Frontiers in Computing and Intelligent Systems
ISSN: 2832-6024 | Vol. 2, No. 3, 2022

A Side-Channel Analysis on the TALE

Yalan Wang *, Zhen Wu

School of Cyberspace Security, Chengdu University of Information Technology, Cheng Du, 610225, China
* **Corresponding author**: Yalan Wang (Email: wang__yl2023@163.com)

Abstract: Lattice-based cryptography, as an active branch of post-quantum cryptography (PQC), has received extensive attention from side-channel analysis (SCA) researchers in recent years. The TALE is a candidate for China's post-quantum project, which aims to standardize cryptographic systems from attacks from quantum and classical computers. Although TALE relies on the theory of quantum-lattice resistance, practical implementations still have the vulnerability of side-channel analysis (SCA). In side-channel analysis (SCA), template attack is considered to be the most effective analysis method, and traditional template analysis are now gradually being replaced by machine learning-based template attack due to low computational efficiency. In this paper, the TALE is analyzed for vulnerability, feature extraction is performed for the leakage information of vulnerable points, and a template attack method based on multilayer perceptron (MLP) is used to attack vulnerable points.

Keywords: Post-quantum cryptography; Lattice-based cryptography; Side-channel analysis; TALE; Multilayer perceptron.

1. Introduction

In 1994, Peter Shor proposed a cracking algorithm, which theoretically proved that this algorithm can solve problems such as discrete logarithms and large integer factorization in polynomial time [1]. This means that once a practical quantum computer appears, the algorithm can be used to crack traditional public-key cryptography such as RSA and ElGamal. In order to prevent the deployment of quantum computers from causing irreversible damage to the current cryptosystem. At present, countries are actively promoting the development of post-quantum cryptography standards, and in 2019 China held a call for calls for post-quantum cryptography competitions, which is a preliminary round for China's post-quantum cryptography standard development.

Among the various types of post-quantum cryptography proposed so far, lattice-based cryptography stands out for various reasons, TALE as a public key cryptosystem based on lattice theory, has the characteristics of anti-quantum attack in theory, but the security of the password needs to be proved in addition to theoretical security, but also needs to consider the security of the specific implementation of the algorithm, that is, physical security. Side-channel attacks can use the time, power consumption, electromagnetic and other information leaked by the cryptographic algorithm when running on the cryptographic chip to analyze and recover the key or encrypted information, which is the main threat to the physical security of cryptography. Traditional side-channel analysis include SPA, CPA, DPA, template attack, and timing attack, and with the rapid development of machine learning techniques, support vector machines, random forests, and neural networks are increasingly used to improve side-channel efficiency.

1.1. Related works

In recent years, there have been a lot of research on the side channel analysis (SCA) of lattice ciphers, and Pessl et al. proposed a single trace attack scheme for NTT in KYBER, and a template attack was carried out on the ARM Cortex M4 microcontroller for the Kyber implementation optimized for constant time, and the key was obtained by constructing 213 templates [2]. Zhuang Xu et al. [3] also targeted Kyber for selective ciphertext attacks, by constructing special ciphertext, information related to the key can be observed from electromagnetic leakage, so as to obtain the key. Ravi et al. [4] targeted Round5, Frodo, New Hope and other lattice cipher algorithms by selecting ciphertext attacks on electromagnetic information leaked in error correction codes and FO transformations. Amiet et al. [5] reported at the 2020 PQC International Conference on simple energy analysis (SPA) and differential energy analysis (DPA) for message coding in NewHope.

1.2. Contributions

In this paper, we propose a side channel analysis method for IND-CCA TALE KEM. We use a multi-layer perceptron-based template attack for message recovery. The problems such as inefficiency and complexity of template construction in traditional template attacks are avoided. We present the following points:

1) We perform a leak point analysis of TALE theoretically to collect the leaked electromagnetic information in the device under test, and also perform a leak analysis of the leak point.

2) We experimentally attacked the leaked information. First, the extraction of feature points was determined; then the optimal parameters of the model were determined.

1.3. Organization

The rest of this paper is organized as follows. In Section 2, we provide an introduction to the TALE algorithm and the template attack. Background. In Section 3, we perform leakage analysis as well as data collection for TALE. In Section 4, we perform the MLP-based template attack on TALE. In Section 5, conclusions are drawn and future research directions are proposed.

2. Preliminaries

2.1. TALE

TALE is a lattice based public key cryptographic, the algorithm is based on the learning with error problem on rings (RLWE) [6], which is an acceleration of the earlier learning with error (LWE) scheme [7]. RLWE-based cryptosystems

require a smaller key size than that of LWE. In addition to the reduced key size, the computation is also sped up. The samples of RLWE are distributed over the ring, and the commonly used ring is the integer ring $R_q = \mathbb{Z}_q[x]/(x^n + 1)$. There are two mechanisms of the TALE algorithm namely the PKE scheme based on IND-CPA security and the KEM scheme based on IND-CCA2 security, and this paper focuses on the KEM.

The KEM is based on the IND-CPA secure PKE through the classic Fujisaki-Okamoto transform, the KEM is designed to establish a shared secret between the two communicating parties, if an attacker exploits the vulnerability of KEM to successfully obtain the shared temporary session key, can eavesdrop on all communications encrypted with this key. The KEM is shown in algorithm 1, where both Hand G are hash functions.

Algorithm 1 $TALE.CCA.Encaps(pk)$

1: $\boldsymbol{m} \leftarrow U(\{0,1\}^{1024})$
2: $\boldsymbol{m} = H_\gamma(m)$
3: $\boldsymbol{K} \parallel \boldsymbol{d} \leftarrow G_{\gamma+\kappa}(\boldsymbol{m} \parallel H_\kappa(pk))$
4: $\boldsymbol{c} \leftarrow TALE.CPA.Enc(pk, \boldsymbol{m}, \boldsymbol{d})$
5: $\boldsymbol{ss} \leftarrow H_\gamma(\boldsymbol{K} \parallel H_\kappa(\boldsymbol{c}))$
6: $\boldsymbol{return}(\boldsymbol{c}, \boldsymbol{ss})$

Algorithm 2 $TALE.CPA.Enc(pk, m, d)$

1: $\boldsymbol{a} \leftarrow B_n^k\ (seed_a)$
2: $\boldsymbol{y}, \boldsymbol{e_1}, \boldsymbol{e_2} \leftarrow B_n^k(\boldsymbol{d})$
3: $\hat{y} \leftarrow KNTT(\boldsymbol{y})$
4: $\boldsymbol{c_0} \leftarrow \boldsymbol{a} \circ \hat{y} + \boldsymbol{e_1}$
5: $\boldsymbol{c} \leftarrow IKNTT(\boldsymbol{b} \circ \hat{y}) + \hat{e}$
6: $\boldsymbol{c_1} \leftarrow \boldsymbol{c_0} + Encode(\boldsymbol{m} \oplus \boldsymbol{c})$
7: $\boldsymbol{c_2} \leftarrow compress(\boldsymbol{c})$
8: $\boldsymbol{return}\ (\boldsymbol{c1}||\boldsymbol{c2})$

In algorithm2, where KNTT is a fast number theory transformation and IKNTT is its inverse transformation. $Encode$ represents the message encoding and compress represents the compression function.

2.2. Template Attack

In 1996, researchers discovered the side channel leakage problem in cryptographic chips and published papers publicly in conferences and journals. In 1996, Kocher et al. proposed timing attacks [8], in 1997, Boneh et al. proposed fault injection [9], and in 1998, Kocher et al. proposed energy analysis attacks [10]. Energy analysis attacks are mainly classified into simple power analysis (SPA), correlated power analysis (CPA) [11], and differential power analysis (DPA)[12].

In 2002, Chari et.al proposed the template attack, which obtains the secret information hidden in the leakage curve by modeling the statistical properties of the random variables in the leakage curve using discriminant analysis [13]. Template attack has been considered as the most effective side channel analysis method since it was proposed. The steps of the template attack include two stages: template construction and template matching.

1) Template construction

If the length of subkey information is kbits, a 2^k template is constructed for each subkey. The template consists of a mean vector $\overline{\mathrm{T}}$ and a covariance matrix C_α.Let n power traces correspond to m sampling points, i.e., $T_{i1}, T_{i2}, \ldots, T_{im} (1 \le i \le n)$ form a matrix $T_{n\times m}$. Thus, an $\overline{\mathrm{T}}$ distribution consists of tuples of the form

$$\overline{\mathrm{T}} = < \tfrac{1}{n}\textstyle\sum_{i=1}^{n} T_{i1}, \tfrac{1}{n}\sum_{i=1}^{n} T_{i2}, \ldots, \tfrac{1}{n}\sum_{i=1}^{n} T_{im} > \qquad (1)$$

Because the power traces correspond to the same operational procedure, the difference between the sample and the sample mean is done to obtain the corresponding noise vector matrix as

$$\mathrm{N}_{\mathrm{n}\times\mathrm{m}} = \begin{bmatrix} \mathrm{T}_{11} - \overline{\mathrm{T}}_1 & \cdots & \mathrm{T}_{1\mathrm{m}} - \overline{\mathrm{T}}_\mathrm{m} \\ \mathrm{T}_{21} - \overline{\mathrm{T}}_1 & \cdots & \mathrm{T}_{2\mathrm{m}} - \overline{\mathrm{T}}_\mathrm{m} \\ \vdots & \cdots & \vdots \\ \mathrm{T}_{\mathrm{n}1} - \overline{\mathrm{T}}_1 & \cdots & \mathrm{T}_{\mathrm{nm}} - \overline{\mathrm{T}}_\mathrm{m} \end{bmatrix} \qquad (2)$$

Each row in $\mathrm{N}_{\mathrm{n}\times\mathrm{m}}$ forms a noise vector for each power trace, and each column is the change in noise amplitude (random variable) at one time, using $\mathrm{cov}(\mathrm{X},\mathrm{Y})$ to denote the covariance of the random variables X and Y. Thus, an C_α distribution consists of a matrix of the form

$$C_\alpha = \begin{bmatrix} cov(N_1, N_1) & \cdots & cov(N_1, N_m) \\ cov(N_2, N_1) & \ldots & cov(N_2, N_m) \\ \vdots & \cdots & \vdots \\ cov(N_m, N_1) & \cdots & cov(N_m, N_m) \end{bmatrix} \qquad (3)$$

2) Template matching

The plaintext p used in the template construction is encrypted and the corresponding leakage curve T' is collected with the same device and compared with the constructed template. the probability of matching T' with template $< \overline{\mathrm{T}}, \mathrm{C}_\alpha >$ is

$$P(T' | < \overline{\mathrm{T}}, C_\alpha >) = \frac{1}{\sqrt{2\pi^m |C|}} exp$$
$$(-\tfrac{1}{2}(T' - \overline{\mathrm{T}})^T C^{-1}(T' - \overline{\mathrm{T}})) \qquad (4)$$

2.3. Multilayer perceptron-based template attack

The traditional template attack relies on multivariate Gaussian distribution modeling, which is computationally inefficient when dealing with multidimensional data and may generate singular matrices, resulting in problems such as inverse covariance matrices. In order to overcome the inefficiency of traditional template attacks, researchers have proposed machine learning-based side channel analysis, which is essentially the process of building classifiers to match data, similar to the template attack process.

The current models that apply machine learning to template attacks include support vector machines, random forests, and neural networks. Multilayer perceptron (MLP), also called deep neural network (DNN) as a model in neural networks, is also often used for template attack.

3. Vulnerability and leakage analysis

3.1. Analysis of Vulnerability Points

The shared key in Algorithm 1 is directly related to the hash function, the parameter $\boldsymbol{K}$ and the ciphertext $\boldsymbol{c}$. The hash function is a pseudo-random, the parameter and the ciphertext are related to the message m and the public key. Among them, the hash functions all use $shake256$, $\boldsymbol{K}$ and $\boldsymbol{c}$ are related to the message and the public key pk, which is public in the

public key cryptosystem. Therefore, if the message m is recovered successfully, the shared key in the KEM can be obtained. In Algorithm 1 and Algorithm 2 the message m is directly involved in the hash operation as well as the message encoding, the hash operation process is complex and the implementation of side channel analysis is diffcult, while the message encoding process is simple, the whole process does not involve variables, and the attack feasibility is high.

Algorithm 3 $Encode(m, c)$

1: $for\ i = 0 \ldots 128\ do$
2: $\quad for\ j = 0 \ldots 7\ do$
3: $\quad\quad t = m[i] \gg j\ \ \& 0x01$
4 $\quad\quad coeff[i * 8 + j] = (t \oplus (c \,\&\, 0x01)) \& 0x01$
5: $\quad end\ for$
6: $end\ for$
7: $return\ coeff[]$

The message encoding process is shown in Algorithm 3. The encoding is the conversion of the message byte into message bit, with each message bit having only two cases, 0x00 or 0x01. and the difference in Hamming weight between the two cases is 1. To get the correct message classification, a template is created for each message bit, and the probability of being able to trace the corresponding template is calculated, and then the template with the highest probability is selected as the value for each message bit.

3.2. Leakage Analysis

In this paper, STM32F1 board is used to execute Algorithm 1, and an oscilloscope, EM probe and Inspector are used to collect and process the signals. In the signal collection process, to avoid excessive noise affecting the experimental results, the EM probe is used to detect the best leakage position of the chip during the execution of Algorithm 1, and the detection results are shown in Figure 1, where the red area is the maximum leakage point.

Move the probe position to the best leakage point using an oscilloscope and Inspector to collect the signal, where the sampling frequency is 500MHz and the trigger voltage is 2.0V, and the message encoding area after the initial processing curve is shown in Figure 2.

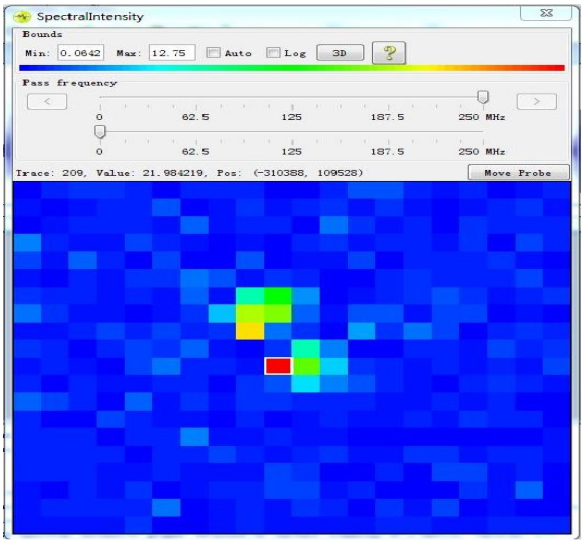

Figure 1. EM scanning situation

To ensure that the acquisition curve can be experimented, a leakage analysis of the curve is required. NICV is a leakage analysis method based on normalized inter-class variance, which can be used to input only public information such as plaintext or ciphertext for leakage detection, and can also be used to test the efficiency of leakage models, the quality of energy traces [14]. Taking the first byte as an example, the leakage analysis of the energy trace using NICV is shown in Figure 3.

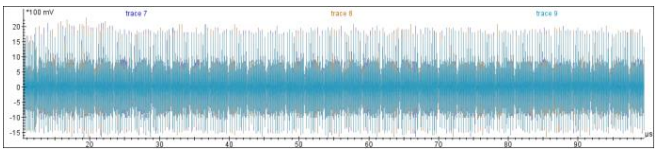

Figure 2. Message encoding area

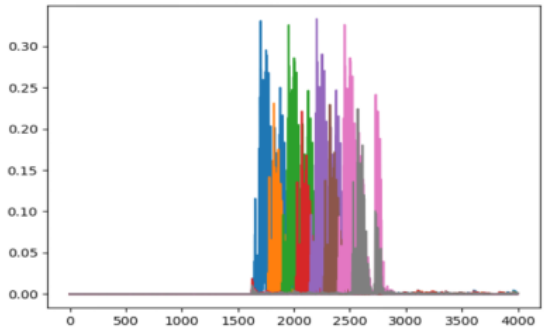

Figure 3. Nicv leakage

From Fig. 3, it can be seen that the leakage interval of the first byte is roughly 1500-2900 in the sample range of 0-4000 with a peak around 0.35, thus proving the leakage of the message encoding process described in 3.1.

4. Experiment

4.1. Attack description

If there are n power traces, the message is

$$M^* = argmax_m \prod_{i=1}^{n} MLP(e_i|comb(m)) \qquad (5)$$

In Equation 5, M^* represents the correct message, e is the energy trace curve, $comb(m)$ is the intermediate value, and $MLP(e_i|comb(m))$ is used as the classifier probability model of the training process. From Equation 5, it is clear that the training effect of the multilayer perceptron plays a decisive role if a better attack effect is to be obtained. To measure the goodness of the model, this paper uses the classification accuracy of the model training as the data classification ability of the model and uses the guess entropy to measure the final effect of the multi-layer perceptron template-based attack.

4.2. Experimental environment and data set

In this paper, we build a multilayer perceptron network model based on Python Tensorflow2.0, and conduct training and attack on GPU server. The experimental data set are collected using the curves in Section 3.1, and the initial collected curves are in .trs format, and we need to convert the trs files to h5 files for the convenience of conducting experiments. A total of 56000 curves were collected for this experiment, of which 46000 were used for training and 10000 were used as attack curves.

4.3. Experimental analysis

In this subsection, to obtain the best multilayer perceptron model, the experiment will be divided into two steps: 1)

determining the extraction method and size of feature points; 2) determining the number of MLP hidden layers and the activation function.

1) Determining the extraction method and size of feature points

In this part, in order to evaluate the influence of feature point extraction method and size on the experimental effect. Tentatively, the structure of the multilayer perceptron is set as follows: the hidden layer is 3 layers, the neuron size of each layer is 64, the activation function uses relu, the number of training rounds epochs is 100, and the batch size is 1300. firstly, in order to verify the effect of the feature extraction method on the training effect of the multilayer perceptron model, the feature extraction algorithms selected in this paper are SOST [15], NICV, and PCA [16].

Table 1. Effectiveness of attacks based on feature extraction

Methods	Train set	Attack	accuracy	entropy
SOST	46000	1	78.8%	4.1
NICV	46000	1	80.7%	3.7
PCA	46000	1	83.2%	3.1

The classification accuracy and guess entropy obtained when template attacks are performed with different feature algorithms for extracting feature points are given in Table 1. When using PCA to extract feature points, the highest classification accuracy and the smallest guess entropy of the attack are obtained, so in the subsequent experiments, PCA principal component analysis will be used to extract feature points for the experimental data.

In the extraction of feature points, in addition to the way of feature extraction will have an impact on the experimental results, if too few feature points are extracted, effective information will be filtered, and too much invalid information will interfere with the experimental results. Table 2 below gives the attack classification accuracy and guess entropy at different PCA sizes, and it can be seen that the attack effect is best when the number of feature points extracted using PCA is 128.

Table 2. Effectiveness of PCA-based attacks

Methods	Train set	Attack	accuracy	entropy
64	46000	1	84.7%	2.7
128	46000	1	85.1%	2.5
200	46000	1	84.5%	2.9
256	46000	1	83.2%	3.1

2) Determining hidden layers and activation function

The number of hidden layers, the number of neurons, and the activation function are the basic structure of the multilayer perceptron. In order to determine the best multilayer perceptron model for the TALE algorithm side channel attack, the number of neurons is set to 32 and the number of layers and activation function are experimented separately.

From Figure 4, it can be seen that when using different layers, the best results are obtained when the hidden layer is 3. Figure 5 shows the experimental results under different activation functions, and it can be seen that for the three activation functions the effects are not very different, and the effect of Selu is slightly better than the remaining two functions. For the TALE algorithm, the best results are obtained when the number of hidden layers of the multilayer perceptron model is 3 and the activation function is Selu.

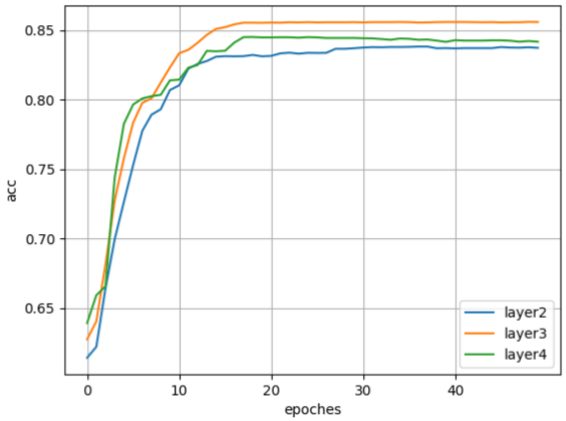

Figure 4. Effect of hidden layer on MLP model

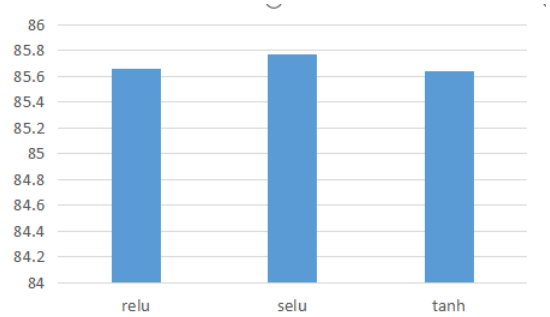

Figure 5. Effect of activation function on MLP model

5. Conclusion

To address the problem of side channel information leakage during the implementation of TALE KEM on STM32 board, this paper theoretically analyzes the vulnerability points on the TALE algorithm and proves the existence of leakage points by analyzing the leakage of the vulnerability points. In order to be able to recover the messages generated randomly by the implementation, an unused feature extraction method is used for effective information extraction and combined with a multilayer perceptron model for template attack. The experimental results show that using PCA to extract feature points is better than other methods, and the number of hidden layers and neurons in the multilayer perceptron has a greater impact on the experimental results. In future research, we will work on finding better models to improve the classification accuracy and reduce the guess entropy.

Acknowledgment

This work was supported in part by the Sichuan Science and Technology Program (No. 2021ZYD0011)

References

[1] Shor PW. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. SIAM Review, Society for Industrial and Applied Mathematics, 1999,41(2):303–332.

[2] Pessl P, Primas R. More practical single-trace attacks on the number theoretic transform. In: Schwabe P, Thériault N, eds. Proc. of the Progress in Cryptology—LATINCRYPT 2019. Cham: Springer Int'l Publishing, 2019. 130–149.

[3] Z. Xu, O. Pemberton, S. S. Roy, D. Oswald, W. Yao and Z. Zheng, "Magnifying Side-Channel Leakage of Lattice-Based Cryptosystems With Chosen Ciphertexts: The Case Study of Kyber," in IEEE Transactions on Computers, vol. 71, no. 9, pp. 2163-2176, 1 Sept. 2022, doi: 10.1109/TC.2021.3122997.

[4] Ravi P, Roy SS, Chattopadhyay A, Bhasin S. Generic side-channel attacks on CCA-secure lattice-based PKE and KEMS. IACR Trans. on Cryptographic Hardware and Embedded Systems, 2020, 2019: 307-335.

[5] Amiet D, Curiger A, Leuenberger L, et al. Defeating newhope with a single trace[C]//International Conference on Post-Quantum.

[6] Lyubashevsky V, Peikert C, Regev O. On ideal lattices and learning with errors over rings. In: Gilbert H, ed. Proc. of the Advances in Cryptology-EUROCRYPT 2010. Berlin, Heidelberg: Springer-Verlag, 2010. 1-23.

[7] Mathan SA, Koedinger KR. Fostering the intelligent novice: Learning from errors with metacognitive tutoring. Educational Psychologist, Routledge, 2005, 40(4): 257-265. [doi:10.1207/s15326985ep4004_7].

[8] Kocher P. Timing attacks on implementations of Diffie-Hellman,RSA, DSS, and other systems[C]ll Advances in Cryptology—CRYPTO 1996, LNCS 1109,1996: 104-113.

[9] Boneh D, DeMillo R A, Lipton R J.On the importance of checking cryptographic protocols forfaults[C] /l Advances in Cryptology—EUROCRYPT 1997, LNCS 1233,1997:37-51.

[10] Kocher P, Jaffe J, Jun B. Differential power analysis[C] /l Advances in Cryptology—CRYPTO1999, LNCS 1666, 1999: 388-397.

[11] Brier E, Clavier C, Olivier F. Correlation Power Analysis with a Leakage Model[M]/ / Joye M, Quisquater J. Cryptographic Hardware and Embedded Systems-CHES 2004. Berlin, Germany: Springer, 2004: 16-29.

[12] Kocher P, Jaffe J, Jun B, et al. Introduction to Differential Power Analysis and Related Attacks[J].Journal of Cryptographic Engineering, 2011, 1(1) : 5-27.

[13] Chari S, Rao JR, Rohatgi P. Template attacks. In: Kaliski BS, Koççetin K, Paar C, eds. Proc. of the Cryptographic Hardware and Embedded Systems-CHES 2002. Berlin, Heidelberg: Springer-Verlag, 2003. 13-28.

[14] S. Bhasin, J. -L. Danger, S. Guilley and Z. Najm, "NICV: Normalized inter-class variance for detection of side-channel leakage," 2014 International Symposium on Electromagnetic Compatibility, Tokyo, Tokyo, Japan, 2014, pp. 310-313.

[15] Gierlichs, B., Lemke-Rust, K., Paar, C. (2006). Templates vs. Stochastic Methods. In: Goubin, L., Matsui, M. (eds) Cryptographic Hardware and Embedded Systems - CHES 2006. CHES 2006. Lecture Notes in Computer Science, vol 4249. Springer, Berlin, Heidelberg. https://doi.org/10.1007/11894063_2.

[16] Archambeau, C., Peeters, E., Standaert, F.X., Quisquater, J.J. (2006). Template Attacks in Principal Subspaces. In: Goubin, L., Matsui, M. (eds) Cryptographic Hardware and Embedded Systems - CHES 2006. CHES 2006. Lecture Notes in Computer Science, vol 4249. Springer, Berlin, Heidelberg. https://doi.org/10.1007/11894063_1.