Hindawi
Security and Communication Networks
Volume 2018, Article ID 7206835, 8 pages
https://doi.org/10.1155/2018/7206835

Research Article

Analysis of Software Implemented Low Entropy Masking Schemes

Dan Li,[1,2] Jiazhe Chen,[2] An Wang,[3] and Xiaoyun Wang[1,4]

[1] *Institute for Advanced Study, Tsinghua University, Beijing 100084, China*
[2] *China Information Technology Security Evaluation Center, Beijing 100085, China*
[3] *School of Computer Science, Beijing Institute of Technology, Beijing 100081, China*
[4] *Key Laboratory of Cryptologic Technology and Information Security, Ministry of Education, Shandong University, Jinan, China*

Correspondence should be addressed to Jiazhe Chen; jiazhechen@gmail.com
and Xiaoyun Wang; xiaoyunwang@mail.tsinghua.edu.cn

Received 31 October 2017; Accepted 16 January 2018; Published 26 March 2018

Academic Editor: Emanuele Maiorana

Low Entropy Masking Schemes (LEMS) are countermeasure techniques to mitigate the high performance overhead of masked hardware and software implementations of symmetric block ciphers by reducing the entropy of the mask sets. The security of LEMS depends on the choice of the mask sets. Previous research mainly focused on searching balanced mask sets for hardware implementations. In this paper, we find that those balanced mask sets may have vulnerabilities in terms of absolute difference when applied in software implemented LEMS. The experiments verify that such vulnerabilities certainly make the software LEMS implementations insecure. To fix the vulnerabilities, we present a selection criterion to choose the mask sets. When some feasible mask sets are already picked out by certain searching algorithms, our selection criterion could be a reference factor to help decide on a more secure one for software LEMS.

1. Introduction

First introduced by Kocher [1], side channel attacks (SCA) can be used to evaluate the implementation security of cryptographic ciphers by analyzing the time, the electromagnetic radiation, the power consumption, and so on [2–6].

To resist SCA, several valid countermeasures have been proposed [7–10]. Among those countermeasures, masking schemes are most popular and widely applied. The main idea of masking schemes is to make the side channel information independent of the sensitive data by randomizing the intermediate values. In general first-order masking scheme, any sensitive intermediate variable denoted by Z will be split into two shares so that $Z = S_0 \oplus S_1$, where the randomly drawn variable S_0 is called the mask. All the computations of the cryptographic algorithm are performed on the shared values independently. At the same time, the sensitive data must be recovered by recombining the two shares. For this purpose, every computation function f of cryptographic algorithms should be designed to satisfy $f(Z) = S_0' \oplus S_1'$, where S_0' and S_1' are the new shares after the operation f. If f is a linear operation with respect to XOR, then $S_0' = f(S_0)$ and $S_1' = f(S_1)$. When f is the substitution box (S-Box), some adjustment is necessary to make up for its nonlinear property. The adjusted S-Box function changes along with the value of the mask, which makes it hard to compute canceling the sensitive intermediate value analytically. Therefore, precomputing and caching the required masked S-Boxes are more relevant and efficient. However, if the mask is drawn randomly from 2^n possible masks, too much memory is required to keep all the possible masked S-Boxes. To offer a reasonable solution to balance the security protection and the performance of implementations, Low Entropy Masking Schemes (LEMS) [10, 11] are designed by limiting the amount of mask entropy.

LEMS use the masks drawn from the limited mask set $\mathcal{M} = \{m_1, m_2, \ldots, m_s\} \subset \mathbb{F}_2^n$ whose mask entropy is $\log_2(s)$. The security of LEMS implementations should be guaranteed

in two aspects. In the architecture aspect, cryptographic algorithms should carefully be implemented to avoid first-order leakage [12]. Some countermeasure techniques such as shuffling [13] can also be combined to help defeat certain bivariate and higher order attacks [14–17]. Another aspect is the chosen mask set which plays significant roles in security. Some research studied how to select them for hardware implemented LEMS [11, 18]. The selection criterion of the mask sets considered finding secure mask sets under two important assumptions [19]. The first one is that the attackers could only exploit the leakage of the masked value $Z \oplus M$. The second one is that the deterministic part of the leakage function $l_{Z\oplus M}$ is linear in the bits of masked variable $Z \oplus M$, such as Hamming weight function. Under those two conditions, the main goal of selecting mask sets for LEMS is to find balanced mask sets resistant to high order univariate CPA (following the definition of [20], the attack combining n different time instances is called n-variate attack and the n_{th} order attack is the one with n_{th} order statistical moments). Therefore, making $\mathsf{E}((l_{Z\oplus M})^{\alpha} \mid Z)$ independent of intermediate Z is the selection criterion of the mask sets for the designer of the hardware countermeasures. However, we find it is not enough for software implemented LEMS. The absolute difference $|l_{z\oplus m} - l_{z'\oplus m'}|$ may bring the unbalance to the intermediate pair (z, z'), which allows attackers to get the information of (z, z') when only the leakages corresponding to the masked values are available.

Our Contributions. In this paper, we study the unbalance in terms of absolute difference on software Low Entropy Masking Schemes (LEMS) implementations and make selection criterion for their mask sets.

(i) We find that the mask sets selected according to selection criteria in [11, 18] have the vulnerabilities based on the absolute difference measurements on software LEMS. Such vulnerabilities make the software LEMS implementations insecure when the leakages corresponding to the masked values could be exploited.

(ii) To fix the vulnerabilities and make software LEMS implementations resistant to high order univariate attacks, we further extend the selection criterion of balanced mask sets. Moreover, we prove the perfect balanced mask sets should not be linear, and their cardinalities should satisfy certain conditions.

(iii) When some feasible mask sets are already picked out by searching algorithms like those in [11], our selection criterion could be a reference factor to help decide on a more secure one from them.

Organization. The rest of the paper is organized as follows. In Section 2, we introduce the notations and some related background knowledge. Section 3 presents vulnerabilities that make the software LEMS insecure. Section 4 proves the necessary conditions that the balanced mask sets should satisfy and discusses the selection methods of mask sets. Finally, Section 5 concludes the paper.

2. Preliminaries

In this paper, sets are denoted with calligraphic letters (e.g., $\mathcal{M}$). We use capital letters (e.g., M) and lowercase ones (e.g., m) for random variables and their realizations, respectively. Throughout the paper, Z and Z' are independent and uniformly distributed random variables representing intermediates. M and M' are two independent random variables drawn from the uniform distribution in the mask set $\mathcal{M}$.

Let l_ω be the value of leakage measurements corresponding to the intermediate value ω, $\omega \in \mathbb{F}_2^n$. To match with realistic leakage functions in practice, the widely applied Hamming weight leakage model is used during the choice of the mask sets in this paper. Thus, in software environments, $l_\omega = \epsilon \mathsf{HW}[\omega] + \delta$, where ϵ is an unknown constant and δ is the Gaussian distributed ($\mathcal{N}(0, \sigma^2)$) noise. In hardware environments, $l_\omega = \epsilon \mathsf{HW}[\omega]$ (to describe the theories in [11, 18] more clearly, we use the same no noise model here). We further denote the absolute difference of two measurements corresponding to the values ω_1 and ω_2 by $|l_{\omega_1} - l_{\omega_2}|$.

Mean and variance are denoted by E and Var, respectively. Let X_1 and X_2 be two independent random variables and f be a certain function. X_1 is randomly drawn from $\mathcal{X}$. $\mathsf{E}(f(X_1, X_2) \mid X_1 = x_1)$ is the conditional expectation when $X_1 = x_1$. The variance among those conditional expectations is

$$\mathsf{Var}\left(\mathsf{E}\left(f\left(X_1, X_2\right) \mid X_1\right)\right) = \frac{1}{|\mathcal{X}|} \cdot \sum_{x_1 \in \mathcal{X}} \left(\mathsf{E}\left(f\left(X_1, X_2\right) \mid X_1 = x_1\right) - \mathsf{E}\left(f\left(X_1, X_2\right)\right)\right)^2 \quad (1)$$

which can measure the dispersion degree of $\mathsf{E}(f(X_1, X_2) \mid X_1)$. Obviously, when $\mathsf{Var}(\mathsf{E}(f(X_1, X_2) \mid X_1)) = 0$, the specific value of X_1 cannot be recognized according to $\mathsf{E}(f(X_1, X_2) \mid X_1)$. This property was mainly applied by some works [11, 18] studying the selection criterion of mask sets for hardware LEMS. Their theories are as follows.

To defeat high order univariate CPA, the value of intermediate Z should be independent of the statistic values of $l_{Z\oplus M} = \epsilon \mathsf{HW}[Z \oplus M]$. Usually, those statistics indicate αth moments denoted by $\mathsf{E}((\epsilon \mathsf{HW}[Z \oplus M])^{\alpha})$. Hence, $\mathsf{Var}(\mathsf{E}((\epsilon \mathsf{HW}[Z \oplus M])^{\alpha} \mid Z)) = 0$ is the selection criterion. The mask set is said to resist univariate dth-order attacks if $\forall\ 1 \le \alpha \le d, \alpha \in \mathbb{N}$, $\mathsf{Var}(\mathsf{E}((\epsilon \mathsf{HW}[Z \oplus M])^{\alpha} \mid Z)) = 0$.

The work in [11] proved that only 12 mask values are sufficient for $d = 2$ when $n = 8$, ($\mathcal{M}_{12} = \{03, 18, 3F, 55, 60, 6E, 8C, A5, B2, CB, D6, F9\}$). The work in [18] further studied the linear code mask sets for different d and n. For example, in [8, 4, 4] linear code mask set can reach the standard of $d = 3$ with 16 mask values when $n = 8$ (like $\mathcal{M}_{16} = \{00, 0F, 36, 39, 53, 5C, 65, 6A, 95, 9A, A3, AC, C6, C9, F0, FF\}$ used in DPA Contest v4). The linear mask set $\mathcal{M}$ has the property that $m_i \oplus m_j \in \mathcal{M}, m_i, m_j \in \mathcal{M}$ [21]. We will discuss and use the property in the following sections.

3. Vulnerabilities on Software LEMS

As stated in Section 2, the selection of the mask sets for hardware LEMS considers the balance between the intermediate

values Z and the leakage measurements $l_{Z\oplus M}$ to avoid leaking the information of Z. Nonetheless, the unbalance of absolute difference measurements $|l_{Z\oplus M} - l_{Z'\oplus M'}|$ may leak the information of intermediate pair (Z, Z') in software LEMS. In this section, we will study (α represents the order with respect to the absolute difference; indeed, the absolute difference itself is not first order according to Taylor expansion [22]; hence, the order with respect to the original leakage measurement here is higher than α) $\mathsf{E}^{(\alpha)}_{(Z,Z')} = \mathsf{E}(|l_{Z\oplus M} - l_{Z'\oplus M'}|^{\alpha} \mid Z, Z')$, $\alpha = 1, 2$. The proofs will show that $\mathsf{E}^{(2)}_{(Z,Z')}$ is independent of (Z, Z') if the mask set satisfies the hardware selection criterion: $\mathsf{Var}(\mathsf{E}(\mathsf{HW}[Z \oplus M]^{\alpha} \mid Z)) = 0$, $\alpha = 1, 2$. And it is uncertain for $\mathsf{E}^{(1)}_{(Z,Z')}$. The unbalanced $\mathsf{E}^{(1)}_{(Z,Z')}$ leads to the unbalanced variance and coefficient of variation (coefficient of variation is the ratio of standard deviation to mean), which can also help identify the intermediate pair (Z, Z') in attacks. The results of experiments show that the unbalance of $\mathsf{E}^{(1)}_{(Z,Z')}$ makes the implementations insecure. Those vulnerabilities are the properties of mask sets and cannot be fixed by the architectures of specific implementations like shuffling. So finding the balanced mask sets in terms of absolute difference is necessary for software LEMS, which will be discussed in the next section.

As $l_{z\oplus m} - l_{z'\oplus m'} \sim \mathcal{N}(\epsilon\mathsf{HW}[z \oplus m] - \epsilon\mathsf{HW}[z' \oplus m'], 2\sigma^2)$, $\mathsf{E}((l_{z\oplus m} - l_{z'\oplus m'})^2) = (\epsilon\mathsf{HW}[z \oplus m] - \epsilon\mathsf{HW}[z' \oplus m'])^2 + 2\sigma^2$ and $\mathsf{E}(|l_{z\oplus m} - l_{z'\oplus m'}|) = f(\epsilon(\mathsf{HW}[z \oplus m] - \mathsf{HW}[z' \oplus m']), \sqrt{2}\sigma)$ according to Appendix A. We deduce that

$$\mathsf{E}^{(1)}_{(z,z')} = \mathsf{E}\left(\left|l_{Z\oplus M} - l_{Z'\oplus M'}\right| \mid Z = z,\ Z' = z'\right) \tag{2}$$

$$= \frac{1}{|\mathscr{M}|^2} \sum_{m,m' \in \mathscr{M}} \mathsf{E}\left(\left|l_{z\oplus m} - l_{z'\oplus m'}\right|\right) \tag{3}$$

$$= \frac{1}{|\mathscr{M}|^2} \sum_{m,m' \in \mathscr{M}} f\left(\epsilon\left(\mathsf{HW}\left[z \oplus m\right] - \mathsf{HW}\left[z' \oplus m'\right]\right), \sqrt{2}\sigma\right), \tag{4}$$

$$\mathsf{E}^{(2)}_{(z,z')} = \mathsf{E}\left(\left(l_{Z\oplus M} - l_{Z'\oplus M'}\right)^2 \mid Z = z,\ Z' = z'\right) \tag{5}$$

$$= \frac{1}{|\mathscr{M}|^2} \sum_{m,m' \in \mathscr{M}} \mathsf{E}\left(\left(l_{z\oplus m} - l_{z'\oplus m'}\right)^2\right) \tag{6}$$

$$= \frac{1}{|\mathscr{M}|^2} \sum_{m,m' \in \mathscr{M}} \left(\left(\epsilon\mathsf{HW}\left[z \oplus m\right] - \epsilon\mathsf{HW}\left[z' \oplus m'\right]\right)^2 + 2\sigma^2\right) \tag{7}$$

$$= \mathsf{E}\left((\epsilon\mathsf{HW}\left[Z \oplus M\right])^2 \mid Z = z\right) + \mathsf{E}\left((\epsilon\mathsf{HW}\left[Z \oplus M\right])^2 \mid Z = z'\right) - 2\mathsf{E}\left(\epsilon\mathsf{HW}\left[Z \oplus M\right] \mid Z = z\right)\mathsf{E}\left(\epsilon\mathsf{HW}\left[Z \oplus M\right] \mid Z = z'\right) + 2\sigma^2. \tag{8}$$

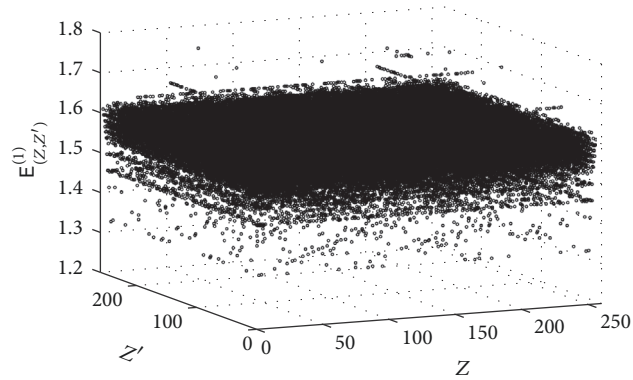

Figure 1: $\mathsf{E}^{(1)}_{(Z,Z')}$ over $\mathscr{M}_{12}$.

Obviously, for the mask set $\mathscr{M}$ which satisfies the hardware selection criterion ($\mathsf{Var}(\mathsf{E}((\epsilon\mathsf{HW}[Z \oplus M])^{\alpha} \mid Z)) = 0$, $\alpha = 1, 2$), $\mathsf{E}^{(2)}_{(Z,Z')}$ is independent of (Z, Z').

$\mathsf{E}^{(1)}_{(Z,Z')}$ is associated with the noise. For certain value σ, the value of $\mathsf{E}^{(1)}_{(Z,Z')}$ converges from $2\sigma/\sqrt{\pi}$ to $(\epsilon/|\mathscr{M}|^2)\sum_{m,m' \in \mathscr{M}} |\mathsf{HW}[z \oplus m] - \mathsf{HW}[z' \oplus m']|$ along with $\epsilon/\sigma = 0 \to \infty$. Hence, we can evaluate the unbalance of $\mathsf{E}^{(1)}_{(Z,Z')}$ for a certain mask set with $(1/|\mathscr{M}|^2)\sum_{m,m' \in \mathscr{M}} |\mathsf{HW}[z \oplus m] - \mathsf{HW}[z' \oplus m']|$. We take the mask set $\mathscr{M}_{12} \subset \mathbb{F}_2^8$ mentioned in Section 2 as an example and draw values of $\mathsf{E}^{(1)}_{(Z,Z')}$ for 2^{2*8} intermediate pairs (Z, Z') in Figure 1 which shows that $\mathscr{M}_{12}$ has vulnerabilities in terms of the absolute difference. Univariate attacks using these vulnerabilities can be performed on one S-Box.

The results of experiments in Appendix B verify that such vulnerabilities we highlighted can really threaten the security of software LEMS implementations. To make software LEMS implementations resistant to high order univariate attacks (CPA and also attacks based on the vulnerabilities above), specific implementations like shuffling are not enough and selecting the balanced mask sets in terms of the absolute difference is necessary.

4. Selection of Balanced Mask Sets

In this section, we will modify the selection criterion to find the balanced mask sets. The proofs give two conditions that the balanced mask sets should satisfy, which considerably narrow down the search for the mask sets.

The selection of the mask sets should first satisfy the criteria for hardware selections: $\mathsf{Var}(\mathsf{E}(\mathsf{HW}[Z\oplus M]^{\alpha}) \mid Z) = 0$ at least for $\alpha = 1, 2$. In such a condition, $\mathsf{E}^{(2)}_{(Z,Z')}$ is balanced as analyzed in Section 3. Hence, if $\mathsf{Var}(\mathsf{E}^{(1)}_{(Z,Z')} \mid Z, Z') = 0$, $\mathsf{E}^{(1)}$, $\mathsf{Var}_{(Z,Z')}$ and $\mathsf{CV}_{(Z,Z')}$ will also be balanced. According to (4), $\mathsf{E}^{(1)}_{(Z,Z')}$ can further be denoted by $\sigma f_e(\epsilon/\sigma, z, z')$. We can deduce that

$$\mathsf{Var}\left(\mathsf{E}^{(1)}_{(Z,Z')} \mid Z, Z'\right) = \mathsf{E}\left(\left(\mathsf{E}^{(1)}_{(Z,Z')}\right)^2\right) - \left(\mathsf{E}\left(\mathsf{E}^{(1)}_{(Z,Z')}\right)\right)^2$$

$$= \frac{\sigma^2}{2^{2n}} \sum_{z,z' \in \mathbb{F}_2^n} \left(f_e\left(\frac{\epsilon}{\sigma}, z, z'\right)\right)^2 - \left(\frac{\sigma}{2^{2n}} \sum_{z,z' \in \mathbb{F}_2^n} f_e\left(\frac{\epsilon}{\sigma}, z, z'\right)\right)^2. \tag{9}$$

$\mathsf{Var}(\mathsf{E}^{(1)}_{(Z,Z')} \mid Z, Z')$ will converge from 0 to $\gamma = \mathsf{Var}(\mathsf{E}(|\epsilon \mathsf{HW}[Z \oplus M] - \epsilon \mathsf{HW}[Z' \oplus M']| \mid Z, Z'))$ when $\epsilon/\sigma = 0 \to \infty$ for any fixed value σ.

The value of γ is an intrinsic property of the mask set. Thus, $\gamma = 0$ is the selection criterion. In this case, $\mathsf{E}^{(1)}_{(Z,Z')}$ will be balanced for any ϵ/σ. Aiming at the selection criterion, we can deduce the following conclusions to help select mask sets.

$\gamma = 0$ indicates $\mathsf{E}(|\mathsf{HW}[Z \oplus M] - \mathsf{HW}[Z' \oplus M']| \mid Z, Z')$ is a constant, the value of which is $\mathsf{E}(|\mathsf{HW}[Z \oplus M] - \mathsf{HW}[Z' \oplus M']|)$. We have the following.

Lemma 1. $\mathsf{E}(|\mathsf{HW}[Z \oplus M] - \mathsf{HW}[Z' \oplus M']|) = (2n-1)!!/2^n(n-1)!$.

Proof. Let $\mathsf{W}_a = \mathsf{E}(|\mathsf{HW}[Z] - \mathsf{HW}[Z']| \mid \mathsf{HW}[Z \oplus Z'] = a)$. $\mathsf{W}_0 = 0$, obviously. For $k \in \mathbb{N}$, we can deduce that

$$\begin{aligned} \mathsf{W}_{2k+1} &= \frac{1}{2^{2k+1}} \sum_{i=0}^{2k+1} \binom{2k+1}{i} |2k+1-2i| \\ &= \frac{1}{2^{2k+1}} \sum_{i=0}^{k} 2 \binom{2k+1}{i} (2k+1-2i) \\ &= \frac{1}{2^{2k}} \left((2k+1)\, 2^{2k} - 2(2k+1) \sum_{i=0}^{k-1} \binom{2k}{i} \right) \\ &= \frac{(2k+1)\binom{2k}{k}}{2^{2k}} = \frac{(2k+1)(2k)!}{2^{2k}(k!)^2} \\ &= \frac{(2k+1)!!}{(2k)!!}. \end{aligned} \tag{10}$$

The second equality uses $\binom{2k+1}{i} = \binom{2k+1}{2k+1-i}$.

The third one is according to $\sum_{i=0}^{k} \binom{2k+1}{i} = 2^{2k}$ and $i\binom{2k+1}{i} = (2k+1)\binom{2k}{i-1}$.

Similarly, $\mathsf{W}_{2k+2} = (2k+1)!!/(2k)!! = \mathsf{W}_{2k+1} = ((2k+1)/2k)\mathsf{W}_{2k}$. Hence

$$\begin{aligned} &\mathsf{E}\left(\left|\mathsf{HW}[Z \oplus M] - \mathsf{HW}\left[Z' \oplus M'\right]\right|\right) \\ &= \frac{1}{|\mathscr{M}|^2 4^n} \sum_{z,z' \in \mathbb{F}_2^n} \sum_{m,m' \in \mathscr{M}} \left|\mathsf{HW}[z \oplus m] - \mathsf{HW}\left[z' \oplus m'\right]\right| \\ &= \frac{1}{|\mathscr{M}|^2 4^n} \sum_{m,m' \in \mathscr{M}} \sum_{z,z' \in \mathbb{F}_2^n} \left|\mathsf{HW}[z] - \mathsf{HW}\left[z'\right]\right| \\ &= \frac{1}{4^n} \sum_{z,z' \in \mathbb{F}_2^n} \left|\mathsf{HW}[z] - \mathsf{HW}\left[z'\right]\right| \\ &= \frac{1}{2^n} \sum_{i=0}^{n} \binom{n}{i} \mathsf{W}_i. \end{aligned} \tag{11}$$

We will use mathematical induction to prove $\mathsf{A}_n = \sum_{i=0}^{n} \binom{n}{i} \mathsf{W}_i = (2n-1)!!/(n-1)!$.

When $n = 1$, $\mathsf{A}_1 = \sum_{i=0}^{1} \binom{1}{i} \mathsf{W}_i = 1 = (2-1)!!/(0)!$.

Suppose $\mathsf{A}_n = (2n-1)!!/(n-1)!$. If n is odd, we have

$$\begin{aligned} \mathsf{A}_{n+1} &= \sum_{i=0}^{n+1} \binom{n+1}{i} \mathsf{W}_i \\ &= \sum_{i=1}^{n} \left(\binom{n}{i} + \binom{n}{i-1} \right) \mathsf{W}_i + \mathsf{W}_{n+1} \\ &= \mathsf{A}_n + \sum_{i=0}^{n} \binom{n}{i} \mathsf{W}_{i+1} \\ &= \mathsf{A}_n + \sum_{i=0}^{(n-1)/2} \left(\binom{n}{2i} + \binom{n}{2i+1} \right) \mathsf{W}_{2i+1} \\ &= \mathsf{A}_n + \sum_{i=0}^{(n-1)/2} \left(\binom{n}{2i} \frac{n+1}{n} \mathsf{W}_{2i} \right. \\ &\quad \left. + \left(\binom{n}{2i} \left(1 - \frac{(n+1)(2i)}{n(2i+1)}\right) + \binom{n}{2i+1} \right) \mathsf{W}_{2i+1} \right) \\ &= \mathsf{A}_n + \frac{n+1}{n} \sum_{i=0}^{(n-1)/2} \left(\binom{n}{2i} \mathsf{W}_{2i} + \binom{n}{2i+1} \mathsf{W}_{2i+1} \right) \\ &= \frac{2n+1}{n} \mathsf{A}_n = \frac{(2n+1)!!}{(n)!}. \end{aligned} \tag{12}$$

The second equality is based on $\binom{n+1}{i} = \binom{n}{i} + \binom{n}{i-1}$. The fourth one follows $\mathsf{W}_{2i+1} = \mathsf{W}_{2i+2}$, and the fifth one uses $\mathsf{W}_{2i} = (2i/(2i+1))\mathsf{W}_{2i+1}$.

The situation when n is even can be proved similarly.

Thus, $\mathsf{E}(|\mathsf{HW}[Z \oplus M] - \mathsf{HW}[Z' \oplus M']|) = \mathsf{A}_n/2^n = (2n-1)!!/2^n(n-1)!$. □

As stated above, $\mathsf{E}(|\mathsf{HW}[Z \oplus M] - \mathsf{HW}[Z' \oplus M']| \mid Z, Z')$ for any pair (Z, Z') and the means of their combinations such as $\mathsf{E}_{\mathscr{M}} = \mathsf{E}(\mathsf{E}(|\mathsf{HW}[Z \oplus M] - \mathsf{HW}[Z \oplus M']| \mid Z))$ should be equal to the constant value $(2n-1)!!/2^n(n-1)!$. We can prove two necessary conditions for balanced mask set $\mathscr{M}$ by analyzing $\mathsf{E}_{\mathscr{M}} = (2n-1)!!/2^n(n-1)!$.

Theorem 2. *One necessary condition for $\gamma = 0$ is $|\mathscr{M}| = k2^{\lfloor n/2 \rfloor + 1}$, $k \in \mathbb{N}$.*

Proof. We deduce that

$$\mathsf{E}_{\mathscr{M}} = \mathsf{E}\left(\mathsf{E}\left(\left|\mathsf{HW}\left[Z \oplus M\right] - \mathsf{HW}\left[Z \oplus M'\right]\right| \mid Z\right)\right) \tag{13}$$

$$= \frac{1}{|\mathscr{M}|^2\, 2^n} \sum_{z \in \mathbb{F}_2^n} \sum_{m,m' \in \mathscr{M}} \left|\mathsf{HW}\left[z \oplus m\right] - \mathsf{HW}\left[z \oplus m'\right]\right| \tag{14}$$

$$= \frac{1}{|\mathscr{M}|^2} \sum_{m,m' \in \mathscr{M}} \frac{1}{2^n} \sum_{z \in \mathbb{F}_2^n} \left|\mathsf{HW}\left[z \oplus m\right] - \mathsf{HW}\left[z \oplus m'\right]\right| \tag{15}$$

$$= \frac{1}{|\mathscr{M}|^2} \sum_{m,m' \in \mathscr{M}} \mathsf{E}\left(\left|\mathsf{HW}\left[Z\right] - \mathsf{HW}\left[Z'\right]\right| \mid Z \oplus Z' = m \oplus m'\right). \tag{16}$$

Let $\mathsf{C}_i = \sum_{m,m' \in \mathscr{M}} \varrho_i(m \oplus m')$, where

$$\varrho_i(x) = \begin{cases} 1, & \mathsf{HW}\left[x\right] = i, \\ 0, & \text{otherwise.} \end{cases} \tag{17}$$

As $m \oplus m' = m' \oplus m$, C_i, $i > 0$, is even. Let $\mathsf{C}_i = 2\mathsf{C}'_i$, $i > 0$. As $\mathsf{W}_0 = 0$, $\mathsf{E}_{\mathscr{M}} = (1/|\mathscr{M}|^2) \sum_{i=0}^{n} \mathsf{C}_i \mathsf{W}_i = (2/|\mathscr{M}|^2) \sum_{i=1}^{n} \mathsf{C}'_i \mathsf{W}_i$. If $\mathsf{E}_{\mathscr{M}} = (2n-1)!!/2^n(n-1)!$, we can deduce

$$\frac{2}{|\mathscr{M}|^2} \sum_{i=1}^{n} \mathsf{C}'_i \mathsf{W}_i = \frac{(2n-1)!!}{2^n\,(n-1)!} \Longrightarrow$$

$$\frac{2^{n+1}}{|\mathscr{M}|^2} \sum_{i=1}^{n} \mathsf{C}'_i \left((n-1)!\mathsf{W}_i\right) = (2n-1)!! \Longrightarrow \tag{18}$$

$$2^{n+1} \mid |\mathscr{M}|^2.$$

The reason of the second arrow is as follows: Recall $\mathsf{W}_{2k+2} = \mathsf{W}_{2k+1} = (2k+1)!!/(2k)!!$ in Lemma 1. $\forall n \geq 2k+1$, $(n-1)!\mathsf{W}_{2k+2} = (n-1)!\mathsf{W}_{2k+1} = ((2k-1)!!)^2(2k+1)\prod_{i=2k+1}^{n-1} i \in \mathbb{N}$. In other words, $\forall i \leq n$, $(n-1)!\mathsf{W}_i \in \mathbb{N}$. $2^{n+1} \sum_{i=1}^{n} \mathsf{C}'_i((n-1)!\mathsf{W}_i) \in \mathbb{N}$ and $(2n-1)!!$ is odd. Therefore, $|\mathscr{M}|^2$ must be divisible by 2^{n+1}.

Hence, $|\mathscr{M}| = k2^{\lfloor n/2 \rfloor + 1}$, $k \in \mathbb{N}$. □

Theorem 3. *$\forall |\mathscr{M}| < 2^n \in \mathbb{N}$, if $\mathscr{M}$ is a linear mask set, $\gamma \neq 0$.*

Proof. If $\mathscr{M}$ is linear, $m \oplus m' \in \mathscr{M}$, $m, m' \in \mathscr{M}$. Let $\mathscr{D}_m = \{m \oplus m' \mid m' \in \mathscr{M}\}$. Obviously, $\forall m \in \mathscr{M}$, $\mathscr{D}_m = \mathscr{M}$. And (16) will further be

$$\mathsf{E}_{\mathscr{M}} = \frac{1}{|\mathscr{M}|^2} \sum_{m,m' \in \mathscr{M}} \mathsf{E}\left(\left|\mathsf{HW}\left[Z\right] - \mathsf{HW}\left[Z'\right]\right| \mid Z \oplus Z' = m \oplus m'\right)$$

$$= \frac{1}{|\mathscr{M}|^2} \sum_{m \in \mathscr{M}} \sum_{m' \in \mathscr{M}} \mathsf{E}\left(\left|\mathsf{HW}\left[Z\right] - \mathsf{HW}\left[Z'\right]\right| \mid Z \oplus Z' = m'\right)$$

$$= \frac{1}{|\mathscr{M}|} \sum_{i=0}^{n} \mathsf{C}_i \mathsf{W}_i, \tag{19}$$

where $\mathsf{C}_i = \sum_{m \in \mathscr{M}} \varrho_i(m)$. $\varrho_i(\cdot)$ is defined by (17).

If $\mathsf{E}_{\mathscr{M}} = (2n-1)!!/2^n(n-1)!$, we can deduce

$$\frac{1}{|\mathscr{M}|} \sum_{i=1}^{n} \mathsf{C}_i \mathsf{W}_i = \frac{(2n-1)!!}{2^n\,(n-1)!} \Longrightarrow$$

$$\frac{2^n}{|\mathscr{M}|} \sum_{i=1}^{n} \mathsf{C}_i \left((n-1)!\mathsf{W}_i\right) = (2n-1)!! \Longrightarrow \tag{20}$$

$$2^n \mid |\mathscr{M}|$$

which contradicts $|\mathscr{M}| < 2^n$. Thus, $\mathsf{E}_{\mathscr{M}} \neq (2n-1)!!/2^n(n-1)!$, which indicates $\mathsf{Var}(\mathsf{E}(|\mathsf{HW}[Z \oplus M] - \mathsf{HW}[Z' \oplus M']| \mid Z, Z')) \neq 0$. □

Theorem 2 indicates that the search should be among mask sets satisfying $|\mathscr{M}| = k2^{\lfloor n/2 \rfloor + 1}$, $k \in \mathbb{N}$, to find the perfect balanced mask set with $\gamma = 0$. However, in consideration of the effect of the noise, $\gamma = 0$ could not be necessary. According to Theorem 3 and the results in Appendix B, the linear mask sets will be more vulnerable because of their linear property. Hence, one can first use the searching algorithms like those in [11] to get some nonlinear mask sets and use our selection criterion as a reference factor to select the one with smaller γ.

5. Conclusion

In this paper, we analyzed the vulnerabilities on the mask sets of software Low Entropy Masking Schemes implementations. We found that satisfying the conditions in [11, 18] was not enough for mask sets used in software LEMS implementations. The experiments verified that such vulnerabilities certainly made the software LEMS implementations insecure. To fix the vulnerabilities, we further gave a selection criterion. Moreover, two theorems were proved, and our selection criterion could be a reference factor when selecting the mask sets picked out by searching algorithms like those in [11].

For future work, there remain two research directions. The first direction is the proof of the existence of such perfect balanced mask sets. The second one is designing more feasible search algorithms and giving the masking values selection rules based on those conditions.

Appendix

A. The Proof of $f(\mu,\sigma)$

$f(\mu,\sigma) = \mathsf{E}(|X|)$, where random variable $X \sim N(\mu,\sigma^2)$. We can deduce that

$$\begin{aligned}\mathsf{E}(|X|) &= \frac{1}{\sqrt{2\pi}\sigma}\int_{-\infty}^{\infty}|x|\,e^{-(x-\mu)^2/2\sigma^2}\,\mathrm{d}x \\ &= \frac{1}{\sqrt{2\pi}\sigma}\left(\int_0^{\infty} xe^{-(x-\mu)^2/2\sigma^2}\right. \\ &\quad \left.-\int_{-\infty}^{0} xe^{-(x-\mu)^2/2\sigma^2}\right)\mathrm{d}x.\end{aligned} \tag{A.1}$$

Here

$$\begin{aligned}\int_0^{\infty} xe^{-(x-\mu)^2/2\sigma^2}\,\mathrm{d}x &= \int_{-\mu}^{\infty}(y+\mu)\,e^{-y^2/2\sigma^2}\,\mathrm{d}y \\ &= \int_{-\mu}^{\infty} ye^{-y^2/2\sigma^2}\,\mathrm{d}y \\ &\quad + \int_{-\mu}^{\infty}\mu e^{-y^2/2\sigma^2}\,\mathrm{d}y\end{aligned} \tag{A.2}$$

$$\begin{aligned}&= \sigma^2 e^{-\mu^2/2\sigma^2} \\ &\quad + \sqrt{2\pi}\sigma\mu\left(1-\phi\left(\frac{-\mu}{\sigma}\right)\right),\end{aligned} \tag{A.3}$$

$$\begin{aligned}\int_{-\infty}^{0} xe^{-(x-\mu)^2/2\sigma^2}\,\mathrm{d}x &= -\sigma^2 e^{-\mu^2/2\sigma^2} \\ &\quad + \sqrt{2\pi}\sigma\mu\phi\left(\frac{-\mu}{\sigma}\right),\end{aligned} \tag{A.4}$$

where $\phi(x) = \int_{-\infty}^{x}(1/\sqrt{2\pi})e^{-y^2/2}\mathrm{d}y$ can be checked on the normal distribution table.

Therefore, using (A.3) and (A.4)

$$\begin{aligned}f(\mu,\sigma) &= \mathsf{E}(|X|) \\ &= \sqrt{\frac{2}{\pi}}\sigma e^{-\mu^2/2\sigma^2} + \mu\left(1-2\phi\left(\frac{-\mu}{\sigma}\right)\right).\end{aligned} \tag{A.5}$$

B. Results of Experiments

We take a typical [8, 4, 4] linear code mask set $\mathscr{M}_{16}$ mentioned in Section 2 and its variant $\mathscr{M}'_{16} = \{m \oplus 0x03 \mid m \in \mathscr{M}_{16}\}$, which are, respectively, used in the RSM (Rotating S-Box Masking (RSM) [10] is a realization of LEMS.) implementations of DPA Contest v4 and DPA Contest v4.2 [15], as examples to analyze the security in different SNR environment in practice. The software implementation of AES-256 in DPAcv4 is protected by basic RSM countermeasure, and the traces are collected from an ATMega-163 smart card. Our attacks are performed on the leakage of the outputs of S-Boxes in first-round AES. As the implementation of AES-128 in DPAcv4.2 is protected by enhanced RSM countermeasure using shuffling techniques, we carry out the attacks on the leakage of the ShiftRow in the first round where the noise is bigger.

Aiming at the vulnerabilities of unbalanced $\mathsf{E}^{(1)}_{(Z,Z')}$, lots of distinguishers can be designed. Here, we will present examples combined with the linear property of the mask set $\mathscr{M}$: $\forall m, m' \in \mathscr{M}, m \oplus m' \in \mathscr{M}$.

Such property results in the following: for any intermediate z, $\mathscr{M}^z = \{z \oplus m \mid m \in \mathscr{M}\}$ is the same as that of $z_i = z \oplus m_i$ [21]. The reason is, $\forall m \in \mathscr{M}, z_i \oplus m = z \oplus (m_i \oplus m) \in \mathscr{M}^z$, which means $\mathscr{M}^{z_i} \subset \mathscr{M}^z$. Moreover, $|\mathscr{M}^{z_i}| = |\mathscr{M}^z| = |\mathscr{M}|$. Hence, $\mathscr{M}^{z_i} = \mathscr{M}^z$. We further find the variants of the linear mask set $\{m \oplus \mathsf{C} \mid m \in \mathscr{M}\}$, where C is a constant also having the same properties. Gathering the intermediates with the same masked values together, $\mathbb{F}_2^n$ is divided into several sets $\mathscr{I}_i$, $i = 1, 2, \ldots, c$ ($z, z' \in \mathscr{I}$, if $\mathscr{M}^z = \mathscr{M}^{z'}$).

Let $\mathscr{O}$ be the set of all the measurements. $\mathscr{O}_i^k$ represents the set of measurements whose corresponding plaintext p satisfies $\psi(p,k) \in \mathscr{I}_i$, where $\psi(\cdot,\cdot)$ is the function of sensitive intermediate. The distinguisher could be

$$\mathsf{D}(k) = \frac{\mathsf{E}\left(\widehat{\theta}\left(\mathscr{O}_i^k, \mathscr{O}_i^k\right)\right)}{\widehat{\theta}(\mathscr{O},\mathscr{O})}, \tag{B.1}$$

where $\widehat{\theta}(\cdot,\cdot)$ is the estimated statistic value of absolute difference values between two measurements sets. When k is wrong, the classification $\mathscr{O}_i^k$ will be wrong and random, which makes the values of numerator and denominator approximate. When k is the correct key, the value of numerator will differ from that of denominator (Theorem 3 in Section 4 will prove this). $k^* = \arg\max_k\{\mathsf{D}(k)\}$ or $k^* = \arg\min_k\{\mathsf{D}(k)\}$.

$\widehat{\theta}$ can be $\mathsf{E}^{(1)}$, obviously. As $\mathsf{E}^{(2)}$ is independent of (Z, Z') and $\mathsf{Var}(X) = \mathsf{E}(X^2) - (\mathsf{E}(X))^2$, $\mathsf{CV}(X) = \sqrt{\mathsf{Var}(X)}/\mathsf{E}(X)$, we can also use Var and CV as $\widehat{\theta}$. We name those distinguishers for different statistics as r_v, r_{cv}, and $r_m^{(1)}$, respectively.

Using the traces in DPAcv4, we obtain 256 r_v, r_{cv}, and $r_m^{(1)}$ curves and show the time samples around the output of one S-Box in Figure 2(a). The correct key's r_v and r_{cv} curves have apparent peaks with 1000 traces. Furthermore, we generate $r_m^{(1)}$, r_v, and r_{cv} curves over the number of traces at the peak time sample and show the results in Figure 2(b). The black and 255 grey curves represent the cases of the correct key and wrong key hypotheses, respectively. The results show that all those distinguishers can recover the key with enough traces.

We then do the second experiment using traces in DPAcv4.2 at the ShiftRow in the first round where the weaker information is leaked. The three distinguishers succeed with

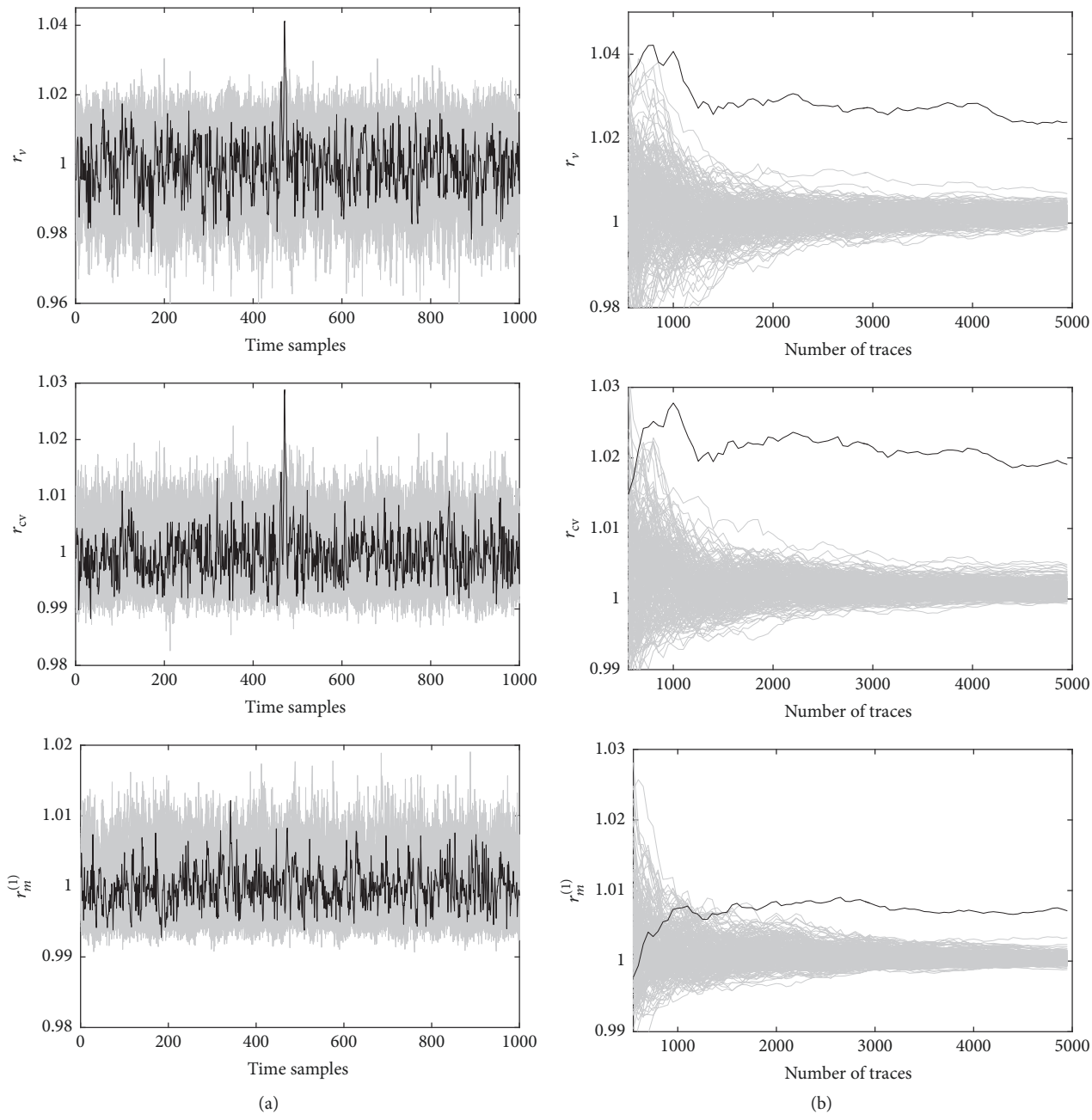

FIGURE 2: r_v, r_{cv}, and $r_m^{(1)}$ over (a) time samples using 1000 traces (b) and number of traces at the peak location.

about 6000 traces because of the lower SNR. We omit similar figures here.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This research was supported by National Key Research and Development Program of China (Grant no. 2017YFA0303903), National Natural Science Foundation of China (Grant nos. 61402536 and 61402252), Beijing Natural Science Foundation (Grant no. 4162053), National Cryptography Development Fund (Grant no. MMJJ20170201), and 973 Program (Grant no. 2013CB834205).

References

[1] P. C. Kocher, "Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems," in *Proceedings of the 16th Annual International Cryptology Conference, CRYPTO '96*, Lecture Notes in Computer Science, pp. 104–113, Springer, August 1996.

[2] S. Bhasin, J.-L. Danger, S. Guilley, and Z. Najm, "Side-channel leakage and trace compression using normalized inter-class variance," in *Proceedings of the 3rd International Workshop on Hardware and Architectural Support for Security and Privacy, HASP 2014*, pp. 7:1–7:9, ACM, USA, June 2014.

[3] G. Dabosville, J. Doget, and E. Prouff, "A new second-order side channel attack based on linear regression," *IEEE Transactions on Computers*, vol. 62, no. 8, pp. 1629–1640, 2013.

[4] M. Kayaalp, N. Abu-Ghazaleh, D. Ponomarev, and A. Jaleel, "A high-resolution side-channel attack on last-level cache," in *Proceedings of the 53rd Annual ACM IEEE Design Automation Conference, DAC 2016*, USA, June 2016.

[5] A. A. Pammu, K.-S. Chong, W.-G. Ho, and B.-H. Gwee, "Interceptive side channel attack on AES-128 wireless communications for IoT applications," in *Proceedings of the 2016 IEEE Asia Pacific Conference on Circuits and Systems, APCCAS 2016*, pp. 650–653, Republic of Korea, October 2016.

[6] Y. Li, M. Chen, and J. Wang, "Introduction to side-channel attacks and fault attacks," in *Proceedings of the 7th Asia-Pacific International Symposium on Electromagnetic Compatibility, APEMC 2016*, pp. 573–575, May 2016.

[7] R. Lumbiarres-Lopez, M. Lopez-Garcia, and E. Canto-Navarro, "Hardware architecture implemented on FPGA for protecting cryptographic keys against side-channel attacks," *IEEE Transactions on Dependable and Secure Computing*, 2016.

[8] T. Backenstrass, M. Blot, S. Pontié, and R. Leveugle, "Protection of ECC computations against side-channel attacks for lightweight implementations," in *Proceedings of the 1st IEEE International Verification and Security Workshop, IVSW 2016*, pp. 1–6, July 2016.

[9] M.-L. Akkar and C. Giraud, "An implementation of DES and AES, secure against some attacks," in *Proceedings of the third International Workshop on Cryptographic Hardware and Embedded Systems, CHES 2001*, vol. 2162 of *Lecture Notes in Computer Science*, pp. 309–318, Springer, May 2001.

[10] M. Nassar, Y. Souissi, S. Guilley, and J.-L. Danger, "RSM: a small and fast countermeasure for AES, secure against 1st and 2nd-order zero-offset SCAs," in *Proceedings of the 2012 Design, Automation & Test in Europe Conference & Exhibition, DATE 2012*, pp. 1173–1178, Dresden, Germany, March 2012.

[11] M. Nassar, S. Guilley, and J.-L. Danger, "Formal analysis of the entropy/security trade-off in first-order masking countermeasures against side-channel attacks," in *Proceedings of the 12th International Conference on Cryptology, INDOCRYPT 2011*, vol. 7107 of *Lecture Notes in Computer Science*, pp. 22–39, Springer, December 2011.

[12] A. Moradi, S. Guilley, and A. Heuser, "Detecting Hidden Leakages," in *Proceedings of the 12th International Conference on Applied Cryptography and Network Security, ACNS 2014*, vol. 8479 of *Lecture Notes in Computer Science*, pp. 324–342, Springer International Publishing, June 2014.

[13] C. Herbst, E. Oswald, and S. Mangard, "An AES smart card implementation resistant to power analysis attacks," in *Proceedings of the 4th International Conference on Applied Cryptography and Network Security, ACNS 2006*, vol. 3989 of *Lecture Notes in Computer Science*, pp. 239–252, Springer, June 2006.

[14] P. Belgarric, S. Bhasin, N. Bruneau et al., "Time-Frequency Analysis for Second-Order Attacks," in *Smart Card Research and Advanced Applications*, vol. 8419 of *Lecture Notes in Computer Science*, pp. 108–122, Springer International Publishing, Cham, 2014.

[15] S. Bhasin, N. Bruneau, J.-L. Danger, S. Guilley, and Z. Najm, "Analysis and improvements of the DPA contest v4 implementation," in *Proceedings of the 4th International Conference on Security, Privacy, and Applied Cryptography Engineering, SPACE 2014*, vol. 8804 of *Lecture Notes in Computer Science*, pp. 201–218, Springer, October 2014.

[16] C. Clavier, B. Feix, G. Gagnerot, M. Roussellet, and V. Verneuil, "Improved collision-correlation power analysis on first order protected AES," in *Proceedings of the 13th International Workshop on Cryptographic Hardware and Embedded Systems, CHES 2011*, vol. 6917, pp. 49–62, Springer, October 2011.

[17] X. Ye and T. Eisenbarth, "On the Vulnerability of Low Entropy Masking Schemes," in *Proceedings of the 12th International Conference on Smart Card Research and Advanced Applications, CARDIS 2013*, vol. 8419 of *Lecture Notes in Computer Science*, pp. 44–60, Springer International Publishing, November 2014.

[18] S. Bhasin, C. Carlet, and S. Guilley, "Theory of masking with codewords in hardware: low-weight dth-order correlation-immune boolean functions," *Cryptology ePrint Archive, IACR*, vol. 2013, p. 303, 2013.

[19] V. Grosso, F.-X. Standaert, and E. Prouff, "Low entropy masking schemes, revisited," in *Proceedings of the 12th International Conference on Smart Card Research and Advanced Applications, CARDIS 2013*, vol. 8419 of *Lecture Notes in Computer Science*, pp. 33–43, Springer, November 2014.

[20] A. Moradi and O. Mischke, "How far should theory be from practice? - evaluation of a countermeasure," in *Proceedings of the 14th International Workshop on Cryptographic Hardware and Embedded Systems, CHES 2012*, vol. 7428 of *Lecture Notes in Computer Science*, pp. 92–106, Springer, September 2012.

[21] B. Ege, T. Eisenbarth, and L. Batina, "Near collision side channel attacks," in *Proceedings of the 22nd International Conference on Selected Areas in Cryptography, SAC 2015*, vol. 9566 of *Lecture Notes in Computer Science*, pp. 277–292, Springer, August 2015.

[22] http://functions.wolfram.com/ComplexComponents/Abs/06/ShowAll.html.