Post-Quantum Authenticated Encryption Against Chosen-Ciphertext Side-Channel Attacks

Melissa Azouaoui, Yulia Kuzovkova, Tobias Schneider and Christine van Vredendaal
firstname.lastname@nxp.com

CHES 2022 - September 18-21, 2022 - Leuven, Belgium
PQC KEY ENCAPSULATION MECHANISM

3rd round of the NIST PQC standardization

- Crystals-Kyber
- Classic McEliece
- Saber
- NTRU
- NTRU-prime
- Frodo-KEM
- BIKE
- HQC
- SIKE

Primary KEM to standardize
KEM moving to 4th round
FUJISAKI OKAMOTO TRANSFORM

3rd round of the NIST PQC standardization

Crystals-Kyber
Classic McEliece
NTRU
NTRU-prime
Saber
NTRU
Frodo-KEM
BIKE
HQC

IND-CPA-secure PKE

FO

IND-CCA-secure KEM

FO = Fujisaki-Okamoto Transform
FUJISAKI OKAMOTO TRANSFORM

- CCA KEM Decapsulation -
FUJISAKI OKAMOTO TRANSFORM

- CCA KEM Decapsulation -
FUJISAKI OKAMOTO TRANSFORM

- CCA KEM Decapsulation -
FUJISAKI OKAMOTO TRANSFORM

- CCA KEM Decapsulation -
SIDE-CHANNEL ATTACKS ON THE FO-TRANSFORM

Chosen-Ciphertext SCA

- CCA KEM Decapsulation -
SIDE-CHANNEL ATTACKS ON THE FO-TRANSFORM

When maliciously crafted ciphertexts are decrypted, they depend on a small/enumerable part of the secret key.
SIDE-CHANNEL ATTACKS ON THE FO-TRANSFORM

Decapsulator

Malicious Encapsulator

Dec $= F(\text{Decapsulator})$
SIDE-CHANNEL ATTACKS ON THE FO-TRANSFORM
SIDE-CHANNEL ATTACKS ON THE FO-TRANSFORM

Decapsulator

Malicious Encapsulator

Dec = Decapsulator

Encapsulator

\[ \text{Dec} = \text{Decapsulator} \]

\[ \text{Encapsulator} = \text{Encapsulator} \]

\[ \text{Dec} = \text{F}(\text{Decapsulator}) \]

\[ \text{Encapsulator} = \text{F}(\text{Encapsulator}) \]

\[ \text{Decapsulator} \xrightarrow{\text{Dec}} \text{F}(\text{Decapsulator}) \xrightarrow{\text{FO}} \text{Encapsulator} \]
SIDE-CHANNEL ATTACKS ON THE FO-TRANSFORM

Decapsulator

Malicious Encapsulator

\[ \text{Decapsulator} = \text{Decapsulator} \]

\[ \text{Encapsulator} = \text{Encapsulator} \]

\[ \text{Dec} = \text{Dec} \]

\[ \text{FO} = \text{FO} \]

\[ \text{Malicious} = \text{Malicious} \]
SIDE-CHANNEL ATTACKS ON THE FO-TRANSFORM

Decapsulator

Malicious Encapsulator

Dec = FO = Encapsulator

Dec

FO = F(Dec)
SIDE-CHANNEL ATTACKS ON THE FO-TRANSFORM

- Ravi et al. “Generic Side-channel attacks on CCA-secure lattice-based PKE and KEMs” TCHES 2020
- Xu et al. “Magnifying Side-Channel Leakage of Lattice-Based Cryptosystems with Chosen Ciphertexts: The Case Study of Kyber” IEEE Transactions on Computers, 2021
- Qin et al. “A Systematic Approach and Analysis of Key Mismatch Attacks on Lattice-Based NIST Candidate KEMs” ASIACRYPT 2021
- Ngo et al. “A Side-Channel Attack on a Masked IND-CCA Secure Saber KEM Implementation” TCHES 2021
- Ravi et al. “Will You Cross the Threshold for Me? - Generic Side-Channel Assisted Chosen-Ciphertext Attacks on NTRU-based KEMs” TCHES 2022
- Ueno et al. “Curse of Re-encryption: A Generic Power/EM Analysis on Post-Quantum KEMs” TCHES 2022
- Shen et al. “Find the Bad Apples: An efficient method for perfect key recovery under imperfect SCA oracles – A case study of Kyber” IACR ePrint archive 2022
- Ngo et al. “Side-Channel Attacks on Lattice-Based KEMs Are Not Prevented by Higher-Order Masking” IACR ePring archive 2022
- Rajedran et al. “Pushing the Limits of Generic Side-Channel Attacks on LWE-based KEMs - Parallel PC Oracle Attacks on Kyber KEM and Beyond” IACR ePrint archive 2022
- …
SIDE-CHANNEL ATTACKS ON THE FO-TRANSFORM

- Ravi et al. “Generic Side-channel attacks on CCA-secure lattice-based PKE and KEMs” TCHES 2020
- Xu et al. “Magnifying Side-Channel Leakage of Lattice-Based Cryptosystems with Chosen Ciphertexts: The Case Study of Kyber” IEEE Transactions on Computers, 2021
- Qin et al. “A Systematic Approach and Analysis of Key Mismatch Attacks on Lattice-Based NIST Candidate KEMs” ASIACRYPT 2021
- Ngo et al. “A Side-Channel Attack on a Masked IND-CCA Secure Saber KEM Implementation” TCHES 2021
- Ravi et al. “Will You Cross the Threshold for Me? - Generic Side-Channel Assisted Chosen-Ciphertext Attacks on NTRU-based KEMs” TCHES 2022
- Ueno et al. “Curse of Re-encryption: A Generic Power/EM Analysis on Post-Quantum KEMs” TCHES 2022
- Shen et al. “Find the Bad Apples: An efficient method for perfect key recovery under imperfect SCA oracles – A case study of Kyber” IACR ePrint archive 2022
- Ngo et al. “Side-Channel Attacks on Lattice-Based KEMs Are Not Prevented by Higher-Order Masking” IACR ePring archive 2022
- Rajedran et al. “Pushing the Limits of Generic Side-Channel Attacks on LWE-based KEMs - Parallel PC Oracle Attacks on Kyber KEM and Beyond” IACR ePrint archive 2022
- ...
High order masking is the main countermeasure against SCA

- The leakage of the FO implies an increase of 1 to 2 masking shares to achieve a target security \([ABF+22]\)
- Implies slowdown factors ranging from \(\times 1.2\) to \(\times 3\)

\[ABH+21\] Azouaoui, M., Bronchain, O., Hoffmann, C., Kuzovkova, Y., Schneider, T., Standaert, FX. “Systematic Study of Decryption and Re-encryption Leakage: The Case of Kyber”. COSADE 2022.
A CLOSER LOOK AT THE COST OF DECAPSULATION

Table 4: STM32F4 ARM Cortex-M4 MCU Performance numbers for masked Kyber.CCAKEM.Dec and its subroutines in kCycles.

| Operation                  | Number of shares |
|----------------------------|------------------|
|                            | 2                | 3                | 4                | 5                | 6                | 7                |
| Kyber.CCAKEM.Decaps        | 3178             | 57141            | 97294            | 174220           | 258437           | 350529           |
| Kyber.CPAPKE.Dec           | 200              | 4203             | 7047             | 13542            | 20323            | 27230            |
| Kyber.CPAPKE.Enc           | 2024             | 18879            | 32594            | 53298            | 75692            | 104191           |
| comparison (c = c')        | 693              | 32293            | 54725            | 102922           | 156075           | 210518           |
| $G$                        | 98               | 1639             | 2801             | 4489             | 6456             | 8794             |
| $H$                        | 113              | 113              | 113              | 113              | 113              | 113              |
| $H'$                       | 13               | 13               | 13               | 13               | 13               | 13               |

- Masked decryption is <8% of the cost of masked decapsulation
- Cost of masked decapsulation is dominated by the masked FO
A VERY SIMPLE IDEA

Replace expensive FO by a signature verification of the ciphertext. Signature verification only uses public data and does not require SCA protection. Never decrypt untrusted ciphertexts.
A VERY SIMPLE IDEA

Replace expensive FO by a signature verification of the ciphertext. Signature verification only uses public data and does not require SCA protection. Never decrypt untrusted ciphertexts.

- Based on the Encrypt-then-Sign ($\mathcal{E}_t\mathcal{S}$) paradigm
- CCA security shown in [ADR02]

**Theorem 2.** If $\mathcal{E}$ is IND-CPA-secure, and $\mathcal{S}$ is UF-CMA-secure, then $\mathcal{E}_t\mathcal{S}$ is IND-gCCA2-secure in the Outsider- and UF-CMA-secure in the Insider-security models.

- Post-quantum CCA security shown in [CPPS20]

[ADR02] An, JH., Dodis, Y., Rabin, R. “On the Security of Joint Signature and Encryption”. EUROCRYPT 2002.

[CPPS20] Chatterjee, S., Pandit, T., Puria, SKP., Shah, A. “Signcryption in a Quantum World”. IACR ePrint Arch., 2020.
- CCA FO KEM Decapsulation -
THE $\mathcal{E}tS$ KEM VS. THE FO KEM

- CCA FO KEM Decapsulation -
THE $E_tS$ KEM VS. THE $FO$ KEM

- CPA PKE Decryption -
THE $\mathcal{E}tS$ KEM VS. THE FO KEM

- CCA $\mathcal{E}tS$ KEM Decapsulation -
Outsider vs. Insider security models

**Outsider security**
- Adversary is not a legitimate user of the system.
- Adversary does not have a trusted signature key pair and cannot sign ciphertexts.

**Insider security**
- Adversary can be the sender.
- Adversary can sign ciphertexts and receiver verifies these signatures.

[ADR02] An, JH., Dodis, Y., Rabin, R. “On the Security of Joint Signature and Encryption”. EUROCRYPT 2002.
OUTSIDER VS. INSIDER SECURITY

Outsider vs. Insider security models

Outsider security
- Adversary is not a legitimate user of the system.
- Adversary does not have a trusted signature key pair and cannot sign ciphertexts.

Insider security
- Adversary can be the sender.
- Adversary can sign ciphertexts and receiver verifies these signatures.

[ADR02] An, JH., Dodis, Y., Rabin, R. “On the Security of Joint Signature and Encryption”. EUROCRYPT 2002.
THE $\mathcal{E}_tS$ KEM FOR SECURE ENCRYPTED UPDATE MECHANISM

Step 1: provisioning

Step 2: update ready

Step 3: KEX/KEM

Step 4: encrypt and sign

Step 5: send update

Outsider security

$\mathcal{E}_tS$ KEM
### THE \( \textit{EtS} \) KEM VS. THE FO KEM

| Num. of shares | Kyber.Decaps | \( \textit{EtS} \) Kyber + Dilithium 3 | \( \textit{EtS} \) Kyber + Falcon-1024 |
|---------------|--------------|---------------------------------|---------------------------------|
| 2             | 3178         | 2568 (80.8%)                    | 1316 (41.41%)                  |
| 3             | 57141        | 6571 (11.5%)                    | 5319 (9.3%)                    |
| 4             | 97294        | 9415 (9.7%)                     | 8163 (8.4%)                    |
| 5             | 174220       | 15910 (9.1%)                    | 14658 (8.4%)                   |
| 6             | 258437       | 22691 (8.9%)                    | 21439 (8.3%)                   |
| 7             | 350529       | 29598 (8.4%)                    | 28346 (8.1%)                   |

Kannwischer, MJ., Rijneveld, J., Schwabe, P., Stoffelen, K. "pqm4: Testing and Benchmarking NIST PQC on ARM Cortex-M4". 2019.
### THE $\mathcal{E}tS$ KEM VS. THE FO KEM

| Num. of shares | Kyber-Decaps | $\mathcal{E}tS$ Kyber + Dilithium 3 | $\mathcal{E}tS$ Kyber + Falcon-1024 |
|---------------|--------------|------------------------------------|------------------------------------|
| 2             | 3178         | 2568 (80.8%)                       | 1316 (41.41%)                     |
| 3             | 57141        | 6571 (11.5%)                       | 5319 (9.3%)                       |
| 4             | 97294        | 9415 (9.7%)                        | 8163 (8.4%)                       |
| 5             | 174220       | 15910 (9.1%)                       | 14658 (8.4%)                      |
| 6             | 258437       | 22691 (8.9%)                       | 21439 (8.3%)                      |
| 7             | 350529       | 29598 (8.4%)                       | 28346 (8.1%)                      |

Kannwischer, MJ., Rijneveld, J., Schwabe, P., Stoffelen, K. "pqm4: Testing and Benchmarking NIST PQC on ARM Cortex-M4". 2019.
## The $\ell tS$ KEM vs. The FO KEM

| Num. of shares | Scheme                  |   |   |
|----------------|-------------------------|---|---|
|                | Kyber.Decaps            | $\ell tS$ Kyber + Dilithium 3 | $\ell tS$ Kyber + Falcon-1024 |
| 2              | 3178                    | 2568 (80.8%)                  | 1316 (41.41%)                 |
| 3              | 57141                   | 6571 (11.5%)                  | 5319 (9.3%)                   |
| 4              | 97294                   | 9415 (9.7%)                   | 8163 (8.4%)                   |
| 5              | 174220                  | 15910 (9.1%)                  | 14658 (8.4%)                  |
| 6              | 258437                  | 22691 (8.9%)                  | 21439 (8.3%)                  |
| 7              | 350529                  | 29598 (8.4%)                  | 28346 (8.1%)                  |

Kannwischer, MJ., Rijneveld, J., Schwabe, P., Stoffelen, K. "pqm4: Testing and Benchmarking NIST PQC on ARM Cortex-M4". 2019.
### THE $\mathcal{E}tS$ KEM VS. THE FO KEM

| Num. of shares | Scheme | Kyber.Decaps | $\mathcal{E}tS$ Kyber + Dilithium 3 | $\mathcal{E}tS$ Kyber + Falcon-1024 |
|----------------|--------|--------------|-----------------------------------|-----------------------------------|
| 2              |        | 3178         | 2568 (80.8%)                      | 1316 (41.41%)                     |
| 3              |        | 57141        | 6571 (11.5%)                      | 5319 (9.3%)                       |
| 4              |        | 97294        | 9415 (9.7%)                       | 8163 (8.4%)                       |
| 5              |        | 174220       | 15910 (9.1%)                      | 14658 (8.4%)                      |
| 6              |        | 258437       | 22691 (8.9%)                      | 21439 (8.3%)                      |
| 7              |        | 350529       | 29598 (8.4%)                      | 28346 (8.1%)                      |

**Ciphertext size**

- 1088 bytes
- 4381 bytes
- 2368 bytes

Kannwischer, MJ., Rijneveld, J., Schwabe, P., Stoffelen, K. "pqm4: Testing and Benchmarking NIST PQC on ARM Cortex-M4". 2019.
# THE $\mathcal{E}tS$ KEM VS. THE FO KEM

| Pros                                      | Cons                                               |
|-------------------------------------------|----------------------------------------------------|
| − More efficient ($\times8$ to $\times12$ depending on signature verification speed and number of masking shares) | − Larger ciphertext ($\times2$ to $\times4$ depending on choice of signature scheme) |
| − We remove the FO SCA vector            | − We introduce the signature verification FIA vector |
THE $\mathcal{E}tS$ KEM VS. THE FO KEM

- FO SCA vector $\rightarrow$ Signature verification FIA vector
- SCA protecting FO vs. FIA protecting signature verification
- Ad hoc countermeasure against FIA is re-computation (Recomputing $m$ times protects against $m - 1$ faults)

Impact of protecting the signature verification against fault injection is trivial compared to the cost of masking the FO at high order
# The $\kem$ vs. The FO KEM

| Pros                                                                 | Cons                                                                 |
|----------------------------------------------------------------------|----------------------------------------------------------------------|
| − More efficient ($\times 8$ to $\times 12$ depending on signature verification speed and number of masking shares) | − Larger ciphertext ($\times 2$ to $\times 4$ depending on choice of signature scheme) |
| − We remove the FO SCA vector                                       | − We introduce the signature verification FIA vector                  |
| − Fault protection of signature verification is less challenging and costly than SCA protection of the FO |                                                                      |
CONCLUSION

- The $\mathcal{E}t\mathcal{S}$ KEM is a simple solution to achieve improved leakage resilience for post-quantum KEMs for practical use cases in the outsider security model
- The $\mathcal{E}t\mathcal{S}$ KEM significantly speeds up and reduces the attack surface for post-quantum secure encrypted updates

OUTLOOK

- Find other applications that could benefit from the $\mathcal{E}t\mathcal{S}$ KEM (e.g., IoT edge communication, banking applications)
- Investigate lattice-based PQC schemes for encryption and signature (e.g., SETLA [GM18])

[GM18] Gérard. F., Merckx. K., “SETLA: Signature and Encryption from Lattices”. CANS 2018.
THANK YOU.

QUESTIONS?

CONTACT: PQC@NXP.COM | NXP.COM/PQC