On the Practical Security of a Leakage Resilient Masking Scheme

Emmanuel Prouff[1], Matthieu Rivain[2], and Thomas Roche[1]

[1] ANSSI, 51, Bd de la Tour-Maubourg, 75700 Paris 07 SP, France
firstname.name@ssi.gouv.fr
[2] CryptoExperts, 41, Bd des Capucines, 75002 Paris, France
matthieu.rivain@cryptoexperts.com

Abstract. Implementations of cryptographic algorithms are vulnerable to Side-Channel Analyses extracting information from the device behaviour. When such an attack targets the manipulation of several, say d, intermediate variables then it is said to be a d^{th}-order one. A privileged way to circumvent this type of attacks is to split any key-dependent variable into n shares, with $n > d$, and to adapt the internal processing in order to securely operate on these shares. The latter step is often very tricky and few schemes have been proposed which address this issue in a sound way.

At Asiacrypt 2012, Balasch *et al.* proposed a new scheme based on the inner-product sharing introduced the same year by Dziembowski and Faust at TCC. This scheme is the first one to aim at provable security in two different security models: the continuous bounded-range leakage model and the d^{th}-order side-channel security model (sometimes called d-probing model).

In this paper, we contradict the d^{th}-order security claim by exhibiting some first-order information leakages. Namely, we show that some intermediate variables of the scheme depend on secret information whatever the number of shares. This result is of importance since this kind of flaw is considered as a dead-end point when evaluating the practical security of an implementation. To illustrate the effectiveness of the flaw, we perform an information theoretic evaluation of the first-order leakage and we provide simulation results for a standard side-channel attack against the scheme.

1 Introduction

In the nineties, Kocher *et al.* showed in [13,14] that cryptosystems implemented in embedded devices are vulnerable to a new kind of attacks called *Side-Channel Analysis* (SCA for short). These attacks exploit the fact that the device behaviour (*e.g.* its power consumption) depends on the logical values being processed, which leaks information about the algorithm secret parameter. Since Kocher *et al.*'s original publications, efficient countermeasures have been developed which essentially consist in implementing the algorithms such that no intermediate variable depends on both a public value and a guessable part of the secret. The efforts made by researchers to design efficient countermeasures and

J. Benaloh (Ed.): CT-RSA 2014, LNCS 8366, pp. 169–182, 2014.

advanced side-channel attacks gave rise to a new research area and to a huge number of publications. In particular, the original attack of [14] was refined to exploit the leakage on several intermediate variables simultaneously [17]. The so-called *higher-order* SCA has been widely studied and improved since then, and its practicality has been demonstrated in several papers [15, 18, 25].

To defeat side-channel attacks, the secret sharing techniques (aka *masking*) are today considered as a good way to design effective countermeasures. They can indeed be applied to get implementations with a scalable security, parametrized by the number of shares and some physical properties of the device [5, 20]. The core principle of a *masking scheme* is to split any sensitive variable occurring in the computation into several (say n) shares, and to process elementary operations on them. The scheme must further ensure that each tuple of intermediate results is independent of any secret-dependent value as long as the tuple size is lower than some threshold d. The latter property is usually called *d^{th}-order security property*. The construction of d^{th}-order secure masking schemes is of great interest for the embedded security community and several works have been published to deal with this issue in the particular context of block cipher implementations.

Related Works. The first scheme achieving d^{th}-order security for an arbitrary chosen d has been designed by Ishai, Sahai and Wagner in [11]. The here-called *ISW scheme* consists in masking the Boolean representation of an algorithm which is composed of logical operations NOT and AND. Most subsequent schemes follow the same strategy and essentially reduces the problem of defining a masking scheme for the entire block cipher algorithm to the problem of defining masking schemes for the internal *elementary operations*, often the addition and multiplication over some finite field. The security of the scheme is then proved locally (*i.e.* for every elementary operations) in a first place, and then globally by composing secure elementary computations with *mask-refreshing* steps.

In [23], Rivain and Prouff extend the ISW scheme to efficiently protect an AES computation. The obtained scheme is based on Boolean masking (*i.e.* intermediate variables are shared using the bitwise addition), and it uses an number of $n = d + 1$ shares to achieve the d^{th}-order security property. Subsequent works have been published to extend and improve this scheme [4, 6, 12]. In a recent paper [20], Prouff and Rivain provide an alternative security proof for these kinds of Bollean masking schemes. They consider an adversary who is not limited in the number of intermediate variables that can be observed, but who get some *noisy leakage* on every elementary computation of the algorithm. Provided that the noise amount can be increased (linearly with the masking order d), and that a leak-free mask-refreshing procedure can be used, the authors show that the overall sensitive information leakage can be made negligible with respect to the masking order.

An alternative to the above Boolean masking schemes has further been proposed by Genelle *et al.* [9] to secure an AES computation by mixing additive and multiplicative sharings, and by involving the ISW scheme to secure the conversion between one sharing to another.

In [22], Prouff and Roche propose masking schemes for the addition and multiplication of variables split thanks to Shamir's secret sharing [24]. The proposed

schemes are straightforward applications of those in [2] in the context of secure multi-party computation (MPC for short). The d^{th}-order security property is also directly deduced from the collusion resistance of the secure MPC schemes. It is moreover proved that this security is not impacted by the presence of hardware glitches which are common in CMOS technology [16]. Eventually, the authors of [22] argue that the algebraic complexity of Shamir's sharing compared to the Boolean masking significantly reduces the amount of information leakage. A counterpart of this *masking strength* and of the resistance to glitches is that the complexity of multiplication scheme is $O(n^3)$ which is higher than the $O(n^2)$ complexity for the multiplication in ISW-based masking schemes.

Recently, another approach has been followed by Balasch *et al.*'s [1] to construct a secure higher-order masking scheme. The initial purpose of this scheme is to benefit the complexity advantage of [23] and the security advantages of [22]. Namely, the proposed addition and multiplication schemes have respective complexities $O(n)$ and $O(n^2)$, and enjoy masking strength and resistance to glitches. For such a purpose, the authors use the inner-product secret sharing (IP-sharing for short) introduced by Dziembowski and Faust [7] to construct leakage resilient circuits. The principle of the IP-sharing is to randomly split each intermediate variable V into n shares R_i and n non-zero shares L_i such that

$$V = (L_1 \otimes R_1) \oplus (L_2 \otimes R_2) \oplus \cdots \oplus (L_n \otimes R_n) \ ,$$

where $\oplus$ and $\otimes$ are respectively the addition and multiplication laws over some finite field. In both cases, proofs are given for two different security models: the λ-limited security model (often referred to as the *continuous bounded-range leakage model*) for $n \geqslant 130$ (see Section 4 of [1]), and the d^{th}-order security model, with $d = n-1$, for any $n \geqslant 2$ (see definitions page 8 of [1]). Those two security proofs together with the masking strength and the resistance to glitches make Balasch *et al.* scheme a valuable alternative to previous higher-order masking schemes.

Our Contribution. In this paper, we contradict the d^{th}-order security claim made by Balasch *et al.* for their IP masking scheme. We indeed exhibit a first-order flaw in the addition and mask-refreshing schemes for any chosen sharing order n. This result is of importance since this kind of flaw is considered as a dead-end point when evaluating the practical security of an implementation. Indeed, a first-order attack is much less influenced by the leakage noise than higher-order attacks are. To confirm this, we quantify the amount of leaking information for different *signal-to-noise ratios* (SNRs) and we present simulations demonstrating the practicality of the exhibited attacks when the SNR is reasonably small.

2 Inner Product Masking Scheme

Let us first recall the basic principle of IP masking. In the following, $\mathbb{F}_q$ will denote some field of characteristic 2 (*i.e.* $q = 2^m$ for some $m \geqslant 1$), and let $\oplus$ and $\otimes$ denote respectively the addition and the multiplication over $\mathbb{F}_q$. The inner

product between two vectors $\boldsymbol{X} = (X_1, X_2, \ldots, X_n)$ and $\boldsymbol{Y} = (Y_1, Y_2, \ldots, Y_n)$ from $\mathbb{F}_q^n$ is denoted by:

$$\langle \boldsymbol{X}, \boldsymbol{Y} \rangle = (X_1 \otimes Y_1) \oplus (X_2 \otimes Y_2) \oplus \cdots \oplus (X_n \otimes Y_n) \ .$$

The principle of the IP masking scheme is to manipulate every sensitive variable V as a sharing composed of $2n$ elements, namely the coordinates of two vectors $\boldsymbol{L} = (L_1, L_2, \ldots, L_n)$ and $\boldsymbol{R} = (R_1, R_2, \ldots, R_n)$ such that $V = \langle \boldsymbol{L}, \boldsymbol{R} \rangle$. In order to prevent a direct first-order flaw, the coordinates of $\boldsymbol{L}$ are randomly drawn from $\mathbb{F}_q^* = \mathbb{F}_q \backslash \{0\}$.

To perform computation in the masked domain, the authors of [1] define an addition scheme `IPAdd` and a multiplication scheme `IPMult` to securely process those operations on shared variables. Both schemes are themselves based on two building blocks: the `IPHalfMask` and `IPRefresh` procedures, which are recalled hereafter.[1]

The `IPHalfMask` procedure (Algorithm 1) takes a variable $V \in \mathbb{F}_q$ and a half sharing $\boldsymbol{L} \in (\mathbb{F}_q^*)^n$ and it outputs random half sharing $\boldsymbol{R} \in \mathbb{F}_q^n$ satisfying $V = \langle \boldsymbol{L}, \boldsymbol{R} \rangle$.

Algorithm 1. Half-Masking a variable: $(\boldsymbol{L}, \boldsymbol{R}) \leftarrow$ `IPHalfMask`$(V, \boldsymbol{L})$
INPUT: a variable $V \in \mathbb{F}_q$ and a vector $\boldsymbol{L}$ of non-zero shares
OUTPUT: a sharing $\boldsymbol{R}$ such that $V = \langle \boldsymbol{L}, \boldsymbol{R} \rangle$

1. **for** $i = 2$ **to** n **do** $R_i \leftarrow$ `rand()`
2. $R_1 \leftarrow (V \oplus \bigoplus_{i=2}^{n} L_i \otimes R_i) \otimes L_1^{-1}$
3. **return** $\boldsymbol{R}$

Remark 1. As it can be seen in Algorithm 1, the half-sharing $\boldsymbol{R}$ statistically depends on $\boldsymbol{V}$. This explains why the security order of the masking is upper bounded by n (the number of shares R_i). In Section 4, the amount of information leaking through the manipulation of the shares R_i will be compared to the flaw exhibited in this paper.

The `IPRefresh` procedure (Algorithm 2), takes a sharing $(\boldsymbol{L}, \boldsymbol{R})$ and computes a new fresh sharing $(\boldsymbol{L}', \boldsymbol{R}')$ such that $\langle \boldsymbol{L}', \boldsymbol{R}' \rangle = \langle \boldsymbol{L}, \boldsymbol{R} \rangle$.

Algorithm 2. Refresh Vector: $(\boldsymbol{L}', \boldsymbol{R}') \leftarrow$ `IPRefresh`$(\boldsymbol{L}, \boldsymbol{R})$
INPUT: a sharing $(\boldsymbol{L}, \boldsymbol{R})$ of V
OUTPUT: New sharing $(\boldsymbol{L}', \boldsymbol{R}')$ such that $\langle \boldsymbol{L}, \boldsymbol{R} \rangle = \langle \boldsymbol{L}', \boldsymbol{R}' \rangle$

1. $\boldsymbol{L}' \leftarrow$ (`randNonZero()`$)^n$
2. **for** $i = 1$ **to** n **do** $A_i \leftarrow L_i \oplus L'_i$ $\quad [\boldsymbol{A} \leftarrow \boldsymbol{L} \oplus \boldsymbol{L}']$
3. $X \leftarrow \langle \boldsymbol{A}, \boldsymbol{R} \rangle$
4. $\boldsymbol{B} \leftarrow$ `IPHalfMask`$(X, \boldsymbol{L}')$
5. $\boldsymbol{R}' \leftarrow \boldsymbol{R} \oplus \boldsymbol{B}$
6. **return** $(\boldsymbol{L}', \boldsymbol{R}')$

[1] We do not use the algorithmic presentation from [1] involving two different processors as it is useless for the analysis of the d^{th}-order security model.

Remark 2. In Algorithm 2, the Steps (1-2) for generating $\boldsymbol{A}$ does not correspond to what is described in [1]. We chose this algorithm for simplicity and because it has no incidence whatsoever on the following.

We now recall the masked addition `IPAdd` and the masked multiplication `IPMult` in the two following algorithms.

Algorithm 3. Masked Addition: $(\boldsymbol{X}, \boldsymbol{Y}) \leftarrow \texttt{IPAdd}((\boldsymbol{L}, \boldsymbol{R}), (\boldsymbol{K}, \boldsymbol{Q}))$
INPUT: Two sharings $(\boldsymbol{L}, \boldsymbol{R})$ and $(\boldsymbol{K}, \boldsymbol{Q})$ of V and V' respectively
OUTPUT: New sharing $(\boldsymbol{X}, \boldsymbol{Y})$ such that $\langle \boldsymbol{X}, \boldsymbol{Y} \rangle = V \oplus V'$

1. $(\boldsymbol{A}, \boldsymbol{B}) \leftarrow \texttt{IPRefresh}(\boldsymbol{K}, \boldsymbol{Q} \oplus \boldsymbol{R})$
2. $(\boldsymbol{C}, \boldsymbol{D}) \leftarrow \texttt{IPRefresh}(\boldsymbol{L} \oplus \boldsymbol{K}, \boldsymbol{R})$
3. $Z \leftarrow \langle \boldsymbol{C}, \boldsymbol{D} \rangle$
4. $\boldsymbol{Y} \leftarrow \texttt{IPHalfMask}(Z, \boldsymbol{A})$
5. $\boldsymbol{X} \leftarrow \boldsymbol{A}$
6. $\boldsymbol{Y} \leftarrow \boldsymbol{Y} \oplus \boldsymbol{B}$
7. **return** $(\boldsymbol{X}, \boldsymbol{Y})$

Algorithm 4. Masked Multiplication: $(\boldsymbol{X}, \boldsymbol{Y}) \leftarrow \texttt{IPMult}((\boldsymbol{L}, \boldsymbol{R}), (\boldsymbol{K}, \boldsymbol{Q}))$
INPUT: Two sharings $(\boldsymbol{L}, \boldsymbol{R})$ and $(\boldsymbol{K}, \boldsymbol{Q})$ of V and V' respectively
OUTPUT: New sharing $(\boldsymbol{X}, \boldsymbol{Y})$ such that $\langle \boldsymbol{X}, \boldsymbol{Y} \rangle = V \otimes V'$

1. **for** $i = 0$ **to** $n - 1$ **do**
2. **for** $j = 1$ **to** n **do**
3. $\tilde{U}_{i*n+j} \leftarrow L_{i+1} \otimes K_j$
4. $\tilde{V}_{i*n+j} \leftarrow R_{i+1} \otimes Q_j$
5. **end for**
6. **end for**
7. $(\boldsymbol{U}, \boldsymbol{V}) \leftarrow \texttt{IPRefresh}(\tilde{\boldsymbol{U}}, \tilde{\boldsymbol{V}})$
8. $\boldsymbol{A} \leftarrow (U_1, \cdots, U_n)$; $\boldsymbol{C} \leftarrow (U_{n+1}, \cdots, U_{n^2})$
9. $\boldsymbol{B} \leftarrow (V_1, \cdots, V_n)$; $\boldsymbol{D} \leftarrow (V_{n+1}, \cdots, V_{n^2})$
10. $Z \leftarrow \langle \boldsymbol{C}, \boldsymbol{D} \rangle$
11. $\boldsymbol{Y} \leftarrow \texttt{IPHalfMask}(Z, \boldsymbol{A})$
12. $\boldsymbol{X} \leftarrow \boldsymbol{A}$
13. $\boldsymbol{Y} \leftarrow \boldsymbol{Y} \oplus \boldsymbol{B}$
14. **return** $(\boldsymbol{X}, \boldsymbol{Y})$

3 A First-Order Flaw

Balasch *et al.* claim that their IP masking scheme is secure against any side-channel attack of order $d = n - 1$, or equivalently, that any family of $n - 1$

intermediate variables is independent of any sensitive variable. We contradict this claim hereafter by showing that for any fixed parameter n, there always exists a first-order side-channel attack on the IP masking scheme. To this end, we exhibit an intermediate variable that is statistically dependent on some sensitive variable in both the `IPRefresh` and `IPAdd` procedures (Algorithms 2 and 3, Section 2).

3.1 Core Idea of the Attack

For the sake of clarity, we start by developing the core idea of our attack in the `IPRefresh` setting. Then, we show that a similar flaw occurs in the `IPAdd` scheme.

Flaw in Mask-Refreshing Procedure. The `IPRefresh` procedure takes an IP masking $(\boldsymbol{L}, \boldsymbol{R})$ of V and returns a fresh masking $(\boldsymbol{L'}, \boldsymbol{R'})$ of it. The first steps of the procedure generate a random vector $\boldsymbol{A} \in \mathbb{F}_q^n$ whose coordinates are all different from the corresponding ones in $\boldsymbol{L}$ (as $A_i = L_i \oplus L'_i$ and $L'_i \neq 0$ for every i). The next steps compute $X = \langle \boldsymbol{A}, \boldsymbol{R} \rangle$ that is $X = \langle \boldsymbol{L} \oplus \boldsymbol{L'}, \boldsymbol{R} \rangle$ where $\boldsymbol{L}$ and $\boldsymbol{L'}$ are mutually independent and both uniformly distributed over $(\mathbb{F}_q^*)^n$. The first-order flaw exhibited in this paper comes from the manipulation of this variable X. Indeed, we will prove in the following sections that this variable statistically depends on V, which implies that its manipulation leaks information on V contrary to what is claimed in [1]. Our dependency proof will consist in showing that the probability mass functions (pmf) $\Pr[X \mid V = v]$ differ according to v. Thanks to the following lemma, the study of the latter functions is reduced to the study of a simpler function f_n.

Lemma 1. *Let $\boldsymbol{L}$, $\boldsymbol{L'}$ and $\boldsymbol{R}$ be three mutually independent random variables such that $\boldsymbol{L}$ and $\boldsymbol{L'}$ are uniformly distributed over $(\mathbb{F}_q^*)^n$ and $\boldsymbol{R}$ is uniformly distributed over $\mathbb{F}_q^n$. Let X and V respectively denote the result of the inner products $\langle \boldsymbol{L} \oplus \boldsymbol{L'}, \boldsymbol{R} \rangle$ and $\langle \boldsymbol{L}, \boldsymbol{R} \rangle$. Then, for any $(x, v) \in \mathbb{F}_q^2$, the probability $\Pr[X = x \mid V = v]$ satisfies:*

$$\Pr[X = x \mid V = v] = \frac{f_n(v, x \oplus v)}{\Pr[V = v]} \ , \tag{1}$$

where f_n is defined for every $(a, b) \in \mathbb{F}_q^2$ by:

$$f_n(a, b) = \Pr[\langle \boldsymbol{L}, \boldsymbol{R} \rangle = a \wedge \langle \boldsymbol{L'}, \boldsymbol{R} \rangle = b] \ . \tag{2}$$

Proof. By definition of a conditional probability, we have:

$$\Pr[X = x \mid V = v] = \frac{\Pr[V = v \wedge X = x]}{\Pr[V = v]} = \frac{\Pr[V = v \wedge X \oplus V = x \oplus v]}{\Pr[V = v]} \ .$$

Then, from $\Pr[V = v \wedge X \oplus V = x \oplus v] = \Pr[\langle \boldsymbol{L}, \boldsymbol{R} \rangle = v \wedge \langle \boldsymbol{L'}, \boldsymbol{R} \rangle = x \oplus v] = f_n(v, x \oplus v)$ we get (1). □

Flaw in the Addition Procedure. The `IPAdd` procedure is subject to a similar flaw that `IPRefresh`. Indeed at Step 3 of Algorithm 3, a variable $Z = \langle \boldsymbol{C}, \boldsymbol{D} \rangle = \langle \boldsymbol{L} \oplus \boldsymbol{K}, \boldsymbol{R} \rangle$ is computed, where $\boldsymbol{L}$ and $\boldsymbol{K}$ are mutually independent and both uniformly distributed over $(\mathbb{F}_q^*)^n$. Therefore, Lemma 1 applies directly (just by replacing the notation $\boldsymbol{L'}$ by $\boldsymbol{K}$) and we get:

$$\Pr[Z = z \mid V = v] = \frac{f_n(v, z \oplus v)}{\Pr[V = v]} \ . \tag{3}$$

Hence, for the addition procedure, proving that Z leaks information on V reduces to prove that f_n is not constant with respect to $v \in \mathbb{F}_q$ (as for `IPRefresh`).

The purpose of the next section is to study the function f_n defined in Lemma 1 and to explicit its expression. In Section 3.3 those expressions will be evaluated to quantify the information flaw.

3.2 Study of f_n

The study of f_n developed in this section is recursive. First, in Lemma 2, we give an explicit expression for f_1. Then, in Lemma 3, we exhibit a recursive relationship for f_n. Both lemmas are eventually involved to provide an explicit expression of f_n (Theorem 1).

Lemma 2. *The function f_1 satisfies*

$$f_1(a,b) = \begin{cases} \frac{1}{q} & \text{if } (a,b) = (0,0) \\ 0 & \text{if } (a,b) \in (\{0\} \times \mathbb{F}_q^*) \cup (\mathbb{F}_q^* \times \{0\}) \\ \frac{1}{q(q-1)} & \text{if } (a,b) \in \mathbb{F}_q^* \times \mathbb{F}_q^* \end{cases}$$

Proof. When n equals 1, vectors $\boldsymbol{A}$ and $\boldsymbol{B}$ are respectively reduced to a single coordinate A_1 and B_1. Since those coordinates are non-zero by definition, f_1 satisfies:

$$f_1(0,0) \ = \ \Pr[A_1 \otimes R_1 = 0 \wedge B_1 \otimes R_1 = 0] \ = \ \Pr[R_1 = 0] \ = \ \frac{1}{q} \ .$$

Moreover, for any $a \neq 0$, we have

$$f_1(a,0) \ = \ \Pr[R_1 = a \otimes A_1^{-1} \wedge R_1 = 0] \ = \ 0 \ ,$$

which, by symmetry of f_n, also implies $f_1(0,b) = 0$ for any $b \neq 0$.
Eventually, the law of total probability together with the mutual independence between A_1, B_1 and R_1, imply

$$f_1(a,b) \ = \ \sum_{a_1 \in \mathbb{F}_q^*} \Pr[A_1 = a_1] \times \Pr[R_1 = a \otimes a_1^{-1} \wedge B_1 \otimes R_1 = b] \ ,$$

which gives for $a \neq 0$ and $b \neq 0$:

$$\begin{aligned} f_1(a,b) &= \sum_{a_1 \in \mathbb{F}_q^*} \Pr[A_1 = a_1] \times \Pr[R_1 = a \otimes a_1^{-1} \wedge B_1 = b \otimes a^{-1} \otimes a_1] \\ &= \frac{1}{q(q-1)} \ . \end{aligned}$$

□

Lemma 3. *For every $n \geqslant 1$, there exist real values f_n^{00}, f_n^{01} and f_n^{11} such that*

$$f_n(a,b) = \begin{cases} f_n^{00} & \text{if } (a,b) = (0,0) \\ f_n^{01} & \text{if } (a,b) \in (\{0\} \times \mathbb{F}_q^*) \cup (\mathbb{F}_q^* \times \{0\}) \\ f_n^{11} & \text{if } (a,b) \in \mathbb{F}_q^* \times \mathbb{F}_q^* \end{cases} .$$

Moreover, we have

$$\begin{aligned} f_{n+1}^{00} &= \frac{1}{q} f_n^{00} + \frac{q-1}{q} f_n^{11} \ , \\ f_{n+1}^{01} &= \frac{2}{q} f_n^{01} + \frac{q-2}{q} f_n^{11} \ , \\ f_{n+1}^{11} &= \frac{1}{q(q-1)} f_n^{00} + \frac{2(q-2)}{q(q-1)} f_n^{01} + \frac{(q-1)+(q-2)^2}{q(q-1)} f_n^{11} \ . \end{aligned}$$

Proof. The first statement is true for $n = 1$ by Lemma 2. It is then implied by recurrence from the second statement. Therefore, we only need to show the latter statement.

For every $n > 1$, the total probability law implies

$$f_{n+1}(a,b) = \sum_{(a_0,b_0) \in \mathbb{F}_q^2} f_n(a \oplus a_0, b \oplus b_0) f_1(a_0,b_0) \ . \tag{4}$$

1. For $(a,b) = (0,0)$, the terms in the sum in (4) equal $T(a_0,b_0) = f_n(a_0,b_0)f_1(a_0,b_0)$. Moreover, by Lemma 2, the latter product satisfies:

$$T(a_0,b_0) = \begin{cases} \frac{1}{q} f_n(0,0) & \text{if } (a_0,b_0) = (0,0) \\ 0 & \text{if } (a_0,b_0) \in (\{0\} \times \mathbb{F}_q^*) \cup (\mathbb{F}_q^* \times \{0\}) \\ \frac{1}{q(q-1)} f_n(a_0,b_0) & \text{if } (a_0,b_0) \in \mathbb{F}_q^* \times \mathbb{F}_q^* \end{cases} .$$

   We deduce

$$f_{n+1}(a,b) = \frac{1}{q} f_n^{00} + (q-1)^2 \frac{1}{q(q-1)} f_n^{11} \ . \tag{5}$$

2. For $(a,b) \in \{0\} \times \mathbb{F}_q^*$, the terms in the sum in (4) equal $T(a_0,b_0) = f_n(a_0, b \oplus b_0) f_1(a_0,b_0)$. Moreover, by Lemma 2, the latter product satisfies:

$$T(a_0,b_0) = \begin{cases} \frac{1}{q} f_n(0,b) & \text{if } (a_0,b_0) = (0,0) \\ 0 & \text{if } (a_0,b_0) \in (\{0\} \times \mathbb{F}_q^*) \cup (\mathbb{F}_q^* \times \{0\}) \\ \frac{1}{q(q-1)} f_n(a_0,0) & \text{if } (a_0,b_0) \in \mathbb{F}_q^* \times \{b\} \\ \frac{1}{q(q-1)} f_n(a_0,b_0) & \text{if } (a_0,b_0) \in \mathbb{F}_q^* \times (\mathbb{F}_q^* \backslash \{b\}) \end{cases} .$$

We deduce

$$f_{n+1}(a,b) = \frac{1}{q} f_n^{01} + (q-1)\frac{1}{q(q-1)} f_n^{01} + (q-1)(q-2)\frac{1}{q(q-1)} f_n^{11} \ . \quad (6)$$

For $(a,b) \in \mathbb{F}_q^* \times \{0\}$, we have the same equality by symmetry of the function $f_{n+1}S$.

3. For $(a,b) \in \mathbb{F}_q^* \times \mathbb{F}_q^*$, the terms in the sum in (4) equal $T(a_0,b_0) = f_n(a \oplus a_0, b \oplus b_0) f_1(a_0,b_0)$. Moreover, by Lemma 2, the latter product satisfies:

$$T(a_0,b_0) = \begin{cases} \frac{1}{q} f_n(a,b) & \text{if } (a_0,b_0) = (0,0) \\ \frac{1}{q(q-1)} f_n(0,0) & \text{if } (a_0,b_0) = (a,b) \\ 0 & \text{if } (a_0,b_0) \in (\{0\} \times \mathbb{F}_q^*) \cup (\mathbb{F}_q^* \times \{0\}) \\ \frac{1}{q(q-1)} f_n(a \oplus a_0, 0) & \text{if } (a_0,b_0) \in (\mathbb{F}_q^* \backslash \{a\}) \times \{b\} \\ \frac{1}{q(q-1)} f_n(0, b \oplus b_0) & \text{if } (a_0,b_0) \in \{a\} \times (\mathbb{F}_q^* \backslash \{b\}) \\ \frac{1}{q(q-1)} f_n(a \oplus a_0, b \oplus b_0) & \text{if } (a_0,b_0) \in (\mathbb{F}_q^* \backslash \{a\}) \times (\mathbb{F}_q^* \backslash \{b\}) \end{cases} .$$

We deduce

$$f_{n+1}(a,b) = \frac{1}{q} f_n^{11} + \frac{1}{q(q-1)} f_n^{00} + 2\Big((q-2)\frac{1}{q(q-1)} f_n^{01}\Big) + (q-2)^2 \frac{1}{q(q-1)} f_n^{11} \ . \quad (7)$$

Equations (5), (6) and (7) directly yield to the second statement. □

Theorem 1. *For every $n \geqslant 1$ we have*

$$f_n(a,b) = \begin{cases} \frac{1}{q^2} + \frac{1}{q^2(q-1)^{n-2}} & \text{if } (a,b) = (0,0) \\ \frac{1}{q^2} - \frac{1}{q^2(q-1)^{n-1}} & \text{if } (a,b) \in (\{0\} \times \mathbb{F}_q^*) \cup (\mathbb{F}_q^* \times \{0\}) \\ \frac{1}{q^2} + \frac{1}{q^2(q-1)^n} & \text{if } (a,b) \in \mathbb{F}_q^* \times \mathbb{F}_q^* \end{cases}$$

Proof. From Lemma 3, we have

$$\begin{pmatrix} f_{n+1}^{00} \\ f_{n+1}^{01} \\ f_{n+1}^{11} \end{pmatrix} = \begin{pmatrix} \frac{1}{q} & 0 & \frac{q-1}{q} \\ 0 & \frac{2}{q} & \frac{q-2}{q} \\ \frac{1}{q(q-1)} & \frac{2(q-2)}{q(q-1)} & \frac{(q-1)+(q-2)^2}{q(q-1)} \end{pmatrix} \cdot \begin{pmatrix} f_n^{00} \\ f_n^{01} \\ f_n^{11} \end{pmatrix} \ ,$$

that is

$$\begin{pmatrix} f_{n+1}^{00} \\ f_{n+1}^{01} \\ f_{n+1}^{11} \end{pmatrix} = P \cdot \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \frac{1}{q-1} \end{pmatrix} \cdot P^{-1} \cdot \begin{pmatrix} f_n^{00} \\ f_n^{01} \\ f_n^{11} \end{pmatrix} \ , \quad (8)$$

where P is the eigenvectors matrix defined by:

$$P = \begin{pmatrix} 1 & 1-q & q^2-2q+1 \\ 1 & \frac{1}{2}(2-q) & 1-q \\ 1 & 1 & 1 \end{pmatrix} \ .$$

After recursively applying (8), we can express $(f_n^{00}, f_n^{01}, f_n^{11})$ with respect to $(f_1^{00}, f_1^{01}, f_1^{11})$ as

$$\begin{pmatrix} f_n^{00} \\ f_n^{01} \\ f_n^{11} \end{pmatrix} = P \cdot \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \frac{1}{(q-1)^{n-1}} \end{pmatrix} \cdot P^{-1} \cdot \begin{pmatrix} f_1^{00} \\ f_1^{01} \\ f_1^{11} \end{pmatrix}$$

Finally, Lemma 2 implies $(f_1^{00}, f_1^{01}, f_1^{11}) = \left(\frac{1}{q}, 0, \frac{1}{q(q-1)}\right)$, which together with the above equation yields to the theorem statement. □

3.3 Exhibiting the Flaws in `IPRefresh` and `IPAdd` Procedures

Due to Lemma 1 and Theorem 1, and given that $\Pr[V = v]$ equals $\frac{1}{q}$, we get:

$$\Pr[X = x \mid V = v] = \begin{cases} \frac{1}{q} + \frac{1}{q(q-1)^{n-2}} & \text{if } x = 0 \\ \frac{1}{q} - \frac{1}{q(q-1)^{n-1}} & \text{if } x \neq 0 \end{cases} \tag{9}$$

for $v = 0$, and

$$\Pr[X = x \mid V = v] = \begin{cases} \frac{1}{q} - \frac{1}{q(q-1)^{n-1}} & \text{if } x = v \\ \frac{1}{q} + \frac{1}{q(q-1)^{n}} & \text{if } x \neq v \end{cases}, \tag{10}$$

otherwise. Hence, when the sensitive variable V equals 0, then the intermediate variable X manipulated in `IPRefresh` is more likely to equal 0 than another value in $\mathbb{F}_q$. On the other hand, when V equals a non-zero value $v \neq 0$, then X is more likely to be any value of $\mathbb{F}_q$ but v. Although the bias is exponentially small in n, for small values of n it may induce a significant information leakage (see Section 4).

For the reasons given in Section 3.1, Equations (9) and (10) also stand for the dependency of Z and V in `IPAdd`. The manipulation of Z hence leaks information on V and $\Pr[Z = z \mid V = v]$ satisfies (9) and (10).

Remark 3. The flaw in `IPMult` seems less informative than in `IPRefresh` and `IPAdd`. Indeed except for the `IPRefresh` call, we did not find any flaw in the actual algorithm. Moreover the `IPRefresh` procedure is called on a sharing of dimension n^2. Hence, even for small values of n, the observed bias quickly becomes very small.

4 Information Theoretic Evaluation of the Flaw

We have seen in Section 3.3 that Balasch *et al.*'s proposal possesses a first-order flaw whatever the masking dimension n of their scheme. To complete our study, we conduct hereafter an information theoretic evaluation of the flaw exhibited in (9) and (10), following the same outlines as the security analyses

in [8, 10, 22, 25]. Moreover, the quantity of sensitive information leakage due to the flaw is compared with the amount of intrinsic information leakage from the manipulation of the right-half sharing $\boldsymbol{R}$.

To quantify the amount of leaking information, we model the relationship between the physical leakage and the manipulated variables as follows. Each tuple of variables $(I_1, I_2, \cdots, I_t)$ is associated with a tuple of leakages $\mathcal{L} = (\mathcal{L}_1, \mathcal{L}_2, \cdots, \mathcal{L}_t)$ s.t. $\mathcal{L}_j = \mathrm{HW}(I_j) + \mathcal{N}_j$, where HW denotes the Hamming weight function and $\mathcal{N}_j$ denotes an independent Gaussian variable with mean 0 and standard deviation σ. We use the notation $\mathcal{L} \hookleftarrow (I_1, I_2, \cdots, I_t)$ to refer to this association. To compare the information revealed by the flaw and that inherently revealed by the leakage on the right-half sharing (see Remark 1 in Section 2), we computed the mutual information[2] $\mathrm{I}(V; \mathcal{L})$ between the sensitive variable $V = \langle \boldsymbol{L}, \boldsymbol{R} \rangle$ and the leakage $\mathcal{L}$ in the following situations where we recall that X equals $\langle \boldsymbol{L} \oplus \boldsymbol{L}', \boldsymbol{R} \rangle$ (see Section 3.1):

$$\text{right-half leakage for } n = 2: \quad \mathcal{L} \hookleftarrow \boldsymbol{R} = (R_1, R_2) \ , \tag{11}$$

$$\text{right-half leakage for } n = 3: \quad \mathcal{L} \hookleftarrow \boldsymbol{R} = (R_1, R_2, R_3) \ , \tag{12}$$

$$\text{first-order flaw for } n = 2: \quad \mathcal{L} \hookleftarrow X \ , \tag{13}$$

$$\text{first-order flaw for } n = 3: \quad \mathcal{L} \hookleftarrow X \ . \tag{14}$$

Figure 1 summarizes the information theoretic evaluation for each leakage (11) to (14). It can be observed that for each sharing dimension $n \in \{2, 3\}$, there exists a threshold for σ up to which the first-order flaw becomes more informative than the overall right-half leakage. For instance, for $n = 2$, this gap value is $\sigma \approx 2$. This observation is in accordance with the soundness of the d^{th}-order security notion: a security at a greater order implies a smaller asymptotic leakage (with respect to an increasing noise).

5 Attack Simulations

To study the difficulty of exploiting the sensitive information leakage exhibited in Figure 1, we compared the effectiveness of a classical Correlation Power Analysis (CPA for short) against the flaw with that of a second-order CPA targeting the half IP-Masking $\boldsymbol{R}$ (which, according to Remark 1, leaks sensitive information).

The target variable V in our attack was defined as the output of the s-box of the light-weight block cipher PRESENT [3], and hence V, R_1, R_2 and X were defined as elements of $\mathbb{F}_{16}$. The leakages on these values were simulated in the Hamming weight model with Gaussian Noise, as in (11) and (13), for different noise standard deviations $\sigma \in [0, 4.5]$. For each key hypothesis, the predictions were computed with the *optimal prediction function* defined in [21] (with the Hamming weight as model function). The results of our attack simulations are reported in Figure 2.

[2] As shown in [25], the number of measurements required to achieve a given success-rate in a maximum likelihood attack is related to the mutual information evaluation and it roughly equals $c \times \mathrm{I}(V; \mathcal{L})^{-1}$, where c is a constant related to the chosen success-rate and the leakage model.

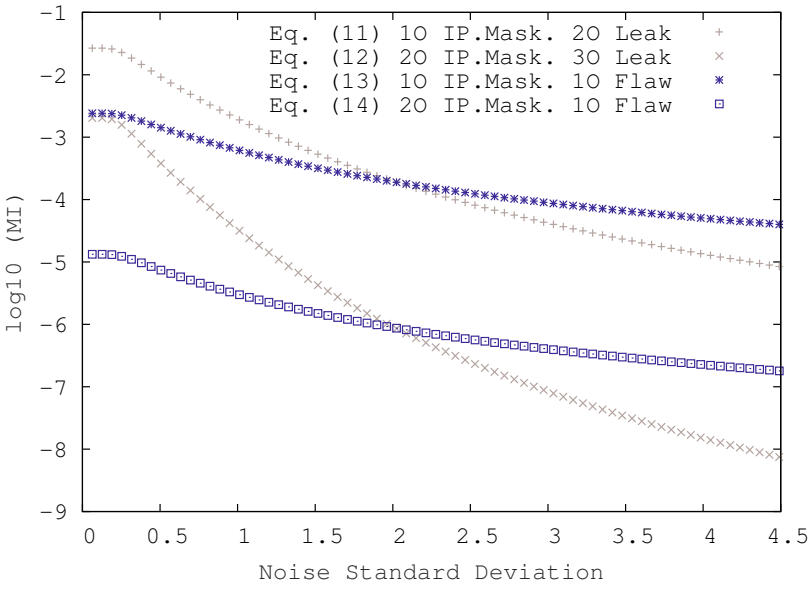

Fig. 1. Mutual information ($\log_{10}$) between the leakage and the sensitive variable over an increasing noise standard deviation (x-axis)

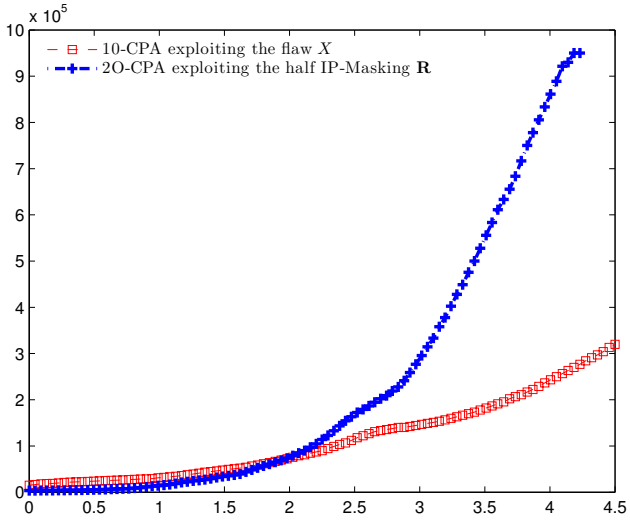

Fig. 2. Number of measurements (y-axis) required to achieve a 90% attack success rate against IP-Masking ($n = 2$) *versus* the noise standard deviation (x-axis)

It may be observed that the attack efficiencies are close when the standard deviation of the noise is lower than 2 (less than 75000 measurements), which corresponds to the crossing point of the mutual information traces in Figure 1. After this threshold, the difference between the slops of the two efficiency traces quickly increases. Eventually, for $\sigma = 4.5$, the second-order CPA against the right-half IP masking fails, even with 1 million measurements, whereas the first-order CPA against the flaw succeeds with around 300 000 measurements. This clearly illustrates the importance of the exhibited flaw. We also emphasize that the resynchronization of leakage traces and the detection of points of interest usually make higher-order attacks much more difficult to mount in practice than first-order ones. This further increases the practical insecurity resulting from a first-order leakage compared to a higher-order leakage.

References

1. Balasch, J., Faust, S., Gierlichs, B., Verbauwhede, I.: Theory and practice of a leakage resilient masking scheme. In: Wang, X., Sako, K. (eds.) ASIACRYPT 2012. LNCS, vol. 7658, pp. 758–775. Springer, Heidelberg (2012)
2. Bellare, M., Goldwasser, S., Micciancio, D.: "Pseudo-random" number generation within cryptographic algorithms: The DSS case. In: Kaliski Jr., B.S. (ed.) CRYPTO 1997. LNCS, vol. 1294, pp. 277–291. Springer, Heidelberg (1997)
3. Bogdanov, A.A., Knudsen, L.R., Leander, G., Paar, C., Poschmann, A., Robshaw, M., Seurin, Y., Vikkelsoe, C.: PRESENT: An Ultra-Lightweight Block Cipher. In: Paillier, P., Verbauwhede, I. (eds.) CHES 2007. LNCS, vol. 4727, pp. 450–466. Springer, Heidelberg (2007)
4. Carlet, C., Goubin, L., Prouff, E., Quisquater, M., Rivain, M.: Higher-order masking schemes for s-boxes. In: Canteaut, A. (ed.) FSE 2012. LNCS, vol. 7549, pp. 366–384. Springer, Heidelberg (2012)
5. Chari, S., Jutla, C., Rao, J., Rohatgi, P.: A Cautionary Note Regarding Evaluation of AES Candidates on Smart-Cards. In: Second AES Candidate Conference – AES 2 (March 1999)
6. Coron, J.-S., Prouff, E., Rivain, M., Roche, T.: Higher-order side channel security and mask refreshing. In: Moriai, S. (ed.) FSE. LNCS, Springer (2013) (to appear)
7. Dziembowski, S., Faust, S.: Leakage-resilient circuits without computational assumptions. In: Cramer, R. (ed.) TCC 2012. LNCS, vol. 7194, pp. 230–247. Springer, Heidelberg (2012)
8. Fumaroli, G., Martinelli, A., Prouff, E., Rivain, M.: Affine Masking against Higher-Order Side Channel Analysis. In: Biryukov, A., Gong, G., Stinson, D.R. (eds.) SAC 2010. LNCS, vol. 6544, pp. 262–280. Springer, Heidelberg (2011)
9. Genelle, L., Prouff, E., Quisquater, M.: Thwarting higher-order side channel analysis with additive and multiplicative maskings. In: Preneel, Takagi [19], pp. 240–255
10. Grosso, V., Standaert, F.-X., Faust, S.: Masking vs. multiparty computation: How large is the gap for aes? In: Bertoni, G., Coron, J.-S. (eds.) CHES 2013. LNCS, vol. 8086, pp. 400–416. Springer, Heidelberg (2013)
11. Ishai, Y., Sahai, A., Wagner, D.: Private Circuits: Securing Hardware against Probing Attacks. In: Boneh, D. (ed.) CRYPTO 2003. LNCS, vol. 2729, pp. 463–481. Springer, Heidelberg (2003)
12. Kim, H., Hong, S., Lim, J.: A fast and provably secure higher-order masking of aes s-box. In: Preneel, Takagi [19], pp. 95–107

13. Kocher, P.: Timing Attacks on Implementations of Diffie-Hellman, RSA, DSS, and Other Systems. In: Koblitz, N. (ed.) CRYPTO 1996. LNCS, vol. 1109, pp. 104–113. Springer, Heidelberg (1996)
14. Kocher, P., Jaffe, J., Jun, B.: Differential Power Analysis. In: Wiener, M. (ed.) CRYPTO 1999. LNCS, vol. 1666, pp. 388–397. Springer, Heidelberg (1999)
15. Lomné, V., Prouff, E., Roche, T.: Behind the Scene of Side Channel Attacks. In: Sako, K., Sarkar, P. (eds.) ASIACRYPT 2013, Part I. LNCS, vol. 8269, pp. 506–525. Springer, Heidelberg (2013)
16. Mangard, S., Popp, T., Gammel, B.M.: Side-Channel Leakage of Masked CMOS Gates. In: Menezes, A. (ed.) CT-RSA 2005. LNCS, vol. 3376, pp. 351–365. Springer, Heidelberg (2005)
17. Messerges, T.: Using Second-order Power Analysis to Attack DPA Resistant software. In: Paar, C., Ko**c**c, **c**C.K. (eds.) CHES 2000. LNCS, vol. 1965, pp. 238–251. Springer, Heidelberg (2000)
18. Oswald, E., Mangard, S., Herbst, C., Tillich, S.: Practical Second-order DPA Attacks for Masked Smart Card Implementations of Block Ciphers. In: Pointcheval, D. (ed.) CT-RSA 2006. LNCS, vol. 3860, pp. 192–207. Springer, Heidelberg (2006)
19. Preneel, B., Takagi, T. (eds.): CHES 2011. LNCS, vol. 6917. Springer, Heidelberg (2011)
20. Prouff, E., Rivain, M.: Masking against Side-Channel Attacks: A Formal Security Proof. In: Johansson, T., Nguyen, P.Q. (eds.) EUROCRYPT 2013. LNCS, vol. 7881, pp. 142–159. Springer, Heidelberg (2013)
21. Prouff, E., Rivain, M., Bévan, R.: Statistical Analysis of Second Order Differential Power Analysis. IEEE Transactions on Computers 58(6), 799–811 (2009)
22. Prouff, E., Roche, T.: Higher-order glitches free implementation of the aes using secure multi-party computation protocols. In: Preneel, Takagi [19], pp. 63–78
23. Rivain, M., Prouff, E.: Provably secure higher-order masking of aes. In: Mangard, S., Standaert, F.-X. (eds.) CHES 2010. LNCS, vol. 6225, pp. 413–427. Springer, Heidelberg (2010)
24. Shamir, A.: How to Share a Secret. Commun. ACM 22(11), 612–613 (1979)
25. Standaert, F.-X., Veyrat-Charvillon, N., Oswald, E., Gierlichs, B., Medwed, M., Kasper, M., Mangard, S.: The World is not Enough: Another Look on Second-Order DPA. In: Abe, M. (ed.) ASIACRYPT 2010. LNCS, vol. 6477, pp. 112–129. Springer, Heidelberg (2010)