Templates vs. Stochastic Methods
A Performance Analysis for Side Channel Cryptanalysis

Benedikt Gierlichs[1,2,⋆], Kerstin Lemke-Rust[2,⋆⋆], and Christof Paar[2]

[1] K.U. Leuven, ESAT/COSIC
Kasteelpark Arenberg 10
B-3001 Leuven-Heverlee, Belgium
benedikt.gierlichs@esat.kuleuven.be
[2] Horst Görtz Institute for IT Security
Ruhr University Bochum
44780 Bochum, Germany
{gierlichs, lemke, cpaar}@crypto.rub.de

Abstract. Template Attacks and the Stochastic Model provide advanced methods for side channel cryptanalysis that make use of 'a-priori' knowledge gained from a profiling step. For a systematic comparison of Template Attacks and the Stochastic Model, we use two sets of measurement data that originate from two different microcontrollers and setups. Our main contribution is to capture performance aspects against crucial parameters such as the number of measurements available during profiling and classification. Moreover, optimization techniques are evaluated for both methods under consideration. Especially for a low number of measurements and noisy samples, the use of a T-Test based algorithm for the choice of relevant instants can lead to significant performance gains. As a main result, T-Test based Templates are the method of choice if a high number of samples is available for profiling. However, in case of a low number of samples for profiling, stochastic methods are an alternative and can reach superior efficiency both in terms of profiling and classification.

Keywords: Template Attack, Stochastic Model, Performance Analysis, Side Channel Cryptanalysis, High-Order Attacks, Power Analysis.

1 Introduction

Side channel cryptanalysis makes use of physical leakage of a cryptographic implementation as an additional source of information for mathematical cryptanalysis. An adversary is successful, if side channel cryptanalysis yields a (sufficient) entropy loss of a secret key used in a cryptographic implementation.

The underlying working hypothesis for side channel cryptanalysis assumes that computations of a cryptographic device have an impact on instantaneous

[⋆] The research was done in cooperation with gemalto.

[⋆⋆] Supported by the European Commission through the IST Contract IST-2002-507932 ECRYPT, the European Network of Excellence in Cryptology.

L. Goubin and M. Matsui (Eds.): CHES 2006, LNCS 4249, pp. 15–29, 2006.

physical observables in the (immediate) vicinity of the device, e.g., power consumption or electromagnetic radiation [6,5]. The dependency of the measurable observables on the internal state of a cryptographic algorithm is specific for each implementation and represents the *side channel*. This relationship can be predicted, e.g., by applying a (standard) power consumption model of the implementation such as the Hamming weight or Hamming distance model [2]. Alternatively, the probability density of the observables can be profiled in advance for every key dependent internal state of the implementation.

The methods under consideration are the Template Attack [3] and the Stochastic Model [7]. Both methods include a profiling step for the estimation of a key dependent multivariate probability density of the physical observable. Our work is driven by the demand for an objective and systematic performance comparison in identical physical conditions since the *quality* of side channel measurements is one of the most crucial factors in terms of attack efficiency. Both methods are applied to measurements from two setups using two different microcontrollers running an AES implementation in software. Moreover, we apply and evaluate optimization strategies, especially with respect to the selection of time instants for the multivariate density.

This work is organized as follows. In Section 2 we give an introduction to Template Attacks and the Stochastic Model, i.e., the two methods under consideration. Our testing framework used for performance analysis is presented in Section 3. Section 4 presents results that were obtained by using the known approach for both methods, whereas Section 5 evaluates optimizations. Our contribution is summarized in Section 6.

2 Side Channel Cryptanalysis

Methods used for side channel cryptanalysis can be distinguished into *one-stage* methods without any prior knowledge about the expected side channel leakage that are directly used for key extraction and *two-stage* methods that make use of a profiling step to obtain ‘a priori’ knowledge on the side channel leakage that can be used for extracting keys later on. Both, Templates and the Stochastic Model are two-stage attacks. For profiling, two-stage methods require a cryptographic device which is identical to the device used at key extraction. While in case of attacks against stream ciphers, a further requirement is that the profiling device must allow to load keys (cp. [3]), our attacks against AES do not require this, which weakens the assumptions on the adversary’s power.

2.1 Template Attack

Templates were introduced as the strongest side channel attack possible from an information theoretic point of view [3]. For each (sub)key-dependency, a Template, i.e., a multivariate characterization of the noise in the instantaneous leakage signal, is produced during profiling. Let us assume K different (sub)key-dependent operations O_i with $1 \leq i \leq K$. During profiling, Templates T_i, one for each key dependency O_i, are generated from a large number N^1 of samples. The first part in a Template estimates the data-dependent portion of the side channel for each time instant, i.e., it is the average $m_i{}^2$ of all available samples representing the same key-dependency O_i. The second part in a Template estimates the probability density of the noise in the side channel. Before starting to characterize the noise, it is highly advisable to identify and select those time instants where the averages m_i differ significantly in order to reduce computational and storage efforts. Reference [3] proposes to compute the sum of pairwise differences between the averages, $\sum_{j,l=1}^{K} m_j - m_l$ for $l \geq j$, and to choose p points $(P_1, \ldots, P_p)$ along the peaks of the resulting difference curve. It is assumed that the noise in the side channel approximately has a multivariate normal distribution with respect to the selected instants. A p-dimensional noise vector $\boldsymbol{n_i}(L)$ is extracted from each sample L representing the Template's key dependency O_i as $\boldsymbol{n_i}(L) = (L[P_1] - m_i[P_1], \ldots, L[P_p] - m_i[P_p])$. One computes the $(p \times p)$ covariance matrix C_i from these noise vectors. The probability density of the noise occurring under key dependency O_i is then given by the p-dimensional multivariate normal distribution $\text{prob}_{C_i}(\cdot)$ where the probability of observing a noise vector $\boldsymbol{z}$ is

$$\text{prob}_{C_i}(\boldsymbol{z}) = \frac{1}{\sqrt{(2\pi)^p |\mathrm{C}_i|}} \exp\left(-\frac{1}{2}\boldsymbol{z}^T \mathrm{C}_i^{-1} \boldsymbol{z}\right), \quad \boldsymbol{z} \in \mathrm{I\!R}^{\mathrm{p}}, \tag{1}$$

$|C_i|$ denotes the determinant of C_i, and C_i^{-1} its inverse.

The strategy to classify a single sample S is a maximum likelihood hypothesis test. For each hypothetical key dependency O_i, one extracts the noise in S by subtracting the average m_i at the p selected instants yielding a noise vector $\boldsymbol{n_i}(S)$ and computes the probability $\text{prob}_{C_i}(\boldsymbol{n_i}(S))$ to observe such a noise vector using (1). The hypothesis O_i maximizing (1) is then the best candidate for the observed key dependency.

Use of Template Attacks against AES. In [3] an "expand and prune" strategy is described that is particularly useful when attacking stream ciphers. Applying this strategy, profiling and classification build a recurring cycle for sieving key candidates which means in particular that the vast effort of the profiling step cannot be precomputed. In contrast, if the attacked key is known to be sufficiently small or assailable in such blocks[3], profiling can be done independently before or after obtaining S from the device under attack. For example, to recover an 128-bit AES key one can precompute $2^8 \cdot 16$ instead of (infeasible) 2^{128} templates and - after obtaining S - immediately start the classification step which may take only a few seconds.

Improvement 1 (concerning the selection of *interesting* instants): We discovered that the sum of pairwise differences of the average signals, i.e., $\sum_{j,l=1}^{K} m_j - m_l$

[1] In this contribution, N is the number of samples available for profiling. The number of samples per key dependency is about N/K in case of a uniform distribution.

[2] We denote that each sample and m_i is a vector of sampled points in time.

[3] This is true for many block ciphers.

for $l \geq j$ is not an appropriate basis for choosing the *interesting* points in time. This is due to the fact that positive and negative differences between the averages may zeroize, which is desirable to filter noise but hides as well valuable peaks that derive from significant signal differences with alternating algebraic sign. Therefore we implemented the sum of *squared* pairwise differences of the average signals $\sum_{j,l=1}^{K}(m_j - m_l)^2$ for $l \geq j$ (also referred to as *sosd* in this work) so that the hiding effect does not emerge anymore at the cost of a non-zero noise floor. Further, large differences get amplified.

Improvement 2 (concerning the classification step): The original Template Attack only provides a sample classification strategy based on one available sample. While this may be a realistic scenario in the context of stream ciphers[4], the situation is probably less tight in the context of block ciphers. Moreover, in case of a low-leakage implementation, one sample may not be sufficient for a reliable classification. For these reasons, a classification strategy that processes one or several samples is applied.

2.2 Stochastic Model

The Stochastic Model [7] assumes that the physical observable $I_t(x,k)$ at time t is composed of two parts, a data-dependent part $h_t(x,k)$ as a function of known data x and subkey k and a noise term R_t with zero mean: $I_t(x,k) = h_t(x,k) + R_t$. $I_t(x,k)$ and R_t are seen as stochastic variables. For this paper, we use the maximum likelihood based approach of [7] and skip the minimum principle as it is already proven to be less efficient in [7]. Profiling processes $N = N_1 + N_2$ samples representing a known subkey k and known data $x_1, x_2, \ldots, x_N$ and consists of two parts. The first part yields an approximation of $h_t(\cdot,\cdot)$, denoted as $\widetilde{h}_t^*(\cdot,\cdot)$, i.e., the data-dependent part of the side channel leakage, in a suitable u-dimensional chosen vector subspace $\mathcal{F}_{u;t}$ for each instant t. The second part then computes a multivariate density of the noise at relevant instants. For the computation of $\widetilde{h}_t^*(\cdot,\cdot)$, an overdetermined system of linear equations has to be solved for each instant t. The $(N_1 \times u)$ design matrix is made up by the representation of the outcome of a selection function combining k and x_n $(1 \leq n \leq N_1)$ in $\mathcal{F}_{u;t}$ and the corresponding N_1-dimensional vector includes the instantiations i_{t_n} of the observable. As preparation step for the computation of the multivariate density, p side channel relevant time instants have to be chosen based on $\widetilde{h}_t^*(\cdot,\cdot)$. The complementary subset of N_2 measurements is then used to compute the covariance matrix C. For this, p-dimensional noise vectors have to be extracted from all N_2 measurements at the p instants by subtracting the corresponding data-dependent part. Given the covariance matrix C, this leads to a Gaussian multivariate density $\widetilde{f}_0 \colon \mathbb{R}^p \to \mathbb{R}$.

Key extraction applies the maximum likelihood principle. Given N_3 measurements at key extraction, one decides for key hypothesis $k \in \{1, \ldots, K\}$ that maximizes

[4] Reference [9] presents an amplified attack against stream ciphers for the case of several available samples.

$$\alpha(x_1, \ldots, x_{N_3}; k) = \prod_{j=1}^{N_3} \widetilde{f}_0 \left(\boldsymbol{i_t}(x_j, k^\circ) - \widetilde{\boldsymbol{h}}_{\boldsymbol{t}}^*(x_j, k) \right) . \tag{2}$$

Herein, k° is the unknown correct key value.

Use of Stochastic Methods Against AES. We chose the vector subspace $\mathcal{F}_9$, i.e., bitwise coefficients at the S-Box outcome as selection function as suggested by [7]. The base vectors $g_l(x \oplus k)$ $(0 \leq l \leq 8)$ are

$$g_l(x \oplus k) = \left\{ \begin{array}{ll} 1 & \text{if } l = 0 \\ l\text{-th bit of S-box}(x \oplus k) & \text{if } 1 \leq l \leq 8 \end{array} \right\} . \tag{3}$$

The choice of relevant time instants is based on sosd[5]. Other parameters are kept fixed, as e.g., we use $N_1 = \frac{N}{2}$ measurements for profiling the data-dependent part and $N_2 = \frac{N}{2}$ measurements for profiling the noise throughout this paper[6].

2.3 Compendium of Differences

Table 1 summarizes the fundamental differences in the approaches of both attacks. Following the notation in [7], Templates estimate the data-dependent part h_t itself, whereas the Stochastic model approximates the linear part of h_t in the chosen vector subspace (e.g., $\mathcal{F}_9$) and is not capable of including non-linear parts. Templates build a covariance matrix for each key dependency whereas the Stochastic Model generates only one covariance matrix, hereby neglecting possible multivariate key dependent noise terms. A further drawback may be that terms of the covariance matrix are distorted because of non-linear parts of h_t in $\mathcal{F}_9$.

Table 1. Fundamental differences between Templates and the Stochastic Model

Sample portion	Template Attack	Stochastic Model
signal	estimation of key dependent signal → 256 average signals	linear approximation of key dependent signal in $\mathcal{F}_9$ → 9 sub-signals
noise	key dependent, characterized → 256 cov matrices	non-key dependent , characterized → one cov matrix

3 Performance Evaluation

In this contribution, performance aspects for side channel cryptanalysis are elaborated for the Template Attack and the Stochastic Model. Our goal is to provide a systematic performance comparison with respect to resources[7] needed for a successful attack. An adversary is *successful* if the (unknown) key value is correctly identified at classification.

[5] The Euclidean norm proposed in [7] produces very similar results.

[6] One may argue that the choice of instants can be done using all N samples.

[7] We focus on the number of available samples (side channel quality) since computational complexity is of minor importance for the attacks under consideration.

3.1 Metrics, Parameters, and Factors to Study

Hence in determining performance of side channel based techniques we first have to answer four related questions: (i) which are the relevant parameters that have an impact on attack performance, (ii) which of these parameters can be controlled resp. their influence measured and hence should be in the scope of our experiments, (iii) on which values for the remaining parameters this case study should be based, and (iv) what metrics should we select in order to best capture performance aspects?

From the standpoint of resources needed for a successful attack, parameters that influence the success rate are manifold ranging from the measurement equipment and its environment, the knowledge about the attacked implementation, the configuration of the implementation during profiling, and the concrete methodical approach used for analysis to the number of measurements in the profiling and classification steps.

Among them, we evaluate (I) the methodical approach, (II) the number of curves for profiling, and (III) the number of curves in the classification step. The remaining parameters are chosen to be identical for both methods evaluated. Because of this, we are able to exclude any measurement or implementation dependent impact on our analysis results for each setup.

We evaluate two methodical approaches as these are the Template Attack and the Stochastic Model. Concrete parameter settings of both methods additionally include the number and composition of time instants chosen for the multivariate probability density. We implemented identical point selection algorithms operating on sosd (cp. Sections 2.1 and 2.2) selecting at most one point per clock cycle. The number of measurements, both during profiling and key extraction, is regarded as the relevant and measurable parameter. Let N be the number of measurements used in the profiling step and N_3 the number of measurements used at key extraction. For both, the Template Attack and the Stochastic Model, the concrete parameter values to study are given in Section 3.2.

Profiling efficiency is measured (1) as efficiency in estimating the data-dependent sample portion (refers only to N) and (2) as ability to determine the correct set of points of interests (refers to N and p). Both metrics relate to reference values obtained for maximal N (referred to as N_{max} below) used in the concrete setting.

Metric 1: The first efficiency metric for profiling evaluates the correlation coefficient ρ of the average vectors $m_i(N)$ obtained from N samples and the reference vectors $m_i(N_{max})$: $\frac{1}{K}\sum_{i=0}^{K}\rho(m_i(N), m_i(N_{max}))$. For the Stochastic Model, we approximate the $m_i(N)$ with $\widetilde{h}_t^*(\cdot,\cdot)$ and use the reference $m_i(N_{max})$ that we assume to be the best possible estimator of the data-dependent part h_t.

Metric 2: The second metric compares the set of selected points based on N samples to the reference set obtained using N_{max} samples and returns the percentage of points that are located in the correct clock cycle.

Metric 3: Classification efficiency (refers to N_3, N and p) is measured as success rate to obtain the correct key value. The success rate at key extraction is empirically determined by classifying N_3 randomly chosen measurements out of the key extraction measurement series. This random choice is repeated one thousand times and the success rate is then defined as the percentage of success in determining the correct key value.

In Section 5 optimizations for both methods are included in the performance analysis.

3.2 Experimental Design

The performance analysis is applied to two experimental units performing AES in software without any countermeasures. Our first experimental unit (device A) is an ATM163 microcontroller. A set of more than 230,000 power measurements was recorded for profiling purposes with a fixed AES key and randomly chosen plaintexts. For classification purposes, we recorded a second set comprising 3000 measurements with a different fixed AES key. The experimental design is full factorial. Our second experimental unit is another 8-bit microcontroller from a different manufacturer (device B). Furthermore, the power measurements of device B stem from a different, low-noise, measurement setup. We obtained a set of 50,000 power measurements for profiling purposes and a classification set of 100 power measurements, both with fixed but different AES keys. Table 2 shows all concrete parameter values we studied. However, Sections 4 and 5 only provide the most relevant results.

Table 2. Concrete parameter values to study

Device	Parameter	Parameter Values
A	N	231k, 50k, 40k, 30k, 25k, 20k, 10k, 5k, 2k[8], 1k[8], 200[8]
A	p	3, 6, 9, x[9]
A	N_3	1, 2, 5, 10
B	N	50k[10], 10k, 5k, 500[8], 100[8]
B	p	x[9]
B	N_3	1, 2, 5

4 Experimental Evaluation: Results for Original Attacks

4.1 Comparison of Profiling Efficiency

Profiling metrics 1 and 2 are summarized in Fig. 1 and Table 3. Metric 1 clearly yields enhanced results for Templates which is reasonable as the Stochastic Model uses only half of the measurements for the determination of the data-dependent part. Though less efficient in determining the data-dependent part,

[8] Stochastic Model only.

[9] x = maximum number identified after profiling.

[10] Template Attack only.

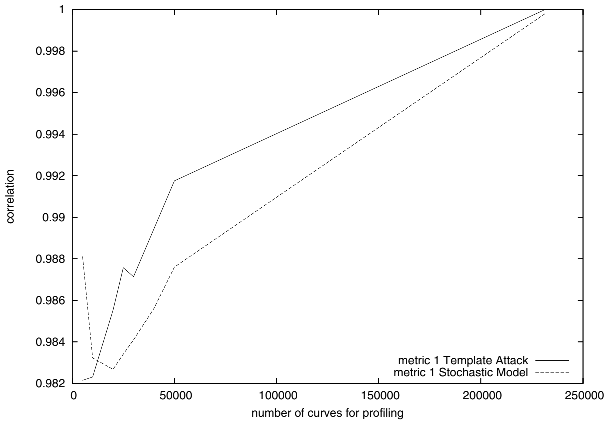

Fig. 1. Metric 1 for device A

Table 3. Metric 2 for device A as function of N

	231k	50k	40k	30k	25k	20k	10k	5k
Template Attack	1	0.89	0.89	0.78	0.67	0.56	0.23	0.23
Stochastic Model	1	1	1	1	1	1	0.67	0.78

Table 3 clearly indicates the superiority of the Stochastic Model in terms of selecting the right points in time.

4.2 Comparison of Classification Efficiency

We compare the success rates for variations of N, $N_3 \in \{1, 10\}$ and the optimal number of selected instants to maximize the success rates. Fig. 2 shows metric 3 plotted as function of these parameters. One can observe, that each pair of plots intersects at least once. Hence, a general statement on which attack yields better success rates is not feasible as this depends on the number of curves that are available in the profiling step. If a large number of samples is available (e.g., more than twenty thousand), the Template Attack yields higher success rates. If only a small number of samples is available (e.g., less than twenty thousand), stochastic methods are the better choice.

4.3 Weaknesses and Strengths

Template Attack The strength of the Template Attack is, that it extracts far more information from the samples than the Stochastic Model. Given sufficient samples in the profiling step, it is clearly superior to the Stochastic model in the classification step, due to the precise estimation of the average signal and the use of 256 covariance matrices. On the other hand, it requires much more

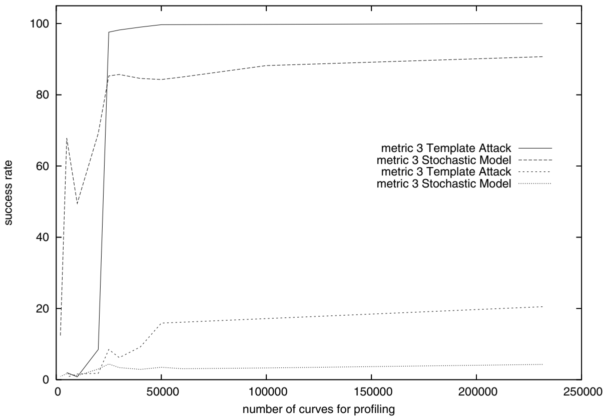

Fig. 2. Metric 3 for device A, $N_3 = 10$ for upper and $N_3 = 1$ for lower curves

samples than stochastic methods to reduce the noise in the side channel and to select correct instants (see Table 3).

Stochastic Model. The Stochastic Model's strength is the ability to "learn" quickly from a small number of samples. One weakness lies in the reduced precision due to the linear approximation in a vector subspace. A second weakness is the usage of only a single covariance matrix. If the approximation of the data-dependent part is not precise enough, errors in the approximation affect the remaining "noise".

5 Experimental Evaluation: Optimized Results

The maximum efficiency achievable at key extraction for each method is of high importance, so that we carried out optimizations for each method. Particularly, Section 4 reveals that the point selection algorithm is crucial for the key extraction efficiency. Both, for Templates and the Stochastic Model, we evaluate the statistical t-distribution as the basis of instant selection in this Section. For the Stochastic Model, the choice of the vector subspace (single intermediate result vs. two intermediate results) is studied additionally.

Template Attack with T-Test. The Template Attack's weakness is its poor ability to reduce the noise in the side channel samples if the adversary is bounded in the number of samples in the profiling step. For small N, the remaining noise distorts the sosd curve, which we used as the basis for the selection of interesting points so far.

The T-Test is a standard statistical tool to meet the challenge of distinguishing noisy signals. When computing the significant difference of two sets (i, j), it does not only consider the distance of their means m_i, m_j but as well their variability (σ_i^2, σ_j^2) in relation to the number of samples (n_i, n_j). We modified our implementation to compute the sum of squared pairwise t-differences (also referred to as *sost* in this work)

$$\sum_{i,j=1}^{K}\left(\frac{m_i-m_j}{\sqrt{\frac{\sigma_i^2}{n_i}+\frac{\sigma_j^2}{n_j}}}\right)^2 \quad \text{for } i \geq j$$

as basis for the point selection instead of sosd. Fig. 3 illustrates the striking difference between sosd and sost for $N = 50000$ and 10000 samples. The scale of the vertical axis is not the same for all plots, but as one is not interested in comparing the absolute height of the peaks, this can be disregarded. What is important is the relative distance between the peaks and the noise floor in each curve. While the reduction of N by a factor 5 leads to a very distorted sosd signal, the significance of sost in terms of where to find interesting points does not change. Apart from the different scale, the peaks have a virtually identical shape.

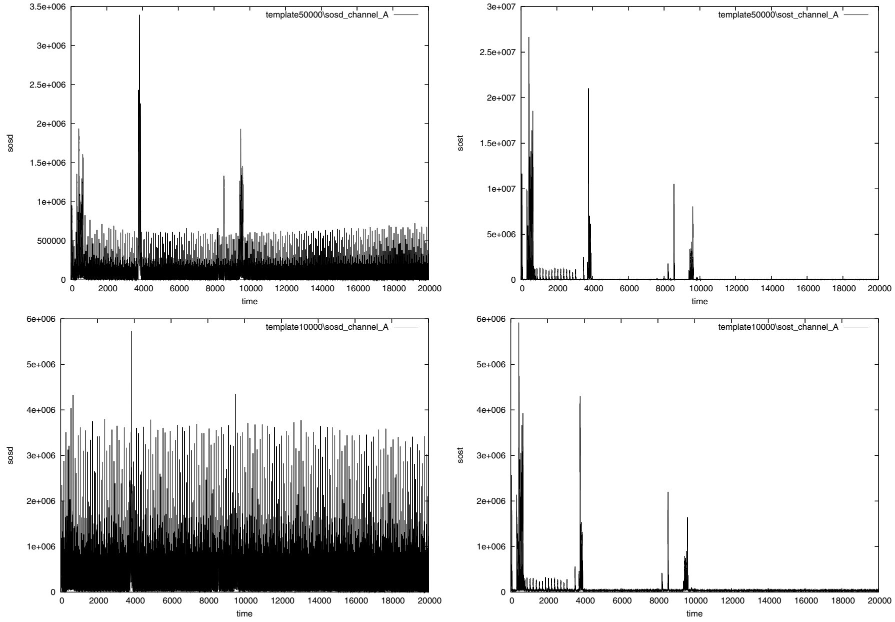

Fig. 3. sosd (left) and sost (right) as functions of time, $N = 50000$ (top) and 10000 (bottom)

High-Order Stochastic Model with F_{17} and T-Test. According to the improvements for Templates, we apply a slightly modified sost for the use with stochastic methods. Here, the data-dependent approximators $\widetilde{h}_t^*(\cdot,\cdot)$ and the empirical variance σ^2 derived from N_1 measurements are used in the computation. As for Templates, we observe a significant improvement of the point selection performance.

The weakness of the Stochastic Model with $\mathcal{F}_9$ is the limited precision due to the approximation of the data-dependent sample portion. An obvious solution to this problem is to increase the number of dimensions of the vector subspace in order to generate a more precise approximator at the cost of needing more samples in the profiling step (trade off problem). But as the authors of [7] already analyzed several high-dimensional vector subspaces and concluded that $\mathcal{F}_9$ seems to be most efficient, we decide to follow a different attempt.

Our approach arises from comparing the sosd curves of the Stochastic Model and the Template Attack. Due to the fact that the underlying samples represent only one fixed key, the Template Attack's sosd curve shows peaks for x, $x \oplus k$, and $\text{Sbox}(x \oplus k)$. Since the Stochastic Model only approximates the data-dependent sample portion at $\text{Sbox}(x \oplus k)$, it can not track bits "through" the Sbox and hence the point selection algorithm only finds instants for $\text{Sbox}(x \oplus k)$. Our approach aims at the fact that the Stochastic Model "overlooks" instants covering the Sbox lookup which yield the strongest peaks in the sosd curve of the Template Attack. We increase the number of dimensions of the vector subspace, but rather than increasing the level of detail at one intermediate result of the AES encryption, we add consideration of a second intermediate result. We (re-)define the selection functions g_l of the 17-dimensional vector subspace $\mathcal{F}_{17}$ as follows:

$$g_l(x \oplus k) = \left\{ \begin{array}{ll} 1 & \text{if } l = 0 \\ l\text{-th bit of S-box}(x \oplus k) & \text{if } 1 \leq l \leq 8 \\ (l-8)\text{-th bit of } x \oplus k & \text{if } 9 \leq l \leq 16 \end{array} \right\} . \tag{4}$$

As desired, additional clear peaks during the Sbox lookup $(x \oplus k)$ were found by the point selection algorithm.

5.1 Comparison Templates vs. T-Test Based Templates

When comparing the optimized Template Attack with the original attack, we evaluate the basis on which the point selection algorithm operates.

Profiling Efficiency

Table 4 shows the efficiency of both attacks in the profiling step using metric 2. The numbers clearly indicate the superiority of the improved version, the T-Test Template Attack, in terms of selecting the right instants and hence, in the profiling step. Considering Fig. 3 again, the improved profiling efficiency obviously derives from the enhanced ability to suppress noise in the side channel.

Classification Efficiency

In the following, we compare the classification success rates of the attacks in Fig. 4. We restrict our attention to variations of N, $N_3 \in \{1, 10\}$ for the sake of clarity, and, each time, the optimal number of selected instants to maximize the

Table 4. Metric 2 for device A as function of N

	231k	50k	40k	30k	20k	10k	5k
Template Attack	1	0.89	0.89	0.78	0.56	0.23	0.23
T-Test Templates	1	1	1	1	1	1	1

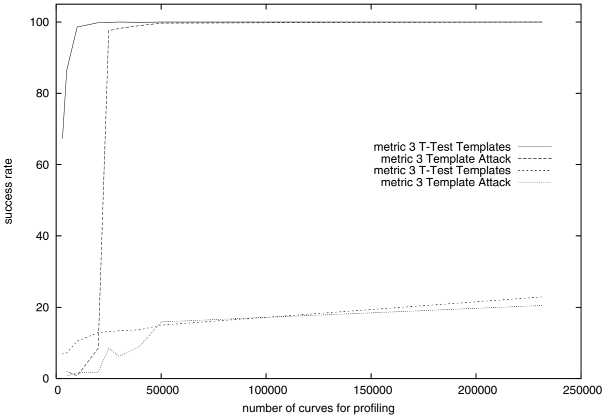

Fig. 4. Metric 3 for device A, $N_3 = 10$ for upper and $N_3 = 1$ for lower curves

success rates. For small N, e.g., N smaller than thirty thousand, the improved profiling of the optimized attack clearly leads to a higher success rate at classification.

5.2 Comparison First-Order Stochastic Model vs. T-Test Based High-Order Stochastic Model

When comparing the optimized Stochastic Model with the original attack, we evaluate the choice of the vector sub-space and the T-Test based point selection.

Profiling Efficiency
Table 5 shows the profiling efficiency of both attacks in metric 2. The numbers indicate the improved attack's advanced ability to select the right points, in particular when processing only a small number of profiling measurements.

Table 5. Metric 2 for device A as function of N

	231k	50k	40k	30k	25k	20k	10k	5k	2k	1k	200
Stochastic Model	1	1	1	1	1	1	0.67	0.78	0.67	-	-
T-Test based Stochastic Model	1	1	1	1	1	1	1	0.9	1	1	0.5

Classification Efficiency
In the following, we compare the classification success rates of both attacks. We restrict our attention to variations of N, $N_3 \in \{1, 10\}$, and, each time, the

optimal number of selected instants to maximize the success rates. Fig. 5 shows metric 3 plotted as function of these parameters.

The benefit of generating eight additional base vectors with respect to the Sbox input and using sost instead of sosd is clearly visible. Following the profiling efficiency (cp. Table 5), the efficiency in the classification step is significantly increased. Particularly, for N larger than thirty thousand and $N_3 = 10$, the T-Test based high-order Stochastic Model clearly exceeds the 90% success rate "boundary" and finally reaches 100% success.

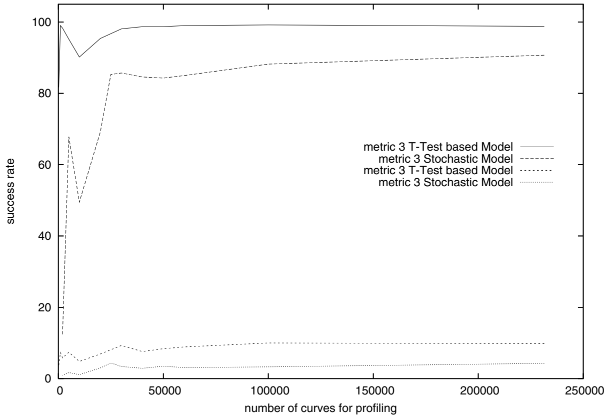

Fig. 5. Metric 3 for device A, $N_3 = 10$ for upper and $N_3 = 1$ for lower curves

5.3 Overall Comparison

In this Section we illustrate the efficiency of the improved methods in the classification step and give a short summary of the observations. We provide them to give an overall survey of our work. Fig. 6 contrasts the classification efficiency of the attacks using metric 3.

The T-Test Template Attack is the best possible choice in almost all parameter ranges. For small N (e.g., N less than five thousand), the T-Test based high-order Stochastic Model leads to better results. We would like to point out that the improved version of the Stochastic Model still operates successfully using extremely small N. For example, using $N = 200$ profiling measurements and $N_3 = 10$ curves for classification it still achieves a success rate of 81.7%.

To stress the impact of the factor "measurement quality" we present success rates of the improved attacks for measurements of device B that stem from the low-noise setup. Table 6 provides the attack efficiencies in metric 3 for variations of N, $N_3 \in \{1, 5\}$, and, each time, the optimal number of selected instants to maximize the success rates.

Besides the fact that the relation of N to success rate of both attacks is better by orders of magnitude when using low-noise measurements, we would like to point out, that the improved Stochastic Model still classifies keys successfully,

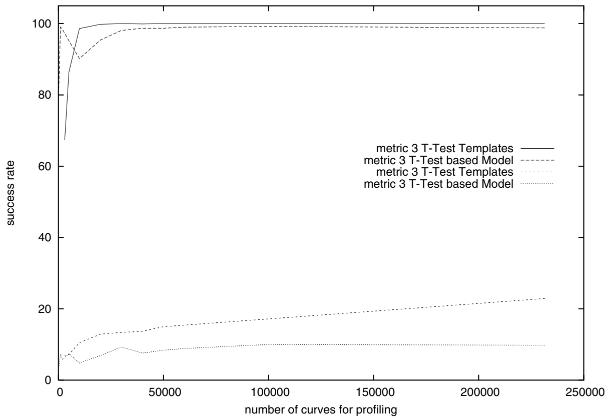

Fig. 6. Metric 3 for device A, $N_3 = 10$ for upper and $N_3 = 1$ for lower curves

Table 6. Metric 3 for device B as function of N

		50k	10k	5k	500	100
T-Test Templates	$N_3 = 1$	94.8	93.0	88.2	-	-
	$N_3 = 5$	100.0	100.0	100.0	-	-
T-Test based Stochastic Model	$N_3 = 1$	-	57.5	60.1	46.8	27.1
	$N_3 = 5$	-	100.0	99.9	100.0	96.5

even if the profiling has been done with as little as $N = 100$ curves, which is far less than the number of subkey hypotheses.

6 Conclusion

In this contribution, an experimental performance analysis is applied to the Template Attack and the Stochastic Model. We concentrate on measurable parameter settings such as the number of curves during profiling and classification. By using the originally proposed attacks, it was revealed that towards a low number of profiling measurements stochastic methods are more efficient whereas towards a high number of profiling samples Templates achieve superior performance results. For improvements, we introduce T-Test based Templates and give experimental results for the use of high-order stochastic methods in combination with a T-Test based choice of instants. It is shown that the improved variants are indeed practical, even at a low number of profiling measurements[11]. As a main result, T-Test based Templates are generally the method of choice. However, in

[11] This is of particular importance when applying these attacks to noisy EM samples. We experimentally proved that the T-Test based attacks yield far better results than the original attacks in such a setting.

case of a low number of samples for profiling, stochastic methods can still turn out to be more efficient.

References

1. D. Agrawal, J.R. Rao, P. Rohatgi: Multi-Channel Attacks. In: C.D. Walter, Ç.K. Koç, C. Paar (eds.): Cryptographic Hardware and Embedded Systems — CHES 2003, Springer, LNCS 2779, 2003, 2–16.
2. E. Brier, C. Clavier, F. Olivier: Correlation Power Analysis with a Leakage Model. In: M. Joye and J.-J. Quisquater (eds.): Cryptographic Hardware and Embedded Systems — CHES 2004, Springer, LNCS 3156, 2004, 16-29.
3. S. Chari, J.R. Rao, P. Rohatgi: Template Attacks. In: B.S. Kaliski Jr., Ç.K. Koç, C. Paar (eds.): Cryptographic Hardware and Embedded Systems — CHES 2002, Springer, LNCS 2523, 2003, 13–28.
4. P.N. Fahn, P.K. Pearson: IPA: A New Class of Power Attacks. In: Ç.K. Koç and C. Paar: Cryptographic Hardware and Embedded Systems — CHES 1999, Springer, LNCS 1717, 1999, 173–186.
5. K. Gandolfi, C. Mourtel, F. Olivier: Electromagnetic Analysis: Concrete Results. In: Ç Koç, D. Naccache, C. Paar (eds.): Cryptographic Hardware and Embedded Systems — CHES 2001, Springer, LNCS 2162, 2001, 251–261.
6. P.C. Kocher, J. Jaffe, B. Jun: Differential Power Analysis. In: M. Wiener (ed.): Advances in Cryptology — CRYPTO '99, Springer, LNCS 1666, 1999, 388–397.
7. W. Schindler, K. Lemke, C. Paar: A Stochastic Model for Differential Side Channel Cryptanalysis. In: J.R. Rao, B. Sunar (eds.): Cryptographic Hardware and Embedded Systems — CHES 2005, Springer, LNCS 3659, 2005, 30–46.
8. W.H. Press, S.A. Teukolsky, W.T. Vetterling, B.P. Flannery: Numerical Recipes in C — The Art of Scientific Computing. Second Edition, Cambridge University Press, 1992.
9. C. Rechberger, Side Channel Analysis of Stream Ciphers, Master Thesis, Technical University Graz, 2004
10. Trochim, William M., The Research Methods Knowledge Base, 2nd Edition, http://trochim.human.cornell.edu/kb/index.htm, January 16 2005