Noisy Leakage Revisited

Stefan Dziembowski*[1], Sebastian Faust[2]**, and Maciej Skorski*

[1] Warsaw University
[2] EPFL Lausanne, Switzerland and Ruhr-University Bochum, Germany.
e-mails: stefan@dziembowski.net; sebastian.faust@gmail.com;
m.skorski@mimuw.edu.pl.

Abstract. Physical side-channel leakages are an important threat for cryptographic implementations. One of the most prominent countermeasures against such leakage attacks is the use of a masking scheme. A masking scheme conceals the sensitive information by randomizing intermediate values thereby making the physical leakage independent of the secret. An important practical leakage model to analyze the security of a masking scheme is the so-called noisy leakage model of Prouff and Rivain (Eurocrypt'13). Unfortunately, security proofs in the noisy leakage model require a technically involved information theoretic argument. Very recently, Duc et al. (Eurocrypt'14) showed that security in the probing model of Ishai et al. (Crypto'03) implies security in the noisy leakage model. Unfortunately, the reduction to the probing model is non-tight and requires a rather counter-intuitive growth of the amount of noise, i.e., the Prouff-Rivain bias parameter decreases proportional to the size of the set $\mathcal{X}$ of the elements that are leaking (e.g., if the leaking elements are bytes, then $|\mathcal{X}| = 256$). The main contribution of our work is to eliminate this non-optimality in the reduction by introducing an alternative leakage model, that we call the *average probing model*. We show a tight reduction between the noisy leakage model and the much simpler average random probing model; in fact, we show that these two models are essentially equivalent. We demonstrate the potential of this equivalence by two applications:

- We show security of the additive masking scheme used in many previous works for a constant bias parameter.
- We show that the compiler of Ishai et al. (Crypto'03) is secure in the average probing model (assuming a simple leak free component). This results into security with an *optimal bias parameter* of the noisy leakage for the ISW construction.

1 Introduction

Side-channel attacks break cryptographic implementations by exploiting physical observations of, e.g., the power consumption [18] or running time [17] of a

* Supported by the WELCOME/2010-4/2 grant founded within the framework of the EU Innovative Economy (National Cohesion Strategy) Operational Programme.
** Received funding from the Marie Curie IEF/FP7 project GAPS, grant number: 626467.

cryptographic device. One of the most well-studied and widely used side-channel attacks are power analysis techniques (see, e.g., [18, 13, 3, 20] and many more). In a power analysis attack the adversary exploits the instantaneous power consumption of a physical cryptographic device, e.g., of a smart card, with the goal to extract sensitive information and breaking the cryptographic implementation. One of the most prominent countermeasures against power analysis attacks are masking schemes [2, 13]. The basic idea of a masking scheme is to secretly share all sensitive information, including the secret key and all intermediate values that depend on it, thereby making the leakage independent of the secret data. The most prominent masking scheme is the Boolean masking: a secret bit X is encoded by random bits $(X_1, \ldots, X_n)$ such that $X = X_1 \oplus \ldots \oplus X_n$. It is easy to extend the Boolean masking to work over larger fields $\mathcal{X}$ with $|\mathcal{X}| > 1$. In this case the shares X_i are random elements in $\mathcal{X}$ and $\oplus$ denotes addition in $\mathcal{X}$.

Amplifying noise with masking. As physical measurements are inherently noisy, one main challenge for a side-channel adversary is to isolate the relevant sensitive information from the noise in the measurement. Indeed, an attack is more likely to succeed if the adversary obtains less noisy measurements. Moreover, in practice noise can be relatively easily amplified using practical techniques [3, 5, 20, 2], where one particular example to amplify noise is the masking countermeasure. The fact that masking amplifies noise in measurements was first formally studied in the the pioneering work of Chari et al. [2]. In particular, their main result considers shares X_i of the binary field and shows that if the adversary observes a noisy version $\nu(X_i)$ for each share X_i, he will need an exponential number (in n) of measurements to recover the secret bit X.

Noisy leakage models for masking. The noisy leakage model of Chari et al. assumes a specific noise model, where the noise χ is assumed to be sampled from a Gaussian distribution and the adversary obtains $X + \chi$ as the noisy leakage. The recent work of Prouff and Rivain [25] generalizes the definition of noise by introducing the concept of a noisy leakage function $\nu(.)$. Informally speaking, a function $\nu(.)$ is δ-noisy if the statistical distance between the uniform distribution X and the conditional distribution $X|\nu(X)$ is bounded by some parameter $\delta \in [0, 1]$. To give a better understanding of the Prouff-Rivain noise model, consider the example when δ is close to 0. In this case the function ν is assumed to be very noisy, i.e., the leakage is non-informative as the noise dominates the signal. On the other extreme, when δ is close to 1 then the noisy component of the leakage $\nu(.)$ is close to deterministic.

The way in which Prouff and Rivain model the noisy leakage has two important advantages over the work of Chari et al.: first, the noise is neither assumed to be sampled from some fixed Gaussian distribution nor is it required to be of an additive nature. Instead, in [25] *any* type of noisy leakage is allowed as long as it satisfies the proposed statistical measure. Second, the noisy leakage model of [25] provides a meaningful and natural interpretation of what it means to obtain noisy leakage from values of larger sets (e.g., leakage from a byte).

For instance, $\nu(.)$ may take as input a byte X and first computes the Hamming weight of X before perturbing the result by a noisy component.

While the model of Prouff and Rivain provides a first good approximation of physical side-channel leakage, which is generally applicable in practice, it is very involved to work in. In particular, in [25] the authors prove the security of the masking scheme of [14] against noisy leakage by going through a technical information theoretic argument. This situation is unsatisfying as proving the security of new masking schemes requires to redo the involved analysis of Prouff and Rivain.

Leakage reductions. The recent work of Duc, Dziembowski and Faust [7] reconsiders the notion of noisy leakages. Their main result is a simple reduction from the noisy leakage model to the much simpler and cleaner random probing model. The ϵ-random probing model – first introduced by Ishai et al. [14] – considers only a single simple noisy leakage function φ, where $\varphi(X) = X$ with probability ϵ, and $\perp$ otherwise. Notice that in case when φ outputs $\perp$ the adversary learns nothing about the underlying secret value X. The consequence of this reduction are twofold: first, it significantly simplifies security proofs in the noisy leakage model as one only needs to analyze the security of the masking scheme in the random probing model. Second, Duc et al. [7] show by a simple Chernoff argument that any scheme that is secure in the t-probing model of Ishai, Sahai and Wagner [14] is secure in the random probing model, which by their reduction also implies security in the noisy leakage model of Prouff and Rivain. Recall that in the t-probing model the leakage is bounded to t-bits and hence eventually the noisy leakage is reduced to the much simpler and cleaner *deterministic* bounded leakage model.

While Duc et al. [7] provide a first step towards a better understanding of the noisy leakage model, one main drawback of their analysis is the fact that the reduction between the noisy leakage model and the random probing model is not tight. More precisely, when one extends the Boolean masking to work over larger fields, i.e., when the shares X_i of the encoding are from a larger field $\mathcal{X}$ with $|\mathcal{X}| > 1$, then ϵ-random probing security implies "only" $\delta := \epsilon/|\mathcal{X}|$ noisy resilience in the Prouff-Rivain model. Recall that in the Prouff-Rivain noise model a smaller value for δ results into a weaker result as the leakage is required to be "more noisy". For instance, consider the situation where the shares X_i of the encoding $(X_1, \ldots, X_n)$ are bytes or words as it would be the case on many standard hardware architectures. In this case, as an artefact of the proof the requirement on the noise needs to take into account an additional factor of 2^8 or 2^{32} in order to compensate for the $1/|\mathcal{X}|$ loss.

The main contribution of our work is to eliminate this unnatural loss in the reduction by developing a tight characterization of the noisy leakage model of Prouff and Rivain. Our main new technique to achieve this goal is to show a tight (up to a constant 2) equivalence between the noisy leakage model and a new leakage model that we call the *average probing model*.

We emphasize that the equivalence between these two models allows us to significantly improve the formally verifiable security guarantees of common masking schemes (see below) when the noisy component of the leakage is small. Moreover, our improved reduction is of particular importance for applications that work with fields of super-polynomial size, e.g, when we use blinding in a discrete-log based scheme. In this case, the reduction of Duc et al. looses a factor that is super-polynomial in the security parameter and hence results into meaningless security guarantees due to requiring almost uniform noise.

1.1 Our contribution

The main contribution of our work is to introduce a new noisy leakage model that we name the *ϵ-average probing model* that provides a much tighter (and essentially optimal) equivalence to the leakage model of Prouff and Rivain. Our approach is in spirit of the recent work of Duc et al. [7] who show that t-probing security implies security in the noisy leakage model. In contrast to [7], however, our reduction does not result in the $1/|\mathcal{X}|$ loss that occurs in the reduction of Duc et al. We demonstrate how to use the new leakage model by two applications that result due to the tightness of the reduction to the noisy leakage model to significantly improved security statements compared to earlier works; namely, we show that using the natural noise model of Prouff and Rivain [25] the additive masking is secure with a δ-noise parameter that is independent of the size of the underlying field. As a second important application, we show that masking schemes based on the ISW construction [14] are secure in the average probing model (under the assumption of a leak-free component). Our analysis results in an asymptotically optimal δ-noise parameter for the ISW construction as for asymptotically higher values of δ the ISW construction can be broken. We provide more details on our contributions below. A summary of our contributions are given in Figure 2 and Figure 1.

The ϵ-average probing model. The definition of the *ϵ-average probing model*, described formally in Section 5, can be viewed as a relaxation of the definition of the random probing model of Ishai et al. [14] and Duc et al. [7] . Intuitively, it comes from a different interpretation of an informal statement "probability of $\varphi(x) \neq \perp$ is equal to ϵ" (where φ is the leakage function). Recall that in [7] it was required that it holds for *every* $x \in \mathcal{X}$, and the randomness in the probability came only from the internal random choices[3] made by φ. In contrast, in the ϵ-average probing model we require that $\mathbb{P}(\varphi(X) \neq \perp)$ is equal to ϵ when the probability is taken *also over* X. This seemingly small change has huge implications. In particular, it allows us to show a tight (up to a factor 2) equivalence between our new probing model and the model of [25]. This, in turn, permits us to obtain much better parameters for the security of additive masking. We elaborate on these points below.

[3] In the sequel we will often make this internal randomness of φ explicit.

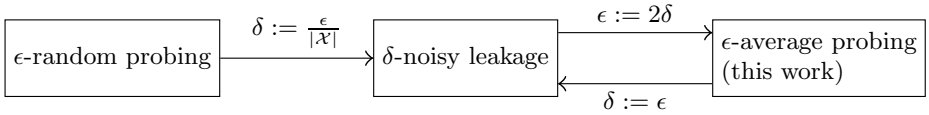

Fig. 1. The figure illustrates the connection between the noisy leakage model of [25], the ϵ-random probing model of [14] and the new average probing model introduced in this work. As shown in the figure, the average probing model is equivalent to the noisy leakage model.

New characterization of noisy leakage functions. We show that the leakage model of Prouff-Rivain is essentially equivalent to the average probing model described above. More concretely, we show (cf. Lemma 7) that every ϵ-noisy adversary can be perfectly simulated by an ϵ-average-probing adversary. We also show the reduction in the other direction (cf. Lemma 8), namely that every ϵ-average-probing adversary can be perfectly simulated by a 2ϵ-noisy adversary. This means that instead of analyzing security against noisy leakage (in the [25] sense) one can use the ϵ-average-probing model. Moreover, we show two important applications of the average probing model that improve the earlier works of [25, 7] when δ is large (i.e., the noise component is rather small compared to the sensitive information). A summary of known reductions between leakage models is given in Figure 1.

Application to masking function. As a first application of our new techniques we show that the additive masking function used in many works [2, 14, 28, 4] is secure for a δ-noise parameter which is independent of the size of the underlying field. Security of the encoding function here means that if the adversary obtains a noisy version $\nu(X_i)$ of each share X_i of an encoding $(X_1, \ldots, X_n)$ he cannot distinguish between an encoding of any two messages. While earlier works showed feasibility with weak bounds [25, 7], i.e., when $\delta < \frac{1}{c|\mathcal{X}|}$ for some constant $c < 1/2$, we are able for the first time to show security of the additive masking function for a constant $\delta < 1/16$ – in particular, δ is independent of $\mathcal{X}$.

Our result also can be viewed as an answer to a question raised in the original work of Chari et al. [2]. In this work (Section 3.7) the authors ask for an extension of the security analysis when leakage is not on bits but from bytes. Unfortunately [2] does not precisely define what "noisy leakage of a byte value" means (e.g., noisy version of each byte share, noisy HW, do we use bit-strings or decimals to represent bytes,...). We believe that a very appealing noise model for bytes is given by the noisy leakage model of Prouff and Rivain [25]. Using the Prouff-Rivain interpretation of a noisy leakage from a byte value, we can provide an answer to the question of Chari et al. [2], namely, we show that security of the encoding can be achieved for a constant δ-noise parameter (which is optimal for the model of [25]). We remark that a constant δ-noise parameter in the model of [25] does not imply that we can show security for a constant noise level in the common leakage model of additive Gaussian noise with a *constant* standard deviation. For instance, if the leakage from the byte is the Hamming weight perturbed by additive Gaussian noise with a constant standard deviation, it is easy

to see that the encoding cannot achieve a strong distance-based security notion when the underlying field size grows. In particular, for the additive Gaussian noise and the Hamming weight leakage function the standard deviation of the noise distribution has to grow at least logarithmically with the size of the field in order to achieve security.

Application to masked computation. While our improvement of the δ-noise parameter for the encoding scheme provides a first indication of the usefulness of the average probing model, we provide a second – practically more relevant – application of the average probing model. Consider the situation where a side-channel adversary attacks a masked implementation of an AES. Of course, in this setting the adversary can target any intermediate value of the computation (e.g., the masked input of an AES S-Box), and hence clearly it does not suffice to only analyze the security of the encoding scheme. Recent works, [25, 7] overcome this restriction and provide the first security analysis of masked computation in the noisy leakage model; in particular, [25, 7] show security of the ISW-construction [14] in the noisy leakage model. Unfortunately, in both cases the requirement on the δ-noise parameter is rather strong: δ decreases linearly with the size of the field $|\mathcal{X}|$ *and* the security parameter n (cf. Figure 2). While the loss in the security parameter is necessary (i.e., one can show easy attacks if the noise is independent of n), there is no fundamental reason why δ has to decrease linearly with the size of the field. We prove that indeed the later loss is unnecessary and show that the ISW construction is secure for noise levels that *only* depend on the necessary factor n – leading to an asymptotically optimal noise rate for the ISW construction. To achieve this goal, we apply the framework of reconstructors introduced by Faust et al. [10]. Quite surprisingly, while at first sight proofs in the average probing model seem more involved as the leakage can implicitly depend on all intermediate values (which is in contrast to the much simpler random probing model), the notion of reconstructors allows for a rather simple security proof. Hence, our analysis of the ISW construction can be seen as a basic *tool box* for proving security of different masking schemes against noisy leakages with tight security bounds. We notice that – similar to the work of Prouff and Rivain [25] – our analysis requires simple leak free components. We leave it as an important direction for future work whether this assumption can be eliminated.

Adaptive noisy leakages. In our proofs we assume that the leakage functions are chosen adaptively, i.e. if the adversary attacks a sequence $X_1, \ldots, X_\ell$ of variables, then his choice of the leakage function φ_i that will be applied to X_i depends on the leakage information that he obtained from $X_1, \ldots, X_{i-1}$. This is in contrast with the proofs in [25, 7]. We believe that in our case assuming the adaptiveness of the adversary makes particular sense, since the adversary in our model has a much bigger choice of leakage functions than in [7] (where his only choice was the ϵ parameter and clearly the best choice for him is to take the maximal ϵ he was allowed to). On a technical level, the only price for going

Author	Proof technique	Noise for Encoding	Noise for any computation	Leak-free gates
Prouff/Rivain [25]	Direct analysis	$O(1/\sqrt{\lvert\mathcal{X}\rvert})$	$O(1/d\lvert\mathcal{X}\rvert)$	yes
Duc et al. [7]	Random probing	$O(1/\lvert\mathcal{X}\rvert)$	$O(1/d\lvert\mathcal{X}\rvert)$	no
Our work	Average probing	$O(1/16)$	$O(1/d)$	yes

Fig. 2. The second column shows the proof technique with which the results are achieved. The third column shows the noise rate that is required for security of encoding. The fourth column shows the noise rate for arbitrary computation. The last column shows under which assumption we can achieve security for arbitrary computation.

to the adaptive case is that instead of relying on the Chernoff's bound we need to use the theory of martingales and the Azuma-Hoeffding inequality.

Additional facts about the [25] leakage model. We also show some simple facts about the [25] leakage model. Although they are not directly needed for our main technical result we believe that they help in understanding this model (this is why we placed it relatively early in the paper, in Section 4), and provide an additional justification why the Prouff-Rivain model is natural. In particular, we show an alternative (but equivalent) definition of the [25] leakage in spirit of the definition of semantic security, and show by a simple hybrid argument how the amount of noise needs to grow when the adversary obtains multiple noisy measurements of the same value X.

1.2 Other related works

Masking & leakage resilient circuits. A large body of work has proposed various masking schemes and studies their security in different security models (see, e.g., [13, 1, 24, 31, 28, 4]). The already mentioned t-probing model has been considered in the work of Rivain and Prouff [28], who show how to extend the work of Ishai et al. to larger fields and propose efficiency improvements. With the emerge of leakage resilient cryptography several works have proposed new security models and alternative masking schemes [10, 29, 15, 11, 8, 12]. The main difference between these security models and the noisy leakage model is that these works typically put a quantitative bound on the amount of leakage – so-called "bounded leakages". While from a theoretical point of view the bounded leakage model offers a beautiful abstraction to analyze the security of cryptographic schemes with weak secrets, it has been questioned [30, 19, 25] whether it models physical leakages in an appropriate way. For instance, a power measurement can typically not be described by a few bits of information but instead requires megabytes if not even gigabytes of information for its description. The noisy leakage model studied in our work more realistically models practical side-channel leakages.

Noisy leakage models. The work of Faust et al. [10] also considers circuit compilers for noisy leakages. Specifically, they propose a construction with security

in the binomial noise model, where each value on a wire is flipped independently with probability $p \in (0, 1/2)$. Besides these works on circuit compilers, several works consider noisy leakages for concrete cryptographic schemes [9, 23, 16]. Typically, the noise model considered in these works is significantly more general than the noise model that is considered for masking schemes. In particular, no strong assumption about the independency of the noise is made.

2 Preliminaries

We start with some standard definitions and lemmas about the statistical distance. If $\mathcal{A}$ is a set then $U \leftarrow \mathcal{A}$ denotes a random variable sampled uniformly from $\mathcal{A}$. Recall that if A and B are random variables over the same set $\mathcal{A}$ then the *statistical distance between A and B* is denoted as $\Delta(A; B)$, and defined as $\Delta(A; B) := \frac{1}{2} \sum_{a \in \mathcal{A}} |\mathbb{P}(A = a) - \mathbb{P}(B = a)|$. It is easy to see that $\Delta(A; B)$ can be also defined in the following alternative ways:

$$\Delta(A; B) = \sum_{a \in \mathcal{A}} \max(0, \mathbb{P}(A = a) - \mathbb{P}(B = a)) \tag{1}$$

$$= 1 - \sum_{a \in \mathcal{A}} \min(\mathbb{P}(A = a), \mathbb{P}(B = a)) \tag{2}$$

$$= \sum_{a : \mathbb{P}(A=a) \geq \mathbb{P}(B=a)} \mathbb{P}(A = a) - \mathbb{P}(B = a). \tag{3}$$

Moreover, Δ satisfies the triangle inequality, i.e. for every A, B and C we have $\Delta(A; B) \leq \Delta(A; C) + \Delta(C; B)$. If $\mathcal{X}, \mathcal{Y}$ are some events then by $\Delta((A|\mathcal{X}) ; (B|\mathcal{Y}))$ we will mean the distance between variables A' and B', distributed according to the conditional distributions $P_{A|\mathcal{X}}$ and $P_{B|\mathcal{Y}}$. If $\mathcal{X}$ is an event of probability 1 then we also write $\Delta(A ; (B|\mathcal{Y}))$ instead of $\Delta((A|\mathcal{X}) ; (B|\mathcal{Y}))$. If C is a random variable then by $\Delta(A ; (B|C))$ we mean $\sum_c \mathbb{P}(C = c) \cdot \Delta(A ; (B|(C = c)))$.

If A, B, and C are random variables and $\mathcal{X}$ is an event then $\Delta((B; C) \mid A)$ denotes $\Delta((BA); (CA))$ (where AB denotes the joint distribution of A and B) and $\Delta((B; C) \mid A, \mathcal{X})$ denotes $\Delta((BA)|\mathcal{X}; (CA)|\mathcal{X})$. It is easy to see that it is equal to $\Delta((B; C) \mid A, \mathcal{X}) = \sum_a \mathbb{P}(A = a|\mathcal{X}) \cdot \Delta((B|A = a, \mathcal{X}) ; (C|A = a, \mathcal{X}))$. If $\Delta(A; B) \leq \epsilon$ then we say that A and B are *ϵ-close*. The "$\stackrel{d}{=}$" symbol denotes the equality of distributions, i.e., $A \stackrel{d}{=} B$ if and only if $\Delta(A; B) = 0$. For A distributed over $\mathcal{A}$ by $d(A)$ we will mean the distance of A from the uniform distribution over $\mathcal{A}$, i.e. $\Delta(A; U)$, where $U \leftarrow \mathcal{A}$. This notation extended to the conditional case in the natural way, e.g. $d(A|\mathcal{X}) := \Delta((A|\mathcal{X}); U)$. The following lemma was proven in [22] (Lemma 1)[4].

Lemma 1 ([22]). *For any two independent random variables A and B over an additive finite group we have $d(A + B) \leq 2d(A)d(B)$*

[4] In [22] it was shown for the quasi-groups, but we do not need this generalization in our paper

It is easy to see that the constant 2 in the lemma above cannot be replaced by a smaller number, at least as long as we quantify over all finite groups. To see why, consider the group $(Z_2^n, \oplus)$, where $n > 1$ and $(x_1, \ldots, x_n) \oplus (y_1, \ldots, y_n) = (x_1 + y_1, \ldots, x_n + y_n)$. Let A and B be uniformly distributed over the set of all elements $x \in Z_2^n$ such that $x_0 = 0$. Then it is easy to verify that $d(A) = d(B) = d(A + B) = 1/2$, and hence $d(A + B)/(d(A)d(B)) = 2$.

The following lemmata are standard information theoretic facts whose proofs are omitted.

Lemma 2. *For any two independent random variables A and B over an additive finite group we have*

$$d(A + B) \leq d(A)$$

Lemma 3. *For any random variables A, B and C that takes values over some set $\mathcal{C}$, and any event $\mathcal{W}$ we have*

$$d(A|B, \mathcal{W}) \leq \sum_{c \in \mathcal{C}} d(A|B, C = c, \mathcal{W}) \cdot \mathbb{P}\left(C = c|\mathcal{W}\right)$$

3 Previous noisy leakage models

In this section we review the most relevant noisy leakage models that have been used to analyze the security of masking schemes. For the lack of space we do not cover several other models used in the literature and refer the reader for some important references to the introduction.

Noisy model of Prouff and Rivain. As discussed in the introduction the noisy model of Prouff and Rivain [25] is a generalization of the model of Chari et al. [2]. In particular, it introduces the notion of a noisy leakage function which is formally defined below.

Definition 1 ([25]). *We say that a function $\nu : \mathcal{X} \times \mathcal{R} \to \mathcal{Y}$ is δ-noisy if*

$$\Delta((\nu(X, R); \nu(X', R)) \mid X) \leq \delta \tag{4}$$

where X and X' are uniform over $\mathcal{X}$ and R is uniform over $\mathcal{R}$.

Some explanations are needed here, since the definition from [25] may appear different from Definition 1 at first sight. First, to make the notation consistent with the rest of this paper, we decided to keep the internal randomness of ν explicit. Secondly, instead of having a bound on "$\Delta((\nu(X, R); \nu(X', R)) \mid X)$" (as in Definition 1)) in the work of [25] the authors impose an upper bound on "$\Delta(X; (X|\nu(X, R)))$" (cf. Eq. (2) in [25]). This is not a problem since as shown by Duc et al. [7] both definitions are equivalent. Finally, the definition in [25] uses as the distance measure the Euclidean norm, while we follow [7] and use the total variation. We refer the reader to [7] for further motivation on this choice and only emphasize here that it corresponds to the maximum distinguishing advantage of

the best possible adversary. This intuitively matches with our understanding of security and is standard in cryptographic research.

Let us now define a notion of an adversary that adaptively attacks a sequence of field elements using the noisy functions. For $\delta \geq 0$ a *δ-noisy adversary on $\mathcal{X}^\ell$* (or *on $\mathcal{X}$* if $\ell = 1$) is a machine $\mathcal{A}$ that, for $i = 1, \ldots, \ell$ plays the following game against an oracle that knows $(x_1, \ldots, x_\ell) \in \mathcal{X}^\ell$.

1. $\mathcal{A}$ specifies a δ_i-noisy function $\nu_i : \mathcal{X} \times \mathcal{R} \to \mathcal{Y}$ such that $\delta_i \leq \delta$.
2. $\mathcal{A}$ receives $\nu_i(x_i, R_i)$, where each R_i is sampled uniformly at random from $\mathcal{R}$.

At the end of the execution $\mathcal{A}$ outputs a value that we denote $out_{\mathcal{A}}(x_1, \ldots, x_\ell)$. We say that $\mathcal{A}$ is *non-adaptive* if he has to specify the functions $\nu_1, \ldots, \nu_\ell$ in advance. If $\mathcal{A}$ works in polynomial time and the noise functions specified by $\mathcal{A}$ are efficiently decidable then we say that $\mathcal{A}$ *is poly-time-noisy* [7].

Random probing model. The following model has been introduced in [14] and used in [7]. Here, we follow the formalism of [7]. We start with the following definition.

Definition 2 ([7]). *A function $\varphi : \mathcal{X} \times \mathcal{R} \to \mathcal{X} \cup \{\perp\}$ is an* ϵ-identity *if for every x and r we have that either $\varphi(x, r) = x$ or $\varphi(x, r) = \perp$ and*

$$\text{for every } x \quad \mathop{\mathbb{P}}_{R \leftarrow \mathcal{R}} (\varphi(x, R) \neq \perp) = \epsilon.$$

For $\epsilon \leq 1$ an *ϵ-probing adversary on $\mathcal{X}^\ell$* (or *on $\mathcal{X}$* if $\ell = 1$) is a machine $\mathcal{A}$ that, for $i = 1, \ldots, \ell$ plays the following game against an oracle that knows $(x_1, \ldots, x_\ell) \in \mathcal{X}^\ell$:

1. $\mathcal{A}$ specifies an ϵ_i-identity function $\varphi_i : \mathcal{X} \times \mathcal{R} \to \mathcal{X} \cup \{\perp\}$ where each $\epsilon_i \leq \epsilon$.
2. $\mathcal{A}$ receives $\varphi_i(x_i, R_i)$, where each R_i is sampled uniformly at random from $\mathcal{R}$.

At the end of the execution $\mathcal{A}$ outputs a value that we denote $out_{\mathcal{A}}(x_1, \ldots, x_\ell)$. We say that $\mathcal{A}$ in *non-adaptive* if he has to specify the functions $\varphi_1, \ldots, \varphi_\ell$ in advance.

4 Useful facts about the Prouff and Rivain noise model

In this section we show some basic facts about the noise model of [25] that, to the best of our knowledge, have not been shown before. We do it because, first of all, we believe that they are of general interests, and may be useful in some future work in the noisy leakage model. Secondly, we think that they may serve as an additional justification why Prouff-Rivain noisy leakage model is natural. This is in particular the case with Lemma 4 below, that essentially provides an alternative and very intuitive interpretation of the [25] noise definition. The proofs are deferred to the full version of this paper.

Lemma 4. *For every δ-noisy function $\nu : \mathcal{X} \times \mathcal{R} \to \mathcal{Y}$ we have*

$$2\delta \geq \Delta((\nu(X_0); \nu(X_1)) \mid X_0, X_1) \geq \delta,$$

where X_0, X_1 are two independent uniform random variables distributed over $\mathcal{X}$.

Let us now argue why this lemma is interesting, by showing a natural interpretation of the "$\Delta((\nu(X_0);\, \nu(X_1)) \mid X_0, X_1)$" formula. To this end, consider the following game played by any adversary $\mathcal{A}$:

1. X_0, X_1 are chosen uniform at random from $\mathcal{X}$ and sent to the adversary,
2. the adversary receives $\nu(X_b)$ for a random $b \leftarrow \{0, 1\}$,
3. the adversary has to guess b (if he does it correctly then we say that he *won the game*).

Note that this game can be essentially summarized as "$\mathcal{A}$ has to distinguish noisy leakage from two random elements X_0 and X_1", and of course it closely resembles a "random message attack" used in defining security of the encryption schemes. Using Lemma 4 it is easy to show the following lemma, which upper bounds the success probability of an adversary in the above game.

Lemma 5. *The probability of any $\mathcal{A}$ winning the game above is upper-bounded by $\Delta((\nu(X_0); \nu(X_1)) \mid X_0, X_1)/2 + 1/2$.*

When considering noisy leakage it is also natural to ask how this notion behaves when the adversary obtains several independent noisy leakage information from the same given element. It turns out that the characterization of noise shown in Lemma 4 is also useful to prove that the success probability of the adversary only increases linearly with the number of measurements. The proof is by a simple hybrid argument.

Lemma 6. *Let $\nu_1, \ldots, \nu_n : \mathcal{X} \to \mathcal{Y}$ be such that for every i and $X_0, X_1 \leftarrow \mathcal{X}$ we have $\Delta((\nu_i(X_0);\, \nu_i(X_1)) \mid X_0, X_1) \leq \delta$. Then*

$$\Delta((\nu_1(X_0), \ldots, \nu_n(X_0)); (\nu_1(X_1), \ldots, \nu_n(X_1)) \mid X_0, X_1) \leq n\delta.$$

5 Epsilon-average probing model

The main contribution of [7] is a reduction from the noisy leakage to the probing model (cf. Lemma 2 of [7]). Although their reduction suffices for improving the results of [25], it suffers from one important weakness which is a significant loss in the error parameter. Namely, in order to "simulate" a δ-noisy function (defined over set $\mathcal{X}$), they need an ϵ-random probing function with $\epsilon = \delta \cdot |\mathcal{X}|$, a consequence of this being that in order to hope for any security one needs to assume that $\delta < 1/|\mathcal{X}|$.

It is relatively straightforward to see that this loss is inherent for this reduction (i.e. Lemma 2 of [7] cannot be improved using better proof techniques). To

see why it is the case, consider the following noisy function (let x_0 be some fixed element of $\mathcal{X}$, and let $\alpha \in [0,1]$):

$$\nu(x) := \begin{cases} x_0 \text{ with probability } \alpha \text{ if } x = x_0 \\ \perp \text{ otherwise.} \end{cases}$$

The following calculation shows that ν defined above is approximately $2\alpha/|\mathcal{X}|$-noisy for large $\mathcal{X}$ (let X, X' and R be uniformly random):

$$\begin{aligned} \Delta((\nu(X);\nu(X')) \mid X) =& \frac{1}{|\mathcal{X}|}\left(\Delta(\nu(x_0);\nu(X')) + \sum_{x \neq x_0} \Delta(\nu(x,R);\nu(X',R))\right) \\ =& \frac{1}{|\mathcal{X}|}\left(\alpha - \frac{\alpha}{|\mathcal{X}|} + \frac{(|\mathcal{X}|-1)\cdot\alpha}{|\mathcal{X}|}\right) \\ =& \frac{2\alpha}{|\mathcal{X}|}\left(1 - \frac{1}{|\mathcal{X}|}\right) \approx 2\alpha/|\mathcal{X}|\,. \end{aligned}$$

On the other hand it is clear that to simulate ν any probing function φ on input x_0 needs to output x_0 with probability at least α. Hence the $|\mathcal{X}|^{-1}$ factor in the security loss is unavoidable.

Our main insight is that this problem can be bypassed by slightly relaxing the definition of the "random probing". Recall that in Definition 2 we had a universal quantifier over all x's from $\mathcal{X}$. In particular, this meant that the probing probability of φ had to depend on the "worst-case" (over all $x \in \mathcal{X}$) behavior of the noisy function ν. This was particularly visible in the example above, where the "worst case" was $x = x_0$ (and the reduction could not take into account that such x occurs with very low probability). Instead, our new definition will look at the *average* $x \in \mathcal{X}$. In other words: it will be possible that φ outputs $\perp$ with a different probability for each x, and the only thing that we will require is that the probability (over both X and R) of receiving $\perp$ is high. A formal definition follows.

Definition 3. *A function $\varphi : \mathcal{X} \times \mathcal{R} \to \mathcal{X} \cup \{\perp\}$ is an* ϵ-average-identity *if for every $x \in \mathcal{X}$ and every $r \in \mathcal{R}$ we have that either $\varphi(x,r) = x$ or $\varphi(x,r) = \perp$ and*

$$\mathop{\mathbb{P}}_{\substack{X \leftarrow \mathcal{X} \\ R \leftarrow \mathcal{R}}} (\varphi(X,R) \neq \perp) = \epsilon. \tag{5}$$

Typically in our applications an adversary will obtain not only $\varphi(X,R)$ but also the randomness R. One way to interpret this situation is as follows: (a) the adversary chooses a set of functions $\{\varphi(\cdot,r) : \mathcal{X} \to \mathcal{X} \cup \{\perp\}\}_{r \in \mathcal{R}}$ (such that (5) holds), then (b) a function $\varphi(\cdot,r)$ is chosen randomly from this set, and finally (c) he learns this function together with $\varphi(x,r)$. Observe that it is enough to restrict ourselves to deterministic functions $\varphi(\cdot,r)$ since anyway a clever adversary will always prefer to make the whole randomness explicit (i.e. to encode it into r), as later he learns it for free.

We will later show (cf. Lemma 7) that the relaxation from Definition 3 allows us to get rid of the $|\mathcal{X}|^{-1}$ factor in the reduction from noisy to probing leakage. Moreover we show that Lemma 7 is essentially optimal, by proving a reduction in the opposite direction (Lemma 8), that looses only factor 2 in the error parameter. Altogether these lemmas provide an alternative but (essentially) equivalent definition of the [25] noise that may be easier to reason about. As an evidence to support this belief we show how Lemma 7 can be used to obtain better error parameters (that do not deepened on $|\mathcal{X}|$) for the additive masking scheme and how it can be used to reason about the ISW masking scheme. This is done in Section 6.

We are now ready to define the ϵ-average probing adversaries (analogously to the ϵ-probing adversaries in Section 3). Let ℓ be some natural parameter and $\mathcal{X}$ be a finite set. For $\epsilon \leq 1$ an *ϵ-average-probing adversary on $\mathcal{X}^\ell$* (or *on $\mathcal{X}$* if $\ell = 1$) is a randomized machine $\mathcal{A}$ that for $i = 1, \ldots, \ell$ plays the following game against an oracle that knows $(x_1, \ldots, x_\ell) \in \mathcal{X}^\ell$:

1. $\mathcal{A}$ specifies an ϵ_i-average-identity function φ_i, where each ϵ_i is at most ϵ.
2. $\mathcal{A}$ receives $(\varphi_i(x_i, R_i), R_i)$, where each R_i is sampled uniformly at random from $\mathcal{R}$.

At the end of the execution $\mathcal{A}$ outputs a value that we denote $out_{\mathcal{A}}(x_1, \ldots, x_\ell)$. We say that $\mathcal{A}$ in *non-adaptive* if he has to specify the functions $\varphi_1, \ldots, \varphi_\ell$ in advance.

5.1 Connection to the noisy leakage

In this section we show a reduction form the δ-noisy model to the δ-average-probing (Lemma 7) and vice versa (Lemma 8), establishing an equivalence between these two models (except of the factor 2 loss in the second reduction). Applications of this equivalence are discussed further in Section 6.

Lemma 7. *For any δ let $\mathcal{A}$ be a δ-noisy adversary on some $\mathcal{X}^\ell$. Then there exists a δ-average-probing adversary $\mathcal{S}$ on $\mathcal{X}^\ell$ such that for every $(x_1, \ldots, x_\ell) \in \mathcal{X}^\ell$ we have*

$$out_{\mathcal{A}}(x_1, \ldots, x_\ell) \stackrel{d}{=} out_{\mathcal{S}}(x_1, \ldots, x_\ell). \tag{6}$$

Moreover if $\mathcal{A}$ is non-adaptive then so is $\mathcal{S}$, and if the noise functions issued by $\mathcal{A}$ are poly-time-decidable then $\mathcal{S}$ works in polynomial time.

Proof. Let the $\nu_1, \ldots, \nu_\ell$ be the functions chosen by $\mathcal{A}$. Each ν_i is δ_i'-noisy (for some $\delta_i' \leq \delta$). Clearly we can assume that $\mathcal{A}$ simply outputs all the values $\nu_1(x_1, R_\nu^1), \ldots, \nu_\ell(x_\ell, R_\nu^\ell)$ that it receives (where the R_i^ν's are uniform over $\mathcal{R}^\nu$ and independent). We construct an adversary $\mathcal{S}$ that for each ν_i chooses a δ_i'-average-identity function φ_i, receives $(\varphi(x_i, R_i^\varphi), R_i^\varphi)$ from the oracle (where R_i^φ is uniform over $\mathcal{R}^\varphi$), and computes a value out_i that is distributed identically to $\nu_i(x_i, R_i^\nu)$. Since these experiments are independent for each i it suffices to consider each i separately. To ease the notation we drop the subscript i in $x_i, \nu_i, R_i^\nu, R_i^\varphi, out_i$ and φ_i. We also assume that $\delta_i = \delta$.

Hence, what we have to show is that for every δ-noisy function $\nu : \mathcal{X} \times \mathcal{R}^\nu \to \mathcal{Y}$ there exists a randomized machine $\mathcal{S}$ that (1) specifies a δ-average-identity function $\varphi : \mathcal{X} \times \mathcal{R}^\varphi \to \mathcal{Y}$, and (2) after receiving $(\varphi(x, R^\varphi), R^\varphi)$ outputs $out_{\mathcal{S}}(x)$ such that for every x we have

$$\nu(x, R^\nu) \stackrel{d}{=} out_S(x), \tag{7}$$

for $R^\nu \leftarrow \mathcal{R}^\nu$ and $R^\varphi \leftarrow \mathcal{R}^\varphi$. We now show how to construct such $\mathcal{S}$. The set $\mathcal{R}^\varphi$ from which the function φ draws its random inputs will be defined as a product of $\mathcal{X}$ and the set $\mathcal{R}^\nu$ of random inputs of ν, i.e: $\mathcal{R}^\varphi := \mathcal{X} \times \mathcal{R}^\nu$. Informally speaking this random input will be used to sample "offline" (i.e. independently of the "real" x) a value y according to the distribution $\nu(X', R^\nu)$ (where $X' \leftarrow \mathcal{X}$ and $R^\nu \leftarrow \mathcal{R}^\nu$).[5] (One can think of such y as a "guess" of the noise value, performed by someone who has no idea about the "real" x.) More precisely the adversary $\mathcal{S}$ constructs the function $\varphi : \mathcal{X} \times \mathcal{R}^\varphi \to \mathcal{X} \cup \{\perp\}$ in the following way. On input $(x, (x', r^\nu))$ the function computes $y = \nu(x', r^\nu)$, and then it outputs[6]

$$\varphi(x, (x', r^\nu)) := \begin{cases} \perp \text{ with probability } \min\left(1, \frac{\mathbb{P}(\nu(x, R^\nu) = y)}{\mathbb{P}(\nu(X, R^\nu) = y)}\right) \\ x \text{ otherwise,} \end{cases} \tag{8}$$

Informally, $w = \perp$ indicates that the function φ (whose input is the "real" x) is happy with the value y that was sampled "off-line". To get some intuitions about (8) consider two extreme cases. First suppose that $\mathbb{P}(\nu(x, R^\nu) = y) \geq \mathbb{P}(\nu(X, R^\nu) = y)$. This means that the value y is at least as likely to happen with the "real" x as it is with a uniformly random X (i.e. when it is sampled "off-line"). Hence intuitively φ is "happy" with this y and wants to communicate to the adversary a message "just output y", which is technically done by outputting $\perp$.

Now, consider the other extreme case, i.e.: $\mathbb{P}(\nu(x, R^\nu) = y) = 0$. Here, in some sense, the value of y is "totally wrong", i.e., it is never going to occur as a noise value for this particular x. Hence the function φ sends a message "wrong y, please resample the noise using x", which is technically done by outputting x. The cases when $0 < \mathbb{P}(\nu(x, R^\nu) = y) < \mathbb{P}(\nu(X, R^\nu) = y)$ are somewhere in between these two extremes and hence φ can either output $\perp$ or x with probability depending on the ratio $\mathbb{P}(\nu(x, R^\nu) = y) / \mathbb{P}(\nu(X, R^\nu) = y)$.

Now, let $(w, (x', r^\nu))$ be the value that $\mathcal{S}$ receives from the oracle. Since $w = \perp$ indicates that y sampled from (x', r^ν) is "correct for the real x" in this

[5] We could also assume that the random input of φ is simply Y that is distributed according to $\nu(X', R^\nu)$, this, however, would lead to more complicated definitions, as in this case we would need to consider randomized functions that take non-uniform random inputs.

[6] A careful reader may notice that φ defined this way is randomized, which seemingly contradicts the definition of the average-identity (where it is required to be deterministic). This is not a problem since we can always extend $\mathcal{R}^\varphi$ to include also the "internal" randomness needed to compute φ. We decided to keep this additional randomness implicit, for the sake of the clarity of the proof (cf. also remarks after Definition 3).

case simply $\mathcal{S}$ outputs y. Otherwise $w = x$. In this case the adversary $\mathcal{S}$ outputs a value z according to the distribution in which every $z \in \mathcal{Y}$ has probability[7]

$$\max\left(0, \frac{\mathbb{P}\left(\nu(x, R^\nu) = z\right) - \mathbb{P}\left(\nu(X, R^\nu) = z\right)}{\Delta(\nu(x, R^\nu); \nu(X, R^\nu))}\right). \tag{9}$$

This distribution is chosen in such a way that it will "compensate" the fact that y's chosen "off-line" have sometimes lower probability than they should have in the "real" distribution.

We first show that $\mathcal{S}$ is δ-average-probing, i.e. that the expected probability of not receiving $\perp$ in the above experiment is equal to δ, more formally:

$$\mathbb{P}\left(\varphi(X, (X', R^\nu)) \neq \perp\right) = \delta \tag{10}$$

(where the variables X, X' and R^ν are uniform and independent). We have

$$\begin{aligned}
&\mathbb{P}\left(\varphi(X, (X', R^\nu)) = \perp\right) \\
&= \sum_{x \in \mathcal{X}} \mathbb{P}\left(X = x\right) \cdot \mathbb{P}\left(\varphi(x, (X', R^\nu)) = \perp\right) \\
&= \sum_{x \in \mathcal{X}} \mathbb{P}\left(X = x\right) \cdot \sum_{y \in \mathcal{Y}} \mathbb{P}\left(\nu(X', R^\nu) = y\right) \mathbb{P}\left(\varphi(x, (X', R^\nu)) = \perp \mid \nu(X', R^\nu) = y\right) \\
&= \sum_{x \in \mathcal{X}} \mathbb{P}\left(X = x\right) \cdot \sum_{y \in \mathcal{Y}} \mathbb{P}\left(\nu(X', R^\nu) = y\right) \cdot \min\left(1, \frac{\mathbb{P}\left(\nu(x, R^\nu) = y\right)}{\mathbb{P}\left(\nu(X', R^\nu) = y\right)}\right) \\
&= \sum_{\substack{x \in \mathcal{X} \\ y \in \mathcal{Y}}} \mathbb{P}\left(X = x\right) \cdot \mathbb{P}\left(\nu(X', R^\nu) = y\right) \cdot \min\left(1, \frac{\mathbb{P}\left(\nu(x, R^\nu) = y\right)}{\mathbb{P}\left(\nu(X', R^\nu) = y\right)}\right) \\
&= \sum_{\substack{x \in \mathcal{X} \\ y \in \mathcal{Y}}} \min\left(\mathbb{P}\left(X = x\right) \cdot \mathbb{P}\left(\nu(X', R^\nu) = y\right) \ , \ \mathbb{P}\left(X = x\right) \cdot \mathbb{P}\left(\nu(x, R^\nu) = y\right)\right) \\
&= \sum_{\substack{x \in \mathcal{X} \\ y \in \mathcal{Y}}} \min\left(\mathbb{P}\left((X, \nu(X', R^\nu)) = (x, y)\right) \ , \ \mathbb{P}\left((X, \nu(X, R^\nu)) = (x, y)\right)\right) \qquad (11) \\
&= 1 - \Delta((X, \nu(X', R^\nu)); (X, \nu(X, R^\nu))) = 1 - \delta \qquad (12)
\end{aligned}$$

where in (11) we used the independence of the variables, and (12) follows from Eq. (2). What remains is to show (7). Take some $x \in \mathcal{X}$ and $y \in \mathcal{Y}$. We have

$$\begin{aligned}
&\mathbb{P}\left(out_{\mathcal{S}}(x) = y\right) = \qquad (13) \\
&\qquad \mathbb{P}\left(\varphi(x, (X', R^\nu)) = \perp \wedge out_{\mathcal{S}}(x) = y\right) + \qquad (14) \\
&\qquad \mathbb{P}\left(\varphi(x, (X', R^\nu)) \neq \perp \wedge out_{\mathcal{S}}(x) = y\right) \qquad (15)
\end{aligned}$$

[7] Eq. (9) defines a probability distribution, since the values in (9) are clearly non-negative and they sum up to 1 (when the sum is computed over all $z \in \mathcal{Y}$), which follows from the fact that $\sum_{z \in \mathcal{Y}} \max(0, \mathbb{P}\left(\nu(x, R^\nu) = z\right) - \mathbb{P}\left(\nu(X, R^\nu) = z\right)) = \Delta(\nu(x, R^\nu); \nu(X, R^\nu))$ (cf. (1)).

It is easy to see that (14) is equal to

$$\begin{aligned}
&\mathbb{P}\left(\varphi(x,(X',R^\nu))=\perp \wedge \nu(X',R^\nu)=y\right)\\
&\mathbb{P}\left(\nu(X',R^\nu)=y\right)\cdot\mathbb{P}\left(\varphi(x,(X',R^\nu))=\perp \mid \nu(X',R^\nu)=y\right)\\
=\ &\mathbb{P}\left(\nu(X',R^\nu)=y\right)\cdot\mathbb{P}\left(\varphi(x,(X',R^\nu))=\perp \mid \nu(X',R^\nu)=y\right)\\
=\ &\mathbb{P}\left(\nu(X',R^\nu)=y\right)\cdot\min\left(1,\frac{\mathbb{P}\left(\nu(x,R^\nu)=y\right)}{\mathbb{P}\left(\nu(X,R^\nu)=y\right)}\right)\\
=\ &\min\left(\mathbb{P}\left(\nu(X',R^\nu)=y\right)\ ,\ \frac{\mathbb{P}\left(\nu(X',R^\nu)=y\right)\cdot\mathbb{P}\left(\nu(x,R^\nu)=y\right)}{\mathbb{P}\left(\nu(X,R^\nu)=y\right)}\right)
\end{aligned}$$

which, since (X,R^ν) and (X',R^ν) have identical distributions is equal to

$$\min\left(\mathbb{P}\left(\nu(X',R^\nu)=y\right),\mathbb{P}\left(\nu(x,R^\nu)=y\right)\right).$$

On the other hand (15) is equal to the product of

$$\mathbb{P}\left(\varphi(x,(X',R^\nu))\neq\perp\right) \tag{16}$$

and

$$\mathbb{P}\left(out_{\mathcal{S}}(x)=y \mid \varphi(x,(X',R^\nu))\neq\perp\right). \tag{17}$$

Clearly (16) is equal to

$$\begin{aligned}
&\sum_{z\in\mathcal{Y}}\mathbb{P}\left(\nu(X',R^\nu)=z\right)\cdot\mathbb{P}\left(\varphi(x,(X',R^\nu))\neq\perp \mid \mathbb{P}\left(\nu(X',R^\nu)=z\right)\right)\\
=\ &\sum_{z\in\mathcal{Y}}\mathbb{P}\left(\nu(X',R^\nu)=z\right)\cdot\left(1-\min\left(1,\frac{\mathbb{P}\left(\nu(x,R^\nu)=z\right)}{\mathbb{P}\left(\nu(X,R^\nu)=z\right)}\right)\right)\\
=\ &\sum_{z\in\mathcal{Y}}\mathbb{P}\left(\nu(X',R^\nu)=z\right)\cdot\max\left(0,\frac{\mathbb{P}\left(\nu(X,R^\nu)=z\right)-\mathbb{P}\left(\nu(x,R^\nu)=z\right)}{\mathbb{P}\left(\nu(X,R^\nu)=z\right)}\right) &(18)\\
=\ &\sum_{z\in\mathcal{Y}}\left(\max\left(0,\mathbb{P}\left(\nu(X,R^\nu)=z\right)-\mathbb{P}\left(\nu(x,R^\nu)=z\right)\right)\right) &(19)\\
=\ &\Delta(\nu(x,R^\nu);\nu(X,R^\nu)), &(20)
\end{aligned}$$

where in (18) we used the fact that for any $0\leq c\leq 1$ we have $1-\min(1,c)=\max(0,1-c)$, in (19) we used $(X,\mathcal{R}^\nu)$ and $(X',\mathcal{R}^\nu)$ are identically distributed, and in (20) we used Eq. (1). In turn, from the construction of $\mathcal{S}$ it is clear that Eq. (17) is equal to

$$\max\left(0,\frac{\mathbb{P}\left(\nu(x,R^\nu)=y\right)-\mathbb{P}\left(\nu(X,R^\nu)=y\right)}{\Delta(\nu(x,R^\nu);\nu(X,R^\nu))}\right). \tag{21}$$

Since (15) is equal to the product of (20) and (21), thus it is equal to

$$\max\left(0,\mathbb{P}\left(\nu(x,R^\nu)=y\right)-\mathbb{P}\left(\nu(X,R^\nu)=y\right)\right),$$

and therefore (13) is equal to

$$\overbrace{\min\left(\mathbb{P}\left(\nu(X',R^\nu)=y\right),\mathbb{P}\left(\nu(x,R^\nu)=y\right)\right)}^{=(14)} \tag{22}$$

$$+\overbrace{\max\left(0,\mathbb{P}\left(\nu(x,R^\nu)=y\right)-\mathbb{P}\left(\nu(X,R^\nu)=y\right)\right)}^{=(15)}$$

$$=\mathbb{P}\left(\nu(x,R^\nu)=y\right), \tag{23}$$

where (23) comes from a simple calculation[8]. In this way we have shown that

$$\mathbb{P}\left(out_{\mathcal{S}}(x)=y\right)=\mathbb{P}\left(\nu(x,R^\nu)=y\right),$$

which implies (7). This finishes the proof of (6). It is also clear from the construction of $\mathcal{S}$ that if $\mathcal{A}$ is non-adaptive then so is $\mathcal{S}$, and that $\mathcal{S}$ works in polynomial time provided the noise functions issued by $\mathcal{A}$ are poly-time-decidable. □

The opposite direction, namely the reduction from the average probing leakage model to the noisy leakage model is given in the lemma below. For space limitations the proof is referred to the full version of this paper.

Lemma 8. *For any ϵ let $\mathcal{A}$ be a ϵ-average-probing adversary on some $\mathcal{X}^\ell$. Then there exists a 2ϵ-noisy adversary $\mathcal{S}$ on $\mathcal{X}^\ell$ such that for a every $(x_1,\ldots,x_\ell)\in\mathcal{X}^\ell$ we have*

$$\mathcal{A}(x_1,\ldots,x_\ell)\stackrel{d}{=}\mathcal{S}(x_1,\ldots,x_\ell). \tag{24}$$

6 Applications of the average probing model

In this section, we present some applications of the average probing model and the reduction to the noisy leakage model of Prouff and Rivain. We first show in Section 6.1 that the standard additive masking function used in numerous works [26, 28, 14, 4] as a building block for masked computation is secure in the ϵ-average probing model. As a second application, we prove in Section 6.2 that the masking scheme of ISW (or rather its extension to larger fields by Rivain and Prouff [28]) is secure in the average probing model using leak-free gates similar to [25, 10]. We emphasize that in both cases we can achieve security with significantly improved δ-noise parameter – in particular, in contrast to earlier works [25, 7] we improve the δ parameter by a factor $|\mathcal{X}|$.

6.1 Security of the additive masking

In this section we show the security of the additive masking scheme over a finite group in the average probing model. Let n be a natural number and $(\mathcal{X},+)$ be

[8] More precisely we use the fact that for every two real numbers A and B we have $\min(A,B)+\max(0,A-B)=A$, with $A:=\mathbb{P}\left(\nu(x,R^\nu)=y\right)$ and $B:=\mathbb{P}\left(\nu(x,R^\nu)=y\right)$.

a finite group. Define an *encoding function* $\mathrm{Enc}^n_{\mathcal{X}} : \mathcal{X} \to \mathcal{X}^n$ and a *decoding function* $\mathrm{Dec}^n_{\mathcal{X}} : \mathcal{X}^n \to \mathcal{X}$ as follows. Let

$$\mathrm{Enc}^n_{\mathcal{X}}(x) = (X_1, \ldots, X_n), \tag{25}$$

where $(X_1, \ldots, X_{n-1}) \leftarrow \mathcal{X}^{n-1}$ and $X_n := x - (X_1 + \cdots + X_{n-1})$ and let $\mathrm{Dec}^n_{\mathcal{X}}(X_1, \ldots, X_n) = X_1 + \cdots + X_n$.

Proof of security. Before we show the security of the encoding scheme, we provide some technical lemmata for the average probing model. The main technical challenge we are facing when applying the average probing model to masking schemes is the fact that average probing leakage reveals non-trivial information about X to an adversary even in the case when $\varphi(X, R) = \perp$, which was not the case in the random probing model. This is because $\varphi(x, R) = \perp$ may be more likely for some x's than for the other (as an example think of φ defined identically to ν in the example at the beginning of Section 5). This technicality makes security proofs in the average probing model more involved than security proofs in the random probing model. Fortunately, in this paper we develop a set of tools that enables us to deal with this technicality of the model. We start by giving some technical lemmata, whose proofs appears in the full version of this paper.

Lemma 9. *Let X and R be random variables with uniform distribution over $\mathcal{X}$ and $\mathcal{R}$, respectively. For any ϵ-average-identity function φ we have*

$$d(X \mid \varphi(X, R) = \perp, R) = \epsilon.$$

The problem with Lemma 9 above is that it only gives information about the expected value (over $r \leftarrow R$) of $d(X \mid \varphi(X, R) = \perp, R = r)$. Hence, for certain r's this value can be very large – or in other words $\varphi(X, R) = \perp$ and $R = r$ can reveal some significant information about X. We deal with this problem by using a Markov-style argument: if the expected value of some term is small, then with good probability this term is small. More precisely, let φ be an ϵ-average-identity function, and for every $\xi \in [0, 1]$ define a function $f : \mathcal{R} \to [0, 1]$ as

$$f(r) := d(X | \varphi(X, r) = \perp, R = r).$$

and let[9]

$$\mathit{Probe}^{\xi}_{\varphi}(x, r) := \begin{cases} ? \text{ if } \varphi(x, r) = \perp \text{ and } f(r) \leq \epsilon/\xi \\ x \text{ otherwise.} \end{cases}$$

In some sense $\mathit{Probe}^{\xi}_{\varphi}$ is more "generous" to the adversary than φ since it outputs x also in cases when φ outputs $\perp$. Clearly $\varphi(x, r)$ can be easily computed from $(\mathit{Probe}^{\xi}_{\varphi}(x, r), r)$, and hence any adversary that learns $(\mathit{Probe}^{\xi}_{\varphi}(X,\ R), R)$ is at least as powerful as an adversary that learns $(\varphi(X, R), R))$. We will use this fact later. First, we show some useful properties of $\mathit{Probe}^{\xi}_{\varphi}(x, r)$.

[9] The "?" symbol is used in a similar way as "$\perp$". We chosen to use "?" in order to avoid confusion in the notation.

Lemma 10. *Let φ be an ϵ-average-identity function. For every $\xi \in [0,1]$ we have that*

$$\underset{r}{\forall}\ d(X|Probe_{\varphi}^{\xi}(X,r) = ?, R = r) \leq \epsilon/\xi \tag{26}$$

and

$$\mathbb{P}\left(Probe_{\varphi}^{\xi}(X,R) \neq ?\right) \leq \xi + \epsilon. \tag{27}$$

The next lemma shows that if we obtain ℓ times the value ? from $Probe_{\varphi}^{\xi}(X_i)$, then the distance of $X_1 + \ldots + X_\ell$ decreases exponentially, and exhibits the first step to show the security of the encoding function.

Lemma 11. *Let $\varphi_1, \ldots, \varphi_\ell$ be ϵ-average-identity functions. Suppose $\mathcal{X}$ is an additive group. Let $(X_1, \ldots, X_\ell) \leftarrow \mathcal{X}^\ell$ and $(R_1, \ldots, R_\ell) \leftarrow \mathcal{R}^\ell$ be uniform and independent random variables and set $X := X_1 + \cdots + X_\ell$. Then for every $(r_1, \ldots, r_\ell)$ we have*

$$d(X \mid \forall_{i=1}^{\ell} Probe_{\varphi}^{\xi}(X_i, r_i) = ?, R_1 = r_1, \ldots, R_\ell = r_\ell) \leq (2\epsilon/\xi)^\ell, \tag{28}$$

and hence

$$d(X \mid \forall_{i=1}^{\ell} Probe_{\varphi}^{\xi}(X_i, R_i) = ?, R_1, \ldots, R_\ell) \leq (2\epsilon/\xi)^\ell. \tag{29}$$

The above already shows that conditioned on the auxiliary information the distance of X from uniform decreases exponentially in ℓ. We start by showing how to translate this into showing security of the encoding scheme $(\mathrm{Enc}_{\mathcal{X}}^n, \mathrm{Dec}_{\mathcal{X}}^n)$, when X is uniform. Later (cf. Corollary 1) we show how to translate this result into one where x is chosen by the adversary, which is the standard indistinguishability-based security definition of leakage resilient encoding schemes. Notice that the lemma below is fully adaptive, i.e., we allow the adversary to obtain $\varphi_i(X_i, R_i)$ and only afterwards he has to decide on which noisy leakage function φ_{i+1} he wants to observe. As such strengthening of the model comes essentially without any additional loss in the parameters (i.e., it comes for free) using the theory of martingales and Azuma inequality, we chose to present the most general version of the fully adaptive adversary below.

Lemma 12. *For every $\epsilon, \lambda, \xi \in [0,1]$ and an ϵ-average-probing adversary $\mathcal{A}$ on $\mathcal{X}^\ell$ and a uniform $X \leftarrow \mathcal{X}$ we have*

$$d(X|out_{\mathcal{A}}(Enc_{\mathcal{X}}^n(X))) \leq (2\epsilon/\xi)^{\lceil (1-\xi-\epsilon-\lambda)n \rceil} + e^{-2\lambda^2 n}. \tag{30}$$

Before we present the proof let us state the basic facts from the theory of martingales (more on this subject can be found, e.g., in [6]). Recall that a sequence $Y_0, Y_1, \ldots$ of random variables is a *submartingale* with respect to a sequence $W_0, W_1, \ldots$ of random variables if every Y_i is a function of $W_0, \ldots, W_{i-1}$ and $\mathbb{E}(Y_i|W_0, \ldots, W_{i-1}) \geq Y_{i-1}$ for every i. The sequence $\{X_i = Y_i - Y_{i-1}\}_{i \geq 1}$ is called a *submartingale difference sequence* (w.r.t. $W_0, W_1, \ldots$). A submartingale $Y_0, Y_1, \ldots$ satisfies the *bounded difference condition with parameters A and B* if for every i it is the case that $X_i \in [A, B]$. We have the following fact (see, e.g., [6], Section 5.3)

Lemma 13 (Azuma-Hoeffding inequality). *Let $Y_0, Y_1, \ldots$ be a submartingale (w.r.t. some other sequence) satisfying the bounded difference condition with parameters A and B. Then for any $t > 0$ we have*

$$\mathbb{P}\left(Y_n < Y_0 - t\right) \leq \exp\left(-\frac{2t^2}{n(B-A)^2}\right).$$

We are now ready for the proof of the lemma.

Proof (of Lemma 12). Let $(X_1, \ldots, X_n) = \mathrm{Enc}^n_{\mathcal{X}}(X)$. Since X is uniform thus $X_1, \ldots, X_n$ are independent. Let $\varphi_1, \ldots, \varphi_n$ be functions specified by $\mathcal{A}$. Since $\mathcal{A}$ is ϵ-average-probing thus each φ_i is an ϵ_i-average identity, where $\epsilon_i \leq \epsilon$. Let, for each i, the function $\mathit{Probe}^{\xi}_{\varphi_i} : \mathcal{X} \times \mathcal{R} \to \mathcal{X} \cup \{?\}$ be defined as above. To simplify notation for each i let $W_i = \mathit{Probe}^{\xi}_{\varphi_i}(X_i, R_i)$ (where $R_i \leftarrow \mathcal{R}$). For each $i = 1, \ldots, n$ define a variable Y_i as

$$Y_i := \begin{cases} 1 \text{ if } W_i = ? \\ 0 \text{ otherwise.} \end{cases}$$

Since the adversary is adaptive, thus his choice of each φ_i can depend on the values $W_1, \ldots, W_{i-1}$. On the other hand, no matter how he behaves, from Lemma 10 we are guaranteed that $\mathbb{P}\left(W_i = ?\right) \geq 1 - \xi - \epsilon$ and hence $\mathbb{E}\left(Y_i | W_1, \ldots, W_{i-1}\right) \geq 1 - \xi - \epsilon$. Define Y'_i as $Y_i - (1 - \xi - \epsilon)$. Obviously then $\mathbb{E}\left(Y'_i | W_1, \ldots, W_{i-1}\right) \geq 0$. Hence $Y'_0, Y'_1, \ldots$ is a submartingale difference sequence w.r.t. $W_0, W_1, \ldots$. Moreover for each i we have

$$-(1 - \xi - \epsilon) \leq Y'_i \leq 1 - (1 - \xi - \epsilon).$$

Hence, if for every $j = 0, \ldots, n$ we let $Z^j := \sum_{i=1}^{j} Y'_i$ then $Z^0, \ldots, Z^n$ is a submartingale[10] w.r.t. $W_0, W_1, \ldots$ satisfying bounded difference condition with parameters $-(1 - \xi - \epsilon)$ and $1 - (1 - \xi - \epsilon)$. Therefore from Azuma-Hoeffding inequality (Lemma 13) we get that

$$\begin{aligned} \mathbb{P}\left(Z^n < -\lambda n\right) &\leq \exp\left(-\frac{2(\lambda n)^2}{n}\right) \\ &= \exp\left(-2\lambda^2 n\right). \end{aligned}$$

Of course $\sum_{i=1}^{n} Y_i = Z^n + n(1 - \xi - \epsilon)$. Therefore

$$\begin{aligned} \exp\left(-2\lambda^2 n\right) &\geq \mathbb{P}\left(\sum_{i=1}^{n} Y_i < -\lambda n + n(1 - \xi - \epsilon)\right) \\ &= \mathbb{P}\left(\sum_{i=1}^{n} Y_i < n(1 - \xi - \epsilon - \lambda)\right) \end{aligned} \tag{31}$$

[10] It is easy to see that if the adversary was non-adaptive then we could also use Chernoff inequality, instead of the Azuma-Hoeffding inequality and martingales.

For every set $\mathcal{I} \subseteq \{1, \ldots, n\}$ such that $|\mathcal{I}| \geq n(1 - \xi - \epsilon - \lambda)$ let $\mathcal{W}^{\mathcal{I}}$ denote the event defined as a following conjunction of events:

$$\mathcal{W}^{\mathcal{I}} := \left(\bigwedge_{j \in \mathcal{I}} \mathit{Probe}^{\xi}_{\varphi_j}(X_j, r_j) = ? \right) \wedge \left(\bigwedge_{j \notin \mathcal{I}} \mathit{Probe}^{\xi}_{\varphi}(X_j, r_j) \neq ? \right)$$

And let: $\mathcal{W} := \bigvee_{\mathcal{I}:|\mathcal{I}| \geq n(1-\xi-\epsilon-\lambda)} \mathcal{W}^{\mathcal{I}}$. From (31) we clearly have

$$\mathbb{P}(\mathcal{W}) \geq 1 - e^{-2\lambda^2 n}. \tag{32}$$

Suppose that $\mathcal{W}^{\mathcal{I}}$ occurred for some $\mathcal{I}$ and let $m = |\mathcal{I}|$. Denote $X^{\mathcal{I}} := X_{i_1} + \cdots + X_{i_m}$. By Lemma 11 we have

$$\begin{aligned}
&(2\epsilon/\xi)^{\lceil n(1-\xi-\epsilon-\lambda)\rceil} \\
&\geq d(X^{\mathcal{I}} | R_{i_1}, \ldots, R_{i_m}, \mathcal{W}^{\mathcal{I}}) \\
&\geq d(X^{\mathcal{I}} | \varphi_{i_1}(X_{i_1}, R_{i_1}), \ldots, \varphi_{i_m}(X_{i_m}, R_{i_m}), R_{i_1}, \ldots, R_{i_m}, \mathcal{W}^{\mathcal{I}}) && (33) \\
&\geq d(X^{\mathcal{I}} | \varphi_1(X_1, R_1), \ldots, \varphi_m(X_n, R_n), R_1, \ldots, R_n, \mathcal{W}^{\mathcal{I}}) && (34) \\
&\geq d(X | \varphi_1(X_1, R_1), \ldots, \varphi_n(X_n, R_n), R_1, \ldots, R_n, \mathcal{W}^{\mathcal{I}}). && (35)
\end{aligned}$$

where (33) comes from the fact that, as observed in Section 5, $\varphi(x, r)$ is a function of $(\mathit{Probe}^{\xi}_{\varphi}(x, r), r)$. Eq. (34) holds because obviously for $i \notin \mathcal{I}$ the value of $(\varphi_i(X_i, R_i), R_i)$ does not bring any additional information about $X^{\mathcal{I}}$. Eq. (35) holds because of Lemma 2 with $A := X^{\mathcal{I}}$ and B equal to the sum of all X_i's with indices not in $\mathcal{I}$. We now have that

$$\begin{aligned}
&d(X | \varphi_1(X_1, R_1), \ldots, \varphi_n(X_n, R_n), R_1, \ldots, R_n, \mathcal{W}) \\
&\leq \sum_{\mathcal{I}:|\mathcal{I}| \geq n(1-\xi-\epsilon-\lambda)} d(X | \varphi_1(X_1, R_1), \ldots, \varphi_n(X_n, R_n), R_1, \ldots, R_n, \mathcal{W}^{\mathcal{I}}) \cdot \mathbb{P}\left(\mathcal{W}^{\mathcal{I}}\right) \\
&\leq (2\epsilon/\xi)^{\lceil n(1-\xi-\epsilon-\lambda)\rceil} \cdot \overbrace{\sum_{\mathcal{I}} \mathbb{P}\left(\mathcal{W}^{\mathcal{I}}\right)}^{\leq 1}, && (36)
\end{aligned}$$

where the first inequality comes from the fact that the events $\mathcal{W}^{\mathcal{I}}$ are pairwise disjoint and hence we can use Lemma 3 (interpreting C as a variable that indicates which $\mathcal{W}^{\mathcal{I}}$ occurred). We therefore obtain that

$$d(X | \mathit{out}_{\mathcal{A}}(\mathrm{Enc}^n_{\mathcal{X}}(X)), \mathcal{W}) \leq (2\epsilon/\xi)^{\lceil n(1-\xi-\epsilon-\lambda)\rceil}. \tag{37}$$

We now have

$$\begin{aligned}
d(X | \mathit{out}_{\mathcal{A}}(\mathrm{Enc}^n_{\mathcal{X}}(X))) &\leq d(X | \mathit{out}_{\mathcal{A}}(\mathrm{Enc}^n_{\mathcal{X}}(X)), \mathcal{W}) + \mathbb{P}(\neg\mathcal{W}) \\
&\leq (2\epsilon/\xi)^{\lceil n(1-\xi-\epsilon-\lambda)\rceil} + e^{-2\lambda^2 n},
\end{aligned}$$

where in the last inequality we used (37) and (32). This finishes the proof. □

Of course, in practice it makes more sense to have the security even if the adversary picks up the encoded element x himself. This is shown in the corollary below. The price is that the error parameter get multiplied by the group size (and a constant). What is important is that this factor simply multiplies the total error, which is much better than in [25, 7], where ϵ was multiplied by $|\mathcal{X}|$. As a consequence, even for very large fields this error can be made negligible by increasing n (which was not the case in [25, 7]). The following is a simple consequence of Lemma 12 (the formal derivation of this corollary appears in the extended version of this paper).

Corollary 1. *For every $\epsilon, \lambda, \xi \in [0,1]$ and an ϵ-average-probing adversary (or equivalently: ϵ-noisy adversary) $\mathcal{A}$ on $\mathcal{X}^\ell$ the information that $\mathcal{A}$ receives about any encoded element x can be "simulated" without access to x, up to a small error. More precisely there exists a random variable Y such that for every $x \in \mathcal{X}$ we have*

$$\Delta(out_{\mathcal{A}}(Enc^n_{\mathcal{X}}(x)) \ ; \ Y) \leq 2\,|\mathcal{X}| \cdot \left((2\epsilon/\xi)^{\lceil (1-\xi-\epsilon-\lambda)(n-1)\rceil} + e^{-2\lambda^2(n-1)}\right) \quad (38)$$

Moreover for any $x_0, x_1 \in \mathcal{X}$ we have

$$\begin{aligned} &\Delta(out_{\mathcal{A}}(Enc^n_{\mathcal{X}}(x_0)) \ ; \ out_{\mathcal{A}}(Enc^n_{\mathcal{X}}(x_1))) \\ &\leq 4\,|\mathcal{X}| \cdot \left((2\epsilon/\xi)^{\lceil (1-\xi-\epsilon-\lambda)(n-1)\rceil} + e^{-2\lambda^2(n-1)}\right), \end{aligned} \quad (39)$$

and in particular (by setting $\xi = \sqrt{\epsilon}$ and $\lambda = 1/2$) we have

$$\begin{aligned} &\Delta(out_{\mathcal{A}}(Enc^n_{\mathcal{X}}(x_0)) \ ; \ out_{\mathcal{A}}(Enc^n_{\mathcal{X}}(x_1))) \\ &\leq 4\,|\mathcal{X}| \cdot \left((4\epsilon)^{\lceil (1/4-\sqrt{\epsilon}/2-\epsilon/2)(n-1)\rceil} + e^{-(n-1)/2}\right). \end{aligned} \quad (40)$$

Moreover fixing $\epsilon = 1/16$ we get that this last term is at most

$$4\,|\mathcal{X}| \cdot \left(e^{-0.13\cdot(n-1)} + e^{-(n-1)/2}\right) \leq 8\,|\mathcal{X}| \cdot e^{-0.13\cdot(n-1)}.$$

From Eq. (39) in the above corollary it is easy to see that with increasing number of shares n and a decreasing ϵ (i.e., more noise) the statistical distance decreases. We notice that the second term of the addition, i.e., $e^{-2\lambda^2(n-1)}$ only gets negligible if n increases, and in particular will dominate the first term when ϵ is negligible. While the same additional error term appeared in the work of Duc et al. [7] (due to the use of a Chernoff bound), the result of Prouff and Rivain [25] did not had this additional error term. We emphasize, however, that this additional error term only becomes relevant when we consider very small values for the δ-bias of the Prouff-Rivain model, i.e., for very noisy leakage functions. In the full version of this paper we show how to eliminate this additional error term using an alternative argument.

Finally, we emphasize that for the noise level in the last part of Corollary 1 ($\epsilon = 1/16$) neither the work of Prouff and Rivain [25], nor the work of Duc et al. [7] gives meaningful bounds unless the field is of a constant size.

6.2 Security of the ISW compiler with leak-free gates

As a second application, we demonstrate that also more complicated masked computation can be proven secure in the average probing model. To this end, we show that the ISW compiler (or rather its extension to larger fields by Prouff and Rivain [25]), which has been widely used as building block for masking schemes [25, 4, 7] is secure in the average probing model assuming leak-free gates. As our reduction from the average probing model to the noisy leakage model of Prouff and Rivain is tight, we improve the noise rate of the work of Prouff and Rivain and Duc et al. [25, 7] significantly – in particular, we are able to eliminate the factor $|\mathcal{X}|$ from the bounds in [7, 25]. We note that compared to the recent work of Duc et al. [7] our analysis of the ISW compiler has one important drawback, namely, that we rely on the assumption that certain parts of the computation are leak-free. We will discuss this assumption in more detail below.

The original circuit Γ. Following the description of [14], we model computation as an arithmetic circuit Γ carrying values from an (arbitrary) finite field $\mathcal{X}$ on their wires and using the following gates to carry out computation in $\mathcal{X}$:

- $+, -$, and $\cdot$, which compute, respectively, the sum, difference, and product in $\mathcal{X}$, of their two inputs,
- the "coin flip" gate coin, which has no inputs and produces a random independently chosen element of $\mathcal{X}$,
- and for every $\alpha \in \mathcal{X}$, the constant gate const_α, which has no inputs and simply outputs α.

Fanout in Γ is handled by a special copy gate that takes as input a single value and outputs two copies. Circuits that only contain the above types of gates are called *stateless*.

Ishai et al. also consider the notion of *stateful circuits*. In addition to the gates described above, stateful circuits also contain memory gates, each of which has a single incoming and a single outgoing wire. Memory gates maintain state: at any round, a memory gate sends its current state down its outgoing wire and updates it according to the value of its incoming wire. The state of all memory gates at clock cycle i is denoted by m_{i-1}, with m_0 denoting the initial state. For instance, the state m_0 of an AES circuit may be its secret key.

The computation of a stateful circuit is performed in several rounds $i = 1, 2, \ldots$. In each of the rounds the circuit will take some public input x, its current internal state m_{i-1} and produces an output y and potentially updates its state to m_i. The evaluation of the circuit proceeds in a straightforward way: when all the input wires of a given gate are known, then the value on the output wire can be computed naturally, i.e., for a multiplication gate with inputs a, b the output wire becomes $c = a \cdot b$. An execution of the circuit Γ with state m_{i-1} on input x is denoted by $(y, m_i) \leftarrow \Gamma(m_{i-1}, x)$. The values that are carried on the wires of the circuit when run on input (m_{i-1}, x) conditioned on the output being (y, m_i) are denoted by the random variable $\mathcal{W}_\Gamma((m_{i-1}, x)|(y, m_i))$.

The protected circuit Γ'. The compiler takes as input the description of the circuit Γ and outputs Γ'. The main building block of Γ' is the encoding scheme $\mathrm{Enc}^n_{\mathcal{X}}$. The initial state m_0 is represented in Γ' in encoded form, i.e., as $M_0 \leftarrow \mathrm{Enc}^n_{\mathcal{X}}(m_0)$. Notice that if m_0 consists of multiple field elements, then we apply the encoding function to each element of m_0 individually. Next, we consider the wires that connect individual gates. In Γ' such wires are represented by *wire bundles* that carry the value of the wire in encoded form. The main difficulty to compile Γ into Γ' is to describe how to transform the gates, i.e., the basic operations described in the last paragraph. For each gate in Γ we have a a sub-circuit – so-called *gadget* – that represents the computation in Γ' and carries out the computation in encoded form. For instance, for a multiplication operation in Γ that takes as input two field elements a, b and outputs $c = a \cdot b$, in Γ' we use a gadget that takes as input two encodings of a (resp.) b and outputs an encoding of c. We emphasize that the computation in the gadgets uses the standard operations defined above and additionally a leak-free gate $\mathcal{O}$. We now provide some details about the most important algorithm of Γ' – the multiplication gadget Mult. The remaining operations, i.e., in particular the addition gadget is done as in the work of Faust et al. (see Figure 3 in [10]) and omitted for space reasons.

The construction of Mult is essentially the construction of Faust et al. [10] from Eurocrypt 2010 (which is essentially the transformation of ISW with leak-free gates) for AC0 leakage functions. In particular, we use their leak-free gate $\mathcal{O}$, which sample from $\mathrm{Enc}^n_{\mathcal{X}}(0)$, i.e., $X \leftarrow \mathcal{O}(1^n)$, where X is a random encoding of 0. We refer to the motivation of this leak-free component to the work of [10] or the work of Prouff and Rivain [25]. The later uses a similar component for their security proof in the noisy leakage model. We only notice that the computation of $\mathcal{O}(1^n)$ can be implemented in a very simple way, namely, sample random field elements $X_1, \ldots, X_{n-1}$ uniformly at random and compute $X_n = -X_1 - \ldots - X_{n-1}$. The output of $\mathcal{O}(1^n)$ is $(X_1, \ldots, X_n)$.

For some finite field $\mathcal{X}$ the multiplication gadget Mult takes as input two vectors $A \leftarrow \mathrm{Enc}^n_{\mathcal{X}}(a)$ and $B \leftarrow \mathrm{Enc}^n_{\mathcal{X}}(b)$, and produces $C \leftarrow \mathrm{Enc}^n_{\mathcal{X}}(c)$, where $c = a \cdot b$. To this end it performs the operations shown in Figure 3. To make the algorithm easier to read, we use small letters to denote elements in $\mathcal{X}$. Vectors over $\mathcal{X}$ will be denoted by capital letters, and matrices are denoted with a "hat" symbol.

The basic property that we require from the protected circuit Γ' is *correctness*. That is, we want that for any input x and any initial state m_0 the circuit Γ and Γ' with initial state $M_0 \leftarrow \mathrm{Enc}^n_{\mathcal{X}}(m_0)$ produce the same output distribution. In addition to correctness, Γ' shall be secure against certain classes of leakages, which we discuss next.

Security definition. Informally, security means that an adversary that obtains leakage from the execution of the protected circuit shall not have any advantage over an adversary that attacks the original circuit with just black-box access. To describe this formally, we use the standard simulation-based paradigm. We start

The multiplication gadget Mult

1. Compute the $n \times n$ matrix $\hat{T} = (a_i \cdot b_j)_{i,j \in [n]}$, where a_i, b_j are the elements of the vector A and B, respectively.
2. Compute the $n \times n$ matrix $\hat{S}$ where the i-th column of $\hat{S}$ is sampled as $S_i \leftarrow \mathcal{O}(1^n)$.
3. Compute $\hat{U} = \hat{T} + \hat{S}$ using matrix addition.
4. Sum the values in each row of $\hat{U}$, i.e., for each $i \in [n]$ compute $q_i = \sum_j u_{i,j}$, where q_i denotes the i-th element of the vector Q.
5. Sample $O \leftarrow \mathcal{O}(1^n)$ and compute the output as $C = Q + O$.

Fig. 3. The multiplication operation takes as input (A, B) and produces the encoding C of ab. The leak-free component $\mathcal{O}(1^n)$ samples from the distribution $\mathrm{Enc}^n_{\mathcal{X}}(0)$ and can be implemented as described in the text above.

by introducing some different types of adversaries. In the following, we assume that the adversary chooses his leakage functions in each round non-adaptively. This can be extended to the adaptive case by making the description of the model more involved and we omit details for space reasons.

A *black-box circuit adversary* $\mathcal{A}$ is a machine that interacts with a circuit Γ via the input and output interface. We denote by $out\left(\mathcal{A} \overset{bb}{\leftrightarrows} \Gamma(m_0)\right)$ the output of $\mathcal{A}$ after interacting with Γ whose initial memory state is m_0. A *δ-noisy circuit adversary* $\mathcal{A}$ is an adversary that has the following additional ability: after each ith round, $\mathcal{A}$ obtains some partial information about the internal state of the computation via the noisy leakage functions. More precisely: let $\mathcal{W}_{\Gamma'}((x, M_{i-1})|(y, M_i))$ be the random variable denoting the values on the wires of $\Gamma'(M_0)$ in the ith round when run on input x and outputting y. Then $\mathcal{A}$ plays the role of a δ-noisy adversary in a game against $\mathcal{W}_{\Gamma'}((x, M_{i-1})|(y, M_i))$ (cf. Section 3), namely: he chooses a sequence $\{\nu_i : \mathcal{X} \times \mathcal{R} \to \mathcal{Y}\}_{i=1}^{\ell}$ of functions such that every ν_i is δ_i-noisy for some $\delta_i \leq \delta$ and he receives $\nu_1(V_1), \ldots, \nu_\ell(V_\ell)$, where V_i denotes a random variable that is part of the wire assignment $\mathcal{W}_{\Gamma'}((x, M_{i-1})|(y, M_i))$. The adversary can repeat this process multiple times for chosen inputs x and we denote the output of $\mathcal{A}$ at the end of this experiment by $out\left(\mathcal{A} \overset{noisy}{\leftrightarrows} \Gamma'(M_0)\right)$.

We can also replace, in the above definition, the "δ-noisy adversary" with the "ϵ-average probing adversary". In this case, after each ith round $\mathcal{A}$ chooses a sequence $(\epsilon_1, \ldots, \epsilon_\ell)$ such that each $\epsilon_i \leq \epsilon$ and he learns $\varphi_1(V_1), \ldots, \varphi_\ell(V_\ell)$, where each φ_i is the ϵ_i-average identity function. Let $out\left(\mathcal{A} \overset{avg}{\leftrightarrows} \Gamma'(M_0)\right)$ denote the output of such $\mathcal{A}$ after interacting with Γ whose initial memory state is M_0. We are now ready to define security of a transformed circuit Γ'.

Definition 4. *Consider a stateful circuit Γ and its transformation Γ' (over some field $\mathcal{X}$) and a randomized encoding function $Enc^n_{\mathcal{X}}$. We say that Γ' is a* (δ, γ)-noise resilient implementation of a circuit Γ w.r.t. $\mathrm{Enc}^n_{\mathcal{X}}$ *if for every δ-noisy circuit adversary $\mathcal{A}$ there exists a black-box circuit adversary $\mathcal{S}$ such that*

for every $m \in \mathcal{X}^\ell$ (for $\ell \in \mathbb{N}$), we have:

$$\Delta\left(\mathit{out}\left(\mathcal{S} \overset{bb}{\leftrightarrows} \Gamma(m)\right)\ ;\ \mathit{out}\left(\mathcal{A} \overset{noisy}{\leftrightarrows} \Gamma'(Enc^n_{\mathcal{X}}(m))\right)\right) \leq \gamma. \tag{41}$$

The definition of Γ' being a (ϵ, γ)-average-probing resilient implementation of a circuit Γ *is identical to the one above, except that we let $\mathcal{A}$ be an ϵ-average-probing circuit adversary $\mathcal{A}$ and Equation 41 is replaced with:*

$$\Delta\left(\mathit{out}\left(\mathcal{S} \overset{bb}{\leftrightarrows} \Gamma(m)\right)\ ;\ \mathit{out}\left(\mathcal{A} \overset{avg}{\leftrightarrows} \Gamma'(Enc^n_{\mathcal{X}}(m))\right)\right) \leq \gamma.$$

In all cases above we will say that Γ' is an implementation of Γ with efficient simulation *if the simulator $\mathcal{S}$ works in time polynomial in $\Gamma' \cdot |\mathcal{X}|$ as long as $\mathcal{A}$ is poly-time and the noise functions specified by $\mathcal{A}$ are efficiently decidable, which will be the case for all our results.*

Security of Γ' against noisy leakages. In contrast to Section 6.1, where we show the security of the additive encoding function in the average probing model, the security analysis of computation is more involved. The reason is that now we have multiple intermediate values that may depend in some predictable way on each other. Intuitively, noise will cancel out the sensitive information in the intermediate values if the sensitive information does not influence too many other intermediate values in the computation, and hence its value is not leaked too many times with independent noise. A similar approach was already exploited in the analysis of Duc et al. [7] – though there the situation was considerably simpler as in the ϵ-probing model the leakage is independent of most of the computation (i.e., large parts of the computation do not leak at all!). In contrast in the average probing model considered in this work, the leakage depends implicitly on *all* intermediate values as even in the case when the leakage function outputs $\perp$ the adversary may learn non-trivial information about the value probed.

To overcome these difficulties we use the framework of *reconstructors* introduced by Faust et al. [10] to argue about the security of masked gadgets. Informally, we give a simulator that just has leakage access to the inputs and outputs of the gadget and from that can simulate the entire leakage from the intermediate values of the gadget. We say that a simulation is good if the simulated leakage is indistinguishable from the real leakage of the intermediate values, when the leakage is assumed to be an ϵ-average probing leakage function. Moreover, we will require the simulator to be from some restricted class of functions. This is important since eventually we want to reduce the security of the protected circuit to the security of the underlying encoding scheme. We here strongly rely on the formalization given in [10] who consider such restricted simulators to achieve security against noisy leakages (albeit in a different noise model).

At a very informal level, we show that the internal values of a gadget can be simulated by a function REC that takes as input X (which is an encoded input of the gadget) and returns two types of values to simulate the internals

of the gadget: (i) either constant values that are independent of the input X, or (ii) values that depend in a very restricted way on REC's input, namely for an input X they have the form $cX + C$, where c and C are constants in $\mathcal{X}$ and $\mathcal{X}^n$ respectively. Now, clearly (i) does not reveal any sensitive information about X (since it is independent of relevant information), and (ii) can essentially be reduced to just (multiple) noisy leakages from the encoding. As the security proof is very similar to [10], and in our work the circuit compiler is merely an application to show how to carry out security proofs of masked computation in the average probing model, we refer the reader to the full version of [10] for further details on the formalization of reconstructors.

To formalize the above informal description of what an admissible simulator REC shall look like, we recall the definition of the function class $\mathsf{LOCAL}(\ell)$ introduced by [10]. Function ins $\mathsf{LOCAL}(\ell)$ depend only in a very restricted way on their inputs, and are hence useful to simulate noisy leakage without revealing too much sensitive information. For some $\ell, n, t, k \in \mathbb{N}$, a function $f : \mathcal{X}^{tn} \to \mathcal{X}^k$ with inputs $X^{(1)}, \ldots, X^{(t)} \in \mathcal{X}^n$ is said to be in $\mathsf{LOCAL}(\ell)$ if the following holds for each $i \in [1, t]$:

> For any fixed $t-1$ inputs $X^{(1)}, \ldots, X^{(i-1)}, X^{(i+1)}, \ldots, X^{(t)}$, all but at most $n\ell$ output values (from $\mathcal{X}$) of the function $f(X^{(1)}, \ldots, X^{(t)})$ (as a function of $X^{(i)}$) are constant (i.e., do not depend on $X^{(i)}$); the remaining outputs are computed as $cX^{(i)} + C$, for some constant $C \in \mathcal{X}^n$ and $c \in \mathcal{X}$.

The identity function, for instance, is in $\mathsf{LOCAL}(1)$, while a function that outputs ℓ copies of its inputs is in $\mathsf{LOCAL}(\ell)$.

We now give a formal definition of efficient simulators (aka reconstructors) tailored to our setting of ϵ average probing leakage functions and for the masked multiplication operation. It is straightforward to generalize the notion to arbitrary masked computation. We then show that the multiplication gadget satisfies the notion. Given that the multiplication gadget is reconstructible, Faust et al. [10] show that security according to Definition 4 can be achieved (cf. Theorem 1 below).

Definition 5 ((ϵ, γ, ℓ)-reconstructors [10]). *Let* Mult *be the masked multiplication with encoded inputs* $X := (A, B)$ *and encoded outputs* $Y := C$*. We say that a pair of strings* (X, Y) *is* plausible for Mult *if* Mult *might output* Y *on input* X*, i.e., if* $\Pr[\mathsf{Mult}(X) = Y] > 0$*.*

Consider a distribution $\mathsf{REC}_{\mathsf{Mult}}$ *over the functions whose input is a plausible pair* (X, Y)*, and whose output is an assignment to the wires of* Mult*. Define* $\mathsf{REC}_{\mathsf{Mult}}(X, Y)$ *as the distribution obtained by sampling a function* R_{Mult} *from* $\mathsf{REC}_{\mathsf{Mult}}$ *and computing* $R_{\mathsf{Mult}}(X, Y)$*. Such a distribution is called a* (ϵ, γ, ℓ)-reconstructor for Mult *if for any plausible* (X, Y) *and any* ϵ*-average probing adversary* $\mathcal{A}$*, the following two distributions are* γ*-close:*

- $out_{\mathcal{A}}(\mathcal{W}_{\mathsf{Mult}}(X|Y))$,
- $out_{\mathcal{A}}(\mathsf{REC}_{\mathsf{Mult}}(X, Y))$.

If the support of the distribution $\mathsf{REC}_{\mathsf{Mult}}$ *is in some set of functions* $\mathsf{LOCAL}(\ell)$*, we say that* Mult *is* (ϵ, γ, ℓ)*-reconstructible.*

Besides the reconstructibility property, we also require that the gadgets of Γ' are *re-randomizing*. We only state it in an informal way here and refer the reader to Definition 3 in [10]. Informally, we say that the masked multiplication operation is re-randomizing if the output of the multiplication is distributed as $\mathrm{Enc}^n_{\mathcal{X}}(c)$ for $c = a \cdot b$ even given the input encoding $A := \mathrm{Enc}^n_{\mathcal{X}}(a)$ and $B := \mathrm{Enc}^n_{\mathcal{X}}(b)$.

It is easy to see that the masked multiplication Mult is re-randomizing. What is more challenging to prove is the fact that Mult is (ϵ, γ, ℓ)-reconstructible, which is shown in the lemma below. The proof of the lemma is very similar to the proof of Lemma 9 in [10], and is deferred to the full version of the paper. To simply notation the lemma below uses the particular parameter setting of Eq. (40) from Corollary 1. It is easy to generalize the lemma for other settings of the parameters.

Lemma 14. *Let n be the security parameter and $\mathcal{X}$ be some finite field. Let ϵ be a function in n defining the noise parameter of the average probing model. The* Mult *operation is $(\epsilon, \gamma, 2n)$-reconstructible for:*

$$\gamma := 4\,|\mathcal{X}|\,n \cdot \left((4(n+1)\epsilon)^{\lceil (1/4 - \sqrt{((n+1)\epsilon}/2 - (n+1)\epsilon/2)(n-1)\rceil} + e^{-(n-1)/2}\right).$$

Given the above lemma we are now ready to apply the framework of Faust et al. [10] and prove that Γ' is secure according to Definition 4. The proof is straightforward and merely puts the different parameters together.

Theorem 1. *Let $n > 1$ be the security parameter. Let Γ be an arbitrary stateful arithmetic circuit over some field $\mathcal{X}$. Let Γ' be the circuit that results from the transformation procedure described above. Let q be the number of observations, then Γ' is a (δ, γ)-noise resilient implementation of Γ (with efficient simulation), where*

$$\gamma = 4\,|\mathcal{X}|\,q\,|\Gamma|\,(n+3) \cdot \left((4(n+1)\delta)^{\lceil (1/4 - \sqrt{((n+1)\delta}/2 - (n+1)\delta/2)(n-1)\rceil} + e^{-(n-1)/2}\right)$$

For concreteness, when we plug-in for $\delta := (24n)^{-1}$ we get for $n > 4$:

$$\gamma := 4\,|\mathcal{X}|\,q\,|\Gamma|\,(n+3) \cdot \exp(-n/12) \tag{42}$$

We notice that the number of measurements/observations (i.e., the number of times the adversary can apply a noisy leakage attack on the implementation Γ') was ignored in the work of [7]. In case we do not consider multiple measurements, we can eliminate the factor q from the above bound. Moreover, if we compare the above concrete bound from Eq. (42) with the bound that was achieved by Duc et al. (see Theorem 1 in [7]), then we see that we improve the noise level not only by a factor $|\mathcal{X}|$ but also the constant is increased from $1/28$ to $1/24$ in our work while achieving (asymptotically for large n) the same bound on the statistical distance.

Acknowledgments. We thank the anonymous reviewers of Eurocrypt 2015 for improving the presentation of our result.

References

1. Johannes Blömer, Jorge Guajardo, and Volker Krummel. Provably Secure Masking of AES. In Helena Handschuh and M. Anwar Hasan, editors, *Selected Areas in Cryptography*, volume 3357 of *Lecture Notes in Computer Science*, pages 69–83. Springer, 2004.
2. Suresh Chari, Charanjit S. Jutla, Josyula R. Rao, and Pankaj Rohatgi. Towards Sound Approaches to Counteract Power-Analysis Attacks. In Michael J. Wiener, editor, *CRYPTO*, volume 1666 of *Lecture Notes in Computer Science*, pages 398–412. Springer, 1999.
3. Christophe Clavier, Jean-Sébastien Coron, and Nora Dabbous. Differential Power Analysis in the Presence of Hardware Countermeasures. In Çetin Kaya Koç and Christof Paar, editors, *CHES*, volume 1965 of *Lecture Notes in Computer Science*, pages 252–263. Springer, 2000.
4. Jean-Sébastien Coron. Higher order masking of look-up tables. In Phong Q. Nguyen and Elisabeth Oswald, editors, *EUROCRYPT*, volume 8441 of *Lecture Notes in Computer Science*, pages 441–458. Springer, 2014.
5. Jean-Sébastien Coron and Ilya Kizhvatov. Analysis and Improvement of the Random Delay Countermeasure of CHES 2009. In Mangard and Standaert [21], pages 95–109.
6. Devdatt P. Dubhashi and Alessandro Panconesi. *Concentration of Measure for the Analysis of Randomized Algorithms.* Cambridge University Press, 2009.
7. Alexandre Duc, Stefan Dziembowski, and Sebastian Faust. Unifying leakage models: From probing attacks to noisy leakage. In PhongQ. Nguyen and Elisabeth Oswald, editors, *Advances in Cryptology - EUROCRYPT 2014*, volume 8441 of *Lecture Notes in Computer Science*, pages 423–440. Springer Berlin Heidelberg, 2014.
8. Stefan Dziembowski and Sebastian Faust. Leakage-Resilient Circuits without Computational Assumptions. In Ronald Cramer, editor, *TCC*, volume 7194 of *Lecture Notes in Computer Science*, pages 230–247. Springer, 2012.
9. Stefan Dziembowski and Krzysztof Pietrzak. Leakage-resilient cryptography in the standard model. *IACR Cryptology ePrint Archive*, 2008:240, 2008.
10. Sebastian Faust, Tal Rabin, Leonid Reyzin, Eran Tromer, and Vinod Vaikuntanathan. Protecting Circuits from Leakage: the Computationally-Bounded and Noisy Cases. In Henri Gilbert, editor, *EUROCRYPT*, volume 6110 of *Lecture Notes in Computer Science*, pages 135–156. Springer, 2010.
11. Shafi Goldwasser and Guy N. Rothblum. Securing computation against continuous leakage. In Rabin [27], pages 59–79.
12. Shafi Goldwasser and Guy N. Rothblum. How to Compute in the Presence of Leakage. In *FOCS*, pages 31–40. IEEE Computer Society, 2012.
13. Louis Goubin and Jacques Patarin. DES and Differential Power Analysis (The "Duplication" Method). In Çetin Kaya Koç and Christof Paar, editors, *CHES*, volume 1717 of *Lecture Notes in Computer Science*, pages 158–172. Springer, 1999.
14. Yuval Ishai, Amit Sahai, and David Wagner. Private Circuits: Securing Hardware against Probing Attacks. In Dan Boneh, editor, *CRYPTO*, volume 2729 of *Lecture Notes in Computer Science*, pages 463–481. Springer, 2003.
15. Ali Juma and Yevgeniy Vahlis. Protecting Cryptographic Keys against Continual Leakage. In Rabin [27], pages 41–58.
16. Jonathan Katz and Vinod Vaikuntanathan. Signature Schemes with Bounded Leakage Resilience. In *ASIACRYPT*, pages 703–720, 2009.

17. Paul C. Kocher. Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems. In *CRYPTO'96*, pages 104–113, 1996.
18. Paul C. Kocher, Joshua Jaffe, and Benjamin Jun. Differential Power Analysis. In *CRYPTO'99*, pages 388–397, 1999.
19. Paul C. Kocher, Joshua Jaffe, Benjamin Jun, and Pankaj Rohatgi. Introduction to differential power analysis. *J. Cryptographic Engineering*, 1(1):5–27, 2011.
20. Stefan Mangard, Elisabeth Oswald, and Thomas Popp. *Power Analysis Attacks: Revealing the Secrets of Smart Cards (Advances in Information Security)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2007.
21. 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*. Springer, 2010.
22. Ueli M. Maurer, Krzysztof Pietrzak, and Renato Renner. Indistinguishability amplification. In Alfred Menezes, editor, *CRYPTO*, volume 4622 of *Lecture Notes in Computer Science*, pages 130–149. Springer, 2007.
23. Moni Naor and Gil Segev. Public-key cryptosystems resilient to key leakage. In *CRYPTO*, pages 18–35, 2009.
24. Elisabeth Oswald, Stefan Mangard, Norbert Pramstaller, and Vincent Rijmen. A Side-Channel Analysis Resistant Description of the AES S-Box. In Henri Gilbert and Helena Handschuh, editors, *FSE*, volume 3557 of *Lecture Notes in Computer Science*, pages 413–423. Springer, 2005.
25. Emmanuel Prouff and Matthieu Rivain. Masking against Side-Channel Attacks: A Formal Security Proof. In Thomas Johansson and Phong Q. Nguyen, editors, *EUROCRYPT*, volume 7881 of *Lecture Notes in Computer Science*, pages 142–159. Springer, 2013.
26. Emmanuel Prouff and Thomas Roche. Higher-Order Glitches Free Implementation of the AES Using Secure Multi-party Computation Protocols. In Bart Preneel and Tsuyoshi Takagi, editors, *CHES*, volume 6917 of *Lecture Notes in Computer Science*, pages 63–78. Springer, 2011.
27. Tal Rabin, editor. *Advances in Cryptology - CRYPTO 2010, 30th Annual Cryptology Conference, Santa Barbara, CA, USA, August 15-19, 2010. Proceedings*, volume 6223 of *Lecture Notes in Computer Science*. Springer, 2010.
28. Matthieu Rivain and Emmanuel Prouff. Provably Secure Higher-Order Masking of AES. In Mangard and Standaert [21], pages 413–427.
29. Guy N. Rothblum. How to Compute under AC0 Leakage without Secure Hardware. In Reihaneh Safavi-Naini and Ran Canetti, editors, *CRYPTO*, volume 7417 of *Lecture Notes in Computer Science*, pages 552–569. Springer, 2012.
30. François-Xavier Standaert, Olivier Pereira, Yu Yu, Moti Yung, and Elisabeth Oswald. Leakage resilient cryptography in practice. In Towards Hardware Intrinsic Security: Foundation and Practice (book chapter), 2010.
31. 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, *ASIACRYPT*, volume 6477 of *Lecture Notes in Computer Science*, pages 112–129. Springer, 2010.