Safe-Error Analysis of Post-Quantum Cryptography Mechanisms

Luk Bettale, *Simon Montoya* and Guénaël Renault

IDEMIA

AGENCE NATIONALE DE LA SÉCURITÉ DES SYSTÈMES D'INFORMATION ANSSI

l'X ÉCOLE POLYTECHNIQUE IP PARIS

Inria

September 1, 2021

Outline

1. Context
2. Safe-error attack
3. Conclusion

Outline

1. Context
2. Safe-error attack
3. Conclusion

Context

- NIST launched in 2016 a call for PQ safe crypto.
  - Key Exchange Mechanism (KEM).
  - Signature.
- Algorithms for a future standardization.
- Here we focus on embedded devices.

Context

- NIST launched in 2016 a call for PQ safe crypto.
  - Key Exchange Mechanism (KEM).
  - Signature.
- Algorithms for a future standardization.
- Here we focus on embedded devices.

Embedded devices

- Less RAM and power consumption.
- Lattice-based schemes seems suitable for embedded devices.

Context

- NIST launched in 2016 a call for PQ safe crypto.
  - Key Exchange Mechanism (KEM).
  - Signature.
- Algorithms for a future standardization.
- Here we focus on embedded devices.

Embedded devices

- Less RAM and power consumption.
- Lattice-based schemes seems suitable for embedded devices.
- In threat of physical attacks:
  - Side-channel.
  - **Fault injection**.
- Fault injection for PQC has not been much investigated.

Context

Safe-error attack

- Safe-error attack (SEA) is a way to use fault injection.
  - Specific fault may or not lead to a faulty output.
  - The faulty or not output gives information.
- Very efficient against constant time implementation.

Context

Safe-error attack

- Safe-error attack (SEA) is a way to use fault injection.
  - Specific fault may or not lead to a faulty output.
  - The faulty or not output gives information.
- Very efficient against constant time implementation.
- In our context the attacker can:
  - Set the fault to a target operation.
  - Skip an instruction or function call.
  - Set a variable to 0.
- Our work focus on NTRU, Saber, Dilithium and Kyber.

Outline

Safe-error attack

Tool for security analysis

- NIST PQC mentioned 5 security categories: 1 to 5.
- However, candidates under/over estimates these categories.

Safe-error attack

Tool for security analysis

- NIST PQC mentioned 5 security categories: 1 to 5.
- However, candidates under/over estimates these categories.
- Then we use the toolkit LWE with side information (L. Ducas, H. Gong and M. Rossi).
- Allow to determine the security lost due to side-channel information.
- The security estimation: *bikz* β.
  - Correspond to the BKZ-β to solve DBDD instance.
  - No conversion between β and bits.

Safe-error attack

High-level attack

- Lattice-based finalists secret distribution $\Rightarrow$ numerous null coeffs.
- Our goal: retrieve the null coefficients.
- Focus on the sign or decrypt algorithms.

Safe-error attack

High-level attack

- Lattice-based finalists secret distribution $\Rightarrow$ numerous null coeffs.
- Our goal: retrieve the null coefficients.
- Focus on the sign or decrypt algorithms.

The attack procedure:

1. Find a function where each secret coefficient is manipulated.
2. Fault the operation.
3. If the output is unchanged: coeff $= 0$.
4. Else: coeff $\neq 0$.

NTRU

- Focus on poly mult.
- Our goal: retrieve 0-coeffs of f.

Algorithm 1 Polynomial Multiplication

Input: a, c, f
Output: a

for $k = 0$ to n **do**
    $a[k] \leftarrow 0$
    for $i = 1$ to n **do**
        $a[k] \leftarrow a[k] + c[k+i] \times f[n-i]$
    end for
    for $i = 0$ to $k+1$ **do**
        $a[k] \leftarrow a[k] + c[k-i] \times f[i]$
    end for
end for
return a

NTRU

- The secret poly f has coefficients in $\{-1, 0, 1\}$ (uniform).
- Fault injection during a poly mult with f.

NTRU

- The secret poly f has coefficients in $\{-1, 0, 1\}$ (uniform).
- Fault injection during a poly mult with f.
- We suppose that the secret coeffs are well distributed: $n/3$ are 0.

	Classical	Attacked
NTRU HPS 1 $n = 509, q = 2048$	Dim = 1018 $\beta = 172.15$	Dim = 680 $\beta = 95.53$
NTRU HPS 2 $n = 677, q = 2048$	Dim = 1354 $\beta = 249.95$	Dim = 904 $\beta = 146.20$
NTRU HPS 3 $n = 821q, q = 4096$	Dim = 1642 $\beta = 308.42$	Dim = 1096 $\beta = 183.35$
NTRU HRSS $n = 701, q = 8192$	Dim = 1402 $\beta = 236.30$	Dim = 936 $\beta = 135.96$

- In average SEA: 42% security loss.

Saber

- The secret poly s has coefficients in $\{-\sigma, \ldots, \sigma\}$ (binomial).
- Fault injection during conversion byte to poly.

Saber

- The secret poly s has coefficients in $\{-\sigma, \ldots, \sigma\}$ (binomial).
- Fault injection during conversion byte to poly.
- We suppose that the secret coeffs are well distributed.

	Classical	Attacked
Light Saber $n, m = 512, \sigma = 5$	Dim = 1025 $\beta = 404.38$	Dim = 900 $\beta = 292.05$
Saber $n, m = 768, \sigma = 4$	Dim = 1537 $\beta = 648.72$	Dim = 1328 $\beta = 462.57$
Fire Saber $n, m = 1024, \sigma = 3$	Dim = 2049 $\beta = 892.21$	Dim = 1729 $\beta = 613.26$

- In average SEA: 30% security loss.

Dilithium

- The secret poly s_1, s_2 have coefficients in $\{-\sigma, \ldots, \sigma\}$ (binomial).
- Fault injection during conversion byte to poly.

Dilithium

- The secret poly s_1, s_2 have coefficients in $\{-\sigma, \dots, \sigma\}$ (binomial).
- Fault injection during conversion byte to poly.
- We suppose that the secret coeffs are well distributed.

	Classical	Attacked
Dilithium 1 $(n, m) = (1024, 1024)$ $\sigma = 2$	Dim = 2049 $\beta = 348.84$	Dim = 1281 $\beta = 192.84$
Dilithium 2 $(n, m) = (1280, 1536)$ $\sigma = 4$	Dim = 2817 $\beta = 499.65$	Dim = 2049 $\beta = 340.06$
Dilithium 3 $(n, m) = (1792, 2048)$ $\sigma = 2$	Dim = 3841 $\beta = 717.52$	Dim = 2401 $\beta = 411.13$

- In average SEA: 40% security loss.

Outline

Conclusion

Countermeasures

- Mask the secret distribution (as Kyber with NTT representation).
- Shuffling.

Conclusion

Countermeasures

- Mask the secret distribution (as Kyber with NTT representation).
- Shuffling.

Conclusion

- Determine the security impact of SEA against lattice-based crypto.
- Decrease significantly the theoretical security.
- Without additional knowledge $\Rightarrow$ difficult to retrieve the entire secret key.
- However, SEA + others side-channel leakage could be devastating.

Questions?

IDEMIA

Join us on

www.idemia.com