Profiling Good Leakage Models For Masked Implementations

Changhai Ou[1,2], Zhu Wang[1,*], Degang Sun[1], and Xinping Zhou[1,2]

[1] Institute of Information Engineering, Chinese Academy of Sciences
[2] School of Cyber Security, University of Chinese Academy of Sciences
{ouchanghai, wangzhu, sundegang, zhouxinping}@iie.ac.cn

Abstract. Leakage model plays a very important role in side channel attacks. An accurate leakage model greatly improves the efficiency of attacks. However, how to profile a "good enough" leakage model, or how to measure the accuracy of a leakage model, is seldom studied. Durvaux et al. proposed leakage certification tests to profile "good enough" leakage model for unmasked implementations. However, they left the leakage model profiling for protected implementations as an open problem. To solve this problem, we propose the first practical higher-order leakage model certification tests for masked implementations. First and second order attacks are performed on the simulations of serial and parallel implementations of a first-order fixed masking. A third-order attack is performed on another simulation of a second-order random masked implementation. The experimental results show that our new tests can profile the leakage models accurately.

Keywords: leakage certification, HODPA, masking, leakage model, side channel attack

1 Introduction

Side channel attacks, such as Differential Power Analysis (DPA) [16], Correlation Power Analysis (CPA)[5], Template Attacks (TA) [6] and Collision Attacks (CA) [25, 24], take advantages of statistical correlations between assumed power consumption of intermediate values and true leakages, successfully attack many types of cryptographic devices. The attackers can build different distinguishers to recover the key used in the cryptographic devices based on their abilities. This means, as long as the devices leak information, the attackers can recover the key with different probabilities. This also means that devices with information leaks are unsafe. So, the evaluator can determine the security level according to whether the device leaks information or how many side channel measurements are required to detect the leakage.

According to [10], the evaluation of leakage devices against DPA attacks exploiting statistical models of leakage distributions (e.g. hamming weight model[2], hamming distance model [5], switch distance model [21]) implies answering two orthogonal questions:

(1). How informative is the leakage model used in the attack (or evaluation)?
(2). Is the leakage model used in the attack (or evaluation) correct?
The answers to these two questions determine whether the leakage model can be used to perform attacks. Here, we define two corresponding preconditions of selecting a good leakage model:
Informativeness, which means that the profiled model is useful. This guarantees that the attacker (or evaluator) can use this model to exploit the leakage of cryptographic implementation to perform effective attacks (or evaluations).
Correctness, which means that the profiled model correctly reflects the actual leakage. In other words, the profiled leakage model is a correct model. However, this precondition does not reflect how good is the model. In general, a more accurate model is more informative.

The first precondition (informativeness) has been highly investigated, which relates to the concrete security level of an implementation given a model. In order to improve the efficiency of the attacks, the attacker always tries to make full use of leakage informations, constructs optimal distinguishers and leakage models. Pre-processing such as power traces alignment [27], de-noise [15] etc, are also used. The defenders always try to reduce or eliminate the leakage of implementations. However, it's difficult for the defenders to ensure the safety of devices that leakages can not be exploited through evaluation. So, a simple method is to guarantee that the devices have no information leaks. The method is named as leakage detection, such as Welch's t-test [8, 23], Normalized Inter-Class Variance(NICV) [3], Mutual Information Analysis (MIA) [17]. The leakage detection only detects whether leakage exists, independent of whether the leakage can be exploited.

However, the second precondition (correctness) is much less investigated, which relates to the risk of a "false sense of security". In general, a correct leakage model is informative. Compared to leakage detection, leakage certification is rarely researched. However, a lot of leakage detection tests such as students' t-test, can be used in leakage certification to profile an "enough accurate" model.

Durvaux et al. performed leakage certification tests on unprotected implementations with first-order leakage, of which the model can be easily profiled. He used cross-validation to gauge the convergence of the estimated Gaussian template models and Linear Regression (LR) based models in [17]. Perceived Information (PI) is a practical tool to evaluate the leakage of a cryptographic implementation. Distance sampling technique introduced in [28] was also combined to measure the assumption and estimation errors here. However, this leakage certification test tool is time-consuming. A much faster moments-based evaluation tool was given in [10]. However, how to certify the leakage of masked implementation is still an open problem.

The goal of every countermeasure is to make the power consumption of a cryptographic device independent of the intermediate values of the cryptographic algorithm. So, It's difficult for the attacker or evaluator to profile leakage model for protected implementations defending side channel attacks, such as hiding [7] and masking. Secret sharing is a general scheme of masking, which means that

several masks are applied to one intermediate variable. A well designed masking with n shares can prevent up to an n-th order DPA attack, while it can be conquered by an $(n+1)$-th order DPA attack.

In this paper, we do a preliminary research on leakage certification tests for masked implementations. We give a practical scheme to profile the leakage model of masked implementations. Our leakage certification tests are performed on several simulated experiments. The experimental results show that our tests can not only verify the correctness but also detect the informativeness of the leakage model of first and second masked implementations.

The rest of the paper is organized as follows. leakage certification test using Gaussian templates given by Durvaux et al. [10] is introduced in Section 2. Moments-Correlating DPA, including MCP-DPA and MCC-DPA, is detailed in Section 3. The secret sharing based masked implementations, and the corresponding HODPA attacks are introduced in Section 4. Then, we simulate first and second order masked implementations in Section 5. Leakage model certification tests for first, second and third order attacks are also given in this section. Finally, Section 6 draws general conclusions.

2 Leakage Certification Tests

Side channel attacks always combine a leakage model with a distinguisher. The correct key corresponds to the most obvious value output by the distinguisher. For example, the largest mutual information of MIA, the largest difference-of-means of DPA, and the largest correlation coefficient of CPA. Different distinguishers has different distinguishing abilities, leading to different success rates [26] when analyzing the same side channel measurement set. In addition, the quality of leakage model is crucial to the performance of attacks. A good leakage model accurately reflects the leakage of implementation and improves the success rates. The goal of leakage certification tests here is to measure the correctness and informativeness of the profiled leakage model (as we introduced in Section 1).

Let us denote a plaintext byte as x,a key byte as k, the execution of the S-box Sbox as $z = \mathsf{Sbox}(x \oplus k)$, the corresponding leakage as l_z. The main idea of leakage certification tests introduced by Durvaux et al. [10] was to compare (actual) dth-order moments $\hat{M}_z^d$ estimated from the leakages with (simulated) dth-order moments $\tilde{M}_z^d$ estimated from the evaluator's model $\tilde{\Pr}_{\text{model}}$ (by sampling this model). We define the certification tests by two steps: model estimation and model certification.

2.1 Model Estimation

The evaluator estimates a probability model from a set of profiling traces $\mathcal{L}_p$: $\hat{\Pr}_{\text{model}} \leftarrow \mathcal{L}_p$. According to [10], a q-fold cross-validation is used. The evaluator splits the full set of traces $\mathcal{L}$ into q non-overlapping sets $\mathcal{L}^{(i)}$ $(1 \leq j \leq q)$ of

approximately the same size. Then, profiling sets $\mathcal{L}_p^{(j)} = \bigcup_{i \neq j} \mathcal{L}^{(i)}$ and test sets $\mathcal{L}_t^{(j)} = \mathcal{L} \setminus \mathcal{L}_p^{(i)}$ are defined.

Since we use Moments-Correlating DPA here, the models we use are the moments. We need to profile models for all possible values of a byte of plaintext. These models are the ones we estimate. Thus, the evaluator then computes an estimate of (actual) dth-order moments $\hat{M}_z^{d,(j)}$ from each set. Actually, the evaluator can profile different leakage models and evaluate the performance of these models. The evaluation procedure is similar to model estimation and model certification introduced in this section.

2.2 Model Certification

The evaluator generates a simulated set of traces by sampling the model: $\tilde{\mathcal{L}} \leftarrow \hat{\Pr}_{\text{model}}$. The size of $\tilde{\mathcal{L}}$ is equal to $\mathcal{L}$. The evaluator then divides this set into q subsets $\tilde{\mathcal{L}}^{(j)}$ $(1 \leq j \leq q)$ and produces a dth-order moments estimate $\tilde{m}_z^{d,(j)}$ from each of them. Then, the evaluator computes the following quantities:

$$\begin{aligned} \hat{\mu}_z^d &= \hat{\mathrm{E}}_j\left(\hat{m}_z^{d,(j)}\right), \qquad \hat{\sigma}_z^d = \sqrt{\hat{\mathrm{var}}_j\left(\hat{m}_z^{d,(j)}\right)}, \\ \tilde{\mu}_z^d &= \tilde{\mathrm{E}}_j\left(\tilde{m}_z^{d,(j)}\right), \qquad \tilde{\sigma}_z^d = \sqrt{\tilde{\mathrm{var}}_j\left(\tilde{m}_z^{d,(j)}\right)}, \end{aligned} \tag{1}$$

using these real and simulated estimates. $E(.)$ is sample mean operator, and $\tilde{\mathrm{var}}$ is the sample variance operator. Then, Students' t-test Δ_z^d is estimated:

$$\Delta_z^d = \frac{\hat{\mu}_z^d - \tilde{\mu}_z^d}{\sqrt{\frac{(\hat{\sigma}_z^d)^2 + (\tilde{\sigma}_z^d)^2}{k}}}. \tag{2}$$

Let CDF_t denote the Student's t cumulative distribution function, and d_f denote the corresponding number of freedom degree (see Sect. 4.1 [10]). Then, the probability that the observed difference is the result of estimations issues

$$p = 2 \times (1 - \mathrm{CDF}_t(|\Delta_z^d, d_f|)). \tag{3}$$

The value of p only reflects that the difference between the real and the simulated estimates has (or does have) statistical significance, it does not reflect the size of differences between them. The larger the value of p, the smaller probability of an incorrect estimated model of chip.

3 Moments-Correlating DPA

Let y denote the univariate random variable here, $E(.)$ denote the expectation operator. The dth-order raw moments are defined as $M_y^d = E(Y^d)$, with $\mu_y = E(Y)$ the mean. The dth-order central moments are defined as $CM_y^d = E((Y - \mu)^d)$, with $\sigma_y^2 = E((Y - \mu)^2)$ the variance. The dth-order standardized moments

are defined as $SM_y^d = E((\frac{Y-\mu}{\sigma})^d)$, with $\alpha = SM_y^3 = E((\frac{Y-\mu}{\sigma})^3)$ the skewness and $\beta = SM_y^4 = E((\frac{Y-\mu}{\sigma})^4)$ the kurtosis (as shown in Fig. 1). In this paper, we use these moments to perform evaluations.

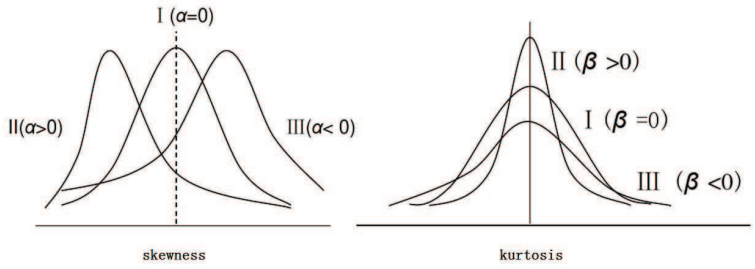

Fig. 1. Probability density function with different skewness and kurtosis values.

Skewness describes asymmetry from the normal distribution in a set of statistical data. A distribution is skewed if one tail is longer than another. As shown in Fig. 1, a left-skewed distribution ($\alpha > 0$) has a long left tail, a right-skewed distribution ($\alpha < 0$) has a long right tail. If $\alpha = 0$, the data is normal distributed. Kurtosis is the measure of the thickness or heaviness of the tails of a distribution. If $\beta = 0$, the distribution that has tails shaped in roughly the same way as any normal distribution. If $\beta > 0$, the distribution has slender tails. If $\beta < 0$, the distribution has heavy tails. When the attacker profiles the leakage model, the probability density function of it plays an important role in the attacks.

3.1 Moments-Correlating Profiled DPA

Let $l_{x,k}$ denote an N-element vector of leakage traces corresponding to N intermediate values z. Suppose that the attacker uses N_p-element vector of profiling traces $l^p_{x,k}$ to profile the dth-order raw moments: $\hat{M}^d_{x,k} \leftarrow l^p_{x,k}$. The distinguisher Moments-Correlating Profiled DPA (MCP-DPA) defined in [19] can be given as:

$$\tilde{k} = \underset{k^*}{\operatorname{argmax}} \hat{\rho} \left(\hat{M}^d_{x,k^*}, \left(l^t_{x,k} \right)^d \right), \tag{4}$$

where $\hat{\rho}$ is Pearson's correlation coefficient, $\hat{M}^d_{x,k^*}$ is the dth-order (estimated) statistical moment vector corresponding to key hypothesis k^*, and $l^t_{x,k}$ is an N_t-element vector of test traces. The attacker can also use central moments or standardized moments here.

In fact, MCP-DPA is very similar to Template Attack (TA). The estimated dth-order statistical moments $\hat{M}^d_{x,k}$ can be regarded as the dth-order profiled template of plaintext x. Equation 4 can be regarded as templates matching in the key recovery stage.

3.2 Moments-Correlating Collision Attack

The first correlation-based collision attack was given in [18], which was named as Correlation Enhanced Power Analysis Collision Attack (CEPACA). Here let us denote the input plaintexts and subkeys of two S-boxes as (x_0, k_0) and (x_1, k_1). If a collision happens, then $\mathsf{Sbox}(x_0 \oplus k_0) = \mathsf{Sbox}(x_1 \oplus k_1)$, we have $x_0 \oplus k_0 = x_1 \oplus k_1$. Finally, we get $\Delta = x_0 \oplus x_1 = k_0 \oplus k_1$. The attacker computes the correlation between two mean power consumption vectors of two different S-boxes:

$$\tilde{\Delta} = \underset{\Delta}{\operatorname{argmax}} \tilde{\rho} \left(\tilde{M}_{x_0,k_0}, \tilde{M}_{x_1,k_1 \oplus \Delta} \right). \tag{5}$$

The maximum correlation coefficient means that collision occurs, the attacker can recover the corresponding Δ. Take AES-128 for example, the attacker gets a Δ from the j-th and $(j+1)$-th sub-keys ($1 \leq j \leq 15$). To recover the full 128-bit key, the attacker only needs to find 15 pairs of collisions and enumerate the first sub-key (as introduced in [4]).

An enhanced CEPACA given in [19] is

$$\tilde{\Delta} = \underset{\Delta}{\operatorname{argmax}} \tilde{\rho} \left(\tilde{M}^d_{x_0,k_0}, \left(l^t_{x_1,k_1 \oplus \Delta} \right)^d \right), \tag{6}$$

which is named as Moments-Correlating Collision DPA (MCC-PDA). The second enhanced CEPACA using dth-order moments is given: $\tilde{\rho} \left(\tilde{M}^d_{x_0,k_0}, \tilde{M}^d_{x_1,k_1 \oplus \Delta} \right)$. The correlation between two estimated dth-order moments are computed. Compared to MCC-DPA, this enhanced CEPACA is more stable and less information is lost. These two enhanced CEPACA can successfully attack unprotected devices or some protected devices with first-order leakage. New attacks are needed for second or higher-order masked implementations. The attacker or evaluator needs to profile moments-based leakage models in these moments-based attacks.

4 Masked Implementations and HODPA Attacks

4.1 Secret Sharing

The masked implementations take advantage of secret sharing. The sensitive intermediate variable is divided into several shares. Thus, the correlation between manipulation of intermediate variable and the side channel leakages is hidden. Masking is a very simple, efficient and feasible protection method of cryptographic implementation.

Let I denote a sensitive intermediate variable, information leakage on which implies information leakage on K. The goal of a defender is to eliminate the correlation between the power consumption and the manipulation of I. The nth-order masked implementations divide the intermediate values (e.g. the outputs of S-boxes) into n shares $S_1, S_2, \ldots, S_n$. The sensitive intermediate variable I here satisfies

$$S_1 \circ S_2 \circ \cdots \circ S_n = I, \tag{7}$$

where $\circ$ denotes a group operation. The $n-1$ shares $S_1, S_2, \ldots, S_{n-1}$ are mutually independent random uniformly distributed variables. The last share S_n is also a random variable, which is determined by $S_1, S_2, \ldots, S_{n-1}$ and satisfies the Equation 7.

4.2 HODPA attacks on Masked Implementations

In order to resist HODPA attacks, the n shares are generally operated at different times without overlapping. Leakage detection can be used to identify data-dependent information in side-channel measurements. Even if an attacker finds $n-1$ shares, he can not get any information about the intermediate variable. He must find at least a sample combination of n shares. In this case, the intermediate variable can be re-combined, the attacker can exploit the correlation between the power consumption and the manipulation of I again. However, the side channel measurements are often very long, and the leakage location of each share is uncertain. So, it's especially difficult to find such a combination. Only several papers [9, 12, 20] talked about this. Most HODPA attacks assume that the leakage positions of n shares are known. Thus, HODPA attacks can be directly performed without interesting points identification.

The leakage models of higher-order maskings are also hard to profile, which is also an open problem of leakage certification in [10]. Only two of Durvaux's papers [11, 10] discussed leakage certification, the corresponding research was limited to unprotected devices. There is no other literature discussing about leakage certification for masked devices. However, many researchers have studied how to profile the leakage model of masked implementations. Thanks to these studies, we can perform leakage certification tests on masked implementations.

In order to recover the key, the attacker needs to find out at least a time samples combination of all these shares. Let $L(t_1)$, $L(t_2)$, ..., $L(t_n)$ denote the corresponding leakages of shares $S_1, S_2, \ldots, S_n$, a combination function used here is to combine the corresponding sample points using normalized product combining function $\mathcal{C}$ prior to the model estimation step:

$$\mathcal{C}\left(L(t_1), L(t_2), \ldots, L(t_n)\right) = \prod_{i=1}^{n} \left(L(t_i) - E\left(L(t_i)\right)\right), \tag{8}$$

,which is proved to be optimal. $E(.)$ here is sample mean operator. Finally, the normalized product becomes a new variable. We establish moments on this new variable for each plaintext x to perform side channel attacks, such as MCP-DPA and MCC-DPA. An another combination function using subtraction introduced in [20] can also be used. By doing this, we can perform the leakage model certification easily.

5 Simulated Experiments

5.1 First-Order Fixed Masking Implementation

In this section, we use low entropy masking introduced in [14] to simulate traces to analyze the performance of our new leakage model certification tests. Let Sbox denote the S-box operation, x denote the input plaintexts, k denote the subkey, N_1, N_2 denote two normal distributed noise variables with mean 0 and variance σ^2, and s denote a fixed value to protect the sensitive S-box output variable. We define the leakages of a serial implementation as:

$$\begin{aligned} L_1 &= L(s) + N_1, \\ L_2 &= L(\mathsf{Sbox}(x \oplus k) \oplus s) + N_2, \end{aligned} \tag{9}$$

where $L(.)$ is a leakage function, here hamming weight function is used. Obviously, this is a flawed first-order masking scheme, the attacker can directly attack the S-box output, though he needs more traces. He can also perform second-order attacks through combining two leakage samples.

The evaluation of our new leakage certification tests against the masking is more feasible thanks to the MATLAB source code on the website [1] provided by Durvaux et al.

We first perform first-order evaluation on the simulated traces. 256 models for all possible plaintext values are profiled. 256000 traces are generated and cross-validation is used to decrease estimation errors. The noise standard deviation is set to 1. The p values output by t-test introduced in Sect. 2 are shown in Fig. 2. The X-axis is the number of traces used in leakage certification, the Y-axis is the plaintext. Each horizontal line denotes the p-values under different number of power traces.

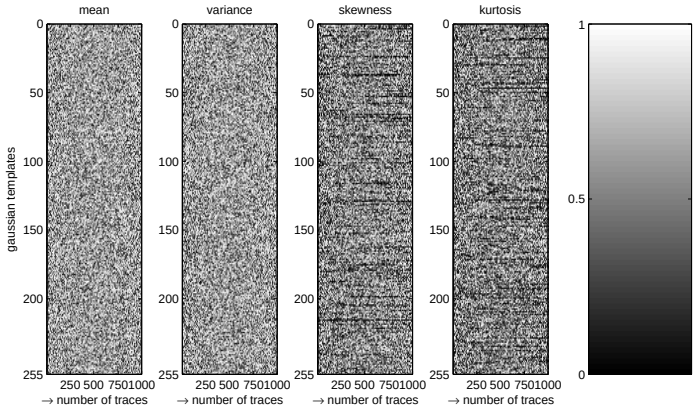

Fig. 2. Leakage certification tests for first-order leakage.

As shown in Fig. 2, the Gaussian templates can capture the simulated traces quite accurately. Almost no assumption errors happen in these four moments (from means to kurtosis). This indicates that the fixed mask doesn't affect the model profiling. However, information leaks happen in the first-order moment. The corresponding moments informativeness are shown in Fig. 3. The correlation coefficient of the means is about 0.81, while other moments are nearly zero. This also indicates that only first-order leakage can be used if we attack the S-box output of our masking schemes.

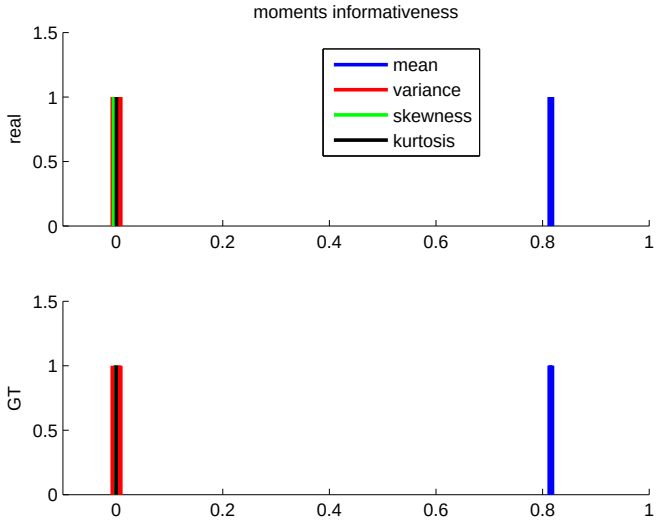

Fig. 3. First-order MCP-DPA attack results for 256×1000 simulated traces.

We then perform second-order evaluation on the simulated traces. The noise standard deviation is also set to 1. The p values output by t-test are shown in Fig. 4. Unlike first-order attack, the noise introduced by mask obviously increases in the second-order attack. The model profiling of means and variances are still accurate. However, assumption errors happen with high probability in the third and fourth order moments.

The corresponding leakage of second-order attack is shown in Fig. 5. The noise introduced by mask seriously affect the informativeness of first moments, which decreases from about 0.81 to about 0.30. The variances do not carry information in first-order attack (as shown in Fig. 3), however, it leaks information in the second-order attack. Its informativeness is almost as much as the means.

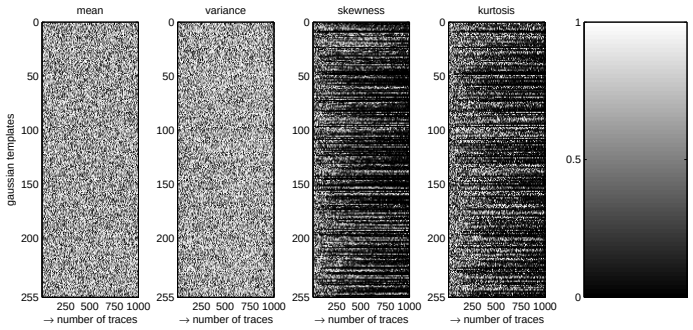

Fig. 4. Second-order MCP-DPA attack results for 256×1000 simulated traces.

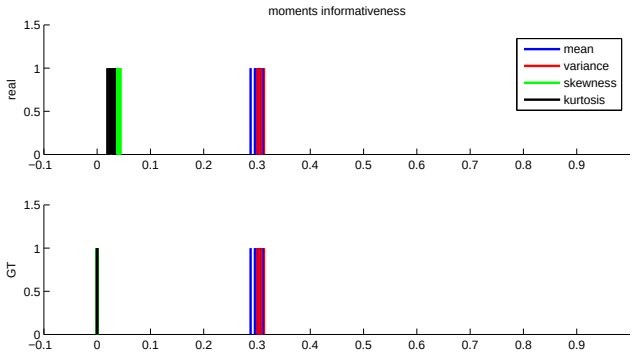

Fig. 5. Leakage certification tests for second-order leakage.

Let N_3 denote a normal distributed noise variable with mean 0 and variance σ_3^2. We also simulate another parallel implementation leakage as:

$$L_3 = L(\mathsf{Sbox}(x \oplus k) \oplus s) + L(s) + N_3, \tag{10}$$

and perform leakage model certification tests on this time sample. The experimental results of which is very similar to the previous one shown in Fig. 2 and Fig. 3. Although information leakage model of moments from means to kurtosis can be accurately profiled, the means are the only one informative moments. It is worth noting that, since only 256 moments-based models are profiled, the procedure of leakage certification is very fast.

5.2 Second-Order Fixed Masking Implementation

We also simulate a serial implementation of second-order masking and evaluate the leakage using our new leakage model certification tests. Let N_3 denote a

normal distributed noise variable with mean 0 and variance σ_3^2, and s_1, s_2 denote two fixed values to protect the sensitive S-box outputs. We define the serial implementation leakages as:

$$\begin{aligned} L_1 &= L(s_1) + N_1, \\ L_2 &= L(s_2) + N_2, \\ L_3 &= L(\mathsf{Sbox}(x \oplus k) \oplus s_1 \oplus s_2) + N_3. \end{aligned} \tag{11}$$

Although this second-order masking implementation exists first-order leakage, the attacker can directly attack the outputs of S-box, we use 3^{rd}-order attack combining these three time samples. The p-values of Students' t-test are shown in Fig. 6. The 256 leakage models of means and variances can be accurately profiled. However, the 3^{rd} and 4^{th} moments are hard to profile. Fig. 7 also shows that the means and variances are the most informative moments, the corresponding correlation coefficients of which are about 0.47 and 0.26 respectively.

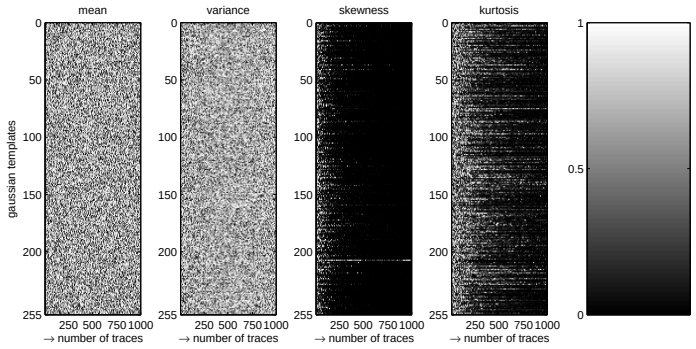

Fig. 6. Leakage certification tests for 3^{rd}-order leakage.

To sum up, the informativeness of moments decreases. This validates the conclusion of Benedikt et al. [13] that HODPA performing significantly worse can be assigned to a wrong assumption error in the leakage model. This can be explained by information loss by using normalized product as combination function. Our experiment results also verify the claim of Benedikt et al. that, the optimality claim of normalized product in [22] that only holds for second-order attack. In [22], the authors claimed that Multivariate Mutual Information Analysis (MMIA) was preferable over higher-order DPA since the information loss in MMIA was small. If an enhanced MMIA can be used in higher-order leakage model certification tests, the information loss can be reduced.

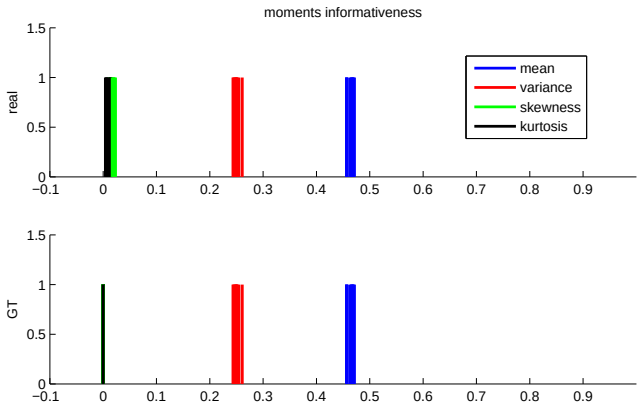

Fig. 7. 3^{rd}-order MCP-DPA attack results for 256×1000 simulated traces.

5.3 Random Masked Implementations

If the plaintext x and the mask m are bijective, or the mask is a fixed value, the evaluator can directly profile the power consumption of intermediate variable through plaintext. Otherwise, the existence of the mask increases the difficulty of profiling. For example, if we randomize the mask s in our experiment (Equation 9) and perform second-order leakage model certification tests, the most informative moments are the means and variances, the corresponding correlation coefficients are about 0.09 and 0.12. It is difficult for the attacker to recover the key from moments from means to kurtosis since the presence of noise, the correlation coefficients corresponding to some error guess keys will larger than them.

Fortunately, random masked implementations can also be profiled. Suppose that $\mathsf{Sbox}(x_0 \oplus k_0) \oplus s)$ and $\mathsf{Sbox}(x_1 \oplus k_1) \oplus s)$ are two Sbox operations masked by the same random variable s. If a collision happens, $\mathsf{Sbox}(x_0 \oplus k_0) \oplus s) = \mathsf{Sbox}(x_1 \oplus k_1) \oplus s)$. This also means, $x_0 \oplus k_0 = x_1 \oplus k_1$. The attacker gets $\Delta = x_0 \oplus x_1 = k_0 \oplus k_1$. If the evaluator or attacker just considers the plaintext x, the model certification tests can also pass too. However, the information of moments from means to kurtosis is lost.

The attacker can profile 256×256 moments-based templates for all possible combinations of x_0 and s. The templates of the second Sbox can be profiled in the same way. By doing this, the attacker or evaluator can profile the leakage models very accurately. The experimental results are shown in Fig. 8. The means are the most informative moments. The correlation is very weak in the variance moments, then lost in the skewness and kurtosis moments. The drawback of this approach is that, the attacker or evaluator needs to profile a large number of moments-based leakage models, a lot of side channel measurements are required, and the certification tests are very time-consuming.

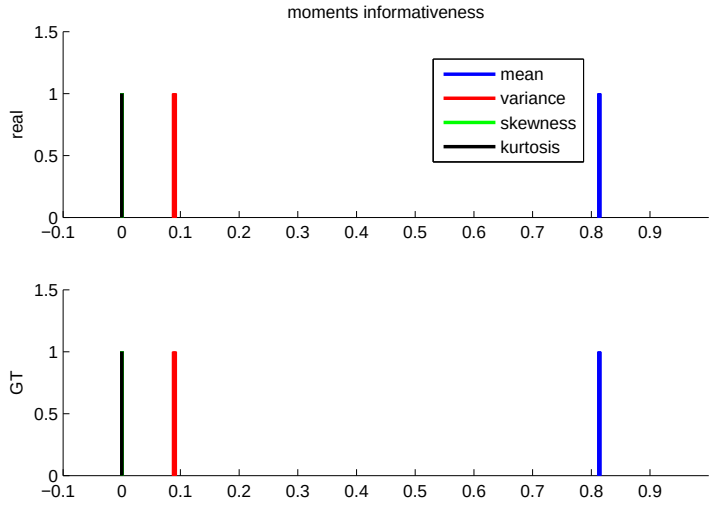

Fig. 8. First-order random masked implementation leakage model certification tests for $256 \times 256 \times 1000$ simulated traces.

6 Conclusions and Future Works

Model profiling is critical to side channel attacks. An accurate leakage model can significantly improve the attack efficiency. Therefore, accurate profiling of leakage model becomes one of the main goals of attackers. An accurate model also allows the evaluator to measure the leakage and security level of cryptographic implementations. It's helpful to design effective protection schemes. However, whether a profiled model is accurate or not, or how to measure the accuracy of a model is rarely studied. Although Durvaux et al. gave several schemes to certify the leakage model of unprotected implementations, no practical schemes for protected ones are given.

In this paper, we give a practical scheme to profile leakage model for masking protected devices. We simulate three masked implementations and perform first, second and third order leakage model certification tests. The experimental results show that our new leakage certification tests can accurately profile the leakage models. This is the first practical research of model profiling for masked implementations. However, it's still difficult to profile the leakage model for random masked implementations. Moreover, how to reduce the information loss when profiling higher-order leakage model is another interesting problem. We hope these problems can be well solved in the future.

References

1. http://perso.uclouvain.be/fstandae/PUBLIS/171.zip.
2. M. Akkar, R. Bevan, P. Dischamp, and D. Moyart. Power analysis, what is now possible... In *Advances in Cryptology - ASIACRYPT 2000, 6th International Conference on the Theory and Application of Cryptology and Information Security, Kyoto, Japan, December 3-7, 2000, Proceedings*, pages 489–502, 2000.
3. S. Bhasin, J. Danger, S. Guilley, and Z. Najm. NICV: normalized inter-class variance for detection of side-channel leakage. *IACR Cryptology ePrint Archive*, 2013:717, 2013.
4. A. Bogdanov and I. Kizhvatov. Beyond the limits of DPA: combined side-channel collision attacks. *IEEE Trans. Computers*, 61(8):1153–1164, 2012.
5. E. Brier, C. Clavier, and F. Olivier. Correlation power analysis with a leakage model. In *Cryptographic Hardware and Embedded Systems - CHES 2004: 6th International Workshop Cambridge, MA, USA, August 11-13, 2004. Proceedings*, pages 16–29, 2004.
6. S. Chari, J. R. Rao, and P. Rohatgi. Template attacks. In *Cryptographic Hardware and Embedded Systems - CHES 2002, 4th International Workshop, Redwood Shores, CA, USA, August 13-15, 2002, Revised Papers*, pages 13–28, 2002.
7. P. Corsonello, S. Perri, and M. Margala. An integrated countermeasure against differential power analysis for secure smart-cards. In *International Symposium on Circuits and Systems (ISCAS 2006), 21-24 May 2006, Island of Kos, Greece*, 2006.
8. A. A. Ding, C. Chen, and T. Eisenbarth. Simpler, faster, and more robust t-test based leakage detection. In *Constructive Side-Channel Analysis and Secure Design - 7th International Workshop, COSADE 2016, Graz, Austria, April 14-15, 2016, Revised Selected Papers*, pages 163–183, 2016.
9. F. Durvaux and F. Standaert. From improved leakage detection to the detection of points of interests in leakage traces. In *Advances in Cryptology - EUROCRYPT 2016 - 35th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Vienna, Austria, May 8-12, 2016, Proceedings, Part I*, pages 240–262, 2016.
10. F. Durvaux, F. Standaert, and S. M. D. Pozo. Towards easy leakage certification. In *Cryptographic Hardware and Embedded Systems - CHES 2016 - 18th International Conference, Santa Barbara, CA, USA, August 17-19, 2016, Proceedings*, pages 40–60, 2016.
11. F. Durvaux, F. Standaert, and N. Veyrat-Charvillon. How to certify the leakage of a chip? In *Advances in Cryptology - EUROCRYPT 2014 - 33rd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Copenhagen, Denmark, May 11-15, 2014. Proceedings*, pages 459–476, 2014.
12. F. Durvaux, F. Standaert, N. Veyrat-Charvillon, J. Mairy, and Y. Deville. Efficient selection of time samples for higher-order DPA with projection pursuits. In *Constructive Side-Channel Analysis and Secure Design - 6th International Workshop, COSADE 2015, Berlin, Germany, April 13-14, 2015. Revised Selected Papers*, pages 34–50, 2015.
13. B. Gierlichs, L. Batina, B. Preneel, and I. Verbauwhede. Revisiting higher-order DPA attacks:. In *Topics in Cryptology - CT-RSA 2010, The Cryptographers' Track at the RSA Conference 2010, San Francisco, CA, USA, March 1-5, 2010. Proceedings*, pages 221–234, 2010.
14. V. Grosso, F. Standaert, and E. Prouff. Low entropy masking schemes, revisited. In *Smart Card Research and Advanced Applications - 12th International Conference,*

CARDIS 2013, Berlin, Germany, November 27-29, 2013. Revised Selected Papers, pages 33–43, 2013.

15. S. Hajra and D. Mukhopadhyay. On the optimal pre-processing for non-profiling differential power analysis. In *Constructive Side-Channel Analysis and Secure Design - 5th International Workshop, COSADE 2014, Paris, France, April 13-15, 2014. Revised Selected Papers*, pages 161–178, 2014.
16. P. C. Kocher, J. Jaffe, and B. Jun. Differential power analysis. In *Advances in Cryptology - CRYPTO '99, 19th Annual International Cryptology Conference, Santa Barbara, California, USA, August 15-19, 1999, Proceedings*, pages 388–397, 1999.
17. L. Mather, E. Oswald, J. Bandenburg, and M. Wójcik. Does my device leak information? an a priori statistical power analysis of leakage detection tests. In *Advances in Cryptology - ASIACRYPT 2013 - 19th International Conference on the Theory and Application of Cryptology and Information Security, Bengaluru, India, December 1-5, 2013, Proceedings, Part I*, pages 486–505, 2013.
18. A. Moradi, O. Mischke, and T. Eisenbarth. Correlation-enhanced power analysis collision attack. In *Cryptographic Hardware and Embedded Systems, CHES 2010, 12th International Workshop, Santa Barbara, CA, USA, August 17-20, 2010. Proceedings*, pages 125–139, 2010.
19. A. Moradi and F. Standaert. Moments-correlating DPA. *IACR Cryptology ePrint Archive*, 2014:409, 2014.
20. E. Oswald, S. Mangard, C. Herbst, and S. Tillich. Practical second-order DPA attacks for masked smart card implementations of block ciphers. In *Topics in Cryptology - CT-RSA 2006, The Cryptographers' Track at the RSA Conference 2006, San Jose, CA, USA, February 13-17, 2006, Proceedings*, pages 192–207, 2006.
21. E. Peeters. *Advanced DPA theory and practice*. Springer, 2013.
22. E. Prouff, M. Rivain, and R. Bevan. Statistical analysis of second order differential power analysis. *IEEE Trans. Computers*, 58(6):799–811, 2009.
23. O. Reparaz. Detecting flawed masking schemes with leakage detection tests. In *Fast Software Encryption - 23rd International Conference, FSE 2016, Bochum, Germany, March 20-23, 2016, Revised Selected Papers*, pages 204–222, 2016.
24. K. Schramm, G. Leander, P. Felke, and C. Paar. A collision-attack on AES: combining side channel- and differential-attack. In *Cryptographic Hardware and Embedded Systems - CHES 2004: 6th International Workshop Cambridge, MA, USA, August 11-13, 2004. Proceedings*, pages 163–175, 2004.
25. K. Schramm, T. J. Wollinger, and C. Paar. A new class of collision attacks and its application to DES. In *Fast Software Encryption, 10th International Workshop, FSE 2003, Lund, Sweden, February 24-26, 2003, Revised Papers*, pages 206–222, 2003.
26. F. Standaert, T. Malkin, and M. Yung. A unified framework for the analysis of side-channel key recovery attacks. In *Advances in Cryptology - EUROCRYPT 2009, 28th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Cologne, Germany, April 26-30, 2009. Proceedings*, pages 443–461, 2009.
27. J. G. J. van Woudenberg, M. F. Witteman, and B. Bakker. Improving differential power analysis by elastic alignment. In *Topics in Cryptology - CT-RSA 2011 - The Cryptographers' Track at the RSA Conference 2011, San Francisco, CA, USA, February 14-18, 2011. Proceedings*, pages 104–119, 2011.

28. N. Veyrat-Charvillon and F. Standaert. Generic side-channel distinguishers: Improvements and limitations. In *Advances in Cryptology - CRYPTO 2011 - 31st Annual Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2011. Proceedings*, pages 354–372, 2011.