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

- extended version -

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 3.0, February 23, 2009.

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 makes a step in this direction and proposes a framework for the analysis of cryptographic implementations that includes a theoretical model and an application methodology. The model is based on 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 information theoretic and security metrics, measuring the quality of an implementation and the strength of an adversary, respectively. 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 attacks. 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 answered in a sound fashion.

1 Introduction

Traditionally, cryptographic algorithms provide security against an adversary who has only black box access to cryptographic devices. However, such a model does not always correspond to the realities of physical implementations. During the last decade, it has been demonstrated that targeting actual hardware rather than abstract algorithms may lead to very serious security issues. In this paper, we investigate the context of side-channel attacks, in which adversaries are enhanced with the possibility to exploit physical leakages such as power consumption [22] or electromagnetic radiation [2, 15]. A large body of experimental work has been created on the subject and although numerous countermeasures are proposed in the literature, protecting implementations against such attacks is usually difficult and expensive. Moreover, most proposals we are aware of only

⋆ Associate researcher of the Belgian Fund for Scientific Research (FNRS - F.R.S.)

increase the difficulty of performing the attacks, but do not fundamentally prevent them. Eventually, due to the device-specific nature of side-channel attacks, the comparison of their efficiency and the evaluation of leaking implementations are challenging issues, *e.g.* as mentioned in [25], page 163.

Following this state-of-the art, our work is mainly motivated by the need of having sound tools (*i.e.* a middle-ware between the abstract models and the concrete devices) to evaluate and compare different implementations and adversaries. 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 in clear contrast with the combination of ad hoc solutions relying on specific ideas designers have in mind. For example, present techniques 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*". But such claims are of limited interest since an unsuccessful attack may theoretically be due both to the quality of the target device or to the ineffectiveness of the adversary. The results in this paper aim to discuss the extent to which more meaningful (adversary independent) statements can be claimed such as: "*An implementation X is better than an implementation Y*". Similarly, when comparing different adversaries, 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 in practice, recovering a small set of key candidates including the correct one after a low number of measurement queries may be more critical for the security of an actual system than recovering the key itself after a high number of measurement queries (*e.g.* further isolating a key from a list can employ classical cryptanalysis techniques exploiting black box queries). The results in this paper aim at providing tools that help claiming more flexible statements and can capture various adversarial strategies[1].

Quite naturally, the previous goals imply the need of a sound model for the analysis of side-channel attacks. But perhaps surprisingly (and to the best of our knowledge), there have been only a few attempts to provably address physical security issues. A significant example is the work of Micali and Reyzin who initiated an analysis of side-channels taking the modularity of physically observable computations into account. The resulting model in [28] is very general, capturing almost any conceivable form of physical leakage. However and as observed by the authors themselves, this generality implies that 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.

[1] We note that if obtained through statistical sampling, these claims have to come with a certain confidence interval. This is frequently neglected concern in the present literature on side-channel attacks where only single experiments are often provided.

In the present work, we consequently give up a part of this generality and concentrate on current attacks (*i.e.* key recovery) and adversaries (*i.e.* statistical procedures to efficiently discriminate the key), trying to keep a sound and systematic approach aside these points. For this purpose, we first separate the implementation issue (*i.e.* "*how good is my implementation?*") and the adversarial issue (*i.e.* "*how strong is my adversary?*") in the physically observable setting. We believe that the methodological division of both concerns brings essential insights and avoids previous confusions in the analysis of side-channel attacks. As a consequence, we introduce two different types of evaluation metrics. First, an information theoretic 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 propose candidates for these metrics and show that they allow comparing different implementations and adversaries. We also demonstrate important connections between them in the practically meaningful context of Gaussian leakage distributions and discuss their intuitive meaning. Eventually, we move from formal definitions to practice-oriented definitions in order to introduce a unified evaluation methodology for side-channel key recovery attacks. We also provide an exemplary application of the model and discuss its limitations.

Related works include a large literature on side-channel issues, ranging from attacks to countermeasures and including statistical analysis concerns. The side-channel lounge [13], DPA book [25] and CHES workshops [9] provide a good list of references, a state-of-the art view of the field and some recent developments, respectively. Most of these previous results can be re-visited in the following framework in order to improve their understanding. The goal of this paper is therefore to facilitate the interface between theoretical and practical aspects in physically observable cryptography. We mention that in parallel to our work, the models in [3, 23] consider a restricted context of noiseless leakages. They allow deriving formal bounds on the efficiency of certain attacks but are not aimed to analyze actual devices (that always have to deal with noise) which is our main goal. Finally, [29] initiated a study of forward secure cryptographic constructions with rigorous security analysis of side-channel attacks. [12, 30] then proposed similar constructions in a more general setting and standard model. These works exploit assumptions such as bounded adversaries or leakages of which the validity can be measured for different devices thanks to the methodology in this paper.

Finally, our analysis employs ideas from the classical communication theory [11, 32, 33]. But whereas source and channel coding attempt to represent the information in an efficient format for transmission, cryptographic engineers have the opposite goal to make their circuit's internal configurations unintelligible to the outside world. This analogy provides a rationale for our metrics. Note that different measures of uncertainty have frequently been used in the cryptographic literature to quantify the effectiveness of various attacks, *e.g.* in [7]. 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 implementations with an information theoretic metric (conditional entropy) and to evaluate attacks and adversaries with security metrics (success rates or guessing entropy). This allows us to consider first implementations as non-adversarial information emitting objects where keys are randomly chosen, and then adversaries which operate under certain (computational and access) restrictions on top of the implementations. This duality enables our model to be reflective of the situation in the real world and therefore to be useful beyond theoretical analysis, *i.e.* applicable to any simulated or actual lab data, for various cryptographic algorithms. In summary, we provide a practical model for side-channel attacks. The model implies the need of two different types of metrics and these metrics lead to a unified evaluation methodology. Then, in order to allow the practical application of the proposed framework, we suggest candidates for the metrics and discuss their relevance.

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 defines our evaluation metrics formally. Section 5 discusses the practical limitations in the application of our model and metrics to actual devices. Section 6 demonstrates some important connections between our evaluation metrics, with their intuitive consequences. Section 7 describes an exemplary application of the model. Section 8 elaborates an evaluation methodology for physically observable cryptographic devices. Finally, conclusions and open problems are in Section 9. Additionally, a practice-oriented definition of a side-channel adversary is in appendix.

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. First, an *abstract computer* was defined in [28] as a collection of special Turing machines, which invoke each other as subroutines and share a common memory. Each member of the collection is denoted as an *abstract virtual-memory Turing machine* (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 residing in some virtual memory. Abstract computers and VTMs are not physical devices: they only represent logical computation and may have different physical realizations. Then, to model the physical leakage of any particular instantiation of an abstract computer, the notion of *physical computer* was introduced. A physical computer is a pair (α, L), where L is a *leakage function*. In this definition, the relation between an abstract computing machine α and its physical realization is only determined by the leakage function that depends on 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 in the measurements.

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 explain 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 that models all the specificities of the side-channel and 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* an algorithm that can query the implementation to get the leakage function results in addition to the traditional black-box access. It has the goal to defeat a given security notion (*e.g.* key recovery) within certain computational bounds and capabilities.

Figure 1 suggests that, similarly to the classical communication theory, two aspects have to be considered (and quantified) in physically observable cryptography. First, actual implementations leak information, independently of the adversary exploiting it. The goal of our information theoretic metric is to measure the side-channel leakages in order to give a sound answer to the question: "*how to compare different implementations?*". Second, an adversary analogous to a specific decoder exploits these leakages. The goal of our security metrics is to measure the extent to which this exploitation efficiently turns the information available into a key recovery. Security metrics are the counterpart of the Bit-Error-Rate in communication problems and aim to answer the question: "*how to compare different adversaries?*". Interestingly, the figure highlights the difference between an actual adversary (of which the goal is simply to recover some secret data) and an evaluator (of which the goal is to analyze and understand the physical leakages). For example, comparing different implementations with an information theoretic metric is only of interest for an evaluator.

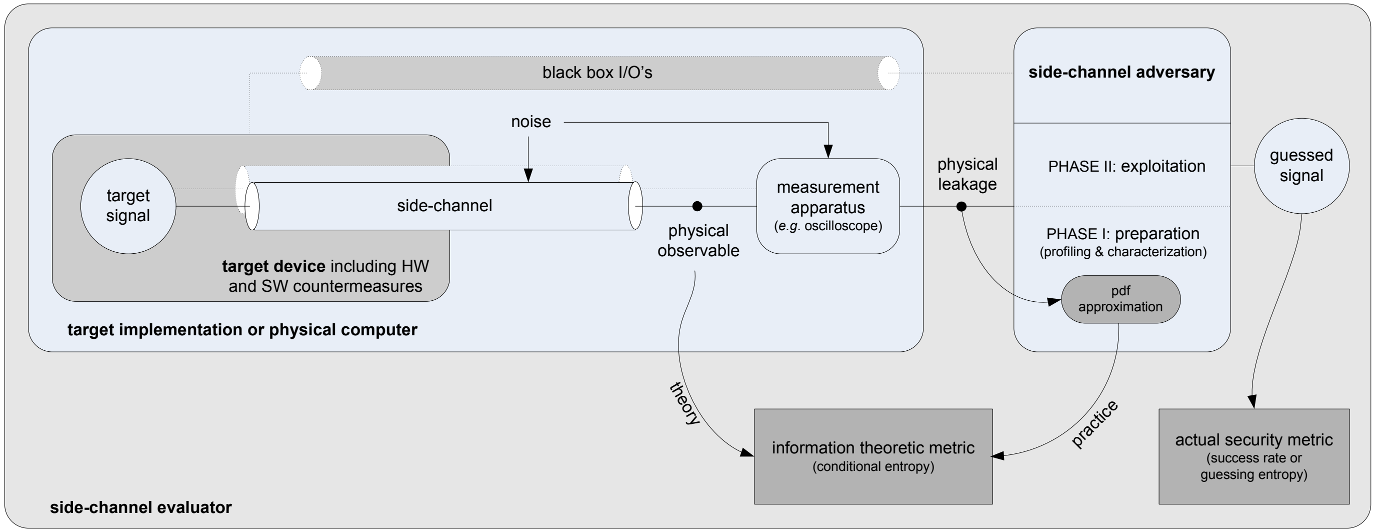

Fig. 1. Intuitive description of a side-channel key recovery attack.

In practice, side-channel attacks are usually divided in two phases. First an (optional) preparation phase provides the adversary with a training device and allows him to profile and characterize the leakages. Second, an exploitation phase is directly mounted against the target device and is aimed to succeed the key recovery. Importantly, actual adversaries do not always have the opportunity to carry out a preparation phase (in which case profiling is done on the fly). By contrast, it is an important phase for evaluators since it allows performing optimized attacks and therefore leads to a better analysis of the physical leakages. Before moving to the definitions of our metrics, we finally mention the "theory" and "practice" arrows leading to the information theoretic metric in Figure 1. 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 is the simulated attack. But such simulations always have to be followed by an *experimental attack* in order to confirm the relevance of the model. Experimental attacks exploit actual leakages obtained from a measurement setup.

4 Formal definitions

In this section, we define the metrics that we suggest for the analysis of physically observable devices. We first detail two possible security metrics, corresponding to different computational strategies. Both metrics relate to the notion of side-channel key recovery. Then, we propose an information theoretic metric driven by two requirements: (1) being independent of the adversary and (2) having the same meaning for any implementation or countermeasure. As a matter of fact and following the standard approach in information theory, Shannon's conditional entropy is a good candidate for such a metric. Typically, the use of an average criteria to compare implementations is justified by the need of adversary independence. By contrast, the interactions of an adversary with a leaking system (*e.g.* adaptive strategies) are quantified with the security metrics in our model. We note that these candidate metrics will be justified by theoretical facts in Section 6 and practical applications in Section 7. However, it is an interesting open problem to determine if other metrics are necessary to evaluate side-channel attacks (*e.g.* min-entropy is briefly discussed in Section 7.3).

4.1 Actual security metrics

Success rate of the adversary. 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 computers corresponding to the association of E_K with a leakage function L. 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 $\gamma : \mathcal{K} \to \mathcal{S}$ which maps each key k onto an equivalent key class[2] $s = \gamma(k)$, such

[2] We focus on recovering key bytes for simplicity and because they are usual targets in side-channel attacks. But any other intermediate value in an implementation could be recovered, *i.e.* in general we can choose $s = \gamma(k, x)$ with x the input of $\mathsf{E}_k(.)$.

that $|\mathcal{S}| \ll |\mathcal{K}|$. 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. Its goal is to guess a key class $s = \gamma(k)$ with non negligible probability, by exploiting its collected (black box and physical) information. 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 the attack result: the most likely candidate being g_1. A practice-oriented description of $\mathsf{A}_{\mathsf{E}_K,\mathsf{L}}$ with a more detailed specification of its features is given in Appendix A. Finally, we define a side-channel key recovery of order o with the experiment:

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

The o^{th}-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:

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

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. In practice, these quantities are generally separated for the preparation and exploitation phases (see Section 6). But 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. It measures the average number of key candidates to test after the side-channel attack. The guessing entropy was originally defined in [26] and has been proposed to quantify the effectiveness of adaptive side-channel attacks in [23]. It can 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 is reduced thanks to an attack [5]. We use it as an alternative to the success rate.

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

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

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:

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

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.

4.2 Information theoretic 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:

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

where s and s^* respectively denote the correct key class and a candidate out of the $|\mathcal{S}|$ possible ones. From (3), we derive Shannon's conditional entropy[3]:

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

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

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: $\widetilde{\mathbf{H}}^q_{s,s^*} = \mathrm{H}[S] - \mathbf{H}^q_{s,s^*}$, 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:

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

We mention that the inputs and outputs of an abstract computer are generally given to the side-channel adversary (but hidden in the formulas for clarity reasons). Therefore, it is implicitly a computational type of entropy that is proposed to evaluate the physical leakages. This is because divide-and-conquer strategies target a key class assuming that the rest of the key is unknown. But from a purely information theoretic point of view, the knowledge of a plaintext-ciphertext pair can determine a key completely (*e.g.* for block ciphers). Hence and as detailed in the next section, the amount of information extracted by a side-channel adversary depends on its computational complexity. Note also that leakage functions can be discrete or (most frequently) continuous. In the latter case, it is formally a conditional differential entropy that is computed. Note finally that in simulated attacks where an analytical model for a continuous leakage probability distribution is assumed, the previous sums over the leakages can be turned into integrals, *e.g.* we have for the conditional entropy:

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

5 Practical limitations

One important goal of the present framework is to allow a sound evaluation of any given implementation, if possible independently of an adversary's algorithmic details. For this purpose, the strategy we follow is to consider an information theoretic metric that directly depends on the leakages probability distribution $\Pr[\mathbf{L}_q|S]$. Unfortunately, there are two practical caveats in this strategy.

First, the conditional probability distribution $\Pr[\mathbf{L}_q|S]$ is generally unknown. It 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 the physical observables is through measurements. Therefore, practical attacks and evaluations have to exploit an approximated distribution $\hat{\Pr}[\mathbf{L}_q|S]$ rather than the actual one $\Pr[\mathbf{L}_q|S]$.

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 distribution for all the leakage samples is computationally intensive. Practical attacks usually approximate the probability distribution of a reduced set of samples, namely $\hat{\Pr}[\tilde{\mathbf{L}}_q|S]$.

We denote side-channel attacks that exploit the approximated probability distribution of a reduced set of leakage samples as generic template attacks. A straightforward consequence of the previous practical limitations is that for any actual device, the mutual information $\mathrm{I}(S;\mathbf{L}_q)$ can only be approximated through statistical sampling, by using generic template attacks.

We note that these generic attacks are generally assumed to be the strongest type of side-channel attacks in the literature. This informally confirms that they are convenient tools to evaluate the security limits of a leaking device. However, the term "generic template" still hides various types of techniques that can be used to approximate the leakage distribution. For example, template attacks such as in [8] use a Gaussian assumption for the leakages which frequently turns out to be very efficient in practice. But this is of course not mandatory. Eventually, if no assumptions at all can be made, it is still possible to approximate $\Pr[\mathbf{L}_q|S]$ with histograms. Also, there are different concerns in the application of template attacks such as: "how to limit the number of leakage samples for which the distribution will be estimated?" or "how to limit the number of templates to build?". The data dimensionality reduction techniques used in [4, 35] and the stochastic models in [17, 31] can be used to answer these questions in a systematic manner. But there is no general theory allowing one to decide what is the best attack for a given device. Hence, in the following we will essentially assume that one uses the "best available tool" to approximate the leakage distribution. Once this assumption is made, there remains the question of "how to use these tools to properly evaluate leaking implementations and side-channel adversaries?". This is the goal of the present framework. Quite naturally, the better generic template attacks perform in practice, the better our framework allows analyzing the causes and consequences of physical information leakages.

6 Relations between the evaluation metrics

In this section, we provide theoretical arguments that justify and connect the previous information theoretic and security metrics. These connections allow us to put forward interesting features and theoretical limitations of our model. In particular, we will consider two important questions. First, as mentioned in Section 5, generic template attacks require to estimate the leakage probability distribution. Such a leakage model is generally built during a preparation phase and then used to perform a key recovery during an exploitation phase (as pictured in Figure 1). And as mentioned in Section 4.1, these phases have to be performed within certain computational limits. Hence, to the previously defined complexity values τ, m, q of the online phase, one has to add the complexities of the preparation phase, denoted as τ_p, m_p, q_p. The first question we tackle is: given some bounds on (τ_p, m_p, q_p), can an adversary build a good estimation of the leakage distribution? We show in Section 6.1 that the conditional entropy matrix of Equation (3) is a good tool to answer this question. We also show how it relates to the asymptotic success rate of a Bayesian adversary.

Then, assuming that one can build a good approximation for the leakage distribution, we investigate the extent to which the resulting estimation of the mutual information allows comparing different implementations. Otherwise said, we analyze the dependencies between our information theoretic and security metrics. We show that there exist practically meaningful contexts of Gaussian side-channels for which strong dependencies can be put forward. But we also emphasize that no general statements can be made for arbitrary distributions. As a consequence, Section 6.2 essentially states that the mutual information is a good metric to compare different implementations, but it always has to be completed with a security analysis (*i.e.* success rate and/or guessing entropy).

Note that these two questions relate to different concerns. The first one deals with computational aspects in statistical inference problems. It only makes sense for experimental attacks and bounded adversaries. The second one deals with ideal relations between the metrics that are even meaningful in simulations.

6.1 Asymptotic meaning of the conditional entropy: "Can I approximate the leakage probability distribution?"

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: $\mathbf{Succ}^{\text{sc-kr-}o,S}_{\mathsf{A}_{\mathsf{E}_K,\mathsf{L}}}(q \rightarrow \infty)$.

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. An approximated leakage distribution $\hat{\Pr}[\tilde{\mathbf{L}}_q|S]$ 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.

In this section, we assume that one has built an approximated leakage distribution $\hat{\Pr}[\tilde{\mathbf{L}}_q|S]$ with some (bounded) measurement queries q_p, memory m_p and time τ_p. We want to evaluate if this approximation is good. For theoretical purposes, we consider an adversary/evaluator who can perform unbounded queries to the target device during the exploitation phase. We use these queries to evaluate the entropy matrix $\widehat{\mathbf{H}}^q_{s,s^*}$ defined in Section 4.2. It directly leads to the following relation with the asymptotic success rate of a Bayesian adversary.

Theorem 1. *Assuming independent leakages for the different queries in a side-channel attack, an approximated leakage probability distribution* $\hat{\Pr}[\tilde{\boldsymbol{L}}_q|S]$ *is sound if and only if the conditional entropy matrix evaluated in an unbounded exploitation phase is such that* $argmin_{s^*}\, \widehat{\boldsymbol{H}}^q_{s,s^*} = s$, $\forall s \in \mathcal{S}$.

Proof. let us consider a target key class s and a leakage matrix $\mathbf{l}_{p,q}$ that contains p realizations of a q-queries leakage vector $\mathbf{L}_q$. A Bayesian adversary having access to these leakages is successful if and only if:

$$s = \underset{s^*}{argmax}\ \hat{\Pr}[s^*|\tilde{\mathbf{l}_{p,q}}]$$

$$s = \underset{s^*}{argmax}\ \frac{\hat{\Pr}[\tilde{\mathbf{l}_{p,q}}|s^*] \cdot \Pr[s^*]}{\hat{\Pr}[\tilde{\mathbf{l}_{p,q}}]}$$

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

$$s = \underset{s^*}{argmax}\ \hat{\Pr}[\tilde{\mathbf{l}_{p,q}}|s^*]$$

Since we assume independent leakages for different queries, we also have:

$$s = \underset{s^*}{argmax}\ \prod_{i=1}^{p} \hat{\Pr}[\tilde{\mathbf{l}_{i,q}}|s^*]$$

Any $\tilde{\mathbf{l}_{i,q}}$ has exactly the same size as a q-element leakage vectors $\tilde{\mathbf{l}_q}$ for which the distribution has been approximated during the preparation phase. Additionally, since we consider an asymptotic attack (*i.e.* an unbounded exploitation phase), p is not bounded and each q queries to the target physical computer determine a leakage trace $\tilde{\mathbf{l}_{i,q}}$ picked up from the real leakage distribution $\Pr[\mathbf{L}_q|s]$. Therefore, an asymptotic attack is successful if and only if:

$$s = \underset{s^*}{argmax}\ \prod_{i} \hat{\Pr}[\tilde{\mathbf{l}_{i,q}}|s^*]^{\Pr[\mathbf{l}_{i,q}|s]}$$

$$s = \underset{s^*}{argmax}\ \prod_{i} \hat{\Pr}[s^*|\tilde{\mathbf{l}_{i,q}}]^{\Pr[\mathbf{l}_{i,q}|s]}$$

$$s = \underset{s^*}{argmax}\ \sum_{i}\ \Pr[\mathbf{l}_{i,q}|s] \cdot \log_2 \hat{\Pr}[s^*|\tilde{\mathbf{l}_{i,q}}] \tag{5}$$

Finally, we just observe that the sum in Equation (5) is equivalent to Equation (3) but for their sign and the approximated probability in the logarithmic factor. That is, it exactly corresponds to how the conditional entropy matrix is estimated in practice. Therefore, if the previous condition holds for all classes s, the Bayesian side-channel attack is asymptotically successful. □

There are several 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. Theorem 1 only makes sense for bounded preparation phases. For unbounded preparations, an adversary would eventually access the exact distribution $\Pr[\mathbf{L}_q|S]$. In this context, the soundness does only depend on the cardinality of the different sets $\{s^*|\Pr[\mathbf{L}_q|s^*] = \Pr[\mathbf{L}_q|s]\}$, $\forall s \in \mathcal{S}$.
3. The condition of independence for consecutive leakages is not expected to be fully verified in practice. For example, there could exist history effects in the side-channel observations. However, it is expected to hold to a sufficient degree for our proof to remain meaningful in most applications.
4. In practice, the exploitation phase in a side-channel attack is bounded as the preparation. Therefore, Theorem 1 will be relevant as long as the number of leakages used to test the approximated leakage distribution and estimate the conditional entropy matrix is sufficient (*i.e.* for a large enough p).
5. Finally, the condition on the entropy matrix $\widehat{\mathbf{H}}^q_{s,s^*}$ is stated for the number of queries q for which the leakage distribution $\Pr[\mathbf{L}_q|S]$ was approximated during the preparation phase. In general, finding a sound approximation for q implies that it should also be feasible to find sound approximations for any $q' > q$. But in practice, computational limitations can make it easier to build a sound approximation for small q values than for larger ones.

We mention that a sound leakage probability distribution could be equivalently defined as giving rise to an asymptotic guessing entropy of one.

6.2 Comparative meaning of the conditional entropy: "Does more entropy imply more security?"

Let us write an exemplary conditional entropy matrix and its estimation as:

$$\mathbf{H}^q_{s,s^*} = \begin{pmatrix} h_{1,1} & h_{1,2} & \dots & h_{1,|\mathcal{S}|} \\ h_{2,1} & h_{2,2} & \dots & h_{2,|\mathcal{S}|} \\ \dots & \dots & \dots & \dots \\ h_{|\mathcal{S}|,1} & h_{|\mathcal{S}|,2} & \dots & h_{|\mathcal{S}|,|\mathcal{S}|} \end{pmatrix} \quad \widehat{\mathbf{H}}^q_{s,s^*} = \begin{pmatrix} \hat{h}_{1,1} & \hat{h}_{1,2} & \dots & \hat{h}_{1,|\mathcal{S}|} \\ \hat{h}_{2,2} & \hat{h}_{2,2} & \dots & \hat{h}_{2,|\mathcal{S}|} \\ \dots & \dots & \dots & \dots \\ \hat{h}_{|\mathcal{S}|,1} & \hat{h}_{|\mathcal{S}|,2} & \dots & \hat{h}_{|\mathcal{S}|,|\mathcal{S}|} \end{pmatrix}$$

Theorem 1 states that if the diagonal values of a (properly approximated) 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 approximated leakage probability distribution. Namely, Theorem 1 answers the question: "*Can one approximate the leakage probability distribution under some computational bounds τ_p,m_p,q_p?*".

Let us now assume that the answer is positive and denote each element $h_{s,s}$ as the residual 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?*". As a matter of fact and contrary to the previous section, this question makes sense both for the ideal conditional entropy matrix that would correspond to an exact leakage distribution and for its approximation. Since general conclusions for arbitrary leakage distributions are not possible to obtain, our following strategy is 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[4] 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 residual entropy of the different key classes s, we also need a more specific definition of the success rate against a given key class s:

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

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:

$$
\mathbf{Succ}^{\text{sc-kr-}o,s}_{\mathsf{A}_{\mathsf{E}_k,\mathsf{L}}}(\tau, m, q) = \Pr\ [\mathbf{Exp}^{\text{sc-kr-}o,s}_{\mathsf{A}_{\mathsf{E}_K,\mathsf{L}}} = 1] \tag{6}
$$

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. Finally, the lower left and right pictures represent the univariate and bivariate leakage distributions of an 8-target side-channel attack. Note that in general, the internal state of an implementation does not only depend on the keys but also on other inputs, *e.g.* the plaintexts in block ciphers. Hence, the different dimensions in a multivariate distribution can represent both the different samples of a single leakage trace (generated with a single plaintext) or different traces (*e.g.* each dimension could correspond to a different plaintext). Eventually, it is an adversary's choice to select the internal states for which templates will

[4] Experimentally observed in a number of works, *e.g.* [25], Section 4.2.

be built. Therefore, we do not claim that these distributions always connect to practical attacks. But as will be seen in the following, even these simple theoretical contexts hardly allow simple connections between information and security.

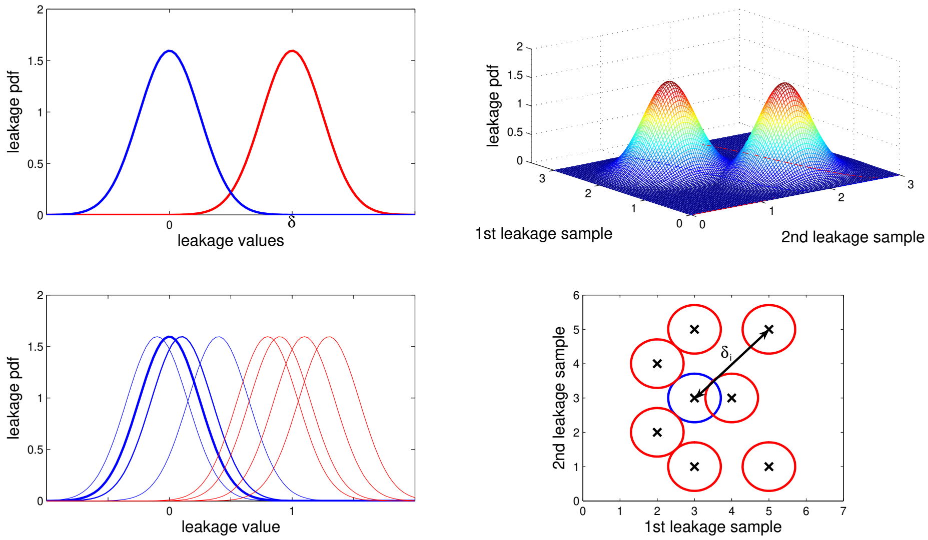

Fig. 2. Illustrative leakage probability distributions $\Pr[\mathbf{L}_q|S]$.

We now discuss formally the connections between the success rate against a key class s and its residual entropy for idealized distributions and attacks.

Definition 7. An ideal side-channel attack is a Bayesian attack in which the leakages are exactly predicted by the adversary's approximated probability density function $\hat{\Pr}[\tilde{\mathbf{L}}_q|S]$ (*e.g.* thanks to an unbounded preparation phase).

Lemma 1. *In an ideal 2-target side-channel attack exploiting a univariate Gaussian leakage distribution, the residual entropy of a key class s is a monotonously decreasing function of the single query (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 4, the single query success rate and the residual entropy of the key class s equal:

$$\mathbf{Succ}^{\text{sc-kr-1},s}_{\mathsf{A}_{\mathsf{E}_k},\mathsf{L}}(\delta,\sigma) = \int_{-\infty}^{\delta/2} \mathsf{N}_x(0,\sigma)\ dx$$

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

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

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

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

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

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

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

Then, we just observe that $\mathbf{Succ}^{\text{sc-kr-1},s}_{\mathsf{A}_{\mathsf{E}_k},\mathsf{L}}$ and $h_{s,s}$ are respectively monotonously increasing and decreasing functions of z, which completes the proof. □

Lemma 2. *In an ideal 2-target side-channel attack exploiting a multivariate Gaussian leakage distribution, with independent leakage samples having the same noise standard deviation, the residual entropy of a key class s is a monotonously decreasing function of the single query (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 residual 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. □

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. One can observe in the lower right part of Figure 2 that 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 8. 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 residual entropy of the key class s is maximum.

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

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

Proof sketch. In 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 residual entropy of s. Therefore, the residual entropy and the success rate against s only depend on the ratio between this Euclidean distance and the noise standard deviation which implies that Theorem 2 is a straightforward consequence of Lemma 2. □

Theorem 2 constitutes our main positive result for the use of the conditional entropy as a comparison metric for different implementations. Its relevance to practice depends on the extent to which actual leakage distributions correspond to the idealized Gaussian curves that we considered. As for Theorem 1, a corollary could be demonstrated with the guessing entropy. Unfortunately, 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 Gaussian leakage distributions, the constant residual 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 residual 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, D_y such that the residual entropy for a class s corresponding to D_x is higher than the residual entropy for a class s' corresponding to D_y and the success rate of every order for a Bayesian adversary exploiting D_x is higher than the success rate for the same adversary exploiting D_y.* This is 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 B.

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

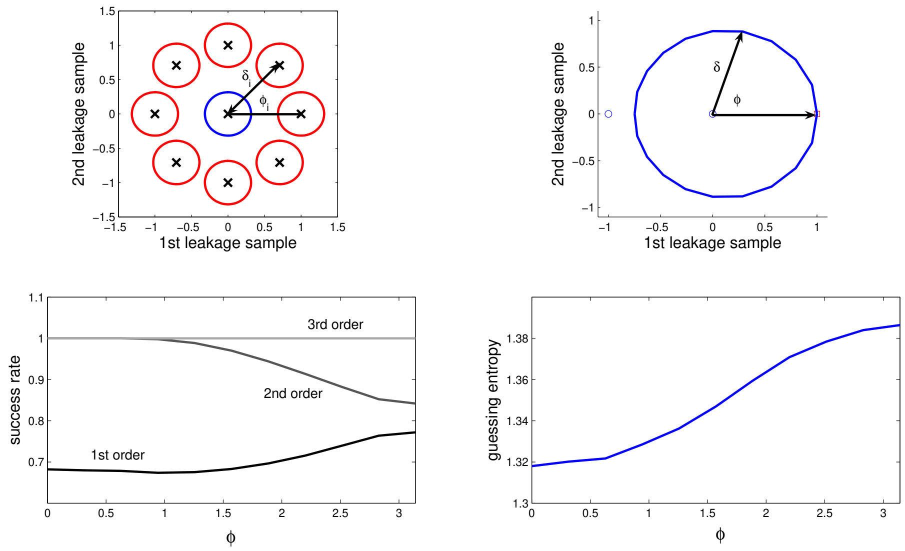

Fig. 3. Perfect leakage distribution and leakage distributions having constant conditional entropy with their associated success rates and guessing entropy .

6.3 Intuition of the metrics

In this section, we recall and 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 the consequences of the previous limitations.

Intuitions related to Theorem 1.

1.1 *Theorem 1 tells if it is possible to approximate a given leakage function in a bounded preparation phase.* As mentioned in Section 5, such an approximation highly depends on the actual tools that are used for this purpose. In general, the better the tools, the better the evaluation. Hence, Theorem 1 allows checking if these tools are powerful enough. If they are not...

1.2 *Theorem 1 indicates some resistance of the target implementation against side-channel attacks.* That is, 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 does not reach one. But this does not imply security against side-channel attacks (*e.g.* think about a caricatural device from which only one key could not be recovered). In this context, it is important to evaluate the actual security metrics for different adversaries in order to check if high success rates (possibly of high orders) can still be reached. The position of the correct key class in the entropy matrix is also informative with this respect. It can be used as an efficient tool to check the similarities between different distributions.

Intuitions related to Theorem 2.

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

2.2 *Theorem 2 confirms that the mutual information is a relevant tool to compare different implementations.* It shows meaningful contexts of Gaussian channels for which less residual entropy for a key class implies a more efficient Bayesian key recovery. This strengthens the intuitive requirements of Section 4, namely the need of an adversary independent metric having the same meaning for any implementation. However, Section 5 and Facts 1, 2 also show that the comparisons based on the conditional entropy only can be misleading, both for theoretical and practical reasons. Hence...

2.3 *The conditional entropy is not a stand-alone metric to compare implementations and always has to be combined with a security analysis.* Additionally to the previously described limitations, this need relates to the methodological separation of implementations and adversaries in our model. For a given amount of information leaked by an implementation, different side-channel distinguishers could be considered (see Appendix A). For example, template attacks that closely relate to the definition of mutual information are not the most practical in terms of adversarial context. Suboptimal distinguishers are frequently used in practice. Therefore, security metrics are useful to evaluate the number of queries for a given attack to succeed.

General remarks.

1. The limitations of the information theoretic metric should not be seen as weaknesses in the model but as related to the inherent complexity of side-channel attacks. Similarly, the existence of theoretical contexts for which the conditional entropy is not perfectly meaningful to compare implementations does not prevent it to be relevant in numerous practical contexts.
2. The mutual information, success rates and guessing entropy are average evaluation criteria. However in practice, the information leakages and security of an implementation could be different for different keys. Therefore, it is important to also consider these notions for the different keys separately (*e.g.* to evaluate the conditional entropy matrix rather than the mutual information). This last remark motivates the following practice-oriented definition.

Definition 9. We say that a side-channel attack against a key class variable S is a weak template attack if all the key classes s have the same residual entropy $h_{s,s}$ and each line of the entropy matrix $\mathbf{H}^q_{s,s^*}$ is a permutation of another line of the matrix. We say that a side-channel attack is a strong template attack if at least one of the previous conditions does not hold.

Intuitively, a weak template attack can be straightforwardly analyzed with the conditional entropy. The evaluation of a strong template attack requires to consider every key class independently. The terms weak and strong relate to the ability of the adversary to characterize key-specific features in his templates.

7 Applications of the model

In this section, we aim to provide paper-and-pencil examples of side-channel attacks that confirm the previous intuitions and can be easily reproduced by the reader. That is, we use idealized leakage functions in order to illustrate interesting features of our model. Applications to more complex and practically meaningful contexts can be found in other publications [24, 34, 36, 35].

For this purpose, we consider a known plaintext attack against a reduced block cipher that we formalize as follows. Let S be a 4-bit substitution box, *e.g.* the one of the AES candidate Serpent. We target the computation of $y = \mathsf{S}(x \oplus k)$, where x is a random plaintext and k a secret key. A Bayesian adversary is provided with observations $(x, \mathsf{L}'(y) + r)$, where r is a gaussian noise with mean 0 and standard deviation σ. For any y value, the deterministic part of the leakage $\mathsf{L}'(y)$ is given by a vector $\mathbf{Z}$. The adversary's goal is to recover the key k.

7.1 Application of Theorem 1

Let us first consider the quite classical Hamming weight leakages where $\mathsf{L}'(y) = \mathrm{H_W}(y)$. Otherwise said, let us consider $\mathbf{Z} = [0, 1, 1, 2, 1, 2, 2, 3, 1, 2, 2, 3, 2, 3, 3, 4]$. We will compute three entropy matrices. In the first case (matrix A), we assume noise-free leakages. Hence the entropy matrix is straightforward to compute and the residual entropies all equal $h_{s,s} = -\sum_{h=0}^{4} \binom{4}{h} / 2^4 \cdot \log_2 1 / \binom{4}{h}$, the other matrix elements being meaningless (nan). In the second case (matrix B), we have $\sigma = 1$ and we perform a bounded preparation using 1000 measurements to approximate each element of $\mathbf{Z}$. As a result, we obtain an approximated vector: $\widehat{\mathbf{Z}} = [0.02, 0.95, 1.03, 2.00, 1.01, 2.02, 2.02, 2.97, 1.01, 2.0, 1.99, 3.0, 1.99, 3.04, 3.0, 4.01]$. We then evaluate the entropy matrix in an unbounded (*i.e.* in practice, sufficiently sampled) exploitation phase. As a result, we observe that the diagonal elements are still minimum in the matrix. This reveals that the approximation of the leakage function is sound, *i.e.* the bounded preparation was successful. But compared to the first example, the conditional entropy is increased. This is caused by two reasons: (1) the approximated leakage model does not exactly correspond to the actual leakage function and (2) there is noise in the leakages. Note that the previous matrices typically correspond to a context of weak template attacks since all the keys can be identified thanks to the same 16 plaintexts.

$$\widehat{\mathbf{H}}^{A,1}_{k,k^*} = \begin{pmatrix} 1.97 & \text{nan} & \ldots & \text{nan} \\ \text{nan} & 1.97 & \ldots & \text{nan} \\ \ldots & \ldots & \ldots & \ldots \\ \text{nan} & \text{nan} & \ldots & 1.97 \end{pmatrix} \quad \widehat{\mathbf{H}}^{B,1}_{k,k^*} = \begin{pmatrix} 3.50 & 5.31 & \ldots & 4.58 \\ 5.11 & 3.50 & \ldots & 4.95 \\ \ldots & \ldots & \ldots & \ldots \\ 5.66 & 4.40 & \ldots & 3.50 \end{pmatrix} \quad \widehat{\mathbf{H}}^{C,1}_{k,k^*} = \begin{pmatrix} 5.19 & 4.55 & \ldots & 5.90 \\ 5.54 & 5.19 & \ldots & 4.46 \\ \ldots & \ldots & \ldots & \ldots \\ 5.09 & 4.64 & \ldots & 5.19 \end{pmatrix}$$

In a third case (matrix C), we assume the same noise variance and an insufficient preparation that leads to a bad leakage model approximated with the vector $\widehat{\mathbf{Z}} = [3, 1, 0, 2, 1, 3, 3, 2, 1, 2, 3, 0, 0, 1, 1, 2]$. As a result, the matrix diagonal elements are not minimum anymore which involves that using this model will not allow a successful key recovery. These examples confirm the intuitions in the previous section. Namely, the important question in side-channel attacks is not: "is there information available in the physical leakages?" but rather "can I exploit it?". The conditional entropy matrix is a convenient tool to answer this question.

7.2 Application of Theorem 2

Let us now consider different leakage functions and assume an unbounded preparation phase (*i.e.* the adversary can exploit the exact leakage distribution) in order to evaluate the extent to which more entropy leads to less success rate in practical contexts. As in the previous section, we start the Hamming weight leakages and $\mathbf{Z}_1 = [0, 1, 1, 2, 1, 2, 2, 3, 1, 2, 2, 3, 2, 3, 3, 4]$. We also evaluate two other leakage functions represented by the vectors: $\mathbf{Z}_2 = [0, 0, 0, 0, 1, 1, 1, 1, 2, 2, 2, 2, 3, 3, 3, 3]$ and $\mathbf{Z}_3 = [0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 2, 2, 3, 3, 4, 4]$. The conditional entropies and single-query success rates with $\sigma = 0$ can be straightforwardly computed as:

$$\mathrm{H}[K|\mathbf{L}_1^{\mathbf{Z}_1}] \simeq 1.97 \qquad \mathrm{H}[K|\mathbf{L}_1^{\mathbf{Z}_2}] = 2 \qquad \mathrm{H}[K|\mathbf{L}_1^{\mathbf{Z}_3}] \simeq 2.16$$
$$\mathbf{Succ}^{\text{sc-kr}}_{\mathbf{L}_1^{\mathbf{Z}_1}}(q=1) = \frac{5}{16} \qquad \mathbf{Succ}^{\text{sc-kr}}_{\mathbf{L}_1^{\mathbf{Z}_2}}(q=1) = \frac{1}{4} \qquad \mathbf{Succ}^{\text{sc-kr}}_{\mathbf{L}_1^{\mathbf{Z}_3}}(q=1) = \frac{5}{16}$$

At first sight, it seems that these leakage functions exactly contradict Theorem 2. For example, when moving from $\mathbf{Z}_2$ to $\mathbf{Z}_3$, we see that both the conditional entropy and the success rate are increased. However, the goal of side-channel attacks is generally to reach high success rates that are not obtained with a single query. Hence, it is also interesting to investigate the success rate for more queries. In the left part of Figure 4, these success rates for increasing q values are plotted. It clearly illustrates that while $\mathbf{Z}_2$ leads to a lower success rate than $\mathbf{Z}_3$ for $q = 1$, the opposite conclusion holds when increasing q. That is, the intuition given by Theorem 2 only reveals itself for $q > 2$. Importantly, these conclusions can vary when noise is inserted in the leakages, *e.g.* assuming $\sigma = 1$, we have:

$$\mathrm{H}[K|\mathbf{L}_1^{\mathbf{Z}_1}] \simeq 3.50 \qquad \mathrm{H}[K|\mathbf{L}_1^{\mathbf{Z}_2}] \simeq 3.42 \qquad \mathrm{H}[K|\mathbf{L}_1^{\mathbf{Z}_3}] \simeq 3.22$$

The right part of Figure 4 plots the success rates of these noisy leakage functions. It again highlights a context in which Theorem 2 is eventually respected. In general, these examples underline another important feature of our metrics. Namely, the more challenging the side-channel attack (*i.e.* the more queries needed to reach high success rates), the more significant the conditional entropy is. Otherwise said: the mutual information better reveals its intuition asymptotically. And in such contexts, the single-query success rate can be misleading.

Note that the examples in this section are more directly reflective of actual side-channel attacks in which different plaintexts can generally be used to identify a key class than the ideal contexts investigated in Section 6.2.

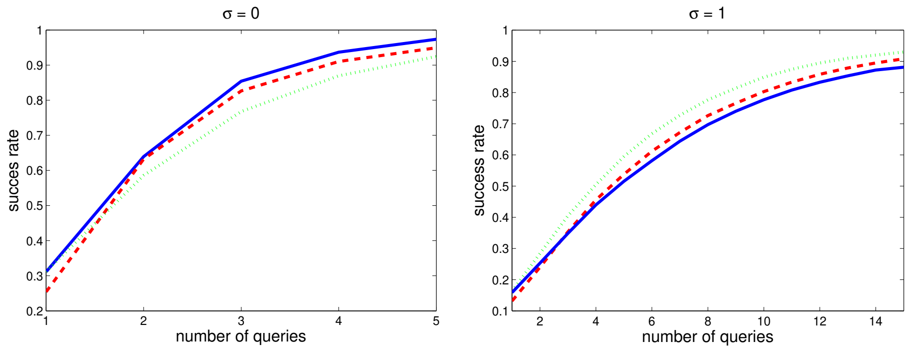

Fig. 4. 1^{st}-order success rates in function of the number of queries for the leakages functions corresponding to $\mathbf{Z}_1$ (solid line), $\mathbf{Z}_2$ (dashed line) and $\mathbf{Z}_3$ (dotted line).

7.3 Summary of the important model features

Before to conclude, we briefly recall some important aspects of our model.

1. The evaluation of implementations and adversaries are different issues and this has to be taken into account in the formalization of side-channel attacks.
2. As a consequence, two different types of metrics are required in these evaluations, that we denote as information theoretic and security metrics.
3. Shannon's conditional entropy, success rates of different orders and the guessing entropy are interesting candidates for such metrics.
4. Information is not equivalent to security: under certain conditions, the conditional entropy can be used to compare implementations but in general, bits of entropy cannot be translated into bits of security.
5. Intuitively, the model can be used as a middle-ware between the evaluation of physical information leakages and their exploitation. It lets the best combination of the two issues as an open algorithm-dependent question. For example, the soundness of an approximated leakage probability distribution is always tested for a given key class. But this still requires to find the best key classes to target if one wants to develop optimized attacks.

With respect to the relevance of other metrics in the model, we finally mention that min-entropy is equivalent to the (first-order) success rate of an adversary with an optimal strategy (for multiple queries, it implies to consider adaptive attacks). As illustrated in this section, we believe that Shannon's conditional entropy nicely captures the information leakages in most practical applications, even with small number of queries. For example, Figure 4 is typical of contexts where the success rate for $q = 1$ is not very significant while the conditional entropy nicely quantifies the evolution of this success rate for any larger number of queries. But as already said, the information theoretic analysis always has to be completed with a security analysis. So, as long as an optimal strategy can be employed by the adversary, the min-entropy will be estimated too.

8 Evaluation methodology

Following 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:

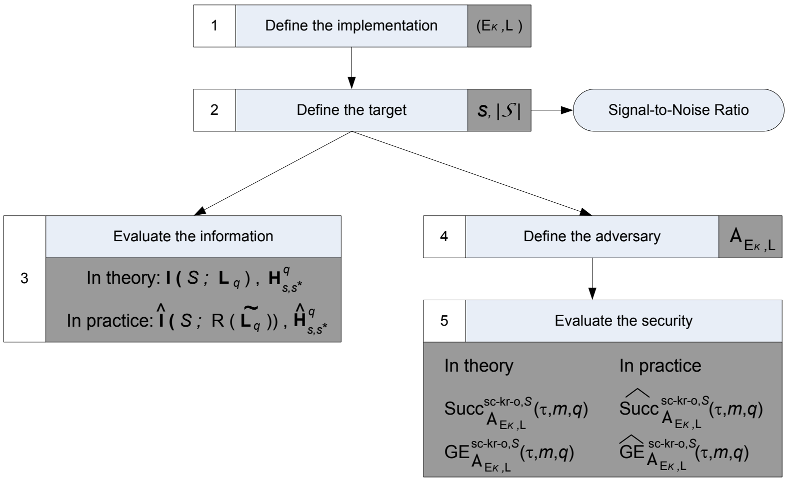

Fig. 5. Evaluation methodology for side-channel attacks.

1. We define the target implementation. 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.
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 exploitable information contained in the physical observations obtained from a leaking device?*". For this purpose, we use the mutual information or entropy matrix. As previously mentioned, 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 simulator in place of the actual leakage function.
4. We define the adversary that is used to exploit the side-channel leakages. It typically implies to specify all the steps of the attack described in Appendix A that includes, *e.g.* leakage modeling, statistical test, ...
5. We finally answer the second question in our evaluation, namely: "*How successfully can an adversary turn his physical 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 4.1.

Figure 5 again indicates that the information theoretic metric can be used to measure an implementation while the actual security metrics are rather useful to evaluate adversaries. Additionally to these metrics, 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.

We note finally that the definition of an implementation requires to evaluate the cost of the equipment used to monitor the leakages. Since quantifying such costs is typically the tasks assigned the standardization bodies, we refer to the common criteria [10] and FIPS 140-2 documents [14] (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.

9 Conclusions and open problems

A framework for analyzing cryptographic implementations is introduced in order to unify the theory and practice of side-channel attacks. It is aimed to bridge the formal understanding of physically observable cryptography to the exploitation of actual leakages in experimental key recoveries. The framework is centered around a theoretical model in which the effect of practically relevant leakage functions is evaluated with a combination of security and information theoretic metrics. It allows a fair discussion of the tradeoffs in physically observable cryptography, *i.e.* "*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 depends 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 relates to the computational strategy considered. As a matter of fact, for comparable amounts of side-channel queries, 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 computation (*i.e.* exploiting the black-box access and classical cryptanalysis techniques), it can be turned into a successful key recovery. Otherwise said, a lack of physical information can be overcome by a more computationally intensive adversarial strategy.

As an interface between an engineering problem (*i.e.* how much is leaked?) and a cryptographic problem (*i.e.* how to exploit it?), our framework consequently helps putting forward properly quantified weaknesses in physically observable devices. The fair evaluations provided by our analysis can then be used in two directions. Either the physical weaknesses can be fed-back to hardware designers to reduce physical leakages. Or they can be sent to cryptographic designers in order to conceive schemes that can cope with physical leakages.

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 investigating the best heuristics to deal with high dimensional leakage data (our model assumes adversaries exploiting such specialized algorithms). 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 [28] (but again in a more specialized fashion). A third possible direction relates to the construction of implementations with provable (or arguable) security against side-channel attacks, *e.g.* as proposed in [12, 30, 29]. Finally, this work could be extended to other physical threats (*e.g.* fault attacks) and combined with other approaches for modeling physical attacks such as [16, 20, 21].

Acknowledgements. The authors would like to thank the different colleagues and reviewers of which the comments improved the presentation of this work.

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. M. Backes, B. Köpf, *Formally Bounding the Side-Channel Leakage in Unknown-Message Attacks*, IACR ePrint archive, 2008, http://eprint.iacr.org/2008/162
4. 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.
5. 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.
6. 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.
7. C. Cachin, *Entropy Measures and Unconditional Security in Cryptography*, PhD Thesis, ETH Dissertation, num 12187, 1997.

8. S. Chari, J. Rao, P. Rohatgi, *Template Attacks*, CHES 2002, Lecture Notes in Computer Science, vol 2523, pp 13-28, CA, USA, August 2002.
9. Cryptographic Hardware and Embedded Systems, *http://www.chesworkshop.org*
10. *Application of Attack Potential to Smart Cards*, Common Criteria Supporting Document, Version 1.1, July 2002, http://www.commoncriteriaportal.org
11. T.M. Cover, J.A. Thomas, *Information Theory*, Wiley and Sons, New York, 1991.
12. S. Dziembowski, K. Pietrzak, *Leakage-Resilient Cryptography*, in the proceedings of FOCS 2008, pp 293-302, Philadelphia, USA, October 2008.
13. ECRYPT Network of Excellence in Cryptology, *The Side-Channel Cryptanalysis Lounge* , http://www.crypto.ruhr-uni-bochum.de/en_sclounge.html.
14. FIPS 140-2, *Security Requirements for Cryptographic Modules*, Federal Information Processing Standard, NIST, U.S. Dept. of Commerce, December 3, 2002.
15. K. Gandolfi, C. Mourtel, F. Olivier, *Electromagnetic Analysis: Concrete Results*, in the proceedings of CHES 2001, Lecture Notes in Computer Science, vol 2162, pp 251-261, Paris, France, May 2001.
16. 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.
17. B. Gierlichs, K. Lemke, C. Paar, *Templates vs. Stochastic Methods*, CHES 2006, LNCS, vol 4249, pp 15-29, Yokohama, Japan, October 2006.
18. B. Gierlichs, L. Batina, P. Tuyls, B. Preneel, *Mutual Information Analysis: A Generic Side-Channel Distinguisher*, in the proceedings of CHES 2008, LNCS, vol 5154, pp 396-410, Washington DC, USA, August 2008.
19. L. Goubin, J. Patarin, *DES and Differential Power Analysis*, CHES 1999, LNCS, vol 1717, pp 158-172, Worcester, MA, USA, August 1999.
20. Y. Ishai, A. Sahai, D. Wagner, *Private Circuits: Securing Hardware against Probing Attacks*, in the proceedings of Crypto 2003, Lecture Notes in Computer Science, vol 2729, pp 463-481, Santa Barbara, CA, USA, August 2003.
21. Y. Ishai, M. Prabhakaran, A. Sahai, D. Wagner, *Private Circuits II: Keeping Secrets in Tamperable Circuits*, Eurocrypt 2006, Lecture Notes in Computer Science, vol 4004, pp 308-327, St. Petersburg, Russia, May 2006.
22. P. Kocher, J. Jaffe, B. Jun, *Differential Power Analysis*, Crypto 1999, LNCS, vol 1666, pp 398-412, Santa-Barbara, CA, USA, August 1999.
23. B. Köpf, D. Basin, *an Information Theoretic Model for Adaptive Side-Channel Attacks*, CCS 2007, Alexandria, VA, USA, October 2007.
24. F. Macé, F.-X. Standaert, J.-J. Quisquater, *Information Theoretic Evaluation of Side-Channel Resistant Logic Styles*, CHES 2007, Lecture Notes in Computer Science, vol 4727, pp 427-442, Vienna, Austria, September 2007.
25. S. Mangard, E. Oswald, T. Popp, *Power Analysis Attacks*, Springer, 2007.
26. J.L. Massey, *Guessing and Entropy*, IEEE International Symposium on Information Theory, pp 204, Trondheim, Norway, June 1994.
27. 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.
28. S. Micali, L. Reyzin, *Physically Observable Cryptography*, TCC 2004, LNCS, vol 2951, pp 278-296, Cambridge, MA, USA, February 2004.
29. C. Petit, F.-X. Standaert, O. Pereira, T.G. Malkin, M. Yung, *A Block Cipher based PRNG Secure Against Side-Channel Key Recovery*, In the proceedings of ASIACCS 2008, pp 56-65, Tokyo, Japan, March 2008.
30. K. Pietrzak, *A Leakage-Resilient Mode of Operation*, in the proceedings of Eurocrypt 2009, Lecture Notes in Computer Science, Cologne, Germany, April 2009.

31. 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.
32. C.E. Shannon, *A Mathematical Theory of Communication*, Bell System Technical Journal, vol 27, pp 379-423 and 623-656, July and October, 1948.
33. C.E. Shannon, *Communication theory of secrecy systems*, Bell System Technical Journal, vol 28, pp 656-715, October 1949.
34. F.-X. Standaert, E. Peeters, C. Archambeau, J.-J. Quisquater, *Towards Security Limits in Side-Channel Attacks*, in the proceedings of CHES 2006, LNCS, vol 4249, pp. 30–45, Yokohama, Japan, October 2006.
35. F.-X. Standaert, C. Archambeau, *Using Subspace-Based Template Attacks to Compare and Combine Power and Electromagnetic Information Leakages*, in the proceedings of CHES 2008, LNCS, vol 5154, Washington DC, USA, August 2008.
36. F.-X. Standaert, B. Gierlichs, I. Verbauwhede, *Partition vs. Comparison Side-Channel Distingsuishers: An Empirical Evaluation of Statistical Tests for Univariate Side-Channel Attacks*, in the proceedings of ICISC 2008, LNCS, vol 5461, pp 253-267, Seoul, Korea, December 2008.
37. 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 Practice-oriented definitions

From the definition of Section 4.1, 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 6. As mentioned in Section 3, it consists of 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 and mention that some of the described steps are facultative.

1. Input selection. The adversary selects its (possibly adaptive) q queries $\mathbf{x}_q$ (defined in Section 4.2) 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 $\mathbf{v}^q_{s^*} = \mathsf{V}(s^*, \mathbf{x}_q)$ containing N_v-element predictions $v^i_{s^*}$, $i \in [1, q]$, 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 [8]. 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: $\mathsf{M}(s^*, \mathbf{v}^q_{s^*})$, as in correlation attacks [6]. 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 $\mathsf{P}(s^*, \mathbf{v}^q_{s^*})$. 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. 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 4.1. Typical statistical tools include the difference of mean test [27], the Pearson correlation coefficient, the mutual information analysis [18] or the Bayesian classifier that is central in this work.
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.

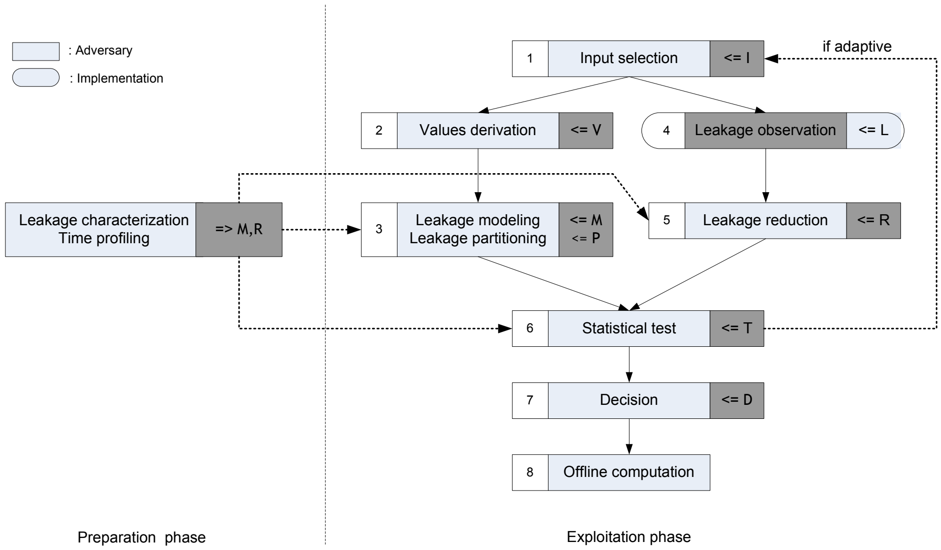

Fig. 6. Practical side-channel adversary.

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. 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 both phases. Importantly, the definition of the preparation and exploitation phases also has to include the description of the adversary's capabilities, as in the classical black box setting, *e.g.* non adaptive/adaptive, unknown/known/chosen, plaintext/ciphertext, ..., with the additional possibility to have known or chosen keys during the preparation phase. An exemplary application of the previous definitions to a set of standard side-channel attacks can be found in [36].

B More entropy sometimes means more success rate

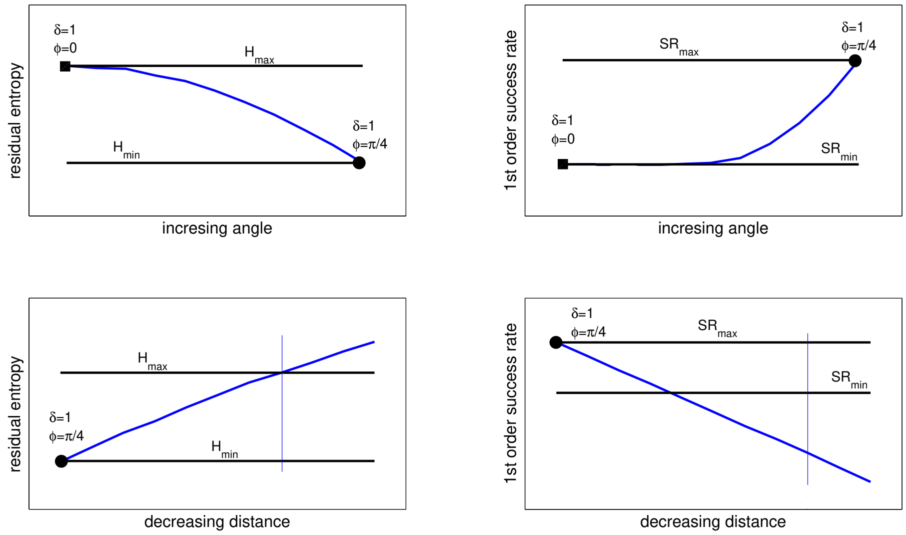

Fig. 7. Finding leakage distributions with more entropy and more success rate.

Figure 7 illustrates the search for leakage distributions having both less residual entropy and less success rate of every order for a 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 7, it implies both a reduction of the residual entropy and an increase of the (first order) success rate against the central key class. We store the maximum and minimum value for the residual 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 residual entropy and a reduction of the (first order) success rate against the central key class. We store the distance were the residual 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 smaller than its initial value. So, there are points were reducing the residual entropy also reduces the (first order) success rate. Since changing the angle straightforwardly implies a reduction of the second order success rate, we have a reduction of the success rate for every meaningful order (the third order success rate is stuck at one).

C 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 4.1
D	Decision function	pp 30, app B
$\mathbf{E}(X)$	The expected value of a random variable X	pp 9, sec 4.1
E_K	Family of cryptographic abstract computers indexed by a variable key K	pp 7, sec 4.1
γ	Key classification function	pp 7, sec 4.1
$\mathbf{GE}^{\text{sc-kr-}S}_{\mathsf{A}_{\mathsf{E}_K,\mathsf{L}}}$	Guessing entropy of a side-channel key recovery adversary against a key class variable S	pp 9, sec 4.1
$\mathbf{H}^q_{s,s^*}$	Conditional entropy matrix	pp 9, sec 4.2
$\widetilde{\mathbf{H}}^q_{s,s^*}$	Entropy reduction matrix	pp 10, sec 4.2
$h_{s,s}$	Residual entropy of a key class s	pp 14, sec 6.2
$\mathrm{H}[S\vert\mathbf{L}_q]$	Conditional entropy	pp 10, sec 4.2
I	Input selection algorithm	pp 29, app B
$\mathrm{I}(S;\mathbf{L}_q)$	Mutual information	pp 10, sec 4.2
$\mathsf{L}(C_\alpha, M, R)$	Leakage function	pp 4, sec 2
$\mathbf{L}_q$, $\mathbf{l}_q$	Side-channel leakage vector	pp 9, sec 4.2
$\mathsf{M}(s^*,.)$	Leakage model for a key class s^*	pp 29, app B
$\mathsf{P}(s^*,.)$	Leakage partition for a key class s^*	pp 29, app B
$\Pr[s\vert\mathbf{l}_q]$	Probability of a key class s given a leakage $\mathbf{l}_q$	pp 9, sec 4.2
$\Pr[S\vert\mathbf{L}_q]$	Probability distribution of a key class variable S given a leakage variable $\mathbf{L}_q$	pp 10, sec 5
R	Leakage reduction mapping	pp 29, app B
$\mathbf{Succ}^{\text{sc-kr-}o,S}_{\mathsf{A}_{\mathsf{E}_K,\mathsf{L}}}$	o^{th}-order success rate of a side-channel key recovery adversary against a key class variable S	pp 8, sec 4.1
$\mathbf{Succ}^{\text{sc-kr-}o,s}_{\mathsf{A}_{\mathsf{E}_k,\mathsf{L}}}$	o^{th}-order success rate of a side-channel key recovery adversary against a key class s	pp 15, sec 6.2
T	Statistical test in a side-channel attack	pp 30, app B
V	Values derivation algorithm	pp 29, app B

Additionally, any quantity estimated through statistical sampling is represented with a hat, *e.g.* a sample mean is denoted as $\hat{\mathbf{E}}(X)$, as sample variance as $\hat{\sigma}^2(X)$, as sample success rate as $\widehat{\mathbf{Succ}}^{\text{sc-kr-}o,S}_{\mathsf{A}_{\mathsf{E}_K,\mathsf{L}}}$, ...