Side-Channel Leakage Evaluation and Detection Based on Communication Theory

Wei Yang, Yuchen Cao, Ke Ma, and Hailong Zhang

University of Chinese Academy of Sciences, Beijing, China
generalyzy@gmail.com

Abstract. Side-channel attacks (SCAs) have been a realistic serious threat to crypto devices. Therefore, evaluating the SCAs resilience of a crypto device is important and necessary. The SCAs-secure evaluation criteria includes the information theoretic metric and the security metric. The former metric, i.e. mutual information (MI), measures the leakage amount of a crypto device. However, because the real leakage distribution of a crypto device is unknown, the leakage evaluation is difficult. Commonly, there are two ways to estimate the leakage distribution of a device, i.e. non-parametric ones and parametric ones. The former may bring a big error since the leakage model is not accurate. The latter is more precise since it can profile the leakage model, but may be infeasible in practice. To combine the merits of the two estimation ways, we bypass the direct estimation of the device's leakage distribution, and propose a non-profiling parametric estimation method. We analyze the side-channel as a communication channel, and use the average MI of the communication channel to estimate the side-channel MI. Besides, we find that the channel capacity can furnish an upper bound of the leakage amount of the device. Interestingly, based on the communication channel characteristic, we find that if we do consistency check for the channel parameters, a leakage detection method can be developed. Furthermore, the proposed method is capable of finding the Point-Of-Interests (POIs) in leakage traces and introducing few leakage points that cannot be used to mount SCAs. Finally, the experiments show the effectiveness of the proposed methods about leakage evaluation and detection.

Keywords: side-channel leakage evaluation and detectioninformation theoretic metriccommunication channelaverage mutual informationchannel capacity

1 Introduction

Side-channel attacks (SCAs) aim to retrieve the secret information by analyzing the physical leakage of a crypto device [3,16,20]. SCAs have been a serious threat to crypto devices [4]. Hence, it is quite important to assess the SCAs resilience of crypto devices, which will be beneficial to design leakage-resilient crypto devices.

Considerable efforts have been done to propose tools for side-channel evaluation [11, 15, 17, 22, 25, 34, 35, 37], and two side-channel evaluation criteria were

proposed [34], [35], i.e. the information theoretic metric and the security metric. The information theoretic metric measures the leakage amount of a crypto algorithm implementation in a device. The existing metrics contain signal-noise ratio (SNR) [23], correlation coefficient [3], [14], mutual information (MI) [34], conditional entropy [35], etc. The last two metrics are equivalent. Since MI has clearer information theoretic meaning, it is usually used to measure the leakage amount of a crypto device. Unfortunately, obtaining the real MI is difficult because the real leakage distribution of a crypto device is generally unknown [13]. Estimation methods should be used. There are two ways to estimate the leakage distribution, i.e. non-parametric and parametric estimations. The first way is to select an approximate leakage model and use non-parametric methods (e.g. histogram and kernel estimations [13], [1]) to estimate the leakage distribution of a crypto device. In many cases, non-parametric ones bring assumption errors [14]. The other way is to estimate the leakage distribution of the device by parametric estimations, such as Gaussian templates, regression-based models [14], [13], etc. In this case, the concept of MI is replaced by perceived information (PI) [14], [13], [32]. Parametric ones are more precise, but they need a mass of leakage data to profile the model, which may be impracticable in real scenarios.

In this paper, we propose a non-profiling parametric estimation method based on communication theory. This paper views the side-channel as a communication channel and revisits MI in the channel model. The side-channel MI can be viewed as the average MI of the communication channel. The method investigates the distribution of the noise in the measured leakage and bypass estimating the leakage distribution of the device directly. The proposed method produces no assumption error and is more accurate than non-parametric methods. In addition, the channel capacity furnishes an upper bound of the leakage amount of a device and can be viewed as another information theoretic metric. It is used to provide a rough estimation of the leakage amount of a crypto device, and it can characterize the leakage amount in the worst scenario a device may leak. We investigate the side-channel leakage amount and its upper bound in both Gaussian and non-Gaussian noise scenarios, respectively. It is favorable for a profound security evaluation.

Interestingly, we also develop a leakage detection method based on the parameter estimations of the above communication channel. Leakage detection is closely related to side-channel security assessment, and also draws great attention recent years. The target of leakage detection is to find leakage points in side-channel leakage traces [12]. If leakage points contain secret information and can be exploited to mount a side-channel attack, they are named as Point-Of-Interests (POIs) [12]. A good leakage detection method should be able to find POIs and produce few useless leakage points for attacking [12]. Current studies on leakage detection are mostly based on T-test. These methods detect leakage by checking if there exist significant differences between two measurement sets through T-test [6, 10, 12, 18, 26, 33]. These two measurement sets corresponding to two different inputs, one fixed input and the other random input. T-test is suitable for the normal population and it at most considers the mean and vari-

ance. Hence many useless leakage points for attacking are often obtained by the T-test based detection methods. Besides, a leakage detection method based on correlation coefficient is also proposed [12], which can find POIs, but it requires the input to traverse all possible values. And a MI based method takes account of sub key is also proposed [26]. In addition, some other researches develop leakage detection methods without special requirements. For instance, a valid method based on variance test in [31] detects the leakage by comparing the variance of the mean of the measurements corresponding to different input. It can find the POIs, but also produces a few non-POIs since it only employs the variance information.

In this paper, we develop a leakage detection method based on communication channel characteristic. Since there is no useful signal at a non-leakage point, the signal can be viewed as noise signal. Intuitively, the communication channel established by the non-leakage signal should be variable-parameter channel. On the contrary, the communication channel established by a leakage signal should be constant parameter channel. Therefore, if we do consistency check for the parameters, some estimated parameters of the communication channel by using the data at a leakage point should be consistent estimators, while the parameters estimated by the non-leakage points are not. In other words, a parameter estimated by the leakage data should have a stronger consistency than the same estimated parameter computed by the non-leakage data. The leakage detection method needs no special requirements about the leakage acquisition and can find the POIs. It only produces few leakage points that cannot be directly accessed to mount an attack. Since the method employs the distribution of the leakage, it outperforms the leakage detection method proposed in [31], which only employs the variance information.

This paper is organized as follows: Section 2 is the preliminaries, Section 3 describes the proposed methods for leakage characterization, Section 4 shows the proposed method for leakage detection, Section 5 shows some extended discussion, and finally the conclusion is given in Section 6.

2 Preliminaries

2.1 Notations

This paper uses capital letters to denote random variables, and their corresponding observations are written as the lowercase letters. For a discrete random variable, e.g. X, its K corresponding observations can be written as $x = \{x_k\}$ with corresponding probabilities $\{p_k = Pr(x_k)\}$, where $k = 1, \ldots, K$, $Pr(\cdot)$ denotes the probability of the discrete random variable. Similarly, this paper denotes $p(\cdot)$ as the probability density function of a continuous random variable.

2.2 Finite Mixture Model, Gaussian Mixture Model and Expectation-Maximization Algorithm

Let y be an observation of a L-dimensional continuous random variable Y. The mixture-density function of a K-component finite mixture model (FMM) can be

expressed as follows [7]:

$$\begin{aligned} p(y|\Theta) &= \sum_{k=1}^{K} \alpha_k p_k(x|\theta_k), \\ s.t.\ 0 \leq \alpha_k \leq 1,\ &\sum_{k=1}^{K} \alpha_k = 1, \end{aligned} \tag{1}$$

where $\Theta = (\alpha_1, \alpha_2, \ldots, \alpha_K; \theta_1, \theta_2, \ldots, \theta_K)$ is the parameter set, and $p_k(x|\theta_k)$ is the probability density function (pdf) of the k-th component.

It can be seen that a FMM is a convex combination of some pdfs. When K components all follow the Gaussian distribution, this FMM is named as Gaussian mixture model (GMM). GMM is the most commonly used FMM [5]. FMM, including GMM, is a powerful and flexible statistical modeling tool which can be used to process complex data. FMM is capable of approximating any distribution with high accuracy [5], [29]. One of the most common related issue of FMMs is the parameter estimation problem, i.e. the estimation of component parameters [29]. A widely used solution is the expectation-maximization (EM) algorithm [9].

The EM algorithm is widely used to find maximum likelihood or posterior estimates of parameters in a model which misses values or contains unobserved latent variables [9]. The EM algorithm is an iterative method and each iteration involves two steps, i.e. the expectation step (E-step) and the maximization step (M-step). In the E-step, the algorithm evaluates the conditional expectation of the log-likelihood function of complete data contains latent variables by using the observed data and current estimates for the model parameters. In the M-step, the maximization of the conditional expectation of the log-likelihood function obtained in E-step is performed. The estimated parameters are then used in the next E-step. The EM algorithm ensures the convergency after finite iterations since the likelihood increases at each iteration. The computational complexity of the EM algorithm for estimating GMM is $O(Ln + Kn^2)$, where n denotes the number of samples.

3 Evaluating Side-Channel leakage based on Communication Theory

3.1 Analyse Side-Channel as a Communication Channel

Let s^* be a sub key used in a crypto device and T be a part of the plaintext or ciphertext. Denote a sensitive variable as $f(T, s^*)$, and the corresponding measured leakage as Y. Then we have

$$Y = X + E = \psi(f(T, s^*)) + E, \tag{2}$$

where X is the real leakage of a sensitive variable, ψ is a device-specific deterministic leakage function and E is a zero mean additive noise independent of X. Generally, X is a discrete random variable, and Y is continuous random variable

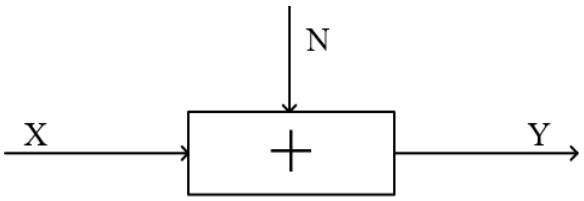


Fig. 1: The communication channel model of side channel.

because E is a continuous random variable. If X is viewed as the input and Y is viewed as the output, the side-channel can be viewed as a communication channel (Fig. 1). Then the side-channel leakage amount $I(X,Y)$ [1] can be seen as the average MI of the communication channel. Furthermore, an upper bound of the leakage amount of the crypto device can be estimated by calculating the communication channel capacity, since the communication channel theory reveals that the capacity is thought as the maximum of the average MI of the channel [2].

Provided that the image set of the sensitive variable has K elements in total, that is $x = \{x_k\},\ \{p_k = Pr(x_k)\},\ k = 1, \ldots, K$, where x is the observation set of X. The k-th input x_k is passed through the channel and the output is $y = \{x_k\} + e$, where y, e are the observations of Y and E, respectively. Then we have

$$p(y) = \sum_{k=1}^{K} p(y|x_k)p_k, \tag{3}$$

$$p(y, x_k) = p(y|x_k)p_k\ . \tag{4}$$

Furthermore, $p(y|x)$ is the channel transition probability and characterizes the channel. It can be seen that, in the communication channel, the MI does not refer to the computation of the sensitive variable and the leakage model.

3.2 Analysis on the Gaussian Channel

Average MI of the Channel The pdf $p(y)$ will be a 1-Dimensional Gaussian mixture model (GMM) if E in Fig. 1 follows the Gaussian distribution. Assume the noise variance is σ^2, then we have

$$p(y|x_k) = p(n)|_{n=y-x_k} = \frac{1}{\sqrt{2\pi}\sigma} exp(-\frac{(y-x_k)^2}{2\sigma^2}), \tag{5}$$

and

$$p(y) = \sum_{k=1}^{K} p(y|x_k)p_k = \sum_{k=1}^{K} p_k \frac{1}{\sqrt{2\pi}\sigma} exp(-\frac{(y-x_k)^2}{2\sigma^2})\ . \tag{6}$$

[1] In this paper, unless otherwise stated, the leakage amount of a device means the leakage amount of a leakage point corresponding to a sensitive variable in an implementation of a crypto algorithm on the device.

Given a set of observations $\{y_1, \ldots, y_N\}$, and assume the noise variance is σ^2, then the estimated parameter set

$$\{x_1, \ldots, x_k, p_1, \ldots, p_k; \sigma\}$$

can be solved by the EM algorithm [9] through maximizing the likelihood function $\prod_{n=1}^{N} p(y_n)$, which is equivalent to maximize the log-likelihood function $\sum_{n=1}^{N} \log(p(y_n))$. Expanding the log-likelihood function of the GMM as follows:

$$\sum_{n=1}^{N} \log(p(y_n)) = \sum_{n=1}^{N} \log \left\{ \sum_{k=1}^{K} p_k \frac{exp(-\frac{(y_n - x_k)^2}{2\sigma^2})}{\sqrt{2\pi}\sigma} \right\} . \tag{7}$$

The estimated parameters of the log-likelihood function can be iteratively solved by exploiting the EM algorithm. By setting the derivative of the likelihood w.r.t. each parameter to zero, respectively, the E-step can be expressed as

$$\hat{\gamma}_{nk}^{(t)} = \frac{p(y_n|x_k)p_k}{\sum_{k=1}^{K} p(y_n|x_k)p_k}, \tag{8}$$

and the M-step can be expressed as

$$\begin{aligned} \hat{x}_k^{(t+1)} &= \frac{\sum_{n=1}^{N} \hat{\gamma}_{nk}^{(t)} y_n}{\sum_{n=1}^{N} \hat{\gamma}_{nk}^{(t)}}, \\ \hat{p}_k^{(t+1)} &= \frac{1}{N} \sum_{n=1}^{N} \hat{\gamma}_{nk}^{(t)}, \\ \hat{\sigma}^{(t+1)} &= \left\{ \frac{1}{N} \sum_{n=1}^{N} \sum_{k=1}^{K} \hat{\gamma}_{nk}^{(t)} (y_n - x_k)^2 \right\}^{1/2} . \end{aligned} \tag{9}$$

where $n = 1, \ldots, N$, $k = 1, \ldots, K$, $\hat{\cdot}$ denotes the estimation of a parameter, t means the t-th iteration, and γ_{nk} means the probability that the k-th component produces the observation y_n. The iterations will be stopped when the evaluated log-likelihood is converged.

When the parameters are estimated, the average MI $I(X, Y)$ can be easily calculated. Then we have

$$I(X, Y) = H(Y) - H(Y|X) = H(Y) - H(N), \tag{10}$$

where $H(\cdot)$ denotes the information entropy of a random variable. Unfortunately, $H(Y)$ generally has no known closed-form solution because $H(Y)$ contains the logarithm of the exponential functions sum [19]. However, the approximation of $H(Y)$ can be obtained by some feasible measures like, using the observations $y_1, \ldots, y_N$ to estimate $H(Y)$, or acquiring the approximate value of $H(Y)$ through amount of samples generated by the known distribution of Y (e.g. Monte Carlo sampling), and so on. In order to ensure the accuracy and computability,

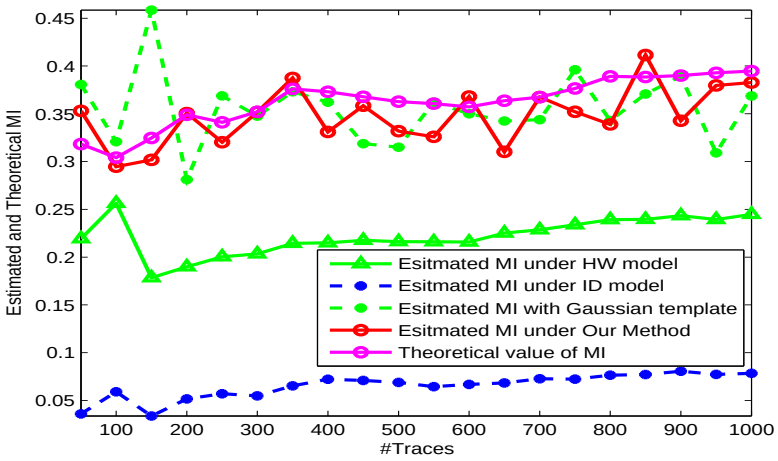


Fig. 2: The estimated and theoretical MI with Gaussian noise in a simulated scenario.

the Taylor-series expansion method is applied to obtain an appropriate approximation of $\log(p(y))$ [19]. Expanding $\log(p(y))$ around x_k of each component, the R-order Taylor-series expansion of $\log(p(y))$ can be denoted as follows:

$$\log(p(y)) = \sum_{r=1}^{R} \left\{ \frac{1}{r!} \frac{d^r}{dy} \{\log(p(y))\} (y - x_k)^r |_{y=x_k} \right\} + O_R, \tag{11}$$

where O_R is the Lagrange remainder term. Therefore, we have

$$H(Y) \approx - \int_{-\infty}^{+\infty} p(y) \left\{ \sum_{r=1}^{R} \frac{1}{r!} \frac{d^r}{dy} \{\log(p(y))\} (y - x_k)^r |_{y=x_k} \right\} dy \, . \tag{12}$$

For example, if the 2^{nd}-order Taylor-series expansion of $\log(p(y))$ is selected (i.e. R=2), the entropy of Y is approximated to

$$H(Y) \approx - \sum_{k=1}^{K} p_k \{ [\log(p(y)) - \frac{1}{2} f(y)] |_{y=x_k} \}, \tag{13}$$

where

$$\begin{aligned} f(y) = & \frac{1}{p(y)\sqrt{2\pi}\sigma} \sum_{i=1}^{K} p_i [\frac{1}{p(y)} (y - x_i) \frac{d}{dy} p(y) + \\ & \frac{1}{\sigma^2} (y - x_i)^2 - 1] exp(-\frac{(y - x_i)^2}{2\sigma^2}) \, . \end{aligned} \tag{14}$$

Note that the parameter K needs to be determined before performing the EM algorithm. The selection of the component number is another important issue of FMM. There are some results related to the selection of the optimum

K [5,39,40], but in this paper this optimal K is the one which leads to the largest MI. The selection of initial values also affects the result of the EM algorithm. Therefore, it is necessary to perform EM algorithm with different initial values several times till the average of MI converges.

Fig. 2 depicts a simulated experiment of leakage estimation of an unprotected AES-128 implementation. We use a stochastic leakage model as the theoretical leakage model, and use Gaussian noise as the noise. Two different leakage models, Hamming Weight (HW) and Identification (ID) models, are used to estimate the information amount by using kernel density estimators [30]. It can be seen that the average MI of the channel are in close proximity to the theoretic MI. The accuracy of the proposed method is similar to the parametric method which uses the Gaussian template to profile the leakage model. The two methods perform better than the non-parametric methods.

Interestingly, an upper bound of the average MI, which is termed as the channel capacity [2], can be used to furnish a rough estimation of the leakage amount of the device.

Channel Capacity The channel capacity C is the extremum which the average MI could achieve, i.e. C satieties

$$C = max\, I(X,Y) = max[H(Y) - H(N)]\ . \tag{15}$$

C is the convex function of the distribution of x and it is determined by $\{x_k\}$ and $\{p_k\}$. In practical device, the power of output signal Y is always limited. Therefore, based on information theory, the average MI will achieve the ultimate value when $\{x_k\}$ follows the Gaussian distribution [38]. The unknown parameters can also be obtained by the EM algorithm. The whole procedure is shown in the following.

Firstly, we classify the observations corresponding to a same plaintext or ciphertext (i.e. T) into the same group. Suppose there are m groups and each group has n_i elements, where $i = 1, \ldots, m$. Denote the j-th observation in the i-th group as y_{ij}. Since each y_{ij} is acquired individually, they are independent of each other and satisfy $(y_{ij}|x_i, \sigma) \sim \phi(x_i, \sigma^2)$, where $j = 1, \ldots, n_i$, $i = 1, \ldots, m$, ϕ means a Gaussian distribution with mean x_i and variance σ^2. Assume that $x_i \sim \phi(\mu, \tau^2)$ and denote the unknown parameters as $y = \{y_{ij}, j = 1, \ldots, n_i;\ i = 1, \ldots, m\}$, $z = (x_1, \ldots, x_m)$, $N = \sum_{i=1}^{m} n_i$ and $\theta = (\mu, log\,\sigma, log\,\tau)$, then from the Bayesian rules, we have

$$\begin{aligned} &p(z, \theta|y) = p(z, \theta, y)/p(y) = p(\theta|y, z)p(z|y), \\ &p(z, \theta, y) = p(\theta)p(z|\theta)p(y|z, \theta)\ . \end{aligned} \tag{16}$$

$p(y)$ and $p(z|y)$ are independent with θ, hence,

$$p(\theta|y, z) \propto p(z, \theta|y) \propto p(z, \theta, y), \tag{17}$$

and then

$$\log(p(\theta|y, z)) \propto \log(p(z, \theta|y)) \propto \log(p(z, \theta, y))\ . \tag{18}$$

Since the prior distribution θ can be considered proportional to τ [24], then we have

$$\log(p(\theta|y,z)) \propto -N\log\sigma-(m-1)\log\tau- \\ \frac{1}{2\tau^2}\sum_{i=1}^{m}(x_i-\mu)^2-\frac{1}{2\sigma^2}\sum_{i=1}^{m}\sum_{j=1}^{n_i}(x_i-y_{ij})^2\,. \tag{19}$$

Consequently, z can be viewed as the latent variable and the EM algorithm can be used. In the E-step, the expectation of Eq. (19) given by $\theta^{(t)}$ and y, $E_z\{\log(p(\theta|y,z))|\theta^{(t)},y\}$, where t means the t-th iterations, should be computed firstly. Because the conjugate prior distribution of x_i still is a Gaussian distribution [24], we have

$$(x_i|\theta^{(t)},y)\sim\phi(\upsilon_i^{(t)},\nu_i^{(t)}), \tag{20}$$

where

$$\begin{aligned}\upsilon_i^{(t)}&=[\frac{\mu}{(\tau^{(t)})^2}+\frac{\sum_{j=1}^{n_i}y_{ij}}{(\sigma^{(t)})^2}]/[\frac{1}{(\tau^{(t)})^2}+\frac{n_i}{(\sigma^{(t)})^2}],\\ \nu_i^{(t)}&=[\frac{1}{(\tau^{(t)})^2}+\frac{n_i}{(\sigma^{(t)})^2}]^{-1},\end{aligned} \tag{21}$$

The two formulas in Eq. (21) are the iterations in the E-step.

In the M-step, setting the derivative of $E_z\{\log(p(\theta|y,z))|\theta^t,y\}$ with respect to μ,σ and τ to zero, respectively, and the iterations can be obtained as

$$\begin{aligned}\hat{\mu}^{(t+1)}&=\frac{1}{m}\sum_{i=1}^{m}\upsilon_i^{(t)},\\ \hat{\sigma}^{(t+1)}&=\left\{\frac{1}{n}\sum_{i=1}^{m}\sum_{j=1}^{n_i}[(y_{ij}-\upsilon_i^{(t)})^2+\nu_i^{(t)}]\right\}^{1/2},\\ \hat{\tau}^{(t+1)}&=\left\{\frac{1}{m-1}\sum_{i=1}^{m}(\upsilon_i^{(t)}-\mu^{(t+1)})^2+\nu_i^{(t)}\right\}^{1/2}.\end{aligned} \tag{22}$$

Due to the independence of two Gaussian variable, the channel input X and the channel noise N, the channel output Y is also a Gaussian variable with mean μ and variance $\tau^2+\sigma^2$. At this time $C=I(X,Y)=\frac{1}{2}\log(1+\frac{\tau^2}{\sigma^2})$. C just describes an upper bound of the leakage amount of a device. The average MI may achieve C, or may not. By the way, m is the component number and needs not to be selected.

Fig. 3 shows an example about an unprotected AES-128 implementation on an 8-bit Micro-Controller Unit (MCU). The measured power leakage corresponds to the 9^{th} S-box output in the 1^{st} round encryption. From the empirical perspective, it can be approximately considered that the noise in the measured signal is an additive Gaussian noise. In Fig. 3, it can be observed that HW model exploits more information than ID model, which is anastomotic with the attack results (see Fig. 10, Appendix A). The average MI of the channel is better to characterize the real power leakage amount of the device because it only relies

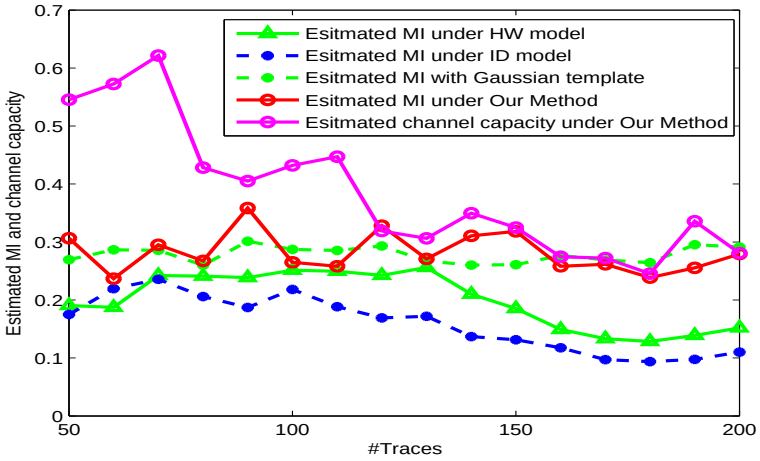


Fig. 3: The estimated MI and channel capacity with Gaussian noise in a real scenario.

on the measured leakage. Its result is about the same as the parametric method which uses Gaussian template to profile the leakage model. And the channel capacity provides an upper bound of the leakage of the device. Note that the average MI closely approaches the capacity when the trace number increases. It is because that both the real leakage of the device and the noise follow approximately Gaussian distribution. The result confirms that the approximation of the Gaussian noise is reasonable.

3.3 Analysis on the Non-Gaussian Channel

The above analysis is under the assumption of Gaussian noise. Nevertheless, the practical noise may be non-Gaussian with unknown closed-form. In this case, it is difficult to estimate parameters of the channel. Fortunately, as mentioned before, any distribution can be approximated by GMM at any accuracy, hence the distribution of a non-Gaussian noise can be characterized by a GMM. Provided that using a 1-Dimensional GMM with M components to approximate the noise, then $p(e)$ can be written as

$$p(e) = \sum_{m=1}^{M} \alpha_m \frac{1}{\sqrt{2\pi}\delta_m} exp(-\frac{e^2}{2\delta_m^2}), \tag{23}$$

where δ_m^2 is the variance of the m-th component. When $M = 1$, $p(e)$ reduces to Gaussian noise. Eq. (3) can be rewritten as

$$p(y) = \sum_{k=1}^{K} p_k \sum_{m=1}^{M} \alpha_m \frac{1}{\sqrt{2\pi}\delta_m} exp(-\frac{(y - x_k)^2}{2\delta_m^2}) \,. \tag{24}$$

Therefore, the log-likelihood function of the GMM is

$$\log\{\sum_{k=1}^{K} p_k[\sum_{m=1}^{M} \alpha_m \frac{1}{\sqrt{2\pi}\delta_m} exp(-\frac{(y-x_k)^2}{2\delta_m^2})]\}\ . \tag{25}$$

The unknown parameter set is

$$\{x_1, \ldots, x_k, p_1, \ldots, p_k; \alpha_1, \ldots, \alpha_m, \delta_1, \ldots, \delta_m\},$$

where $\sum_{k=1}^{K} p_k = 1$, $\sum_{m=1}^{M} \alpha_m = 1$, and $\forall k, m$, $p_k \geq 0$, $\alpha_m \geq 0$. Similarly, this estimation problem can be solved by the EM algorithm straightly. The E-step is

$$\begin{aligned} \hat{\gamma}_{nk}^{(t)} &= \frac{p(y_n|x_k)p_k}{\sum_{k=1}^{K} p(y_n|x_k)p_k}, \\ \hat{\beta}_{nm}^{(t)} &= \frac{\sum_{k=1}^{K} p_k \alpha_m \frac{1}{\sqrt{2\pi}\delta_m} exp(-\frac{(y-x_k)^2}{2\delta_m^2})}{\sum_{k=1}^{K} p(y_n|x_k)p_k}, \end{aligned} \tag{26}$$

and the M-step is

$$\begin{aligned} \hat{x}_k^{(t+1)} &= \frac{\sum_{n=1}^{N} \hat{\gamma}_{nk}^{(t)} y_n}{\sum_{n=1}^{N} \hat{\gamma}_{nk}^{(t)}}, \\ \hat{\alpha}_m^{(t+1)} &= \frac{1}{N} \sum_{n=1}^{N} \hat{\beta}_{nm}^{(t)}, \\ \hat{p}_k^{(t+1)} &= \frac{1}{N} \sum_{n=1}^{N} \hat{\gamma}_{nk}^{(t)}, \\ \hat{\delta}_m^{(t+1)} &= \left\{ \frac{1}{\sum_{n=1}^{N} \hat{\beta}_{nm}^{(t)}} \sum_{n=1}^{N} \hat{\beta}_{nm}^{(t)} (y_n - x_k)^2 \right\}^{1/2}, \end{aligned} \tag{27}$$

where $n = 1, \ldots, N$, $k = 1, \ldots, K$, $m = 1, \ldots, M$.

Afterwards, $H(E)$ can be approximated by the Taylor-series expansion and $H(Y)$ can be estimated by utilizing the observations $y_1, \ldots, y_N$. At last, $I(X, Y)$ can be obtained. The optimum K and M are selected to make the value of MI culminate.

Similarly, the channel capacity C can furnish a rough estimation of the leakage amount of the device. However, the closed-form solution of C is hard to be obtained. Fortunately, a bound of C is much easier to be provided. Denote σ^2 as the noise variance, C satisfies [28]

$$\frac{1}{2}\log(1 + \frac{\sigma_x^2}{\sigma^2}) \leq C \leq \frac{1}{2}\log(\frac{\sigma^2 + \sigma_x^2}{\sigma_e^2}) \tag{28}$$

if the input power $E(x^2) \leq \sigma_x^2$, where σ_e^2 is the entropy power of noise and it satisfies $H(E) = \frac{1}{2}\log(2\pi e\sigma_e^2)$. Therefore, the right term of Eq. (28) can be seen as an upper bound of the average MI of the channel.

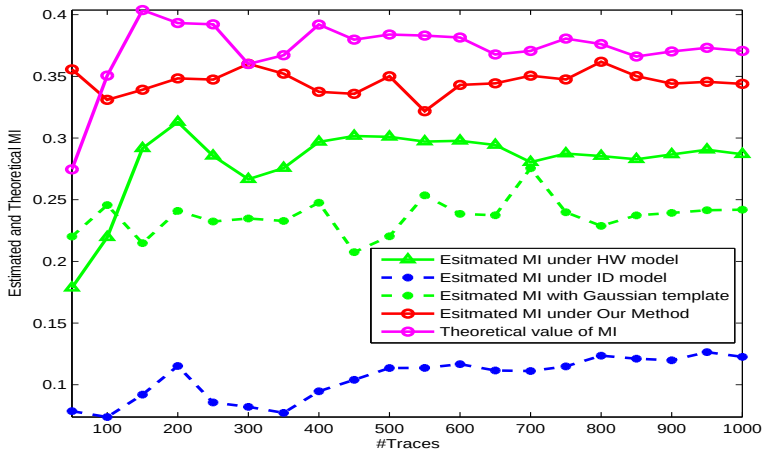


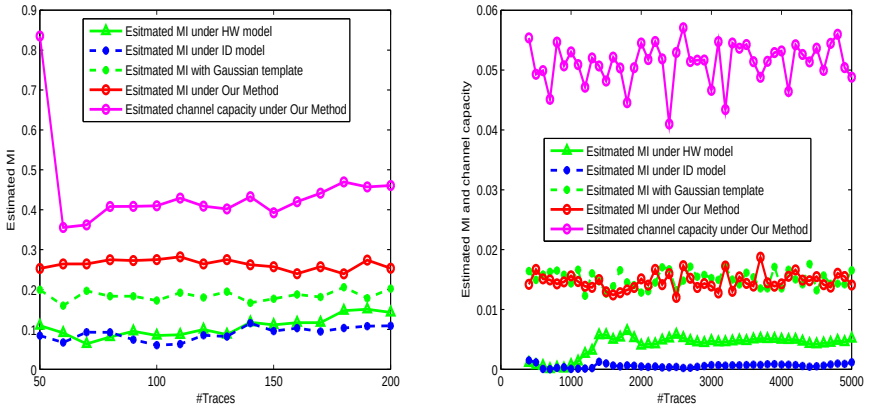
Fig. 4: The estimated and theoretical MI with non-Gaussian noise in a simulated scenario.

Fig. 4 depicts a simulated experiment of leakage estimation of an unprotected AES-128 implementation with a HW leakage model. The noise in the leakage signal is non-Gaussian. The distribution of the noise is a 3-component GMM. HW and ID models are used to estimate the leakage amount by using kernel density estimators. It can be seen that the average MI of the channel is close to the theoretic MI. Its accuracy is higher than the parametric method which uses Gaussian template to profile the leakage model. The proposed method is also better than non-parametric methods. Interestingly, the MI estimated by HW model performs better than the MI estimated by Gaussian template. It is because the noise is non-Gaussian noise and the simulated leakage model is HW model.

Fig. 5(a) describes an example of leakage estimation of an unprotected AES-128 implementation on an 8-bit MCU. The measured electromagnetic emanation leakage corresponds to the 9^{th} S-box output of 1^{st} round encryption and the noise is non-Gaussian according to empirical observation. HW and ID models are used to estimate the leakage amount. It can be observed that HW model outperforms ID model, which is anastomotic with the attack results. And the MI profiled by Gaussian template performs better than the non-parametric methods. Since the noise is non-Gaussian noise, the average MI of the channel is higher than the MI estimated by Gaussian template, and it is always lower than C with the increasing trace number.

Fig. 5(b) depicts an example of an AES-128 implementation with boolean masking on a smart card. It has a similar result as Fig. 3. The leakage have been preprocessed [8] to make the 1^{st}-order leakage information exposed. Due to the inevitable loss of preprocessing [8], the estimate values of the leakage amount of the device are all lower the device with unprotected implementations.

In conclusion, the proposed methods combine the broad applicability of non-



(a) (b)

Fig. 5: The estimated MI and channel capacity with non-Gaussian noise in real scenarios: (a) an unprotected AES-128 implementation on 8-bit MCU; (b) a protected AES-128 implementation on smart card.

parametric and the accuracy of parametric ways. And in practice, the average MI and capacity of the communication channel can be efficiently estimated because our analysis is base on the EM procedure, which is efficient as mentioned before.

4 Detecting Side-Channel Leakage Based on Communication Theory and Consistency Check

Interestingly, we also develop a leakage detection method based on the communication channel characteristic. Since there is no useful signal at a non-leakage point, it can be viewed as noise signal. Intuitively, the communication channel established with the non-leakage signal should be a variable-parameter channel. On the contrary, the communication channel established with a leakage signal should be a constant parameter channel. Therefore, if we do consistency check for the parameters, some estimated parameters of the communication channel by using the data at a leakage point should be consistent estimators, while the parameters estimated by the non-leakage points are not. In other words, a parameter estimated by the leakage data should have a stronger consistency than the same estimated by the non-leakage data.

Based on this judgement, the analysis for computing the average MI and capacity of the channel in Section 3.2 is also suitable for leakage detection. The estimated parameters to check consistency can be σ in Eq. (9), or σ, μ, τ in Eq. (22). The concept of consistency [24] is reviewed in the following.

Assume $\{Z_1, \ldots, Z_n\}$ is a sample of the population Z, $\theta \in \Theta$ is the parameter in Z, and $\hat{\theta} = \hat{\theta}(Z_1, \ldots, Z_n)$ is an estimation of θ. $\hat{\theta}$ is called a consistent

estimator of θ, if $\forall\theta \in \Theta$, when $n \to \infty$, $\hat{\theta}$ converges to θ with probability one, i.e.

$$\lim_{n\to\infty} Pr\{|\hat{\theta} - \theta| < \varepsilon\} = 1,\ \forall\varepsilon > 0\ . \tag{29}$$

It is not easy to carry out the consistency check in practice. To ensure a strong practical maneuverability of consistency check, an alternative scheme should replace the consistency check. The scheme computes the standard deviation of $\hat{\theta}$ in all iterations of the whole EM algorithm, and it assumes that the standard deviation of $\hat{\theta}$ at a leakage point should have a much less standard deviation than non-leakage points. It can reach a similar effect as consistency check because the fluctuation of the values of $\hat{\theta}$ should be small if $\hat{\theta}$ converge to θ with probability one.

Furthermore, to obtain a more robust result, the mean absolute deviation [36] of $\hat{\theta}$ is recommended to replace the standard deviation in this paper. The form of the mean absolute deviation of the samples in Z is

$$d_n = \frac{1}{n}\sum_{i=1}^{n} |Z_i - \bar{Z}|, \tag{30}$$

where $\bar{Z} = \frac{1}{n}\sum_{i=1}^{n} Z_i$ is the sample mean. Certainly, an empirical threshold is set to judge whether there exists leakage at a sample point. If d_n at a point is lower than the threshold, the point is viewed as a leakage point.

Some practical experiments verified the effectiveness of the proposed method (Figs. 6 and 7). The compared method is the variance detection technique [31], which also has no special requirements about the leakage acquisition.

Note that, for all figures, the horizontal axis represents the time samples, the vertical axis means the variance of the means of the traces with the same input or d_n of the estimator of τ in Eq. (22). In all figures, the dash line shows the threshold. The points have a value greater than the threshold of the variance, or less than the threshold of d_n, will be considered as leakage points. Fig. 6 shows the power leakage points of an unprotected AES-128 implementation on an 8-bit MCU detected by the proposed method and the variance detection technique [31], respectively. Fig. 7 depicts the power leakage points of an unprotected AES-128 implementation on FPGA detected by using the proposed method and the variance detection, respectively.

In Figs. 6 and 7, leakage points found by the two methods are almost exactly the same as those found by correlation power analysis (CPA) (see Fig. 11, Appendix A). These leakage points correspond to the 1^{st} S-box output of the MCU implementation or the XOR between the 1^{st} S-box input and output of the FPGA implementation. That is, leakage points detected by these methods are POIs, which can be used to mount an attack, while the T-test based detection methods often detect many useless leakage points for attacking [12]. In Figs. 6 and 7, the proposed method finds all POIs of CPA, but the compared method does not. Moreover, the discrimination between POIs and other points in Fig. 6(a) is larger than Fig. 6(b), and the discrimination in Fig. 7(a) is also larger than Fig. 7(b). Finally, to investigate anti-noise performance of the proposed

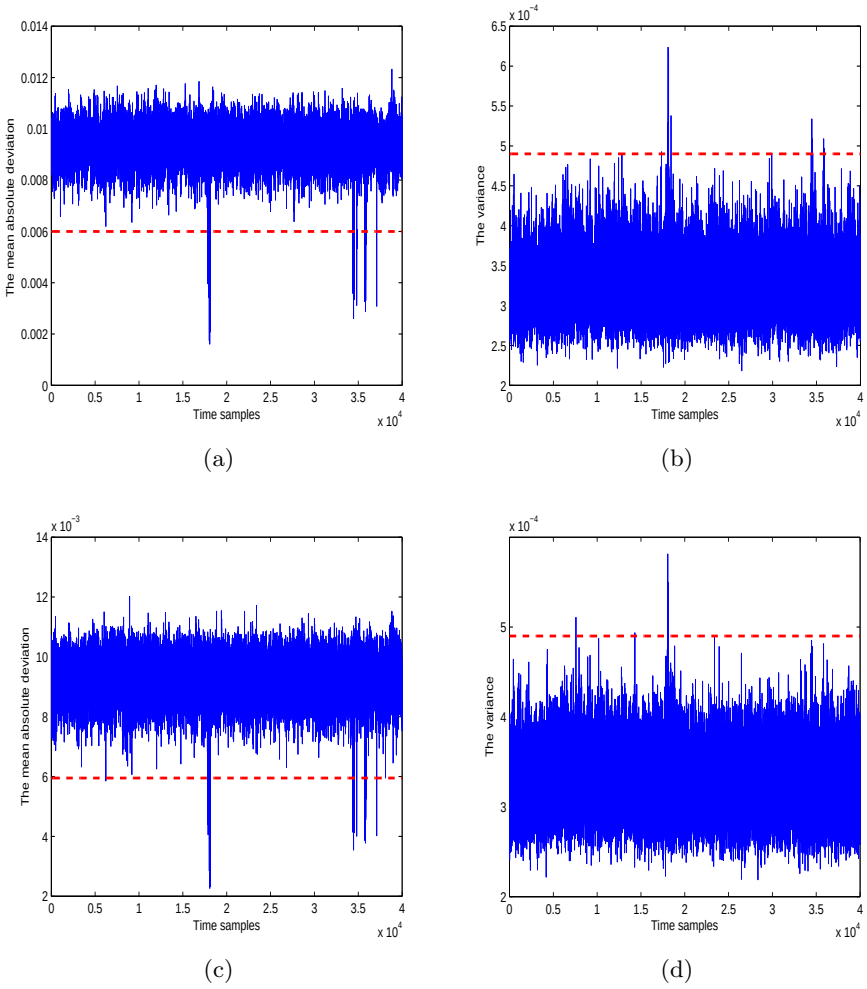


Fig. 6: The leakage detection results of the proposed method and the variance based method on an 8-bit MCU implementation (classifying traces according to 1^{st} byte of the plain texts): (a) the proposed method (1,000 traces, SNR=$+\infty$); (b) the variance based method (1,000 traces, SNR=$+\infty$); (c) the proposed method (1,000 traces, SNR=30dB); (d) the variance based method (1,000 traces, SNR=30dB).

method, some extra experiments are performed by adding noise to the original signals. The results are shown in Figs. 6(c)(d), 7(c) and 7(d). It can be observed that the proposed method still finds all POIs of CPA, but the compared method lose some POIs. However, both the two methods will perform worse if the noise continues to increase. In this case, increasing the number of leakage can counteract the negative effects of the increased noise.

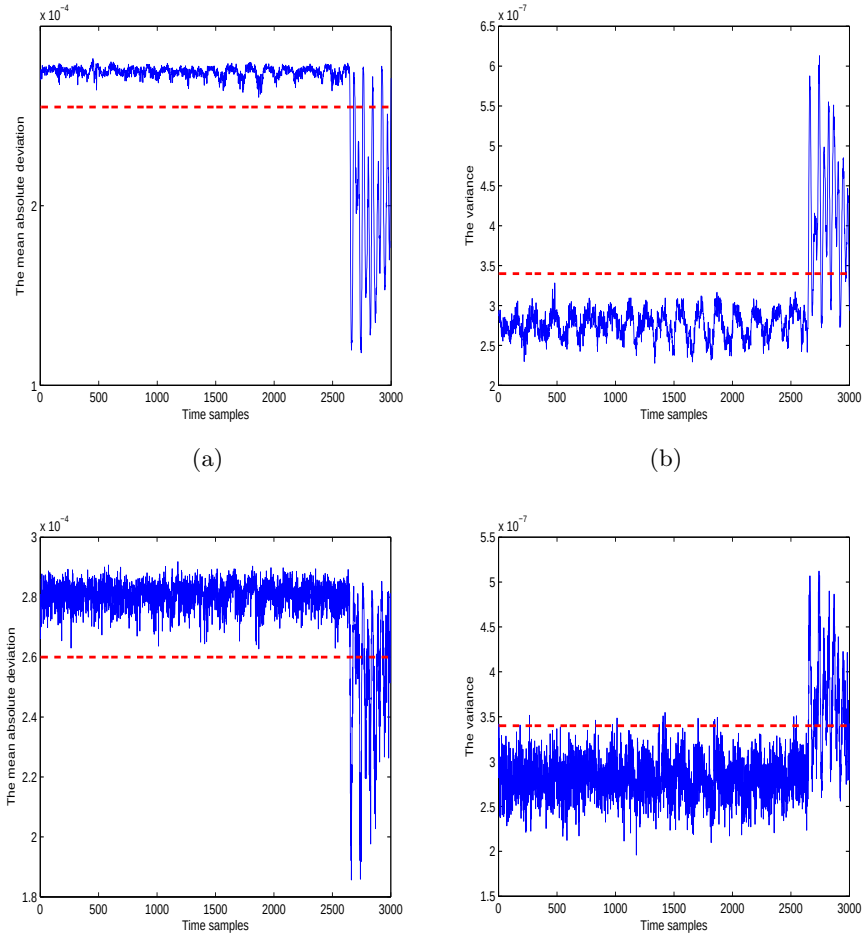
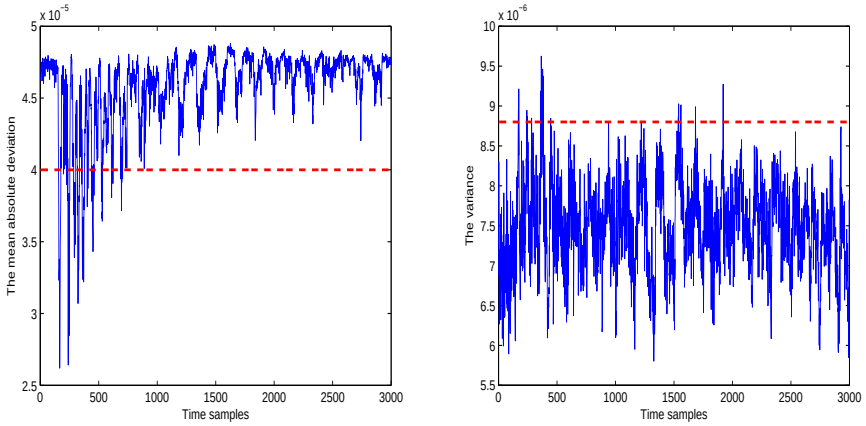


Fig. 7: The leakage detection results of the proposed method and the variance based method on a FPGA implementation (classifying traces according to 1^{st} byte of the cipher texts): (a) the proposed method (30,000 traces, SNR=$+\infty$); (b) the variance based method (30,000 traces, SNR=$+\infty$); (c) the proposed method (30,000 traces, SNR=80dB); (d) the variance based method (30,000 traces, SNR=80dB).

Fig. 8 shows the leakage points of the first round of the same FPGA implementation by using the two methods, respectively. The proposed method perform much better than the variance based method. A noteworthy exception is that sometimes the leakage points of some bytes found by the two methods may contain few POIs. For instance, in the last round of AES-128, the operation



(a) (b)

Fig. 8: The leakage detection results of the proposed method and the variance based method on a FPGA implementation (classifying traces according to 1^{st} byte of the plain texts): (a) the proposed method (30,000 traces); (b) the variance based method (30,000 traces).

"ShiftRows" will make the search of POIs of some bytes infeasible (see Fig. 12, Appendix A).

Additionally, the method works well for detecting other types of leakage, e.g. electromagnetic emission (Fig. 9). Moreover, the method cannot detect the POIs of a masking implementation (e.g. a masked AES-128 implementation, see Fig. 13, Appendix A) since the mask makes the leakage change randomly, unless the leakage have been preprocessed before detection (see Fig. 14, Appendix A). Furthermore, similar results can be obtained if other parameters (e.g. σ in Eq. (9)) are used and they are no longer shown here.

Since the method employs the distribution of the leakage, it performs slightly better than the leakage detection method based on variance test, which only employs the variance information.The proposed leakage detection method is under the assumption that the measured leakage has a Gaussian noise. Gaussian noise make a communication channel have a minimum capacity [28], that is, the proposed method can works even the device has the minimum bound of leakage amount.

5 Discussion

5.1 Multiple Leakage Points Analysis

If there exists multiple leakage points in the measurements, the input becomes an extended information source, and each leakage point corresponds to one communication channel. The entropy of the input and the average MI will increase,

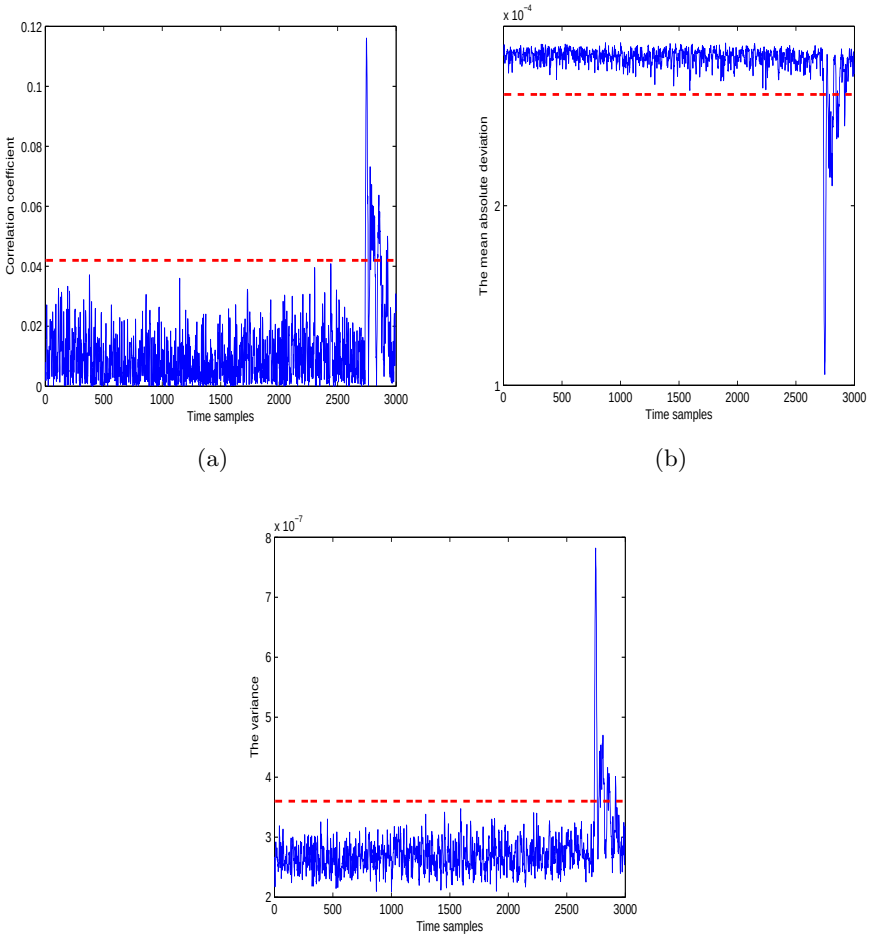


Fig. 9: The electromagnetic emission leakage detection results of CPA, the proposed method and the variance based method on a FPGA implementation (classifying traces according to 1^{st} byte of the plain texts): (a) CPA (30,000 traces); (b) the proposed method (30,000 traces); (c) the variance based method (30,000 traces).

which explains why multi-points SCAs outperform the single point SCAs [27] from the view of information theory.

5.2 Leakage Profiling

It is also possible to perform leakage profiling if the key is known. Because γ_{nk} in Eq. (8) means the probability that the k-th component produces the observation y_n, the leakage value corresponding to each T can be determined by utilizing

some match algorithms, e.g. Hungary algorithm [21]. When the key is known, the intermediate value is also known, hence the leakage model can be obtained.

5.3 Collision Attack

After determining the leakage value corresponds to each T by utilizing a match algorithm, a novel collision attack can be developed to recover the key if the key is unknown. Take AES-128 as an example, we can divide the whole plaintext or ciphertext into 16 bytes and each byte corresponds to a T. If the leakage value corresponds to each T is known, then the relations of all sub key are determined. Just enumerating all possible value of a sub key, the master key can be recovered by verifying the cipher-plain texts.

5.4 Noise Reduction

By the way, a preprocessing method can be developed to reduce the noise of the measured leakage. After the estimation of the noise parameters in Eq. (9), Eq. (22) or Eq. (27), the noise can be characterized and weakened. Moreover, SNR can be estimated, too.

6 Conclusion

In this paper, we researched side-channel leakage evaluation and detection in a communication channel model, and proposed non-profiling parametric estimations to compute the leakage amount of a crypto device under different communication channel models. We used the average MI to characterize the leakage amount of a crypto device. Besides, we also found that the channel capacity can be viewed as a rough evaluation of the upper bound of the leakage amount of the device according to communication theory. Furthermore, the average MI and the capacity of the channel both can be efficiently computed. Interestingly, we also proposed a novel leakage detection method based on the communication channel characteristic and the consistency check of channel parameters. In the future, we will continue the research of the four byproducts mentioned in Section 5, and investigate the leakage detection on the measured leakage of masking implementations without the knowledge of masks.

References

1. L. Batina, B. Gierlichs, E. Prouff, M. Rivain, F. Standaert, and N. Veyrat-Charvillon, Mutual information analysis: a comprehensive study, J. Cryptol. 24 (2) (2011) 269–291.
2. R. Bose, Information theory, coding and cryptography, Tata McGraw-Hill Education, Columbus, 2008.
3. E. Brier, C. Clavier, and F. Olivier, Correlation power analysis with a leakage model, in: CHES 2004, Cambridge, MA, USA, August, 2004, pp. 16–29.

4. C. Carlet, E. Prouff, M. Rivain, and T. Roche, Algebraic decomposition for probing security, in: CRYPTO 2015, Santa Barbara, CA, USA, August, 2015, pp. 742–763.
5. X. Chen, X. Liu, and Y. Jia, Unsupervised selection and discriminative estimation of orthogonal gaussian mixture models for handwritten digit recognition, in: ICDAR 2009, Barcelona, Spain, July 2009, pp. 1151–1155.
6. J. Cooper, E. DeMulder, G. Goodwill, J. Jaffe, G. Kenworthy, P. Rohatgi, et al, Test vector leakage assessment (tvla) methodology in practice, in: International Cryptographic Module Conference, 2013.
7. I. G. Costa Filho, Mixture Models for the Analysis of Gene Expression, PhD thesis, Freie Universität Berlin, 2008.
8. G. Dabosville, J. Doget, and E. Prouff, A new second-order side channel attack based on linear regression, IEEE Trans. Comput. 62 (8) (2013) 1629–1640.
9. A. P. Dempster, N. M. Laird, and D. B. Rubin, Maximum likelihood from incomplete data via the em algorithm, Journal of the royal statistical society, Series B (methodological), 39 (1) (1977) 1–38.
10. A. A. Ding, C. Chen, and T. Eisenbarth, Simpler, faster, and more robust t-test based leakage detection, in: COSADE 2016, Graz, Austria, April, 2016, pp. 163–183.
11. A. Duc, S. Faust, and F. Standaert, Making masking security proofs concrete - or how to evaluate the security of any leaking device, in: EUROCRYPT 2015, Sofia, Bulgaria, April, 2015, pp. 401–429.
12. F. Durvaux and F. Standaert, From improved leakage detection to the detection of points of interests in leakage traces, in: EUROCRYPT 2016, Vienna, Austria, May, 2016, pp. 240–262.
13. F. Durvaux, F. Standaert, and S. M. D. Pozo, Towards easy leakage certification, in: CHES 2016, Santa Barbara, CA, USA, August, 2016, pp. 40–60.
14. F. Durvaux, F. Standaert, and N. Veyrat-Charvillon, How to certify the leakage of a chip? in: EUROCRYPT 2014, Copenhagen, Denmark, May, 2014, pp. 459–476.
15. Y. Fei, Q. Luo, and A. A. Ding, A statistical model for DPA with novel algorithmic confusion analysis, in: CHES 2012, Leuven, Belgium, September, 2012, pp. 233–250.
16. K. Gandolfi, C. Mourtel, and F. Olivier, Electromagnetic analysis: Concrete results, in: CHES 2001, Paris, France, May, 2001, pp. 251–261.
17. B. Gierlichs, L. Batina, P. Tuyls, and B. Preneel, Mutual information analysis, in: CHES 2008, Washington, D.C., USA, August, 2008, pp. 426–442.
18. B. J. Gilbert Goodwill, J. Jaffe, and P. Rohatgi, A testing methodology for side-channel resistance validation, in: NIST non-invasive attack testing workshop, 2011.
19. M. F. Huber, T. Bailey, H. Durrant-Whyte, and U. D. Hanebeck, On entropy approximation for gaussian mixture random vectors, in MFI 2008, Seoul, Korea, August, 2008, pp. 181–188.
20. P. C. Kocher, Timing attacks on implementations of diffie-hellman, rsa, dss, and other systems, in: CRYPTO 1996, Santa Barbara, California, USA, August, 1996, pp. 104–113.
21. H. W. Kuhn, The Hungarian Method for the Assignment Problem, Springer, Heidelberg, 2010.
22. V. Lomné, E. Prouff, M. Rivain, T. Roche, and A. Thillard, How to estimate the success rate of higher-order side-channel attacks, in: CHES 2014, Busan, South Korea, September, 2014, pp. 35–54.
23. S. Mangard, Hardware countermeasures against DPA ? A statistical analysis of their effectiveness, in: CT-RSA 2004, San Francisco, CA, USA, February, 2004, pp. 222–235.
24. S. Mao, J. Wang, and X. Pu, Advanced Mathematical Statistics, Higher Education Press, Beijing, 2009.

25. D. P. Martin, J. F. O'Connell, E. Oswald, and M. Stam, Counting keys in parallel after a side channel attack, in: ASIACRYPT 2015, Auckland, New Zealand, November, 2015, pp. 313–337.
26. 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: ASIACRYPT 2013, Bengaluru, India, December, 2013, pp. 486–505.
27. L. Mather, E. Oswald, and C. Whitnall, Multi-target DPA attacks: Pushing DPA beyond the limits of a desktop computer, in: ASIACRYPT 2014, Kaoshiung, Taiwan, December, 2014, pp. 243–261.
28. R. McEliece, The theory of information and coding, Cambridge University Press, Cambridge, 2002.
29. G. McLachlan and D. Peel, Finite mixture models, John Wiley and Sons, New York, 2004.
30. Y.-I. Moon, B. Rajagopalan, and U. Lall, Estimation of mutual information using kernel density estimators, Physical Review E 52 (3) (1995) 2318.
31. A. Moradi, S. Guilley, and A. Heuser, Detecting hidden leakages, in: ACNS 2014, Lausanne, Switzerland, June, 2014, pp. 324–342.
32. M. Renauld, F. Standaert, N. Veyrat-Charvillon, D. Kamel, and D. Flandre, A formal study of power variability issues and side-channel attacks for nanoscale devices, in: EUROCRYPT 2011, Tallinn, Estonia, May, 2011, pp. 109–128.
33. T. Schneider and A. Moradi, Leakage assessment methodology - A clear roadmap for side-channel evaluations, in: CHES 2015, Saint-Malo, France, September, 2015, pp. 495–513.
34. F. Standaert, T. Malkin, and M. Yung, A unified framework for the analysis of side-channel key recovery attacks (extended version), IACR Cryptology ePrint Archive 2006 (2006) 139.
35. F. Standaert, T. Malkin, and M. Yung, A unified framework for the analysis of side-channel key recovery attacks, in: EUROCRYPT 2009, Cologne, Germany, April, 2009, pp. 443–461.
36. M. A. Tanner, Tools for statistical inference: methods for the exploration of posterior distributions and likelihood functions, Biometrics 54 (2) (1998) 560-563.
37. N. Veyrat-Charvillon, B. Gérard, and F. Standaert, Security evaluations beyond computing power, in: EUROCRYPT 2013, Athens, Greece, May, 2013, pp. 126–141.
38. L. Wei, Channel capacity and constellation optimization of m-pam input awgn channels with non-equiprobable symbols, `http://people.cecs.ucf.edu/lei/letter_ieee_capacityMPSK_v2.pdf`.
39. C.-H. Xie, J.-Y. Chang, and Y.-J. Liu, Estimating the number of components in gaussian mixture models adaptively for medical image, Optik-International Journal for Light and Electron Optics 124 (23) (2013) 6216–6221.
40. M. Zhang and Q. Cheng, Determine the number of components in a mixture model by the extended KS test, Pattern Recogn. Lett. 25 (2) (2004) 211–216.

Appendix: Additional Figures

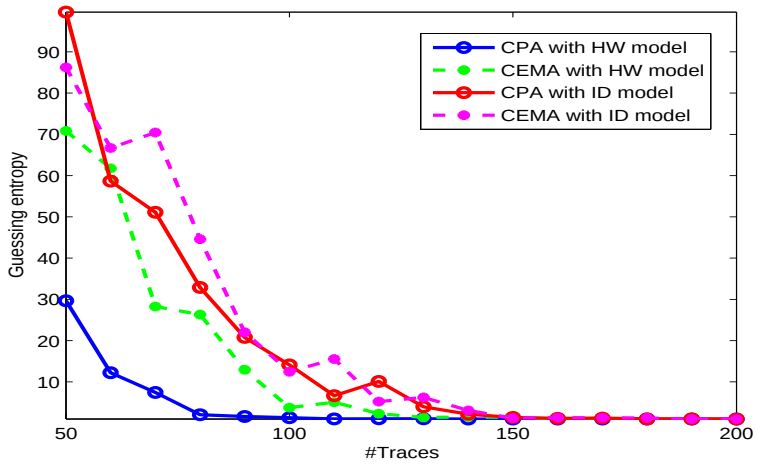
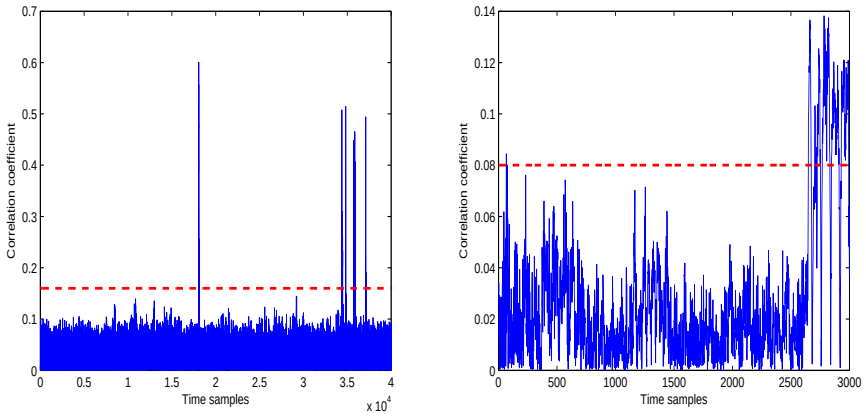
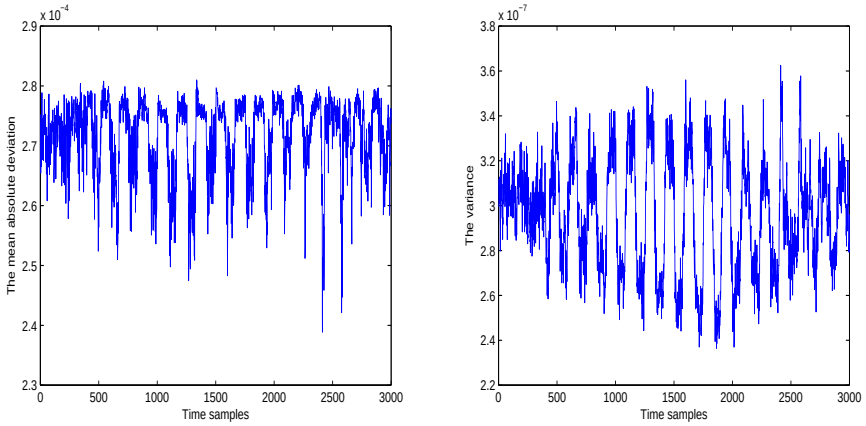


Fig. 10: The guessing entropy of CPA and CEMA under HW and ID model on an unprotected AES-128 implemented on an 8-bit MCU (the target intermediate value is 9^{th} S-box output of 1^{st} round).



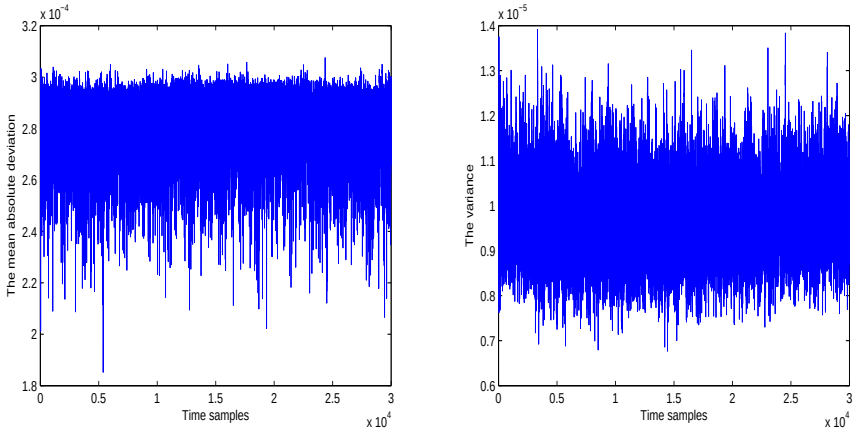
(a) (b)

Fig. 11: POIs of an unprotected AES-128 implementation on MCU (target intermediate value: 1^{st} S-box output of 1^{st} round) and FPGA (target intermediate value: XOR between the 1^{st} S-box input and output of the last round) found by CPA: (a) MCU (1,000 traces); (b) FPGA (30,000 traces).



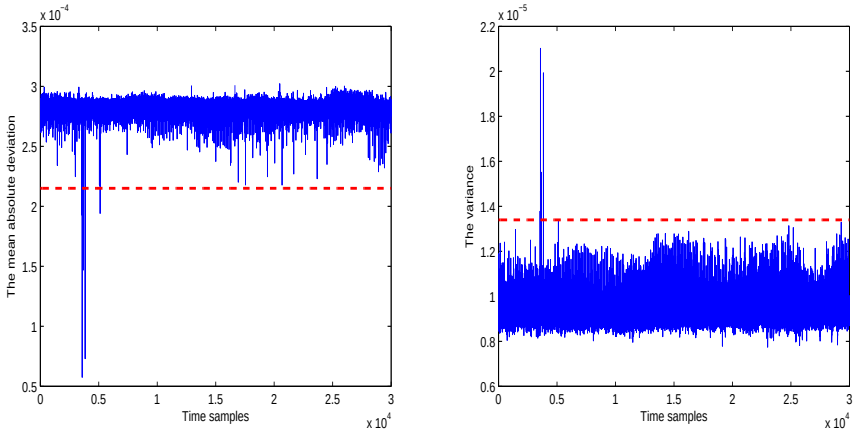
(a) (b)

Fig. 12: The leakage detection results of an unprotected AES-128 implementation on FPGA found by the proposed method and the variance based method (classifying traces according to 2^{st} bytes of the cipher texts): (a) the proposed method (30,000 traces); (b) the variance based method (30,000 traces).



(a) (b)

Fig. 13: The leakage detection results of a masked AES-128 implementation on a smart card found by the proposed method and the variance based method (without preprocessing, and classifying traces according to 1^{st} bytes of the plain texts): (a) the proposed method (5,000 traces); (b) the variance based method (5,000 traces).



(a) (b)

Fig. 14: POIs of a masked AES-128 implementation on a smart card found by the proposed method and the variance based method (preprocessing and classifying 5,000 traces according to 1^{st} bytes of the plain texts): (a) the proposed method; (b) the variance based method.