The Insecurity of Masked Comparisons: SCAs on ML-KEM's FO-Transform

Julius Hermelink[1] Kai-Chun Ning[1] Richard Petri[1] Emanuele Strieder[2,3]

[1]Max Planck Institute for Security and Privacy

[2]Fraunhofer AISEC

[3]Technical University of Munich

Post-Quantum Cryptography

We are in the process of migrating to Post-Quantum Cryptography.

NIST started a standardization process in 2016.

- Fourth round ongoing.
- Four candidates already selected.
- Kyber selected as KEM (Kyber $\mapsto$ ML-KEM).

Post-Quantum Cryptography

We are in the process of migrating to Post-Quantum Cryptography.

NIST started a standardization process in 2016.

- Fourth round ongoing.
- Four candidates already selected.
- Kyber selected as KEM (Kyber $\mapsto$ ML-KEM).

ML-KEM is already actively being used

- ML-KEM used in Signal, iMessage, ...
- Available in Chrome, Firefox, ...

Post-Quantum Cryptography

We are in the process of migrating to Post-Quantum Cryptography.

NIST started a standardization process in 2016.

- Fourth round ongoing.
- Four candidates already selected.
- Kyber selected as KEM (Kyber $\mapsto$ ML-KEM).

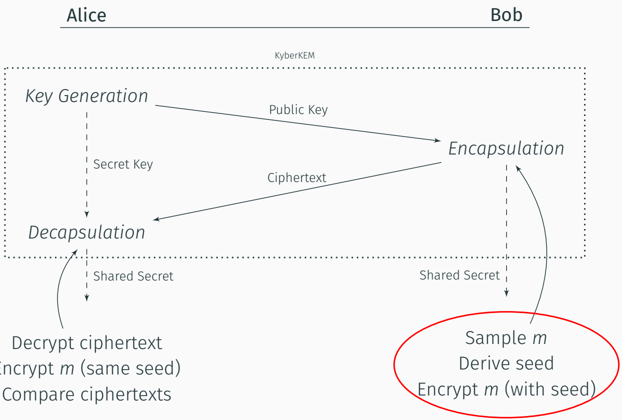
ML-KEM is already actively being used

- ML-KEM used in Signal, iMessage, ...
- Available in Chrome, Firefox, ...

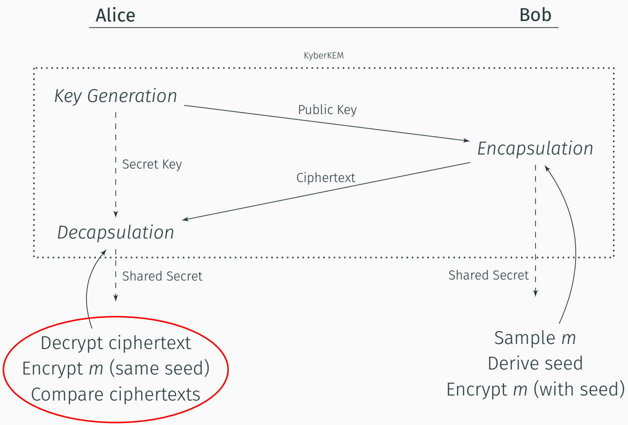
We have to assume that usage on embedded devices will soon become widespread.

The Fujisaki-Okamoto Transform in ML-KEM

Key Encapsulation Mechanism derived from Public Key Encryption



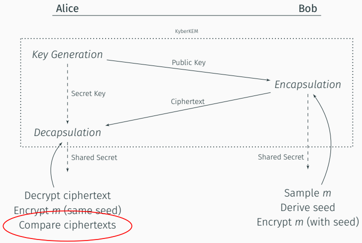
Key Encapsulation Mechanism derived from Public Key Encryption



Masked Comparisons – A Highly Sensitive Operation

In ML-KEM:

- We compare a re-computed ($\mathtt{ct}'$) and a submitted ciphertext ($\mathtt{ct}$).
- If outcome is leaked, chosen-ciphertext attacks are possible (see, e.g., [BDH+21; DHP+22; RRD+23]).



Attacker can force two cases (see, e.g., [BDH+21]):

1. $\mathtt{ct}$ and $\mathtt{ct}'$ differ in one coefficient.
2. $\mathtt{ct}$ and $\mathtt{ct}'$ differ in about half the bits.

On embedded devices, we have to protect against power side channels.

Masked Comparisons – Most Recent Proposal

Most recent protected method [DBV23] works by

- $\Delta\mathtt{ct} = \mathtt{ct} - \mathtt{ct}'$ in Boolean masking.
- Multiply shares of coefficients with random value over finite field.
- Check if shares sums zero.

Attacker targets Boolean shared $\Delta\mathtt{ct}$:

Formally verified in the t-probing model.

Building a Leakage Model

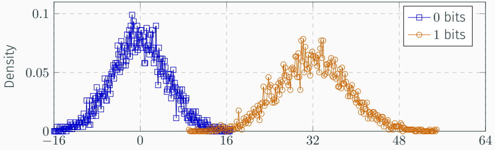
Implementation needs to multiply Δ`ct`-bits with random value

We suspected: this should amplify leakage of secret bits

Implementation needs to multiply Δ`ct`-bits with random value

We suspected: this should amplify leakage of secret bits

Our model (simulation for $\sigma = 5$):

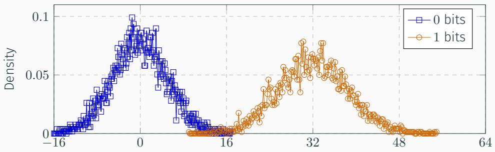


Power consumption of processing a single shared bit.

Implementation needs to multiply Δ`ct`-bits with random value

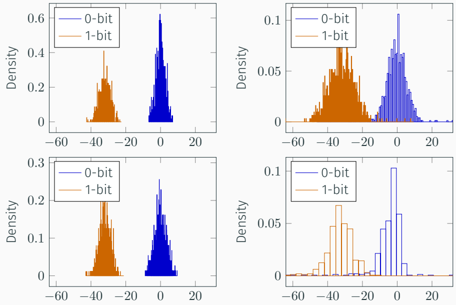
We suspected: this should amplify leakage of secret bits

Our model (simulation for $\sigma = 5$):



Power consumption of processing a single shared bit.

Actual leakage confirms our model:



How to classify traces based on our leakage model?

General attack:

1. Submit n chosen-ciphertexts potentially causing decryption failures and record power traces.
2. Classify into decryption failures and decryption successes.
3. Derive inequalities, recover secret key with [HMS+23].

How to classify traces based on our leakage model?

General attack:

1. Submit n chosen-ciphertexts potentially causing decryption failures and record power traces.
2. Classify into decryption failures and decryption successes.
3. Derive inequalities, recover secret key with [HMS+23].

Classifying traces based on model:

- Goal: Learn distributions for 0 and 1 bits for each bit of $\Delta\mathtt{ct}$.
- Then: Classify bits based on measurement and distribution.
- Based on "reliably" classified bits: Decide if failure (at least one 1-bit) or success (only 0-bits).

How to classify traces based on our leakage model?

General attack:

1. Submit n chosen-ciphertexts potentially causing decryption failures and record power traces.
2. Classify into decryption failures and decryption successes.
3. Derive inequalities, recover secret key with [HMS+23].

Classifying traces based on model:

- Goal: Learn distributions for 0 and 1 bits for each bit of $\Delta\mathtt{ct}$.
- Then: Classify bits based on measurement and distribution.
- Based on “reliably” classified bits: Decide if failure (at least one 1-bit) or success (only 0-bits).

To classify trace: Classifying one 1-bit reliably suffices.

To recover secret key: 55% trace classification success rate suffices.

Designing Attacks

How to classify traces based on our leakage model?

Instead of a profiled attack:

Shared bits correspond to locations in power trace.

- Each ciphertext gives trace.
- Vertical: Over multiple traces, same relative location.
- Horizontal: Same trace, different locations.

Vertical Analysis: Learn joint distribution individually for each shared bit from all traces.

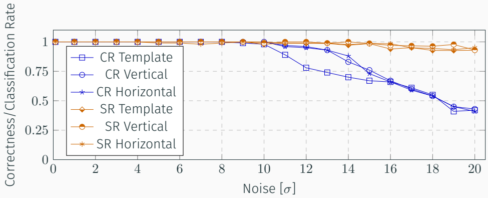
Horizontal Analysis: Learn (the same) joint distribution for all shared bits from one trace.

Then: Separate distributions into two normal distributions.

Results

We simulated the attacks for different noise levels.

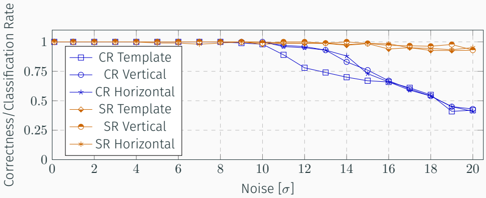
Simulated results with 4 shares:



t-probing security proven, but:
Noise/masking order necessary to prevent attacks extraordinarily high.

We simulated the attacks for different noise levels.

Simulated results with 4 shares:



t-probing security proven, but:
Noise/masking order necessary to prevent attacks extraordinarily high.

Why do these attacks work so well?

- Information (1 bit!) is stored/processed in several hundred bits.
- Slight advantage over guessing suffices for attack.
- No instructions for used arithmetic amplifies leakage.

$\rightarrow$ High noise requirements.

Summary and Conclusion

Summary

To assess the security of a recent masked comparison proposal, we:

- Built a leakage model based on the noisy HW model.
- Derived several attacks working under high noise/masking orders.
- Replaced profiling by vertical/horizontal analysis.
- Verified model and attacks on several devices.

Conclusion

In particular for post-quantum schemes:

- Even if t-probing secure, noise/masking orders necessary to prevent the attack in practice may be unrealistically high.
- Commonly used methodology ignores factors that are highly relevant for post-quantum schemes.

Summary and Conclusion

Summary

To assess the security of a recent masked comparison proposal, we:

- Built a leakage model based on the noisy HW model.
- Derived several attacks working under high noise/masking orders.
- Replaced profiling by vertical/horizontal analysis.
- Verified model and attacks on several devices.

Conclusion

In particular for post-quantum schemes:

- Even if t-probing secure, noise/masking orders necessary to prevent the attack in practice may be unrealistically high.
- Commonly used methodology ignores factors that are highly relevant for post-quantum schemes.

Thank you for your attention!

References (1)

[BDH+21] Shivam Bhasin, Jan-Pieter D'Anvers, Daniel Heinz, Thomas Pöppelmann, and Michiel Van Beirendonck. "Attacking and Defending Masked Polynomial Comparison for Lattice-Based Cryptography". In: *IACR Trans. Cryptogr. Hardw. Embed. Syst.* 2021.3 (2021), pp. 334–359. URL: `https://doi.org/10.46586/tches.v2021.i3.334-359`.

[DBV23] Jan-Pieter D'Anvers, Michiel Van Beirendonck, and Ingrid Verbauwhede. "Revisiting Higher-Order Masked Comparison for Lattice-Based Cryptography: Algorithms and Bit-Sliced Implementations". In: *IEEE Trans. Computers* 72.2 (2023), pp. 321–332. URL: `https://doi.org/10.1109/TC.2022.3197074`.

[DHP+22] Jan-Pieter D'Anvers, Daniel Heinz, Peter Pessl, Michiel Van Beirendonck, and Ingrid Verbauwhede. "Higher-Order Masked Ciphertext Comparison for Lattice-Based Cryptography". In: *IACR Trans. Cryptogr. Hardw. Embed. Syst.* 2022.2 (2022), pp. 115–139. URL: `https://doi.org/10.46586/tches.v2022.i2.115-139`.

[HMS+23] Julius Hermelink, Erik Mårtensson, Simona Samardjiska, Peter Pessl, and Gabi Dreo Rodosek. "Belief Propagation Meets Lattice Reduction: Security Estimates for Error-Tolerant Key Recovery from Decryption Errors". In: *IACR Trans. Cryptogr. Hardw. Embed. Syst.* 2023.4 (2023), pp. 287–317. URL: `https://doi.org/10.46586/tches.v2023.i4.287-317`.

[RRD+23] Gokulnath Rajendran, Prasanna Ravi, Jan-Pieter D'Anvers, Shivam Bhasin, and Anupam Chattopadhyay. "Pushing the Limits of Generic Side-Channel Attacks on LWE-based KEMs - Parallel PC Oracle Attacks on Kyber KEM and Beyond". In: *IACR Trans. Cryptogr. Hardw. Embed. Syst.* 2023.2 (2023), pp. 418–446. URL: `https://doi.org/10.46586/tches.v2023.i2.418-446`.