On Generic Side-Channel Assisted Chosen Ciphertext Attacks on NTRU-based Schemes

Prasanna Ravi[1],

Martianus Frederic Ezerman[1],

Shivam Bhasin[1],

Anupam Chattopadhyay[1],

Sujoy Sinha Roy[2]

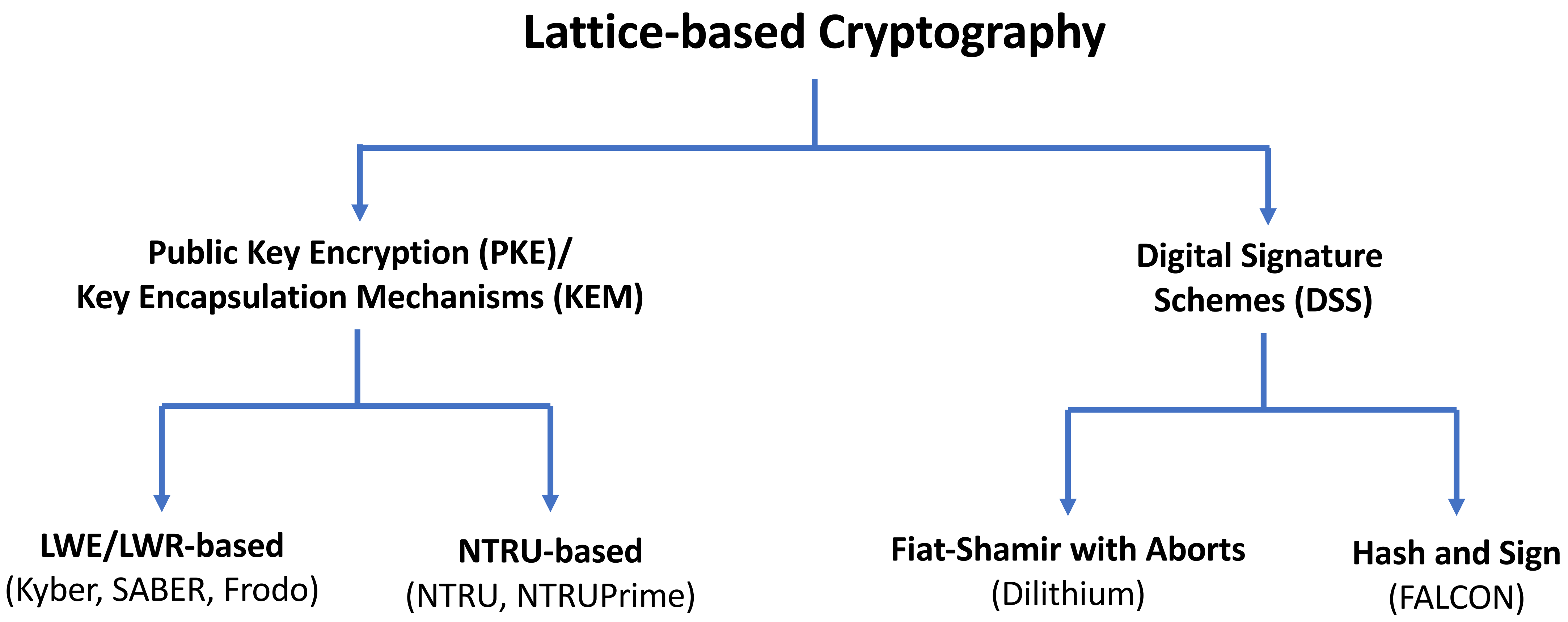
1. Nanyang Technological University, Singapore
2. TU Graz, Austria

Third NIST PQC Conference, 8th June, 2021

Outline

- ❑ **Motivation**
- ❑ **Background: Chosen-Ciphertext Attacks (Classical and SCA Assisted)**
- ❑ **Plaintext Checking Oracle-based SCA on Streamlined NTRU Prime**
- ❑ **Decryption Failure Oracle-based SCA on Streamlined NTRU Prime**
- ❑ **Conclusion and Future Works:**

Classification: Lattice-based NIST PQC Finalists



Lattice-based Cryptography

Public Key Encryption (PKE)/ Key Encapsulation Mechanisms (KEM)

Digital Signature Schemes (DSS)

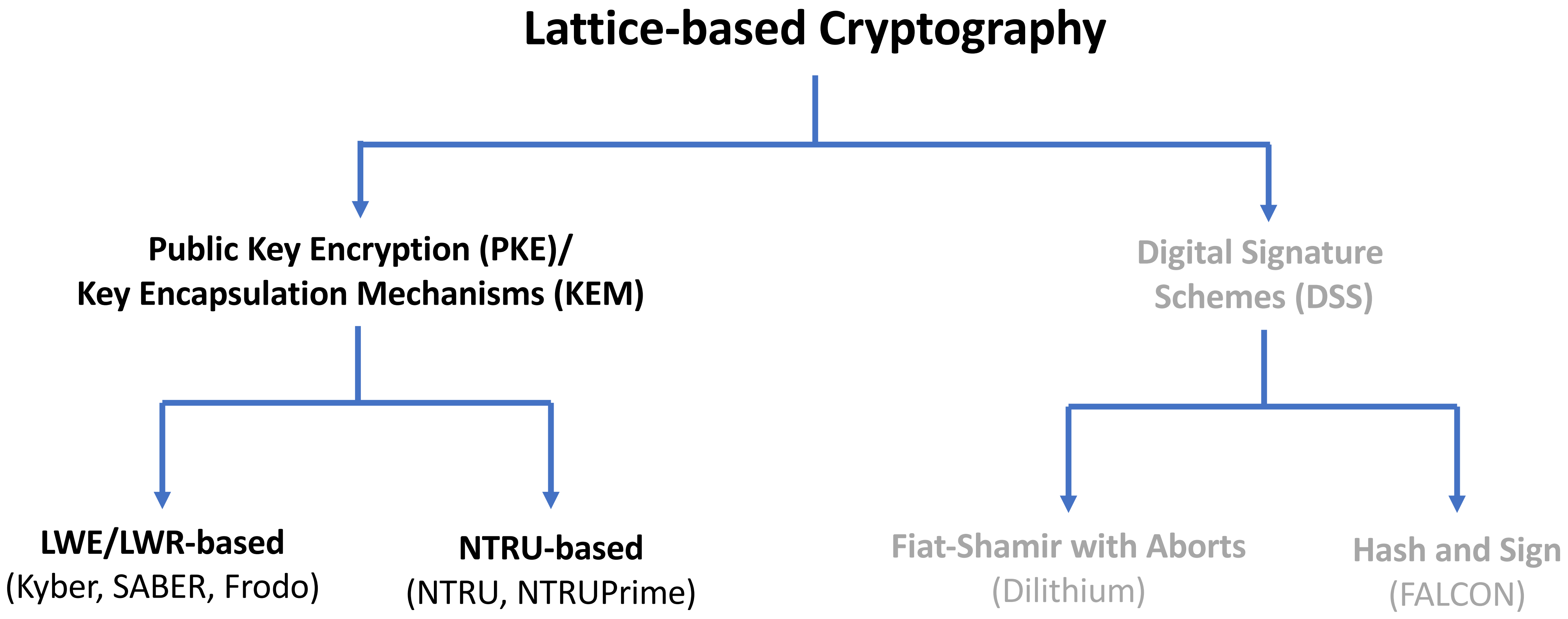
LWE/LWR-based
(Kyber, SABER, Frodo)

NTRU-based
(NTRU, NTRUPrime)

Fiat-Shamir with Aborts
(Dilithium)

Hash and Sign
(FALCON)

Classification: Lattice-based NIST PQC Finalists



Lattice-based Cryptography

Public Key Encryption (PKE)/
Key Encapsulation Mechanisms (KEM)

LWE/LWR-based
(Kyber, SABER, Frodo)

NTRU-based
(NTRU, NTRUPrime)

Digital Signature
Schemes (DSS)

Fiat-Shamir with Aborts
(Dilithium)

Hash and Sign
(FALCON)

Motivation

- ❑ Attention(SCA of LWE/LWR-based schemes) **>>** Attention(SCA of NTRU-based schemes)

- ❑ If side-channel attacker has the ability to query with chosen-inputs, very effective attacks are possible!!!

- ❑ **SCA Assisted Chosen Ciphertext Attacks:**

    - ❑ Practical attacks on LWE/LWR-based schemes [DTV$^+$19, RRC$^+$20, XPR$^+$20, GJN20, BDH$^+$21]

    - ❑ **Advantages**:
        - ❑ Generic (Adaptable to different implementations or target platforms)
        - ❑ Work with low SNR
        - ❑ Low Trace Complexity (Few thousand queries - EM/Power side-channel, Timing Side-channel)

Motivation

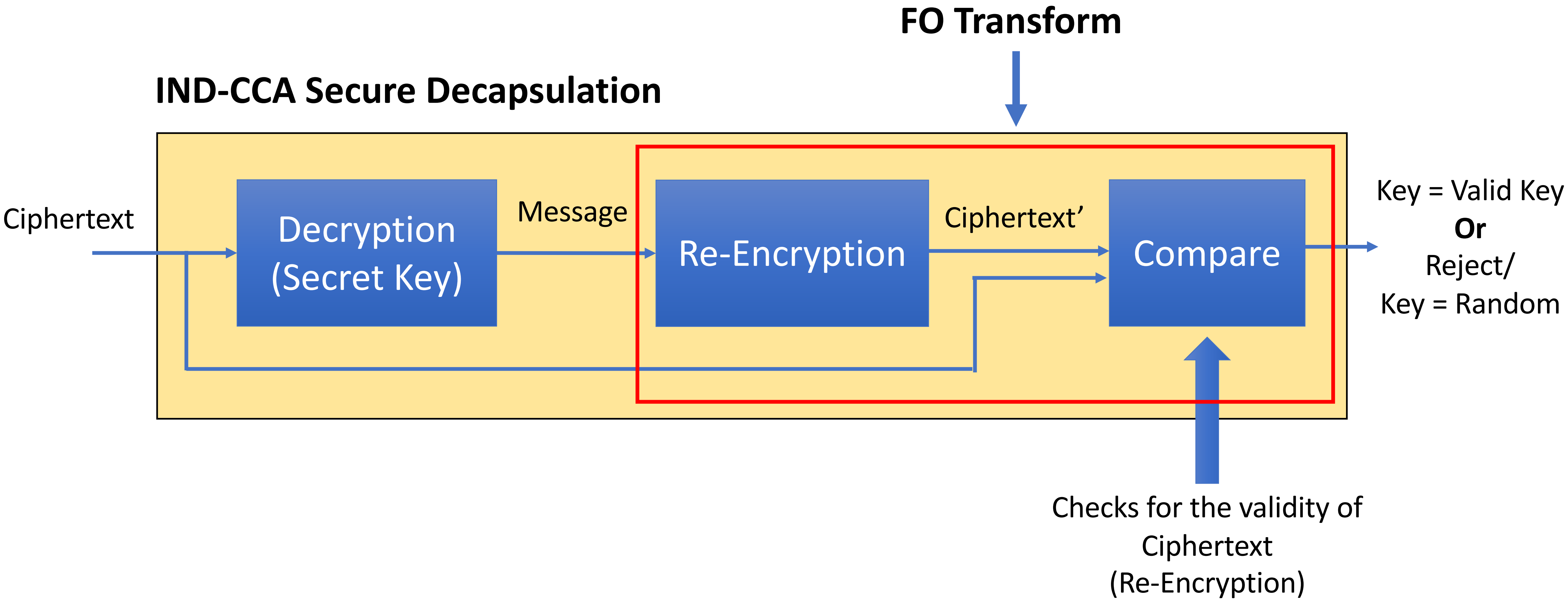
- ❑ **Questions:**
  - ❑ Are similar attacks **possible** on NTRU-based schemes?
  - ❑ If so, are NTRU-based schemes more **easy/difficult** to be attacked compared to LWE/LWR-based schemes?

- ❑ **In this work:**
  - ❑ Generic SCA assisted chosen-ciphertext attacks applicable to NTRU-based schemes
  - ❑ No significant difference in attacker’s effort to break NTRU-based schemes compared to LWE/LWR-based schemes.

Outline

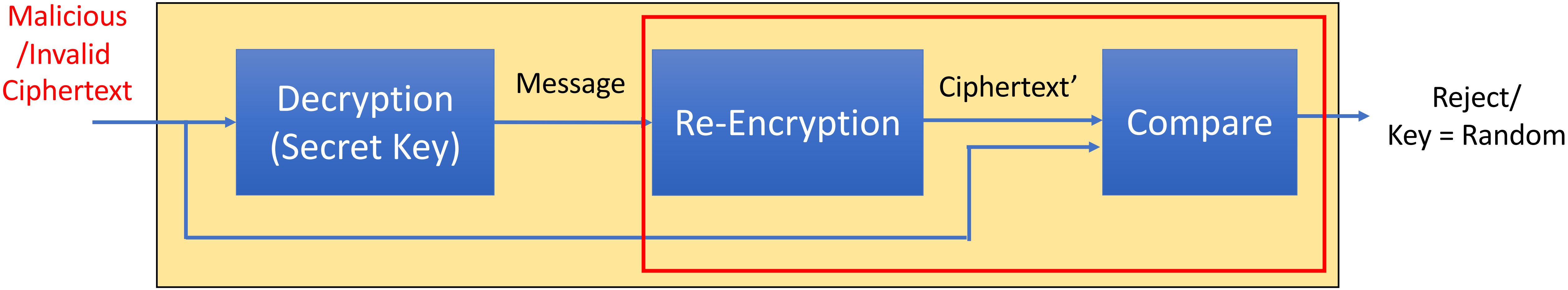
- ❑ **Motivation**
- ❑ **Background: Chosen-Ciphertext Attacks**
- ❑ **Plaintext Checking Oracle-based SCA on Streamlined NTRU Prime**
- ❑ **Decryption Failure Oracle-based SCA on Streamlined NTRU Prime**
- ❑ **Conclusion and Future Works:**

Chosen Ciphertext Attack-secure KEMs



Chosen Ciphertext Attacks: Key Recovery

IND-CCA Secure Decapsulation

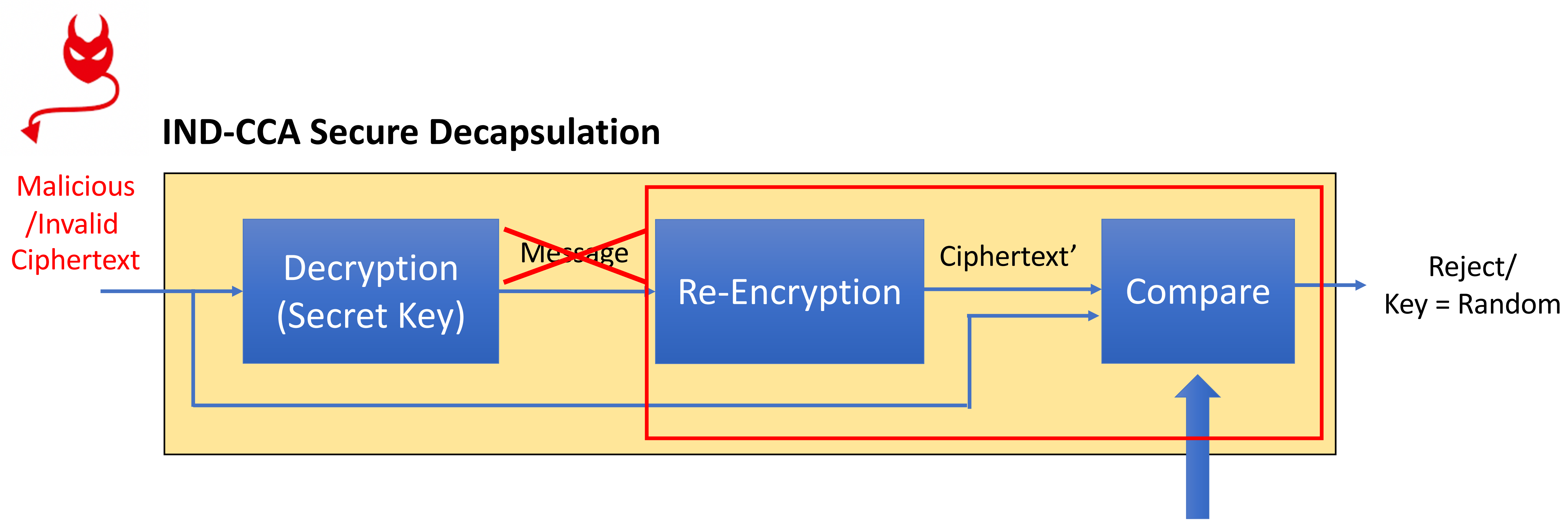


Message → f(secret key)

Ciphertext	Message
Chosen CT1	M2’
Chosen CT2	M3’
Chosen CT3	M0’

→ Full Secret Recovery

Chosen Ciphertext Attacks: Key Recovery

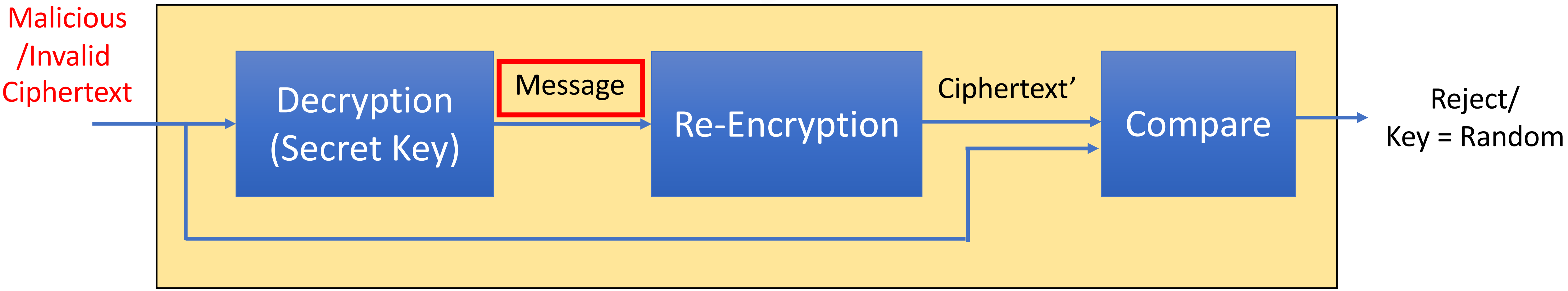


Checks for the validity of Ciphertext (Re-Encryption)

Chosen Ciphertext Attacks: Key Recovery

Side-Channel-based Oracle

IND-CCA Secure Decapsulation



Malicious /Invalid Ciphertext

Decryption (Secret Key)

Message

Re-Encryption

Ciphertext’

Compare

Reject/ Key = Random

SCA Assisted Chosen Ciphertext Attacks

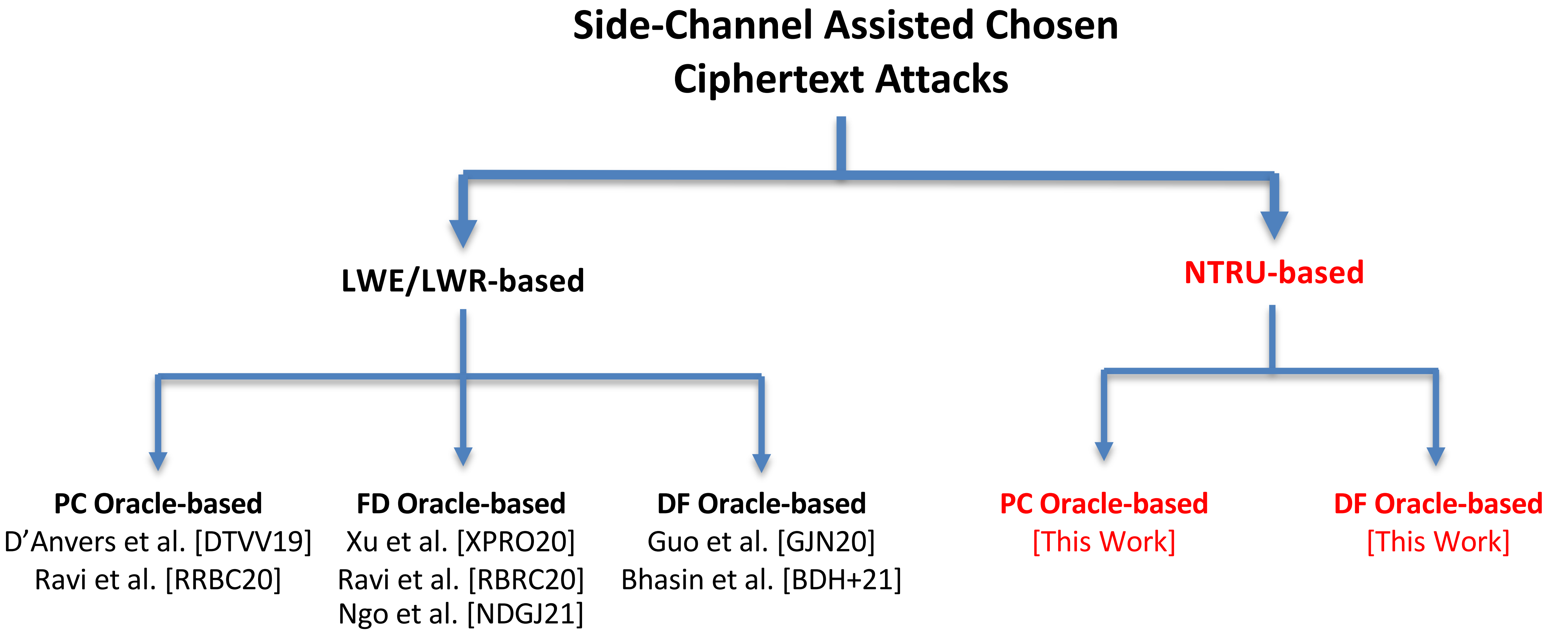
□ Based on available side-channel information (leakage), attacker can instantiate different types of oracles:

Type of Oracle	Oracle Response
Plaintext Checking (**PC**) Oracle [DTV^{+}19, RRC^{+}20]	msg = m_0 or m_1
Decryption Failure (**DF**) Oracle [GJN20, BDH^{+}21]	msg = m_{valid} or $m_{invalid}$
Full Decryption (**FD**) Oracle [XPR^{+}20,RBR^{+}20,NDG^{+}21]	msg = m

□ **Advantages of PC/DF Oracle-based SCA**:

- □ Only rely on binary classification (at worst, very few classes)
- □ Low SNR (Simple SCA)
- □ Agnostic to implementation or target platform or leakage model
- □ Low trace complexity (few thousand traces)

SCA Assisted Chosen Ciphertext Attacks



Side-Channel Assisted Chosen Ciphertext Attacks

LWE/LWR-based

NTRU-based

PC Oracle-based
D'Anvers et al. [DTVV19]
Ravi et al. [RRBC20]

FD Oracle-based
Xu et al. [XPRO20]
Ravi et al. [RBRC20]
Ngo et al. [NDGJ21]

DF Oracle-based
Guo et al. [GJN20]
Bhasin et al. [BDH+21]

PC Oracle-based
[This Work]

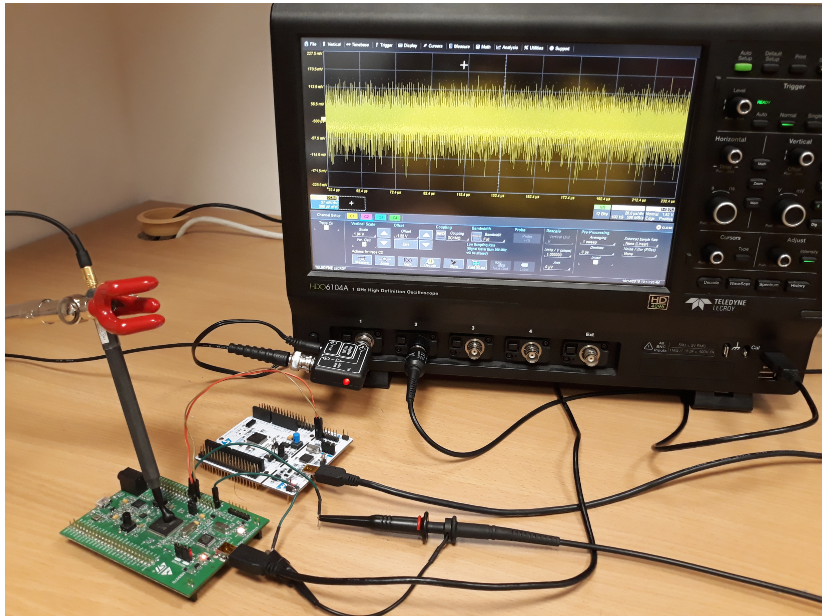
DF Oracle-based
[This Work]

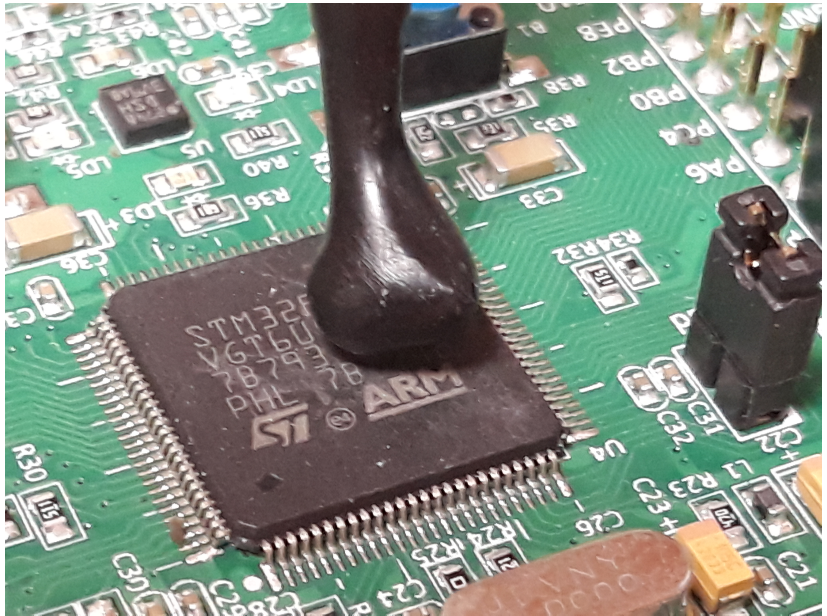
Experimental Setup:

❑ **Target**: Optimized Implementation of Streamlined NTRU Prime (sntrup761) from pqm4 library.

❑ **Platform**: STM32F407VG MCU based on the 32-bit ARM Cortex-M4 processor (24 MHz).

❑ We utilize the near field EM probe and record measurements on the Lecroy 610Zi oscilloscope at a sampling rate of 500 Msam/s.





Outline

- ❑ **Motivation**
- ❑ **Background: Chosen-Ciphertext Attacks (Classical and SCA Assisted)**
- ❑ **Plaintext Checking (PC) Oracle-based SCA on Streamlined NTRU Prime**
- ❑ **Decryption Failure (DF) Oracle-based SCA on Streamlined NTRU Prime**
- ❑ **Conclusion and Future Works:**

Chosen Ciphertexts for Streamlined NTRU Prime

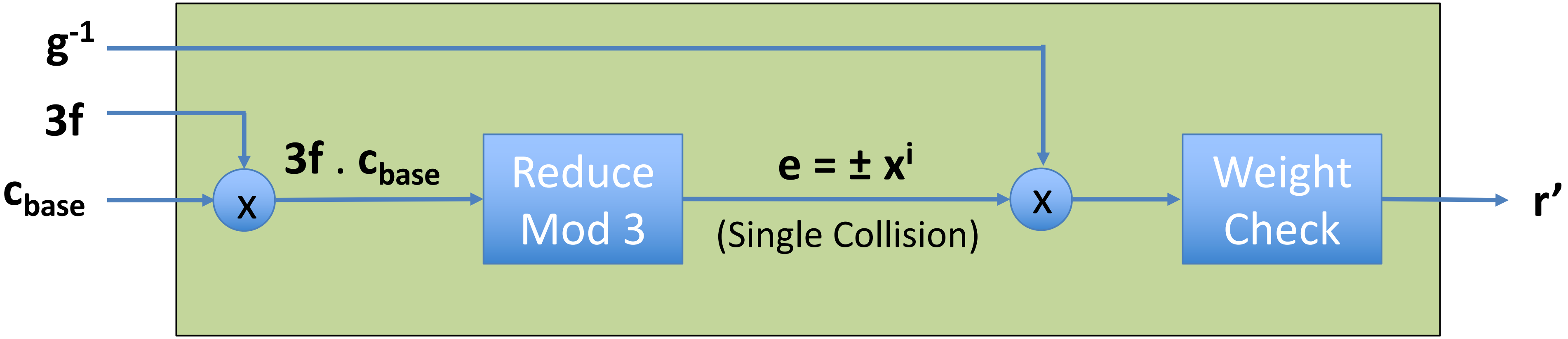
❑ Inspired from [JJ00] in Crypto 2000 on the chosen-ciphertext attack on classical IND-CPA secure NTRU scheme

❑ **Two Step Procedure**:

- ❑ **Step-1**: Identify a base ciphertext (critical info. about secret key)
- ❑ **Step-2**: Use base ciphertext to build attack ciphertexts for key recovery

Step-1: Identifying c_{base}

Public Key (**pk**): (**h**)
Secret Key (**sk**): (**f,g**)
Ciphertext (**ct**): $\mathbf{c_{attack}}$
Message (**r'**): **r'**

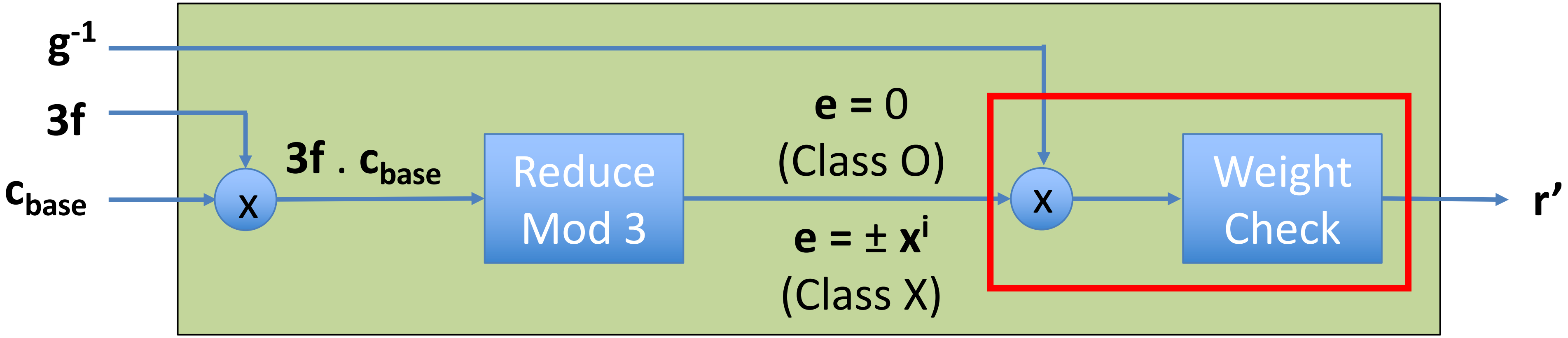


- ❑ Carefully build ciphertexts to identify a base ciphertext $\mathbf{c_{base}}$ whose e has a single non-zero coeff.
- ❑ If **e** = ± x^i, this reveals important information about secret polynomials **f** and **g** (Single Collision Event)

Step-1: Identifying c_{base}

Public Key (**pk**): (**h**)
Secret Key (**sk**): (**f**,**g**)
Ciphertext (**ct**): $\mathbf{c_{attack}}$
Message (**r'**): **r'**

Side-Channel based PC oracle



- ❑ If No Collision: **e** = 0 (Class O)
- ❑ If Single Collision: e = ± x^i (Class X)
- ❑ Ciphertexts are built so as to restrict to these two classes **with very high probability**
- ❑ Side-Channel leakage can be used to differentiate between the two classes (Class O/X)

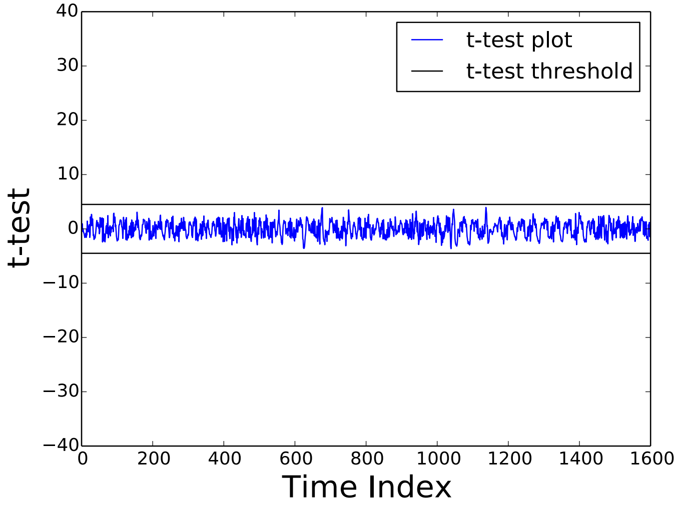
Step-1: Identifying c_{base} using SCA

❑ Two Class Classification: **Welch's t-test** for Collision Detection

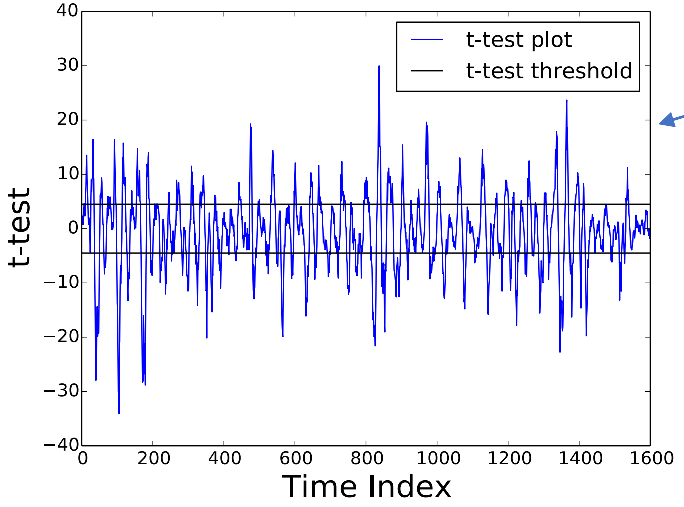
❑ Decapsulate c = 0 (**e** = 0) : T_o ("n" executions)

❑ Decapsulate chosen ciphertext c' : T_X ("n" executions)

❑ Compute the Welch's t-test between T_o and T_X



No Collision



Single Collision

Select features above threshold as PoI

Use PoI to construct template for both classes

RT_O – Class **O**

RT_X – Class **X**

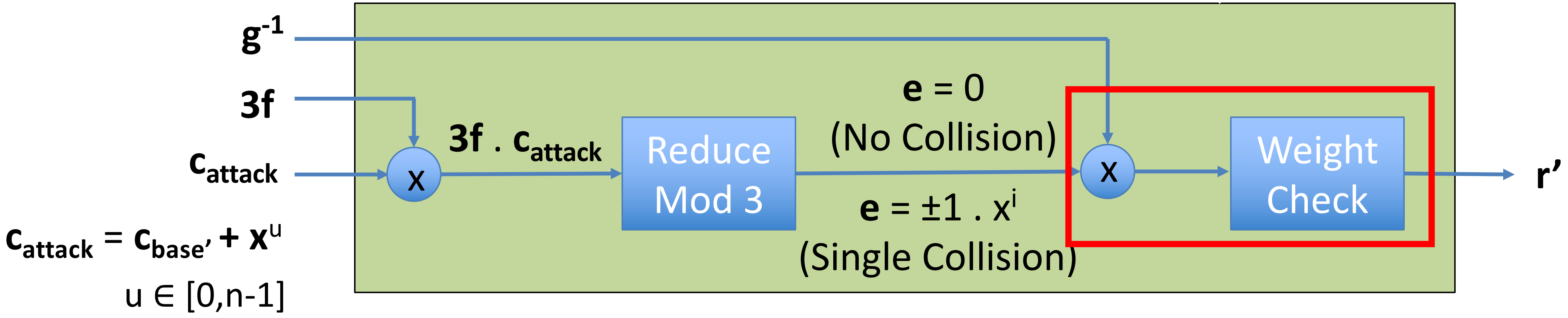
Step-2: Build c_{attack} using c_{base} for Key Recovery

Decrypt(**sk**, **ct**) = **m**

Secret Key (**sk**): (**f**,**g**)
Ciphertext (**ct**): $\mathbf{c_{attack}}$
Message (**r'**): **r'**

Side-Channel based PC oracle

Class O/X



- ❑ Attack ciphertexts $\mathbf{c_{attack}}$ built from the base ciphertext $\mathbf{c_{base}}$
- ❑ Value of **e** (Class O/X) depends upon a targeted portion of the secret key
- ❑ Side-Channel templates used to classify a given attack ciphertext as Class O/Class X
- ❑ This information (O/X) can be used as a binary distinguisher to recover single secret coefficients

Experimental Results (PC Oracle-based SCA)

- ❑ **Target Implementation**: sntrup761 (n = 761)
- ❑ Identifying $\mathbf{c}_{\mathbf{base}}$ ≅ 61 attempts (n = 10 traces each) ≅ 610 traces
- ❑ Recovering each secret coeff. takes 4 queries (761 x 4 ≅ 3.04k traces)
- ❑ **Avg. traces for full secret key recovery**: **4.5k traces** (considering attacking re-tries)
- ❑ **Success Rate**: 100%
- ❑ PC Oracle-based SCA on LWE/LWR-based schemes [RRC$^+$20]: 2k – 5k traces

Limitations of PC Oracle-based SCA

Decrypt(**sk**, **ct**) = **m**

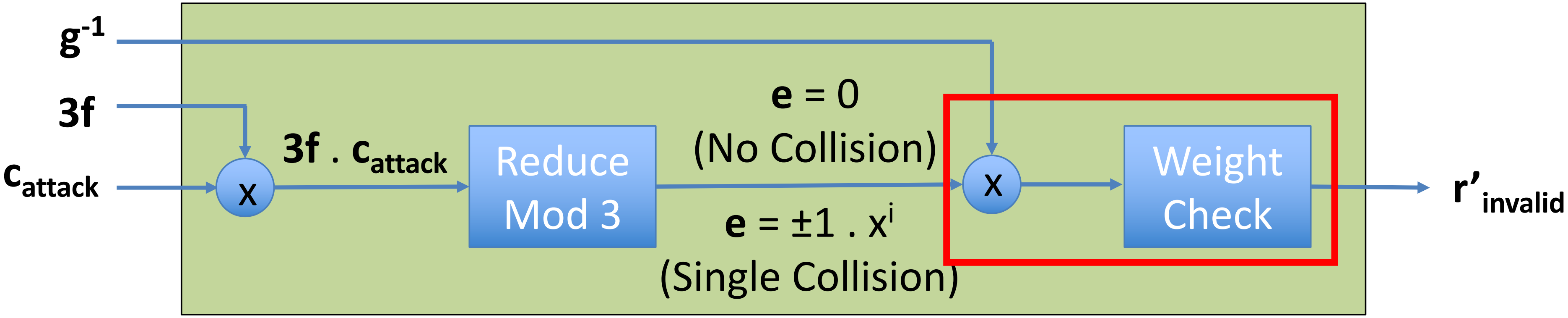
Secret Key (**sk**): (**f**,**g**)

Ciphertext (**ct**): $\mathbf{c_{attack}}$

Message (**r'**): **r'**

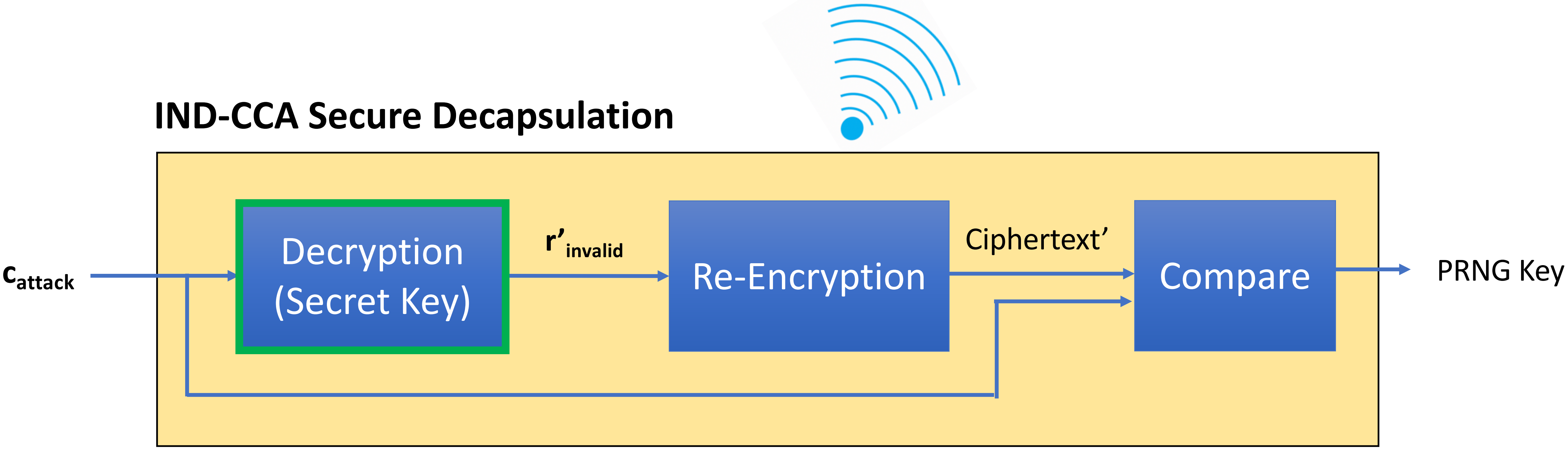
Side-Channel based PC oracle

Class O/X



- ❑ Attack ciphertexts used for the PC oracle-based SCA always return an $r'_{invalid}$ message (Weight Check Failure)
- ❑ The secret dependent information about **e** does not propagate beyond the decryption procedure

Limitations of PC Oracle-based SCA



- ❑ Attack ciphertexts used for the PC oracle-based SCA always return an $r'_{invalid}$ message (Weight Check Failure)
- ❑ The secret dependent information about **e** does not propagate beyond the decryption procedure
- ❑ Countermeasure: **Masking** the decryption procedure
- ❑ Can we widen the scope of the attack (target side-channel leakage from re-encryption procedure) ??

Outline

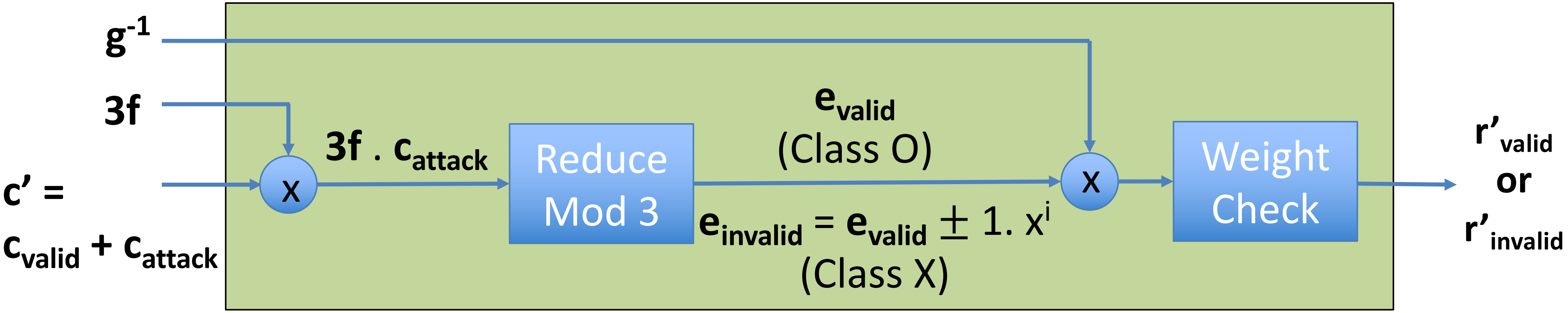
- Motivation
- Background: Chosen-Ciphertext Attacks (Classical and SCA Assisted)
- Plaintext Checking Oracle-based SCA on Streamlined NTRU Prime
- **Decryption Failure (DF) Oracle-based SCA on Streamlined NTRU Prime**
- Conclusion and Future Works:

DF Oracle-based SCA (Streamlined NTRUPrime)

- ❑ **Intuition:** We perturb valid ciphertexts $\mathbf{c_{valid}}$ with the attack ciphertexts $\mathbf{c_{attack}}$ (PC Oracle-based SCA)

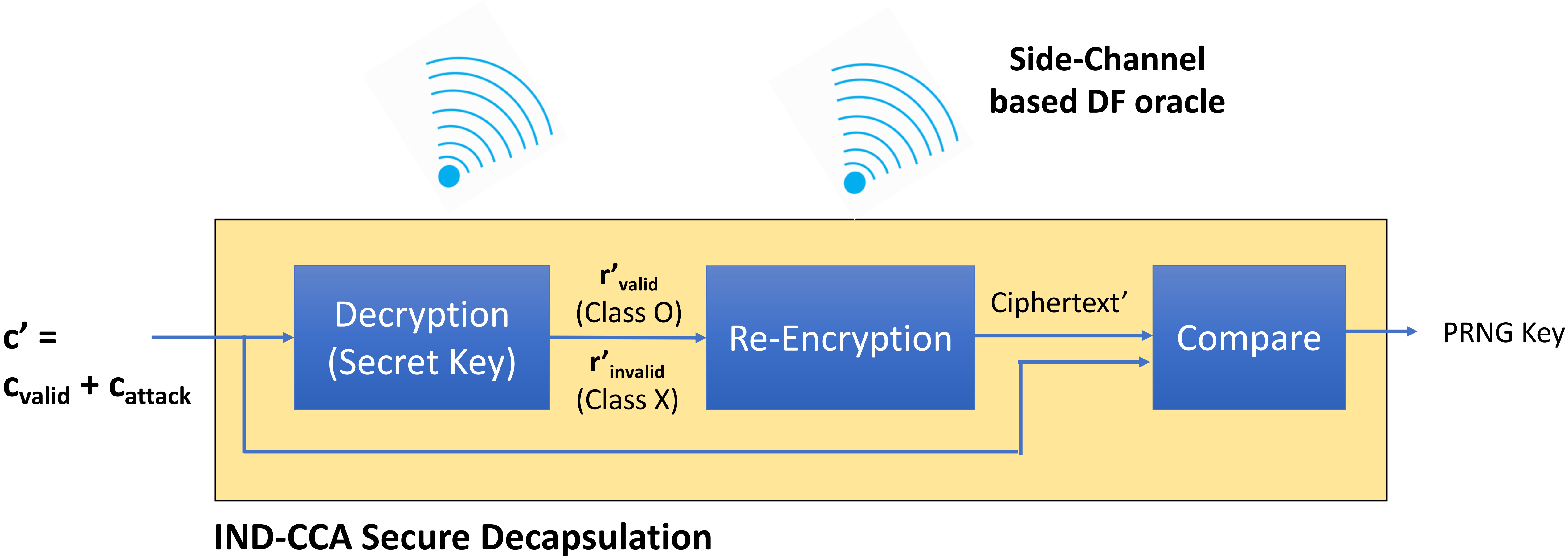
Decrypt(**sk**, **ct**) = **m**

Secret Key (**sk**): (**f**,**g**)
Ciphertext (**ct**): $\mathbf{c_{attack}}$
Message (**r'**): **r'**



DF Oracle-based SCA (Streamlined NTRUPrime)

❑ **Intuition:** We perturb valid ciphertexts $\mathbf{c_{valid}}$ with the attack ciphertexts $\mathbf{c_{attack}}$ (PC Oracle-based SCA)



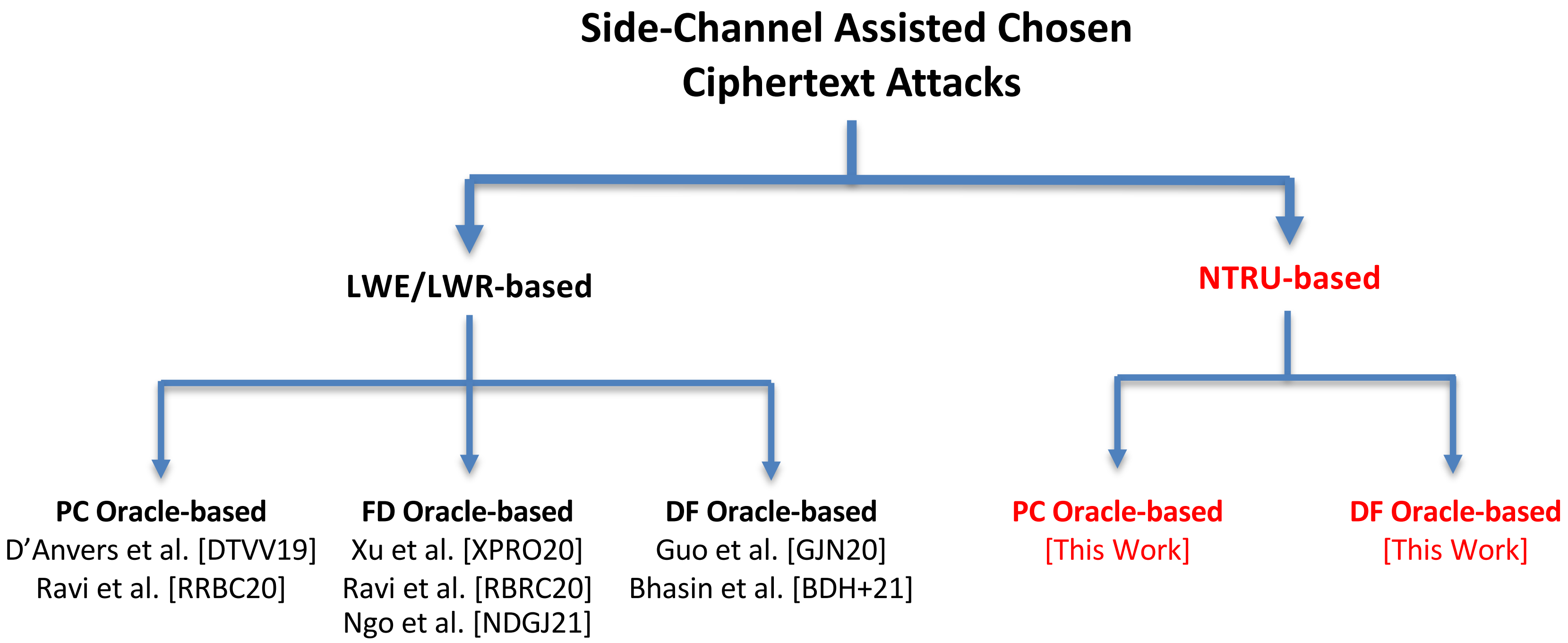
Experimental Results (DF Oracle-based SCA)

- ❑ **Target Implementation**: sntrup761 (n = 761)
- ❑ Identifying $\mathbf{c}_{\mathbf{base}}$ ≅ 425 attempts (n = 10 traces each) ≅ 4.25k traces
- ❑ Recovering each secret coeff. takes 4 queries (761 x 4 = 3.04k traces)
- ❑ **Avg. traces for full secret key recovery**: **8.1k traces** (considering attacking re-tries)
- ❑ **Success Rate**: 100%
- ❑ DF Oracle-based SCA on LWE/LWR-based schemes:
  - ❑ Guo et al. [GJN+20]: 2^{30} (Frodo - Timing side-channel)
  - ❑ Bhasin et al. [BDH+21]: 2^{17} + offline key-search (SCA Protected Kyber - EM side-channel)
- ❑ **COUNTERMEASURE:** Concrete Masking of full decapsulation procedure

Outline

- ❑ **Motivation**
- ❑ **Background: Chosen-Ciphertext Attacks (Classical and SCA Assisted)**
- ❑ **Plaintext Checking Oracle-based SCA on Streamlined NTRU Prime**
- ❑ **Decryption Failure Oracle-based SCA on Streamlined NTRU Prime**
- ❑ **Conclusion**

Conclusion



Side-Channel Assisted Chosen Ciphertext Attacks

LWE/LWR-based

NTRU-based

PC Oracle-based
D'Anvers et al. [DTVV19]
Ravi et al. [RRBC20]

FD Oracle-based
Xu et al. [XPRO20]
Ravi et al. [RBRC20]
Ngo et al. [NDGJ21]

DF Oracle-based
Guo et al. [GJN20]
Bhasin et al. [BDH+21]

PC Oracle-based
[This Work]

DF Oracle-based
[This Work]

- ❑ Plaintext Checking (PC) Oracle-based SCA on Streamlined NTRU Prime (sntrup761):
  - ❑ 4.5k traces (100% success rate)

- ❑ Decryption Failure Oracle-based SCA on Streamlined NTRU Prime (sntrup761):
  - ❑ 8.1k traces (100% success rate)

- ❑ Our attacks reiterate the need for strong masking countermeasures for NTRU-based schemes

Thank you!!!

References

[DTV+19] D'Anvers, Jan-Pieter, Marcel Tiepelt, Frederik Vercauteren, and Ingrid Verbauwhede. "Timing attacks on error correcting codes in post-quantum schemes." In *Proceedings of ACM Workshop on Theory of Implementation Security Workshop*, pp. 2-9. 2019.

[RRC+20] Ravi, Prasanna, Sujoy Sinha Roy, Anupam Chattopadhyay, and Shivam Bhasin. "Generic Side-channel attacks on CCA-secure lattice-based PKE and KEMs." *IACR Transactions on Cryptographic Hardware and Embedded Systems* (2020): 307-335.

[XPR+20] Xu, Zhuang, Owen Pemberton, Sujoy Sinha Roy, and David Oswald. *Magnifying Side-Channel Leakage of Lattice-Based Cryptosystems with Chosen Ciphertexts: The Case Study of Kyber*. Cryptology ePrint Archive, Report 2020/912, 2020. https://eprint.iacr.org/2020/912, 2020.

[RBR+20] Ravi, Prasanna, Shivam Bhasin, Sujoy Sinha Roy, Anupam Chattopadhyay. "On Exploiting Message Leakage in (few) NIST PQC Candidates for Practical Message Recovery and Key Recovery Attacks." Cryptology ePrint Archive, Report 2020/1559, 2020. https://eprint.iacr.org/2020/1559, 2020.

References

[NDG$^+$21] Ngo, Kalle, Elena Dubrova, Qian Guo, and Thomas Johansson. "A Side-Channel Attack on a Masked IND-CCA Secure Saber KEM.” Cryptology ePrint Archive, Report 2021/079, 2021. https://eprint.iacr.org/2021/079, 2021.

[GJN20] Qian Guo, Thomas Johansson, Alexander Nilsson. "A key-recovery timing attack on post-quantum primitives using the Fujisaki-Okamoto transformation and its application on FrodoKEM." https://eprint.iacr.org/2020/743 In IACR-CRYPTO 2020.

[BDH$^+$21] Bhasin, Shivam, Jan-Pieter D’Anvers, Daniel Heinz, Thomas Pöppelmann, and Michiel Van Beirendonck. "Attacking and Defending Masked Polynomial Comparison for Lattice-Based Cryptography.” In IACR-TCHES 2021.

[JJ00] Jaulmes, Éliane, and Antoine Joux. "A chosen-ciphertext attack against NTRU." In Annual International Cryptology Conference, pp. 20-35. Springer, Berlin, Heidelberg, 2000.

[KRSS19] Kannwischer, Matthias J., Joost Rijneveld, Peter Schwabe, and Ko Stoffelen. "pqm4: Testing and Benchmarking NIST PQC on ARM Cortex-M4." (2019).