How to Choose Interesting Points for Template Attacks?

Guangjun Fan[1], Yongbin Zhou[2], Hailong Zhang[2], Dengguo Feng[1]

[1] Trusted Computing and Information Assurance Laboratory,
Institute of Software, Chinese Academy of Sciences,
No.4, ZhongGuanCun South Fourth Street, Haidian District, Beijing, 100190, China
guangjunfan@163.com, feng@tca.iscas.ac.cn
[2] State Key Laboratory of Information Security,
Institute of Information Engineering, Chinese Academy of Sciences,
89-A, Minzhuang Rd, Haidian District, Beijing, 100093, China
zhouyongbin@iie.ac.cn, zhanghailong@iie.ac.cn

Abstract. Template Attacks are widely accepted to be the most powerful side-channel attacks from an information theoretic point of view. For Template Attacks, many papers suggested an accepted guideline for choosing interesting points. The accepted guideline is that one should only choose one point as the interesting point per clock cycle. Up to now, many different methods of choosing interesting points were introduced. However, it is still *unclear* that which method will lead to the best classification performance for Template Attacks. In this paper, we *comprehensively* evaluate and compare the classification performance of Template Attacks when using different methods of choosing interesting points. Evaluation results show that the classification performance of Template Attacks has *obvious* difference when different methods of choosing interesting points are used. The CPA based method and the SOST based method will lead to the best classification performance. Moreover, we find that some of the methods of choosing interesting points provide the same results in the same circumstance. Finally, we correctly and experimentally prove the accepted guideline for choosing interesting points for Template Attacks is correct by presenting a new way of conducting Template Attacks.

Keywords: Cryptographic Devices, Physical Security, Side-Channel Attacks, Power Analysis Attacks, Profiled Side-Channel Attacks, Template Attacks, Interesting Points.

1 Introduction

Side-channel attacks are one of the most important threats against modern cryptographic implementations. The basic idea of these attacks is to determine the key of a cryptographic device by exploiting its power consumption [11], its electromagnetic radiation [20], its execution time [19], and many more [21]. Traditional security notions (such as chosen-ciphertext security for public-key encryption schemes) do not provide any security guarantee against such attacks,

and many implementations of provably secure cryptosystems were broken by side-channel attacks.

Power analysis attacks have received such a large amount of attention because they are very powerful and can be conducted relatively easily. Therefore, let us focus exclusively on power analysis attacks. As an important method of Power Analysis Attacks, Template Attacks were firstly proposed by S. Chari et al. in 2002 [1]. Template Attacks are profiled side-channel attacks. Under the assumption that one (an actual attacker or an evaluator) has a reference device identical or similar to the targeted device, and thus be well capable of characterizing power leakages of the targeted device, Template Attacks are widely accepted to be the strongest side-channel attacks from an information theoretic point of view [1]. We note that, Template Attacks are also important tools to evaluate the physical security of a cryptographic device.

Template Attacks consist of two stages. The first stage is the profiling stage and the second stage is the extraction stage. In the profiling stage, one captures some actual power traces from a reference device identical or similar to the targeted device and builds templates for each key-dependent operation with the actual power traces. In the extraction stage, one can exploit a small number of actual power traces measured from the targeted device and the templates obtained from the profiling stage to classify the correct (sub)key.

Motivations Note that for real-world implementation of cryptography devices, one side-channel leakage trace (i.e. one actual power trace for the case of power analysis attacks) usually contains multiple samples corresponding to the targeted intermediate value. The reason is that the key-dependent operations usually take more than one instruction cycles. In addition, according to Nyquist-Shannon sampling theorem, the acquisition rate of the signal acquisition device is always set to be several times faster than the working frequency of the targeted cryptographic device.

For Template Attacks to be practical, it is paramount that not all the samples of an actual power trace are part of the templates. To reduce the number of samples and the size of the templates, one needs to choose some special samples as the interesting points in actual power traces. The interesting points are those time samples that contain the most information about the characterized key-dependent operations. For classical Template Attacks, many papers [2,3,5,10,12] suggested an accepted guideline for choosing interesting points. The accepted guideline is that one should *only* choose *one* point as the interesting point per clock cycle.

Up to now, many different methods of choosing interesting points were introduced. They are *Difference Of Means based method* [1] (DOM), *Sum Of Squared Differences based method* [10] (SOSD), *Correlation Power Analysis based method* (Chapter 6 in [11]) (CPA), *Sum Of Squared pairwise T-differences based method* [10] (SOST), *Signal-to-Noise Ratios based method* (pp.73 in [11]) (SNR), *Variance based method* [16] (VAR), *Mutual Information Analysis based method* [17] (MIA). *Kolmogorov-Smirnov Analysis based method* [18] (KSA), and *Principal Component Analysis based method* [3] (PCA). On one hand, for

the selection of interesting points, an important observation is that applying different methods of choosing interesting points onto the same actual power traces may induce different classification performance for Template Attacks, even if it is explicitly required that all interesting points selected must correspond to the same targeted intermediate value. On the other hand, although these methods could also be used to choose interesting points for one stage attacks (Such as Differential Power Analysis), Template Attacks are two stages attacks and have distinct principle compared with one stage attacks. Therefore, putting these two things together, it makes a very practical sense to investigate the question that which method of choosing interesting points will lead to the best classification performance for Template Attacks.

Another important question which needs to be considered is as follows. Previous papers [2, 4] suggested that one should obey the accepted guideline for choosing interesting points for Template Attacks since more points in the same clock cycle do not provide more information. Moreover, for classical Template Attacks, numerical obstacles will arise when inverting the covariance matrix if one uses more than one point as the interesting point per clock cycle and using more points leads to poorer classification performance for Template Attacks. However, these do not mean that the accepted guideline for choosing interesting points is proved in a correct and reasonable way. First, no previous work directly verifies (theoretically or experimentally) the statement that more points in the same clock cycle do not provide more information. Second, no one clearly gives out what the numerical obstacles accurately are. Finally, when the numerical obstacles do not exist, whether using more than one point as the interesting point per clock cycle will lead to better classification performance is still *unknown* (If the answer is positive, one can further use more than one point as the interesting point per clock cycle to improve the classification performance of Template Attacks.). Therefore, the accepted guideline for choosing interesting points is still not proved in a correct and reasonable way.

In this paper, we try to answer the above two important questions.

Contributions Main contributions of this paper are two-folds.

Firstly, we comprehensively evaluate and compare the classification performance of Template Attacks when using different methods of choosing interesting points. Evaluation results show that the classification performance of Template Attacks has obvious difference when different methods of choosing interesting points are used. The Correlation Power Analysis based method and the Sum Of Squared pairwise T-differences based method will lead to the best classification performance. Moreover, some methods of choosing interesting points will lead to the same results in the same circumstance.

Secondly, we correctly and experimentally prove the accepted guideline for choosing interesting points for Template Attacks is correct by presenting a new way of conducting Template Attacks.

Related Work Template Attacks were firstly introduced in [1]. Answers to some basic and practical issues of Template Attacks were provided in [2], such as how to choose interesting points in an efficient way and how to preprocess noisy

data. Efficient methods were proposed in [4] to avoid several possible numerical obstacles when implementing Template Attacks. N. Hanley et al. [12] presented a variant of Template Attacks that can be applied to block ciphers when the plaintext and ciphertext used are unknown. In [8], Template Attacks were used to attack a masking protected implementation of a block cipher. Recently, a simple pre-processing technique of Template Attacks, normalizing the sample values using the means and variances was evaluated for various sizes of test data [7]. B. Gierlichs et al. [10] made a systematic comparison of Template Attacks and Stochastic Model based Attacks [24]. How to best evaluate the profiling stage and the extraction stage of profiled side-channel attacks by using the information theoretic metric and the security metric was shown in [22].

LDA-based Template Attacks were introduced in [9]. However, this kind of Template Attacks depends on the condition of equal covariances [4] (Please see Section 2.1.1 for more details.), which does not hold in most settings. Therefore, it is not a better choice compared with PCA-based Template Attacks (Please see Section 2.3 for more details.) in most settings [4] and we ignore this kind of attacks here.

Organization of This Paper The rest of this paper is organized as follows. In Section 2, we briefly review the different methods of choosing interesting points, Template Attacks as well as PCA-based Template Attacks. In Section 3, we introduce the classical way and our new way of conducting Template Attacks. The two ways are used to show our contributions. In Section 4, we comprehensively evaluate and compare the classification performance of Template Attacks when using different methods of choosing interesting points. We also correctly prove the accepted guideline for choosing interesting points for Template Attacks by our new way in this section. In Section 5, we conclude the paper and show the future work.

2 Preliminaries

In this section, we briefly review the different methods of choosing interesting points, Template Attacks as well as PCA-based Template Attacks.

2.1 The Methods of Choosing Interesting Points

Now, we briefly review the different methods of choosing interesting points. They are DOM, SOSD, CPA, SOST, SNR, VAR, MIA, and KSA. The Principal Component Analysis based method (PCA) will be introduced in Section 2.3.

We compute the signal-strength estimate $\mathsf{F}(t)$ for every point P_t corresponding to the targeted intermediate value in actual power traces using the different methods of choosing interesting points. Then, the interesting points are chosen based on the value of the signal-strength estimate $\mathsf{F}(t)$.

Assume that there are K key-dependent operations O_i $(i = 0, 1, \ldots, K-1)$ corresponding to the targeted intermediate value of the implementation of the

targeted cryptographic algorithm (e.g. AES-128). Every key-dependent operation O can be expressed by a function of the (sub)key key and the input plaintext m, namely $O = g(m, key)$ (For example, when one chooses the outputs of the S-boxes (denoted by $Sbox$) in the first round of the AES-128 implementation as the targeted intermediate value, $O = Sbox(m \oplus key)$.). One can predicate the hypothetical power consumption value of the key-dependent operation $O = g(m, key)$ by $h(g(m, key))$, where h is a hypothetical leakage function (In this paper, we choose the typical Hamming Weight function (pp.40-41 in [11]) as the leakage function.).

Before introducing specific methods of choosing interesting points, we first assume that one invokes the implementation of the targeted cryptographic algorithm n times and obtains n actual power traces (The n actual power traces are denoted by $\mathbf{S}_1, \mathbf{S}_2, \ldots, \mathbf{S}_n$. For a point P_t in the actual power trace $\mathbf{S}_i$, its power consumption value is denoted by $S_i[P_t]$.) which are used to choose interesting points. The n actual power traces are sampled from the implementation with a fixed key and random input plaintexts. In the scenario of choosing interesting points for Template Attacks, one knows the key and every input plaintexts for the n invocations (Note that, in the profiling stage, the reference device is under one's full control.).

For a fixed point P_t, the power consumption values of the n actual power traces are stored in the vector $\mathbf{R}(t) := (S_1[P_t], S_2[P_t], \ldots, S_n[P_t])$. The hypothetical power consumption values for the n invocations are stored in vector $\mathbf{H}(n) := (h(g(m_1, key)), h(g(m_2, key)), \ldots, h(g(m_n, key)))$, where m_i is the input plaintext of the i^{th} invocation and key is the (sub)key. Let's define the sets $G_i := \{\mathbf{S}_j | g(m_j, key) = O_i\}, i = 0, 1, \ldots, K-1$. For the point P_t, we let $M_i[P_t] = \sum_{l=1}^{|G_i|} S_l[P_t]/|G_i|$ $(i = 0, 1, \ldots, K-1)$, where $\mathbf{S}_l \in G_i$ and the symbol $|G_i|$ denotes the cardinality of the set G_i. The methods of choosing interesting points are shown in the following.

DOM In this method,

$$\mathsf{F}(t) = \sum_{i \neq j} M_i[P_t] - M_j[P_t],$$

where $i, j \in \{0, 1, \ldots, K-1\}$.

SOSD This method of choosing interesting points is similar to DOM, but the signal-strength estimate is computed as follows

$$\mathsf{F}(t) = \sum_{i \neq j} (M_i[P_t] - M_j[P_t])^2,$$

where $i, j \in \{0, 1, \ldots, K-1\}$.

CPA In this method, the signal-strength estimate equals to the Pearson correlation coefficient between actual power traces and hypothetical power consumptions. Specifically speaking,

$$\mathsf{F}(t) = \rho(\mathbf{H}(n), \mathbf{R}(t)) = \frac{\sum_{i=1}^{n} (h(g(m_i, key)) - \overline{\mathbf{H}(n)}) \cdot (S_i[P_t] - \overline{\mathbf{R}(t)})}{\sqrt{\sum_{i=1}^{n} (h(g(m_i, key)) - \overline{\mathbf{H}(n)})^2 \cdot (S_i[P_t] - \overline{\mathbf{R}(t)})^2}},$$

where $\overline{\mathbf{H}(n)}$ and $\overline{\mathbf{R}(t)}$ respectively denote the mean values of the vectors $\mathbf{H}(n)$ and $\mathbf{R}(t)$.

SOST The SOST is based on the T-Test, which is a standard statistical tool to meet the challenge of distinguishing noisy signals. For a point P_t, let $\sigma_i^2(t)$ denote the variance of all the sample data $S_i[P_t]$, where $\mathbf{S}_i \in G_i$. The signal-strength estimate of this method is shown as follows:

$$\mathsf{F}(t) = \sum_{i \neq j} \left(\frac{M_i[t] - M_j[t]}{\sqrt{\sigma_i^2(t)/|G_i| + \sigma_j^2(t)/|G_j|}} \right)^2,$$

where $i, j \in \{0, 1, \ldots, K-1\}$.

SNR Signal-to-noise ratios are commonly used in electrical engineering and signal processing. An SNR is the ratio between the signal and the noise component of a measurement. The general definition of SNR in a digital environment is given in the following:

$$SNR = \frac{Var(Signal)}{Var(Noise)}.$$

In case of a power analysis attack, for a point P_t, the *signal* corresponds to PC_{exp}^t, which is the component of the power consumption that is exploitable. The exploitable power consumption PC_{exp}^t is the only component that contains relevant information for an attacker in a given attack scenario. The total power consumption of point P_t (denoted by PC^t) minus PC_{exp}^t is viewed as the *noise* component. Therefore, it has that

$$\mathsf{F}(t) = \frac{Var(PC_{exp}^t)}{Var(PC^t - PC_{exp}^t)},$$

where the variances are computed based on the n actual power traces.

VAR In this method, for point P_t, the signal-strength estimate is computed as follows

$$\mathsf{F}(t) = Var(\mathbf{R}(t)).$$

MIA The mutual information [23] between two discrete variables X and Y is defined to be $I(X;Y) = H(X) - H(X|Y)$, where $H(X)$ is the entropy of X, $H(X|Y)$ is the conditional entropy of X given Y. One can compute

$$I(X,Y) = \sum_{x \in X, y \in Y} \mathsf{Pr}[X = x, Y = y] \cdot log_2\Big(\frac{\mathsf{Pr}[X = x, Y = y]}{\mathsf{Pr}[X = x] \cdot \mathsf{Pr}[X = x]}\Big).$$

In order to choose interesting points for Template Attacks, we compute

$$\mathsf{F}(t) = I(\mathbf{H}(n), \mathbf{R}(t)).$$

KSA The Kolmogorov-Smirnov Analysis is a nonparametric test method, quantifies a distance between the empirical cumulative distribution function

(CDF) of two random variables to determine the similarity of them. Assume that the random variable A has n samples (denoted by $A_1, A_2, \ldots, A_n$), its empirical CDF is $U_A(x) = \frac{1}{n}\sum_{i=1}^{n} I_{A_i \leq x}$, where $I_{A_i \leq x}$ is an indicator function. The value of $I_{A_i \leq x}$ is 1 when $A_i \leq x$; otherwise, it is 0. Similarly, the empirical CDF of random variable B is $U_B(x)$. The distance between A and B is defined to be $KS(A \parallel B) = sup_{x \in A \cup B} |U_A(x) - U_B(x)|$, where sup_x is the supremum of the set of distances. The signal-strength estimate is computed as follows:

$$\mathsf{F}(t) = E[KS(\mathbf{R}(t) \parallel (\mathbf{R}(t)|\mathbf{H}(n)))].$$

2.2 Template Attacks

Template Attacks consist of two stages. The first stage is the profiling stage and the second stage is the extraction stage. We will introduce the two stages in the following.

The Profiling Stage Assume that there exist K different (sub)keys $key_i, i = 0, 1, \ldots, K-1$ which need to be classified. Also, there exist K different key-dependent operations O_i, $i = 0, 1, \ldots, K-1$. Usually, one will built K templates, one for each key-dependent operation O_i. One can exploit some methods to choose N interesting points $(P_0, P_1, \ldots, P_{N-1})$. Each template is composed of a mean vector and a covariance matrix. Specifically speaking, the mean vector is used to estimate the data-dependent portion of side-channel leakages. It is the average signal vector $\mathbf{M}_i = (M_i[P_0], \ldots, M_i[P_{N-1}])$ for each one of the key-dependent operations. The covariance matrix is used to estimate the probability density of the noises at different interesting points. It is assumed that noises at different interesting points approximately follow the multivariate normal distribution. A N dimensional noise vector $\mathbf{n}_i(\mathbf{S})$ is extracted from each power trace $\mathbf{S} = (S[P_0], \ldots, S[P_{N-1}])$ representing the template's key dependency O_i as $\mathbf{n}_i(\mathbf{S}) = (S[P_0] - M_i[P_0], \ldots, S[P_{N-1}] - M_i[P_{N-1}])$. One computes the $(N \times N)$ covariance matrix $\mathbf{C}_i$ from these noise vectors. The probability density of the noises occurring under key-dependent operation O_i is given by the N dimensional multivariate Gaussian distribution $p_i(\cdot)$, where the probability of observing a noise vector $\mathbf{n}_i(\mathbf{S})$ is:

$$p_i(\mathbf{n}_i(\mathbf{S})) = \frac{1}{\sqrt{(2\pi)^N |\mathbf{C}_i|}} exp\Big(-\frac{1}{2}\mathbf{n}_i(\mathbf{S})\mathbf{C}_i^{-1}\mathbf{n}_i(\mathbf{S})^T \Big) \quad \mathbf{n}_i(\mathbf{S}) \in \mathbb{R}^N. \qquad (1)$$

In equation (1), the symbol $|\mathbf{C}_i|$ denotes the determinant of $\mathbf{C}_i$ and the symbol $\mathbf{C}_i^{-1}$ denotes its inverse. We know that the matrix $\mathbf{C}_i$ is the estimation of the true covariance $\mathbf{\Sigma}_i$. The condition of equal covariances [4] means that the leakages from different key-dependent operations have the same true covariance $\mathbf{\Sigma} = \mathbf{\Sigma}_0 = \mathbf{\Sigma}_1 = \cdots = \mathbf{\Sigma}_{K-1}$. In most settings, the condition of equal covariances does not hold. Therefore, in this paper, we only consider the device in which the condition of equal covariances does not hold.

The Extraction Stage Assume that one obtains t power traces (denoted by $\mathbf{S}_1, \mathbf{S}_2, \ldots, \mathbf{S}_t$) from the targeted device in the extraction stage. For example,

when the power traces are statistically independent, one will apply maximum likelihood approach on the product of conditional probabilities (pp.156 in [11]), i.e.

$$key_{ck} := argmax_{key_i}\left\{\prod_{j=1}^{t}\mathsf{Pr}(\mathbf{S}_j|key_i), i=0,1,\ldots,K-1\right\},$$

where $\mathsf{Pr}(\mathbf{S}_j|key_i) = p_{f(\mathbf{S}_j,key_i)}(n_{f(\mathbf{S}_j,key_i)}(\mathbf{S}_j))$. The key_{ck} is considered to be the correct (sub)key. The output of the function $f(\mathbf{S}_j, key_i)$ is the index of a key-dependent operation. For example, when the output of the first S-box in the first round of AES-128 is chosen as the targeted intermediate value, one builds templates for each output of the S-box. In this case, $f(\mathbf{S}_j, key_i) = Sbox(m_j \oplus key_i)$, where m_j is the input plaintext corresponding to the power trace $\mathbf{S}_j$.

2.3 PCA-based Template Attacks

A more systematic approach of choosing interesting points, which relies on the data variability, is to choose the interesting points based on Principal Component Analysis (PCA). PCA-based Template Attacks [3] exploit a continual point fragment correspond to the targeted intermediate value in actual power traces (We assume the length of the continual point fragment is N.). One first computes the empirical covariance matrix, which is given by

$$\mathbf{ECM} = \frac{1}{K}\sum_{i=0}^{K-1}(\mathbf{M}_i - \overline{\mathbf{M}})(\mathbf{M}_i - \overline{\mathbf{M}})^T.$$

The quantity $\overline{\mathbf{M}} = \sum_{i=0}^{K-1}\mathbf{M}_i/K$ is the average of the mean vectors. Let us denote the matrixes of eigenvectors and eigenvalues of $\mathbf{ECM}$ by $\mathbf{U}$ and Δ, i.e. $\mathbf{ECM} = \mathbf{U}\Delta\mathbf{U}^T$. The principal directions $\{w_i\}_{i=1}^{L}$ are the columns of $\mathbf{U}$ that correspond to the L largest eigenvalues of Δ. The corresponding matrix of principal directions is denoted $\mathbf{W} \in \mathbb{R}^{N\times L}$. The *Cumulative Percentage of Total Variation* (CPTV) [13] is often used to determine how many principal directions should be exploited (i.e. to determine the concrete value of L). One uses projected mean vectors $\{\mathbf{W}^T\mathbf{M}_i^T\}_{i=0}^{K-1}$ and projected covariance matrices $\{\mathbf{W}^T\mathbf{C}_i\mathbf{W}\}_{i=0}^{K-1}$ to conduct the attacks. Specifically speaking, the probability of observing a noise vector when one assumes the key-dependent operation is O_i is computed by

$$p_i(\mathbf{n}_i(\mathbf{S})) = \frac{exp(-\frac{1}{2}\mathbf{n}_i(\mathbf{S})\mathbf{W}(\mathbf{W}^T\mathbf{C}_i\mathbf{W})_i^{-1}(\mathbf{n}_i(\mathbf{S})\mathbf{W})^T)}{\sqrt{(2\pi)^L|\mathbf{W}^T\mathbf{C}_i\mathbf{W}|}} \quad \mathbf{n}_i(\mathbf{S}) \in \mathbb{R}^N. \qquad (2)$$

One classifies the correct (sub)key based on the probability computed by equation (2). One can use the method introduced in paper [4] to compute the projected mean vectors and the projected covariance matrices. Using this method, one can avoid both numerical obstacles and the computation of large covariance matrices. We uses this advanced method to conduct PCA-based Template Attacks in this paper.

3 Strategies to Conduct Template Attacks

In this section, for the purpose of comparison, we will introduce three different strategies to conduct Template Attacks. The first strategy and the second strategy are the classical way of conducting Template Attacks but using different number of interesting points per clock cycle. The third strategy is our new way of conducting Template Attacks.

Assume that, in one actual power trace, there is a *continual* point fragment

$$(P_0, P_1, \ldots, P_{N-1}),$$

which is correspond to the targeted intermediate value and has length N. We also assume that these N points are in c continual clock cycles. Therefore, there are N/c points per clock cycle. Let the symbol $P_{(i,j)}$ denotes the j^{th}, $j \in \{1, \ldots, N/c\}$ interesting point in the i^{th}, $i \in \{1, \ldots, c\}$ clock cycle. For interesting points in the same clock cycle, their orders are determined by the signal-strength estimate $\mathsf{F}(t)$ which is computed by a kind of method of choosing interesting points. For example, when one uses Correlation Power Analysis based method, he computes the coefficient of correlation of each point in the clock cycle. The point with the highest coefficient of correlation is set to be $P_{(i,1)}$ and the point with the lowest coefficient of correlation is set to be $P_{(i,N/c)}$.

3.1 Strategy 1

In this strategy, one only uses one point as the interesting point per clock cycle and chooses c points $\{P_{(1,1)}, P_{(2,1)}, \ldots, P_{(c,1)}\}$ from the N continual points as the c interesting points. Then, one conducts classical Template Attacks with templates which are built based on the c interesting points. We call the attack with this strategy as "ATTACK-1".

3.2 Strategy 2

In this strategy, one uses more than one point as the interesting points per clock cycle. In order to show this strategy more clearly, we take the simplest case as an example. We assume that one uses two points as the interesting points per clock cycle. Therefore, $2c$ points are chosen from the N continual points as the interesting points: $\{(P_{(1,1)}, P_{(1,2)}), (P_{(2,1)}, P_{(2,2)}), \ldots, (P_{(c,1)}, P_{(c,2)})\}$.

Then, one conducts classical Template Attacks with templates which are built based on the $2c$ interesting points. This means that one needs to compute a $(1 \times 2c)$ mean vector and a $(2c \times 2c)$ covariance matrix for each template. We call the attack with Strategy 2 as "ATTACK-2". Note that, the success rate of Strategy 2 will reduce when one uses more points as the interesting points per clock cycle to conduct Template Attacks [2, 4]. Our experiments in the next section also verified this fact.

3.3 Strategy 3 (Our New Way)

Strategy 3 is our new way of conducting Template Attacks, which is exploited to correctly prove the accepted guideline for choosing interesting points for Template Attacks. In our new way, during the profiling stage, one uses more than one point as the interesting points per clock cycle. We also take the simplest case as an example. Therefore, $2c$ points are chosen from the N continual points as the interesting points:

$$\big\{(P_{(1,1)}, P_{(1,2)}), (P_{(2,1)}, P_{(2,2)}), \ldots, (P_{(c,1)}, P_{(c,2)})\big\}.$$

Then, one divides the $2c$ interesting points into two sets. In the first set $\mathsf{Set1}$, there are c interesting points $\mathsf{Set1} = \{P_{(1,1)}, P_{(2,1)}, \ldots, P_{(c,1)}\}$. The rest c interesting points are in the second set $\mathsf{Set2} = \{P_{(1,2)}, P_{(2,2)}, \ldots, P_{(c,2)}\}$. Note that, in each set, any 2 points of the c interesting points are not in the same clock cycle. But the 2 points $(P_{(i,1)}, P_{(i,2)}),\ i \in \{1, 2, \ldots, c\}$ are in the same clock cycle. In the following, one builds templates in the same way as classical Template Attacks with the c interesting points in $\mathsf{Set1}$ and obtains a group of templates (denoted by $\mathsf{G1}$). Similarly, with the same power traces used for obtaining $\mathsf{G1}$, one builds templates with the c interesting points in $\mathsf{Set2}$ and obtains another group of templates (denoted by $\mathsf{G2}$). At this point, the profiling stage is finished.

In the extraction stage, one first computes a sequence

$$\big\{\mathsf{Pr}(1,0), \mathsf{Pr}(1,1), \ldots, \mathsf{Pr}(1,K-1)\big\},$$

where the value $\mathsf{Pr}(1,i)$ represents the probability of the i^{th} (sub)key is the correct (sub)key (In the example of Section 2.2, $\mathsf{Pr}(1,i)$ equals to $\prod_{j=1}^{t} \mathsf{Pr}(S_j|key_i)$.) using $\mathsf{G1}$ with some actual power traces obtained from the targeted device in the same way as classical Template Attacks. Then, one sorts the sequence in decreasing order and computes $\mathsf{Index}(1,i),\ i = 0, 1, \ldots, K-1$ for each (sub)key. The value $\mathsf{Index}(1,i)$ represents the sequence number of $\mathsf{Pr}(1,i)$ in the sorted sequence. Similarly, one computes another sequence $\{\mathsf{Pr}(2,0), \mathsf{Pr}(2,1), \ldots, \mathsf{Pr}(2,K-1)\}$ using $\mathsf{G2}$ with the same actual power traces obtained from the targeted device. Then, he computes $\mathsf{Index}(2,i),\ i = 0, 1, \ldots, K-1$ for each (sub)key. The candidate value of the correct key is computed by

$$key_{ck} := argmin_i\big\{\mathsf{Index}(1,i) + \mathsf{Index}(2,i), i \in \{0, 1, \ldots, K-1\}\big\}.$$

We call the attack with our new way as "ATTACK-3".

Discussions There is a kind of generalization about our new way of conducting Template Attacks. The generalization is to use more points as the interesting points per clock cycle to conduct our new way. Assume that one uses s $(2 < s \leq N/c)$ points as the interesting points per clock cycle, he will divide the cs interesting points into s different sets and build s different groups of templates in the profiling stage similarly to the way introduced above. In the extraction stage, one classifies the correct (sub)key by computing

$$key_{ck} := argmin_i\big\{\mathsf{Index}(1,i) + \ldots + \mathsf{Index}(s,i), i \in \{0, 1, \ldots, K-1\}\big\}.$$

In our way, in the same set (e.g. Set1), the points from different clock cycles have nearly equal informational level. For example, in the first and the second clock cycle, we respectively choose $P_{(1,1)}$ and $P_{(2,1)}$ (rather than $P_{(2,2)}$, $P_{(2,3)}$, or $P_{(2,4)}$) to be the interesting points in Set1. Building templates with interesting points from different clock cycles but have different informational levels (e.g. $P_{(1,1)}$ and $P_{(2,4)}$) will lead to poorer classification performance.

Note that, in our new way, we *do not* build templates with interesting points in the same clock cycle *simultaneously*. Therefore, the new way *completely* avoids the numerical obstacles during inverting the covariance matrix when one uses more than one point as the interesting point per clock cycle. If the classification performance of the new way remains almost unchanged when more points are used as the interesting points per clock cycle, it will demonstrate that more points in the same clock cycle do not provide more information which can be exploit to improve the classification performance of Template Attacks. Furthermore, this means that the accepted guideline for choosing interesting points for Template Attacks is correct. Therefore, we can correctly prove the accepted guideline for choosing interesting points for Template Attacks is correct by using our new way.

4 Experimental Evaluations

In this section, we will introduce two groups of experiments. In the first group of experiments (denoted by Group 1), we comprehensively evaluate and compare the classification performance of Template Attacks when using different methods of choosing interesting points. In the second group of experiments (denoted by Group 2), we correctly prove that the accepted guideline for choosing interesting points for Template Attacks by using our new way.

For the implementation of a cryptographic algorithm with countermeasures, one usually first tries his best to use some methods to delete the countermeasures from actual power traces. If the countermeasures can be deleted, then one tries to recover the correct (sub)key using classical attack methods against unprotected implementation. For example, if one has actual power traces with random delays [15], he may first use the method proposed in [14] to remove the random delays from actual power traces and then uses classical attack methods to recover the correct (sub)key. The methods of deleting countermeasures from actual power traces are beyond the scope of this paper. Moreover, considering actual power traces without any countermeasures shows the upper bound of the physical security of the targeted cryptographic device. Therefore, we take unprotected AES-128 implementation as example. We tried to attack the outputs of all the S-boxes in the first round of an unprotected AES-128 software implementation on an typical 8-bit microcontroller STC89C58RD+ whose operating frequency is 11MHz. The actual power traces were acquired with a sampling rate of 50MS/s. The average number of actual power traces during the sampling process was 10 times.

We generated three sets of actual power traces, Set A, Set B, and Set C. The Set A captured 20,000 actual power traces which were generated with a fixed main key and random plaintext inputs. The Set B also captured 20,000 actual power traces which were generated with another fixed main key and random plaintext inputs. The set C captured 40,000 actual power traces which were generated with a fixed main key and random plaintext inputs. We used the same device to generate the three sets of actual power traces, which provides a good setting for the focuses of our research. For our device, the condition of equal covariances does not hold. This means that the differences between different covariance matrixes $\mathbf{C}_i$ are very evident (can easily be observed from visual inspection).

In all our experiments, we chose the same 3 continual clock cycles about the outputs of all the S-boxes in the first round of the unprotected AES-128 software implementation. In each clock cycle, there are 4 points. Therefore, there are 12 points (denoted by $(P_0, P_1, \ldots, P_{11})$) totally. We implemented all the methods of choosing interesting points for Template Attacks including DOM, SOSD, CPA, SOST, SNR, VAR, MIA, KSA, and PCA.

In all the experiments, PCA-based Template Attacks used the first 6 principal directions (i.e. $L = 6$ Please see Section 2.2 for more details.). For our device, the first 6 principal directions are sufficient to ensure the classification performance of PCA-based Template Attacks. The CPTV is larger than 0.997 when the first 6 principal directions are used. Using the rest few principal directions only *slightly* increase the power of this kind of attacks and it is not necessary to use all the principal directions [3,4]. For simplicity, let np denote the number of actual power traces used in the profiling stage and let ne denote the number of actual power traces used in the extraction stage. In this paper, we use the most typical metric *success rate* [6] as the metric about the classification performance of Template Attacks. We only show the evaluation results of the first S-box. For other S-boxes in the first round of the unprotected AES-128 software implementation, similar evaluation results were obtained by us for both the two groups of experiments.

4.1 Group 1

We chose interesting points by using the 40,000 actual power traces in Set C. In Table 1, we show the interesting points chosen by different methods (DOM, SOSD, CPA, SOST, SNR, VAR, MIA, and KSA) by using the 40,000 actual power traces in Set C. In Table 1, the symbol "(i, j)" denote the j^{th} interesting point in the i^{th} clock cycle (i.e. $P_{(i,j)}$).

From Table 1, we find that some methods of choosing interesting points provide the same results in the same circumstance. For example, Difference Of Means based method and Sum Of Squared Differences based method provide the same results. Correlation Power Analysis based method and Sum Of Squared pairwise T-differences based method provide the same results. Signal to Noise Ratio based method and Variance based method provide the same results.

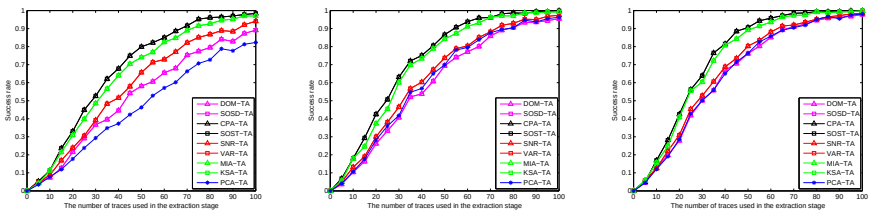
We will show the success rates of Template Attacks using different methods of choosing interesting points. According to the accepted guideline for choosing

Table 1. The interesting points chosen by different methods

	(1,1)	(1,2)	(1,3)	(1,4)	(2,1)	(2,2)	(2,3)	(2,4)	(3,1)	(3,2)	(3,3)	(3,4)
DOM	P_2	P_3	P_0	P_1	P_7	P_6	P_4	P_5	P_{11}	P_9	P_{10}	P_8
SOSD	P_2	P_3	P_0	P_1	P_7	P_6	P_4	P_5	P_{11}	P_9	P_{10}	P_8
CPA	P_3	P_2	P_1	P_0	P_6	P_5	P_4	P_7	P_9	P_{11}	P_{10}	P_8
SOST	P_3	P_2	P_1	P_0	P_6	P_5	P_4	P_7	P_9	P_{11}	P_{10}	P_8
SNR	P_0	P_2	P_3	P_1	P_7	P_4	P_6	P_5	P_{11}	P_9	P_{10}	P_8
VAR	P_0	P_2	P_3	P_1	P_7	P_4	P_6	P_5	P_{11}	P_9	P_{10}	P_8
MIA	P_2	P_3	P_1	P_0	P_6	P_5	P_7	P_4	P_9	P_{11}	P_{10}	P_8
KSA	P_2	P_3	P_1	P_0	P_6	P_5	P_7	P_4	P_9	P_{11}	P_8	P_{10}

interesting points (In the second group of experiments, we will prove the accepted guideline is correct.), for the above 8 methods of choosing interesting points, we built templates based on the points $\{P_{(1,1)}, P_{(2,1)}, P_{(3,1)}\}$, one in each clock cycle. We conducted Template Attacks using different methods of choosing interesting points with the same actual power traces both in the profiling stage and the extraction stage. Specifically speaking, we respectively chose 10,000, 15,000, and 20,000 different actual power traces from Set A to build the 256 templates using different methods of choosing interesting points. Template Attacks using the method A to choose interesting points is denoted by the symbol "A-TA". We tested the success rates of Template Attacks using different methods of choosing interesting points when one uses ne actual power traces in the extraction stage as follows. We repeated the 9 attacks (DOM-TA, SOSD-TA, CPA-TA, SOST-TA, SNR-TA, VAR-TA, MIA-TA, KSA-TA, and PCA-TA) 1,000 times. For each time, we chose ne actual power traces from Set B uniformly at random and the 9 attacks were conducted with the same ne actual power traces. We respectively recorded how many times the 9 attacks can successfully recover the correct subkey.

The success rates of Template Attacks using different methods of choosing interesting points are shown in Figure 1. The success rates of Template Attacks using different methods of choosing interesting points when ne is fixed to 50 are shown in Table 2.



(a) np =10,000 (b) np =15,000 (c) np =20,000

Fig. 1. The experiment results of different methods of choosing interesting points

From Figure 1 and Table 2, we find that CPA-based method and SOST-based method lead to the highest success rates in all cases. When np is small (e.g. $np = 10,000$), PCA-based Template Attacks lead to the lowest success rates. When np is large, DOM-based Template Attacks and SOSD-based Template Attacks lead to the lowest success rates.

Table 2. The success rates for the case $ne = 50$

np	DOM	SOSD	CPA	SOST	SNR	VAR	MIA	KSA	PCA
$10,000$	0.58	0.58	0.80	0.80	0.66	0.66	0.74	0.74	0.46
$15,000$	0.69	0.69	0.87	0.87	0.74	0.74	0.84	0.84	0.70
$20,000$	0.76	0.76	0.91	0.91	0.80	0.80	0.89	0.89	0.76

4.2 Group 2

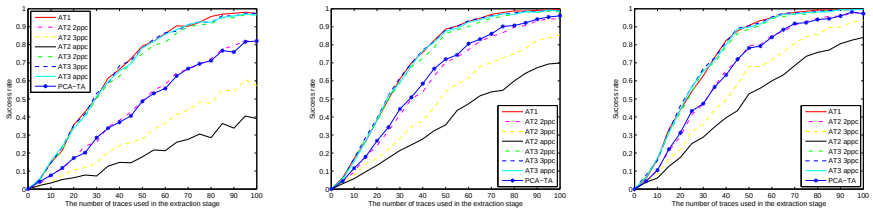
We correctly prove the accepted guideline for choosing interesting points for Template Attacks is correct with both the best and the worst methods of choosing interesting points. Based on the discoveries of the first group of experiments, we chose Correlation Power Analysis based method as the best method and Difference Of Means based method as the worst method. We also conducted PCA-based Template Attacks for the purpose of comparison.

Let "AT1" denote ATTACK-1. Let "AT2 2ppc" and "AT3 2ppc" respectively denote the case of ATTACK-2 and ATTACK-3 using the same 2 points as the interesting points per clock cycle. Let "AT2 3ppc" and "AT3 3ppc" respectively denote the case of ATTACK-2 and ATTACK-3 using the same 3 points as the interesting points per clock cycle. Let "AT2 appc" and "AT3 appc" respectively denote the case of ATTACK-2 and ATTACK-3 using all the points as the interesting points per clock cycle. Let the symbol "A>B" denotes the case that Attack A has *obvious* higher success rate than Attack B. Let the symbol "A≈B" denotes the case that Attack A has almost the same success rate as Attack B.

For both the two methods of choosing interesting points (CPA and DOM), we conducted the 8 attacks (AT1, AT2 2ppc, AT2 3ppc, AT2 appc, AT3 2ppc, AT3 3ppc, AT3 appc, and PCA-TA) with the same actual power traces both in the profiling stage and the extraction stage. We respectively chose 10,000, 15,000, and 20,000 different actual power traces from Set A to build the 256 templates for the 8 attacks. We tested the success rates of the 8 attacks when one uses ne actual power traces in the extraction stage as follows. We repeated the 8 attacks 1,000 times. For each time, we chose ne actual power traces from Set B uniformly at random and the 8 attacks were conducted with the same ne actual power traces. We respectively recorded how many times the 8 attacks can successfully recover the correct subkey.

The success rates of the 8 attacks when Correlation Power Analysis based method was used as the method of choosing interesting points are shown in

Figure 2. The success rates of the 8 attacks when Correlation Power Analysis based method was used as the method of choosing interesting points and ne is fixed to 50 are shown in Table 3. From Figure 2 and Table 3, we find that AT1≈AT3 2ppc≈AT3 3ppc≈AT3 appc>AT2 2ppc≈PCA-TA>AT2 3ppc>AT2 appc. When more points are used, the success rates of our new way (AT3 2ppc, AT3 3ppc, and AT3 appc) are almost unchanged as the success rate of ATTACK-1. This discovery shows that more points in the same clock cycle do not provide more information and the accepted guideline for choosing interesting points for Template Attacks is correct.



(a) np =10,000 (b) np =15,000 (c) np =20,000

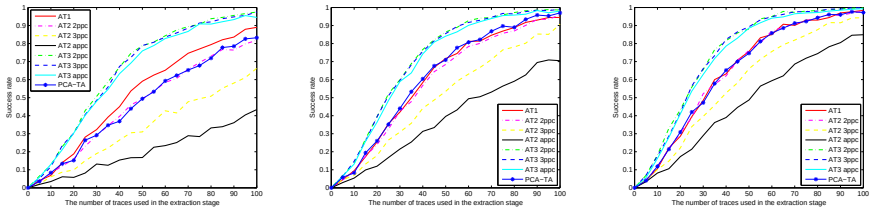
Fig. 2. The experiment results of Correlation Power Analysis based method

Table 3. The success rates for the 8 attacks when $ne = 50$ (CPA)

Attacks	$np = 10,000$	$np = 15,000$	$np = 20,000$
AT1	0.79	0.89	0.91
AT2 2ppc	0.49	0.70	0.80
AT2 3ppc	0.28	0.54	0.68
AT2 appc	0.18	0.36	0.53
AT3 2ppc	0.75	0.86	0.89
AT3 3ppc	0.78	0.88	0.91
AT3 appc	0.78	0.88	0.90
PCA-TA	0.49	0.72	0.78

The success rates of the 8 attacks when Difference Of Means based method was used as the method of choosing interesting points are shown in Figure 3. The success rates of the 8 attacks when Difference Of Means based method was used as the method of choosing interesting points and ne is fixed to 50 are shown in Table 4. However, from Figure 3 and Table 4, we find that AT3 2ppc≈AT3 3ppc≈AT3 appc>AT1≈AT2 2ppc≈PCA-TA>AT2 3ppc>AT2 appc. When more points are used, the success rates of our new way (AT3 2ppc, AT3 3ppc, and AT3 appc) are obvious higher than the success rate of ATTACK-1. This discovery shows that Difference Of Means based method is not a good

method to choose interesting points for Template Attacks once more. Because the rest points in the clock cycles (i.e. $P_{(i,2)}, P_{(i,3)}, P_{(i,4)},\ i = 1, 2, 3$) also contain valuable information which can be exploited to achieve higher success rate.



(a) np =10,000 (b) np =15,000 (c) np =20,000

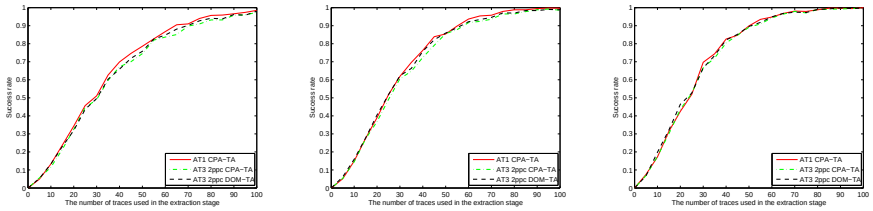
Fig. 3. The experiment results of Difference Of Means based method

Table 4. The success rates for the 8 attacks when $ne = 50$ (DOM)

Attacks	$np = 10,000$	$np = 15,000$	$np = 20,000$
AT1	0.59	0.72	0.76
AT2 2ppc	0.49	0.68	0.76
AT2 3ppc	0.31	0.54	0.66
AT2 appc	0.17	0.40	0.49
AT3 2ppc	0.79	0.86	0.88
AT3 3ppc	0.79	0.86	0.89
AT3 appc	0.76	0.84	0.89
PCA-TA	0.49	0.71	0.75

In Figure 4, we show the experiment results for AT1 and AT3 2ppc using Correlation Power Analysis based method as method of choosing interesting points (respectively denoted by AT1 CPA-TA and AT3 2ppc CPA-TA), AT3 2ppc using Difference Of Means based method as method of choosing interesting points (denoted by AT3 2ppc DOM-TA). Figure 4 shows that different methods of choosing interesting points (except PCA) lead to almost the same success rates when more than one point are used as the interesting points per clock cycle (by using our new way of conducting Template Attacks). Therefore, we suggest that one should obey the accepted guideline for choosing interesting points and uses the best method of choosing interesting points (CPA or SOST) when he conducts Template Attacks.

From Figure 2 and Figure 3, we can clearly find that the classical way of conducting Template Attacks (AT2 2ppc, AT2 3ppc, and AT2 appc) achieves lower success rates when more points are used as the interesting points per clock cycle.



(a) np =10,000 (b) np =15,000 (c) np =20,000

Fig. 4. The experiment results for CPA and DOM

Note that, the success rates of the same kind of attack method shown in the above figures and tables may have slight differences because we conducted the different groups of experiments independently and respectively even for fixed np and ne.

5 Conclusions

In this paper, we show that Correlation Power Analysis based method and Sum Of Squared pairwise T-differences based method are the best choices of choosing interesting points for Template Attacks. Moreover, we find that some methods of choosing interesting points will provide the same results. In additional, we correctly and experimentally prove the accepted guideline for choosing interesting points for Template Attacks is correct by presenting a new way of conducting Template Attacks. In the future, it is necessary to research how to choose interesting points for other profiled side-channel attacks (such as Stochastic Model based Attacks [24]) and to further verify our results in other devices such as ASIC and FPGA.

Acknowledgments This work was supported by the National Basic Research Program of China (No.2013CB338002) and National Natural Science Foundation of China (Nos.61272478, 61100225 and 61170282).

References

[1] Chari, S., Rao, J.R., Rohatgi, P.: Template Attacks. CHES2002, LNCS 2523, pp.13-28, 2003.
[2] Rechberger, C., Oswald, E.: Practical Template Attacks. WISA2004, LNCS 3325, pp.440-456, 2004.
[3] Archambeau, C., Peeters, E., Standaert, F.-X., Quisquater, J.-J.: Template Attacks in Principal Subspaces. CHES2006, LNCS 4249, pp.1-14, 2006.
[4] Choudary, O., Kuhn, M.G.: Efficient Template Attacks. CARDIS2013, LNCS 8419, pp.253-270, 2013.

[5] Bär, M., Drexler, H., Pulkus, J.: Improved Template Attacks. COSADE2010, 2010.
[6] Standaert, F.-X., Malkin, T.G., Yung, M.: A Unified Framework for the Analysis of Side-Channel Key Recovery Attacks. EUROCRYPT2009, LNCS 5479, pp.443-461, 2009.
[7] Montminy, D.P., Baldwin, R.O., Temple, M.A., Laspe, E.D.: Improving cross-device attacks using zero-mean unit-variance mormalization. Journal of Cryptographic Engineering, Volume 3, Issue 2, pp.99-110, June 2013.
[8] Oswald, E., Mangard, S.: Template Attacks on Masking—Resistance Is Futile. CT-RSA2007, LNCS 4377, pp.243-256, 2007.
[9] Standaert, F.-X., Archambeau, C.: Using Subspace-Based Template Attacks to Compare and Combine Power and Electromagnetic Information Leakages. CHES2008, LNCS 5154, pp.411-425, 2008.
[10] Gierlichs, B., Lemke-Rust, K., Paar, C.: Templates vs. Stochastic Methods A Performance Analysis for Side Channel Cryptanalysis. CHES2006, LNCS4249, pp.15-29, 2006.
[11] Mangard, S., Oswald, E., Popp, T.: Power Analysis Attacks: Revealing the Secrets of Smart Cards. Springer 2007.
[12] Hanley, N., Tunstall, M., Marnane, W.P.: Unknown Plaintext Template Attacks. WISA2009, LNCS 5932, pp.148-162, 2009.
[13] Jolliffe, I.: "Principal Component Analysis", John Wiley & Sons, Ltd, 2005.
[14] Durvaux, F., Renauld, M., Standaert, F.-X. et al.: Efficient Removal of Random Delays from Embedded Software Implementations Using Hidden Markov Models. CARDIS2012, LNCS 7771, pp. 123-140, 2013.
[15] Coron, J.-S., Kizhvatov, I.: Analysis and Improvement of the Random Delay Countermeasure of CHES 2009. CHES2010, LNCS 6225, pp.95 – 109, 2010.
[16] Mather, L., Oswald, E., Bandenburg, J., Wójcik, M.: Does My Device Leak Information? An a *priori* Statistical Power Analysis of Leakage Detection Tests. ASIACRYPT2013 Part I, LNCS 8269, pp.486 – 505, 2013.
[17] Gierlichs, B., Batina, L., Tuyls, P., Preneel, B.: Mutual Information Analysis. CHES2008, LNCS 5154, pp.426 – 442, 2008.
[18] Whitnall, C., Oswald,E., Mather, L.: An Exploration of the Kolmogorov-Smirnov Test as a Competitor to Mutual Information Analysis. CARDIS2011, LNCS 7079, pp.234 – 251, 2011.
[19] Kocher, P.C.: Timing Attacks on Implementations of Diffie-Hellman, RSA, DSS, and Other Systems. CRYPTO1996, LNCS 1109, pp.104-113, 1996.
[20] Gandolfi, K., Mourtel. C., Olivier, F.: Electromagnetic Analysis: Concrete Results. CHES2001, LNCS 2162, pp.251-261, 2001.
[21] European Network of Excellence (ECRYPT). The side channel cryptanalysis lounge. http://www.crypto.ruhr-uni-bochum.de/en sclounge.html.
[22] Standaert, F.-X., Koeune, F., Schindler, W.: How to Compare Profiled Side-Channel Attacks? ACNS2009, LNCS 5536, pp.485-498, 2009.
[23] Cover, T.M., Thomas, J.A.: Elements of Information Theory. John Wiley & Sons, Chichester, 2006.
[24] Schindler, W., Lemke, K., Paar, C.: A Stochastic Model for Differential Side Channel Cryptanalysis. CHES2005, LNCS 3659, pp.30-46, 2005.