A Unified Framework for the Analysis of Side-Channel Key Recovery Attacks

François-Xavier Standaert[1], Tal G. Malkin[2], Moti Yung[2,3]

[1] UCL Crypto Group, Université Catholique de Louvain.
[2] Dept. of Computer Science, Columbia University. [3] Google Inc.
e-mails: fstandae@uclouvain.be, tal,moti@cs.columbia.edu

Version 2.0, January 2, 2008.

Abstract. The fair evaluation and comparison of side-channel attacks and countermeasures has been a long standing open question, limiting further developments in the field. Motivated by this challenge, this work proposes a framework for the analysis of cryptographic implementations that includes a theoretical model and an application methodology. The model is based on weak and commonly accepted hypotheses about side-channels that computations give rise to. It allows quantifying the effect of practically relevant leakage functions with a combination of security and information theoretic metrics, respectively measuring the quality of an implementation and the strength of an adversary. From a theoretical point of view, we demonstrate formal connections between these metrics and discuss their intuitive meaning. From a practical point of view, the model implies a unified methodology for the analysis of side-channel key recovery. The proposed solution allows getting rid of most of the subjective parameters that were limiting previous specialized and often ad hoc approaches in the evaluation of physically observable devices. It typically determines the extent to which basic (but practically essential) questions such as "*How to compare two implementations?*" or "*How to compare two side-channel adversaries?*" can be fairly answered.

1 Introduction

Traditionally, cryptographic algorithms provide security against an adversary who has only black box access to cryptographic devices. That is, the only thing the adversary can do is to query the cryptographic algorithm on inputs of its choice and analyze the responses, which are always computed according to the correct original secret information. However, such a model does not always correspond to the realities of physical implementations. During the last decade, significant attention has been paid to the physical security evaluation of cryptographic devices. In particular, it has been demonstrated that actual attackers may be much more powerful than what can be captured by the black box model. In this paper, we investigate the security of cryptographic implementations with respect to side-channel attacks, in which adversaries are enhanced with the possibility to exploit physical leakages such as power consumption [18] or electromagnetic radiation [2]. A large body of experimental work has been created on the subject, and although numerous countermeasures are proposed in the liter-

ature, protecting implementations against such attacks is usually difficult and expensive. Moreover, most proposals we are aware of only increase the difficulty of performing the attacks, but do not fundamentally prevent them.

As a consequence of this state-of-the art, our following work was first motivated by theoretical concerns. Perhaps surprisingly (and to the best of our knowledge), there have been only a few attempts to model physical attacks properly, and to provably address their security. A notable example is the work of Micali and Reyzin who initiated an analysis of side-channels taking the modularity of physically observable computations into account. It notably defines the notion of *physical computer* that is the combination of an abstract computer (*i.e.* a Turing machine) and a leakage function. The model in [25] is very general, capturing almost any conceivable form of physical leakage. However, arguably because of the great generality of the assumptions, the obtained positive results (*i.e.* leading to useful constructions) are quite restricted in nature, and it is not clear how they apply to practice. This is especially true for primitives such as modern block ciphers for which even the black box security cannot be proven. Thus, the study of more specialized contexts and specific scenarios which may lead to practical applications was suggested as a scope for further research.

But most importantly, our work was motivated by practical issues in the analysis of side-channel attacks. In particular, the difficulty of comparing different implementations or adversaries (*e.g.* mentioned in [22], page 163) was the main starting point of our investigations. As a matter of fact, the evaluation criteria in physically observable cryptography should be unified in the sense that they should be adequate and have the same meaning for analyzing any type of implementation or adversary. This is clearly opposed to the combination of ad hoc solutions relying on specific ideas designers have in mind. As a typical illustration, let us consider the comparison of two implementations X and Y of the same algorithm. Let us also assume that one protects X by randomizing its computations (*e.g.* with [15]) and protects Y by making its leakage as constant as possible (*e.g.* with [33]). Good evaluation metrics should allow the comparison of both countermeasures. But former techniques for their analysis do not provide such metrics, thus limiting both the understanding of side-channel attacks and the ability to trade performance for security on a fair basis.

As far as comparing different implementations is concerned, present solutions for the analysis of side-channel attacks typically allow the statement of claims such as: "*An implementation X is "better" than an implementation Y against an adversary A*". The results in this paper aim to discuss the extend to which more meaningful (adversary independent) statements can be claimed such as: "*An implementation X is "better" than an implementation Y*". We show that such claims can actually be stated in practically meaningful contexts. As far as evaluating different adversaries is concerned, present solutions for the analysis of side-channel attacks typically allow the statement of claims such as: "*An adversary A successfully recovers one key byte of an implementation X after the observation of q measurement queries.*". But such a hard definition for a success may not be adapted to model any type of adversarial strategy. And in

practice, it may also be interesting to evaluate how the security evolves with respect to the number of observations q. The metrics introduced in this paper consequently allows the claim of more flexible statements such as: "*An adversary A has probability p to have the target key byte of an implementation X rated* 1^{st} *(resp.* 2^{nd}, 3^{rd}, ...*) among the possible key candidates after the observation of q measurement queries*". We note that if obtained through statistical sampling, these claims have to come with a certain confidence interval (that is frequently neglected in the present literature on physically observable cryptography).

Following these examples, our results aim to get rid of previous limitations in the evaluation of side-channel attacks. For this purpose, we restrict the most general model of Micali and Reyzin to reasonable (*i.e.* practically relevant) adversaries. Namely, and as a first step in the investigation of physically observable devices, we focus on the side-channel key recovery problem that is the most frequently considered in practice. Then, we argue that the evaluation of side-channels actually requires two metrics. First, an information theoretic metric (also denoted as asymptotic security metric) is used to measure the amount of information that is provided by a given implementation. Second, an actual security metric is used to measure how this information can be turned into a successful attack. We show that these metrics allow comparing different implementations or adversaries. We also demonstrate some important formal connections between them and discuss their intuitive meaning. Eventually, we move from formal definitions towards practice-oriented definitions in order to introduce a unified evaluation methodology for side-channel attacks. We also provide exemplary applications of the model in a number of practical contexts.

Related works mainly include a large literature on side-channel issues, ranging from attacks to countermeasures and including statistical analysis concerns. The side-channel lounge [11], DPA book [22] and the CHES series of workshops [8] respectively provide a good list of reference, a state-of-the art view of the field and some recent developments. Most of these previous works can be included in the following framework. It generally provides an improvement of their understanding. The goal of this report is therefore to facilitate the interface between theoretical and practical aspects in physically observable cryptography.

Finally, the following modelling exploits several ideas from the classical communication theory [10, 28, 29]. But while source and channel coding attempt to put the information in an efficient format for its transmission, cryptographic engineers have the opposite goal to make their circuit's internal configurations as unintelligible as possible to the outside world. This analogy provides a background and rationale for our metrics. We mention that different measures of uncertainty have also been used in [19] to quantify the effectiveness of adaptive strategies in side-channel attacks. These results nicely illustrate that various solutions can be considered in the evaluation of side-channel attacks. Our line of research follows a slightly different approach in the sense that we assign specific tasks to different metrics. Namely, we suggest to evaluate asymptotic security (hence, implementations) with the conditional entropy and to evaluate actual security (hence adversaries) with either a success rate or the guessing entropy.

Otherwise said, we bring motivation and analysis for these three metrics. In our model, the use of an average metric (over a uniform key space) to compare implementations is justified by the need of adversary independence. By contrast, specific (*e.g.* worst case) strategies can be quantified with the security metrics. This does not prevent other solutions to be meaningful, depending on the applications. However, we believe the following approach provides sound and necessary tools for a better understanding of physically observable cryptography.

The rest of the paper is structured as follows. Section 2 contains the background necessary for the understanding of our results. Section 3 provides an intuitive description of our model and terminology. Section 4 introduces the main assumption for the analysis of side-channel attacks with the conditional entropy. Section 5 defines our evaluation metrics formally and Section 6 demonstrates some important connections between these metrics, with their intuitive consequences. Sections 7 and 8 respectively contain the practice-oriented definition of a side-channel adversary and an evaluation methodology for physically observable cryptographic devices. Finally, some exemplary applications of the model are referred to in Section 9 and conclusions are in Section 10.

2 Background

In order to enable the analysis of physically observable cryptography, Micali and Reyzin introduced a model of computation of which we recall certain definitions of interest with respect to our following results. It is based on the five informal axioms given in Appendix A. From these axioms, an *abstract computer* was defined in [25] as a collection of special Turing machines, which invoke each other as subroutines and share a special common memory. Each member of the collection is denoted as an *abstract virtual-memory Turing machine* (abstract VTM or simply VTM for short). One writes $\alpha = (\alpha_1, \alpha_2, ..., \alpha_n)$ to mean that an abstract computer α consists of abstract VTMs $\alpha_1, \alpha_2, ..., \alpha_n$. All VTM inputs and outputs are binary strings always residing in some virtual memory. Abstract computers and VTMs are not physical devices: they only represent logical computation and may have many different physical realizations.

Then, to model the physical leakage of any particular instantiation of an abstract computer, the notion of *physical VTM* was introduced. A physical VTM is a pair (L_i, α_i), where α_i is an abstract VTM and L_i is a *leakage function*. If $\alpha = (\alpha_1, \alpha_2, ..., \alpha_n)$ is an abstract computer, then $\varphi_i = (\mathsf{L}_i, \alpha_i)$ represents one physical realization of α_i and $\varphi = (\varphi_1, \varphi_2, ..., \varphi_n)$ is defined as a physical realization of the abstract computer α, also called physical computer for short. It can be denoted as the combination $\varphi = (\alpha, \mathsf{L})$ with $\mathsf{L} = (\mathsf{L}_1, \mathsf{L}_2, \ldots, \mathsf{L}_n)$. In these definitions, the relation between an abstract computing machine and a physical realization is only determined by the leakage function that is qualitatively defined as a function of three inputs, $\mathsf{L}(C_\alpha, M, R)$. The first input is the current internal configuration C_α of an abstract computer α, which incorporates anything that is in principle measurable. The second input M is the setting of the measuring apparatus (*i.e.* a specification of what and how the adversary chooses to measure). The third input R is a random string to model the randomness of the measurement process.

3 Intuitive description of the model and terminology

As a matter of fact, the previous definition of leakage function models the physical observations of a target device. But it does not specify how an adversary could exploit this side-channel information. This section consequently intends to intuitively describe the side-channel key recovery attacks that will be formally investigated in the rest of the paper, with the metrics used to quantify them.

A generic side-channel key recovery is pictured in Figure 1 that we detail as follows. First, the term *primitive* is used to denote cryptographic routines corresponding to the practical instantiation of some idealized functions required to solve cryptographic problems. For example, the AES Rijndael is a cryptographic primitive. With respect to the model of Micali and Reyzin, cryptographic primitives are abstract computers. They can be viewed as black boxes, parametrized by some secret argument. Second, the term *device* is used to denote the physical realization of a cryptographic primitive. For example, a smart card or and FPGA running the AES Rijndael can be the target devices of a side-channel attack. With respect to the model of Micali and Reyzin, a device corresponds to the division of an abstract computer or primitive into different abstract VTMs. A *side-channel* is an unintended communication channel that leaks some information from a device through a physical media. For example, the power consumption or the electromagnetic radiation of a target device can be used as side-channels. The output of a side-channel is a *physical observable*. Then, the *leakage function* is an abstraction used to model all the physical specificities of a side-channel adversary, up to the measurement setup used to monitor the physical observables (the leakage function output equals this setup output). An *implementation* (or physical computer) is the combination of an abstract computer (or primitive) and a leakage function. Finally, a *side-channel adversary* is composed of a physical part (the *measurement setup* included in the leakage function abstraction) and an algorithmic part (sometimes denoted as a *distinguisher*) that turns these physical leakages into a guess for the target signal.

Figure 1 suggests that, similarly to the classical communication theory, two metrics are needed to quantify the effectiveness of a side-channel attack. First, an information theoretic metric (also denoted as asymptotic security metric) evaluates the amount of information in the side-channel leakages. It aims to measure what is achievable by an unbounded adversary. Since it purposely relates the the strongest possible adversarial context, it can be used to answer our first question, namely: "*how to compare different implementations?*". Second, an actual security metric evaluates to which extent an adversarial strategy can turn the side-channel information into a successful attack. It is the typical counterpart of the Bit-Error-Rate in communication problems and can therefore be used to answer our second question: "*how to compare different adversaries?*".

Compared to the original work of Micali and Reyzin, we add a description of the side-channel adversary to the model (Section 7), define metrics to quantify the attacks (Sections 5, 6) and derive an evaluation methodology (Section 8). For these purposes, we will require one main assumption that we now detail.

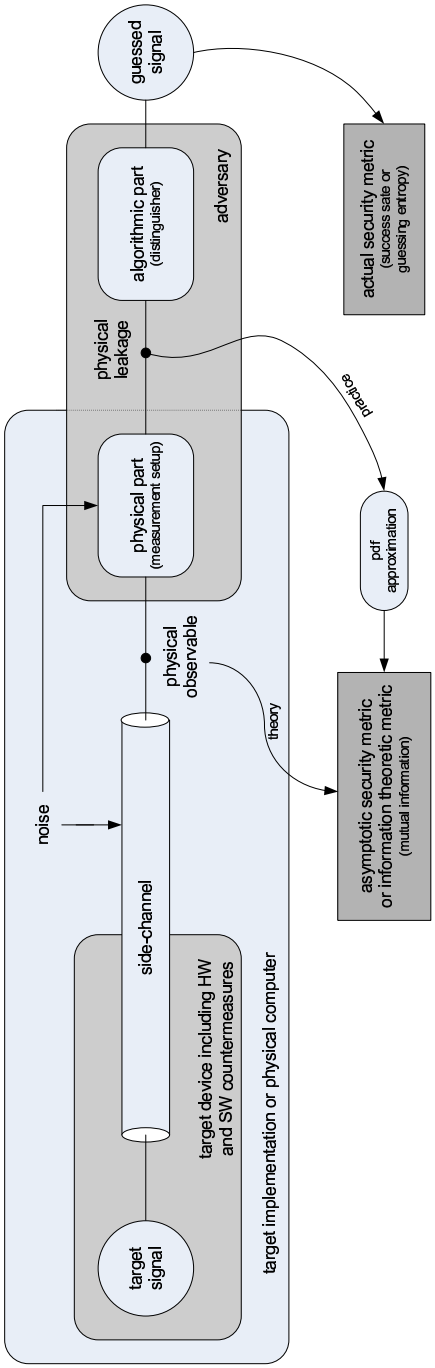


Fig. 1: Intuitive description of a side-channel key recovery attack.

4 Model assumption

One important goal of the information theoretic metric is to allow a sound evaluation of a given implementation, if possible independently of an adversary's algorithmic details. Therefore, the strategy we propose in this paper can be summarized as: "consider the strongest possible adversary and give him unbounded means in terms of measurement queries". As a matter of fact, this raises the question of which is the most powerful side-channel distinguisher.

Let S and $\mathbf{L}$ be two random variables respectively denoting the target signal in a side-channel attack and the corresponding leakage (to be defined formally in Section 5). Let s and $\mathbf{l}$ denote the realizations of these random variables. Assuming that an adversary has access to different leakages and knows the conditional probabilities $\Pr[\mathbf{L} = \mathbf{l}|S = s^*]$, or $\Pr[\mathbf{l}|s^*]$ for short, the best strategy would be to use a Bayesian distinguisher and to select the key candidate as $argmax_{s^*} \Pr[s^*|\mathbf{l}]$. Unfortunately, there are two caveats in application of this optimal strategy which implies the need of an assumption in our model.

First, the conditional probability distribution $\Pr[\mathbf{L}|S]$ cannot be known by the adversary. They can only be approximated through physical observations. This is the reason for the leakage function abstraction in the model of Micali and Reyzin. It informally states that the only way an adversary knows a leakage function is through actual measurements. As a consequence, actual attacks have to exploit approximated distribution $\hat{\Pr}[\mathbf{L}|S]$ rather than actual one $\Pr[\mathbf{L}|S]$.

Second, actual leakages may have very large dimensions since they are typically the output of a high sampling rate acquisition device like an oscilloscope. As a consequence, the approximation of the probability distributions for all the leakage samples is computationally intensive. Practical attacks usually approximate the probability distribution of a reduced set of samples, denoted as $\hat{\Pr}[\tilde{\mathbf{L}}|S]$.

Side-channel attacks that apply a Bayesian classifier and exploit the approximated probability distributions of a reduced set of leakage samples are usually known as template attacks [7]. It directly leads to our main assumption:

Assumption: *template attacks are the strongest possible side-channel attacks.*

We note that this is a very common hypothesis in the side-channel literature. However, the generic term of template hides a certain amount of complexity. For example, the question of how to select the relevant leakage samples to approximate is an important one from a practical point of view [3]. In general, the better practical template attacks perform against an implementation, the more relevant the evaluation of the information theoretic metric will be. We also mention that the security metrics in Section 5.1 do not depend on this assumption. Only their relation with the information theoretic metric does. We note finally that stochastic models can be used as an alternative to template attacks for the leakage probability distribution approximation [14, 27].

Before moving to the formal definitions of our different metrics, we introduce a last notion related to Figure 1. More specifically, we consider the "theory" and "practice" arrows leading to the information theoretic metric. These arrows underline the fact that one can always assume a theoretical model for the side-channel and perform a *simulated attack*. If the model is meaningful, so the simulated attack will be. But such simulations always have to be followed by an *experimental attack* in order to confirm the relevance of the model. Experimental attack exploit actual leakages obtained from a measurement setup.

5 Formal definitions

In this section, we formally define the metrics that we suggest for the evaluation of a side-channel key recovery adversary. We first consider the actual security and detail two possible metrics, corresponding to different (more or less flexible) computational strategies. Then, following the standard approach in information theory, we propose the use of Shannon's definition of entropy to quantify the amount of information leaked by a cryptographic device.

5.1 Actual security metrics

Success rate of the adversary. As most cryptanalytic techniques, side-channel attacks are usually based on a divide-and-conquer strategy in which different (computationally tractable) parts of a secret key are recovered separately. In general, the attack defines a function $\delta : \mathcal{K} \rightarrow \mathcal{S}$ which maps each key k onto an equivalent key class $s = \delta(k)$, such that $|\mathcal{S}| \ll |\mathcal{K}|$.

Let $\mathsf{E}_K(.) = \{\mathsf{E}_k\}_{k \in \mathcal{K}}$ be a family of cryptographic abstract computers indexed by a variable key K. Let $(\mathsf{E}_K, \mathsf{L})$ be the physical computer corresponding to the association of E_K with a leakage function L. We define a side-channel key recovery adversary as an algorithm $\mathsf{A}_{\mathsf{E}_K,\mathsf{L}}$ with time complexity τ, memory complexity m and q queries to the target physical computer. The aim of a side-channel adversary is to guess a key class $s = \delta(k)$ with non negligible probability. For this purpose, we assume that the output of the adversary $\mathsf{A}_{\mathsf{E}_K,\mathsf{L}}$ is a guess vector $\mathbf{g} = [g_1, g_2, \ldots, g_{|\mathcal{S}|}]$ with the different key candidates sorted according to their likelihood: the most likely candidate being g_1. Finally, we define a side-channel key recovery of order o with the following experiment:

Experiment $\mathbf{Exp}^{\text{sc-kr-}o}_{\mathsf{E}_K,\mathsf{L}}$
$k \xleftarrow{R} \mathcal{K}$;
$s = \delta(k)$;
$\mathbf{g} \leftarrow \mathsf{A}_{\mathsf{E}_k,\mathsf{L}}$;
if $s \in [g_1, \ldots, g_o]$ **then** return 1;
else return 0;

The oth order success rate of the side-channel key recovery adversary $\mathsf{A}_{\mathsf{E}_K,\mathsf{L}}$ against a key class variable S is straightforwardly defined as:

$$\mathbf{Succ}^{\text{sc-kr-}o,S}_{\mathsf{A}_{\mathsf{E}_K,\mathsf{L}}}(\tau, m, q) = \Pr\ [\mathbf{Exp}^{\text{sc-kr-}o}_{\mathsf{E}_K,\mathsf{L}} = 1] \quad (1)$$

Intuitively, a success rate of order 1 (*resp.* 2, ...) relates to the probability that the correct key is sorted first (*resp.* among the two first ones, ...) by the adversary. When not specified, a first order success rate is assumed.

Computational restrictions. Similarly to black box security, computational restrictions have to be imposed to side-channel adversaries in order to capture the reality of physically observable cryptographic devices. This is the reason for the parameters τ, m, q. Namely, the attack time complexity τ and memory complexity m (mainly dependent on the number of key classes $|\mathcal{S}|$) are limited by present computer technologies. The number of measurement queries q is limited by the adversary's ability to monitor the device.

However, additionally to the computational cost of the side-channel attack itself, another important parameter is the remaining workload after the attack. For example, considering a success rate of order o implies that the adversary still has a maximum of o key candidates to test after the attack. If this has to be repeated for different parts of the key, it may become a non negligible task. As a matter of fact, the previously defined success rate measures an adversary with a fixed maximum workload after the side-channel attack.

A more flexible metric that is also convenient in our context is the guessing entropy [6]. It measures the average number of key candidates to test after the side-channel attack. The guessing entropy was originally defined in [23] and has been proposed to quantify the effectiveness of adaptive side-channel attacks in [19]. It can also be related to the notion of gain that has been used in the context of multiple linear cryptanalysis to measure how much the complexity of an exhaustive key search has been reduced thanks to an attack [4].

Guessing entropy. Using the same notations as for the success rate, we can define a side-channel key guessing experiment:

Experiment $\mathbf{Exp}^{\text{sc-kg}}_{\mathsf{E}_K,\mathsf{L}}$
$k \xleftarrow{R} \mathcal{K}$;
$s = \delta(k)$;
$\mathbf{g} \leftarrow \mathsf{A}_{\mathsf{E}_k,\mathsf{L}}$;
return i such that $g_i = s$;

The guessing entropy of the side-channel key recovery adversary $\mathsf{A}_{\mathsf{E}_K,\mathsf{L}}$ against a key class variable S is then defined as:

$$\mathbf{GE}^{\text{sc-kr-}S}_{\mathsf{A}_{\mathsf{E}_K,\mathsf{L}}}(\tau, m, q) = \mathbf{E}\big(\mathbf{Exp}^{\text{sc-kg}}_{\mathsf{E}_K,\mathsf{L}}\big), \tag{2}$$

Interestingly, while a low success rate of order o does not prevent having large success rates of orders $o+1, o+2, \ldots$, the guessing entropy directly indicates the average remaining workload of the side-channel adversary.

We now define the information theoretic (or asymptotic security) metric.

5.2 Information theoretic (or asymptotic security) metric

Let S be the previously used target key class discrete variable of a side-channel attack and s be a realization of this variable. Let $\mathbf{X}_q = [X_1, X_2, \ldots, X_q]$ be a vector of variables containing a sequence of inputs to the target physical computer and $\mathbf{x}_q = [x_1, x_2, \ldots, x_q]$ be a realization of this vector. Let $\mathbf{L}_q$ be a random vector denoting the side-channel observations generated with q queries to the target physical computer and $\mathbf{l}_q = [l_1, l_2, \ldots, l_q]$ be a realization of this random vector, *i.e.* one actual output of the leakage function L corresponding to the input vector $\mathbf{x}_q$. Let finally $\Pr[s|\mathbf{l}_q]$ be the conditional probability of a key class s given a leakage $\mathbf{l}_q$. We define the conditional entropy matrix as:

$$\mathbf{H}^q_{s,s^*} = -\sum_{\mathbf{l}_q} \Pr[\mathbf{l}_q|s] \cdot \log_2 \Pr[s^*|\mathbf{l}_q], \tag{3}$$

from which we derive Shannon's conditional entropy[1]:

$$\mathrm{H}[S|\mathbf{L}_q] = -\sum_{s} \Pr[s] \sum_{\mathbf{l}_q} \Pr[\mathbf{l}_q|s] \cdot \log_2 \Pr[s|\mathbf{l}_q] = \mathop{\mathbf{E}}_{s} \ \mathbf{H}^q_{s,s} \tag{4}$$

We note that this definition is equivalent to the classical one since:

$$\begin{aligned} \mathrm{H}[S|\mathbf{L}_q] &= -\sum_{\mathbf{l}_q} \Pr[\mathbf{l}_q] \sum_{s} \Pr[s|\mathbf{l}_q] \cdot \log_2 \Pr[s|\mathbf{l}_q] \\ &= -\sum_{s} \Pr[s] \sum_{\mathbf{l}_q} \Pr[\mathbf{l}_q|s] \cdot \log_2 \Pr[s|\mathbf{l}_q] \end{aligned}$$

Then, we define an entropy reduction matrix: $\mathbf{H'}^q_{s,s^*} = \mathrm{H}[S] - \mathbf{H}^q_{s,s^*}$, where $\mathrm{H}[S]$ is the entropy of the key class variable S before any side-channel attack has been performed: $\mathrm{H}[S] = \mathbf{E}_s - \log_2 \Pr[s]$. It directly yields the mutual information:

$$\mathrm{I}(S; \mathbf{L}_q) = \mathrm{H}[S] - \mathrm{H}[S|\mathbf{L}_q] = \mathop{\mathbf{E}}_{s} \ \mathbf{H'}^q_{s,s} \tag{5}$$

Let us finally mention that in the context of simulated attacks where an analytical model for the leakage probability distribution is known, the previous sums can be turned into integrals, *e.g.* we have for the conditional entropy:

$$\mathrm{H}[S|\mathbf{L}_q] = -\sum_{s} \Pr[s] \int_{-\infty}^{+\infty} \Pr[\mathbf{l}_q|s] \cdot \log_2 \Pr[s|\mathbf{l}_q] \ d\mathbf{l}_q$$

In the next section, we investigate the formal connections between the different metrics that we introduced for the analysis of side-channel attacks.

[1] With $\Pr[s|\mathbf{l}_q] = \frac{\Pr[\mathbf{l}_q|s] \cdot \Pr[s]}{\sum_{s^*} \Pr[\mathbf{l}_q|s^*] \cdot \Pr[s^*]}$.

6 Relations between the evaluation metrics

6.1 Asymptotic meaning of the conditional entropy

In this first subsection, we show how the information theoretic metric is related to the strongest possible adversarial context for a side-channel attack. For simplicity, we make no distinction between the real probability distribution $\Pr[\mathbf{L}_q|S]$ and the approximated one $\hat{\Pr}[\tilde{\mathbf{L}_q}|S]$, *i.e.* we use our assumption of Section 4. The consequences of this assumption are discussed in Section 6.3.

We start with three definitions.

Definition 1. The asymptotic success rate of a side-channel adversary $\mathsf{A}_{\mathsf{E}_K,\mathsf{L}}$ against a key class variable S is its success rate when the number of measurement queries q tends to the infinity. It is denoted as: $\mathbf{Succ}^{\text{sc-kr-}S}_{\mathsf{A}_{\mathsf{E}_K,\mathsf{L}}}(q \rightarrow \infty)$.

Definition 2. Given a leakage probability distribution $\Pr[\mathbf{L}_q|S]$ and a number of side-channel queries stored in a leakage vector $\mathbf{l}_q$, a Bayesian side-channel adversary is an adversary that selects the key as $argmax_{s^*} \Pr[s^*|\mathbf{l}_q]$.

Definition 3. We say that a leakage probability distribution $\Pr[\mathbf{L}_q|S]$ is sound if the first-order asymptotic success rate of a Bayesian side-channel adversary exploiting this leakage distribution against the key class variable S equals one.

We now demonstrate the relation between the information theoretic metric and the asymptotic success rate of a Bayesian adversary.

Theorem 1. *Under the condition of independent leakages for a fixed key (i.e.* $\Pr[l_1, l_2|s] = \Pr[l_1|s] \cdot \Pr[l_2|s]$*), a leakage probability distribution is sound if and only if* $argmin_{s^*} \boldsymbol{H}^1_{s,s^*} = s$ *(or* $argmax_{s^*} \boldsymbol{H'}^1_{s,s^*} = s$*),* $\forall s \in \mathcal{S}$.

Proof. let us consider a target key class s and a leakage vector $\mathbf{l}_q$. A Bayesian adversary having access to these leakages is successful if and only if:

$$s = \underset{s^*}{argmax} \ \Pr[s^*|\mathbf{l}_q]$$

$$s = \underset{s^*}{argmax} \ \frac{\Pr[\mathbf{l}_q|s^*] \cdot \Pr[s^*]}{\Pr[\mathbf{l}_q]}$$

Assuming that the probabilities $\Pr[s^*]$ are equal and since $\Pr[\mathbf{l}_q]$ is independent of s^* (it only depends on the correct class s), it directly yields:

$$s = \underset{s^*}{argmax} \ \Pr[\mathbf{l}_q|s^*]$$

Since we have independent leakages for different queries by hypothesis, we find:

$$s = \underset{s^*}{argmax} \ \prod_{i=1}^{q} \Pr[l_i|s^*]$$

When considering an asymptotic attack, each query to the target physical computer determines a leakage trace l_i picked up from a leakage distribution $\Pr[L_i|S]$. Therefore, an asymptotic attack is successful if and only if:

$$
\begin{aligned}
s &= \underset{s^*}{argmax} \prod_{l_i} \Pr[l_i|s^*]^{\Pr[l_i|s]} \\
s &= \underset{s^*}{argmax} \prod_{l_i} \Pr[s^*|l_i]^{\Pr[l_i|s]} \\
s &= \underset{s^*}{argmax} \sum_{l_i} \Pr[l_i|s] \cdot \log_2 \Pr[s^*|l_i] \qquad (6)
\end{aligned}
$$

Finally, we just observe that Equation (6) is equivalent to Equation (3) when $q = 1$ but for their sign. Therefore, if the previous condition holds for all classes s, the Bayesian side-channel attack is asymptotically successful. □

There are three important remarks:

1. q queries to a target device can be seen both as q realizations of a single query leakage vector $\mathbf{L}_1$ or as a single realization of a q-query leakage vector $\mathbf{L}_q$.
2. The condition on the entropy matrix $\mathbf{H}^1_{s,s^*}$ is stated for $q = 1$ since a leakage trace l_i in Equation (6) corresponds to a single query vector $\mathbf{l}_1$. The condition for $q = 1$ straightforwardly involves the condition for any $q > 1$.
3. *Most importantly:* since the condition of independent leakages is conditional to the key classes s, it only requires that the noise in the observations is independent of these classes. With respect to the definition of a leakage function, it means that we assume $\mathsf{L}(C_\alpha, M, R) = \mathsf{L}'(C_\alpha, M) + \mathsf{L}''(R)$, *i.e.* the leakage function is the sum of a deterministic part and a random part. We note that this condition is expected to hold to a sufficient degree for our proof to remain meaningful in most applications.

We mention that a sound leakage probability distribution could be equivalently defined as giving rise to an asymptotic guessing entropy of one. It would yield a similar theorem. Also, Theorem 1 only considers a first order asymptotic success rate. A more general corollary is as follows:

Corollary. *For a given key class variable S, probability distribution $\Pr[\boldsymbol{L}_q|S]$ and under the same condition of independent leakages as in Theorem 1, the asymptotic success rate of order o equals one and the asymptotic guessing entropy equals o against this key class variable if and only if $\boldsymbol{H}^1_{s,s}$ is the o^{th} smallest value of the entropy matrix line $\boldsymbol{H}^1_{s,s^*}$, for all the key classes s.*

We omit the proof of the corollary for conciseness and because it follows exactly the same line of reasoning as Theorem 1. In the next subsection, we investigate the non-asymptotic meaning of the information theoretic metric. It allows discussing the extend to which it can be used to answer the question: "*how to compare two implementations?*" that is one motivation of the metric.

6.2 Non-asymptotic meaning of the conditional entropy

Let us write an exemplary conditional entropy matrix as follows:

$$\mathbf{H}^{q}_{s,s^*} = \begin{pmatrix} h_{0,0} & h_{0,1} & ... & h_{0,|\mathcal{S}|} \\ h_{1,0} & h_{1,1} & ... & h_{1,|\mathcal{S}|} \\ ... & ... & ... & ... \\ h_{|\mathcal{S}|,0} & h_{|\mathcal{S}|,1} & ... & h_{|\mathcal{S}|,|\mathcal{S}|} \end{pmatrix}$$

Theorem 1 states that if the diagonal values of this matrix are minimum for all key classes $s \in \mathcal{S}$, then these key classes can be asymptotically recovered by a Bayesian adversary. As a matter of fact, it gives rise to a binary conclusion about the leakage probability distributions. Namely, Theorem 1 answers the question: "*Do the leakages provide enough information to carry out an attack?*".

Let us now assume the answer is positive (*i.e.* there is enough information in the leakages) and denote each element $h_{s,s}$ as the entropy of a key class s. In this subsection, we are rather interested in the values of these entropy matrix elements. In particular, we aim to highlight the relation between these values and the effectiveness of a side-channel attack, measured with the success rate. Otherwise said, we are interested in the question: "*Does less entropy systematically implies a faster convergence towards a 100% success rate?*". Since general conclusions for arbitrary leakage distributions are not possible to obtain, our following strategy if to first consider simple Gaussian distributions and to extrapolate the resulting conclusions towards more complex cases.

We start with three definitions.

Definition 4. An $|\mathcal{S}|$-target side-channel attack is an attack where an adversary tries to identify one key class s out of $|\mathcal{S}|$ possible candidates.

Definition 5. An univariate (*resp.* multivariate) leakage distribution is a probability distribution predicting the behavior of one (*resp.* several) leakage samples.

Definition 6. A Gaussian leakage distribution is the probability distribution of a leakage function $\mathsf{L}(C_\alpha, M, R)$ such that $\mathsf{L}(C_\alpha, M, R) = \mathsf{L}'(C_\alpha, M) + \mathsf{L}''(R)$ and the random part of the leakages $\mathsf{L}''(R)$ is a normally distributed random noise[2] with mean zero and standard deviation σ.

Finally, since we plan to consider the entropy matrix $\mathbf{H}^{q}_{s,s^*}$ line by line and therefore, the entropy of the different key classes s, we also need a more specific definition of the success rate against a given key class s:

Experiment $\mathbf{Exp}^{\text{sc-kr-}o,s}_{\mathsf{E}_K,\mathsf{L}}$
$\mathbf{g} \leftarrow \mathsf{A}_{\mathsf{E}_k,\mathsf{L}}$;
if $s \in [g_1, \ldots, g_o]$ **then** return 1;
else return 0;

[2] Experimentally observed in a number of works, *e.g.* [22], Section 4.2.

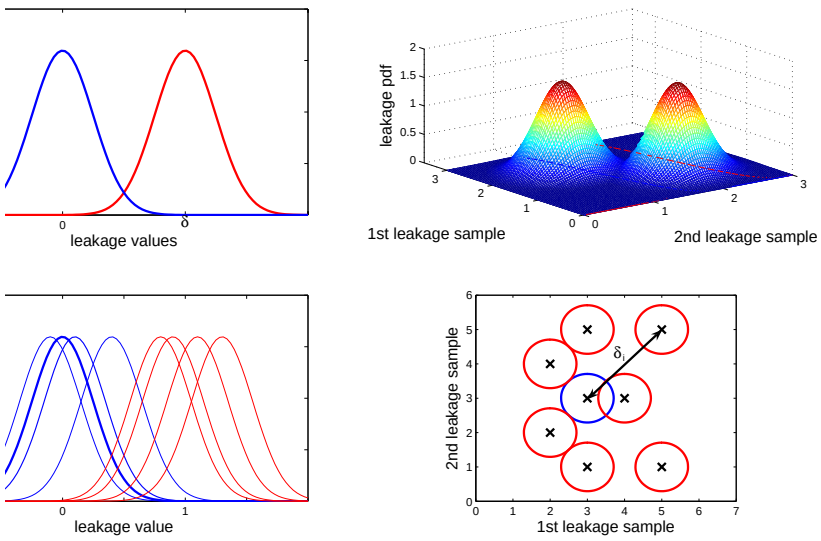


Fig. 2: Illustrative leakage probability distributions $\Pr[\mathbf{L}_q|S]$.

The o^{th} order success rate of the side-channel key recovery adversary $\mathsf{A}_{\mathsf{E}_K,\mathsf{L}}$ against a key class s (*i.e.* a realization of the variable S) is then defined as:

$$\mathbf{Succ}^{\text{sc-kr-}o,s}_{\mathsf{A}_{\mathsf{E}_K,\mathsf{L}}}(\tau, m, q) = \Pr\ [\mathbf{Exp}^{\text{sc-kr-}o,s}_{\mathsf{E}_K,\mathsf{L}} = 1] \tag{7}$$

Examples. Figure 2 illustrates several Gaussian leakage distributions. The upper left picture represents the univariate leakage distributions of a 2-target side-channel attack, each Gaussian curve corresponding to one key class s. The upper right picture represents the bivariate leakage distributions of a 2-target side-channel attack. The lower left picture represents the univariate leakage distributions of an 8-target side-channel attack. The same picture could also represent the leakage distributions of a 2-target side-channel attack protected by a masking scheme. In this latter context, each key class can give rise to four (equally likely) Gaussian events. Finally, the lower right picture represents the bivariate leakage distributions of an 8-target side-channel attack.

Lemma 1. *In a 2-target side-channel attack exploiting a sound univariate Gaussian leakage distribution, the entropy of a key class s is a monotonously decreasing function of the single query (and hence multi-queries) success rate against s.*

Proof. Let us consider the Gaussian univariate leakage distributions of the 2-target side-channel attack in the upper left part of Figure 2. Without loss of generality, we assume the correct key class to have mean zero and the wrong key class to have mean δ. Let us also assume a noise standard deviation σ. Let us

finally denote the probability density function of a Gaussian random variable X as $\mathsf{N}_x(\mu,\sigma) = \frac{1}{\sigma\sqrt{2\pi}} \cdot \exp\left(\frac{-(x-\mu)^2}{2\sigma^2}\right)$. According to the definitions of Section 5, the single query success rate and the entropy of the key class s can be written as:

$$\mathbf{Succ}^{\text{sc-kr-}s}_{\mathsf{A}_{\mathsf{E}_K,\mathsf{L}}}(\delta,\sigma) = \int_{-\infty}^{\delta/2} \mathsf{N}_x(0,\sigma)\ dx$$

$$h_{s,s}(\delta,\sigma) = -\int_{-\infty}^{+\infty} N_x(0,\sigma) \cdot \log_2 \frac{N_x(0,\sigma)}{N_x(0,\sigma) + N_x(\delta,\sigma)}\ dx$$

By applying a change of variable $u = x/\sigma$, we can rewrite:

$$\mathbf{Succ}^{\text{sc-kr-}s}_{\mathsf{A}_{\mathsf{E}_K,\mathsf{L}}}(\delta,\sigma) = \int_{-\infty}^{\frac{\delta}{2\sigma}} \mathsf{N}_u(0,1)\ du$$

$$h_{s,s}(\delta,\sigma) = -\int_{-\infty}^{+\infty} N_u(0,1) \cdot \log_2 \frac{N_u(0,1)}{N_u(0,1) + N_u(\delta/\sigma,1)}\ du$$

Defining a variable $z = \delta/\sigma$, we finally have:

$$\mathbf{Succ}^{\text{sc-kr-}s}_{\mathsf{A}_{\mathsf{E}_K,\mathsf{L}}}(z) = \int_{-\infty}^{z/2} \mathsf{N}_u(0,1)\ du$$

$$h_{s,s}(z) = -\int_{-\infty}^{+\infty} N_u(0,1) \cdot \log_2 \frac{N_u(0,1)}{N_u(0,1) + N_u(z,1)}\ du$$

Then, we just observe that $\mathbf{Succ}^{\text{sc-kr-}s}_{\mathsf{A}_{\mathsf{E}_K,\mathsf{L}}}$ and $h_{s,s}$ are respectively monotonously increasing and decreasing functions of z, which completes the proof. □

We now extrapolate this lemma towards the multivariate case. For conciseness purposes, we only provide a sketch for the following proof.

Lemma 2. *In a 2-target side-channel attack exploiting a sound multivariate Gaussian leakage distribution, with independent leakage samples having the same noise standard deviation, the entropy of a key class s is a monotonously decreasing function of the single query (and hence multi-queries) success rate against s.*

Proof sketch. We just move to a multivariate case such as the bivariate example of the upper right picture in Figure 2. Since the covariance matrix is diagonal, the success rate and the entropy only depend on the ratio between:

1. The Euclidean distance δ between the multivariate Gaussian mean values.
2. The leakage noise standard deviation σ.

By defining a variable $z = \delta/\sigma$, the same reasoning as in Lemma 1 applies. □

We now discuss the context of $|\mathcal{S}|$-target side-channel attacks.

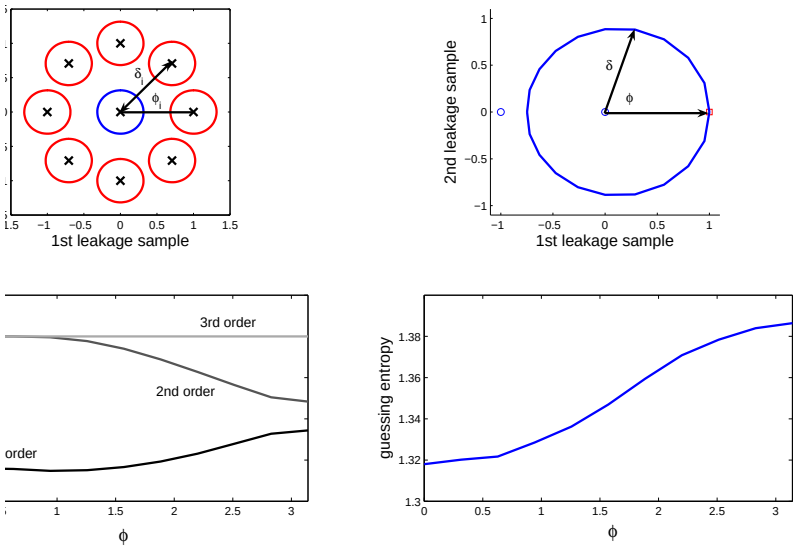


Fig. 3: Perfect leakage distribution and leakage distributions having constant conditional entropy with their associated success rates and guessing entropy .

The previous lemmas essentially state that (under certain conditions) the entropy and success rate in a 2-target side-channel attack only depend on the normalized distance δ/σ. It implies the straightforward intuition that more entropy means less success rate. Unfortunately, when moving to the $|\mathcal{S}|$-target case with $|\mathcal{S}| > 2$, such a perfect dependency does not exist anymore. It is easily observed in the lower right part of Figure 2 where the entropy and success rate not only depend on the normalized distances δ_i/σ but also on how the keys are distributed within the leakage space. Therefore, we now define a more specific context in which formal statements can be proven. Thereafter, we discuss the limitations of the entropy *vs.* success rate dependencies in a general setting.

Definition 7. A perfect Gaussian leakage distribution $\Pr[\mathbf{L}_q|s]$ for a key class s is a Gaussian leakage distribution with independent leakage samples having the same noise standard deviation such that the Euclidean distance between each key class candidate mean value and the correct key class candidate mean value is equal and the entropy of the key class s is maximum.

An example of perfect leakage distribution is in the upper left part of Figure 3.

Definition 8. We say that we have side-channel key equivalence for a key class s if the average probabilities $\mathbf{E}_{\mathbf{l}_q|s} \Pr[s^*|\mathbf{l}_q]$ and $\mathbf{E}_{\mathbf{l}_q|s} \Pr[s^{**}|\mathbf{l}_q]$ are identical for all wrong key candidates s^*, s^{**} such that $s^* \neq s^{**}$, $s^* \neq s$, $s^{**} \neq s$.

We note that key equivalence is a usual assumption in cryptanalysis. In the context of side-channel attacks, it is usually better verified when q increases.

It is straightforward to see that perfect leakage distributions imply side-channel key equivalence. By contrast, side-channel key equivalence does not imply perfect leakage distributions, as discussed in Appendix B. These definitions directly lead to our second theorem.

Theorem 2. *For any side-channel attack exploiting a perfect Gaussian leakage distribution, the entropy of a key class s is a monotonously decreasing function of the single query (and hence multi-queries) success rate against s.*

Proof sketch. In the context of perfect leakage distributions, the Euclidean distance between each key class candidate mean value and the correct key class candidate mean value is equal. Additionally, the distribution of the different key classes is fixed in the leakage space in order to maximize the the entropy of s. Therefore, the entropy of s and the success rate against s only depend on the ratio between the this Euclidean distance and the noise standard deviation which implies that Theorem 2 is a straightforward consequence of Lemma 2. □

Theorem 2 constitute our main positive result for the use of the conditional entropy as a comparison metric for different implementations. Interestingly, in the context of perfect leakage distributions, increasing the entropy involves a reduction of the success rates of every order. It yields the following corollary.

Corollary. *For any side-channel attack exploiting a perfect Gaussian leakage distribution, the entropy of a key class s is a monotonously increasing function of the single query (and hence multi-queries) guessing entropy against s.*

By contrast, in the most general context of non perfect leakage distributions, those general statements do not hold. In the remaining of this section, we point out two important facts that highlight the limitations of the conditional entropy.

Fact 1. *In the context of non-perfect and sound Gaussian leakage distributions, the constant entropy of a key class does not imply a constant success rate (or guessing entropy) against this key class.* This is illustrated in Figure 3 for a 3-key system. The upper right part of the figure shows different positions of the right key candidate leading to a constant entropy. They are obtained by changing the angle ϕ and reducing the distance δ accordingly, starting from a perfect distribution. The lower parts of the figure show the corresponding success rates and guessing entropy. As a matter of fact, they are not constant.

Fact 2. *There exist leakage distributions D_x and D_y such that the entropy corresponding to D_x is higher than the entropy corresponding to D_y and the success rate of every order for a Bayesian adversary exploiting D_x is also higher than the success rate for the same adversary exploiting D_y.* This is typically illustrated by the small reduction of the first order success rate in the lower left part of Figure 3, for $\phi \simeq 1$ and further discussed in Appendix C.

These facts essentially underline that there are no generally true dependencies between the conditional entropy and the success rate in the general setting.

6.3 Intuition of the metrics

In this section, we detail a number of important intuitions that can be extracted from the previous theory. We also discuss how they can be exploited in practical applications and highlight some of their limiting features.

Intuitions related to Theorem 1.

1.1 *Theorem 1 tells if an approximated leakage probability distribution is sound.* It is therefore a preliminary step in the analysis of any leakage function. If the approximated leakage probability distribution is sound, then the analysis related to Theorem 2 can be performed. If not, ...

1.2 *Theorem 1 tells if an implementation is secure.* Otherwise said, if one cannot build a sound approximation of the leakage probability distribution, even with intensive efforts, then the 1^{st}-order asymptotic success rate of the Bayesian side-channel adversary equals zero. It means that a certain level of security against key recovery attacks is achieved by the implementation. One must then look at the position of the correct key classes in the entropy matrix. If a class is rated second, the 2^{nd}-order asymptotic success rate against this class equals one. If it is badly rated, then physical and computational security may be both achieved by the implementation.

1.3 *Theorem 1 is not meaningful for simulated attacks,* since it (only) tells if a given leakage probability distribution is sound. As a matter of fact, in the context of simulated attacks, the leakage model was chosen by the evaluator and is expected to lead to a successful key recovery anyway. The only question is "*how good is this leakage model?*".

Intuitions related to Theorem 2.

2.1 *The conditional entropy allows comparing different implementations.* Theorem 2 states that under the assumption (introduced in Section 4) that template attacks are the strongest ones and within the theoretical limits detailed in the previous section, less entropy implies a more efficient Bayesian side-channel attack (*i.e.* an attack that requires less measurement queries). "How much more efficient?" has to be quantified with a security metric. In general, the better one approximates the leakage probability distribution $\Pr[\mathbf{L}_q|S]$, the better is the comparison of different implementations.

2.2 *Theorem 2 only applies to sound leakage distributions.* Intuitively, it means that comparing the conditional entropy provided by different leakage functions only make sense if the corresponding approximated leakage probability distributions lead to asymptotically successful attacks.

2.3 *The conditional entropy does not directly translate into actual security metrics.* This is an important feature of the information theoretic (or asymptotic security) metric that relates to the following observations:

(a) For a given amount of information leaked by a target implementation, different side-channel distinguishers could be considered (see Section 7). For example, the template attacks that closely relate to the definition of mutual information are not the most practical ones in terms of adversarial context. Suboptimal distinguishers are frequently used in practice.

(b) In the context of $|\mathcal{S}|$-target side-channel attacks with $|\mathcal{S}| > 2$, Theorem 2 assumes perfect leakage distributions that may not be observed in actual measurements. In most practical applications, more entropy implies that there is at least one order for the success rate to be decreased. But this order can only be found by investigating the actual leakage probability distributions and the complete entropy matrix[3]. Additionally, Fact 2 in Section 6.2 highlights that there exist counterintuitive implementations for which a higher conditional entropy $\mathrm{H}[S|\mathbf{L}_q]$ could also lead to a higher success rate of every order for the Bayesian adversary.

2.5 *Theorem 2 is meaningful for both simulated and experimental attacks.* It allows measuring the effectiveness of an abstract leakage model (independently of its practical significance) and the quality of an actual implementation.

Importantly, the observations 2.4 (a) and (b) emphasize that there is a certain degree of independence between the conditional entropy and the actual security metrics. It implies that the information theoretic metric always has to be completed with an actual security analysis (using the success rate or guessing entropy). First because it is the only way to determine the number of queries required for an attack to succeed. Second because in the context of non perfect leakage distributions, one has to verify that the investigated device does not fall into the previously mentioned counterintuitive category.

We mention that such counterintuitions only occur for small variations of the entropy and do not prevent the information theoretic metric to be the method of choice for the comparison of different physically observable implementations. In general and for all the assumptions introduced in this work, small deviations from the theoretical expectations may be observed in practice. However, we believe that the proposed metrics have more theoretical justification and allow a more concrete foundation for the analysis of side-channel attacks than most previously considered ad hoc criteria. As long as practical applications are close enough to the theoretical assumptions (as will be confirmed in Section 9 for actual examples), the intuitions described in this section hold. This motivates our following practice-oriented definitions and evaluation methodology.

[3] We mention that this is not a negative feature in itself. The aim of the information theoretic metric is to be independent of an adversary's algorithmic details. Being independent of the success rate order implies that the conditional entropy does not relate to one particular computational strategy either.

7 Practice-oriented definitions

From the definition of Section 5.1, a side-channel key recovery adversary is defined as an algorithm trying to recover a key class s from a number of queries to an implementation $(\mathsf{E}_K, \mathsf{L})$. In this section, we aim to give a more detailed description of such an adversary, considering the different steps in the side-channel attack illustrated in Figure 4. It actually consists in two phases that we respectively denote as the exploitation phase (which is the main core of the attack) and the preparation phase (which is the counterpart of the learning phase in artificial intelligence problems). We first describe the exploitation phase:

1. Input selection. The adversary selects its (possibly adaptive) q queries $\mathbf{x}_q$ (defined in Section 5.2) to the target device thanks to an algorithm I.
2. Values derivation. For each key class candidate s^*, the adversary predicts some values within the target device using an algorithm V. As a result, it obtains $|\mathcal{S}|$ vectors $\mathbf{v}_{s^*}^q = \mathsf{V}(s^*, \mathbf{x}_q)$ containing N_v-element predictions $v_{s^*}^i$, $i \in [1, q]$, where N_v is the number of internal values predicted per query.
3. (a) Leakages modelling. For each key class candidate s^*, the adversary models a part/function of the actual leakage emitted by the target device. Depending on the attacks, the model can be the approximated probability density function of a reduced set of leakage samples denoted: $\mathsf{M}(s^*, \tilde{\mathbf{l}}_q) = \hat{\Pr}[s^*, \tilde{\mathbf{l}}_q]$, as when using templates [2]. In this context, $\tilde{\mathbf{l}}_q = [\tilde{l}_1, \tilde{l}_2, \ldots, \tilde{l}_q]$ is the vector of leakage samples that are actually modelled by the adversary and $\tilde{l}_i$ is an N_m-element trace corresponding to the i^{th} query to the target device (N_m is the number of samples modelled per query). Or the model is a deterministic function (*e.g.* the Hamming weight) of the previously defined values: $\mathsf{M}(s^*, \mathbf{v}_{s^*}^q)$, as in correlation attacks [5]. We denote attacks exploiting a model as comparison attacks.

   (b) Leakages partitioning. If no leakage model is available, the adversary can define partitions (for each key class candidate s^*) according to a function of the previously defined values that we denote as $\mathsf{P}(s^*, \mathbf{v}_{s^*}^q)$. This is typically what was proposed in Kocher's original DPA in which the leakages are partitioned according to the value of one bit in the implementation [18]. We denote such attacks as partition attacks.
4. Leakages observation (or measurement). The adversary monitors the leakages of the target device containing the correct key class s. He stores these observations in the previously defined vector $\mathbf{l}_q$ containing N_l-sample traces l_i, $i \in [1, q]$, where N_l is the number of leakage samples stored per query.
5. Leakages reduction. In comparison attacks, the leakages and predictions possibly have different number of samples $N_l \neq N_m$. Therefore, a mapping R is used to transform the leakages such that $\mathsf{R}(l_i)$ is a N_m-sample trace. Additionally, the mapping possibly includes the post-processing of the traces, *e.g.* filtering, averaging. In the context of partition attacks, the reduction simply determines the leakage samples for which the partition will be tested.

6. Statistical test. For each key class candidate s^*, the adversary applies a statistical test T to either compare a model $\mathsf{M}(s^*, .)$ with the transformed leakages or to check if a partition $\mathsf{P}(s^*, .)$ is meaningful. It obtains an $|\mathcal{S}|$-element vector $\mathbf{g}_q = \mathsf{T}(\mathsf{M}(s^*, .), \mathsf{R}(\mathbf{l}_q))$ or $\mathbf{g}_q = \mathsf{T}(\mathsf{P}(s^*, .), \mathsf{R}(\mathbf{l}_q))$ containing the attack result, as in Section 5.1. Typical statistical tools include the difference of mean test [24], the correlation coefficient and the Bayesian classifier.
7. Decision: from the previous result, the adversary selects a key candidate (*i.e.* does a hard decision) or a list of key candidates (*i.e.* does soft decision) and stores them in a N_d-element vector $\mathbf{d}_q = \mathsf{D}(\mathbf{g}_q)$.
8. Offline computation: if a soft strategy is applied, the adversary finally tests the remaining candidates by a number of executions of the target algorithm.

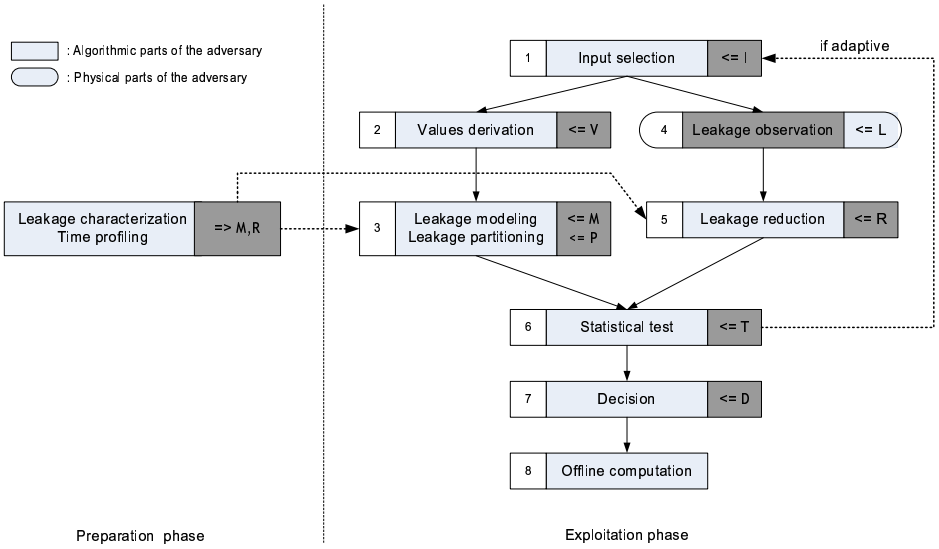


Fig. 4: Practical side-channel adversary.

Preliminary to the exploitation phase, the preparation phase produces the leakage model M and reduction mapping R, *e.g.* by profiling and characterizing the device. In the context of template attacks, the result of this preparation phase is the approximated leakage probability distributions of a reduced set of leakage samples $\hat{\Pr}[\tilde{\mathbf{L}}_q|S]$. However, simpler leakage models may take advantage of profiling and characterization as well. As a matter of fact, deriving these functions sometimes involves the same steps as the exploitation phase. But since the preparation can be performed once and then used in several exploitations, it is interesting to separate the complexities for both phases. We note that one could also design attacks in which preparation and exploitation are closely connected and therefore, only the overall complexities are relevant in the evaluations.

As previously detailed, a side-channel adversary is composed of a physical part modelled by the leakage function and an algorithmic part (sometimes denoted as the distinguisher) modelled by all the steps but the 4^{th} in Figure 4. Since the definition of a leakage function in [25] includes measurement setups, it depends on the adversary's ability to perform good measurements. In this practice-oriented view, the leakage function is not an oracle accessed by the adversary (as in the more theoretical view of Section 5.1) but a part of it. But this is not in contradiction with Micali and Reyzin: once the leakage function has been determined, the adversary's algorithmic part in Figure 4 can theoretically access it as an oracle which makes the previous definitions meaningful.

8 Evaluation methodology

Following the definitions in the previous sections, an evaluation methodology for side-channel attacks intends to analyze both the quality of an implementation and the strength of an adversary. It involves the five steps illustrated in Figure 5:

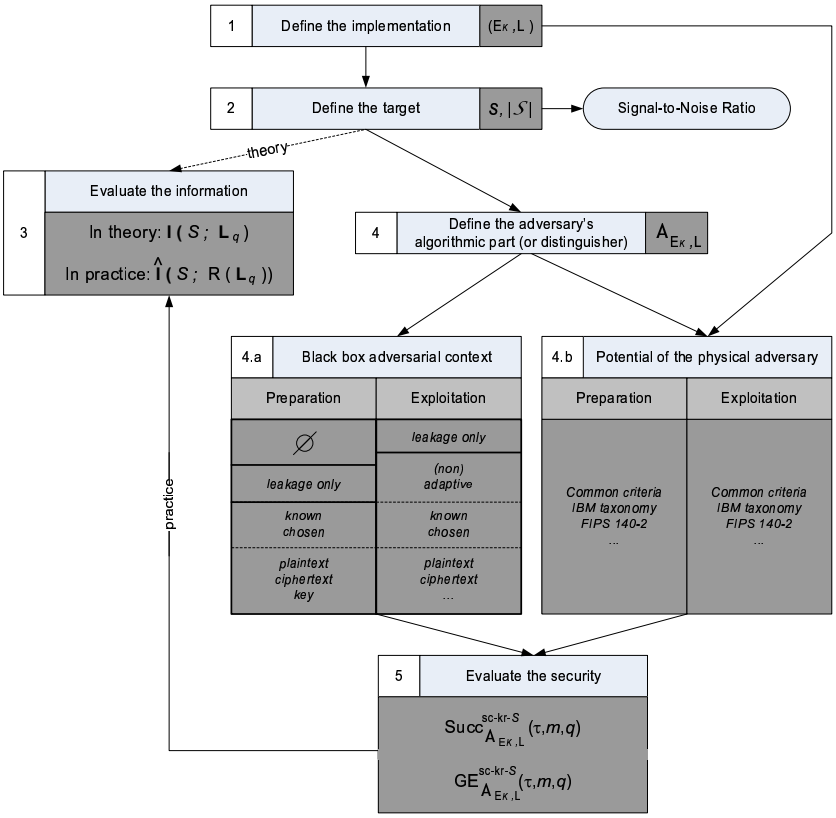


Fig. 5: Evaluation methodology for side-channel attacks.

1. We define the target implementation as modeled by Micali and Reyzin. That is, we define the combination of an abstract computer and a leakage function. In practice, the target implementation is a physical object, *e.g.* a smart card, FPGA or ASIC running some cryptographic primitive associated with some measurement setup. This determines the physical part of the adversary.
2. We define the target secret class s for the side-channel attack.
3. Once the target has been specified, we answer the first question in our evaluation, namely: "*What is the amount of information contained in the physical observations obtained from a leaking device?*". For this purpose, we use the mutual information. As previously mentioned and pictured on the figure arrows, in practice it can only be approximated from a number of leakage samples, through an actual (template-like) adversary's measurements. Alternatively and as a preliminary step in the evaluations, it can also be estimated theoretically by using a model $\mathsf{M}(s, .)$ (*e.g.* Hamming weight-based, SPICE simulations, ...) in place of the actual leakage function.
4. We complete the definition of the side-channel adversary (*i.e.* with its algorithmic part) and specify its black box adversarial context and physical potential. The latter are actually the most delicate part of an evaluation and will be shortly discussed in the following of the section.
5. We finally answer the second question in our evaluation, namely: "*How successfully can an adversary turn this information into a practical attack?*". For this purpose, we use the success rate of the side-channel key recovery adversary (or the guessing entropy) defined in Section 5.1.

Figure 5 again indicates that the information theoretic (or asymptotic security) metric can be used to measure an adversary's physical part while the actual security metrics are rather useful to evaluate its algorithmic part. Importantly, the mutual information is an average metric computed over an uniform key space. It is meaningful to compare implementations since this goal requires to be independent of a given adversary. By contrast, worst case behaviors are typically dependent on the adversarial strategy (*e.g.* adaptive). They can consequently be quantified with the security metrics in order to discriminate different distinguishers for a given implementation. Additionally, it is often interesting to define a Signal-to-Noise Ratio (SNR) in order to determine the amount of noise in the physical observations. Since noise insertion is a generic countermeasure to improve resistance against side-channel attacks, it can be used to plot the information theoretic and security metrics with its respect.

As already mentioned, the most delicate part in this methodology is to describe the black box adversarial context and physical potential of a side-channel adversary. Black box assumptions relate to the adversary's abilities to monitor and tamper with the primitives inputs and outputs. For this purpose, we refer to classical notions (*e.g.* non adaptive/adaptive, known/chosen, plaintext/ciphertext, ...) with the additional possibility to have known or chosen keys during the preparation phase. The physical potential of an adversary relates to its level of expertise, the cost of its equipment, ... Since quantifying such potential is typically the tasks assigned the standardization bodies, we refer to

the common criteria [9] and FIPS 140-2 documents [12] (or alternatively to the IBM taxonomy [1]) for these purposes. In general, the benefit of the presently introduced model is not to solve these practical issues but to state the side-channel problem in a sound framework for its analysis. Namely, it is expected that the proposed security and information theoretic metrics can be used for the fair analysis, evaluation and comparison of any physical implementation or countermeasure against any type of side-channel attack.

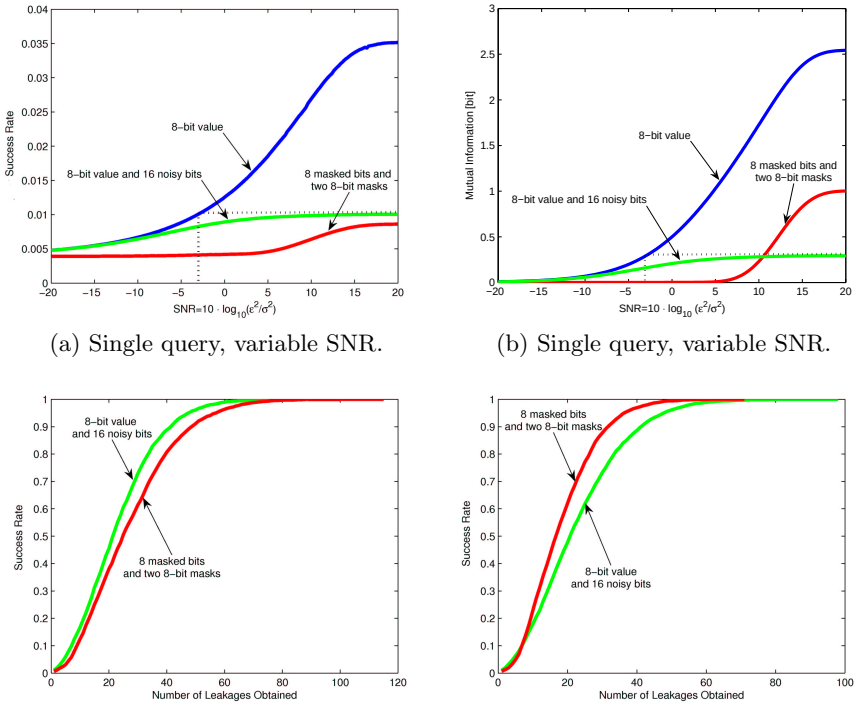
Let us finally mention that any information theoretic or security analysis of actual devices has to come with a good statistical evaluation with confidence intervals for the estimated evaluation criteria. Section 4.6 in [22] provides a good introduction to these issues. By contrast, the evaluation of simulated attacks where sums are turned into integrals do not require statistical sampling.

9 Applications of the model

Before to conclude this paper, this section aims to come back on our initial claims. Namely, we summarize the extend to which the proposed metrics and methodology allow the comparison of implementations and adversaries.

Starting with the comparison of different implementations, our results essentially suggest that an implementation X is "better" than an implementation Y if it is more secure against the (strongest possible) Bayesian side-channel adversary (formally defined in Section 6.1). But this raises the question: "*Why do we need two metrics for this purpose?*". Otherwise said: cannot one just compare the success rates of this Bayesian adversary? The main answer to this question relates to the independence of the mutual information on the number of queries q. As Theorem 1 in Section 6 details, the mutual information allows determining if there is enough information in the leakages, by exploiting a fixed number of queries (in general, q=1). When comparing two implementations, this independence is a very desirable property that is nicely illustrated in [30].

In this reference, the effectiveness of two countermeasures against side-channel attacks (namely, noise addition and masking) is evaluated with our methodology. The upper parts of Figure 6 illustrate the results of this comparison: both the success rate and the mutual information are computed for the two countermeasures, in function of the measurement noise (quantified with an SNR). The single-query success rate (Figure 6.a) suggests that the noisy implementation is always easier to target than the masked one. By contrast, the mutual information (Figure 6.b) suggests that for high SNRs, the masked information leaks more information than the noisy one; the conclusion is inverted when decreasing the SNR. Interestingly, the mutual information highlights a certain SNR value for which masking becomes a better countermeasure than noise addition (corresponding to the intersection of the curves in Figure 6.b). It is confirmed by the success rate when you increase the number of queries, as illustrated in the lower parts of the figure. Left of the intersection (*e.g.* SNR=10), masking is a better countermeasure, right of the intersection (*e.g.* SNR=11), noise addition is!



(a) Single query, variable SNR.

(b) Single query, variable SNR.

(c) SNR=10, variable q.

(d) SNR=11, variable q.

Fig. 6: Success rate and mutual information for the comparison of two countermeasures.

In summary, if the analyzed implementations have a leakage probability distribution sufficiently close to perfect as in the previous example, the mutual information allows their fair comparison. It gives rise to design criteria (*e.g.* the intersection of the curves in Figure 6.b) that are not directly accessible when computing success rates (although computing success rates can confirm these criteria). If the leakage probability distributions are not sufficiently perfect, the mutual information is useful to determine if an attack of a given order is asymptotically successful, before performing the attack and evaluate its efficiency.

As far as implementation contexts are concerned, the security analysis in [30] considers an abstract (Hamming weight-based) leakage model. In [20], an information theoretic analysis is performed in order to compare different side-channel resistant logic styles, exploiting gate-level simulated leakage traces. This analysis is extended in [21] towards multivariate leakages, more complex circuits and combined with a security analysis. In [31], a practical information theoretic and security evaluation is applied to actual measurements obtained from different circuit-level side-channel countermeasures. Those results illustrate the applicability of the framework to a wide variety of contexts and platforms.

Finally, the comparison of different side-channel adversaries (or distinguishers) using our security metrics is extensively exemplified in [32].

10 Conclusions and open problems

A unified framework for the analysis of cryptographic implementations against side-channel key recovery is introduced as a specialization of Micali and Reyzin's "physically observable cryptography" paradigm. It is based on a theoretical model in which the effect of practically relevant leakage functions is evaluated with a combination of security and information theoretic metrics. The framework allows both the practical evaluation of actual side-channel attacks and the understanding of the underlying tradeoffs in physically observable cryptography, namely "*flexibility vs. efficiency*" and "*information vs. computation*".

The flexibility *vs.* efficiency tradeoff typically relates to the adversarial context considered. As a matter of fact, an adaptive adversary using a carefully profiled leakage model will generally exploit the available physical information (much) more efficiently than a non-adaptive one, using a non profiled leakage model. However, simpler prediction models do not only involve a sub-optimal information extraction from side-channel traces. They may also be more easily reproducible to different devices. As a typical example, Kocher's original Differential Power Analysis only assumes that somewhere in a physical observation, the leakage will depend on a single bit value. The simplicity of this assumption made it straightforwardly applicable to a wide range of platforms, without any adaptation. Correlation attacks, template attacks or stochastic models are trading some of this flexibility for a more efficient information extraction.

By contrast, the information *vs.* computation tradeoff rather relates to the computational strategy considered. As a matter of fact, for comparable amounts of side-channel queries q, a soft strategy trying to extract a list of key candidates including the correct one will generally have a higher success rate than a hard strategy, trying to extract the correct key value only. However, if this list of candidates can be tested with some computational power, it can be turned into a successful key recovery. Otherwise said, a lack of information can be overcome by a more computationally intensive adversarial strategy.

In summary, as an interface between theory and practice, our framework allows putting forward properly quantified weaknesses in physically observable devices. As a consequence, these weaknesses can be either feeded back to hardware designers in order to reduce the physical leakages or sent to cryptographic designers in order to conceive schemes that can cope with physical leakage.

Open questions derive from this model in different directions. A first one relates to the best exploitation of large side-channel traces, *i.e.* to the construction of (ideally) optimal distinguishers. This requires to investigate the best heuristics to deal with high dimensional leakage data in order to confirm the model assumption that template attacks are the strongest form of side-channel attacks. A second one relates to the investigation of stronger security notions than side-channel key recovery. That is, the different security notions considered in the black box model (*e.g.* the undistinguishability from an idealized primitive) should be considered in the physical world, as initiated in [25]. A third

directions relates to the construction of implementations with provable (or arguable) security against side-channel attacks, *e.g.* as proposed in [26]. Finally the extension to other physical adversaries is a long term scope for cryptographic research. With this respect, the present work appears as a complement to other approaches for modelling physical attacks such as [13, 16, 17].

Acknowledgements. The authors would like to thank Christophe Petit, Leonid Reyzin and the different reviewers of which the comments improved the presentation of this work. François-Xavier Standaert is a postdoctoral researcher of the Belgian Fund for Scientific Research (FNRS).

References

1. D.G. Abraham, G.M. Dolan, G.P. Double, J.V. Stevens, *Transaction Security System*, in IBM Systems Journal, vol 30, num 2, pp 206- 229, 1991.
2. D. Agrawal, B. Archambeault, J. Rao, P. Rohatgi, *The EM Side-Channel(s)*, CHES 2002, LNCS, vol 2523, pp 29-45, Redwood City, CA, USA, August 2002.
3. C. Archambeau, E. Peeters, F.-X. Standaert, J.-J. Quisquater, *Template Attacks in Principal Subspaces*, CHES 2006, Lecture Notes in Computer Science, vol 4249, pp. 1–14, Yokohama, Japan, October 2006.
4. A. Biryukov, C. De Cannière, M. Quisquater, *On Multiple Linear Approximations*, Crypto 2004, LNCS, vol 3152, pp 1-22, Santa Barbara, CA, USA, August 2004.
5. E. Brier, C. Clavier, F. Olivier, *Correlation Power Analysis with a Leakage Model*, CHES 2004, LNCS, vol 3156, pp 16-29, Boston, MA, USA, August 2004.
6. C. Cachin, *Entropy Measures and Unconditional Security in Cryptography*, PhD Thesis, ETH Dissertation, num 12187, 1997.
7. S. Chari, J. Rao, P. Rohatgi, *Template Attacks*, CHES 2002, LNCS, vol 2523, pp 13-28, CA, USA, August 2002.
8. *http://www.chesworkshop.org/*.
9. *Application of Attack Potential to Smart Cards*, Common Criteria Supporting Document, Version 1.1, July 2002, http://www.commoncriteriaportal.org
10. T.M. Cover, J.A. Thomas, *Information Theory*, Wiley and Sons, New York, 1991.
11. ECRYPT Network of Excellence in Cryptology, *The Side-Channel Cryptanalysis Lounge* , http://www.crypto.ruhr-uni-bochum.de/en_sclounge.html.
12. FIPS 140-2, *Security Requirements for Cryptographic Modules*, Federal Information Processing Standard, NIST, U.S. Dept. of Commerce, December 3, 2002.
13. R. Gennaro, A. Lysyanskaya, T. Malkin, S. Micali, T. Rabin, *Algorithmic Tamper-Proof Security: Theoretical Foundations for Security Against Tampering*, TCC 2004, LNCS, vol 2951, pp 258-277, Cambridge, MA, USA, February 2004.
14. B. Gierlichs, K. Lemke, C. Paar, *Templates vs. Stochastic Methods*, CHES 2006, LNCS, vol 4249, pp 15-29, Yokohama, Japan, October 2006.
15. L. Goubin, J. Patarin, *DES and Differential Power Analysis*, CHES 1999, LNCS, vol 1717, pp 158-172, Worcester, MA, USA, August 1999.
16. Y. Ishai, A. Sahai, D. Wagner, *Private Circuits: Securing Hardware against Probing Attacks*, Crypto 2003, Lecture Notes in Computer Science, vol 2729, pp 463-481, Santa Barbara, CA, USA, August 2003.
17. Y. Ishai, M. Prabhakaran, A. Sahai, D. Wagner, *Private Circuits II: Keeping Secrets in Tamperable Circuits*, Eurocrypt 2006, LNCS, vol 4004, pp 308-327, St. Petersburg, Russia, May 2006.

18. P. Kocher, J. Jaffe, B. Jun, *Differential Power Analysis*, Crypto 1999, LNCS, vol 1666, pp 398-412, Santa-Barbara, CA, USA, August 1999.
19. B. Köpf, D. Basin, *an Information Theoretic Model for Adaptive Side-Channel Attacks*, CCS 2007, Alexandria, VA, USA, October 2007.
20. F. Macé, F.-X. Standaert, J.-J. Quisquater, *Information Theoretic Evaluation of Side-Channel Resistant Logic Styles*, CHES 2007, LNCS, vol 4727, pp 427-442, Vienna, Austria, September 2007.
21. F. Macé, F.-X. Standaert, *A Simulation-Based Information Theoretic and Security Evaluation of Side-Channel Resistant Logic Styles*, available on: http://www.dice.ucl.ac.be/ fstandae/tsca/.
22. S. Mangard, E. Oswald, T. Popp, *Power Analysis Attacks*, Springer, 2007.
23. J.L. Massey, *Guessing and Entropy*, IEEE International Symposium on Information Theory, pp 204, Trondheim, Norway, June 1994.
24. T.S. Messerges, E.A. Dabbish, R.H. Sloan, *Examining Smart-Card Security under the Threat of Power Analysis Attacks*, IEEE Transactions on Computers, vol 51, num 5, pp 541-552, May 2002.
25. S. Micali, L. Reyzin, *Physically Observable Cryptography*, TCC 2004, LNCS, vol 2951, pp 278-296, Cambridge, MA, USA, February 2004.
26. C. Petit, F.-X. Standaert, O. Pereira, T.G. Malkin, M. Yung, *A Block Cipher based PRNG Secure Against Side-Channel Key Recovery*, Cryptology ePrint Archive, Report 2007/356, 2007, http://eprint.iacr.org.
27. W. Schindler, K. Lemke, C. Paar, *A Stochastic Model for Differential Side-Channel Cryptanalysis*, CHES 2005, Lecture Notes in Computer Science, vol 3659, pp 30-46, Edinburgh, Scotland, September 2005.
28. C.E. Shannon, *A Mathematical Theory of Communication*, Bell System Technical Journal, vol 27, pp 379-423 and 623-656, July and October, 1948.
29. C.E. Shannon, *Communication theory of secrecy systems*, Bell System Technical Journal, vol 28, pp 656-715, October 1949.
30. F.-X. Standaert, E. Peeters, C. Archambeau, J.-J. Quisquater, *Towards Security Limits in Side-Channel Attacks*, CHES 2006, LNCS, vol 4249, pp. 30–45, Yokohama, Japan, October 2006. Available on: http://eprint.iacr.org/2007/222.
31. F.-X. Standaert, C. Archambeau, F. Macé, *A Practical Information Theoretic and Security Evaluation of Side-Channel Resistant Logic Styles*, available on http://www.dice.ucl.ac.be/ fstandae/tsca/.
32. F.-X. Standaert, *Partition vs. Comparison Side-Channel Distingsuishers*, available on http://www.dice.ucl.ac.be/ fstandae/tsca/.
33. K. Tiri, M. Akmal, I. Verbauwhede, *A Dynamic and Differential CMOS Logic with Signal Independent Power Consumption to Withstand Differential Power Analysis on Smart Cards*, ESSCIRC 2003, Estoril, Portugal, September 2003.

A Micali & Reyzin's informal axioms

Axiom 1. Computation and only computation leaks information.

That is, we assume that it is possible to store some secret information securely in a cryptographic device. No leakages will compromise this secret as long as it is not used in any computation. This implies that probing attacks are out of the scope of our analysis and we rely on physical protections to prevent them.

Axiom 2. The same computation leaks different information on different computers.

In other words, an algorithm is an abstraction: a set of general instructions whose physical implementation may vary. As a result, the same elementary operation may leak different information on different platforms.

Axiom 3. The information leakage depends on the chosen measurement.

The amount of information that is recovered by an adversary during a side-channel attack depends on the measurement process, that possibly introduces some randomness due to the presence of noise.

Axiom 4. The information leakage is local.

In other words, the maximum amount of information that may be leaked by a physically observable device is the same in any execution of the algorithm with the same inputs, since it relates to the target device's internal configuration.

Axiom 5. All the information leaked through physical observations can be efficiently computed from a target device's internal configuration.

That is, given a physical computer, the information leakage is a polynomial time computable function of (1) the computer's internal configuration (because of Axiom 4), (2) the chosen measurement (because of Axiom 3), and possibly (3) some randomness outside anybodys control (also because of Axiom 3).

We note that, from the practical point of view, these axioms may not reflect the entire physical phenomenons observed. For example, as far as Axiom 1 is concerned, volatile memories such as RAMs regularly require a small amount of energy to refresh their values and this could be used to mount a side-channel attack. However, such leakages are significantly more difficult to exploit than computational leakages. Our expectation is therefore that these axioms approximates the physical reality to a sufficient degree.

B Key equivalence $\not\Rightarrow$ perfect leakage distributions

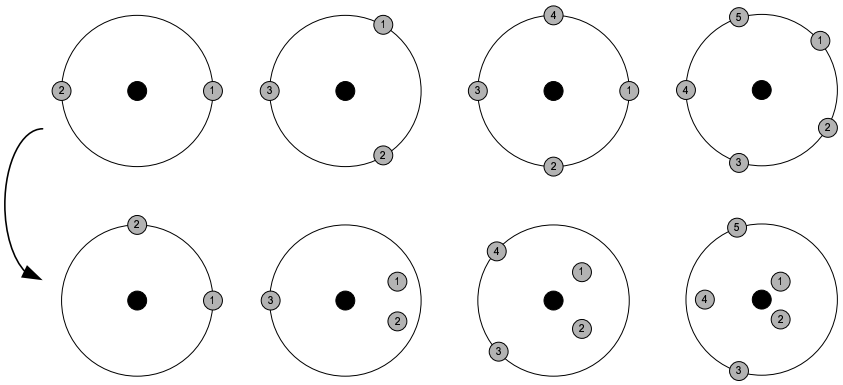


Fig. 7: Non-perfect leakage distributions with key equivalence.

We illustrate that for bivariate leakage distributions, it is always possible to build a non-perfect distribution with key equivalence. Figure 7 illustrates this claim for $|\mathcal{S}| \in [3, 4, 5, 6]$. Perfect leakage distributions are in the upper part of the figure. Non-perfect leakage distributions with key equivalence are in its lower part. As previously, we denote the position of the Gaussian mean values with their distance δ_i and angle ϕ_i with respect to the correct key class.

1. In the 3-key example (left part of the figure), one can change the position of, *e.g.* s_2 by keeping δ_2 constant and only changing the angle ϕ_2. The resulting leakage distribution is not perfect and still has key equivalence.
2. For any larger $|\mathcal{S}|$. We can transform the distributions as follows. Let us say we move k_1 by changing ϕ_1. The first step in the transform is to change the other angles such that we again obtain an axial symmetry among the X axis. Then, the distances δ_1 and δ_2 have to be reduced identically in order to have $\mathbf{E}_{\mathbf{l}_q} \Pr[s_1|\mathbf{l}_q] = \mathbf{E}_{\mathbf{l}_q} \Pr[s_2|\mathbf{l}_q]$ and this average probability equal to the average probability of another pair of symmetrical classes (*e.g.* s_3 and s_5 in the right part Figure 7). Finally, we successively reduce the distances of any other pair of symmetric classes in order to have key equivalence[4].

Note finally that key equivalence combined with equal normalized distances δ_i/σ for all key class candidates implies perfect leakage distributions.

[4] The distances δ_i have been increased in Figure 7 to emphasize their variations.

C More entropy sometimes means more success rate

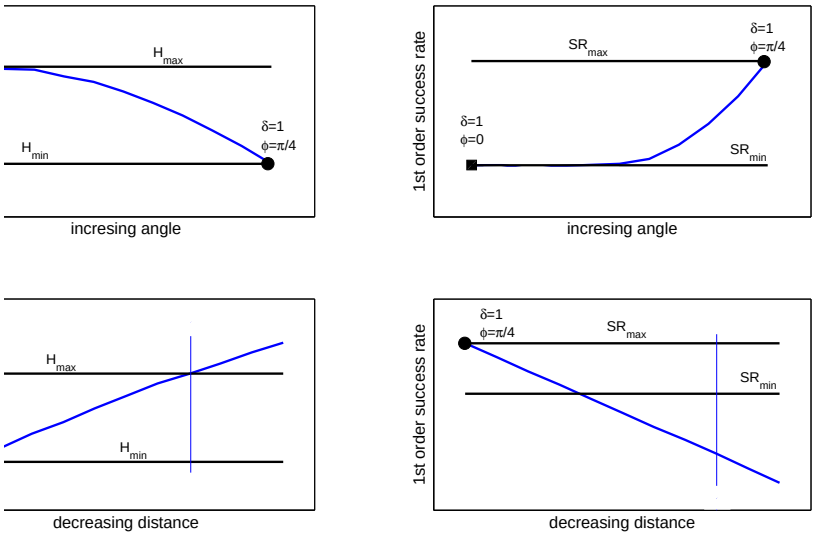


Fig. 8: Finding leakage distributions with more entropy and more success rate.

Figure 8 illustrates the search for leakage distributions having both less entropy and less success rate of every order for the Bayesian adversary. It considers a 3-key system such as the one in the upper right part of Figure 3.

Let us assume we first modify the angle ϕ from 0 to $\pi/4$, without changing the distance δ. As illustrated in the upper part of Figure 8, it implies both a reduction of the entropy and an increase of the (first order) success rate. We store the maximum and minimum value for the entropy. Then, we reduce the distance δ without changing the angle ϕ. As illustrated in the lower part of the figure, it implies both an increase of the entropy and a reduction of the (first order) success rate. We store the distance were the entropy is back to its initial value. This distance is represented in the lower parts of the figure with a vertical line. But at this distance, the success rate is already smaller than its initial value. So, there are points were reducing the entropy also reduces the (first order) success rate. Since changing the angle straightforwardly implies a reduction of the second order success rate as well, we have a reduction of the success rate for every meaningful order (the third order success rate being stuck at one).

D Notation index

In general and excepted if explicitly mentioned otherwise, capital letters represent variables X, small letters represent particular values of the variables x and sets or alphabets are denoted with calligraphic letters $\mathcal{X}$. Bold letters denote vectors and matrices $\mathbf{X}$. Sans serif fonts are used for algorithms and functions X,x.

α	Abstract computer/cryptographic primitive	pp 4, sec 2
α_i	Virtual Memory Turing Machine (VTM)	pp 4, sec 2
$\mathsf{A}_{\mathsf{E}_K,\mathsf{L}}$	Side-channel key recovery adversary	pp 8, sec 5.1
δ	Key classification function	pp 8, sec 5.1
D	Decision function	pp 21, sec 7
$\mathsf{E}_K(.)$	Family of cryptographic abstract computers indexed by a variable key K	pp 8, sec 5.1
$\mathbf{GE}^{\text{sc-kr-}S}_{\mathsf{A}_{\mathsf{E}_K},\mathsf{L}}$	Guessing entropy of a side-channel key recovery adversary against a key class variable S	pp 9, sec 5.1
$\mathbf{H}^q_{s,s^*}$	Conditional entropy matrix	pp 10, sec 5.2
$\mathrm{H}[S\|\mathbf{L}_q]$	Conditional entropy	pp 10, sec 5.2
I	Input selection algorithm	pp 20, sec 7
$\mathrm{I}(S;\mathbf{L}_q)$	Mutual information	pp 10, sec 5.2
$\mathsf{L}(C_\alpha, M, R)$	Leakage function	pp 4, sec 2
$\mathbf{L}_q$, $\mathbf{l}_q$	Side-channel observations vector	pp 10, sec 5.2
$\mathsf{M}(s^*,.)$	Leakage model for a key class s^*	pp 20, sec 7
$\mathsf{P}(s^*,.)$	Leakage partition for a key class s^*	pp 20, sec 7
φ	Physical computer/cryptographic implementation	pp 4, sec 2
φ_i	Physical Virtual Memory Turing Machine	pp 4, sec 2
$\Pr[s\|\mathbf{l}_q]$	Probability of a key class s given a leakage $\mathbf{l}_q$	pp 10, sec 5.2
$\Pr[S\|\mathbf{L}_q]$	Probability distribution of a key class variable S given a leakage variable $\mathbf{L}_q$	pp 7, sec 4
R	Leakage reduction mapping	pp 20, sec 7
$\mathbf{Succ}^{\text{sc-kr-}o,S}_{\mathsf{A}_{\mathsf{E}_K},\mathsf{L}}$	o$^{\text{th}}$-order success rate of a side-channel key recovery adversary against a key class variable S	pp 8, sec 5.1
$\mathbf{Succ}^{\text{sc-kr-}o,s}_{\mathsf{A}_{\mathsf{E}_K},\mathsf{L}}$	o$^{\text{th}}$-order success rate of a side-channel key recovery adversary against a key class s	pp 14, sec 6.2
T	Statistical test in a side-channel attack	pp 21, sec 7
V	Values derivation algorithm	pp 20, sec 7