The Challenge of Side-Channel Countermeasures on Post-Quantum Crypto

Rina Zeitoun - rina.zeitoun@idemia.com
IDEMIA - Crypto & Security Labs

November 29 – December 1, 2022

Outline

Context

IDEMIA: The leader in identity technologies

- Identity (3B ID docs, 5M biometric terminals).
- Payment (800M payment products - 2021).
- Telecoms (900M SIM cards - 2021).

Into the wild

- Our products are deployed in hostile environments.
- Attackers have physical access to the device.
- Must be resistant to side-channel/fault attacks.

☞ Security against side-channel attacks is **mandatory**.

Side-Channel Attacks

Main Powerful Attacks

- Timing Attacks, Simple Power Analysis, Differential/Correlation Power/Electromagnetic Analysis, Template Attacks, Fault Attacks, etc.

Into Specifications of Selected NIST PQC Algorithms

- Resistance to Timing Attacks is always addressed.
- All other attacks are mainly **left for research**.

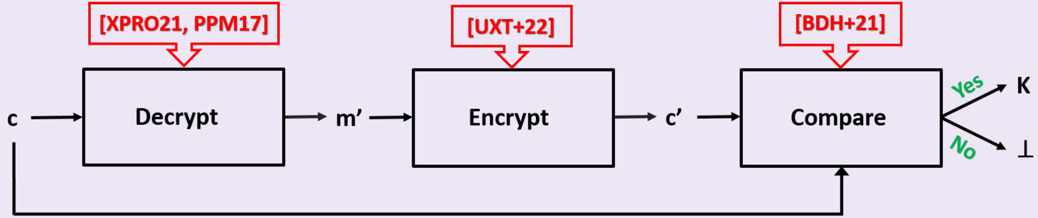
Smartcards: In real life

- Timing attacks are indeed important to consider.
- But **all** other classical side-channel attacks are definitely **real threats**!
- Main powerful attacks should **systematically** be studied in NIST submissions.

Outline

Side-channel Attacks on Lattice-based KEM

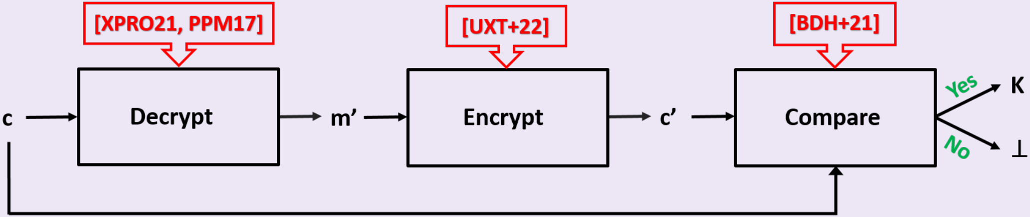
Power/EM Attacks on Decapsulation based on FO Transform



☞ **Whole Decapsulation needs to be masked**

Side-channel Attacks on Lattice-based KEM

Power/EM Attacks on Decapsulation based on FO Transform



☞ **Whole Decapsulation needs to be masked**

Template Attacks on Key Generation

- Template attacks require detailed knowledge of target but can be a real threat!
- Investigated in security certifications (Common Criteria and EMVco).

Outline

Masking Countermeasure

High-Order Masking Countermeasure

- Each sensitive variable x is shared into n variables: $x = x_1 \oplus x_2 \oplus \cdots \oplus x_n$
- Manipulate $x_1, x_2, \ldots, x_n$ independently

Masking Countermeasure

High-Order Masking Countermeasure

- Each sensitive variable x is shared into n variables: $x = x_1 \oplus x_2 \oplus \cdots \oplus x_n$
- Manipulate $x_1, x_2, \ldots, x_n$ independently

Computing with Boolean Masking

Given $x = x_1 \oplus \cdots \oplus x_n$ and $y = y_1 \oplus \cdots \oplus y_n$, how can we compute $x \oplus y$?

- Compute $x_1 \oplus y_1, \ \cdots, \ x_n \oplus y_n$

Masking Countermeasure

High-Order Masking Countermeasure

- Each sensitive variable x is shared into n variables: $x = x_1 \oplus x_2 \oplus \cdots \oplus x_n$
- Manipulate $x_1, x_2, \ldots, x_n$ independently

Computing with Boolean Masking

Given $x = x_1 \oplus \cdots \oplus x_n$ and $y = y_1 \oplus \cdots \oplus y_n$, how can we compute $x \oplus y$?

- Compute $x_1 \oplus y_1, \;\cdots,\; x_n \oplus y_n$

Arithmetic Masking Countermeasure

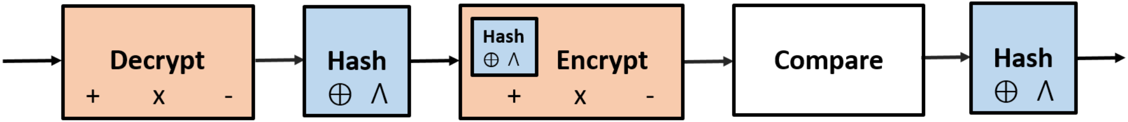
Generate arithmetic sharings s.t. $x = x_1 + \cdots + x_n \mod 2^k$ and $y = y_1 + \cdots + y_n \mod 2^k$

- Compute $x_1 + y_1 \mod 2^k, \cdots, x_n + y_n \mod 2^k$

Arithmetic and Boolean Masking

Masks Conversions

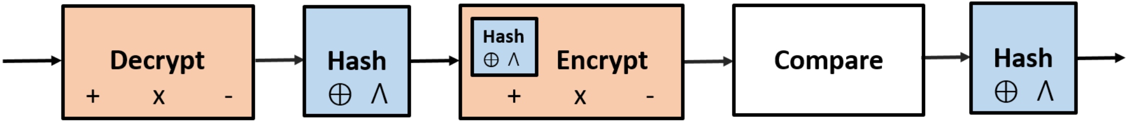
- Need to convert between arithmetic and Boolean masking.
- Efficient classical masks conversions exist ([Gou01],[CGV14],[CGTV15],[BCZ18], etc.)



Arithmetic and Boolean Masking

Masks Conversions

- Need to convert between arithmetic and Boolean masking.
- Efficient classical masks conversions exist ([Gou01],[CGV14],[CGTV15],[BCZ18], etc.)



Difference with previous schemes

- **Symmetric schemes:** k-bit Boolean $\Leftrightarrow$ arithmetic modulo 2^k; usually $k = 32$
- **Post-Quantum schemes:** k-bit Boolean $\Leftrightarrow$ arithmetic modulo q; **arbitrary** k, q

New Problematics with Post-Quantum Crypto

Arbitrary Masks Conversions

- Generic conversions suitable for PQ schemes exist ([BBE+18]: generalization of [CGTV15])
- Downside: Can be too costly in practice

New Problematics with Post-Quantum Crypto

Arbitrary Masks Conversions

- Generic conversions suitable for PQ schemes exist ([BBE+18]: generalization of [CGTV15])
- Downside: Can be too costly in practice

Other Problematics

- Secure polynomials comparison (Kyber, Dilithium)
- Secure computation of compression: $\lceil (2^d/q) \cdot x \rfloor \bmod 2^d$ (Kyber)
- Secure generation of a random in a given interval (Dilithium)
- Secure Euclidean division (NTRU, Dilithium)
- etc.

New Problematics with Post-Quantum Crypto

Arbitrary Masks Conversions

- Generic conversions suitable for PQ schemes exist ([BBE+18]: generalization of [CGTV15])
- Downside: Can be too costly in practice

Other Problematics

- Secure polynomials comparison (Kyber, Dilithium)
- Secure computation of compression: $\lceil (2^d/q) \cdot x \rfloor \bmod 2^d$ (Kyber)
- Secure generation of a random in a given interval (Dilithium)
- Secure Euclidean division (NTRU, Dilithium)
- etc.

☞ **Need specific solution for each problem**

Outline

Kyber Encryption

Encryption Problematic: Securely compute $\lfloor q/2 \rceil \cdot m$ (prime $q = 3329$)

- We have $m = m_1 \oplus \cdots \oplus m_n$ where m_i are 1-bit long.
- Compute $y_1 + \cdots + y_n \bmod q = 1665 \cdot (m_1 \oplus \cdots \oplus m_n)$.

Kyber Encryption

Encryption Problematic: Securely compute $\lfloor q/2 \rceil \cdot m \quad (\text{prime } q = 3329)$

- We have $m = m_1 \oplus \cdots \oplus m_n$ where m_i are 1-bit long.
- Compute $y_1 + \cdots + y_n \bmod q = 1665 \cdot (m_1 \oplus \cdots \oplus m_n)$.

Encryption Solution

Convert 1-bit Boolean sharing $m_1, \cdots, m_n$ into arithmetic modulo q

- Use [BBE+18] with complexity $\mathcal{O}(n^2 \cdot \log \log q)$.
- Use [SPOG19] or [CGMZ21a] with complexity $\mathcal{O}(n^2)$.

Kyber Encryption

Encryption Problematic: Securely compute $\lfloor q/2 \rceil \cdot m \quad$ (prime $q = 3329$)

- We have $m = m_1 \oplus \cdots \oplus m_n$ where m_i are 1-bit long.
- Compute $y_1 + \cdots + y_n \bmod q = 1665 \cdot (m_1 \oplus \cdots \oplus m_n)$.

Encryption Solution

Convert 1-bit Boolean sharing $m_1, \cdots, m_n$ into arithmetic modulo q

- Use [BBE+18] with complexity $\mathcal{O}(n^2 \cdot \log \log q)$.
- Use [SPOG19] or [CGMZ21a] with complexity $\mathcal{O}(n^2)$.

Centered Binomial Distribution (CBD)

- Similar problematic and solution to securely compute $e = HW(x) - HW(y)$ in CBD.

Kyber Decryption

Decryption Problematic: Securely compute $m = \lceil (2/q) \cdot x \rfloor \bmod 2$

- We have $x = x_1 + \cdots + x_n \bmod q$.
- Compute $m_1 \oplus \cdots \oplus m_n = \lceil (2/q) \cdot (x_1 + \cdots + x_n) \rfloor \bmod 2$.

Kyber Decryption

Decryption Problematic: Securely compute $m = \lceil (2/q) \cdot x \rfloor \bmod 2$

- We have $x = x_1 + \cdots + x_n \bmod q$.
- Compute $m_1 \oplus \cdots \oplus m_n = \lceil (2/q) \cdot (x_1 + \cdots + x_n) \rfloor \bmod 2$.

Decryption Solution

- [BGR+21]: Convert A mod $q \hookrightarrow$ B (BBE+18) + 4 secure AND. Complexity $\mathcal{O}(n^2 \cdot \log q)$.
- [CGMZ21a]: Switch A mod $q \hookrightarrow$ A mod 2^k; convert A mod $2^k \hookrightarrow$ B. Complexity $\mathcal{O}(n^2 \cdot \log n)$.

Kyber other problematics: Compress & Compare

$\text{Compress}_{q,d}(x)$ Problematic: Securely compute $\lceil (2^d/q) \cdot x \rfloor \bmod 2^d$

- We have $x = x_1 + \cdots + x_n \bmod q$ and $d = 4, 5, 10$ or 11.
- Compute $y_1 \oplus \cdots \oplus y_n = \lceil (2^d/q) \cdot (x_1 + \cdots + x_n) \rfloor \bmod 2^d$.

Kyber other problematics: Compress & Compare

$\text{Compress}_{q,d}(x)$ Problematic: Securely compute $\lceil (2^d/q) \cdot x \rfloor \bmod 2^d$

- We have $x = x_1 + \cdots + x_n \bmod q$ and $d = 4, 5, 10$ or 11.
- Compute $y_1 \oplus \cdots \oplus y_n = \lceil (2^d/q) \cdot (x_1 + \cdots + x_n) \rfloor \bmod 2^d$.

Comparison Problematic

- We are given ℓ coefficients $x^{(\ell)} = x_1^{(\ell)} \oplus \cdots \oplus x_n^{(\ell)}$ and $y^{(\ell)}$.
- Compare $x_1^{(\ell)} \oplus \cdots \oplus x_n^{(\ell)}$ and $y^{(\ell)}$ without revealing which coefficients fail.

Kyber other problematics: Compress & Compare

$\text{Compress}_{q,d}(x)$ Problematic: Securely compute $\lceil (2^d/q) \cdot x \rfloor \bmod 2^d$

- We have $x = x_1 + \cdots + x_n \bmod q$ and $d = 4, 5, 10$ or 11.
- Compute $y_1 \oplus \cdots \oplus y_n = \lceil (2^d/q) \cdot (x_1 + \cdots + x_n) \rfloor \bmod 2^d$.

Comparison Problematic

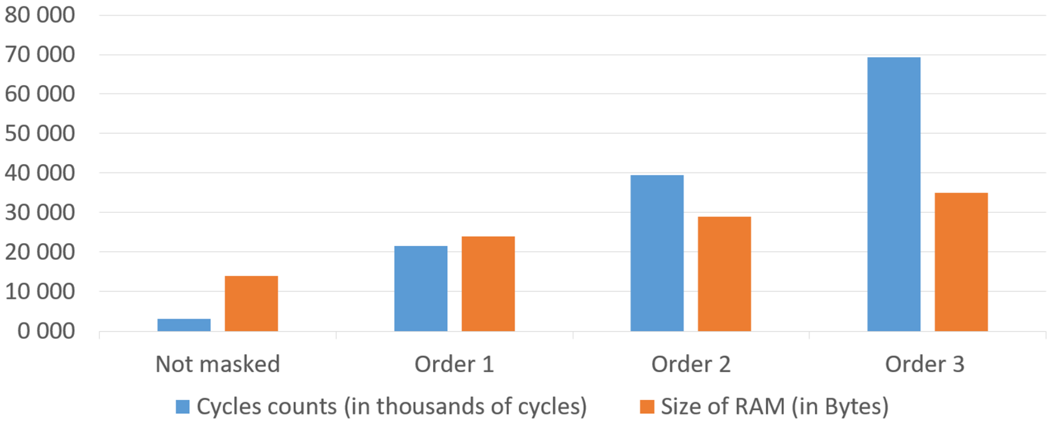
- We are given ℓ coefficients $x^{(\ell)} = x_1^{(\ell)} \oplus \cdots \oplus x_n^{(\ell)}$ and $y^{(\ell)}$.
- Compare $x_1^{(\ell)} \oplus \cdots \oplus x_n^{(\ell)}$ and $y^{(\ell)}$ without revealing which coefficients fail.

Solutions

- [FBR+21]: First order only; compare $H(x_1^{(\ell)} \oplus y^{(\ell)})$ with $H(x_2^{(\ell)})$.
- [BGR+21]: Do not compress: compare uncompressed coefficients with public ones.
- [CGMZ21b]: Hybrid approach with new compression and comparison methods.

Fully masked implementation of Kyber [CGMZ21a/b]

Kyber768 Decapsulation on ARM Cortex-M3 for given security order:



- For security order $t > 3$, required RAM too large for ARM Cortex-M3 target device.

Outline

Conclusion

Smartcards:

- Real need to secure implementations against all SCA.

Standard specifications:

- Resistance against timing attacks studied in standardized PQ algorithms.
- Other Side-Channel Attacks (Power/EM DPA, templates, fault) mainly left for research.

Attacks in practice:

- Many practical Side-Channel Attacks published.

Countermeasures:

- New challenges for PQ crypto countermeasures.
- Not trivial and imply large overhead (can be unacceptable for many products).

Going Forward:

- Encourage designers to study classical SCA at an early stage ("Masking friendly" PQ crypto).

Bibliography

- [Gou01] *A Sound Method for Switching between Boolean and Arithmetic Masking.* Goubin, CHES'01.
- [CGV14] *Secure Conversion Between Boolean and Arithmetic Masking of Any Order.* Coron, Grobschadl, Vadnala, CHES'14.
- [BCZ18] *Improved High-Order Conversion From Boolean to Arithmetic.* Bettale, Coron, Zeitoun, CHES'18.
- [BBE+18] *Masking the GLP Lattice-Based Signature Scheme at Any Order.* Barthe, Belaïd, Espitau, Fouque, Grégoire, Rossi, Tibouchi, EUROCRYPT'18.
- [SPOG19] *Efficiently Masking Binomial Sampling at Arbitrary Orders for Lattice-Based Crypto.* Schneider, Paglialonga, Oder, Güneysu, PKC'19.
- [CGMZ21a] *High-order Table-based Conversion Alg. and Masking Lattice-based Encryption.* Coron, Gérard, Montoya, Zeitoun, CHES'22.
- [CGMZ21b] *High-order Polynomial Comparison and Masking Lattice-based Encryption.* Coron, Gérard, Montoya, Zeitoun, CHES'23.
- [BGR+21] *Masking Kyber: First- and Higher-Order Implementations.* Bos, Gourjon, Renes, Schneider, Vredendaal, CHES'21.
- [FBR+21] *Masked Accelerators and Instruction Set Extensions for PQ.* Fritzmann, Beirendonck, Roy, Karl, Schamberger, Verbauwhede, Sigl, CHES'22.
- [PPM17] *Single-trace side-channel attacks on masked lattice-based encryption.* Primas, Pessl, Mangard, CHES'17.
- [XPRO21] *Magnifying side-channel leakage of lattice-based cryptosystems with chosen ciphertexts: The case study of Kyber.* Xu, Pemberton, Roy, Oswald, IEEE'21.
- [BDH+21] *Attacking and defending masked polynomial comparison for lattice-based cryptography.* Bhasin, D'Anvers, Heinz, Poppelmann, Beirendonck, CHES'21.
- [UXT+22] *Curse of Re-encryption: A Generic Power/EM Analysis on Post-Quantum KEMs.* Ueno, Xagawa, Tanaka, Ito, Takahashi, Homma, CHES'22.

Thank you for your attention!

rina.zeitoun@idemia.com

IDEMIA

Join us on

www.idemia.com