Masking Proofs are Tight
and How to Exploit it in Security Evaluations

Vincent Grosso[1], François-Xavier Standaert[2]

[1] Horst Görtz Institute for IT Security, Ruhr-Universität Bochum, Germany.
[2] ICTEAM - Crypto Group, Université catholique de Louvain, Belgium.

Abstract. Evaluating the security level of a leaking implementation against side-channel attacks is a challenging task. This is especially true when countermeasures such as masking are implemented since in this case: (*i*) the amount of measurements to perform a key recovery may become prohibitive for certification laboratories, and (*ii*) applying optimal (multivariate) attacks may be computationally intensive and technically challenging. In this paper, we show that by taking advantage of the tightness of masking security proofs, we can significantly simplify this evaluation task in a very general manner. More precisely, we show that the evaluation of a masked implementation can essentially be reduced to the one of an unprotected implementation. In addition, we show that despite optimal attacks against masking schemes are computationally intensive for large number of shares, heuristic (soft analytical side-channel) attacks can approach optimality very efficiently. As part of this second contribution, we also improve over the recent multivariate (aka horizontal) side-channel attacks proposed at CHES 2016 by Battistello et al.

1 Introduction

Say you design a new block cipher and want to argue about its resistance against linear cryptanalysis [39]. One naive approach for this purpose would be to launch many experimental attacks. Yet, such a naive approach rapidly turns out to be unsuccessful if the goal is to argue about security levels beyond the computational power of the designer (e.g., 80-bit or 128-bit security for current standards). Hence, symmetric cryptographers have developed a variety of tools allowing them to bound the security of a block cipher against linear cryptanalysis, under sound and well-defined assumptions. As a typical example of these tools, one can cite the wide-trail strategy that has been used in the design of the AES Rijndael [16]. Its main idea is to minimize the bias (i.e., the informativeness) of the best linear characteristics within the cipher, which can be estimated under some independence assumptions thanks to the piling-up lemma.

Interestingly, the last years have shown a similar trend in the field of side-channel security evaluations. That is, while certification practices are still heavily dominated by "attack-based evaluations", solutions have emerged in order to both extend the guarantees and reduce the cost of these evaluations. More precisely, current certification practices focus either on the automatic verification of some minimum (non-quantitative) properties based on so-called leakage detection tools (e.g., [28, 13, 37, 49, 21]), or on the exhibition of concrete attack paths exploiting the detected leakages (typically taking advantages of standard distinguishers such as [11, 9, 48, 26]). But they are anyway unable to claim security levels beyond the measurement efforts of the evaluation laboratory. In order to mitigate this limitation, one first intuitive line of papers proposed tools allowing to easily predict the success rate of some specialized distinguishers, based on parameters such as the noise level of the implementation [46, 25, 17, 34]. In parallel, and following a more standard cryptographic approach trying to be independent of the adversarial strategy, significant progresses have been made in the mathematical treatment of physical security. In particular the masking countermeasure, which is one of the most common methods to improve the security of leaking cryptographic implementations, has been analyzed in several more or less formal models [10, 32, 51, 45, 18]. These works suggest that physical security via masking has strong analogies with the case of linear cryptanalysis. That is, while security against linear cryptanalysis is obtained by ensuring that the XOR of many (local) linear approximations has low bias, masking ensures that every sensitive variable within an implementation is split (e.g., XORed) into several shares that the adversary has to recombine. So intuitively, masking security proofs can be viewed as a noisy version of the piling-up lemma.

Following these advances, the integration of masking proofs as a part of concrete security evaluation practices, undertaken in [19], appears as a necessary next step. And this is especially true when envisioning future cryptographic implementations with high (e.g., $>$ 80-bit) security levels, for which an attack-based certification process is unlikely to bring any meaningful conclusion. So the main objective of this paper is to follow such an approach and to show how masking security proofs can be used to gradually simplify side-channel security evaluations, at the cost of some conservative assumptions, but also some more critical ones (e.g., related to the independence of the shares' leakages).

More precisely, we start from the observation that a so far under-discussed issue in physical security evaluations is the case of attacks taking advantage of multiple leaking intermediate variables (e.g., see [38, 52, 31] for recent references).[1] As put forward in a recent CHES 2016 paper, this issue gains relevance in the context of masked implementations, in view of the (quadratic) cost overheads those implementations generally imply [7]. In this respect, our first contribution is to extend the analysis of masking security proofs from [19] and to show that these proofs remain essentially tight also for multi-target attacks.

Next, and since we aim to discuss the cost of side-channel security evaluations, we propose a simple metric for the evaluation complexity, and use it to extensively discuss the tradeoff between the time needed for a security evaluation and the risks related to the (e.g., independence) assumptions it exploits. As our

[1] Which is an orthogonal concern to the more studied one of exploiting multiple leakage samples per intermediate variable (e.g., see [1] and follow up works).

investigations suggest that the time complexity of optimal side-channel attacks can become a bottleneck when the security levels of masked implementations increase, we additionally study efficient (heuristic) multi-target attacks against masked implementations. Our best attack significantly improves the multivariate (aka horizontal) iterative attack proposed at CHES 2016 by Battistello et al., that we re-frame as a Soft Analytical Side-Channel Attack [52, 31]. Note that our results also provide a complementary view to those of Battistello et al., since they typically fix the masking noise parameter and look for the number of masking shares such that their attack is feasible, while we rather fix the number of shares and estimate the resulting security level in function of the noise.

Eventually, our results show that the security evaluation of a leaking implementation against worst-case attacks taking advantage of all the target intermediate variables that can be enumerated by an adversary (so still limited to the first/last cipher rounds) boils down to the information theoretic analysis of a couple of its samples, for which good tools exist to guarantee a sound treatment [23, 22]. By combining information theoretic evaluations with metric-based bounds for the complexity of key enumeration [43], we can even obtain security graphs for optimal attacks, plotting the success rate in function of the measurement and time complexity, within seconds of computation on a desktop computer.

2 Cautionary remarks

Admittedly, the more efficient evaluations we discuss next are based on a number of simplifying assumptions. In this respect, we first recall that secure masking depends on two conditions: sufficient noise and independent leakages. This paper is about the first condition only. That is, we assume that the independence condition is fulfilled (to a sufficient extent), and study how exploiting all the leakage samples in an implementation allows reducing its noise.

Yet, we note that tools to test/ensure the independence condition are already widely discussed in the literature. Concretely, there are two main issues that can break this assumption. First, imperfect refreshing schemes can cause d'-tuples of leakage samples to be key-dependent with d' lower than the number of shares used in the masking scheme d. A typical example of such an issue was put forward in [15]. The natural solution to avoid it is to use “composable” (e.g., SNI [4]) gadgets and to test the security of the masking description code (i.e., all the instructions defining a masked algorithm) thanks to formal methods [3].

Second, and more critically, different case studies have shown that actual leakage functions can break the independence assumption and recombine (a part of) the shares, e.g., because of transitions in software implementations [14] or glitches in hardware implementations [36]. Nevertheless, in practice such (partial) recombinations typically reduce the (statistical) “security order” of the implementations, captured by the lowest statistical moment of the leakage distribution that is key-dependent (minus one) [5], to some value d'' below the optimal $(d-1)$, while leaving security margins (i.e., $d'' > 1$). As a result, by increasing the number of shares d, one can generally mitigate these physical defaults to a good extent [41, 2]. Furthermore, simple leakage detection tools such as [28, 13, 37, 49, 21] can be used to test what is the security order of an implementation, and these non-independence issues can be reflected in information theoretic evaluations (see [19], Section 4.2). So overall, ensuring the independence of the shares' leakages in a masked implementation is an orthogonal concern to ours. And while non-independence issues may indeed increase the information leakage of the tuples of samples exploited in an high-order side-channel attack, it does not affect the importance/relevance of taking all the exploitable tuples into account in a (worst-case) security evaluation, which is our main concern.

Besides, we also insist that this work is prospective in the sense that our typical targets are masked implementations with (very) large number of shares, aimed at (very) high security levels (e.g., no key recovery with less than 2^{40} measurements). In this respect, we refer to two recently accepted papers (to Eurocrypt 2017) as an excellent motivation for our purposes [29, 5]. In particular, [29] describes AES implementations masked with 5 to 10 shares, that are typical targets for which attack-based evaluations are unlikely to bring meaningful conclusions. Our following discussions describe theoretical tools allowing one to state sound security claims for such implementations. The important message they carry is that even when the independent shares' leakage assumption is guaranteed, one also needs to pay attention to noise. Simple univariate tests are not enough for this purpose. Performing highly multivariate attacks is (very) expensive. We introduce an intermediate path that we believe will become increasingly relevant in the future. Quite naturally, this intermediate path also comes with limitations. Namely, since we focus on (very) high security levels, the bounds we provide are also less accurate, and reported as log-scaled plots for convenience (i.e., we typically ignore the impact of small constants as a first step).

3 Background

3.1 S-box implementations

Our investigations will consider both the unprotected and the masked implementation of an n-bit S-box S taking place in the first round of a block cipher.

For the unprotected case, we denote the input plaintext with x and the secret key with k. We define $y_a = x \oplus k$ as the result of a key addition between x and k, and $y_b = \mathsf{S}(y_a)$ as the S-box output. The vector of the target intermediate variables is further denoted with $\boldsymbol{y} = [y_a, y_b]$ and the leakage vector corresponding to these variables with $\boldsymbol{L} = [L_a, L_b] + \boldsymbol{N}$, where $\boldsymbol{N}$ is a bivariate random variable representing an additive Gaussian noise. We make the usual assumption that the noise covariance matrix is diagonal and each sample L_i has a similar noise variance σ_n^2.[2] Eventually, the deterministic part of the leakage samples are the output of a leakage function L such that $L_i = \mathsf{L}_i(y_i)$, $\forall i \in \{a, b\}$. For simplicity,

[2] The impact of this noise assumption is specifically discussed in Section 5.3.

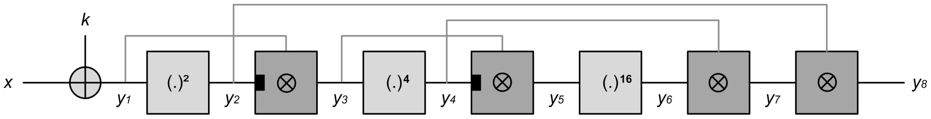

Fig. 1. Multiplication chain for the inversion in GF(2^8) from [47].

our experiments will consider L_i to be the Hamming weight function $\forall i$'s. As discussed in Section 3.2 this choice does not affect our conclusions.

For the masked case, we will focus on the secure inversion in GF(2^8) proposed in [47], which is the core of the AES S-box and illustrated in Figure 1. More precisely, we use a slightly modified version of the algorithms of [47], with the secure refreshing from [32, 4], in order to avoid the attack put forward in [15].[3] Next, we define the notations $y_1 = y_a = x \oplus k$, $y_2 = (y_1)^2$, $y_3 = y_1 \otimes y_2 = (y_1)^3$, $y_4 = (y_3)^4 = (y_1)^{12}$, $y_5 = y_3 \otimes y_4 = (y_1)^{15}$, $y_6 = (y_5)^{16} = (y_1)^{240}$, $y_7 = y_4 \otimes y_6 = (y_1)^{252}$, $y_8 = y_2 \otimes y_7 = (y_1)^{254}$, with $\otimes$ the field multiplication. This leads to a vector of target intermediate variables $\boldsymbol{y} = [y_1, y_2, \ldots, y_8]$. For an implementation masked with d shares, we additionally have a vector of shares $\bar{\boldsymbol{y}} = [\bar{y}_1, \bar{y}_2, \ldots, \bar{y}_8]$ such that $\bar{y}_i = [y_i(1), y_i(2), \ldots, y_i(d)]$ $\forall i \in \{1, 2, \ldots, 8\}$. This leads to a leakage vector $\bar{\boldsymbol{L}} = [\bar{L}_1, \bar{L}_2, \ldots, \bar{L}_8] + \boldsymbol{N}$, where each leakage d-tuple is denoted as $\bar{L}_i = [L_i(1), L_i(2), \ldots, L_i(d)]$ and made of d samples, the multivariate noise variable is defined as in the unprotected case (but with more dimensions) and $L_i(j) = \mathsf{L}_{i,j}(y_i(j))$ $\forall i \in \{1, 2, \ldots, 8\}, j \in \{1, 2, \ldots, d\}$. Such a masking scheme has security order $(d-1)$, meaning that any $(d-1)$-tuple of leakage samples is independent of k, *given that the leakage of each share is independent.* We call this assumption the Independent Shares' Leakage (ISL) assumption.

Concretely, the multiplication chain of Figure 1 is made of squarings, that are GF(2)-linear, and multiplications. In order to evaluate them securely, we use Algorithms 1 and 2 given in Appendix A. For the squarings, the operations are applied to each share independently and therefore can be tabulized. For the multiplications, the different shares need to interact and the algorithm has quadratic overheads that correspond to the computation of all the partial products and their refreshing. For example, for $x = x_1 \oplus \cdots \oplus x_d$ and $y = y_1 \oplus \cdots \oplus y_d$, producing the shares of $x \otimes y$ requires to compute (for $d = 3$):

$$\begin{pmatrix} x_1 \otimes y_1 & x_1 \otimes y_2 & x_1 \otimes y_3 \\ x_2 \otimes y_1 & x_2 \otimes y_2 & x_2 \otimes y_3 \\ x_3 \otimes y_1 & x_3 \otimes y_2 & x_3 \otimes y_3 \end{pmatrix} \oplus \begin{pmatrix} 0 & r_{1,2} & r_{1,3} \\ -r_{1,2} & 0 & r_{2,3} \\ -r_{1,3} & -\,r_{2,3} & 0 \end{pmatrix}. \qquad (1)$$

This directly implies that whenever such a multiplication is targeted by the adversary, we need to add d leakage d-tuples to the leakage vector $\bar{\boldsymbol{L}}$ he is provided with, that we next denote as $[\bar{L}_i^1, \bar{L}_i^2, \ldots, \bar{L}_i^d]$, with $i \in \{3, 5, 7, 8\}$.

[3] Note that more efficient solutions for this secure inversion exist, such as [30]. We kept the chain of Rivain and Prouff because for its simpler description.

Eventually, the GF(2^8) field multiplication is usually implemented using log/alog tables, as described in Appendix A, Algorithm 3. In case the adversary additionally targets these operations, another set of d leakage d-tuples must be added to $\bar{\boldsymbol{L}}$, next denoted as $[\bar{L}_i^{d+1}, \bar{L}_i^{d+2}, \ldots, \bar{L}_i^{2d}]$, with $i \in \{3, 5, 7, 8\}$.

In the following, we will consider different (more or less powerful) attacks cases:

C1. The adversary targets only a single d-tuple (e.g.,, the S-box output one).
C2. The adversary exploits the eight d-tuples of the multiplication chain.
C3. The adversary additionally exploits the leakage of the four secure multiplications (i.e., Algorithm 2), leading to a total of 8 d-tuples and 4 d^2-tuples.
C4. The adversary additionally exploits the leakage of the field multiplications (i.e., Algorithm 3), leading to a total of 8 d-tuples and 8 d^2-tuples.

Furthermore, since a number of these d-tuples contain fresh randomness (e.g., the ones corresponding to multiplications algorithms) while other ones are deterministically related to each other, we will denote with $\delta = \lambda + \ell$ the number of d-tuples exploited, such that we have λ fresh ones and ℓ deterministic ones.[4]

Note that our notations describe serial implementations where the adversary can observe the noisy leakage of each share in his d-tuples separately. This is a relevant choice since serial implementations are typically very expensive to analyze due to their large number of dimensions / leakage samples to consider. Yet, as recently discussed in [5], side-channel security for a serial implementation generally implies side-channel security for its parallel counterpart (as long as the ISL assumption remains fulfilled). So our conclusions apply in this case too.

3.2 Mutual information metric

In order to evaluate the worst-case security level of our different (unprotected and masked) simulated implementations, we will use the mutual information metric first put forward in [50]. The motivation of this choice is twofold. First, it was shown recently that this metric is proportional to the measurement complexity of the corresponding (worst-case) Bayesian adversary [19]. Second, it is significantly faster to estimate than the success rate, which is specially important/relevant in our context where we aim to minimize the evaluator's workload. We illustrate this fact with a simple example. Say an evaluator has 1000,000 measurements to estimate the security of an implementation with a worst-case Bayesian attack that is roughly successful after the collection of 1000 traces. In this case, it means that he can repeat 1000 independent experiments to estimate the success rate with 1000 traces (with good confidence). But say now that the implementation to evaluate can only be broken after (roughly) 1000,000 traces. Then it means that from his set of traces, the evaluator can only estimate the success rate based on a single experiment (which will not lead to any statistical confidence). By contrast, as discussed in [23], cross-validation allows him to exploit most of

[4] The black squares in Figure 1 correspond to additional refreshings needed for the secure multiplication of intermediate variables that are dependent.

his 1000,000 evaluation traces to estimate the mutual information metric, which will then be correlated with the success rate (for any number of traces).[5]

Concretely, computing the mutual information for an unprotected implementation simply requires to estimate the following sum of log probabilities:

$$\mathrm{MI}(K;X,\boldsymbol{L}) = \mathrm{H}[K] + \sum_{k\in\mathcal{K}} \Pr[k]\cdot\sum_{x\in\mathcal{X}}\Pr[x]\cdot\underbrace{\sum_{\boldsymbol{l}\in\mathcal{L}^{\delta}}\Pr[\boldsymbol{l}|k,x]\cdot\log_2\Pr[k|x,\boldsymbol{l}]}_{\delta-\text{dimension integral}}\ , \quad (2)$$

where the conditional probability $\Pr[k|x,\boldsymbol{l}]$ is computed from the Probability Density Function (PDF) $\mathsf{f}[\boldsymbol{l}|x,k]$ thanks to Bayes' theorem as: $\frac{\mathsf{f}[\boldsymbol{l}|x,k]}{\sum_{k^*}\mathsf{f}[\boldsymbol{l}|x,k^*]}\cdot$ This corresponds to performing δ-dimensional integrals over the leakage samples, for each combination of the key k and plaintext x, or each bitwise XOR between k and x if taking advantage of the Equivalence under Independent Subkeys (EIS) assumption formalized in [48]. There are numerous publications where this metric has been computed, via numerical integrals or sampling (e.g., [19] provides an open source code for it), so we do not detail its derivation further.

When moving to masked implementations, the computation of the metric remains essentially similar. The only difference is that we need to sum over the randomness vector $\bar{\boldsymbol{y}}$ (which may become computationally intensive as the number of shares increases, as discussed in the next sections):

$$\begin{aligned}\mathrm{MI}(K;X,\bar{\boldsymbol{L}}) = \mathrm{H}[K] + \sum_{k\in\mathcal{K}}\Pr[k]\cdot\sum_{x\in\mathcal{X}}\Pr[x]\cdot\\ \sum_{\bar{\boldsymbol{y}}\in\mathcal{Y}^{(d-1)\cdot\lambda}}\Pr[\bar{\boldsymbol{y}}]\ \cdot\ \underbrace{\sum_{\bar{\boldsymbol{l}}\in\mathcal{L}^{d\cdot\delta}}\Pr[\bar{\boldsymbol{l}}|k,x,\bar{\boldsymbol{y}}]\cdot\log_2\Pr[k|x,\bar{\boldsymbol{l}}]}_{\delta-\text{dimension integral}}\ .\end{aligned} \quad (3)$$

The computation of the conditional probability $\Pr[k|x,\boldsymbol{l}]$ follows similar guidelines as in the unprotected case, where the PDF of masked implementations becomes a mixture that can be written as $\mathsf{f}[\boldsymbol{l}|x,k] = \sum_{\bar{\boldsymbol{y}}}\mathsf{f}[\boldsymbol{l}|x,k,\bar{\boldsymbol{y}}]$ [33, 51].

Remark. In our experiments where the (simulated) noise is Gaussian, we will use a Gaussian PDF in the unprotected case, and a Gaussian mixture PDF in the masked case. Since we know the PDF exactly in these cases, we can compute the MI metric exactly and perform worst-case security evaluations. However, we insist that our discussions relate to the *complexity* of side-channel security evaluations, not their *optimality*. More precisely, our goal is to show that we can significantly simplify the evaluation of a highly protected implementation.

[5] Note that the mutual information metric is not the only one allowing to simplify the estimation of a security level for a leaking cryptographic implementation. However, it is the most generic one since it does not require assumptions on the leakage distribution, nor on the choice of concrete distinguisher chosen by the adversary. More specialized (and sometimes more efficient) solutions include [46, 25, 17, 34].

These efficiency gains are independent of the leakage function and model used by a concrete adversary. The main difference, if a concrete adversarial model was used in place of the perfect one, is that the log probabilities in Equations 2 and 3 would be evaluated based on it. This implies that less information would be extracted in case of model estimation or assumption errors, which is again an orthogonal concern to ours. Furthermore, leakage certification could then be used to test whether estimation and assumption errors are small enough [23, 22].

4 Unprotected implementations

Evaluation complexity. Since our goal is to make side-channel security evaluations more efficient, a first question is to specify how we will evaluate complexity. Eventually we are interested in the measurement complexity of the attacks, which masking is expected to increase exponentially (in the number of shares). But of course, we also want to be able to evaluate the security of implementations of which the security is beyond what we can actually measure as evaluators. As just mentioned, computing the mutual information metric is an interesting tool for this purpose. But it means that we still have to compute Equations 2 and 3, which are essentially made of a sum of δ-dimension integrals. Concretely, the complexity of computing the integrals is highly dependent on the choice of PDF estimation tool chosen by the adversary/evaluator. So this suggests the number of integrals to perform as a natural candidate for the complexity of a side-channel evaluation, which we will denote with $\mathsf{E}_\$$ in the following.

In the case of an unprotected S-box implementation in GF(2^m), this leads to $\mathsf{E}_\$ = 2^{2m}$ in general (since we sum over 2^m key bytes and 2^m plaintext bytes). This complexity is reduced to $\mathsf{E}_\$ = 2^m$ if we take advantage of the EIS assumption. Since the latter assumption is generally correct in the "standard DPA" attack context we consider in this paper [35], we will always consider the complexity of evaluations taking advantage of EIS in the following (ignoring this simplification implies an additional 2^m factor in the evaluation complexities).

Practical evaluation results. As suggested by the previous formula, evaluating the security of an unprotected (8-bit) S-box is cheap. We now report on some exemplary results which we use to introduce an important assumption regarding our following simplifications. We consider different attack cases:

- Univariate, no repetition: the adversary observes the S-box output leakage.
- Univariate, with repetitions: the adversary observes the S-box output leakage several times with independent noise samples (e.g., 2 times, 4 times).
- Bivariate: the adversary observes the S-box input and output leakage.

Additionally, we consider a "bivariate attack bound" which is just the sum of two "univariate, no repetition" curves. In order to allow an easier interpretation of the results, we use the Signal-to-Noise Ratio (SNR) as X axis, defined as the variance of the noise-free traces (i.e., $m/4$ for a Hamming weight model) divided by the variance of the noise. It better reflects the fact that the impact of the noise

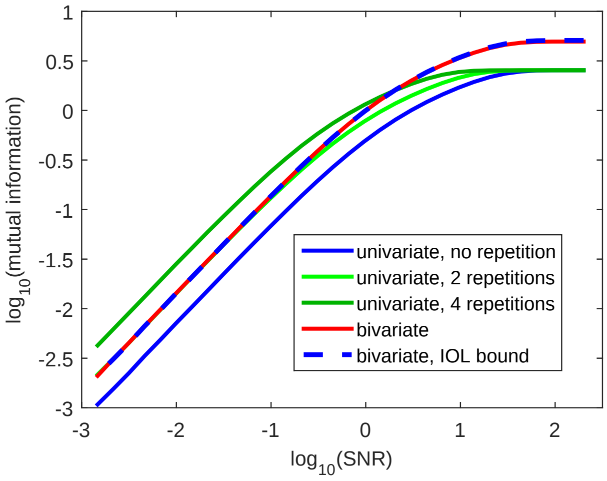

Fig. 2. Unprotected AES S-box evaluation results.

depends on the scaling of the signal. The results of these information evaluations are given in Figure 2 from which two main observations can be extracted.

First, we observe the difference between the impact of repeated observations, which just reduce the noise and therefore translate the information curves on the left, and bivariate attacks which (may) add information and shift these curves vertically. Interestingly, the latter observation is dependent on the S-boxes [44]: an identity S-box would lead to a repetition without information gain; a truly random one would lead to independent information for the two observations.

Second, we observe that the bivariate attack bound is tight. This suggests that the AES S-box leads to quite independent information for the leakage samples L_a and L_b of our case study, which is consistent with the conclusions in [44]. Formally, we will say that this bound is tight if the Independent Operations' Leakages (IOL) assumption holds, which considers that the inputs/outputs of an operation (i.e., the AES S-box in our case study) are independent.

Note that as for the ISL assumption, the latter does not mean that the noise of the leakage samples has to be independent (which will be discussed in Section 5.2). Note also that the impact of a deviation from this IOL assumption is very different than with the ISL assumption. Namely, if the share's leakages are not independent, then the formal security guarantees of masking vanish. By contrast, if the operation leakages are not independent, this will lead to less information and therefore less effective attacks. So the IOL assumption is not critical for the conclusion of a security evaluation: overstating IOL may only lead to less tight security bounds. Note finally that in our AES case study, assuming IOL (also for the multiplicative chain of Figure 1) led to tight bounds.

5 Masked implementations

We now move to the context of masked implementations which is the main contribution of this paper. We start by arguing that an exhaustive security evaluation is rapidly unreachable as the number of shares in masking increases. We then gradually simplify the evaluations, first without critical assumptions on the leakage distributions, second by exploiting the ISL assumption.

5.1 Exhaustive approach

By visual inspection of Equation 3, we directly find that the evaluation complexity $\mathsf{E}_\$ = 2^{dm\lambda} + \ell \cdot 2^{dm}$, where we recall that λ is the number of fresh dimensions and ℓ the number of deterministic ones. For the case C1 in Section 3.1 with $d = 2$ shares, where the adversary targets only one 2-tuple of leakage samples corresponding to the masked S-box output y_8 in Figure 1, this means a reachable 2^{2m} integrals. But as soon as we move to a (slightly) more powerful adversary, the complexity explodes. For example, the adversary of case C2 (who is still not optimal) with $m = 8$, $d = 2$, $\lambda = 4$ (due to the four multiplications in Figure 1) and $\ell = 4$ (due to the key addition and squarings), already leads to $\mathsf{E}_\$ > 2^{64}$ integrals which is by far too expensive for evaluation laboratories.[6]

5.2 Reducing dimensionality with the IOL assumption

The first factor in cause in the complexity explosion of the exhaustive approach is the number of fresh dimensions. In this respect, a natural simplification is to exploit the IOL assumption. Indeed, by considering the operations in the multiplication chain of Figure 1 as independent, the evaluation complexity of the previous (C2) adversary can be reduced to $\mathsf{E}_\$ = \delta \cdot (2^{dm}) = 8 \cdot 2^{16}$ integrals. This is an interesting simplification since it in fact directly corresponds to the strategy of an adversary willing to perform a multivariate attack against such a leaking masked implementation. Namely, he will identify the eight d-tuples of interest and combine their results via a maximum likelihood approach. We report the result of an information theoretic evaluation of this C2 adversary in Figure 3, where we also plot the IOL bound provided by multiplying the information theoretic curve of the C1 adversary by eight. As for the case of unprotected implementations, the bound is tight, confirming its relevance.

Nevertheless, this simplification also implies two important technical questions. First, and since we assume the leakage of independent operations to be independent, what would be the impact of a dependent noise? Second, how to generalize this simplification to the adversaries C3 and C4 which imply the need of considering d^2-tuples jointly (rather than d-tuples jointly in the C2 case)?

[6] Still ignoring the additional refreshings mentioned in Footnote 1.

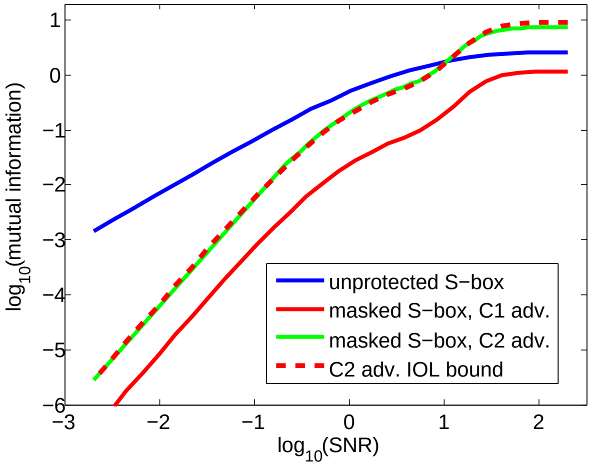

Fig. 3. Masked AES S-box evaluation results: cases C1 & C2 ($d = 2$).

5.3 The dependent noise issue

To the best of our knowledge, this noise dependency issue has not been specifically discussed in the literature on masking, although the existence of correlated noise has been put forward in other contexts (e.g., see the discussion in [12], Chapter 6). We therefore launched an information theoretic evaluation of our masked S-box (case C1) with $d = 2$ and the covariance matrix such that the correlation between the noise samples of the two shares equals 0, 0.25, 0.5 and 0.75. The results of these evaluations are in Figure 4. As expected, a correlated noise does not impact the security order of the countermeasure, defined as the lowest key-dependent moment in the leakage distribution $\Pr[k|x, \bar{\boldsymbol{l}}]$ minus one, and reflected by the slope of the information theoretic curves in the high-noise region (i.e., where the curves are linear) minus one. By contrast, correlated noise implies a shift of the curves by a factor that can be significant (e.g., ×2 for correlation 0.5 and ×8 for correlation 0.75). Such large correlations typically vanish after a couple of clock cycles. Yet, our results highlight that estimating the non-diagonal elements of the noise covariance matrices in masked implementations is an important sanity check that should be part of a certification process.

5.4 Secure multiplication leakages

When also considering the leakages of the d^2 cross products involved in a secure multiplications (such as the ones of Equation 1 in Section 3.1 for $d = 3$), and additional problem is that computing an integral of d^2 dimensions rapidly becomes computationally intensive. This is particularly true if one considers an optimal Gaussian mixture model for the PDF since in this case the computation of the

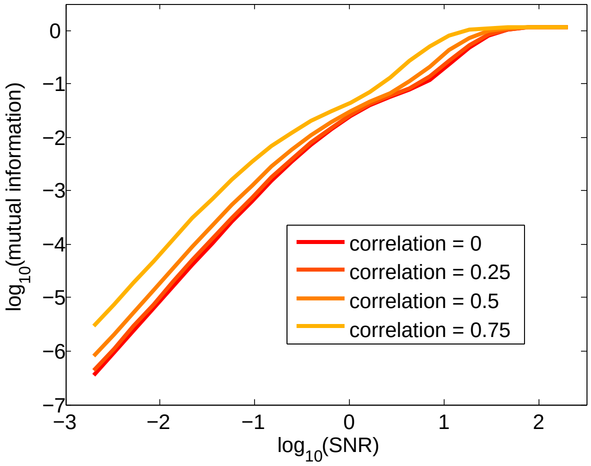

Fig. 4. Masked AES S-box evaluation results: case C1 with correlated noise ($d = 2$).

integral requires summing over the randomness vector. In fact, already for small field sizes and number of shares, the problem is hard. For example, for $d = 2$ and $m = 8$, the multiplication between two dependent values such as required in the multiplication chain of Figure 1 requires performing 2^{24} integrals (corresponding to 8 bits of secret and twice 8 bits of randomness) of a 9-dimensional PDF.[7]

In order to deal with this limitation, a solution is to look at masking proofs. In particular, Theorem 3 in [45] and Theorem 2 in [18] both provide bounds on the amount of information leaked by the multiplication of two secrets shared with Boolean masking, roughly corresponding to $(1.72d + 2.72)$ and $(28d + 16)$ times the information leakage of a single d-tuple. In this respect, there are again two important questions. First, are these bounds (and in particular the first one) tight? Second, given that the evaluation with an optimal attack becomes computationally intensive for large d values as just argued, does it mean that these bounds are unreachable by adversaries with realistic computing power?

We answer these questions in two steps. First, we investigate a simplified context with small d and m values such that the optimal attack is applicable. Second, we discuss heuristic attacks which approach the optimal attack efficiently.

Simplified case study. Figure 5 shows the information theoretic evaluation of a secure multiplication with $d = 3$ and $m = 2$.[8] We can clearly observe the larger leakage of optimal attack exploiting the $\delta = 9$ dimensions of the multiplication

[7] And as aforementioned, the latter integral itself requires to sum over 2^{24} values if an optimal Gaussian mixture model is used by the adversary/evaluator.

[8] Due to the large number of dimensions, the integrals were computed via sampling in this case, which also explains the lower noise variances that we could reach. However,

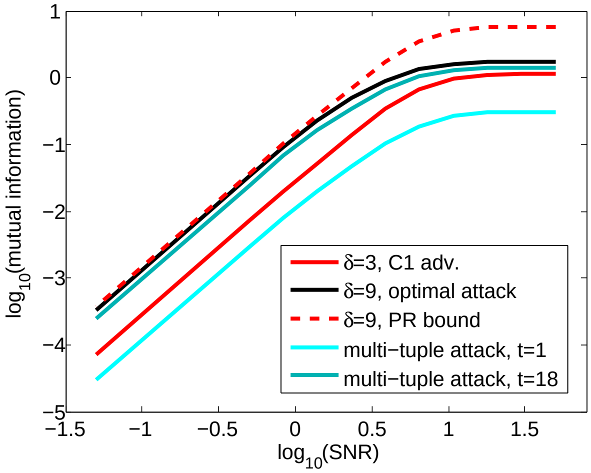

Fig. 5. Secure multiplication evaluation results ($d = 3$, $m = 2$).

jointly, compared to the information provided by the encoding (i.e., the C1 adversary). As for the bounds, we first note that a simple (intuitive) bound is to assume that given two dependent values that are multiplied together, one leaks d horizontal d-tuples corresponding to one value (assuming the other to be known) and another d vertical d-tuples corresponding to the other value (under the same assumption). This leads to an estimation of the multiplication matrix leakage as $2d$ times the one of a single d-tuple, which is close to the $1.72d$ factor given by Prouff and Rivain in [45]. Hence, we added the latter bound on the figure (under the name PR bound). Concretely, it simply consists in multiplying the information of the encoding by $1.72d$ and turns out to be remarkably tight as soon as a sufficient amount of noise affects the measurements.[9]

Heuristic attacks. As the optimal attack in the previous paragraph becomes computationally intensive for large d and m values, we now consider alternatives that allow an adversary to exploit the information leakage of the multiplication matrix without summing over all the randomness and considering all the dimensions jointly. A first candidate is the recursive attack proposed by Battistello et al. at CHES 2016 [7]. In the following, we revisit and improve this attack by framing it as a Soft Analytical Side-Channel Attack (SASCA) [52, 31].

In a SASCA, the adversary essentially describes all the leaking operations in his target implementation as a “factor graph” and then decodes the leakage information by exploiting the Belief Propagation (BP) algorithm. The main in-

we note that these lower noise levels were sufficient to reach the asymptotic (i.e., linear) regions of the information theoretic curves supporting our conclusions.

[9] Note that a parallel implementation would lead to a slightly better bound of $\approx d$ since reducing the amount of observable leakage samples by a factor d [5].

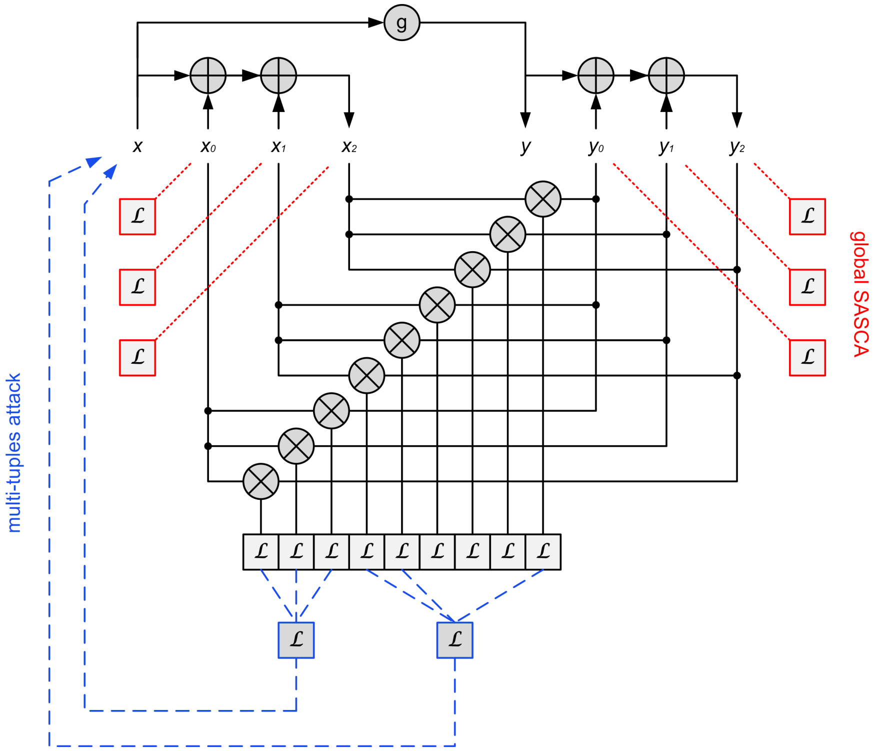

Fig. 6. Factor graph of a secure multiplication ($d = 3$).

terest of this approach is that it allows combining the information of multiple leaking instructions (e.g., the cross products in a secure multiplication) locally, without the need to consider them jointly. Its time complexity depends on the diameter of the factor graph (which is constant when all target intermediate variables are directly connected as in the secure multiplication), the cost of the probabilities' updates (which is constant and depends on the bit size of the operations considered) and the number of these updates (which depends on the size of the factor graph and grows quadratically in d). The factor graph of a secure multiplication with $d = 3$ shares is pictured in Figure 6. Its only specificity is that for the BP algorithm to succeed, we need to initialize the leakage on the shares x_0, x_1, x_2 and y_0, y_1, y_2, which means that a SASCA must consider the target operations more globally. In our experiments, we just add the leakage of these shares which can be obtained, e.g., when loading them into a register.

An alternative (and conceptually simple) approach allowing to get rid of the need of initialization is to always target d-tuples of informative leakage samples jointly. Such a "multi-tuple attack" can be viewed as an intermediate between the optimal attack targeting d^2 samples jointly and the previous SASCA targeting samples one by one, as illustrated in Figure 6. More precisely, the optimal attack outlined in Section 3.2 exploits a leakage PDF $\Pr[\bar{\boldsymbol{l}}_{d^2}|k,x,\bar{\boldsymbol{y}}_{d^2}]$, where the d^2 subscripts of the vectors $\bar{\boldsymbol{l}}_{d^2}$ and $\bar{\boldsymbol{y}}_{d^2}$ now highlight their number of dimensions. In a multi-tuple attack, we simply select a number of d-tuples of which the combination depends on the target secret and approximate:

$$\Pr\left[\bar{\boldsymbol{l}}_{d^2}|k,x,\bar{\boldsymbol{y}}_{d^2}\right] \approx \Pr\left[\bar{\boldsymbol{l}}_d^1|k,x,\bar{\boldsymbol{y}}_d^1\right] \cdot \Pr\left[\bar{\boldsymbol{l}}_d^2|k,x,\bar{\boldsymbol{y}}_d^2\right] \cdot \ldots \cdot \Pr\left[\bar{\boldsymbol{l}}_d^t|k,x,\bar{\boldsymbol{y}}_d^t\right],$$

where t is the number of tuples exploited.[10] As illustrated in Figure 5, an attack using a single d-tuple (e.g., here a matrix line) only leads to little exploitable information, which is consistent with the observations in [7]. By contrast, increasing t rapidly allows reaching a close-to-optimal attack.

Note that the multi-tuples attack still does not scale well since the total number of informative d-tuples in the matrix multiplications grows following a binomial rule. So the most appealing attacks to target the secure multiplication algorithm are the CHES 2016 iterative and the SASCA. Unfortunately, in these cases we face the problem that the heuristic nature of the decoding algorithms (which both propagate information locally without formal guarantees of convergence) does not formally lead them to output probabilities. Typically, by iterating the CHES 2016 and BP algorithms more, it is possible to artificially crush the probabilities of the variable nodes in the factor graph. So formally, we cannot evaluate the mutual information metric in this case. As a result, and for this part of our experiments only, we directly evaluated the success rate of an optimal attack, a SASCA and the CHES 2016 iterative attack (using exactly the same leaking operations as the SASCA) for various noise levels. Figure 7 contains the result of one such experiments (for $\sigma_n^2 = 10$) where we observe that (i) the SASCA leads to attack efficiencies that approach the optimal one, and (ii) the SASCA outperforms the CHES 2016 iterative attack. The latter observation is easily explained since the CHES 2016 iterative attack can in fact be viewed as a modified version of SASCA. Namely, the main difference between the SASCA and the CHES 2016 iterative attack is the fact we take advantage of the relation between the two secrets that are multiplied (i.e., the g function in Figure 6), which allows the BP algorithm to extract more information (while the factor graph of the CHES 2016 iterative attack ignores this connection).[11]

Remark. As previously mentioned, extending these experiments to larger d and m values is not possible because the optimal attack becomes too expensive (computationally). By contrast, we could check that the success rate curves of the SASCA consistently outperform the ones of the CHES 2016 iterative attack by

[10] Note that whenever an imperfect model is used by the adversary/evaluator, the estimation of Equations 2 and 3 does not strictly converge towards the mutual information, but only to the so-called perceived information discussed in [23].

[11] Technically, the rules used for updating the probabilities in the CHES 2016 attack are also presented slightly differently than in SASCA, where the BP algorithm is explicitly invoked with variable to factors and factors to variable message passing.

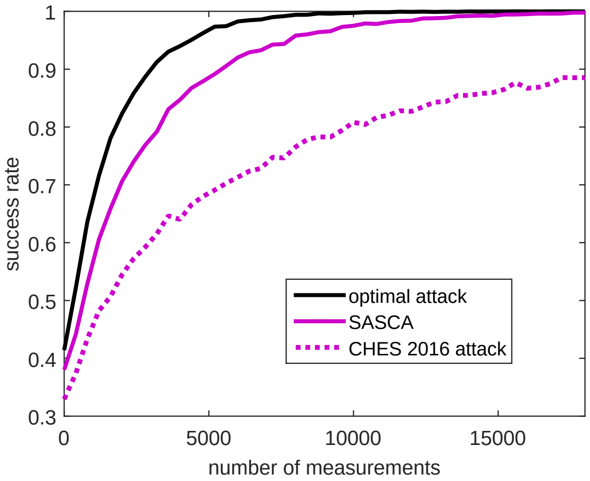

Fig. 7. Optimal attack vs. efficient heuristics ($d = 3$, $m = 2$, $\sigma_n^2 = 10$).

an approximate factor > 2 in measurement complexity, for larger m values. For example, we report the result of such a comparison for the relevant $m = 8$-bit case corresponding to the AES S-box in Appendix B, Figure 12.

Overall, we conclude from this section that the IOL assumption and the PR bound for secure multiplications give rise to quite tight estimations of the information leakage of a masked implementation (at least for the leakage functions and noise levels considered). Furthermore, this leakage can generally be exploited quite efficiently using heuristics such as the BP algorithm. We conjecture that this observation generally remains correct for most leakage functions, especially when the number of shares in the masking schemes increases.

5.5 Reducing cardinality with the ISL assumption

Eventually, the previous experiments suggest that the evaluation of a masked implementation against multivariate attacks can boil down to the evaluation of the information leakage of a d-tuple. Yet, this still has evaluation cost proportional to 2^{dm}. Fortunately, at this stage we can use the ISL assumption and the bound discussed at Eurocrypt 2015 showing that this information can be (very efficiently) computed based on the information of a single share (essentially by raising this information to the security order), which has (now minimal) evaluation cost $\mathsf{E}_\$ = \delta \cdot 2^m$ (or even 2^m if one assumes that the leakage function of the target implementation is similar for all operations, or if we bound the overall leakage based on the most informative d-tuple found) [19]. For completeness, we illustrate such a result in Figure 8, where we compare the bound (denoted as DFS) and the true information leakage for $d = 2$, and only plot the bound

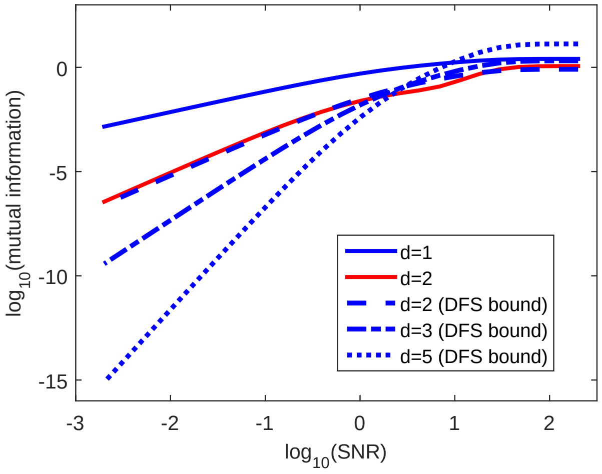

Fig. 8. Masked AES S-box evaluation results: case C1 with ISL assumption.

for larger d's. As already mentioned, the big conceptual change at this step of our simplifications is that the ISL assumption is no longer a conservative one. If it turns out to be incorrect, then the security order of higher-order masking schemes may be less than predicted by the number of shares. Yet, as discussed in Section 2, this does not decrease the relevance of our method and bounds: it simply implies that applying them first requires to assess the security order.

Note also that as carefully discussed in [19], the DFS bound is only conjectured and ignores a square root loss in the reduction from the mutual information to the statistical distance used in the proofs. Yet, this square root loss vanishes when the noise increases (as per the upper bound in [45]), which was also confirmed experimentally in previous works such as [51]. In this respect, we recall that masking proofs are anyway only relevant for large enough noises (or low enough SNRs), which corresponds to the linear (left) parts of the information theoretic curves of Figure 8 (i.e., where the DFS bound is tight).[12]

6 Fast and sound leakage assessment

6.1 Putting things together

By combining the IOL assumption, the PR bound for evaluating the leakage of a secure multiplication, the ISL assumption and the DFS bound for evaluating the leakage of an encoding with large number of shares, all evaluated and discussed in

[12] Technically, this is reflected by a mutual information that can go beyond the maximum m when the noise is too low, which corresponds to the fact that the bound then raises the information of a single share that is larger than one to a certain power. For convenience, the following plots will limit the mutual information to m.

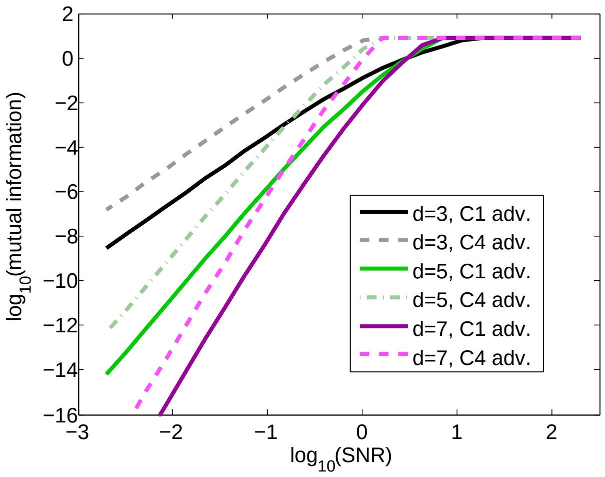

Fig. 9. Masked AES S-box evaluation results: cases C1 & C4 (with all assumptions).

the previous section, we can now easily obtain the results of a security evaluation for the four adversaries outlined in Section 3.1. For example, Figure 9 plots them for $d = 3, 5$ and 7 shares, for various noise levels. For readability, we only provide the results of the extreme attacks (C1 and C4). These curves are simply obtained by performing powers and sums of the information theoretic curve for the simplest possible case $d = 1$. In other words, we can evaluate the leakage of a masked implementation against optimal (highly multivariate) side-channel attacks at the cost of the evaluation of an unprotected implementation.

Note that the curves clearly highlight the need of a higher noise level when implementing higher-order masking schemes, in order to mitigate the noise reduction that is caused by the possibility to perform highly multivariate attacks (reflected by a shift of the curves towards the left of the figure). And quite naturally, they directly allow one to quantify the increasing impact of such attacks when the security order increases. For example, the factor between the measurement complexity of the adversary C1 (exploiting one tuple of leakage samples) and the optimal C4 ranges from 50 (for $d = 3$) to 100 (for $d = 7$).

In this respect, there is one final remark. In concrete implementations, it frequently happens that some of the target intermediate values appear several times (e.g., because they need to be reloaded for performing the cross products in a secure multiplication). In this case, the adversary can additionally average the noise for these target intermediate values, as proposed in [7]. As mentioned in Section 4, such an effect is also easy to integrate into our evaluations since it only corresponds to a shift of the information theoretic curves. However, it is worth emphasizing that this averaging process is applied to the shares (i.e., before their combination provides noise amplification), which implies that it is

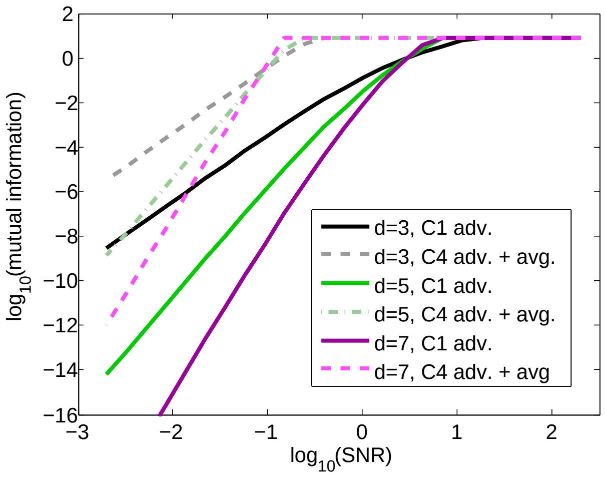

Fig. 10. Masked AES S-box evaluation results: cases C1 & C4 (with all assumptions & d-times averaging applied to the shares of the secure multiplications).

extremely damaging for the security of masking. Concretely, this means that averaging the leakage samples of a masked implementation with d shares by a factor d (because these shares are loaded d times to perform cross products) may lead to a reduction of the security level by a factor d^d. For illustration, Figure 10 shows the result of such a security evaluation in a context similar to Figure 9, where the shares of each the masked multiplication are averaged d times, this times causing reductions of the security level by several orders of magnitude.

6.2 Exploiting computational power

Eventually, and given some mutual information value extracted from the previous plots, we mention that one can easily insert this value in a metric-based bound in order to build a security graph, such as suggested in [43, 20] and illustrated in Figure 11. While such metric-based bounds only provide a conservative estimation of the impact of key enumeration in a side-channel attack, they are again obtained withing seconds of computation on a desktop computer. We detail how to build such a graph and the heuristics we rely on in Appendix C.

6.3 Conclusions

1. On too simple evaluation methodologies. Looking at the power of multivariate (aka horizontal) side-channel attacks taking advantage of all the leaking operations in the multiplicative chain of a masked AES S-box, an important conclusion is that simple (univariate) evaluation strategies become increasingly irrelevant as the number of shares in a masked implementation increases.

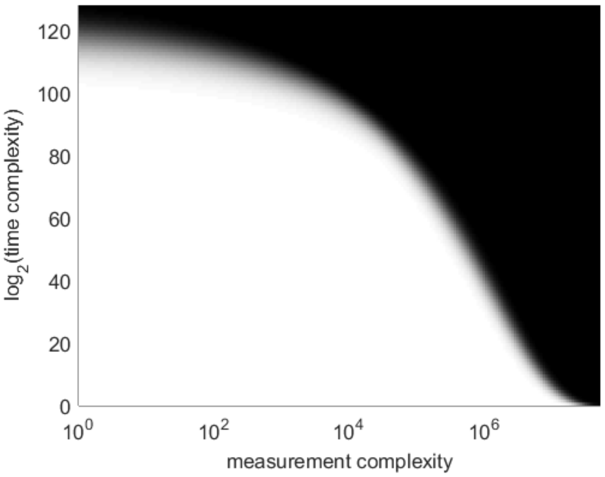

Fig. 11. Exemplary metric-based bound for a security graph (with MI $= 10^{-7}$).

2. On the need of formal methods and security order detection. As made clear in Section 2, the tools we provide in this paper only solve the "noise" part of the security evaluation problem for masked implementations. Hence, their combination with formal methods and security order detection techniques is an interesting scope for further research. Typically, one could extend the tools put forward in [3] in order to detect all the leaking operations in an assembly code (possibly with repetitions), then use leakage detection methods such as [28, 13, 37, 49, 21] to assess the security order of actual measurements, and finally evaluate their informativeness as we suggest in this paper, in order to obtain a fast assessment of the worst-case security level of an implementation.

3. On how to reach high security levels. Eventually, our results show that ensuring high security levels against optimal adversaries taking advantage of all the information provided by a masked implementation is very challenging. It typically requires many shares, high noise levels and independence. In this respect, the application of our theoretical progresses to the aforementioned implementations [29, 5], to alternative multiplication chains [15, 30], to the optimized algorithms in [8], or even to new primitives allowing more efficient masking (e.g., the proposal in [24] of which the complexity scales linearly in the number of shares and is well suited to guarantee the ISL assumption), and the combination of these ideas with parallel implementations (which inherently improve security against multivariate attacks), is another interesting scope for further research.

Acknowledgments. Vincent Grosso is funded by the DFG Research Training Group GRK 1817 Ubicrypt. François-Xavier Standaert is an associate researcher of the Belgian Fund for Scientific Research (FNRS-F.R.S.). This work has been funded in parts by the ERC project 724725 (acronym SWORD).

References

1. Cédric Archambeau, Eric Peeters, François-Xavier Standaert, and Jean-Jacques Quisquater. Template attacks in principal subspaces. In Louis Goubin and Mitsuru Matsui, editors, *Cryptographic Hardware and Embedded Systems - CHES 2006, 8th International Workshop, Yokohama, Japan, October 10-13, 2006, Proceedings*, volume 4249 of *Lecture Notes in Computer Science*, pages 1–14. Springer, 2006.
2. Josep Balasch, Benedikt Gierlichs, Vincent Grosso, Oscar Reparaz, and François-Xavier Standaert. On the cost of lazy engineering for masked software implementations. In Marc Joye and Amir Moradi, editors, *Smart Card Research and Advanced Applications - 13th International Conference, CARDIS 2014, Paris, France, November 5-7, 2014. Revised Selected Papers*, volume 8968 of *Lecture Notes in Computer Science*, pages 64–81. Springer, 2014.
3. Gilles Barthe, Sonia Belaïd, François Dupressoir, Pierre-Alain Fouque, Benjamin Grégoire, and Pierre-Yves Strub. Verified proofs of higher-order masking. In Oswald and Fischlin [42], pages 457–485.
4. Gilles Barthe, Sonia Belaïd, François Dupressoir, Pierre-Alain Fouque, Benjamin Grégoire, Pierre-Yves Strub, and Rébecca Zucchini. Strong non-interference and type-directed higher-order masking. In Edgar R. Weippl, Stefan Katzenbeisser, Christopher Kruegel, Andrew C. Myers, and Shai Halevi, editors, *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, Vienna, Austria, October 24-28, 2016*, pages 116–129. ACM, 2016.
5. Gilles Barthe, François Dupressoir, Sebastian Faust, Benjamin Grégoire, François-Xavier Standaert, and Pierre-Yves Strub. Parallel implementations of masking schemes and the bounded moment leakage model. *IACR Cryptology ePrint Archive*, 2016:912, 2016.
6. Lejla Batina and Matthew Robshaw, editors. *Cryptographic Hardware and Embedded Systems - CHES 2014 - 16th International Workshop, Busan, South Korea, September 23-26, 2014. Proceedings*, volume 8731 of *Lecture Notes in Computer Science*. Springer, 2014.
7. Alberto Battistello, Jean-Sébastien Coron, Emmanuel Prouff, and Rina Zeitoun. Horizontal side-channel attacks and countermeasures on the ISW masking scheme. In Gierlichs and Poschmann [27], pages 23–39.
8. Sonia Belaïd, Fabrice Benhamouda, Alain Passelègue, Emmanuel Prouff, Adrian Thillard, and Damien Vergnaud. Randomness complexity of private circuits for multiplication. In Marc Fischlin and Jean-Sébastien Coron, editors, *Advances in Cryptology - EUROCRYPT 2016 - 35th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Vienna, Austria, May 8-12, 2016, Proceedings, Part II*, volume 9666 of *Lecture Notes in Computer Science*, pages 616–648. Springer, 2016.
9. Eric Brier, Christophe Clavier, and Francis Olivier. Correlation power analysis with a leakage model. In Marc Joye and Jean-Jacques Quisquater, editors, *Cryptographic Hardware and Embedded Systems - CHES 2004: 6th International Workshop Cambridge, MA, USA, August 11-13, 2004. Proceedings*, volume 3156 of *Lecture Notes in Computer Science*, pages 16–29. Springer, 2004.
10. Suresh Chari, Charanjit S. Jutla, Josyula R. Rao, and Pankaj Rohatgi. Towards sound approaches to counteract power-analysis attacks. In Michael J. Wiener, editor, *Advances in Cryptology - CRYPTO '99, 19th Annual International Cryptology Conference, Santa Barbara, California, USA, August 15-19, 1999, Proceedings*, volume 1666 of *Lecture Notes in Computer Science*, pages 398–412. Springer, 1999.

11. Suresh Chari, Josyula R. Rao, and Pankaj Rohatgi. Template attacks. In Burton S. Kaliski Jr., Çetin Kaya Koç, and Christof Paar, editors, *Cryptographic Hardware and Embedded Systems - CHES 2002, 4th International Workshop, Redwood Shores, CA, USA, August 13-15, 2002, Revised Papers*, volume 2523 of *Lecture Notes in Computer Science*, pages 13–28. Springer, 2002.
12. Omar Marios Choudary. Efficient multivariate statistical techniques for extracting secrets from electronic devices. PhD Thesis, University of Cambridge, 2014.
13. Jeremy Cooper, Elke De Mulder, Gilbert Goodwill, Josh Jaffe, Gary Kenworthy, and Pankaj Rohatgi. Test vector leakage assessment (TVLA) methodology in practice (extended abstract). ICMC 2013. `http://icmc-2013.org/wp/wp-content/uploads/2013/09/goodwillkenworthtestvector.pdf`.
14. Jean-Sébastien Coron, Christophe Giraud, Emmanuel Prouff, Soline Renner, Matthieu Rivain, and Praveen Kumar Vadnala. Conversion of security proofs from one leakage model to another: A new issue. In Werner Schindler and Sorin A. Huss, editors, *Constructive Side-Channel Analysis and Secure Design - Third International Workshop, COSADE 2012, Darmstadt, Germany, May 3-4, 2012. Proceedings*, volume 7275 of *Lecture Notes in Computer Science*, pages 69–81. Springer, 2012.
15. Jean-Sébastien Coron, Emmanuel Prouff, Matthieu Rivain, and Thomas Roche. Higher-order side channel security and mask refreshing. In Shiho Moriai, editor, *Fast Software Encryption - 20th International Workshop, FSE 2013, Singapore, March 11-13, 2013. Revised Selected Papers*, volume 8424 of *Lecture Notes in Computer Science*, pages 410–424. Springer, 2013.
16. Joan Daemen and Vincent Rijmen. The wide trail design strategy. In Bahram Honary, editor, *Cryptography and Coding, 8th IMA International Conference, Cirencester, UK, December 17-19, 2001, Proceedings*, volume 2260 of *Lecture Notes in Computer Science*, pages 222–238. Springer, 2001.
17. A. Adam Ding, Liwei Zhang, Yunsi Fei, and Pei Luo. A statistical model for higher order DPA on masked devices. In Batina and Robshaw [6], pages 147–169.
18. Alexandre Duc, Stefan Dziembowski, and Sebastian Faust. Unifying leakage models: From probing attacks to noisy leakage. In Nguyen and Oswald [40], pages 423–440.
19. Alexandre Duc, Sebastian Faust, and François-Xavier Standaert. Making masking security proofs concrete - or how to evaluate the security of any leaking device. In Oswald and Fischlin [42], pages 401–429.
20. Alexandre Duc, Sebastian Faust, and François-Xavier Standaert. Making masking security proofs concrete or how to evaluate the security of any leaking device (extended version). *IACR Cryptology ePrint Archive*, 2015:119, 2015.
21. François Durvaux and François-Xavier Standaert. From improved leakage detection to the detection of points of interests in leakage traces. In Marc Fischlin and Jean-Sébastien Coron, editors, *Advances in Cryptology - EUROCRYPT 2016 - 35th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Vienna, Austria, May 8-12, 2016, Proceedings, Part I*, volume 9665 of *Lecture Notes in Computer Science*, pages 240–262. Springer, 2016.
22. François Durvaux, François-Xavier Standaert, and Santos Merino Del Pozo. Towards easy leakage certification. In Gierlichs and Poschmann [27], pages 40–60.
23. François Durvaux, François-Xavier Standaert, and Nicolas Veyrat-Charvillon. How to certify the leakage of a chip? In Nguyen and Oswald [40], pages 459–476.
24. Stefan Dziembowski, Sebastian Faust, Gottfried Herold, Anthony Journault, Daniel Masny, and François-Xavier Standaert. Towards sound fresh re-keying with

hard (physical) learning problems. In Matthew Robshaw and Jonathan Katz, editors, *Advances in Cryptology - CRYPTO 2016 - 36th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2016, Proceedings, Part II*, volume 9815 of *Lecture Notes in Computer Science*, pages 272–301. Springer, 2016.

25. Yunsi Fei, Qiasi Luo, and A. Adam Ding. A statistical model for DPA with novel algorithmic confusion analysis. In Emmanuel Prouff and Patrick Schaumont, editors, *Cryptographic Hardware and Embedded Systems - CHES 2012 - 14th International Workshop, Leuven, Belgium, September 9-12, 2012. Proceedings*, volume 7428 of *Lecture Notes in Computer Science*, pages 233–250. Springer, 2012.
26. Benedikt Gierlichs, Lejla Batina, Pim Tuyls, and Bart Preneel. Mutual information analysis. In Elisabeth Oswald and Pankaj Rohatgi, editors, *Cryptographic Hardware and Embedded Systems - CHES 2008, 10th International Workshop, Washington, D.C., USA, August 10-13, 2008. Proceedings*, volume 5154 of *Lecture Notes in Computer Science*, pages 426–442. Springer, 2008.
27. Benedikt Gierlichs and Axel Y. Poschmann, editors. *Cryptographic Hardware and Embedded Systems - CHES 2016 - 18th International Conference, Santa Barbara, CA, USA, August 17-19, 2016, Proceedings*, volume 9813 of *Lecture Notes in Computer Science*. Springer, 2016.
28. Gilbert Goodwill, Benjamin Jun, Josh Jaffe, and Pankaj Rohatgi. A testing methodology for side channel resistance validation. NIST non-invasive attack testing workshop, 2011. `http://csrc.nist.gov/news_events/non-invasive-attack-testing-workshop/papers/08_Goodwill.pdf`.
29. Dahmun Goudarzi and Matthieu Rivain. How fast can higher-order masking be in software? *IACR Cryptology ePrint Archive*, 2016:264, 2016.
30. Vincent Grosso, Emmanuel Prouff, and François-Xavier Standaert. Efficient masked S-Boxes processing - A step forward -. In David Pointcheval and Damien Vergnaud, editors, *Progress in Cryptology - AFRICACRYPT 2014 - 7th International Conference on Cryptology in Africa, Marrakesh, Morocco, May 28-30, 2014. Proceedings*, volume 8469 of *Lecture Notes in Computer Science*, pages 251–266. Springer, 2014.
31. Vincent Grosso and François-Xavier Standaert. ASCA, SASCA and DPA with enumeration: Which one beats the other and when? In Tetsu Iwata and Jung Hee Cheon, editors, *Advances in Cryptology - ASIACRYPT 2015 - 21st International Conference on the Theory and Application of Cryptology and Information Security, Auckland, New Zealand, November 29 - December 3, 2015, Proceedings, Part II*, volume 9453 of *Lecture Notes in Computer Science*, pages 291–312. Springer, 2015.
32. Yuval Ishai, Amit Sahai, and David Wagner. Private circuits: Securing hardware against probing attacks. In Dan Boneh, editor, *Advances in Cryptology - CRYPTO 2003, 23rd Annual International Cryptology Conference, Santa Barbara, California, USA, August 17-21, 2003, Proceedings*, volume 2729 of *Lecture Notes in Computer Science*, pages 463–481. Springer, 2003.
33. Kerstin Lemke-Rust and Christof Paar. Gaussian mixture models for higher-order side channel analysis. In Pascal Paillier and Ingrid Verbauwhede, editors, *Cryptographic Hardware and Embedded Systems - CHES 2007, 9th International Workshop, Vienna, Austria, September 10-13, 2007, Proceedings*, volume 4727 of *Lecture Notes in Computer Science*, pages 14–27. Springer, 2007.
34. Victor Lomné, Emmanuel Prouff, Matthieu Rivain, Thomas Roche, and Adrian Thillard. How to estimate the success rate of higher-order side-channel attacks. In Batina and Robshaw [6], pages 35–54.

35. Stefan Mangard, Elisabeth Oswald, and François-Xavier Standaert. One for all - all for one: unifying standard differential power analysis attacks. *IET Information Security*, 5(2):100–110, 2011.
36. Stefan Mangard, Thomas Popp, and Berndt M. Gammel. Side-channel leakage of masked CMOS gates. In Alfred Menezes, editor, *Topics in Cryptology - CT-RSA 2005, The Cryptographers' Track at the RSA Conference 2005, San Francisco, CA, USA, February 14-18, 2005, Proceedings*, volume 3376 of *Lecture Notes in Computer Science*, pages 351–365. Springer, 2005.
37. Luke Mather, Elisabeth Oswald, Joe Bandenburg, and Marcin Wójcik. Does my device leak information? an a priori statistical power analysis of leakage detection tests. In Kazue Sako and Palash Sarkar, editors, *Advances in Cryptology - ASIACRYPT 2013 - 19th International Conference on the Theory and Application of Cryptology and Information Security, Bengaluru, India, December 1-5, 2013, Proceedings, Part I*, volume 8269 of *Lecture Notes in Computer Science*, pages 486–505. Springer, 2013.
38. Luke Mather, Elisabeth Oswald, and Carolyn Whitnall. Multi-target DPA attacks: Pushing DPA beyond the limits of a desktop computer. In *Advances in Cryptology - ASIACRYPT 2014 - 20th International Conference on the Theory and Application of Cryptology and Information Security, Kaoshiung, Taiwan, R.O.C., December 7-11, 2014. Proceedings, Part I*, pages 243–261, 2014.
39. Mitsuru Matsui. Linear cryptanalysis method for DES cipher. In Tor Helleseth, editor, *Advances in Cryptology - EUROCRYPT '93, Workshop on the Theory and Application of of Cryptographic Techniques, Lofthus, Norway, May 23-27, 1993, Proceedings*, volume 765 of *Lecture Notes in Computer Science*, pages 386–397. Springer, 1993.
40. Phong Q. Nguyen and Elisabeth Oswald, editors. *Advances in Cryptology - EUROCRYPT 2014 - 33rd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Copenhagen, Denmark, May 11-15, 2014. Proceedings*, volume 8441 of *Lecture Notes in Computer Science*. Springer, 2014.
41. Svetla Nikova, Vincent Rijmen, and Martin Schläffer. Secure hardware implementation of nonlinear functions in the presence of glitches. *J. Cryptology*, 24(2):292–321, 2011.
42. Elisabeth Oswald and Marc Fischlin, editors. *Advances in Cryptology - EUROCRYPT 2015 - 34th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Sofia, Bulgaria, April 26-30, 2015, Proceedings, Part I*, volume 9056 of *Lecture Notes in Computer Science*. Springer, 2015.
43. Romain Poussier, Vincent Grosso, and François-Xavier Standaert. Comparing approaches to rank estimation for side-channel security evaluations. In Naofumi Homma and Marcel Medwed, editors, *Smart Card Research and Advanced Applications - 14th International Conference, CARDIS 2015, Bochum, Germany, November 4-6, 2015. Revised Selected Papers*, volume 9514 of *Lecture Notes in Computer Science*, pages 125–142. Springer, 2015.
44. Emmanuel Prouff. DPA attacks and S-Boxes. In Henri Gilbert and Helena Handschuh, editors, *Fast Software Encryption: 12th International Workshop, FSE 2005, Paris, France, February 21-23, 2005, Revised Selected Papers*, volume 3557 of *Lecture Notes in Computer Science*, pages 424–441. Springer, 2005.
45. Emmanuel Prouff and Matthieu Rivain. Masking against side-channel attacks: A formal security proof. In Thomas Johansson and Phong Q. Nguyen, editors, *Advances in Cryptology - EUROCRYPT 2013, 32nd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Athens, Greece,*

May 26-30, 2013. Proceedings, volume 7881 of *Lecture Notes in Computer Science*, pages 142–159. Springer, 2013.

46. Matthieu Rivain. On the exact success rate of side channel analysis in the Gaussian model. In Roberto Maria Avanzi, Liam Keliher, and Francesco Sica, editors, *Selected Areas in Cryptography, 15th International Workshop, SAC 2008, Sackville, New Brunswick, Canada, August 14-15, Revised Selected Papers*, volume 5381 of *Lecture Notes in Computer Science*, pages 165–183. Springer, 2008.
47. Matthieu Rivain and Emmanuel Prouff. Provably secure higher-order masking of AES. In Stefan Mangard and François-Xavier Standaert, editors, *Cryptographic Hardware and Embedded Systems, CHES 2010, 12th International Workshop, Santa Barbara, CA, USA, August 17-20, 2010. Proceedings*, volume 6225 of *Lecture Notes in Computer Science*, pages 413–427. Springer, 2010.
48. Werner Schindler, Kerstin Lemke, and Christof Paar. A stochastic model for differential side channel cryptanalysis. In Josyula R. Rao and Berk Sunar, editors, *Cryptographic Hardware and Embedded Systems - CHES 2005, 7th International Workshop, Edinburgh, UK, August 29 - September 1, 2005, Proceedings*, volume 3659 of *Lecture Notes in Computer Science*, pages 30–46. Springer, 2005.
49. Tobias Schneider and Amir Moradi. Leakage assessment methodology - extended version. *J. Cryptographic Engineering*, 6(2):85–99, 2016.
50. François-Xavier Standaert, Tal Malkin, and Moti Yung. A unified framework for the analysis of side-channel key recovery attacks. In Antoine Joux, editor, *Advances in Cryptology - EUROCRYPT 2009, 28th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Cologne, Germany, April 26-30, 2009. Proceedings*, volume 5479 of *Lecture Notes in Computer Science*, pages 443–461. Springer, 2009.
51. François-Xavier Standaert, Nicolas Veyrat-Charvillon, Elisabeth Oswald, Benedikt Gierlichs, Marcel Medwed, Markus Kasper, and Stefan Mangard. The world is not enough: Another look on second-order DPA. In Masayuki Abe, editor, *Advances in Cryptology - ASIACRYPT 2010 - 16th International Conference on the Theory and Application of Cryptology and Information Security, Singapore, December 5-9, 2010. Proceedings*, volume 6477 of *Lecture Notes in Computer Science*, pages 112–129. Springer, 2010.
52. Nicolas Veyrat-Charvillon, Benoît Gérard, and François-Xavier Standaert. Soft analytical side-channel attacks. In *Advances in Cryptology - ASIACRYPT 2014 - 20th International Conference on the Theory and Application of Cryptology and Information Security, Kaoshiung, Taiwan, R.O.C., December 7-11, 2014. Proceedings, Part I*, pages 282–296, 2014.

A Algorithms for the masked S-box

Algorithm 1 Secure evaluation of a GF(2)-linear function g.

```
Require: Shares $x_i$ such that $x = x_1 \oplus \cdots \oplus x_d$.
Ensure: Shares $y_i$ such that $\mathsf{g}(x) = y = y_1 \oplus \cdots \oplus y_d$.
for $i$ from 1 to $d$ do
    $y_i \leftarrow \mathsf{g}(x_i)$
end for
```

Algorithm 2 Multiplication of two masked secrets $\in \mathrm{GF}(2^m)$.

```
Require: Shares $x_i$ and $y_i$ such that $x = x_1 \oplus \cdots \oplus x_d$ and $y = y_1 \oplus \cdots \oplus y_d$.
Ensure: Shares $z_i$ such that $x \otimes y = z = z_1 \oplus \cdots \oplus z_d$.
for $i$ from 1 to $d$ do
    for $j$ from $i + 1$ to $d$ do
        $r_{i,j} \overset{\mathbf{r}}{\leftarrow} \mathrm{GF}(2^m)$
        $r_{j,i} \leftarrow (r_{i,j} \oplus x_i \otimes y_j) \oplus x_j \otimes y_i$
    end for
end for
for $i$ from 1 to $d$ do
    $z_i \leftarrow x_i \otimes y_i$
    for $j$ from 1 to $d, j \neq i$ do
        $z_i \leftarrow w_i \oplus r_{i,j}$
    end for
end for
return $(z_1, ..., z_d)$
```

Algorithm 3 Field multiplication of two elements $\in \mathrm{GF}(2^m)$.

```
Require: $x, y \in \mathrm{GF}(2^m)$.
Ensure: $z$ such that $z = x \otimes y$.
$x' \leftarrow \mathsf{LogTab}[x]$
$y' \leftarrow \mathsf{LogTab}[y]$
$z' \leftarrow x' + y' \bmod 2^m - 1$
$z \leftarrow (x \neq 0 \wedge y \neq 0)\ \mathsf{aLogTab}[z']$
return $z$
```

B Additional figures

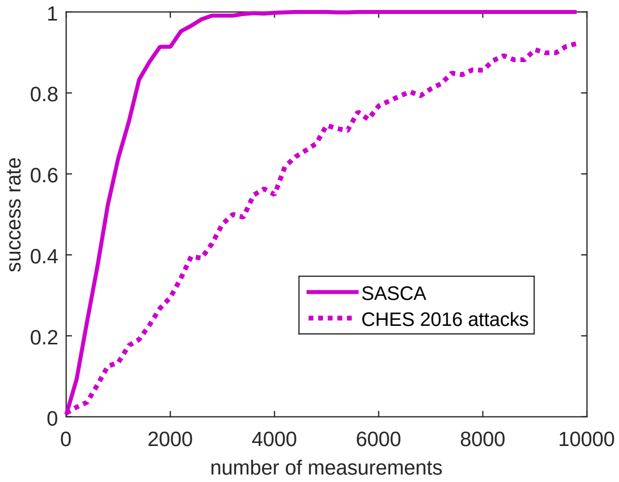

Fig. 12. Efficient heuristic attacks ($d = 3$, $m = 8$, $\sigma_n^2 = 1$).

C Metric-based bound for the key rank

Very summarized, the two core ideas used in [20] to take the computational (enumeration) power of a divide-and-conquer adversary into account in a side-channel evaluation are: (i) to bound the success rate per S-box in function of the adversary's computational power thanks to the mutual information of an aggregated key variable K^c_{agg}, where c is an aggregation parameter (corresponding to the computational power), and (ii) to plug these success rate bounds into the metric-based rank-estimation algorithm of [43]. So technically, the only ingredient needed to exploit the same tools is the mutual information of the aggregated key variable (i.e., the so-called NAMI, for Normalized Aggregated Mutual Information). Unfortunately, the exact computation of the NAMI is impossible in our case, since we do not have access to the probabilities of all the key candidates (that are combined during the aggregation process). So we need a way to bound the NAMI based on its first value $\mathrm{NAMI}(c = 1) = \mathrm{MI}(K; X, \bar{\boldsymbol{L}})$.

For this purpose, a simple observation is that for $c \leq 2^{m-1}$, aggregating $c = 2^q$ key candidates together can at most multiply the NAMI by $q+1$. The behavior of the NAMI for $c > 2^{m-1}$ is less intuitive (since in general, the definition of the NAMI is most intuitive when c is a power of two). Yet, as illustrated by the example in Figure 13, a simple heuristic to bound it is then to connect the value of the NAMI at $c = 2^{m-1}$ and the maximum value of 1 that is reached at $c = 2^m$ by a straight line (which is obviously conservative as well, since the figure is in log-lin scale). Alternatively, when $\mathrm{MI}(K; X, \bar{\boldsymbol{L}}) < \frac{1}{2^m}$, an even simpler bound is to connect $\log(\mathrm{NAMI}(c = 1)) = \log(\mathrm{MI}(K; X, \bar{\boldsymbol{L}}))$ and $\log(\mathrm{NAMI}(c = 2^m)) = 0$ by a straight line. More accurate bounds are certainly reachable, yet not useful here since the general focus of the paper is on providing fast intuitions regarding the computational security of a key manipulated by a leaking device,

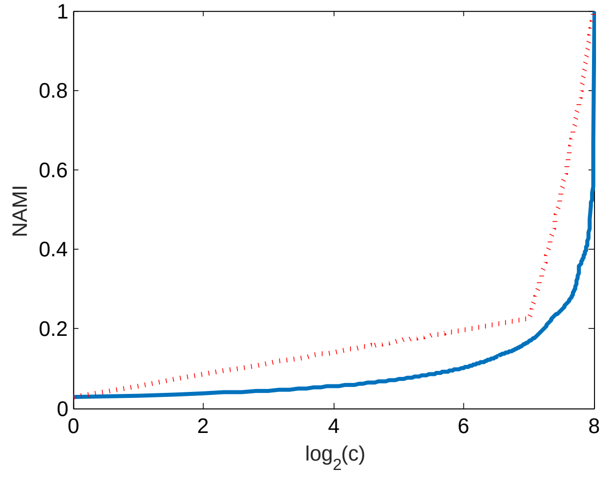

Fig. 13. Bound on the Normalized Aggregated Mutual Information.