SNR-centric power trace extractors for side-channel attacks

Ou, Changhai; Lam, Siew-Kei; Sun, Degang; Zhou, Xinping; Qiao, Kevin; Wang, Qu

2021

Ou, C., Lam, S., Sun, D., Zhou, X., Qiao, K. & Wang, Q. (2021). SNR-centric power trace extractors for side-channel attacks. IEEE Transactions On Computer-Aided Design of Integrated Circuits and Systems, 40(4), 620-632.

<https://dx.doi.org/10.1109/TCAD.2020.3003849>

<https://hdl.handle.net/10356/147252>

<https://doi.org/10.1109/TCAD.2020.3003849>

© 2020 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works. The published version is available at:

<https://doi.org/10.1109/TCAD.2020.3003849>

Downloaded on 26 Apr 2023 17:30:43 SGT

SNR-Centric Power Trace Extractors for Side-Channel Attacks

Changhai Ou, Siew-Kei Lam, Degang Sun, Xinping Zhou, Kexin Qiao, and Qu Wang

Abstract—Existing power trace extractors consider the case where the number of power traces available to the attacker is sufficient to guarantee successful attacks, and the goal of power trace extraction is to extract a small part of traces with high Signal-to-Noise Ratio (SNR) to reduce the complexity of attacks rather than to increase the success rates. Although strict theoretical proofs are given, the existing power trace extractors are too simple and leakage characteristics of Points-Of-Interest (POIs) have not been thoroughly analyzed. They only maximize the variance of data-dependent power consumption component and ignore the noise component, which results in very limited SNR that hampers the performance of extractors. In this paper, we provide a rigorous theoretical analysis of SNR of power traces, and propose a simple yet efficient SNR-centric extractor, named Shortest Distance First (SDF), to extract power traces with the smallest estimated noise by taking advantage of known plaintexts. In addition, to maximize the variance of the exploitable component while minimizing the noise, we refer to the SNR estimation model and propose another novel extractor named Maximizing Estimated SNR First (MESF). Finally, we further propose an advanced extractor called Mean-optimized MESF (MMESF) that exploits the mean power consumption of each plaintext byte value to more accurately and reasonably estimate the data-dependent power consumption of the corresponding samples. Experiments on both simulated power traces and measurements from an ATmega328p micro-controller demonstrate the superiority of our new extractors.

Index Terms—shortest distance first, SDF, MESF, signal-to-noise ratio, SNR, power trace extractor, side-channel attack

I. INTRODUCTION

SECRET information may leak from devices through side-channels such as electromagnetic radiation [2], cache patterns [21], acoustic [11] and power consumption [18] during the implementation of cryptographic algorithms. These leakages are usually obscure and cannot be easily discovered. By applying statistical analysis on assumed power consumption

of intermediate values and side-channel leakages, an attacker can recover sensitive information (e.g. encryption key) in the targeted devices. Side-channel attacks on power consumption channel, such as Differential Power Analysis (DPA) [18], Correlation Power Analysis (CPA) [4], Template Attacks (TA) [7] and Mutual Information Analysis [12], pose serious threats to the security of cryptographic implementations. To improve the attack efficiency, an attacker always tries his best to construct optimal distinguishers and profiles more accurate leakage models to better exploit leaky informations.

The Signal-to-Noise Ratio (SNR) becomes one of the key factors that affect the attack performance. It varies with algorithm implementation, hardware devices, sampling equipment, etc. For example, the probe may not be placed in a reasonable position for electromagnetic attacks or the implementation is highly parallel. In such cases, classic pre-processing schemes such as power traces alignment [34], averaging [24], [22], [31], higher-order cumulant [20], dimensionality reduction [30], [6] and Points-Of-Interest (POIs) extraction [8], are usually performed before side-channel attacks to “de-noise” [13] the traces. The attacker can obtain power traces with higher SNR after pre-processing, thus achieving higher attack performance. This paper considers the attack scenario where the attacker has enough power traces to guarantee successful attacks. We focus on another powerful pre-processing scheme that recognizes and extracts a part of power traces with high SNR from the original set to reduce the computing complexity and make the attacks more efficient. The related works will be given in the next section before introducing our contributions.

A. Related Works

Extracting power traces with high SNR to enhance attacks was first exploited by Kris et al. in [33]. The SNR of a time sample on power traces is the ratio of the variance of the exploitable power consumption component to the variance of noise (see Section II-A). The power consumption of a POI of the outputs of an S-box could be approximated by a normal distribution. Parts of samples were extracted from both tails of the distribution to enlarge the variance of the exploitable power consumption component [17]. Therefore, this scheme does not change the variance of noise. For simplicity, we call a power trace extraction scheme an extractor.

Noura et al. exploited a new leakage model to extract power traces to improve CPA in [26]. Hu et al. proposed an Adaptive Chosen Plaintext Correlation Power Analysis (ACPCPA) in [15]. They tried to solve the problem of discarding too many power traces in the extractor proposed by Yongade et al.

The research described in this paper has been supported in part by the National Research Foundation, Singapore, under grant number NRF2016NCR-NCR001-006 and the FinTech research project of Beijing Unionpay Card Technology Co., Ltd, China (Corresponding author: Kexin Qiao).

C. Ou and S.K. Lam are with Hardware and Embedded Systems Lab (H-ESL), School of Computer Science and Engineering, Nanyang Technological University, Singapore (email: chou@ntu.edu.sg, asklam@ntu.edu.sg).

D. Sun is with Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China (email: sundegang@iie.ac.cn).

X. Zhou is with Hardware Trustworthiness Technology & Engineering Lab, 2012 Laboratories, Huawei Technologies Co., Ltd (email: zhouxinping3@huawei.com).

K. Qiao is with Beijing Unionpay Card Technology Co., Ltd, China (email: qiaokexin@bctest.com).

Q. Wang is with School of Information and Communication Engineering, Beijing University of Posts and Telecommunication, Beijing 100876, China (email: wangqu@ict.ac.cn).

in [17]. Specifically, they analyzed the correlation between the Hamming weights of S-box outputs and power consumption of POIs, and concluded that Hamming weights 0, 1, 7 and 8 corresponded to power traces with high SNR. They selected the best two candidates of a sub-key with the largest correlation coefficients in CPA, and encrypted plaintexts with Hamming weights 0, 1, 7 and 8. Then, the newly acquired power traces were combined with the original power trace set independently to perform CPA. This extractor was similar to the one given by Haruki et al. in [29] and our improved extractor in [27], since they also aimed to enlarge the variance of the exploitable power consumption component rather than reduce the noise to improve SNR. The improvement in SNR with these extractors was very limited.

It is worth mentioning that it is difficult for the existing extractors to obtain a higher success rate [32] than the attack directly performed on the original power trace set. Therefore, we only consider the case where the number of power traces available to the attacker is sufficient to guarantee successful attacks, and the goal of extraction is to reduce the attack complexity rather than to increase the success rate. This applies to practical scenarios where the attacker has sufficient power traces to obtain a success rate of 1.00, but the samples are too noisy. As such, the attacker can extract a very small part of the traces to significantly reduce the computational complexity, while achieving the same success rate.

Recently, other schemes such as Principal Component Analysis (PCA) [23] were also exploited to extract power traces [16]. The above-mentioned extractors can accurately extract power traces with high SNR if the noise level is low. However, if the noise level is high, the distribution of power consumption is critically affected. This degrades the recognition accuracy of power traces with high SNR. Moreover, the variance of exploitable power consumption component that could be improved is also very limited compared with variance of noise in many cases. This indicates that it is not enough to increase SNR by enlarging the variance of exploitable power consumption component only. Accurately recognizing power traces with high SNR to improve the attack efficiency is still a very important and challenging issue.

B. Our Contributions

We aim to build more advanced extractors to recognize power traces with high SNR, thus enhancing their practical significance and enabling the launch of more efficient attacks. The main contributions of this paper are as follows:

- (i) We provide rigorous theoretical analysis of the SNR of power traces, and propose a novel extractor named Shortest Distance First (SDF) to extract the power traces with the smallest estimated noise rather than enlarge variance of exploitable power consumption. Samples with small noise occupy the majority of the whole signal population, thus providing ample opportunities to improve the SNR. We will show that SDF achieves significantly better performance than the existing extractors. The proposed method paves the way for new directions in power trace extraction.

- (ii) To maximize the exploitable component while minimize the estimated noise, we analyze the SNR model of the extracted samples. We further introduce the data-dependent power consumption component exploited in the existing extractors into SDF, and propose a more reasonable extractor called Maximizing Estimated SNR First (MESF) based on this model.
- (iii) Supported by theoretical analysis, we estimate the exploitable component by utilizing the mean power consumption of each plaintext byte value. This overcomes the ineffectiveness of directly estimating the exploitable component using samples with noise. Based on this scheme, we further build an extractor called Mean-optimized MESF (MMESF), which takes advantage of known plaintexts in the side-channel community.

Experiments show that our new extractors only need a very small part of the original power trace set (e.g., one-tenth) to achieve a similar success rate compared with the existing extractors. This demonstrates the superiority of our method in extracting power traces with high SNR.

C. Organization

The rest of this paper is organized as follows: Leakage characteristics of POIs, CPA and the existing power trace extractors, which we collectively denote as TAILS, are introduced in Section II. Our extractor SDF to minimize the estimated noise on power traces is described in Section III. To further maximize the variance of the data-dependent component, another two extractors MESF and MMESF are introduced in detail in Section IV. Experiments on both simulated power traces and an ATmega328p micro-controller are presented in Sections V and VI to illustrate the superiority of our new extractors. Finally, we conclude the paper in Section VII.

II. PRELIMINARIES

A. Leakage Characteristics of POIs

Let $x = x_0^j || x_1^j || \dots || x_{15}^j$ denote the j -th encrypted plain-text in AES-128 algorithm, $\kappa = \kappa_0 || \kappa_1 || \dots || \kappa_{15}$ denote the 16-byte key used in cryptographic device, $z_i^j = \text{Sbox}(x_i^j \oplus \kappa_i)$ denote the look-up table operation and $l(\tau)$ denote the corresponding sample leaked at time τ . The attacker encrypts n plaintexts $\mathcal{X} = (x^1, x^2, \dots, x^n)$ and acquires n power traces $\mathcal{L} = (l^1, l^2, \dots, l^n)$. According to [22], the power consumption of a single sample $l(\tau)$ can be modeled as the sum of an operation-dependent component $l_o(\tau)$, a data-dependent component $l_d(\tau)$, electronic noise $l_{el.n}(\tau)$, switching noise $l_{sw.n}(\tau)$, and the constant component $l_c(\tau)$:

$$l(\tau) = l_o(\tau) + l_d(\tau) + l_{el.n}(\tau) + l_{sw.n}(\tau) + l_c(\tau). \quad (1)$$

These 5 components are independent of each other, and the exploitable component $l_e(\tau)$ consists of the operation-dependent component $l_o(\tau)$ and data-dependent component $l_d(\tau)$:

$$l_e(\tau) = l_o(\tau) + l_d(\tau). \quad (2)$$

For a time sample $\mathcal{L}(\tau)$, the variance of the constant component $\sigma^2(\mathcal{L}_c(\tau)) = 0$. For classic DPA, the attacker only

considers one of the 8-bit intermediate values (i.e. the outputs of an S-box), the power consumption of the other 7 bits is switching noise (i.e. algorithm noise). The variance of switching noise here is larger than 0. For classic CPA considering all bits of intermediate values, the variance of switching noise $\sigma^2(\mathcal{L}_{sw.n}(\tau)) = 0$.

Let $\mathcal{L}_n(\tau)$ denote the noise component including $\mathcal{L}_{el.n}(\tau)$ and $\mathcal{L}_{sw.n}(\tau)$. The electronic noise is normal distributed. The SNR of the time sample $\mathcal{L}(\tau)$ is the ratio of variance of exploitable power consumption component $\mathcal{L}_e(\tau)$ to the variance of noise component. Thus, SNR can be simplified as:

$$\hat{\text{SNR}}(\tau) = \frac{\sigma^2(\mathcal{L}_e(\tau))}{\sigma^2(\mathcal{L}_n(\tau))}. \quad (3)$$

The same conclusion were drawn in Eq. 4.10 in [22]. For simplicity, the variances of 4 components $\sigma^2(\mathcal{L}_o(\tau))$, $\sigma^2(\mathcal{L}_d(\tau))$, $\sigma^2(\mathcal{L}_c(\tau))$ and $\sigma^2(\mathcal{L}_n(\tau))$ are expressed as $\sigma_o^2(\tau)$, $\sigma_d^2(\tau)$, $\sigma_c^2(\tau)$ and $\sigma_n^2(\tau)$.

This paper considers extracting the power traces with high SNR for more general differential power attacks [19] (e.g. CPA introduced in Section II-B). Each time sample is considered separately when performing extraction. Here we push the Eq. 3 a step further. Since the data-dependent component $\mathcal{L}_d(\tau)$ and operation-dependent component $\mathcal{L}_o(\tau)$ are independent, Eq. 3 can be further expressed as:

$$\hat{\text{SNR}}(\tau) = \frac{\sigma^2(\mathcal{L}_o(\tau)) + \sigma^2(\mathcal{L}_d(\tau))}{\sigma^2(\mathcal{L}_n(\tau))}. \quad (4)$$

For a time sample $\mathcal{L}(\tau)$, its operation on all power traces is usually the same. In this case, the variance of operation-dependent power consumption $\sigma^2(\mathcal{L}_o(\tau)) = 0$. SNR can be further simplified as:

$$\hat{\text{SNR}}(\tau) = \frac{\sigma^2(\mathcal{L}_d(\tau))}{\sigma^2(\mathcal{L}_n(\tau))} \quad (5)$$

(i.e., $\sigma_d^2(\tau) / \sigma_n^2(\tau)$).

B. Correlation Power Analysis

For all components of power consumption, $\sigma_d^2(\tau)$ is the only one correlating to the leakage model (e.g. Hamming weight model). Let $\mathcal{M}(\tau)$ denote the assumed leakage model of the intermediate values, the correlation coefficient between it and the total power consumption $\mathcal{L}(\tau)$ is:

$$\hat{\rho}(\mathcal{M}(\tau), \mathcal{L}(\tau)) = \frac{\text{cov}(\mathcal{M}(\tau), \mathcal{L}(\tau))}{\sigma(\mathcal{M}(\tau)) \cdot \sigma(\mathcal{L}(\tau))}. \quad (6)$$

“cov” here is the covariance matrix operator. Mangard et al. further analyzed the correlation between power consumption and $\mathcal{M}(\tau)$ in [22]. The important Formula 6.5 given in their paper can be expressed as:

$$\hat{\rho}(\mathcal{M}(\tau), \mathcal{L}(\tau)) = \frac{\hat{\rho}(\mathcal{M}(\tau), \mathcal{L}_d(\tau))}{\sqrt{1 + \frac{1}{\hat{\text{SNR}}(\tau)}}}. \quad (7)$$

Here $\hat{\rho}(\cdot, \cdot)$ is Pearson correlation coefficient operator. For a classic CPA attack, $\hat{\rho}(\mathcal{M}(\tau), \mathcal{L}_d(\tau))$ is a constant for a time sample, SNR determines the correlation coefficient $\hat{\rho}(\mathcal{M}(\tau), \mathcal{L}(\tau))$. Therefore, $\hat{\rho}$ approaches a constant when

the number of power traces exploited in an attack is large enough. It can also be seen from Eqs. 3 and 7 that SNR plays a very important role in attacks, and its improvement will bring a higher success rate to the attacker.

C. Existing Power Trace Extractors

Let R denote the Gaussian distributed noise component $\mathcal{L}_n$ with mean 0 and variance σ_n^2 , C denote the sum of constant component $\mathcal{L}_c$ and operation-dependent component $\mathcal{L}_o$. The leakage of the look-up table operation in AES-128 algorithm can be modeled as:

$$\mathcal{L} = \text{HW}(\mathcal{Z}) + C + R. \quad (8)$$

Here $\text{HW}(\cdot)$ is the Hamming weight function, and $\mathcal{Z} = (z^1, z^2, \dots, z^n)$ are intermediate values. We simulate 25, 600 power traces and exploit the existing extractors to extract 12, 800 from them. Here C in Eq. 8 is set to 100, noise level σ_n^2 is set to 0.09, thus the samples of each Hamming weight in $\text{HW}(\mathcal{Z})$ approximately follow normal distribution $\mathcal{N}(\text{HW}(z), 0.09)$. This small noise is convenient for readers to observe the power consumption distribution of each Hamming weight (see Fig. 1(a)).

The principles of classic power trace extractors in [17], [15], [27] are the same. They improve SNR by enlarging the variance of component $\mathcal{L}_d$. Here we define an evaluation function:

$$\mathcal{D}^i(\tau) = |l^i(\tau) - \bar{\mathcal{M}}(\tau)| \quad (9)$$

to quantify this. Here $l^i(\tau)$ denotes the i -th trace in $\mathcal{L}$ and $\bar{\mathcal{M}}(\tau)$ denotes the mean power consumption of all traces, which satisfies:

$$\bar{\mathcal{M}}(\tau) = \frac{1}{n} \sum_{i=1}^n l^i(\tau). \quad (10)$$

Here “ $|\cdot|$ ” denotes the absolute value operator. The power traces extracted from both two tails of the overall normal distribution are shown in Fig. 1(b). Due to the small noise, the leakage distribution of intermediate values can be well distinguished. For simplicity, we collectively name these power trace extractors as “TAILS”. The extracted values occupy four-fifths of the power consumption region in Fig. 1(a). They have the largest distances from the mean value $\bar{\mathcal{M}}(\tau)$ when extracting half of the samples. According to the variance of the estimated data-dependent component:

$$\sigma_d^2(\tau) = \frac{1}{n-1} \sum_{i=1}^n (l^i(\tau) - \bar{\mathcal{M}}(\tau))^2, \quad (11)$$

they are also with the largest $\sigma_d^2(\tau)$.

TAILS in [17] was based on distribution of power consumption, while the improved extractor given by Hu et al. in [15] was based on the distribution of Hamming weights. Since data-dependent component linearly depends on Hamming weights (i.e., $\mathcal{L}_d(\tau) \propto \text{HW}(\mathcal{Z})$) under very small noise, if these traces can be well recognized, the variance of data-dependent component $\sigma^2(\text{HW}(\mathcal{Z}))$ improves from 2.0078 to 10.3529 in leakage model provided by Eq. 8. It is worth noting that noisy samples corresponding to Hamming weights close to 4 also

Fig. 1. Exploiting TAILS to extract 12, 800 from 25, 600 power traces.

appear at both tails of the distribution if SNR is low. Similarly, samples corresponding to smaller or larger Hamming weights may also appear in the middle of the distribution. This will seriously affect the power trace extraction of TAILS. Obviously, Hamming weight based extractor is more accurate in this case. However, Hu et al. in [15] extracted the two best candidates with the highest correlation coefficients and exploited them to encrypt additional plaintexts for analysis. In other words, they introduced additional power traces with high SNR to improve the SNR of the combined power trace set, and is orthogonal to our original goal. Thus, we do not consider it in our paper.

III. EXTRACTORS TO MINIMIZE NOISE

The Eq. 5 indicates that there are two ways to improve SNR. One is to enlarge $\sigma_d^2(\tau)$ as we introduced in Section II-C, and the other is noise reduction. The existing works adopt the first approach. In this section, we adopt the second approach to improve SNR, and show that the new extractor is more superior. The samples with small noise are distributed near the average power consumption of the corresponding intermediate values. We will introduce this phenomenon in Section III-A. With the help of this phenomenon, we construct an optimal extractor named IDEAL based on the known key in Section III-B, which represents the best case of noise reduction and can be used for evaluations. The optimal extractor named Shortest Distance First (SDF) based on noise estimation in the attack scenario where the key is unknown, will be described in Section III-C.

A. Noise of Samples

Noise plays a very important role in side-channel attacks. The samples with small noise are distributed near the average power consumption of the corresponding intermediate values. Taking two normal distributions with mean $\mu = 104$ and variance $\sigma_n^2 = 1, 0.1$ shown in Fig. 2 as an example, most samples in the distributions lie close to their means μ -s. Since

the samples with the largest variance are at both tails, if the same number of samples are extracted from the middle of distributions, the samples corresponding to $\mathcal{N}(104, 0.1)$ have a smaller variance.

Fig. 2. Two time samples following normal distributions $\mathcal{N}(104, 1)$ and $\mathcal{N}(104, 0.1)$.

We will provide an intuitive example using the 25, 600 samples in our simulation with $\sigma_n^2 = 0.09$. We rank the samples according to their noise component, and select 12, 800 samples with the minimum noise from them (as shown in Fig. 3). These time samples are very close to the mean power consumption of their corresponding Hamming weights. There are many more samples with low noise in the middle than two tails. Therefore, noise can be significantly reduced by extracting samples close to their mean values. Moreover, reducing σ_n^2 has more potential to improve SNR than enlarging σ_d^2 . The SNR of the power traces extracted by this is increased by more than 7 times, compared with less than 2 times in TAILS. The noise of the extracted power traces can be infinitely small in theory if there are enough samples to extract. This will be explained in Section IV-C.

Fig. 3. Power traces extracted from the middle of each Hamming weight distribution.

B. Optimal Extractor

According to Section III-A, we can rank the power traces according to their noise component, and extract a part with the smallest noise from them. In this case, these samples are with the highest SNR if we only consider the noise component $\mathcal{L}_n(\tau)$. We name this extractor IDEAL. To achieve this optimal extraction, it requires the knowledge of key and is only exploitable as an evaluation tool. Therefore, IDEAL can be a profiled extractor. In other words, we can profile power consumption templates for 9 Hamming weights corresponding to intermediate values. We randomly generate 100 samples from the leakage model provided in Eq. 8 with $\sigma_n^2 = 30$, and extract parts of samples with the smallest noise from them. The existing works only exploited the result of a single CPA attack to compare the correlation coefficients of different sub-key candidates, which were not accurate enough. Since the correlation coefficient of each candidate under different repetitions may be very different, we exploit the average correlation coefficients under 200 repetitions to measure their variations. The experimental results under different number of extracted power traces are shown in Fig. 4(a). The corresponding results of CPA are also given. IDEAL achieves significantly higher SNR than random extraction.

Fig. 4. Correlation coefficients under random extraction and optimal extractor IDEAL.

The noise of the samples near the Hamming weight templates is very small. The correlation coefficient will be very high if we perform CPA on these traces. The one corresponding to the correct sub-key in CPA with random power trace extraction is significantly higher than others, but still much lower than the one corresponding to samples with the smallest noise (the upper two curves in Fig. 4(a)). The correlation coefficient of IDEAL (red curve) decreases gradually with the increasing number of power traces extracted. Finally, it is consistent with the one corresponding to classic CPA when exploiting all traces. In this case, the SNR of the extracted samples monotonically decreases (as shown in Fig. 5(a)).

Fig. 5. SNR and success rate under random extraction and optimal extractor IDEAL.

With the knowledge of key, the noise of power traces can be accurately estimated according to the Hamming weight templates, and the ones with the highest SNR can be extracted. In this case, the optimal extractor IDEAL can be a profiled attack and these means can be exploited from the templates to achieve higher accuracy. However, this is impossible in attacks as we have explained before since as long as a sample is close enough to a “template” under the case of unknown key, it will be extracted to launch attacks even if it is wrongly matched. In other words, each guessing sub-key will “frantically” extracts the samples that are most beneficial to itself in attacks. This will result in very high linear correlation between the means (or templates) and the power traces they extract, and obtaining correlation coefficients even significantly higher than the one corresponding to the correct sub-key under the known key case (as shown in Fig. 4(b)). Thus, it is impossible to distinguish the correct key.

The success rate may be even worse than that of random extraction if IDEAL is exploited to extract power traces for each candidate in attacks (as shown in Fig. 5(b)). Although it is impossible to build such a good extractor in reality, IDEAL tells us that with more accurate SNR estimation, success rate can be significantly improved. In other words, it provides a reference for us to build a highly accurate extractor. In the next section, we will explore how to build such an efficient extractor for attacks (i.e. without knowing the key) and make its performance close to IDEAL in the known-key case.

C. Shortest Distance First Extractor

The existing power trace extractors only consider the case where the number of power traces available to the attackers is sufficient to guarantee successful attacks, and the goal of their extraction is to reduce the attack complexity rather than

to increase the success rate. As explained earlier, this is due to the difficulty in achieving a success rate higher than the attack directly performed on the original power trace set. We have also explained in Section III-B that if we exploit profiled Hamming-weight templates to estimate the noise component of samples and extract a part that best match templates from them, the correlation coefficients of all the candidates are very high, so that profiled attacks cannot be exploited. Here we focus on the most common implicit assumption of known plaintexts in side-channel community, and take the second-best to compute the mean power consumption of each plaintext byte value for power trace extraction. In this case, although these means may not be so accurately estimated when the power traces are limited, they are a good reference.

Our Shortest Distance First (SDF) extractor is very simple (see Algorithm 1). Suppose that we employ it to extract n' power traces with the minimum estimated noise from the n traces $\mathcal{L}(\tau) = (l^1(\tau), l^2(\tau), \dots, l^n(\tau))$ leaked at time τ . SDF keeps a counter $\mathcal{C}$ to record the current number of power traces of 256 plain-text byte values (Step 2). The corresponding power consumption $l^i(\tau)$ is added to $\mathcal{M}^{x^i}(\tau)$ (Step 3). Here $\mathcal{M}^{x^i}(\tau)$ denotes the mean power consumption of plaintext byte value x^i . We then average $\mathcal{M}(\tau)$ and get the mean power consumption vector of all 256 plain-text byte values after traversing all power traces (Step 5).

Here, we need to clarify the difference between profiled and non-profiled attacks. The former means that the attacker has the same device as the targeted one. He can collect as many power traces as he wishes to profile sufficiently accurate templates and exploits them to launch attacks on the targeted device. The classic profiled attack is TA as introduced in Section I. Non-profiled attacks such as classic CPA and DPA do not have such a profiling stage. The power traces are employed to not only estimate the mean power consumption but also launch attacks in SDF. We can also see that the same power traces from $\mathcal{L}(\tau)$ are employed in Steps 3 and 7 of Algorithm 1. In other words, they are all from the targeted device, and no additional power traces are introduced when estimating the means. In this case, SDF is still a non-profiled attack.

Algorithm 1: Shortest Distance First (SDF) extractor.

Input: Power traces $\mathcal{L}(\tau)$, plain-texts $\mathcal{X}$ and the number of traces to be extracted n' .
Output: The new power trace set $\mathcal{L}'(\tau)$ and their plain-texts $\mathcal{X}'$.

```

1 for  $i$  from 1 to  $n$  do
2    $\mathcal{C}^{x^i} \leftarrow \mathcal{C}^{x^i} + 1$ ;
3    $\mathcal{M}^{x^i}(\tau) \leftarrow \mathcal{M}^{x^i}(\tau) + l^i(\tau)$ ;
4 end
5  $\mathcal{M}(\tau) \leftarrow \mathcal{M}(\tau) / \mathcal{C}$ ;
6 for  $i$  from 1 to  $n$  do
7    $\mathcal{D}^i(\tau) \leftarrow |l^i(\tau) - \mathcal{M}^{x^i}(\tau)|$ ;
8 end
9  $(\mathcal{T}(\tau), \mathcal{P}) \leftarrow \text{AscendSort}(\mathcal{L}(\tau), \mathcal{X}, \mathcal{D}(\tau))$ ;
10  $\mathcal{L}'(\tau) = \mathcal{T}^{1 \dots n'}(\tau); \mathcal{X}' = \mathcal{P}^{1 \dots n'}$ ;

```

The noise component of a sample $l^i(\tau)$ ($1 \leq i \leq n$) in power trace set $\mathcal{L}$ can be estimated as:

$$\mathcal{D}^i(\tau) = |l^i(\tau) - \mathcal{M}^{x^i}(\tau)| \quad (12)$$

rather than using the mean $\bar{\mathcal{M}}(\tau)$ of all samples in Eq. 9 (see Steps 6 ~ 8). Here vector $\mathcal{D}$ is employed to save the distances. We sort the estimated noise components in ascending order and get the corresponding re-ranked power trace set $\mathcal{T}(\tau)$ and plaintext byte values $\mathcal{P}$ (Step 9). Finally, n' traces with the smallest distances (i.e. the smallest estimated noise component) and their corresponding plaintexts are extracted from the first n' elements of $\mathcal{T}(\tau)$ and $\mathcal{P}$ (Step 10). Let $\bar{\mathcal{T}}^{hw_{x^i}}$ denote the template corresponding to the Hamming weight of plaintext byte value $x^i \in \{0, 1, \dots, 255\}$ ($1 \leq i \leq n$). The estimated SNR of the extracted power traces $l^i(\tau)$ in $\mathcal{L}'(\tau)$ can be expressed as:

$$\hat{\text{SNR}}(\tau) = \frac{\sum_{i=1}^{n'} (\bar{\mathcal{T}}^{hw_{x^i}}(\tau) - \bar{\mathcal{M}}(\tau))^2}{\sum_{i=1}^{n'} (l^i(\tau) - \bar{\mathcal{T}}^{hw_{x^i}}(\tau))^2}, \quad (13)$$

since SNR estimation can exploit the knowledge of key and the well profiled templates. Taking the Hamming weight model given in Eq. 8 as an example, the templates can use 0, 1, ..., 7 and 8 directly, since infinite power traces can be exploited to profile sufficiently accurate templates in theory. However, the leakage model is unknown in the real leakage, so we can only employ a large number of power traces in profiling to make the templates as accurate as possible. More advanced profiling technology like leakage certification [5], [9] can also be exploited in this case.

It is worth mentioning that the extraction is performed on all rather than only one time sample of power traces, and its target is to launch more efficient attacks. In other words, Algorithm 1 is implemented on each time sample. The same extractor may extract very different power traces on different time samples. However, this does not contradict the purpose of the extractors, since they aim to maximize the SNR of the power traces extracted from each time sample to achieve the optimal attack performance.

IV. EXTRACTORS TO MAXIMIZE ESTIMATED SNR

We extracted power traces with the minimum noise component to improve SNR in Section III. In this section, we further attempt to enlarge $\sigma^2(\mathcal{L}_d(\tau))$ of the extracted power traces on the basis of the noise reduction. SNR estimation will be more accurate in this case (see Eq. 5).

A. SNR Maximization

The variance of data-dependent power consumption and noise satisfy $\sigma_d^2(\tau) = \frac{1}{n-1} \sum_{i=1}^n (l^i(\tau) - \bar{\mathcal{M}}(\tau))^2$ and $\sigma_n^2(\tau) = \frac{1}{n-1} \sum_{i=1}^n (l^i(\tau) - \mathcal{M}^{x^i}(\tau))^2$, respectively. We can quantify the data-dependent power consumption of a sample $l^i(\tau)$ using Eq. 9. In other words, to maximize $\sigma_d^2(\tau)$, we only need to maximize evaluation function $\mathcal{D}^i(\tau)$ provided by this formula. This corresponds to TAILS that extracts samples from two tails of the total distribution introduced in

Section II-C. Similarly, to minimize $\sigma_n^2(\tau)$, we can minimize evaluation function $\mathcal{D}^i(\tau)$ of SDF in Eq. 12 for each extracted sample.

We aim to maximize the SNR here, that is, to minimize the variance of noise while maximizing the variance of the data-dependent component (see Eq. 5). However, we only extract one sample at each time and we cannot exploit the formula of SNR as the evaluation function given in Step 7 of Algorithm 1. Therefore, we define the evaluation function as the ratio of two absolute values:

$$\mathcal{D}^i(\tau) = \frac{|l^i(\tau) - \bar{\mathcal{M}}(\tau)|}{|l^i(\tau) - \mathcal{M}^{x^i}(\tau)|}. \quad (14)$$

In other words, we take the second-best balance strategy and transform the SNR maximization to maximize the evaluation function in Eq. 14. Although the local optimum of the evaluation function does not mean that the global SNR is optimum and this balance strategy is not perfect, it is a very good one. We name this new power trace extractor as Maximizing Estimated SNR First (MESF).

B. Mean-Optimized MESF

This paper considers the case that the number of power traces is enough to guarantee successful attack. On one hand, directly exploiting $l^i(\tau)$ to estimate data-dependent power consumption is ineffective, as this will introduce significant but unnecessary estimation error caused by noise. On the other hand, the attacker may not have enough power traces to accurately measure mean power consumption for each plaintext byte value. In this case, the accuracy of estimation of the data-dependent component and noise component will also be significantly affected. This estimation error increases in the case of very large noise. However, compared with $l^i(\tau)$, the mean power consumption of each plaintext byte value is still more accurate and theoretically more reasonable in measuring data-dependent power consumption.

Overall, a better strategy is to exploit the mean power consumption of each plaintext byte value when measuring their data-dependent component. In this case, Eq. 14 can be further optimized as:

$$\mathcal{D}^i(\tau) = \frac{|\mathcal{M}^{x^i}(\tau) - \bar{\mathcal{M}}(\tau)|}{|l^i(\tau) - \mathcal{M}^{x^i}(\tau)|}. \quad (15)$$

We name this new power trace extractor as Mean-optimized MESF (MMESF). Since the extractors TAILS, SDF, MESF and MMESF rely on different mathematical principles, their accuracy may vary with a different number of power traces extracted and different noise levels, which will be demonstrated in our experiments.

C. Comparisons

Our three extractors SDF, MESF and MMESF lead to large differences in the extracted power traces. Their results of extracting 12, 800 from 25, 600 power traces in Fig. 1 are shown in Fig. 6. The SNR of the 25, 600 samples generated by the leakage model in Eq. 8 is estimated to be 22.0781 when

the noise level is $\sigma_n^2 = 0.09$, approaching the theoretical value 22.3049. SDF is designed to minimize the estimated noise component, and its SNR is about 146.9011, approaching the upper limit 155.8411 of the optimal extractor IDEAL under known key. It can also be seen from Figs. 3 and 6 that the traces extracted by them are similar to the optimally extracted ones, which fully shows their advantages under small noise.

Fig. 6. The samples extracted by different extractors.

The power traces extracted by MESF and MMESF are quite different from those extracted by SDF. The distribution of samples extracted by MESF appears to be small in the middle and large at both two ends. These samples are not concentrated at both two tails like TAILS, since MESF benefits from the advantages of both SDF and TAILS. The SNR of the extracted samples corresponding to MESF is 40.8126. Compared with MESF, MMESF exploits the mean power consumption of each plaintext byte value to measure the data-dependent component, which is more accurate and more reasonable in theory. Both MESF and MMESF benefit from the data-dependent power consumption component, but MMESF benefits more, and the samples it extracts mainly distribute at both two tails. Moreover, these samples are initially concentrated on the mean power consumption of Hamming weights and then spread to both ends. The SNR of the samples extracted by MMESF is 88.5545, which is more than double that of MESF.

The SNR of the samples extracted by the above three extractors is significantly higher than that of TAILS, which is only about 38.6273 in this experiment. This shows that they have effectively recognized the power traces with high SNR in different degrees, thus improving the success rates. MESF and MMESF achieve lower SNR than SDF, but this does not mean that they are worse than SDF. These extractors will be

further discussed in Sections V and VI.

V. SIMULATIONS

To compare the performance of extractors under different noise levels, our first experiment is performed on simulated power traces. They are randomly generated from the leakage model provided in Eq. 8. We first compare the performance of extractors under different number of power traces, the noise level σ_n^2 is set to 400. Due to the large noise, too many traces are required for attacks, which makes the evaluation time-consuming. Therefore, we only repeat the above experiments 200 times, and the corresponding experimental results are shown in Fig. 7. To guarantee a stable success rate (SR) about 1.00 for classic CPA, nearly 6 000 traces are required in the original sets. All extractors TAILS, SDF, MESF and MMESF achieve higher success rates than random extraction and are even close to the ideal extraction in Figs. 7(b) and 7(c), which correspond to the case that the attacker has sufficient traces. SDF and MMESF achieve success rates 1.00 even under the poor condition that only 300 traces are extracted, which fully demonstrates their superiority.

The SNR corresponding to MESF and TAILS is the lowest in Fig. 7, and only a tiny change occurs when extracting more power traces. They are designed to maximize the variance of the data-dependent power consumption component rather than reduce the noise on the samples. The estimated SNR of power traces extracted by SDF and MMESF decreases monotonically. Both success rate and the estimated SNR are higher when the original sets have more power traces (see Figs. 7(a), 7(b) and 7(c)). This indicates: 1) they can still recognize the power traces with high SNR even in the case of very large noise, and 2) SNR has become an important factor determining the success rate. The performance of our extractors SDF and MMESF is also more stable, and the areas with high success rates are significantly wider than that of TAILS. This makes it much easier for attackers to determine the number of power traces to extract.

Noise has a great influence on the estimation of mean power consumption of each plaintext byte value, and reduces the accuracy of power trace extraction. Therefore, it is necessary to further discuss its impact on the success rate. In this case, we also consider the success rate and the estimated SNR of the extracted power traces under different noise levels $\sigma_n^2 = 350$, $\sigma_n^2 = 300$ and $\sigma_n^2 = 250$. The corresponding experimental results are shown in Fig. 8. Here, 6000 power traces are randomly generated for extraction, and each experiment is repeated 200 times. With the decrease of noise level, both success rates and the accuracy of all extractors increase. SDF and MMESF still achieve the highest success rates. Although the success rates of TAILS and MESF do not reach such a height when they extract a small number of samples, they increase rapidly when more power traces are extracted. MESF and TAILS not only change the SNR, but also increase the $\hat{\rho}(\mathcal{M}(\tau), \mathcal{L}_d(\tau))$ in Eq. 7. Their success rates benefit from at least these two factors. It is worth mentioning that the relationship between success rate and the noise is much more complex than this, and more details can be seen in [10], [37].

The success rate of classic CPA without extraction is about 1.00 under $\sigma_n^2 = 250$ when randomly generating 4000 power traces to perform attacks. SDF and MMESF achieve success rates about 1.00 even when only extracting 400 traces. If we launch the attacks on more traces, the success rates will be stable, hence guaranteeing the key recovery. However, the extractors significantly reduce the number of traces used for attacks thus significantly reducing the computational complexity and highlighting their significance.

Fig. 9. Estimated SNR of the traces extracted by IDEAL under different number of power traces and different noise levels in simulations.

SNR of the traces extracted by IDEAL under different number of power traces and different noise levels is also shown in Fig. 9. We can obtain higher SNR if extracting the same number of samples from a larger power trace set. On one hand, more traces make the estimation of mean power consumption more accurate. On the other hand, the number of traces with small noise also increases. The decrease of noise level from $\sigma_n^2 = 400$ to $\sigma_n^2 = 350$, $\sigma_n^2 = 300$ and $\sigma_n^2 = 250$ also makes the mean power consumption more accurate under fewer power traces. Unlike the results given in Section IV-C, IDEAL achieves significantly higher SNR than all other extractors. However, SDF and MMESF achieve success rates close to it. They are more accurate compared with other extractors.

VI. EXPERIMENTS ON ATMEGA328P MICRO-CONTROLLER

Our second experiment is performed on an ATmega328p micro-controller. We first consider the unprotected AES-128 implementation provided by [1], then another AES-128 protected by the widely used Rotated S-boxes Masking (RSM) in [25]. The clock operating frequency of this micro-controller is 16 MHz. We exploit a WaveRunner 8104 oscilloscope to acquire the power traces. The sampling rate is set to 1 GS/s.

A. Unprotected Case

We first consider the unprotected case, and encrypt 300, 000 random plaintexts. We then perform classic CPA on 10, 000 power traces to extract a time sample with SNR about 0.0016 and correlation coefficient about 0.09399 rather than the best POIs of the first S-box in the first round to simulate

Fig. 7. Success rates and estimated SNR (of the extracted traces) under different number of power traces in simulations.

Fig. 8. Success rates and estimated SNR (of the extracted traces) under different noise levels in simulations.

the very low SNR scenario. Actually, we will achieve better performance if extractions are directly performed on better POIs. Although the components of power consumption are more complex than those simulated in actual attacks, we can still draw conclusions similar to those in Section V (as shown in Fig. 10). Moreover, all extractors still have monotonically decreasing SNR as shown in Fig. 7 and 8.

To estimate the SNR more accurately, we exploit 1 000 pow-

er traces acquired to estimate the mean power consumption of each Hamming weight. We then randomly select 4, 000, 6, 000 and 8, 000 power traces from them for extraction. This operation is repeated 200 times, and the corresponding success rates are about 0.990, 1.00 and 1.00. Fig. 10(a) corresponds to the case of insufficient power traces, and Figs. 10(b) and 10(c) correspond to the case of sufficient power traces. Extracting a small number of power traces can significantly improve the

Fig. 10. Success rates and estimated SNR (of the extracted traces) under different number of traces (unprotected case).

SNR. Almost all extractors achieve a success rate of 1.00 when extracting 400 power traces. Therefore, they significantly reduce the complexity compared with the attacks directly performed on the original power trace set.

The experimental results clearly show that all extractors achieve success rates that are significantly higher than random extraction. Our SDF almost achieves the same success rates compared with TAILS, while MMESF achieves success rate even higher than IDEAL. This illustrates the accuracy of MMESF to exploit the mean power consumption of each plaintext byte value when estimating the data-dependent component. MMESF outperforms SDF because of this component. These means $\mathcal{M}^{x^i} (x^i = 0, 1, \dots, 255)$ are more accurately estimated with the increasing number of power traces, which improves the extraction performance. They can extract better power traces than other extractors and achieve the highest success rates in most cases. Similar conclusions can also be drawn from Section V.

With respect to SNR, SDF is superior to MMESF. This indicates that it still accurately extracts the power traces with high SNR even under very large noise. However, SDF ignores the data-dependent power consumption component, thus making its $\hat{\rho}(\mathcal{M}(\tau), \mathcal{L}_d(\tau))$ smaller than MMESF's. Therefore, it achieves a lower success rate than MMESF. The SNR of MESF and TAILS is always much lower than that of MMESF. It is worth noting that the SNR of MMESF and SDF decreases when more traces are extracted, contrary to the SNR of MESF and TAILS, although this growth is not obvious and is difficult to observe. Similar conclusions can also be drawn from Figs. 7 and 8. On one hand, compared with the extraction based on enlarging the data-dependent power consumption component, reducing the noise can improve the SNR of the extracted power traces to a greater extent. On the other hand, this illustrates that SNR is only one of the

important factors to determine the success rate. For example, the SNR improves with the increasing number of power traces in the original set. This is also an important factor.

Fig. 11. Estimated SNR of the traces extracted by IDEAL under different number of power traces and different noise levels (unprotected case).

SNR of traces extracted from the original set with 4, 000, 6, 000 and 8, 000 traces using IDEAL is shown in Fig. 11. Compared with the simulated power traces, the composition of real leakage is much more complicated. The extractors achieve SNR about 10 ~ 15 times lower than the ideal extraction (as shown in Figs. 10 and 11). This gap decreases rapidly with a larger proportion of power traces extracted. In terms of success rate, these extractors have achieved outstanding results, but there is still a lot of room for improvement.

B. Protected Case

Next, we consider the protected case, masked implementation [28], which follows the theory of secret sharing. Let d denote a positive integer, I denote the encoded intermediate

value (e.g. the output of S-box) and “ $\circ$ ” denote the XOR operation over a finite field. The d^{th} -order masking of I is a $(d + 1)$ -tuple $(I_i)_{i \in \{0,1,\dots,d\}}$ satisfying $I_0 \circ I_1 \circ \dots \circ I_d = I$. Here we implement an AES-128 protected by RSM. We exploit the classic second-order CPA, traverse all combinations between any two time samples and compute the correlation coefficients between their mean-removed products and the Hamming weights of the intermediate values. Two time samples corresponding to two shared values are obtained.

We encrypt 400, 000 random plaintexts and exploit 1 000 power traces acquired to estimate the mean power consumption of each plaintext byte value. We then randomly select 10, 000, 20, 000 and 30, 000 power traces from them for extraction. This operation is repeated 200 times, and their corresponding success rates are about 0.550, 0.895 and 1.00. Success rates and estimated SNR when extracting different number of traces from them are given in Fig. 12. Although the mask will bring large estimation errors to the SNR of the extracted power traces, the performance of all extractors is significantly better than that of random extraction. The success rate of TAILS is even higher than that of the attack directly performed on the original power trace set (as shown in Fig. 12(a)). With more power traces being used for extraction, its success rate is gradually surpassed by MMESF’s (see Figs. 12(b) and 12(c)).

Fig. 13. Estimated SNR of the traces extracted by IDEAL under different number of power traces (protected case).

In terms of the estimated SNR, SDF is the highest, MMESF is the second, and they are always significantly higher than TAILS and MESF, which fully illustrates their superiority (as shown in Fig. 12). Estimated SNR under different number of power traces for IDEAL’s extraction is shown in Fig. 13, which is still much higher than other extractors. However, IDEAL never succeeds since its noise model is not suitable for this scenario.

VII. CONCLUSIONS

Power trace extractors can reduce the computational complexity of attacks when the power traces are sufficient. The

existing extractors simply extract samples with the largest estimated variance of exploitable power consumption to improve the SNR. Although strict theoretical proofs are given, they are too simple and leakage characteristics of POIs have not been thoroughly analyzed. These limit their accuracy on power trace extraction. In this paper, we provide rigorous theoretical analysis of the SNR of power traces, and propose a novel extractor named SDF to extract the power traces with the smallest estimated noise. SDF can achieve SNR and success rate that are significantly higher than the existing extractors. Thus, it has wider application scenarios and provides a new avenue for power trace extraction.

To maximize the data-dependent component while minimizing the noise, we further propose a novel extractor named MESF. TAILS directly exploits the distance between single sample and population mean as an estimation of its data-dependent component. As this measurement is noisy, we further exploit the mean power consumption of each plaintext byte value to estimate the data-dependent power consumption of the corresponding samples, and propose a more advanced extractor named MMESF, which is also more reasonable in theory. The above three extractors lead to higher extraction accuracy and significantly reduce the computational complexity compared with the existing extractors.

All estimates in this paper are based on the mean power consumption of plaintext byte values. This limits our extractors to scenarios where the attacker has a large number of power traces to estimate the means. We leave the introduction of profiled attacks into extractors as an open problem and believe this will significantly improve their performance. In our future work, we plan to theoretically analyze the relationship between success rate and the noise given in [10], [37], and introduce it into the extractors to enhance power trace extraction. In addition, the extractors for symmetric ciphers mentioned in this paper require the knowledge of plaintexts. Recently, several high-efficient attacks on asymmetric ciphers like RSA have been presented in [14], [3], [36]. Wan et al. in [35] even proposed a cluster CPA against double blinding exponentiation (i.e. without knowing the plaintexts and key) of RSA, which could still exploit k -means clustering to reduce noise. These important works motivate us to apply our experience in building power trace extractors for the unknown-plaintext case in our future work.

REFERENCES

- [1] AVR-Crypto-Lib. <https://github.com/DavyLandman/AESLib>.
- [2] D. Agrawal, B. Archambeault, J. R. Rao, and P. Rohatgi. The EM Side-Channel(s). In *Cryptographic Hardware and Embedded Systems - CHES 2002, 4th International Workshop, Redwood Shores, CA, USA, August 13-15, 2002, Revised Papers*, pages 29–45, 2002.
- [3] F. Amiel, B. Feix, and K. Villegas. Power Analysis for Secret Recovering and Reverse Engineering of Public Key Algorithms. In C. M. Adams, A. Miri, and M. J. Wiener, editors, *Selected Areas in Cryptography, 14th International Workshop, SAC 2007, Ottawa, Canada, August 16-17, 2007, Revised Selected Papers*, volume 4876 of *Lecture Notes in Computer Science*, pages 110–125. Springer, 2007.
- [4] 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.

Fig. 12. Success rates and estimated SNR (of the extracted traces) under different number of traces (protected case).

- [5] O. Bronchain, J. M. Hendrickx, C. Massart, A. Olshevsky, and F. Standaert. Leakage Certification Revisited: Bounding Model Errors in Side-Channel Security Evaluations. In *Advances in Cryptology - CRYPTO 2019 - 39th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2019, Proceedings, Part I*, pages 713–737, 2019.
- [6] E. Cagli, C. Dumas, and E. Prouff. Kernel Discriminant Analysis for Information Extraction in the Presence of Masking. In *Smart Card Research and Advanced Applications - 15th International Conference, CARDIS 2016, Cannes, France, November 7-9, 2016, Revised Selected Papers*, pages 1–22, 2016.
- [7] 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.
- [8] 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.
- [9] 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.
- [10] Y. Fei, A. A. Ding, J. Lao, and L. Zhang. A Statistics-Based Success Rate Model for DPA and CPA. *J. Cryptographic Engineering*, 5(4):227–243, 2015.
- [11] D. Genkin, A. Shamir, and E. Tromer. RSA Key Extraction via Low-Bandwidth Acoustic Cryptanalysis. In *Advances in Cryptology - CRYPTO 2014 - 34th Annual Cryptology Conference, Santa Barbara, CA, USA, August 17-21, 2014, Proceedings, Part I*, pages 444–461, 2014.
- [12] B. Gierlichs, L. Batina, P. Tuyls, and B. Preneel. Mutual Information Analysis. In *Cryptographic Hardware and Embedded Systems - CHES 2008, 10th International Workshop, Washington, D.C., USA, August 10-13, 2008. Proceedings*, pages 426–442, 2008.
- [13] 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.
- [14] N. Homma, A. Miyamoto, T. Aoki, A. Satoh, and A. Shamir. Comparative Power Analysis of Modular Exponentiation Algorithms. *IEEE Trans. Computers*, 59(6):795–807, 2010.
- [15] W. Hu, L. Wu, A. Wang, X. Xie, Z. Zhu, and S. Luo. Adaptive Chosen-Plaintext Correlation Power Analysis. In *Tenth International Conference on Computational Intelligence and Security, CIS 2014, Kunming, Yunnan, China, November 15-16, 2014*, pages 494–498, 2014.
- [16] Y. Kim and H. Ko. Using Principal Component Analysis for Practical Biasing of Power Traces to Improve Power Analysis Attacks. In *Information Security and Cryptology - ICISC 2013 - 16th International Conference, Seoul, Korea, November 27-29, 2013, Revised Selected Papers*, pages 109–120, 2013.
- [17] Y. Kim, T. Sugawara, N. Homma, T. Aoki, and A. Satoh. Biasing Power Traces to Improve Correlation Power Analysis Attacks. In *Constructive Side-Channel Analysis and Secure Design - COSADE 2010 - First International Workshop, Daemstadt, Germany, February 4-5, 2010*, pages 77–80, 2010.
- [18] 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.
- [19] P. C. Kocher, J. Jaffe, B. Jun, and P. Rohatgi. Introduction to Differential Power Analysis. *J. Cryptographic Engineering*, 1(1):5–27, 2011.
- [20] T. Le, J. Clédière, C. Servière, and J. Lacoume. Noise Reduction in Side Channel Attack Using Fourth-Order Cumulant. *IEEE Trans. Information Forensics and Security*, 2(4):710–720, 2007.
- [21] F. Liu, Y. Yarom, Q. Ge, G. Heiser, and R. B. Lee. Last-Level Cache Side-Channel Attacks are Practical. In *2015 IEEE Symposium on Security and Privacy, SP 2015, San Jose, CA, USA, May 17-21, 2015*, pages 605–622, 2015.
- [22] S. Mangard, E. Oswald, and T. Popp. *Power Analysis Attacks - Revealing the Secrets of Smart Cards*. Springer, 2007.
- [23] D. Mavroeidis, L. Batina, T. van Laarhoven, and E. Marchiori. PCA, Eigenvector Localization and Clustering for Side-Channel Attacks on Cryptographic Hardware Devices. In *Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD 2012, Bristol, UK, September 24-28, 2012. Proceedings, Part I*, pages 253–268, 2012.
- [24] T. S. Messerges, E. A. Dabbish, and R. H. Sloan. Examining Smart-Card Security under the Threat of Power Analysis Attacks. *IEEE Trans. Computers*, 51(5):541–552, 2002.
- [25] M. Nassar, Y. Souissi, S. Guilley, and J. Danger. RSM: A Small and Fast Countermeasure for AES, Secure Against 1st and 2nd-Order Zero-Offset SCAs. In *2012 Design, Automation & Test in Europe Conference*

& Exhibition, DATE 2012, Dresden, Germany, March 12-16, 2012, pages 1173–1178, 2012.

- [26] B. Noura, M. Mohsen, and T. Rached. Optimized Power Trace Numbers in CPA Attacks. In *The 8-th International Multi-Conference on Systems, Signals and Devices, Sousse, Tunisia, March 22-25, 2011. Proceedings*, pages 1–5, 2011.
- [27] C. Ou, Z. Wang, D. Sun, X. Zhou, J. Ai, and N. Pang. Enhanced Correlation Power Analysis by Biasing Power Traces. In *Information Security - 19th International Conference, ISC 2016, Honolulu, HI, USA, September 3-6, 2016. Proceedings*, pages 59–72, 2016.
- [28] E. Prouff and M. Rivain. Masking against Side-Channel Attacks: A Formal Security Proof. In *Advances in Cryptology - EUROCRYPT 2013, 32nd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Athens, Greece, May 26-30, 2013. Proceedings*, pages 142–159, 2013.
- [29] H. Shimada, Y.-i. Hayashi, N. Homma, T. Mizuki, T. Aoki, and H. Sone. Using Selected-Plaintext Sets for Efficient Evaluation of EM Information Leakage from Cryptographic Devices. In *2012 Proceedings of SICE Annual Conference, SICE 2014, Akita University, Akita, Japan, August 20-23, 2012. Proceedings*, pages 64–67. IEEE, 2012.
- [30] Y. Souissi, M. Nassar, S. Guilley, J. Danger, and F. Flament. First Principal Components Analysis: A New Side Channel Distinguisher. In *Information Security and Cryptology - ICISC 2010 - 13th International Conference, Seoul, Korea, December 1-3, 2010, Revised Selected Papers*, pages 407–419, 2010.
- [31] F. Standaert. How (Not) to Use Welch's T-Test in Side-Channel Security Evaluations. In *Smart Card Research and Advanced Applications, 17th International Conference, CARDIS 2018, Montpellier, France, November 12-14, 2018, Revised Selected Papers*, pages 65–79. Springer, 2018.
- [32] 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.
- [33] K. Tiri and P. Schaumont. Changing the Odds Against Masked Logic. In *Selected Areas in Cryptography, 13th International Workshop, SAC 2006, Montreal, Canada, August 17-18, 2006 Revised Selected Papers*, pages 134–146, 2006.
- [34] 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.
- [35] W. Wan, J. Chen, S. Zhang, and J. Xia. A Cluster Correlation Power Analysis against Double Blinding Exponentiation. *J. Inf. Secur. Appl.*, 48, 2019.
- [36] W. Wan, W. Yang, and J. Chen. An Optimized Cross Correlation Power Attack of Message Blinding Exponentiation Algorithms. *China Communications*, 12(6):22–32, 2015.
- [37] H. Zhang. On the Exact Relationship Between the Success Rate of Template Attack and Different Parameters. *IEEE Trans. Information Forensics and Security*, 15:681–694, 2020.

Changhai Ou received his B.S. degree in Computer Science and Technology from School of Computer and Information Technology, Beijing Jiaotong University, China. He received his Ph.D degree in Cyber Security from Institute of Information Engineering, Chinese Academy of Sciences (i.e. School of Cyber Security, University of Chinese Academy of Sciences) in July, 2018. He is now a Research Fellow in Hardware & Embedded Systems Lab (H-ESL), School of Computer Science and Engineering, Nanyang Technological University, Singapore. His

current research interests include hardware security, side-channel attacks and AI security.

Siew-Kei Lam received his BAsC, MEng and PhD from School of Computer Engineering (SCE), NTU. His PhD dissertation investigated techniques to automatically generate custom instructions for re-configurable processors. He was a Visiting Research Fellow in the Imperial College of London, University of Warwick, and RWTH Aachen, Germany. He is currently an Assistant Professor in SCE and his research investigates methods for realizing custom computing solutions in embedded systems.

Degang Sun is a research professor, PhD supervisor of Institute of Information Engineering, Chinese Academy of Sciences. He is also a professor in the School of Cyber Security, University of Chinese Academy of Sciences. His research interests include electromagnetic leakage protection, wireless communication technology and high security level information system protection technology.

Xinping Zhou received the bachelor's degree in mathematics from the Huazhong University of Science and Technology and the Ph.D. degree in computer science from the Institute of Information Engineering, Chinese Academy of Sciences, in 2012 and 2018, respectively. He is currently with Hardware Trustworthiness Technology & Engineering Lab, 2012 Laboratories, Huawei Technologies Co., Ltd. His research interests include side-channel attacks and fault attacks in hardware security.

Kexin Qiao received the bachelor's degree in mathematics from the Shandong University and the Ph.D. degree in computer science from the Institute of Information Engineering, Chinese Academy of Sciences, in 2012 and 2017, respectively. She is currently with Beijing Unionpay Card Technology Co., Ltd, China (BCTC). Her research interests include side-channel attacks, cryptanalysis, and cryptographic security implementation.

Qu Wang received the B.S. degree from the School of Software Engineering, Beijing University of Posts and Telecommunication, China, in 2013, the M.S. degree from the University of Chinese Academy of Sciences, Beijing, China, in 2017. He is currently pursuing the Ph.D. degree with the School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, China. His current main interests include location-based services, pervasive computing, computer vision, machine learning and security.