Towards Security Limits in Side-Channel Attacks
(With an Application to Block Ciphers)

F.-X. Standaert*, E. Peeters, C. Archambeau, J.-J. Quisquater

UCL Crypto Group, Place du Levant 3, B-1348 Louvain-la-Neuve, Belgium

e-mails: fstandae,peeters,archambeau,quisquater@dice.ucl.ac.be

Abstract. In this paper, we consider a recently introduced framework that investigates physically observable implementations from a theoretical point of view. The model allows quantifying the effect of practically relevant leakage functions with a combination of security and information theoretic metrics. More specifically, we apply our evaluation methodology to an exemplary block cipher. We first consider a Hamming weight leakage function and evaluate the efficiency of two commonly investigated countermeasures, namely noise addition and masking. Then, we show that the proposed methodology allows capturing certain non-trivial intuitions, *e.g.* about the respective effectiveness of these countermeasures. Finally, we justify the need of combined metrics for the evaluation, comparison and understanding of side-channel attacks.

1 Introduction

In [14], a formal practice-oriented model for the analysis of cryptographic primitives against side-channel attacks was introduced as a specialization of Micali and Reyzin's "physically observable cryptography" paradigm [8]. The model is based on an theoretical framework in which the effect of practically relevant leakage functions is evaluated with a combination of security and information theoretic measurements. A central objective of the model was to provide a fair evaluation methodology for side-channel attacks. This objective is motivated by the fact that side-channel attacks may take advantage of different statistical tools (*e.g.* difference of means [5], correlation [2], Bayesian classification [1], stochastic models [13]) and are therefore not straightforward to compare. Additionally to the comparisons of side-channel attacks, a more theoretical goal was the understanding of the underlying mechanisms of physically observable cryptography.

Specifically, [14] suggests to combine the average success rate of a (well specified) adversary with some information theoretic metrics in order to capture the intuition summarized in Figure 1. Namely, an information theoretic metric should measure the average amount of information that is available in some physical observations while a security metric measures how efficiently an actual adversary can turn this information into a successful key recovery.

* François-Xavier Standaert is a post doctoral researcher funded by the FNRS (Funds for National Scientific Research, Belgium).

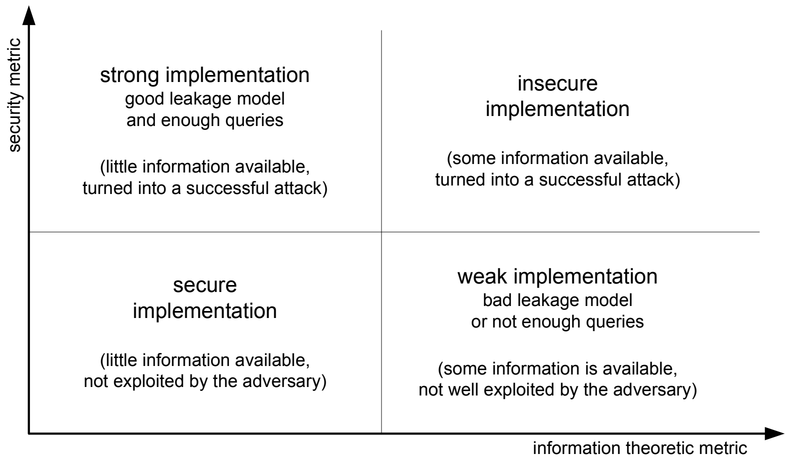


Fig. 1: Summary of side-channel evaluation criteria.

In this paper, we consequently study the relevance of the suggested methodology, by the analysis of a practical case. For this purpose, we investigate an exemplary block cipher and consider a Hamming weight leakage function in different attack scenarios. First, we consider an unprotected implementation and evaluate the information leakages resulting from various number of Hamming weight queries. We discuss how actual block cipher components compare to random oracles with respect to side-channel leakages. Then, we evaluate the security of two commonly admitted countermeasures against side-channel attacks, *i.e.* noise addition and masking. Through these experiments, we show that the proposed evaluation criteria allows capturing certain non-trivial intuitions about the respective effectiveness of these countermeasures. Finally, we provide some experimental validations of our analysis and discuss the advantages of our combination of metrics with respect to other evaluation techniques.

Importantly, in our theoretical framework, side-channel analysis can be viewed as a classification problem. Our results consequently tend to estimate the security limits of side-channel adversaries with two respects. First, because of our information theoretic approach, we aim to evaluate precisely the average amount of information that is available in some physical observations. Second, because we consider (one of) the most efficient classification test(s), namely Bayesian classification, it is expected that the computed success rates also correspond to the best possible adversarial strategy. However, we mention that the best evaluation and comparison metrics to use in the context of side-channel attacks are still under discussion. Our results intend to show that both are useful, but other similar metrics should still be investigated and compared.

2 Model specifications

In general, the model of computation we consider in this paper is the one initially presented in [8] with the specializations introduced in [14]. In this section, we first describe our target block cipher implementation. Then, we specify the leakage function, the adversarial context and adversarial strategy that we consider in this work. Finally, we provide the definitions of our security and information theoretic metrics for the evaluation of the attacks in the next sections. Both the adversarial classifications and the metrics were introduced and detailed in [14].

2.1 Target implementation

Our target block cipher implementation is represented in Figure 2. For convenience, we only represent the combination of a bitwise key addition and a layer of substitution boxes. We make a distinction between a *single block* and a *multiple block* implementation. This difference refers to the way the key guess is performed by the adversary. In a single block implementation (*e.g.* typically, an 8-bit processor), the adversary is able to guess (and therefore exploit) all the bits in the implementation. In a multiple block implementation (*e.g.* typically, a hardware implementation with data processed in parallel), the adversary is only able to guess the bits at the output of one block of the target design. That is, the other blocks are producing what is frequently referred to as algorithmic noise.

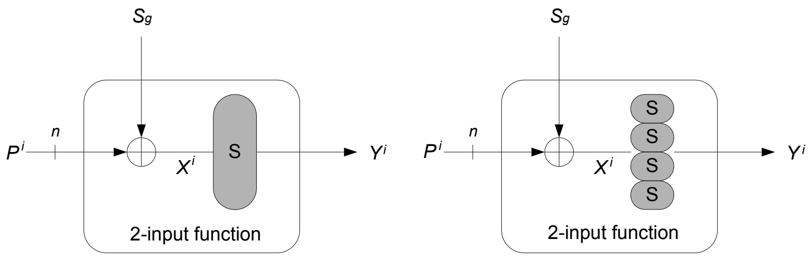


Fig. 2: Single block and multiple block cipher implementation.

2.2 Leakage function

Our results consider the example of a Hamming weight leakage function. Specifically, we assume a side-channel adversary that is provided with the (possibly noisy) Hamming weight leakages of the S-boxes outputs in Figure 2, *i.e.* $W_H(Y^i)$. With respect to the classification introduced in [14], perfect Hamming weights correspond to non-profiled leakage functions while noisy Hamming weights relate to the context of device profiled (stochastic) leakage functions. In the latter one, the leakage function includes a characterization of the noise in the target device. For this purpose, we assume a Gaussian noise distribution. We note also that our exemplary leakage functions are univariate since they only consider one leaking point in the implementations, namely the S-boxes outputs.

2.3 Adversarial context

We consider a non-adaptive known plaintext adversary that can perform an arbitrary number of side-channel queries to the target implementation of Figure 2 but cannot choose its queries in function of the previously observed leakages.

2.4 Adversarial strategy

We consider a side-channel key recovery adversary with the following (hard) strategy: "*given some physical observations and a resulting classification of key candidates, select the best classified key only*".

2.5 Security metric: average success rate of the adversary

The success rate of a side-channel key recovery attack can be written as follows. Let S and O be two random variables in the discrete domains $\mathcal{S}$ and $\mathcal{O}$, respectively denoting the target secret signals and the side-channel observations. Let $O^i_{S_g}$ be an observation generated by a secret signal S_g. Let finally $\mathbf{C}(\mathcal{L}(S), O^i_{S_g})$ be the statistical tool used by the adversary to compare an actual observation of a device with its prediction according to a leakage function $\mathcal{L}$[1]. This statistical tool could be a difference of mean test, a correlation test, a Bayesian classification, or any other tool, possibly inspired from classical cryptanalysis. For each observation $O^i_{S_g}$, we define the set of keys selected by the adversary as:

$$M^i_{S_g} = \{\hat{s} \mid \hat{s} = \underset{S}{argmax}\, \mathbf{C}[\mathcal{L}(S)|O^i_{S_g}]\}$$

Then, we define the result of the attack with the index matrix:

$$\mathrm{I}^i_{S_g,S} = \qquad \frac{1}{|M^i_{S_g}|} \textbf{ if } S \in M^i_{S_g}, \qquad \textbf{else } 0.$$

The success rate of the adversary for a secret signal S_g is estimated as:

$$\mathbf{S_R}(S_g) = \underset{O^i_{S_g}}{\mathbf{E}}\ \mathrm{I}_{S_g,S_g}, \tag{1}$$

and the average success rate of the adversary is defined as:

$$\overline{\mathbf{S_R}} = \underset{S_g}{\mathbf{E}}\ \underset{O^i_{S_g}}{\mathbf{E}}\ \mathrm{I}_{S_g,S_g} \tag{2}$$

In the following, we will *only* consider a Bayesian classifier, *i.e.* an adversary that selects the keys such that $\mathbf{P}[S|O^i_{S_g}]$ is maximum, since it corresponds to (one of) the most efficient way(s) to perform a side-channel key recovery.

Finally, it is interesting to remark that one can use the complete index matrix to build a confusion matrix $\mathrm{C}_{S_g,S} = \mathbf{E}_{O^i_{S_g}}\ \mathrm{I}_{S_g,S}$. The previously defined average success rate simply corresponds to the averaged diagonal of this matrix.

[1] In our following examples, $\mathcal{L}$ is the Hamming weight function.

2.6 Information theoretic metric: conditional entropy

In addition to the average success rate, [14] suggests the use of an information theoretic metric to evaluate the information contained in side-channel observations. We note (again) that different proposals could be used for such evaluation purposes and their comparison is a scope for further research. In the present paper, we selected the classical notion of Shannon conditional entropy and investigate how one can take advantage of the approach to understand and evaluate side-channel attacks. Let $\mathbf{P}[S|O^i_{S_g}]$ be the probability vector of the different key candidates S given an observation $O^i_{S_g}$ generated by a correct key S_g. Similarly to the confusion matrix of the previous section, we define a probability matrix: $\mathrm{P}_{S_g,S} = \mathbf{E}_{O^i_{S_g}}\ \mathbf{P}[S|O^i_{S_g}]$ and an entropy matrix $\mathrm{H}_{S_g,S} = \mathbf{E}_{O^i_{S_g}}\ -\log_2 \mathbf{P}[S|O^i_{S_g}]$. Then, we define the average probability of the correct key as:

$$\overline{\mathbf{P}[S_g|O_{S_g}]} = \mathop{\mathbf{E}}_{S_g}\ \mathrm{P}_{S_g,S_g} \tag{3}$$

And the conditional entropy:

$$\mathbf{H}[S_g|O_{S_g}] = \mathop{\mathbf{E}}_{S_g}\ \mathrm{H}_{S_g,S_g} \tag{4}$$

We note that this definition is equivalent to Shannon conditional entropy[2]. We simply used the previous notation because it is convenient to compute the probability (or entropy) matrices. For example, it allows to detect a good leakage function, *i.e.* a leakage function such that $\max_S \mathrm{H}_{S_g,S} = \mathrm{H}_{S_g,S_g}$. In the following, the leakages will be quantified as conditional entropy reductions that corresponds to the mutual information $\mathbf{I}[S_g; O_{S_g}] = \mathbf{H}[S_g] - \mathbf{H}[S_g|O_{S_g}]$.

It is important to observe that the average success rate fundamentally describes an adversary. In general, it has to be computed for different number of queries in order to evaluate how much observations are required to perform a successful attack. By contrast, the information theoretic measurement says nothing about the actual security of an implementation but characterizes the leakage function, independently of the number of queries.

3 Investigation of single leakages

In this section, we analyze a situation where an adversary is provided with the observation of one single Hamming weight leakage. First, we evaluate single block implementations. Then, we discuss multiple block implementations and key guesses. Finally, we evaluate the effect of noise addition in this context.

[2] Since: $\mathbf{H}[S_g|O] = \mathbf{E}_{O^i}\ \mathbf{E}_{S_g}\ \mathbf{H}[S_g|O^i]$

$$\begin{aligned}
&= \textstyle\sum_{O^i} \mathbf{P}[O^i] \sum_{S_g} \mathbf{P}[S_g|O^i] \cdot -\log_2(\mathbf{P}[S_g|O^i]) \\
&= \textstyle\sum_{O^i} \mathbf{P}[O^i] \sum_{S_g} \frac{\mathbf{P}[O^i|S_g]\cdot\mathbf{P}[S_g]}{\mathbf{P}[O^i]} \cdot -\log_2(\mathbf{P}[S_g|O^i]) \\
&= \textstyle\sum_{O^i} \sum_{S_g} \mathbf{P}[O^i|S_g] \cdot \mathbf{P}[S_g] \cdot -\log_2(\mathbf{P}[S_g|O^i]) \\
&= \textstyle\sum_{S_g} \sum_{O^i} \mathbf{P}[O^i|S_g] \cdot \mathbf{P}[S_g] \cdot -\log_2(\mathbf{P}[S_g|O^i]) \\
&= \textstyle\sum_{S_g} \mathbf{P}[S_g] \sum_{O^i} \mathbf{P}[O^i|S_g] \cdot -\log_2(\mathbf{P}[S_g|O^i]) = \mathbf{E}_{S_g}\ \mathrm{H}_{S_g,S_g}
\end{aligned}$$

3.1 Single block implementations

Let us assume the following situation: we have an n-bit secret key S_g and an adversary is provided with the leakage corresponding to a computation $Y^i = f(S_g, P^i) = \mathrm{S}(P^i \oplus S_g)$. That is, it obtains an observation $O^i_{S_g} = W_H(Y^i)$ and we assume a single block implementation as the one in the left part of Figure 2. Therefore, the adversary can potentially observe the $n+1$ Hamming weights of Y^i. Since the Hamming weights of a random value are distributed as binomials, one can easily evaluate the average success rate of the adversary as:

$$\overline{\mathbf{S_R}} = \mathop{\mathbf{E}}_{S_g} \mathop{\mathbf{E}}_{O^i_{S_g}} \mathrm{I}_{S_g, S_g} = \sum_{h=0}^{n} \frac{\binom{n}{h}}{2^n} \cdot \frac{1}{\binom{n}{h}} = \frac{n+1}{2^n} \tag{5}$$

This equation means that on average, obtaining the Hamming weight of a secret n-bit value increases the success rate of a key-recovery adversary from $\frac{1}{2^n}$ to $\frac{n+1}{2^n}$. Similar evaluations will be performed for the conditional entropy in Section 3.3.

3.2 Multiple blocks and key guesses

Let us now assume a situation similar to the previous one, but the adversary tries to target a multiple block implementation. Therefore, it is provided with the Hamming weight of an n-bit secret value of which it can only guess b bits, typically corresponding to one block of the implementation. Such a key guess situation can be analyzed by considering the un-exploited bits as a source of algorithmic noise approximated with a Gaussian distribution. This will be done in the next section. The quality of this estimation will then be demonstrated in Section 5, by relaxing the Gaussian estimation.

3.3 Noise addition

Noise is a central issue in side-channel attacks and more generally in any signal processing application. In our specific context, various types of noise are usually considered, including physical noise (*i.e.* produced by the environment), measurement noise (*i.e.* caused by the sampling process and tools), model matching noise (*i.e.* meaning that the leakage function used to attack does possibly not perfectly fit to real observations) or algorithmic noise (*i.e.* produced by the untargeted values in an implementation). All these disturbances similarly affect the efficiency of a side-channel attack and their consequence is that the information delivered by a single leakage point is reduced. For this reason, a usually accepted method to evaluate the effect of noise is to assume that *there is an additive effect between all the noise sources and their overall effect can be quantified by a Gaussian distribution.* We note that this assumption may not be perfectly verified in practice and that better noise models may allow to improve the efficiency of side-channel attacks. However, this assumption is reasonable in a number of contexts and particularly convenient for a first investigation.

In our experiments, we will consequently assume that the leakage function is affected by some Gaussian noise such that the physical observations are represented by a variable: $O^i_{S_g} = W_H(Y^i) + N(0, \sigma^2)$. It is then possible to estimate the average success rate of the adversary and the conditional entropy as follows:

$$\overline{\mathbf{S_R}} = \mathop{\mathbf{E}}_{S_g} \mathop{\mathbf{E}}_{O^i_{S_g}} \mathrm{I}_{S_g,S_g} = \sum_{h=0}^{n} \frac{\binom{n}{h}}{2^n} \cdot \int_{-\infty}^{+\infty} \mathbf{P}[O_{S_g}|h] \cdot \mathrm{I}_{S_g,S_g} \, do, \tag{6}$$

$$\mathbf{H}[S_g|O_{S_g}] = \mathop{\mathbf{E}}_{S_g} \mathrm{H}_{S_g,S_g} = \sum_{h=0}^{n} \frac{\binom{n}{h}}{2^n} \cdot \int_{-\infty}^{+\infty} \mathbf{P}[O_{S_g}|h] \cdot -\log_2(\mathbf{P}[S_g|O_{S_g}]) \, do, \tag{7}$$

where $\mathbf{P}[O_{S_g} = o|W_H(Y^i) = h] = \frac{1}{\sigma\sqrt{2\pi}} \exp^{\frac{-(o-h)^2}{2\sigma^2}}$ and the a posteriori probability $\mathbf{P}[S_g|O_{S_g}]$ can be computed thanks to Bayes's formula: $\mathbf{P}[S_g|O_{S_g}] = \frac{\mathbf{P}[O_{S_g}|S_g] \cdot \mathbf{P}[S_g]}{\mathbf{P}[O_{S_g}]}$, with $\mathbf{P}[O_{s_g}] = \sum_S \mathbf{P}[O_{S_g}|S] \cdot \mathbf{P}[S]$. As an illustration, the average success rate and the mutual information are represented in Figure 3 for an 8-bit value, in function of the observation signal-to-noise ratio (SNR=$10 \cdot \log_{10}(\frac{\varepsilon^2}{\sigma^2})$, where ε and σ respectively denote the standard deviation of the signal and the noise emanated from the implementation).

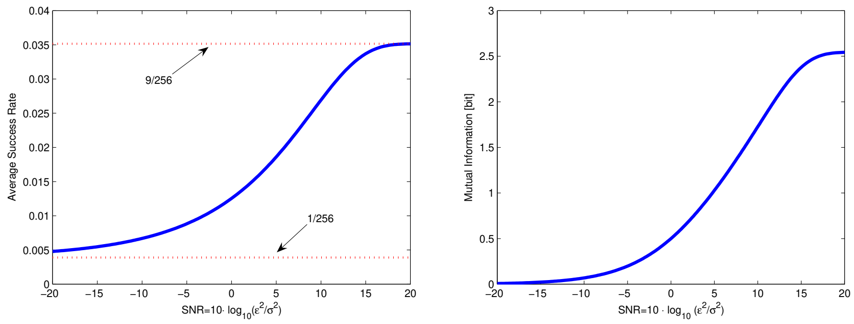


Fig. 3: Average success rate and mutual information in function of the SNR.

Note that the average success rate starts at 9/256, *i.e.* the noise-free value computed with Equation (5) and tends to 1/256 which basically means that very little information can be retrieved from the leakage. The figures also shows the correlation between the information available and the resulting success rate.

4 Investigation of multiple leakages

In the previous section, we analyzed a situation in which an adversary performs one single query to a leaking implementation and evaluated the resulting average success rate and mutual information. However, looking at Figure 3, it is clear that such a context involves limited success rates, even in case of high SNRs. As a matter of fact, actual adversaries would not only perform one single query to the target device but multiple ones, in order to increase their success rates. This section consequently studies the problem of multiple leakages.

For this purpose, let us consider the following situation: we have an n-bit secret key S_g and an adversary is provided with the leakages corresponding to two computations $Y^1 = f(S_g, P^1)$ and $Y^2 = f(S_g, P^2)$. That is, it obtains $W_H(Y^1)$ and $W_H(Y^2)$ and we would like to evaluate the average predictability of S_g. The

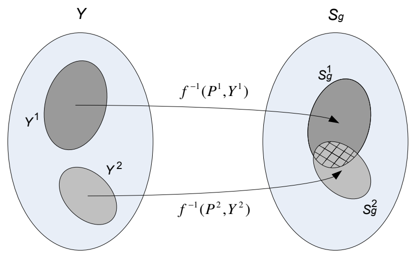


Fig. 4: Multiple point leakages.

consequence of such an experiment (illustrated in Figure 4) is that the key will be contained in the intersection of two sets of candidates obtained by inverting the 2-input functions $Y^1 = f(S_g, P_1)$ and $Y^2 = f(S_g, P_2)$. The aim of our analysis is therefore to determine how the keys within this intersection are distributed. Importantly, and contrary to the single query context, this analysis requires to characterize the cryptographic functions used in the target implementation, since they will determine how the intersection between the sets of candidates behaves. Therefore, we will consider two possible models for these functions.

4.1 Assuming random S-boxes

A first (approximated) solution is to consider the functions $f^{-1}(P^i, Y^i)$ to behave randomly. As a consequence, each observed Hamming weight leakage $h_i = W_H(Y^i)$ will give rise to a uniform list of candidates for the key S_g of size $n_i = \binom{n}{h_i}$, without any particular dependencies between these sets but the key. Let us denote the size of the set containing S_g after the observation of q leakages respectively giving rise to these uniform lists of n_i candidates by a random variable $I_q(n_1, n_2, \ldots, n_q)$. From the probability density function of I_q (given in appendix A), it is straightforward to extend the single leakage analysis of Section 3.1 to multiple leakages. The average success rate can be expressed as:

$$\overline{\mathbf{S_R}} = \sum_{h_1=0}^{n} \sum_{h_2=0}^{n} \cdots \sum_{h_q=0}^{n} \frac{\binom{n}{h_1}}{2^n} \cdot \frac{\binom{n}{h_2}}{2^n} \cdots \frac{\binom{n}{h_q}}{2^n} \cdot \sum_i \mathbf{P}[I_q = i] \cdot \frac{1}{i} \qquad (8)$$

4.2 Using real block cipher components

In order to validate the previous theoretical predictions of the average success rate, we performed the experiments illustrated in Figure 5. In the first (upper) experiment, we generated a number of plaintexts, observed the outputs of the function $f = \mathrm{S}(P^i \oplus S_g)$ through its Hamming weights $W_H(Y^i)$, derived lists of n_i candidates for Y^i corresponding to these Hamming weights and went through the inverted function $f^{-1}(P^i, Y^i)$ to obtain lists of key candidates. In the second (lower) experiment, a similar procedure is applied but the n_i key candidates were selected from random lists (including the correct key). As a matter of fact, the first experiment corresponds to a side-channel attack against a real block cipher (we used the AES Rijndael S-box) while the second experiment emulates the previous random S-box estimation.

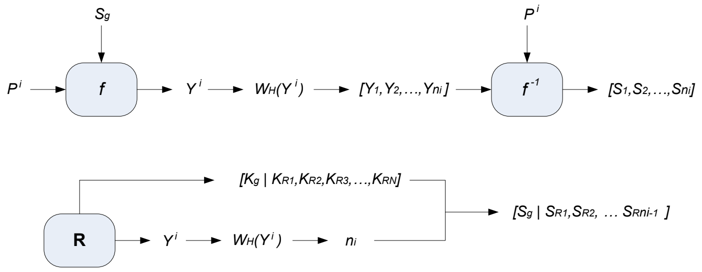


Fig. 5: Multiple leakages experiments: real S-boxes and random S-boxes simulation.

We generated a large number (namely 100 000) of observations and, for these generated observations, derived the experimental average success rate in the two previous contexts. Additionally, we compared these experiments with the theoretical predictions of the previous section. The results of our analysis are pictured in Figure 6, where we can observe that the real S-box gives rise to lower success rates (*i.e.* to less information) than a random function. The reason of this phenomenon is that actual S-boxes give rise to correlated lists of key candidates and therefore to less independence between consecutive observations, as already suggested in [2, 11]. These experiments suggest that even if not perfectly correct, the assumption that block cipher components are reasonably approximated by random functions with respect to side-channel attacks is acceptable. We note that this assumption is better verified for large bit sizes since large S-boxes better approximate the behavior of a random function than small ones.

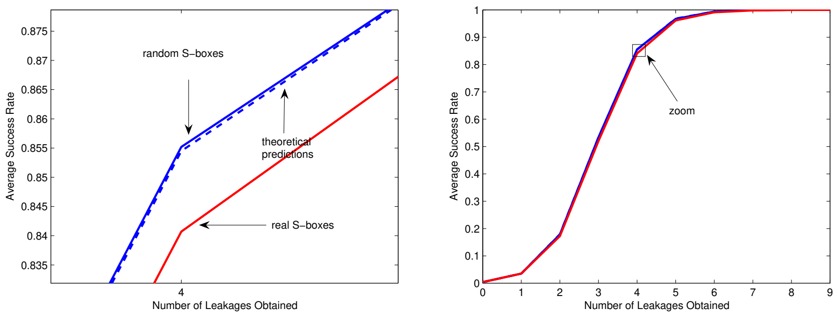


Fig. 6: Multiple leakages experimental results.

5 Investigation of masked implementations

The previous sections illustrated the evaluation of simple side-channel attacks based on a Hamming weight leakage function thanks to the average success rate and mutual information. However, due to the simplicity of the investigated contexts, these notions appeared to be closely correlated. Therefore it was not clear how one could need both criteria for our evaluation purposes. In this section, we consequently study a more complex case, namely masked implementations and higher-order side-channel attacks. This example is of particular interest since it allows us to emphasize the importance of a combination of security and information theoretic metrics for the physical security evaluation process of an implementation. As a result of our analysis, we provide (non-trivial) observations about the respective effectiveness of masking and algorithmic noise addition that can be easily turned into design criteria for actual countermeasures.

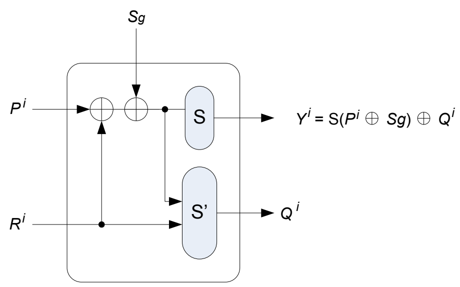
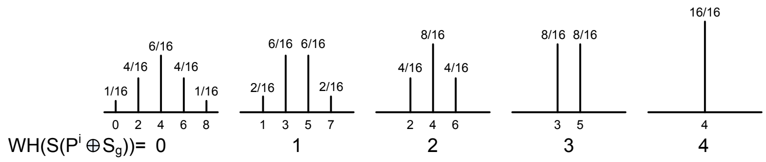


Fig. 7: 1^{st} order boolean masking.

The masking technique (*e.g.* [4]) is one of the most popular ways to prevent block cipher implementations from Differential Power Analysis. However, recent results suggested that it is not as secure as initially thought. Originally proposed by Messerges [7], second and higher-order power analysis attacks can be successfully implemented against various kinds of designs and may not require more hypotheses than a standard DPA [9]. In [12], an analysis of higher-order masking schemes is performed with respect to the correlation coefficient. In the following, we intend to extend this analysis to the (more powerful but less flexible) case of a Bayesian adversary, as introduced in [10].

For the purposes of our analysis, we will use the masked implementation illustrated in Figure 7 in which the plaintext P^i is initially XORed with a random mask R^i. We use two S-boxes S and S' such that: $\mathrm{S}(P^i \oplus R^i \oplus S_g) = \mathrm{S}(P^i \oplus S_g) \oplus Q^i$, with $Q^i = \mathrm{S}'(P^i \oplus R^i \oplus S_g, R^i)$. According to the notations introduced in [10], it is particularly convenient to introduce the secret state of the implementation as $\Sigma_g = \mathrm{S}(P^i \oplus S_g)$ and assume an adversary that obtains (possibly noisy) observations: $O^i_{\Sigma_g} = W_H[\Sigma_g \oplus Q^i] + W_H[Q^i] + N(0, \sigma^2)$. Similarly to a first-order side-channel attack, the objective of an adversary is then to determine the secret state Σ_g (it directly yields the secret key S_g). Because of the masking, Σ_g is not directly observable through side-channel measurements but its associated

PDFs do, since these PDFs only depend on the Hamming weight of the secret state $W_H(\Sigma_g)$. As an illustration, we provide the different discrete PDFs (over the random mask values) for a 4-bit masked design in Figure 8, in function of the secret state Σ_g. We also depict the shapes of the discrete PDFs corresponding to an unmasked secret state affected by four bits of algorithmic noise (*i.e.* we add 4 random bits to the 4-bit target and the PDF is computed over these random bits). Similar distributions can be obtained for any bit size. In general, knowing the probability distributions of the secret state, the average success rate and conditional entropy can be straightforwardly derived:



(a) 4-bit masked value

(b) 4-bit value and 4 noisy bits

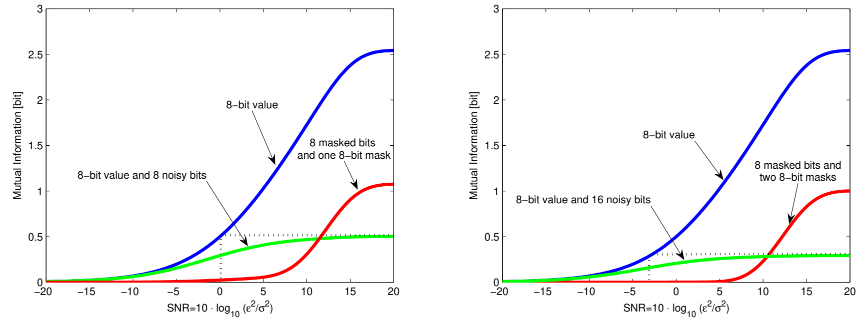
Fig. 8: Exemplary discrete leakage PDFs.

$$\overline{\mathbf{S_R}} = \underset{\Sigma_g}{\mathbf{E}} \; \underset{O^i_{\Sigma_g}}{\mathbf{E}} \; \mathrm{I}_{\Sigma_g,\Sigma_g} = \sum_{h=0}^{n} \frac{\binom{n}{h}}{2^n} \cdot \int_{-\infty}^{+\infty} \mathbf{P}[O_{\Sigma_g}|h] \cdot \mathrm{I}_{\Sigma_g,\Sigma_g} \; do, \tag{9}$$

$$\mathbf{H}[S_g|O_{S_g}] = \underset{\Sigma_g}{\mathbf{E}} \; \mathrm{H}_{\Sigma_g,\Sigma_g} = \sum_{h=0}^{n} \frac{\binom{n}{h}}{2^n} \cdot \int_{-\infty}^{+\infty} \mathbf{P}[O_{\Sigma_g}|h] \cdot -\log_2(\mathbf{P}[\Sigma_g|O_{\Sigma_g}]) \; do, \tag{10}$$

where $\mathbf{P}[O_{\Sigma_g} = o|W_H(\Sigma_g) = h]$ can be computed as in Section 3.3, assuming that the O_{Σ_g} are distributed as a mixture of Gaussians. In the following, we illustrate these metrics in different contexts. First, we consider 2^{nd} and 3^{rd} order masking schemes for 8-bit S-boxes. Then, we consider unmasked implementations where 8 (*resp.* 16) random bits of algorithmic noise are added to the secret signal S_g, corresponding to the 2^{nd} (*resp.* 3^{rd}) order mask bits.

The first (and somewhat surprising) conclusion of our experiments appears in Figure 9. Namely, looking at the mutual information for high SNRs, the use of a n-bit mask is less resistant (*i.e.* leads to lower leakages) than the addition of n random bits to the implementation. Fortunately, beyond a certain amount of noise the masking appears to be a more efficient protection. The reason of this behavior appears clearly when observing the evolution of the PDFs associated to each secret state in function of the SNR, pictured in Appendix B, Figures 13 and 14. Clearly, the PDFs of the masked implementation are very different with small

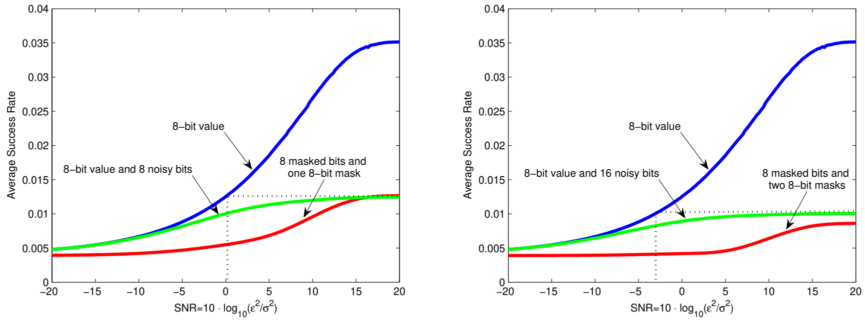


(a) 2^{nd} order masking (b) 3^{rd} order masking

Fig. 9: Mutual information of 2^{nd}, 3^{rd} order masking and equivalent algorithmic noise.

noise values (*e.g.* in Figure 13.a, the probability that an observation belong to both PDFs is very small) but becomes almost identical when the noise increases, since they are all identically centered (*e.g.* in Figure 13.b). Conversely, the means of each PDF in the unmasked implementations stay different whatever the noise level (*e.g.* in Figure 14.b). Therefore the Bayesian classification is easier than in the masked case when noise increases. These observations confirm the usually accepted fact that efficient protections against side-channel attacks require to combine different countermeasures. A practically important consequence of our results is the possibility to derive the exact design criteria (*e.g.* the required amount of noise) to obtain an efficient masking.

It is also interesting to observe that Figure 9 confirms that algorithmic noise is nicely modeled by Gaussians. Indeed, *e.g.* for the 2^{nd} order case, the mutual information of an 8-bit value with 8 noisy bits for high SNRs exactly corresponds to the one of an unprotected 8-bit value with SRN=0.



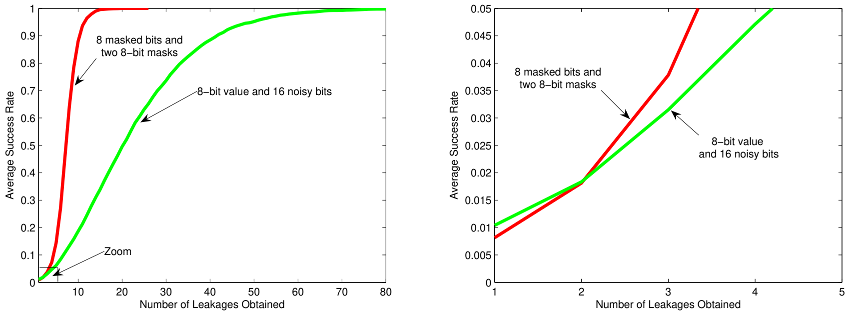
(a) 2^{nd} order masking (b) 3^{rd} order masking

Fig. 10: Avg. success rate of 2^{nd}, 3^{rd} order masking and equivalent algorithmic noise.

The second interesting conclusion is that the average success rate after one query (pictured in Figure 10) does *not* follow an identical trend. Namely, the masked implementations and their equivalent noisy counterparts do *not* cross over at the same SRN. This situation typically corresponds to the intutive category of weak implementations in Figure 1. That is, some information is available but the number of queries is too low to turn it into a successful attack. If our information theoretic measurement is meaningful, higher number of queries should therefore confirm the intuition in Figure 9.

Success rates with higher number of queries for a 3^{rd} order masking scheme (and noisy equivalent) were simulated in Figures 11, 12. In Figure 11, a very high SNR=20 is considered. As a consequence, we observe that the masks bring much less protection than their equivalent in random bits, although the initial value (for one single query) suggests the opposite. Figure 12 performs similar experiments for two SNRs that are just next to the crossing point. It illustrates the same intuition that the efficiency of the key recovery when increasing the number of queries is actually dependent on the information content in the observations.

Importantly, these experiments illustrate a typical context where the combination of security and information theoretic metrics is meaningful. While the average success rate is the only possible metric for the comparison of different side-channel attacks (since it could be evaluated for different statistical tools), the information theoretic metric allows to infer the behavior of an attack when increasing the number of queries. As an illustration, the correlation-based analysis performed in [12] only relates to one particular (sub-optimal) statistical tool and was not able to lead to the observations illustrated in Figure 9.

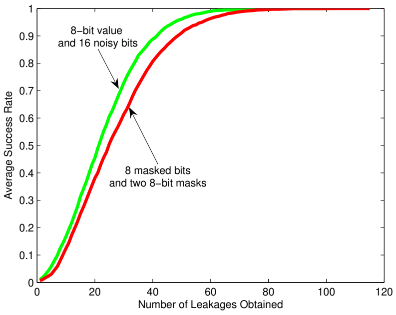


(a) Comparison (b) Zoom

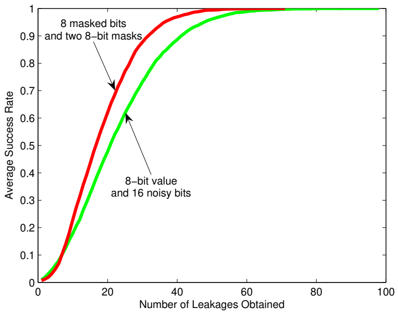
Fig. 11: Avg. success rate of an 8-bit 3^{rd} order masking scheme with noisy counterpart.

6 Concluding remarks

This paper discusses the relevance of a recently introduced theoretical framework for the analysis of cryptographic implementations against side-channel attacks. By the investigation of a number of implementation contexts, we illustrate the interest of a combination of security and information theoretic metrics for the evaluation, comparison and understanding of side-channel attacks. Specifically, in a well defined adversarial context and strategy, the average success rate would allow the comparison of different usually considered side-channel attacks (*e.g.*



(a) SNR=10



(b) SNR=11

Fig. 12: Avg. success rate of an 8-bit 3^{rd} order masking scheme with noisy counterpart.

DPA, correlation analysis, template attacks). By contrast, independently of the statistical tools used by the adversary, an information theoretic metric provides theoretical insights about the behavior and effects of a particular leakage function that can possibly be turned into practical design criteria.

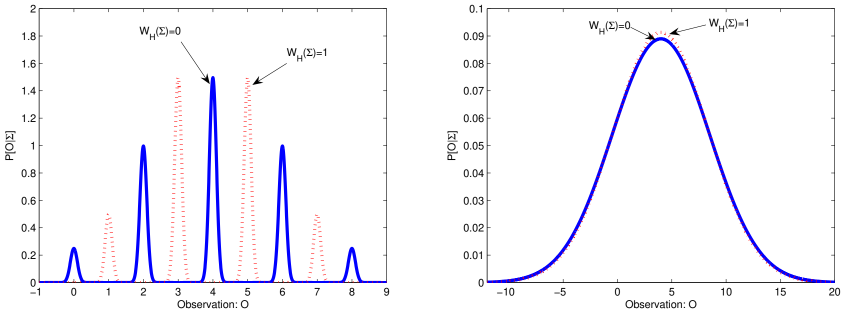
References

1. S. Chari, J.R. Rao, P. Rohatgi, *Template Attacks*, CHES 2002, LNCS, vol. 1965, pp. 13–28.
2. E. Brier, C. Clavier, F. Olivier, *Correlation Power Analysis with a Leakage Model*, CHES 2004, LNCS, vol 3156, pp 16-29.
3. J.-S. Coron, P. Kocher, D. Naccache, *Statistics and Secret Leakage*, Financial Crypto 2000, LNCS, vol. 1972, pp. 157–173.
4. L. Goubin, J. Patarin, *DES and Differential Power Analysis*, CHES 1999, LNCS, vol. 1717, pp. 158-172.
5. P. Kocher, J. Jaffe, B. Jun, *Differential Power Analysis*, CRYPTO 1999, LNCS, vol. 1666, pp. 15–19.
6. S. Mangard, *Hardware Countermeasures against DPA - a Statistical Analysis of their Effectiveness*, CT-RSA 2004, LNCS, vol. 2964, pp. 222-235.
7. T.S. Messerges, *Using Second-Order Power Analysis to Attack DPA Resistant Software.*, CHES 2000, LNCS, vol. 2523, pp. 238–251.
8. S. Micali, L. Reyzin, *Physically Observable Cryptography (extended abstract).*, TCC 2004, LNCS, vol. 2951, pp. 278–296.
9. E. Oswald, S. Mangard, C. Herbst, S. Tillich, *Practical Second-Order DPA Attacks for Masked Smart Card Implementations of Block Ciphers.*, CT-RSA 2006, LNCS, vol. 3860, pp. 192–207.
10. E. Peeters, F.-X. Standaert, N. Donckers, J.-J. Quisquater, *Improved Higher-Order Side-Channel Attacks with FPGA Experiments*, CHES 2005, LNCS, vol. 3659, pp. 309–323.
11. E. Prouff, *DPA Attacks and S-Boxes*, FSE 2005, LNCS, vol. 3557, pp. 424-441.
12. K. Schramm, C. Paar, *Higher Order Masking of the AES*, CT-RSA 2006, LNCS, vol. 3860, 208-225.
13. W. Schindler, K. Lemke, C. Paar, *A Stochastic Model for Differential Side-Channel Cryptanalysis*, CHES 2005, LNCS, vol 3659, pp 30-46.
14. F.-X. Standaert, T.G. Malkin, M. Yung, *A Formal Practice-Oriented Model For The Analysis of Side-Channel Attacks*, Cryptology ePrint Archive, Report 2006/139, 2006, `http://eprint.iacr.org/`.

A Probability density function of the variable I_q

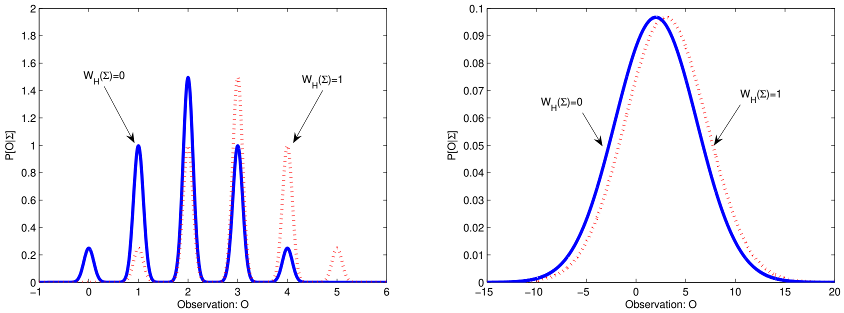
We take an iterative approach and first consider the intersection after two leakages. Assuming that the leakages respectively give rise to uniform lists of n_1 and n_2 candidates and the the key space has size $N = 2^n$, it yields $\mathbf{P}[I_2 = i|n_1, n_2] = \frac{\binom{n_1-1}{i-1}\cdot\binom{N-n_1}{n_2-i}}{\binom{N-1}{n_2-1}}$, where the binomials are taken among sets of $N-1$ possible elements since there is one fixed key that is not chosen uniformly. Then, assuming the knowledge of the distribution of $I_q(n_1, n_2, ..., n_q)$ and an additional leakage that gives rise to a uniform list of n_{new} candidates, we can derive the distribution of I_{q+1} as follows: $\mathbf{P}[I_{q+1} = j|I_q, n_{new}] = \sum_i \mathbf{P}[I_{q+1} = j|I_q = i, n_{new}]\cdot\mathbf{P}[I_q = i]$, with: $\mathbf{P}[I_{q+1} = j|I_q = i, n_{new}] = \frac{\binom{i-1}{j-1}\cdot\binom{N-i}{n_{new}-j}}{\binom{N-1}{n_{new}-1}}$.

B Additional figures



(a) SNR=10 (b) SNR=−6

Fig. 13: Leakages PDFs in function of the noise: masked implementation.



(a) SNR=10 (b) SNR=−6

Fig. 14: Leakages PDFs in function of the noise: unmasked implementation.