Coefficient Classification Template Attack on the NTT Accelerator for ML-KEM

Munkhbaatar Chinbat [1,2], Liji Wu [1,2] ✉, and Xiangmin Zhang [1,2]
[1] School of Integrated Circuits, Tsinghua University, Beijing, China
[2] Beijing National Research Center for Information Science and Technology, Beijing, China
✉ e-mail: lijiwu@tsinghua.edu.cn

Motivation - Breaking NTT accelerator

- The Module Lattice-based Key Encapsulation Mechanism (ML-KEM), formerly known as CRYSTALS-Kyber, is a lattice-based algorithm selected for post-quantum cryptography standardization [1]. Its performance relies heavily on the Number Theoretic Transform (NTT) for efficient polynomial multiplication. According to Algorithm 1, the secret key is encoded from the vector s, and as shown in Table 1, ML-KEM-768 and ML-KEM-1024 use the noise parameter eta1 = 2, meaning the coefficients of s come from a small set of values.
- In this work, we demonstrate that by targeting the NTT operation—specifically the transformation of s to s_hat—side-channel analysis can be used to recover these coefficients. Since the entire secret key is derived from s, accurately classifying its values during the NTT enables full reconstruction of the secret key. To support this attack, we adopt and modify a lightweight NTT hardware design to expose critical leakage points for trace collection and profiling.

Table 1: Parameter Sets for ML-KEM

Algorithm	n	q	k	η_1	η_2	d_u	d_v
ML-KEM-512	256	3329	2	3	2	10	4
ML-KEM-768	256	3329	3	2	2	10	4
ML-KEM-1024	256	3329	4	2	2	11	5

Algorithm 1 Secret Key Generation for ML-KEM

Output: Secret key $sk \in B^{12 \cdot k \cdot n/8}$

1: $d \leftarrow B^{32}$
2: $(\rho, \sigma) \leftarrow G(d)$
3: $N \leftarrow 0$
4: **for** $i = 0$ **to** $k - 1$ **do** ▷ Sample $s \in R_q^k$ from B_{η_1}
5: $s[i] \leftarrow \mathrm{CBD}_{\eta_1}(\mathrm{PRF}(\sigma, N))$
6: $N \leftarrow N + 1$
7: **end for**
8: $\hat{s} \leftarrow \mathrm{NTT}(s)$
9: $sk \leftarrow \mathrm{Encode}_{12}(\hat{s} \mod q)$ ▷ $sk \leftarrow s$
10: **return** sk

Idea for Breaking NTT accelerator

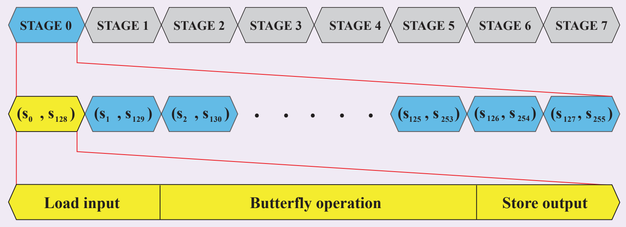

Fig. 1: Overview of general NTT operation

In ML-KEM hardware implementations, the secret vector s contains small coefficients sampled from a centered distribution. For example, when eta1 = 2, the possible values are [0, 1, 2, -1, -2]. Since hardware circuits operate over a finite field defined by the modulus q = 3329, negative values are represented using their modular equivalents: -1 becomes 3328 and -2 becomes 3327. This mapping ensures all coefficients remain in the range [0, q-1].

The Number Theoretic Transform (NTT) is used to convert s into its transformed version s_hat during key generation. In the first stage of the NTT, computations involve input-dependent coefficient pairs like s[i] and s[i + N/2], which are multiplied and added using precomputed constants. These operations create data-dependent intermediate values that may leak through power consumption. If these leakages are captured and analyzed, they can be used to infer the original values of s. Since the full secret key is derived from s, recovering its coefficients through template attack enables complete key reconstruction.

Experimental Setup

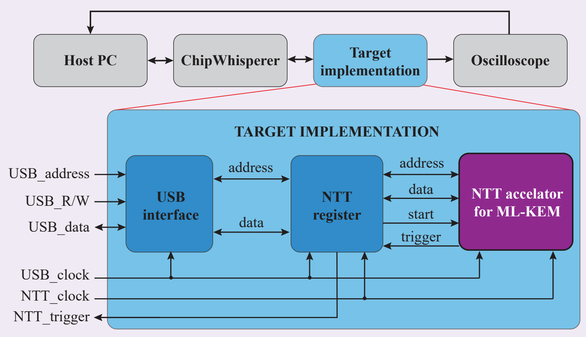

Fig. 2: Modified NTT implementation

- In this work, we used the lightweight hardware design from [2], which employs a single butterfly unit to efficiently perform NTT, INTT, and CWM operations for ML-KEM on FPGA platforms.
- We modified the design to enable side-channel analysis by exposing key intermediate operations. The measurement setup includes the CW305 Artix-7 FPGA board, a Pico oscilloscope, and a ChipWhisperer-Lite for trace collection and data communication.

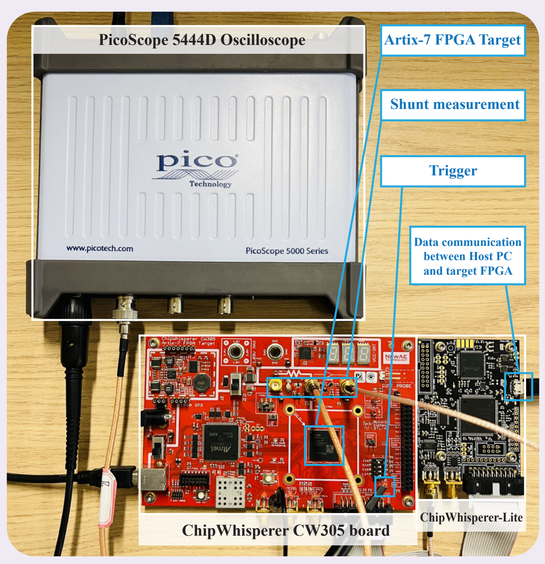

Fig. 3: Measurement setup overview

Coefficient Classification Template Attack

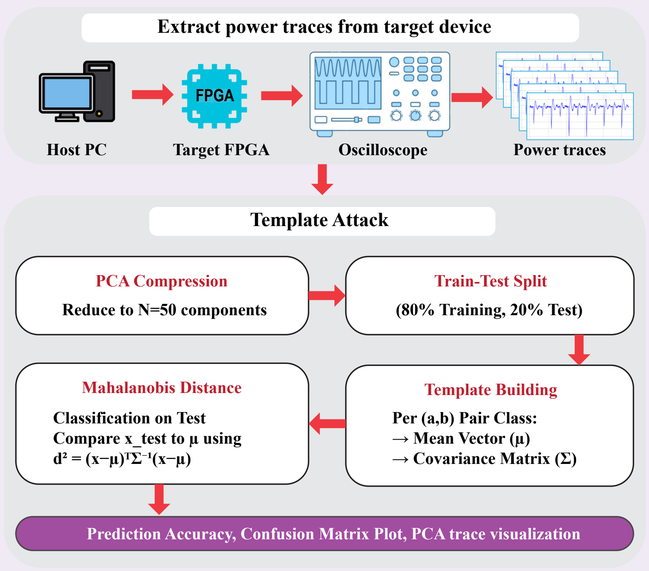

Fig. 4: Template Attack with PCA + Mahalanobis

PCA Compression

High-dimensional power traces are reduced to principal components using PCA. This step filters out noise and redundancy while retaining the most informative features, enabling efficient and accurate classification.

Train-Test Split

The dataset is divided into training and test sets. The training traces are used to build statistical templates, while the test traces are used for evaluation.

Template Building

For each coefficient pair (a, b), we compute a class-specific mean vector and covariance matrix in PCA space. These form the template representing the leakage profile of each class.

Classification with Mahalanobis Distance

Each test trace is compared to all class templates using the Mahalanobis distance, which accounts for both the center and shape of the distributions.

Prediction & Evaluation

Predicted labels are compared against the true ones. A confusion matrix visualizes performance, and the overall classification accuracy is reported.

Coefficient Classification Results

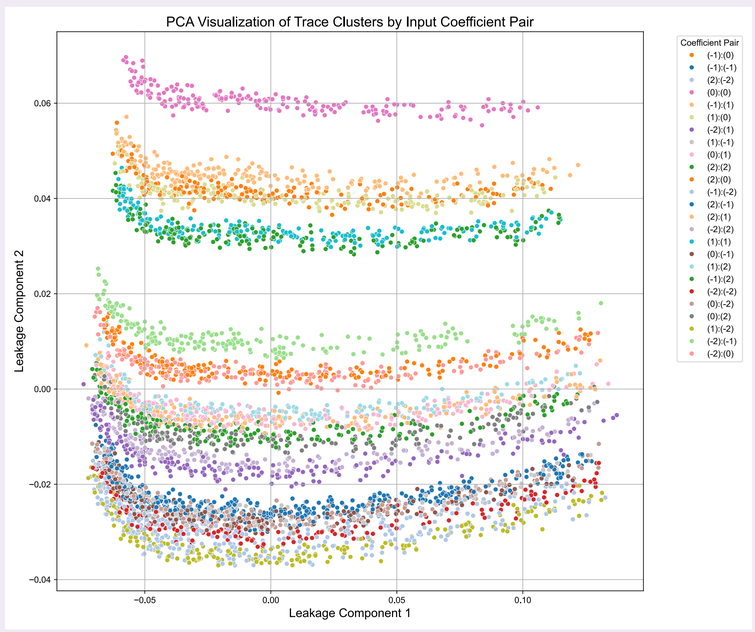

Fig. 5: PCA Visualization

- We performed the template attack using **100,000 power traces** collected during the Forward NTT operation in the secret key generation for **ML-KEM768** and **ML-KEM1024**, with noise parameter **eta = 2.**
- The PCA plot reveals distinct clustering of trace features corresponding to different input coefficient pairs. **Each cluster represents a unique label**, showing that the power traces contain separable leakage patterns related to specific key and input values.
- This clear separation confirms that even a lightweight NTT design leaks sufficient information to enable effective classification.

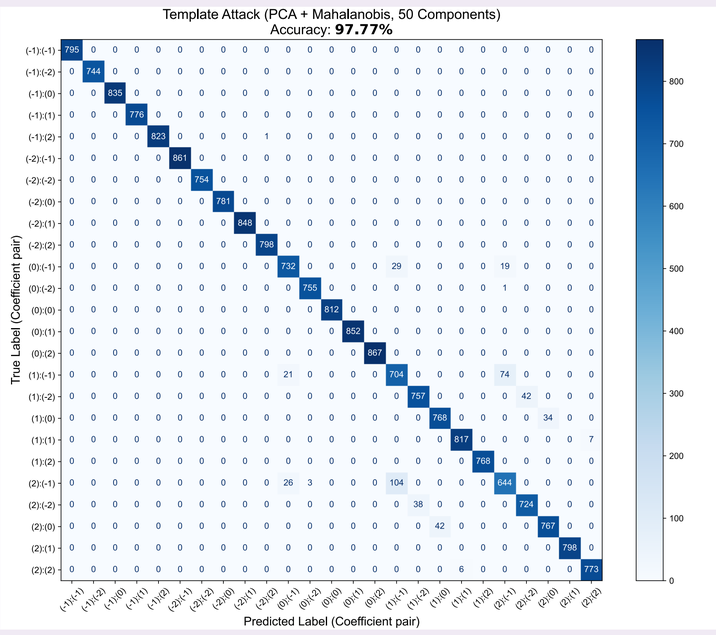

Fig. 6: Confusion Matrix

- Using PCA with 50 components and Mahalanobis distance for classification, we achieved an **accuracy of 97.77%**.
- The confusion matrix shows that most coefficient pairs are correctly classified. However, some misclassifications occur between the labels (2):(-1) and (1):(-1). Specifically, 104 traces labeled as (2):(-1) were predicted as (1):(-1), and 74 traces labeled as (1):(-1) were predicted as (2):(-1). This suggests that the **power consumption patterns for inputs "1" and "2" are highly similar**, leading to confusion during classification.
- These results highlight both the effectiveness and the subtle limitations of template attacks when input-dependent leakage overlaps.

Conclusion

- This work demonstrates a template-based side-channel attack on a lightweight NTT accelerator used in ML-KEM. We modified the hardware design to expose key internal operations and collected 100,000 power traces during the forward NTT. Using PCA and Mahalanobis distance, we achieved 97.77% classification accuracy for coefficient pairs.
- The PCA visualization showed clear clustering of different coefficient combinations, although some confusion occurred between inputs like 1 and 2 due to similar power usage. This highlights that even simple unprotected hardware designs can leak exploitable information.
- These results indicate the possibility of reconstructing the entire secret key, especially for ML-KEM-768 and ML-KEM-1024, which use a limited set of coefficient values. Future work will focus on full key recovery and evaluating countermeasures such as masking and shuffling to enhance side-channel resistance in post-quantum hardware.

References

[1] National Institute of Standards and Technology, "Module-Lattice-Based Key-Encapsulation Mechanism Standard," 8 2024, federal Information Processing Standards Publication 203. [Online].Available:https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.203.pdf
[2] F. Yaman, A. C. Mert, E. Öztürk, and E. Savaş, "A hardware Accelerator for Polynomial Multiplication Operation of CRYSTALS-Kyber PQC Scheme," in 2021 Design, Automation & Test in Europe Conference & Exhibition (DATE). IEEE, 2021, pp. 1020–1025.

IACR Conference on Cryptographic Hardware and Embedded Systems, CHES 2025