}} \leq \mathrm{Adv}^{\mathrm{IN}}_{\mathrm{FO}^{\not\perp}_m} + k_1 \frac{q^2}{2^{128}}$$

"Indifferentiability loss".

$q \leq 2^{64}$

[Chen-Lu-Jia-Li @Inscrypt'22]

$$\mathrm{Adv}^{\mathrm{IND-CCA}}_{\mathrm{Kyber}} \leq \mathrm{Adv}^{\mathrm{IND-C}}_{\mathrm{FO}^{\not\perp}_m} + k_1 \sqrt{\frac{q^3}{2^{128}}} + k_2 \frac{q^3}{2^{256}}$$

Loss incurred w.r.t. simulating random invertible permutations in the QROM.

$q \leq 2^{43}$

# Discussion

[Maram-Xagawa @PKC'23]

$$\mathrm{Adv}^{\mathrm{IND-CCA}}_{\mathrm{Kyber}} \leq \mathrm{Adv}^{\mathrm{IND-CCA}}_{\mathrm{FO}^{\not\perp}_m} + k_1 \frac{q^3}{2^{256}} + k_2 \frac{q}{2^{128}}$$

Bound on number of QRO queries by the IND-CCA adversary.

$q \leq 2^{86}$

Use the "compressed oracle technique" of [Zhandry @Crypto'19].

[Zhandry @Crypto'19]

$$\mathrm{Adv}^{\mathrm{IND-CCA}}_{\mathrm{Kyber}} \leq \mathrm{Adv}^{\mathrm{IND-CCA}}_{\mathrm{FO}^{\not\perp}_m} + k_1 \frac{q^2}{2^{128}}$$

"Indifferentiability loss".

$q \leq 2^{64}$

[Chen-Lu-Jia-Li @Inscrypt'22]

$$\mathrm{Adv}^{\mathrm{IND-CCA}}_{\mathrm{Kyber}} \leq \mathrm{Adv}^{\mathrm{IND-CCA}}_{\mathrm{FO}^{\not\perp}_m} + k_1 \sqrt{\frac{q^3}{2^{128}}} + k_2 \frac{q^3}{2^{256}}$$

Loss incurred w.r.t. simulating random invertible permutations in the QROM.

$q \leq 2^{43}$

# Discussion

Uses the "One-Way To Hiding (OW2H) lemma" of [Ambainis-Hamburg-Unruh@Crypto'19].

[Maram-Xagawa @PKC'23]

$$\mathrm{Adv}^{\mathrm{IND-CCA}}_{\mathrm{Kyber}} \leq \mathrm{Adv}^{\mathrm{IND-C}}_{\mathrm{FO}^{\not\perp}_m} + k_1 \frac{q^3}{2^{256}} + k_2 \frac{q}{2^{128}}$$

Bound on number of QRO queries by the IND-CCA adversary.

$q \leq 2^{86}$

Use the "compressed oracle technique" of [Zhandry @Crypto'19].

[Zhandry @Crypto'19]

$$\mathrm{Adv}^{\mathrm{IND-C}}_{\mathrm{Kyber}} \leq \mathrm{Adv}^{\mathrm{IND-C}}_{\mathrm{FO}^{\not\perp}_m} + k_1 \frac{q^2}{2^{128}}$$

"Indifferentiability loss".

$q \leq 2^{64}$

[Chen-Lu-Jia-Li @Inscrypt'22]

$$\mathrm{Adv}^{\mathrm{IND-C}}_{\mathrm{Kyber}} \leq \mathrm{Adv}^{\mathrm{IND-CCA}}_{\mathrm{FO}^{\not\perp}_m} + k_1 \sqrt{\frac{q^3}{2^{128}}} + k_2 \frac{q^3}{2^{256}}$$

Loss incurred w.r.t. simulating random invertible permutations in the QROM.

$q \leq 2^{43}$

# Discussion

Amenable to formal verification! [Unruh@Asiacrypt'20]

Uses the "One-Way To Hiding (OW2H) lemma" of [Ambainis-Hamburg-Unruh@Crypto'19].

[Maram-Xagawa @PKC'23]

$$\mathrm{Adv}^{\mathrm{IND-CCA}}_{\mathrm{Kyber}} \leq \mathrm{Adv}^{\mathrm{IND-CCA}}_{\mathrm{FO}^{\not\perp}_m} + k_1 \frac{q^3}{2^{256}} + k_2 \frac{q}{2^{128}}$$

Bound on number of QRO queries by the IND-CCA adversary.

$q \leq 2^{86}$

Use the "compressed oracle technique" of [Zhandry @Crypto'19].

[Zhandry @Crypto'19]

$$\mathrm{Adv}^{\mathrm{IND-CCA}}_{\mathrm{Kyber}} \leq \mathrm{Adv}^{\mathrm{IND-CCA}}_{\mathrm{FO}^{\not\perp}_m} + k_1 \frac{q^2}{2^{128}}$$

"Indifferentiability loss".

$q \leq 2^{64}$

[Chen-Lu-Jia-Li @Inscrypt'22]

$$\mathrm{Adv}^{\mathrm{IND-CCA}}_{\mathrm{Kyber}} \leq \mathrm{Adv}^{\mathrm{IND-CCA}}_{\mathrm{FO}^{\not\perp}_m} + k_1 \sqrt{\frac{q^3}{2^{128}}} + k_2 \frac{q^3}{2^{256}}$$

Loss incurred w.r.t. simulating random invertible permutations in the QROM.

$q \leq 2^{43}$

# Discussion

Amenable to formal verification! [Unruh@Asiacrypt'20]

Uses the "One-Way To Hiding (OW2H) lemma" of [Ambainis-Hamburg-Unruh@Crypto'19].

[Maram-Xagawa @PKC'23]

$$\mathrm{Adv}^{\mathrm{IND-C}}_{\mathrm{Kyber}} \leq \mathrm{Adv}^{\mathrm{IND-C}}_{\mathrm{FO}^{\not\perp}_m} + k_1 \frac{q^3}{2^{256}} + k_2 \frac{q}{2^{128}}$$

- Formal verification?

# Discussion

Amenable to formal verification! [Unruh@Asiacrypt'20]

Uses the "One-Way To Hiding (OW2H) lemma" of [Ambainis-Hamburg-Unruh@Crypto'19].

[Maram-Xagawa @PKC'23]

$$\mathrm{Adv}^{\mathrm{IND-CCA}}_{\mathrm{Kyber}} \leq \mathrm{Adv}^{\mathrm{IND-CCA}}_{\mathsf{FO}^{\not\perp}_m} + k_1 \frac{q^3}{2^{256}} + k_2 \frac{q}{2^{128}}$$

- Formal verification?
- Optimality of our above bounds?

# Discussion

Amenable to formal verification! [Unruh@Asiacrypt'20]

Uses the "One-Way To Hiding (OW2H) lemma" of [Ambainis-Hamburg-Unruh@Crypto'19].

[Maram-Xagawa @PKC'23]

$$\mathrm{Adv}^{\mathrm{IND-C}}_{\mathrm{Kyber}} \leq \mathrm{Adv}^{\mathrm{IND-CCA}}_{\mathrm{FO}^{\not\perp}_m} + k_1 \frac{q^3}{2^{256}} + k_2 \frac{q}{2^{128}}$$

Collision-resistance of QROs.

- Formal verification?
- Optimality of our above bounds?
  - Tighter proof of IND-CCA security for Kyber <u>without</u> relying on collision-resistance of QROs?

# Discussion

Amenable to formal verification! [Unruh@Asiacrypt'20]

Uses the "One-Way To Hiding (OW2H) lemma" of [Ambainis-Hamburg-Unruh@Crypto'19].

[Maram-Xagawa @PKC'23]

$$\mathrm{Adv}^{\mathrm{IND-CCA}}_{\mathrm{Kyber}} \leq \mathrm{Adv}^{\mathrm{IND-CCA}}_{\mathrm{FO}^{\not\perp}_m} + k_1 \frac{q^3}{2^{256}} + k_2 \frac{q}{2^{128}}$$

Collision-resistance of QROs.

- Formal verification?
- Optimality of our above bounds?
  - Tighter proof of IND-CCA security for Kyber <u>without</u> relying on collision-resistance of QROs?
  - <u>Matching attack</u> on IND-CCA security of Kyber in the QROM by finding collisions in the nested hash "$H(c)$"?

# Extra Slides

# Technical Overview

$\mathsf{FO}^{\not\perp}_m$

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : **return** $(c, \overline{k})$ | 5 : **if** $c' = c$ **then** |
| | | 6 : **return** $G_k(m')$ |
| | | 7 : **else return** $G_k(s, c)$ |

IND-CCA adversary

# Technical Overview

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_\$ \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_\$ \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : **return** $(c, \overline{k})$ | 5 : **if** $c' = c$ **then** |
| $\mathsf{FO}_m^{\not\perp}$ | | 6 : **return** $G_k(m')$ |
| | | 7 : **else return** $G_k(s, c)$ |

IND-CCA adversary

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_\$ \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_\$ \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : **return** $(c, \overline{k})$ | 5 : **if** $c' = c$ **then** |
| $\mathsf{FO}_m^{\not\perp}$ modified | | 6 : **return** $G_k(m')$ |
| | | 7 : **else return** $\overline{H}(H(c))$ |

# Technical Overview

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_\$ \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_\$ \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : **return** $(c, \overline{k})$ | 5 : **if** $c' = c$ **then** |
| | | 6 : **return** $G_k(m')$ |
| | | 7 : **else return** $G_k(s, c)$ |

$\mathsf{FO}_m^{\not\perp}$

IND-CCA adversary

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_\$ \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_\$ \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : **return** $(c, \overline{k})$ | 5 : **if** $c' = c$ **then** |
| | | 6 : **return** $G_k(m')$ |
| | | 7 : **else return** $H''(c)$ |

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_\$ \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_\$ \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : **return** $(c, \overline{k})$ | 5 : **if** $c' = c$ **then** |
| | | 6 : **return** $G_k(m')$ |
| | | 7 : **else return** $\overline{H}(H(c))$ |

$\mathsf{FO}_m^{\not\perp}$ modified

# Technical Overview

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_\$ \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_\$ \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : **return** $(c, \overline{k})$ | 5 : **if** $c' = c$ **then** |
| | | 6 : **return** $G_k(m')$ |
| | | 7 : **else return** $G_k(s, c)$ |

$\mathsf{FO}_m^{\not\perp}$

… pseudorandomness of QRO "$G_k(s,\cdot)$".

≈

IND-CCA adversary

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_\$ \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_\$ \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : **return** $(c, \overline{k})$ | 5 : **if** $c' = c$ **then** |
| | | 6 : **return** $G_k(m')$ |
| | | 7 : **else return** $H''(c)$ |

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_\$ \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_\$ \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : **return** $(c, \overline{k})$ | 5 : **if** $c' = c$ **then** |
| | | 6 : **return** $G_k(m')$ |
| | | 7 : **else return** $\overline{H}(H(c))$ |

$\mathsf{FO}_m^{\not\perp}$ modified

# Technical Overview

$\mathsf{FO}_m^{\not\perp}$

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_\$ \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_\$ \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : return (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : return $(c, \overline{k})$ | 5 : if $c' = c$ then |
| | | 6 : return $G_k(m')$ |
| | | 7 : else return $G_k(s, c)$ |

IND-CCA adversary

$\mathsf{FO}_m^{\not\perp}$ modified

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_\$ \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_\$ \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : return (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : return $(c, \overline{k})$ | 5 : if $c' = c$ then |
| | | 6 : return $G_k(m')$ |
| | | 7 : else return $\overline{H}(H(c))$ |

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_\$ \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_\$ \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : return (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : return $(c, \overline{k})$ | 5 : if $c' = c$ then |
| | | 6 : return $G_k(m')$ |
| | | 7 : else return $H''(c)$ |

# Technical Overview

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_\$ \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_\$ \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : return (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : return $(c, \overline{k})$ | 5 : if $c' = c$ then |
| | | 6 : return $G_k(m')$ |
| | | 7 : else return $G_k(s, c)$ |

$\mathsf{FO}_m^{\not\perp}$

IND-CCA adversary

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_\$ \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_\$ \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : return (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : return $(c, \overline{k})$ | 5 : if $c' = c$ then |
| | | 6 : return $G_k(m')$ |
| | | 7 : else return $H''(c)$ |

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_\$ \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_\$ \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : return (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : return $(c, \overline{k})$ | 5 : if $c' = c$ then |
| | | 6 : return $G_k(m')$ |
| | | 7 : else return $\overline{H}(H(c))$ |

$\mathsf{FO}_m^{\not\perp}$ modified

… collision-resistance of QRO “$H(\cdot)$”.

# Technical Overview

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_\$ \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_\$ \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : return (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : return $(c, \overline{k})$ | 5 : if $c' = c$ then |
| | | 6 : return $G_k(m')$ |
| | | 7 : else return $G_k(s, c)$ |

$\mathsf{FO}_m^{\not\perp}$

… pseudorandomness of QRO "$G_k(s,\cdot)$".

≈

≈

IND-CCA adversary

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_\$ \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_\$ \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : return (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : return $(c, \overline{k})$ | 5 : if $c' = c$ then |
| | | 6 : return $G_k(m')$ |
| | | 7 : else return $H''(c)$ |

≈

… collision-resistance of QRO "$H(\cdot)$".

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_\$ \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_\$ \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : return (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : return $(c, \overline{k})$ | 5 : if $c' = c$ then |
| | | 6 : return $G_k(m')$ |
| | | 7 : else return $\overline{H}(H(c))$ |

$\mathsf{FO}_m^{\not\perp}$ modified

# Technical Overview

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : $k \leftarrow H'(\overline{k}, H(c))$ | 5 : $\overline{k'} \leftarrow G_k(m')$ |
| | 6 : **return** (c, k) | 6 : **if** $c' = c$ **then** |
| | | 7 : **return** $H'(\overline{k'}, H(c))$ |
| | | 8 : **else return** $H'(s, H(c))$ |

(simplified)
Kyber


IND-CCA
adversary

# Technical Overview

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse $\mathsf{sk}' = (\mathsf{sk}, s)$ |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : $\mathsf{sk}' \leftarrow (\mathsf{sk}, s)$ | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** $(\mathsf{pk}, \mathsf{sk}')$ | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : $k \leftarrow H'(\overline{k}, H(c))$ | 5 : $\overline{k'} \leftarrow G_k(m')$ |
| | 6 : **return** $(c, k)$ | 6 : **if** $c' = c$ **then** |
| | | 7 : **return** $H'(\overline{k'}, H(c))$ |
| | | 8 : **else return** $H'(s, H(c))$ |

(simplified)
Kyber

IND-CCA
adversary

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse $\mathsf{sk}' = (\mathsf{sk}, s)$ |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : $\mathsf{sk}' \leftarrow (\mathsf{sk}, s)$ | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** $(\mathsf{pk}, \mathsf{sk}')$ | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : $k \leftarrow H'(\overline{k}, H(c))$ | 5 : $\overline{k'} \leftarrow G_k(m')$ |
| | 6 : **return** $(c, k)$ | 6 : **if** $c' = c$ **then** |
| | | 7 : **return** $H'(\overline{k'}, H(c))$ |
| | | 8 : **else return** $H'(\overline{H}(H(c)), H(c))$ |

(simplified)
Kyber modified

# Technical Overview

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse $\mathsf{sk}' = (\mathsf{sk}, s)$ |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : $\mathsf{sk}' \leftarrow (\mathsf{sk}, s)$ | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** $(\mathsf{pk}, \mathsf{sk}')$ | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : $k \leftarrow H'(\overline{k}, H(c))$ | 5 : $\overline{k'} \leftarrow G_k(m')$ |
| | 6 : **return** $(c, k)$ | 6 : **if** $c' = c$ **then** |
| | | 7 : **return** $H'(\overline{k'}, H(c))$ |
| | | 8 : **else return** $H'(s, H(c))$ |

(simplified)
Kyber

IND-CCA adversary

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse $\mathsf{sk}' = (\mathsf{sk}, s)$ |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : $\mathsf{sk}' \leftarrow (\mathsf{sk}, s)$ | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** $(\mathsf{pk}, \mathsf{sk}')$ | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : $k \leftarrow H'(\overline{k}, H(c))$ | 5 : $\overline{k'} \leftarrow G_k(m')$ |
| | 6 : **return** $(c, k)$ | 6 : **if** $c' = c$ **then** |
| | | 7 : **return** $H'(\overline{k'}, H(c))$ |
| | | 8 : **else return** $H''(H(c))$ |

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse $\mathsf{sk}' = (\mathsf{sk}, s)$ |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : $\mathsf{sk}' \leftarrow (\mathsf{sk}, s)$ | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** $(\mathsf{pk}, \mathsf{sk}')$ | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : $k \leftarrow H'(\overline{k}, H(c))$ | 5 : $\overline{k'} \leftarrow G_k(m')$ |
| | 6 : **return** $(c, k)$ | 6 : **if** $c' = c$ **then** |
| | | 7 : **return** $H'(\overline{k'}, H(c))$ |
| | | 8 : **else return** $H'(\overline{H}(H(c)), H(c))$ |

(simplified)
Kyber modified

# Technical Overview

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : return (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : $k \leftarrow H'(\overline{k}, H(c))$ | 5 : $\overline{k'} \leftarrow G_k(m')$ |
| | 6 : return (c, k) | 6 : if $c' = c$ then |
| | | 7 : return $H'(\overline{k'}, H(c))$ |
| | | 8 : else return $H'(s, H(c))$ |

(simplified)
Kyber

… pseudorandomness of QRO "$H'(s,\cdot)$".

≈

IND-CCA adversary

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : return (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : $k \leftarrow H'(\overline{k}, H(c))$ | 5 : $\overline{k'} \leftarrow G_k(m')$ |
| | 6 : return (c, k) | 6 : if $c' = c$ then |
| | | 7 : return $H'(\overline{k'}, H(c))$ |
| | | 8 : else return $H''(H(c))$ |

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : return (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : $k \leftarrow H'(\overline{k}, H(c))$ | 5 : $\overline{k'} \leftarrow G_k(m')$ |
| | 6 : return (c, k) | 6 : if $c' = c$ then |
| | | 7 : return $H'(\overline{k'}, H(c))$ |
| | | 8 : else return $H'(\overline{H}(H(c)), H(c))$ |

(simplified)
Kyber modified

# Technical Overview

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse $\mathsf{sk}' = (\mathsf{sk}, s)$ |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : $\mathsf{sk}' \leftarrow (\mathsf{sk}, s)$ | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** $(\mathsf{pk}, \mathsf{sk}')$ | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : $k \leftarrow H'(\overline{k}, H(c))$ | 5 : $\overline{k'} \leftarrow G_k(m')$ |
| | 6 : **return** $(c, k)$ | 6 : **if** $c' = c$ **then** |
| | | 7 : **return** $H'(\overline{k'}, H(c))$ |
| | | 8 : **else return** $H'(s, H(c))$ |

(simplified)
Kyber

IND-CCA adversary

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse $\mathsf{sk}' = (\mathsf{sk}, s)$ |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : $\mathsf{sk}' \leftarrow (\mathsf{sk}, s)$ | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** $(\mathsf{pk}, \mathsf{sk}')$ | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : $k \leftarrow H'(\overline{k}, H(c))$ | 5 : $\overline{k'} \leftarrow G_k(m')$ |
| | 6 : **return** $(c, k)$ | 6 : **if** $c' = c$ **then** |
| | | 7 : **return** $H'(\overline{k'}, H(c))$ |
| | | 8 : **else return** $H''(H(c))$ |

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse $\mathsf{sk}' = (\mathsf{sk}, s)$ |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : $\mathsf{sk}' \leftarrow (\mathsf{sk}, s)$ | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** $(\mathsf{pk}, \mathsf{sk}')$ | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : $k \leftarrow H'(\overline{k}, H(c))$ | 5 : $\overline{k'} \leftarrow G_k(m')$ |
| | 6 : **return** $(c, k)$ | 6 : **if** $c' = c$ **then** |
| | | 7 : **return** $H'(\overline{k'}, H(c))$ |
| | | 8 : **else return** $H'(\overline{H}(H(c)), H(c))$ |

(simplified)
Kyber modified

# Technical Overview

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : return (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : $k \leftarrow H'(\overline{k}, H(c))$ | 5 : $\overline{k'} \leftarrow G_k(m')$ |
| | 6 : return (c, k) | 6 : if $c' = c$ then |
| | | 7 : return $H'(\overline{k'}, H(c))$ |
| | | 8 : else return $H'(s, H(c))$ |

(simplified)
Kyber

IND-CCA adversary

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : return (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : $k \leftarrow H'(\overline{k}, H(c))$ | 5 : $\overline{k'} \leftarrow G_k(m')$ |
| | 6 : return (c, k) | 6 : if $c' = c$ then |
| | | 7 : return $H'(\overline{k'}, H(c))$ |
| | | 8 : else return $H''(H(c))$ |

… generalized “One-Way To Hiding (OW2H)” lemma in the QROM [Ambainis-Hamburg-Unruh@Crypto’19].

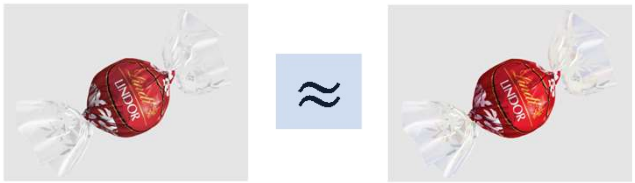
| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : return (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : $k \leftarrow H'(\overline{k}, H(c))$ | 5 : $\overline{k'} \leftarrow G_k(m')$ |
| | 6 : return (c, k) | 6 : if $c' = c$ then |
| | | 7 : return $H'(\overline{k'}, H(c))$ |
| | | 8 : else return $H'(\overline{H}(H(c)), H(c))$ |

(simplified)
Kyber modified

# Technical Overview

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : return (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : $k \leftarrow H'(\overline{k}, H(c))$ | 5 : $\overline{k'} \leftarrow G_k(m')$ |
| | 6 : return (c, k) | 6 : if $c' = c$ then |
| | | 7 : return $H'(\overline{k'}, H(c))$ |
| | | 8 : else return $H'(s, H(c))$ |

(simplified)
Kyber





IND-CCA adversary

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : return (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : $k \leftarrow H'(\overline{k}, H(c))$ | 5 : $\overline{k'} \leftarrow G_k(m')$ |
| | 6 : return (c, k) | 6 : if $c' = c$ then |
| | | 7 : return $H'(\overline{k'}, H(c))$ |
| | | 8 : else return $H''(H(c))$ |

… pseudorandomness of QRO "$H'(s,\cdot)$".

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_{\$} \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_{\$} \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : return (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : $k \leftarrow H'(\overline{k}, H(c))$ | 5 : $\overline{k'} \leftarrow G_k(m')$ |
| | 6 : return (c, k) | 6 : if $c' = c$ then |
| | | 7 : return $H'(\overline{k'}, H(c))$ |
| | | 8 : else return $H'(\overline{H}(H(c)), H(c))$ |

(simplified)
Kyber modified

… generalized "One-Way To Hiding (OW2H)" lemma in the QROM [Ambainis-Hamburg-Unruh@Crypto'19].

# Discussion

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_\$ \mathcal{M}$ | 1 : Parse $\mathsf{sk}' = (\mathsf{sk}, s)$ |
| 2 : $s \leftarrow_\$ \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : $\mathsf{sk}' \leftarrow (\mathsf{sk}, s)$ | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** $(\mathsf{pk}, \mathsf{sk}')$ | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : **return** $(c, \overline{k})$ | 5 : **if** $c' = c$ **then** |
| | | 6 : **return** $G_k(m')$ |
| | | 7 : **else return** $G_k(s, c)$ |

$\mathsf{FO}_m^{\not\perp}$ [Hofheinz-Hövelmanns-Kiltz@TCC'17]

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_\$ \mathcal{M}$ | 1 : Parse $\mathsf{sk}' = (\mathsf{sk}, s)$ |
| 2 : $s \leftarrow_\$ \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : $\mathsf{sk}' \leftarrow (\mathsf{sk}, s)$ | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : **return** $(\mathsf{pk}, \mathsf{sk}')$ | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : $k \leftarrow H'(\overline{k}, H(c))$ | 5 : $\overline{k'} \leftarrow G_k(m')$ |
| | 6 : **return** $(c, k)$ | 6 : **if** $c' = c$ **then** |
| | | 7 : **return** $H'(\overline{k'}, H(c))$ |
| | | 8 : **else return** $H'(s, H(c))$ |

Kyber (simplified)

# Discussion

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_\$ \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_\$ \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : return (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : return $(c, \overline{k})$ | 5 : if $c' = c$ then |
| | | 6 : return $G_k(m')$ |
| | | 7 : else return $G_k(s, c)$ |

$\mathsf{FO}_m^{\not\perp}$ [Hofheinz-Hövelmanns-Kiltz@TCC'17]

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_\$ \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_\$ \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : return (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : $k \leftarrow H'(\overline{k}, H(c))$ | 5 : $\overline{k'} \leftarrow G_k(m')$ |
| | 6 : return $(c, k)$ | 6 : if $c' = c$ then |
| | | 7 : return $H'(\overline{k'}, H(c))$ |
| | | 8 : else return $H'(s, H(c))$ |

Kyber (simplified)

$H'(\overline{k}, H(c))$ "quantum indifferentiable" from $H''(\overline{k}, c)$, as proven in [Zhandry@Crypto'19].

# Discussion

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_\$ \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_\$ \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : return (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : return $(c, \overline{k})$ | 5 : if $c' = c$ then |
| | | 6 : return $G_k(m')$ |
| | | 7 : else return $G_k(s, c)$ |

$\mathsf{FO}^{\not\perp}_m$ [Hofheinz-Hövelmanns-Kiltz@TCC'17]

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_\$ \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_\$ \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : return (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : $k \leftarrow H'(\overline{k}, H(c))$ | 5 : $\overline{k'} \leftarrow G_k(m')$ |
| | 6 : return $(c, k)$ | 6 : if $c' = c$ then |
| | | 7 : return $H'(\overline{k'}, H(c))$ |
| | | 8 : else return $H'(s, H(c))$ |

Kyber (simplified)

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_\$ \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_\$ \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : return (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m, c)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : return $(c, \overline{k})$ | 5 : if $c' = c$ then |
| | | 6 : return $G_k(m', c)$ |
| | | 7 : else return $G_k(s, c)$ |

$\mathsf{FO}^{\not\perp}$ [Hofheinz-Hövelmanns-Kiltz @TCC'17]

$H'(\overline{k}, H(c))$ "quantum indifferentiable" from $H''(\overline{k}, c)$, as proven in [Zhandry@Crypto'19].

# Discussion

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_\$ \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_\$ \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : return (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : return $(c, \overline{k})$ | 5 : if $c' = c$ then |
| | | 6 : return $G_k(m')$ |
| | | 7 : else return $G_k(s, c)$ |

$\mathsf{FO}_m^{\not\perp}$ [Hofheinz-Hövelmanns-Kiltz@TCC'17]

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_\$ \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_\$ \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : return (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : $k \leftarrow H'(\overline{k}, H(c))$ | 5 : $\overline{k'} \leftarrow G_k(m')$ |
| | 6 : return $(c, k)$ | 6 : if $c' = c$ then |
| | | 7 : return $H'(\overline{k'}, H(c))$ |
| | | 8 : else return $H'(s, H(c))$ |

Kyber (simplified)

| KGen′ | Encap(pk) | Decap(sk′, c) |
|---|---|---|
| 1 : (pk, sk) ← KGen | 1 : $m \leftarrow_\$ \mathcal{M}$ | 1 : Parse sk′ = (sk, s) |
| 2 : $s \leftarrow_\$ \mathcal{M}$ | 2 : $r \leftarrow G_r(m)$ | 2 : $m' \leftarrow \mathsf{Dec}(\mathsf{sk}, c)$ |
| 3 : sk′ ← (sk, s) | 3 : $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m; r)$ | 3 : $r' \leftarrow G_r(m')$ |
| 4 : return (pk, sk′) | 4 : $\overline{k} \leftarrow G_k(m, c)$ | 4 : $c' \leftarrow \mathsf{Enc}(\mathsf{pk}, m'; r')$ |
| | 5 : return $(c, \overline{k})$ | 5 : if $c' = c$ then |
| | | 6 : return $G_k(m', c)$ |
| | | 7 : else return $G_k(s, c)$ |

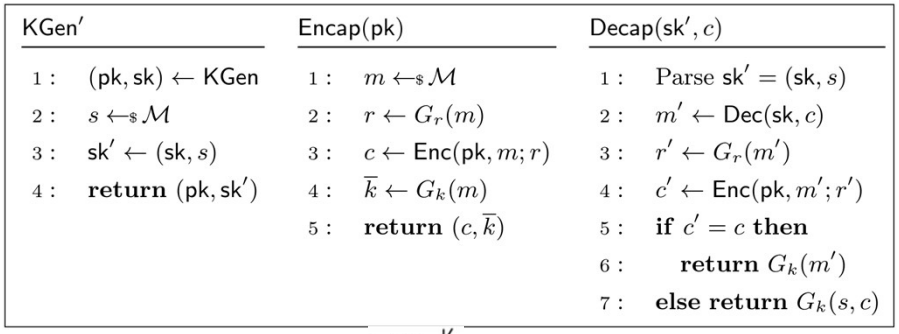
$\mathsf{FO}^{\not\perp}$ [Hofheinz-Hövelmanns-Kiltz @TCC'17]

≈

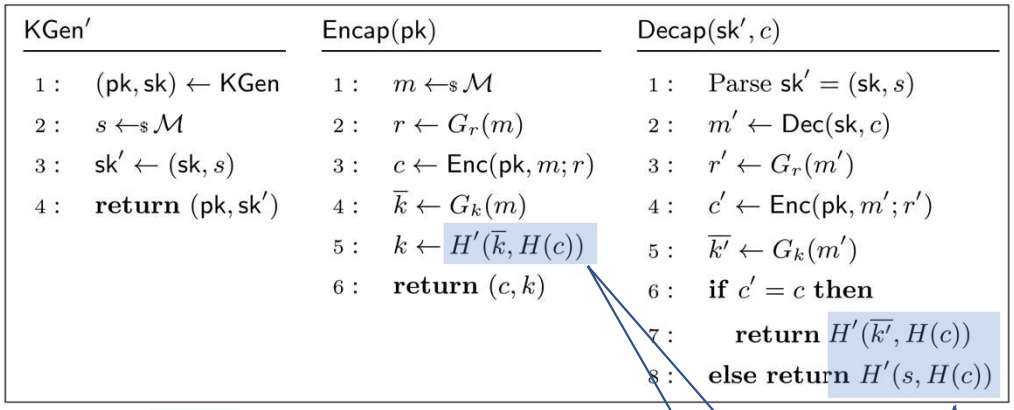
IND-CCA security equivalent upto "indifferentiability loss", as implied by [Grubbs-Maram-Paterson@Eurocrypt'22].

$H'(\overline{k}, H(c))$ "quantum indifferentiable" from $H''(\overline{k}, c)$, as proven in [Zhandry@Crypto'19].

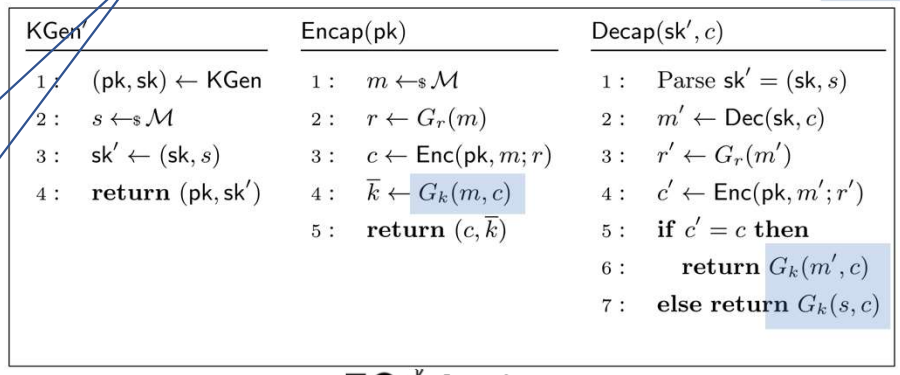
# Discussion




$\mathsf{FO}_m^{\not\perp}$ [Hofheinz-Hövelmanns-Kiltz@TCC'17]




Kyber (simplified)




$\mathsf{FO}^{\not\perp}$ [Hofheinz-Hövelmanns-Kiltz @TCC'17]

IND-CCA security equivalent, as shown in [Bindel-Hamburg-Hövelmanns-Hülsing-Persichetti@TCC'19].

IND-CCA security equivalent upto "indifferentiability loss", as implied by [Grubbs-Maram-Paterson@Eurocrypt'22].

$H'(\bar{k}, H(c))$ "quantum indifferentiable" from $H''(\bar{k}, c)$, as proven in [Zhandry@Crypto'19].