Failing gracefully: Decryption failures and the Fujisaki-Okamoto transform

Kathrin Hövelmanns Andreas Hülsing Christian Majenz
Motivation

Computational problem
(LWE, NTRU, SD)...

PKE
Passively secure
(OW/IND-CPA)

Key Encapsulation
IND-CCA

Decryption failures and the FO transform - Kathrin Hövelmanns
Motivation

**Computational problem**
(LWE, NTRU, SD)...

**PKE**
Passively secure
(OW/IND-CPA)

**Key Encapsulation**
IND-CCA

---

**Fujisaki-Okamoto transform**

Originally (FO99): no decryption failures (lattices, codes 😐)

Revisited (HHK17):
☑ small failure probability \( \delta \)
different rejection methods
Motivation

Computational problem (LWE, NTRU, SD)...

PKE
Passively secure (OW/IND-CPA)

Key Encapsulation
IND-CCA

Fujisaki-Okamoto transform
Originally (FO99): no decryption failures (lattices, codes 🙄)
Revisited (HHK17):
☑️ small failure probability \( \delta \)
different rejection methods

Weird QROM thing 1
ROM: Rejection-method-agnostic
Quantum ROM:
Different methods \( \rightarrow \) bounds vastly differ

Decryption failures and the FO transform - Kathrin Hövelmanns
Motivation

Computational problem
(LWE, NTRU, SD)...

PKE
Passively secure
(OW/IND-CPA)

Key Encapsulation
IND-CCA

Fujisaki-Okamoto transform
Originally (FO99): no decryption failures (lattices, codes 😐)
Revisited (HHK17):
☑ small failure probability $\delta$
different rejection methods

Weird QROM thing 1
ROM: Rejection-method-agnostic
Quantum ROM:
Different methods $\rightarrow$ bounds vastly differ

Weird QROM thing 2
Grover-like $\delta$ – term: $q^2 \cdot \delta$
...can attackers quantum search?

Suboptimal bounds?
**Motivation**

- **Computational problem** (LWE, NTRU, SD)...
  - PKE Passively secure (OW/IND-CPA)

- **Fujisaki-Okamoto transform**
  - Originally (FO99): no decryption failures (lattices, codes 😐)
  - Revisited (HHK17):
    - ✓ small failure probability $\delta$
    - different rejection methods

- **Weird QROM thing 1**
  - ROM: Rejection-method-agnostic
  - Quantum ROM: Different methods $\rightarrow$ bounds vastly differ

- **Weird QROM thing 2**
  - Grover-like $\delta$ – term: $q^2 \cdot \delta$
  - ...can attackers quantum search?

**Suboptimal bounds?**

**Applicability issue**

- Concrete $\delta$ – estimations ⚡️ security proofs
$\delta$ - estimations vs security proofs

$\delta \triangleq$ advantage in

**Correctness game**

\[ c \leftarrow \text{Enc}(pk, m) \]

\[ \text{return } [\text{Dec}(sk, c) = m] \]

\[ (pk, sk) \]

\[ m \]

**Attacker**

**Applicability issue**

Concrete $\delta$ – estimations $\Rightarrow$ security proofs
$\delta$ - estimations vs security proofs

$\delta \triangleq$ advantage in

**Correctness game**

\[
c \leftarrow \text{Enc}(pk, m) \\
\text{return } [\text{Dec}(sk, c) = m]
\]

**(pk, sk)**

Attacker

```
m
```

Necessary?

**Applicability issue**

Concrete $\delta$ – estimations vs security proofs
\( \delta \) - estimations vs security proofs

\( \delta \triangleq \) advantage in

Correctness game

\[
c \leftarrow \text{Enc}(pk, m) \\
\text{return } [\text{Dec}(sk, c) = m]
\]

Attacker

Necessary?

(\( pk, sk \))

\( m \)

\( \delta \)-estimator scripts:

\( \triangleq \) advantage in game without \( sk \)

Applicability issue

Concrete \( \delta \) – estimations

security proofs

\( \triangleright \) observed by Manuel Barbosa
Our results (nutshell)

Tighter bound for FO with explicit rejection \((FO^\perp)\) for randomised schemes:

\[ \rightarrow \text{Aligns QROM results for the two rejection types} \]

Bounds work with sk-less failure notions \(\rightarrow\) estimator-script-compatible ☺
Our results

Tighter bound for FO with explicit rejection (FO⊥) for randomised schemes:

$$\text{INDCCA}(\text{FO}^\perp(\text{PKE})) \leq \text{INDCPA}(\text{FO}^\perp(\text{PKE})) + T_{\text{SPREAD}} + T_{\text{FAIL}}$$
Our results

Tighter bound for FO with explicit rejection (FO⊥) for randomised schemes:

\[
\text{INDCCA}(\text{FO}^\perp(\text{PKE})) \leq \text{INDCPA}(\text{FO}^\perp(\text{PKE})) + T_{\text{SPREAD}} + T_{\text{FAIL}}
\]

Essentially \(4 \cdot \sqrt{\# \text{ queries}} \cdot \text{INDCPA}(\text{PKE})\)

How? Semi-classical One-Way to Hiding (tailored)

Why not double-sided? Same bound

Why not MRM? \(4 \cdot \# \text{ queries}^2 \cdot \text{INDCPA}(\text{PKE})\)
Our results

Tighter bound for FO with explicit rejection (FO⊥) for randomised schemes:

\[ \text{INDCCA}(\text{FO}^\perp(\text{PKE})) \leq \text{INDCPA}(\text{FO}^\perp(\text{PKE})) + T_{\text{SPREAD}} + T_{\text{FAIL}} \]

\[ T_{\text{SPREAD}} = \frac{2^{65} \cdot q}{\sqrt{2^\gamma}} \]

\( \gamma \): PKE spreadness (‘entropy’)

DFMS22: \[ \frac{24 \cdot q \cdot \sqrt{q \cdot q_{\text{Decaps}}}}{4 \sqrt{2^\gamma}} \]

\( q \): # RO queries \hspace{1cm} q_{\text{Decaps}}: # CCA queries (NIST: \( 2^{64} \))
Our results

Tighter bound for FO with explicit rejection ($\text{FO}^\perp$) for randomised schemes:

\[
\text{INDCCA}(\text{FO}^\perp(\text{PKE})) \leq \text{INDCPA}(\text{FO}^\perp(\text{PKE})) + T_{\text{SPREAD}} + T_{\text{FAIL}}
\]

Bound also works for implicit rejection (due to BH+19).

Conjecture

Implicit: smaller $T_{\text{SPREAD}}$ possible
Our results

Tighter bound for FO with explicit rejection (FO\textdagger) for randomised schemes:

\[
\text{INDCCA}(\text{FO}^\perp(\text{PKE})) \leq \text{INDCPA}(\text{FO}^\perp(\text{PKE})) + T_{\text{SPREAD}} + T_{\text{FAIL}}
\]

\(T_{\text{FAIL}}\): failure-finding game advantage \textbf{without sk}

Previous work:

<table>
<thead>
<tr>
<th>Implicit</th>
<th>Explicit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Essentially 8q^2 \cdot \delta</td>
<td>24 \cdot q^2 \cdot \delta</td>
</tr>
</tbody>
</table>
Our results

Tighter bound for FO with explicit rejection ($\text{FO}^\perp$) for randomised schemes:

$$\text{INDCCA(FO}^\perp(\text{PKE})) \leq \text{INDCPA(FO}^\perp(\text{PKE})) + T_{\text{SPREAD}} + T_{\text{FAIL}}$$

$T_{\text{FAIL}}$: failure-finding game advantage without sk

3 ways to bound:

Failure attacker with CCA oracle, somewhat contrived ROM:

$$T_{\text{FAIL}} = \text{FAILURE} - \text{CCA (PKE}^{\text{derand}})$$

Previous work:

- Implicit: Essentially $8q^2 \cdot \delta$
- Explicit: $24 \cdot q^2 \cdot \delta$
Our results

Tighter bound for FO with explicit rejection (FO⊥) for randomised schemes:

$$\text{INDCCA}(\text{FO}^\bot(\text{PKE})) \leq \text{INDCPA}(\text{FO}^\bot(\text{PKE})) + T_{\text{SPREAD}} + T_{\text{FAIL}}$$

$T_{\text{FAIL}}$: failure-finding game advantage without sk

3 ways to bound:

Failure attacker w'out CCA oracle, somewhat contrived ROM:

$$T_{\text{FAIL}} = q_{\text{Decaps}} \cdot \text{FAILURE} - \text{CPA (PKE}^{\text{derand}})$$

Previous work:

Implicit: Essentially $8q^2 \cdot \delta$

Explicit: $24 \cdot q^2 \cdot \delta$
Our results

Tighter bound for FO with explicit rejection (FO⊥) for randomised schemes:

\[
\text{INDCCA}(\text{FO}^\perp(\text{PKE})) \leq \text{INDCPA}(\text{FO}^\perp(\text{PKE})) + T_{\text{SPREAD}} + T_{\text{FAIL}}
\]

\(T_{\text{FAIL}}\): failure-finding game advantage without sk

3 ways to bound:

Breaking down FAILURE — CPA (PKE_{\text{derand}}), generically

• in terms of PKE, no contrived ROM
• fine-grained term compatible with existing \(\delta\)-estimator scripts

Previous work:

Implicit: Essentially \(8q^2 \cdot \delta\)
Explicit: \(24 \cdot q^2 \cdot \delta\)

\(\text{PKE}_{\text{derand}}\): \(c = \text{Encrypt}(pk, m; r), \ r = \text{Hash}_{\text{rand}}(m)\)
Our results

FAILURE − CPA (PKE\textsuperscript{derand}) = sum of two bounds:

• Finding non-generic (key-dependent) failures for PKE
• Finding generic (key-independent) failures for PKE\textsuperscript{derand}

\[ \text{PKE}_{\text{derand}}: c = \text{Encrypt}(pk, m; r), \quad r = \text{Hash}_{\text{rand}}(m) \]
Our results

FAILURE − CPA (PKE_{derand}) = sum of two bounds:

• Finding non-generic (key-dependent) failures for PKE
• Finding generic (key-independent) failures for PKE_{derand}

NonGenFail game

Task: Tell key pairs apart with single ‘does this fail’ query
Our results

FAILURE $-\text{CPA (PKE}^{\text{derand}}\text{)}$ = sum of two bounds:

- Finding non-generic (key-dependent) failures for PKE
- Finding generic (key-independent) failures for PKE$^{\text{derand}}$

**NonGenFail game**

Task: Tell key pairs apart with single ‘does this fail’ query

Decryption failures and the FO transform - Kathrin Hövelmanns
Our results

FAILURE – CPA ($\text{PKE}^\text{derand}$) = sum of two bounds:

- Finding non-generic (key-dependent) failures
- Finding generic (key-independent) failures for $\text{PKE}^\text{derand}$

**GenFail game**

Task: Find $m$ failing for $\text{PKE}^\text{derand}$ without even knowing $pk$

\[ (pk, sk) \leftarrow \text{KG} \]
\[ c := \text{Enc}(pk, m; \text{Hash}_{rand}(m)) \]
\[ m' := \text{Dec}(sk, c) \]
\[ \text{return } [m' \neq m] \]
Our results

FAILURE – CPA \( (\text{PKE}^{\text{derand}}) \) = sum of two bounds:

- Finding non-generic (key-dependent) failures
- **Finding generic (key-independent) failures** for \( \text{PKE}^{\text{derand}} \)

**GenFail game**

Analysis via new QROM 'find large values' bounds

Task: Find \( m \) failing for \( \text{PKE}^{\text{derand}} \) without even knowing \( pk \)

\[
\begin{align*}
(pk, sk) &\leftarrow \text{KG} \\
c &:= \text{Enc}(pk, m; \text{Hash}_{\text{rand}}(m)) \\
m' &:\text{ Dec}(sk, c) \\
\text{return} \ [m' \neq m]
\end{align*}
\]

\( \text{PKE}^{\text{derand}} \): \( c = \text{Encrypt}(pk, m; r), r = \text{Hash}_{\text{rand}}(m) \)
Finding generic failures

‘Generic Failure’ term = \( \tilde{\delta} + T_{\tilde{\delta}} \):

\[
T_{\tilde{\delta}} \approx \left( \sqrt{-\ln(\tilde{\delta})} + \sqrt{\ln(q_{RO})} \right) \cdot \tilde{\delta} \quad \text{if failure tail envelope has Gaussian tail bound}
\]

Otherwise:

\[
T_{\tilde{\delta}} \approx q_{RO} \cdot \text{decryption failure rate variance}
\]

\( \tilde{\delta} \) := computed \( \delta \)-estimate

Conjecture

Lattice-based: variance very small

pessimistic: < \( \delta \)
Proof technique: Extractable QROM (DFMS22)

Idea: ROM-like reduction via preimage extraction

FO proof:

\[ 0 = \text{Hash}_{\text{rand}} : M \rightarrow R \]

CCA simulation:
Book-keep \( \text{Hash}_{\text{rand}} \) queries
Proof technique: Extractable QROM (DFMS22)

**Idea:** ROM-like reduction via preimage extraction

QROM $O: X \rightarrow Y$ via compressed oracle (Zha19)

**FO proof:**

$O = \text{Hash}_{\text{rand}}: M \rightarrow R$
Proof technique: Extractable QROM (DFMS22)

**Idea:** ROM-like reduction via preimage extraction

QROM $O: X \rightarrow Y$ via compressed oracle (Zha19)

+ interface $\text{Extract}_f$ for $f: X \times Y \rightarrow T$:

$\text{Extract}_f (t)$:
- Collapse oracle database such that
  - for one $x$, $f(x, y) = t$ for all $y$ in $x$'s database superposition
- Return $x$

**FO proof:**

$O = \text{Hash}_{\text{rand}}: M \rightarrow R$
Proof technique: Extractable QROM (DFMS22)

**Idea:** ROM-like reduction via preimage extraction

QROM $O: X \to Y$ via compressed oracle (Zha19)

+ **interface** $\text{Extract}_f$ for $f: X \times Y \to T$

$\text{Extract}_f(t)$:

- Collapse oracle database such that
  - for one $x$, $f(x, y) = t$ for all $y$ in $x$'s database superposition
- Return $x$

**FO proof:**

$O = \text{Hash}_{\text{rand}}: M \to R$

$f = \text{Encrypt}: M \times R \to C$

$\text{Extract}_f(c) =$ ‘preimage’ $m$
Proof technique: Extractable QROM (DFMS22)

**Idea:** ROM-like reduction via preimage extraction

QROM $O: X \rightarrow Y$ via compressed oracle (Zha19)

+ interface $\text{Extract}_f$ for $f: X \times Y \rightarrow T$:

$\text{Extract}_f (t)$:

Collapse oracle database such that
- for one $x$, $f(x, y) = t$ for all $y$ in $x$’s database superposition

Return $x$

$\text{Extract}_f$ commutes nicely with $O$-operations for sufficiently surprising $f$.

**FO proof:**

$O = \text{Hash}_\text{rand}: M \rightarrow R$

$f = \text{Encrypt}: M \times R \rightarrow C$

$\text{Extract}_f (c) = \text{‘preimage’ } m$

‘Surprising’ $\triangleq$ PKE spreadness
Proof technique: Extractable QROM (DFMS22)

**Idea:** ROM-like reduction via preimage extraction

QROM $O: X \rightarrow Y$ via compressed oracle (Zha19)

+ interface $\text{Extract}_f$ for $f: X \times Y \rightarrow T$:

$\text{Extract}_f(t)$:

- Collapse oracle database such that
  - for one $x$, $f(x, y) = t$ for all $y$ in $x$’s
database superposition
- Return $x$

$\text{Extract}_f$ commutes nicely with $O$-operations for sufficiently surprising $f$.

**FO proof:**

$O = \text{Hash}_\text{rand}: M \rightarrow R$

$f = \text{Encrypt}: M \times R \rightarrow C$

$\text{Extract}_f(c) = \text{‘preimage’} m$

‘Surprising’ $\triangleq$ PKE spreadness

**Contribution:** extractable QROM OWTH
Conclusion

Tighter bound for $\text{FO}^\bot$, alternative bound for implicit

- **FAILURE** $\rightarrow$ **CCA** ($\text{PKE}^{\text{derand}}$)
  - **FAILURE** $\rightarrow$ **CPA** ($\text{PKE}^{\text{derand}}$)
  - **NONGENFAIL** ($\text{PKE}$) + **GENFAIL** ($\text{PKE}^{\text{derand}}$)
  - sublogarithmic $\cdot \delta$- estimate

**QROM tools**: ‘large value search’ results + proof strategy:
- Reduction needs to
  - Book-keep queries
  - Simulate hash values via $\mathcal{F}$

**Trade-off**: Possibly better bounds vs. less work for concrete schemes

- Extractable OWTH
**Bonus: IND-CCA of FO in the ROM**

<table>
<thead>
<tr>
<th>FO encapsulation</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Key:</strong> k = Hash\textsubscript{key}(m), m = $</td>
</tr>
<tr>
<td><strong>Ciphertext:</strong> r = Hash\textsubscript{rand}(m)</td>
</tr>
<tr>
<td>c = Encrypt(pk, m; r)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>FO decapsulation</th>
</tr>
</thead>
<tbody>
<tr>
<td>m' = Decrypt(sk, c)</td>
</tr>
<tr>
<td>k = Hash\textsubscript{key}(m')</td>
</tr>
</tbody>
</table>

**IND:** Breaking IND = breaking PKE

**CCA:** Book-keep queries to Hash\textsubscript{rand}

Look up m encrypting to c

Return Hash\textsubscript{key}(m)

Decaps simulation fails if:
- c valid, but m not yet queried $\rightarrow$ $\gamma$-spreadness
- c stems from ‘failing’ m:
  - c = Encrypt(m) with r = Hash\textsubscript{rand}(m)
  - Decrypt(c) $\neq$ m

Correctness game against derandomised PKE

Advantage $< q_{RO} \cdot \delta$

Decryption failures and the FO transform - Kathrin Hövelmanns
In the quantum ROM?

<table>
<thead>
<tr>
<th>FO encapsulation</th>
<th>FO decapsulation</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Key:</strong> ( k = \text{Hash}_{\text{key}}(m), ; m = $ )</td>
<td><strong>m’ = Decrypt(sk, c)</strong></td>
</tr>
<tr>
<td><strong>Ciphertext:</strong> ( r = \text{Hash}_{\text{rand}}(m) )</td>
<td><strong>k = \text{Hash}_{\text{key}}(m’)</strong></td>
</tr>
<tr>
<td>( c = \text{Encrypt}(pk, m; r) )</td>
<td></td>
</tr>
</tbody>
</table>

**IND:** Breaking IND = breaking PKE

**CCA:** Simulation that still fails for failing \( m \)

Replace \( \text{Hash}_{\text{rand}} \) with ‘perfectly correct’ oracle

Advantage < \( q_{\text{RO}}^2 \cdot \delta \)
In the quantum ROM?

**FO encapsulation**

Key: \( k = \text{Hash}_{\text{key}}(m), \ m = \$ \)

Ciphertext: \( r = \text{Hash}_{\text{rand}}(m) \)
\( c = \text{Encrypt}(pk, m; r) \)

**FO decapsulation**

\( m' = \text{Decrypt}(sk, c) \)
\( k = \text{Hash}_{\text{key}}(m') \)

- **IND:** Breaking IND = breaking PKE
- **CCA:** Book-keep queries to \( \text{Hash}_{\text{rand}} \)
  Look up \( m \) encrypting to \( c \)
  Return \( \text{Hash}_{\text{key}}(m) \)

This work

ROM-like simulation via extractable QROM
+ OWTH in extractable QROM

☑ One-way to hiding (OWTH)
U14, AHU19, BH+19, KS+21
Bonus: tail bound of failure tail envelope

\[ T_{\tilde{\delta}} \approx \left( \sqrt{-\ln(\tilde{\delta})} + \sqrt{\ln(q_{RO})} \right) \cdot \tilde{\delta} \text{ if failure tail envelope has Gaussian tail bound} \]

Failure tail envelope: \( \tau(t) := \max_m \Pr_r \left[ \Pr_{pk,sk} [m, r \text{ fail for } pk, sk] \geq t \right] \)

Gaussian tail bound: \( \tau(t) \leq \exp \left( -\frac{1}{\tilde{\delta}^2} \cdot (t - \delta_{ik})^2 \right) \)

\[ \max_m \Pr_{r, pk, sk} [m, r \text{ fail for } pk, sk] \]
Bonus: Compressed oracle (Zha19)

- Oracle database initialised to $D := \bigotimes_{x \in \text{query domain}} |x, \bot >_{D_x}$
- Process queries $|x, y >$ by applying
  - $F_{D_x}$ to output register of $D_x$
    \[ F_{D_x} |\psi > := \begin{cases} \text{uniform superposition,} & |\psi > = \bot \\ \bot, & |\psi > = \text{uniform superposition} \\ |\psi >, & |\psi > = \text{orthogonal to } \bot, \text{uniform} \end{cases} \]
  - CNOT$_{D_x:Y}$ to $D_x$, query output register $Y$
  - $F_{D_x}$ to output register of $D_x$