Leakage Characterizing and Detecting Based on Communication Theory

Wei Yang[1,2,⋆], Yuchen Cao[1,2], Ke Ma[1,2], Hailong Zhang[1], Yongbin Zhou[1], and Baofeng Li[3]

[1]State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
[2]University of Chinese Academy of Sciences, Beijing, China
generalyzy@gmail.com, {caoyuchen,make,zhanghailong,zhouyongbin}@iie.ac.cn
[3]China Electric Power Research Institute, Beijing, China
libaofeng@epri.sgcc.com.cn

Abstract. Evaluating the side-channel attacks (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 measures the leakage amount of a crypto device. It should be independent with the evaluator. However, the current metrics, e.g. mutual information (MI), conditional entropy and perceived information, are related to the leakage model selected by the evaluator. They only reflect the leakage utilization, rather than the real leakage level of a crypto device. In light of this, we analysis the side-channel as a communication channel and develop two objective metrics, the average MI and capacity of the channel, to characterize the real leakage amount and its upper bound of a crypto device through communication theory. Although the channel capacity is a rough estimation of the leakage amount of the device, it can furnish the leakage amount at the worst case scenario the device may leak. We investigate the estimation methods of the two metric in different noise scenes. Besides, a leakage detection method based on consistency check is developed subsequently. The proposed method are capable of finding the Point-Of-Interests (POIs) in leakage traces and introducing few leakage points cannot be used to mount SCAs. The experiments show the effectiveness of the proposed method.

Keywords: Side-Channel Leakage Characterizing, Information Theoretic Metric, Communication Channel, Average Mutual Information, Channel Capacity, Leakage Detecting

1 Introduction

Side-channel attacks (SCAs) aim to retrieve the secret information by analyzing the physical leakage of a crypto implementation, e.g. timing [1], power consumption [2], or electromagnetic emanation [3], etc. The physical leakage is emitted

⋆ Corresponding author, Email: generalyzy@gmail.com.

during internal computations relating to the secret key used in a crypto implementation. SCAs have been a realistic serious threat to crypto devices [4]. It is important and necessary to assess the SCAs resilience of crypto devices and it will be beneficial to design leakage-resilient crypto implementations.

Considerable research about side channel evaluation techniques are developed [5–12]. And two side-channel evaluation criteria were proposed by [5], [7]: one is the information theoretic metric, which measures the amount of leakage information provided by a given cryptographic algorithm implementation in a device; and the other is the security metric, which measures the ability of an actual adversary to turn the leakage information into a successful attack. The information theoretic metric should meet "two requirements: (1) being independent of the adversary and (2) having the same meaning for any implementation or countermeasure" [7], i.e. the information theoretic metric should be objective. The existing metrics contain signal-noise ratio (SNR) [13], correlation coefficient [2], [14], mutual information (MI) [5], conditional entropy [7], perceived information (PI) [14], [15], etc. Comparing with the first two metrics, the last three metrics have clearer information theoretic meaning. However, the last three metrics and correlation coefficient are related to the profiling of the leakage model. It leads to that the last three metrics characterize the amount of leakage information which the evaluator can employ, rather than the real amount of leakage information of a crypto device. These metrics violates the first requirement.

Another work closely related to side-channel security assessment, leakage detection, also drew great attention recent years. Leakage detection is to find the leakage points in side-channel measurements [18]. If the leakage points contain the secret information and can be exploited to mount a side-channel attack, they are termed Point-Of-Interests (POIs) [18]. A good leakage detection method should be able to find POIs and few useless leakage points for attacking [18]. Current studies on leakage detection are most based on t-test. This kind of methods detect leakage by checking if there exist significant differences between two measurement sets through t-test [16–21]. These two measurement sets are corresponding to two differing inputs, one fixed input and the other random input. T-test is serviced for the normal population and it at most considers the mean and variance. And 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 was also proposed in [18]. It can find POIs, but it requires that the input traverses all possible values. And a MI based method needs take account of sub key was also provided in [19]. Besides, some other research develops leakage detection methods without special requirements. For instance, a valid method based on variance test in [22] detects the leakage by comparing the variance of the measurements' mean of different input. It can not only find the POIs, but also produce a few non-POIs points.

Our Contribution. First, we consider the side channel as a communication channel and revisit information theoretic metric in this communication channel model. The input of the channel is the real leakage of a sensitive intermediate value corresponding to an operation in the cryptographic implementation, and

the output is the measured leakage. We redefine the average MI of the channel as the information theoretic metric to measure the leakage amount of the cryptographic device. The metric is independent with the evaluator and produces no assumption error of the side-channel evaluator's model. It is more objective than the existing MI, conditional entropy and PI metrics.

Second, we provide another objective metric, the channel capacity, furnishing an upper bound of the leakage amount of a device. It is used to afford a rough estimation of the leakage amount of the device, and can characterize the leakage amount at the worst case scenario the device may leak.

Third, we investigate the leakage amount and its upper bound of a crypto device in different noise scenes, and deduce the methods to estimate the two metrics. It is important for a more profound security evaluation.

At last, we also develop a detection method without special requirements and can find the POIs. It only produces few leakage points cannot direct access to mount an attack. The method is based on the side-channel analysis as a communication channel problem. Since the method employs the distribution of the leakage, it outperforms the leakage detection method based on variance test proposed in [22].

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

2 Preliminaries

2.1 Notations

Using 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. Similarly, denoting $p(\cdot)$ as the probability density function to represent probability distribution features of continuous random variables.

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

Let y is an observation of a L-dimensional continuous random variable Y. The mixture-density function for a K-component finite mixture model (FMM) is shown as follows [23]:

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

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 the K components all obey to Gaussian distribution, this FMM is named Gaussian mixture model (GMM). GMM is the most commonly used FMM [24]. FMMs, including GMM, is a kind of powerful and flexible statistical modeling tool for processing complex data. They are capable of approximating any distribution with high accuracy [24], [25]. One of the most common related issue of FMMs is parameter estimation problem, i.e. the estimation of component parameters [25]. A widely used solution is the expectation-maximization (EM) algorithm [26].

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 [26]. The EM algorithm is an iterative method and each iteration involves two steps: 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.

3 Analysis Side Channel as a Communication Channel

3.1 Modeling Side Channel as a Communication Channel

Let s^* denote a sub key used in an implementation of a crypto algorithm (e.g. AES) on a device and, T denote part of the plain text or cipher text. Considering a sensitive variable $f(T, s^*)$, of an operation related to s^* and T, such as the output of a substitution box in the first round. Denote the measured leakage of this sensitive variable as Y. It can be expressed as follows:

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

where X is the real leakage of a sensitive variable, ψ is a device-specific deterministic leakage function and N is an zero mean additive noise independent from X. Generally, X is a discrete random variable, and Y is continuous random variable because N is a continuous random variable. If X is viewed as the input and Y as the output, the side channel can be investigated as a communication channel (Fig. 1). Then the spillage problem of the side channel[1] can be rewritten as the average mutual information problem of the communication channel. Furthermore, an upper bound of the leakage amount of the crypto device can be

[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.

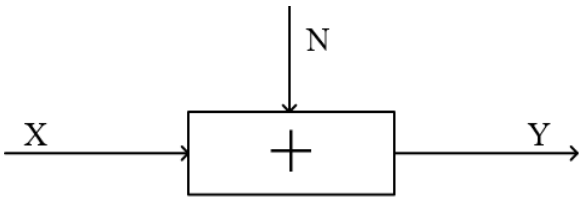


Fig. 1: The communication channel model of side channel.

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 [28].

Our setup is inspired by [27], which uses a communication channel to recover the key. The input and output of the model in [27] are respectively the secret key and key guess. But our model is for leakage characterizing and the input and output are different from the model in [27]. In our communication channel, the input and output do not refer to the computation of the sensitive variable and the leakage function. Therefore, the average mutual information of our communication channel model is independent of the evaluator and can be an information theoretic metric [7]. It produces no assumption error of the side-channel evaluator's model, and is more objective and better suitable for characterizing the amount of leakage information of a crypto device than some existing information theoretic metrics based on the profiling of the leakage of targeted intermediate variable, e.g. the MI or conditional entropy between the key and the measured leakage of a targeted device [7], PI [14], [15], correlation coefficient [2], [14], etc.

Provided that the image set of the sensitive variable mapped by ψ has K elements in total, that is, the observations of X have K possible values. It can be written as

$$x = \{x_k\},\ \{p_k = Pr(x_k)\}, \tag{3}$$

where $k = 1, \ldots, K$. The k-th input signal x_k is passed through the channel and the output is

$$y = \{x_k\} + n, \tag{4}$$

where y, n are the observations of Y and N, respectively. Then we have

$$p(y) = \sum_{k=1}^{K} p(y|x_k)p_k,\ p(y, x_k) = p(y|x_k)p_k\ . \tag{5}$$

Furthermore, $p(y|x)$ is the channel transition probability distribution and characterizes the communication channel.

3.2 Analysis on the Communication Channel with Gaussian Noise

Average MI of the Communication Channel. The pdf $p(y)$ in Eq. (5) will be a 1-Dimensional GMM if the zero mean additive noise N in Fig. 1 follows

Gaussian distribution. Assume the noise variance is σ^2, and then we have

$$p(y|x_k) = p(n)|_{n=y-x_k} = \frac{1}{\sqrt{2\pi}\sigma} exp(-\frac{(y-x_k)^2}{2\sigma^2}), \tag{6}$$

and

$$p(y) = \sum_{k=1}^{K} p(y|x_k)p_k = \sum_{k=1}^{K} p_k \frac{1}{\sqrt{2\pi}\sigma} exp(-\frac{(y-x_k)^2}{2\sigma^2}) \ . \tag{7}$$

Given a set of observations $\{y_1, \ldots, y_N\}$, a desired estimated parameter set $\{x_1, \ldots, x_k, p_1, \ldots, p_k; \sigma\}$ should maximize the probability that the GMM generates these given observations, i.e. maximize 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:

$$\sum_{n=1}^{N} log(p(y_n)) = \sum_{n=1}^{N} log\{\sum_{k=1}^{K} p_k \frac{1}{\sqrt{2\pi}\sigma} exp(-\frac{(y_n-x_k)^2}{2\sigma^2})\} \ . \tag{8}$$

The estimated parameters of the log-likelihood function can be iteratively solved through the EM algorithm. By setting the derivative of the likelihood w.r.t. each parameter to zero, the iteration of E-step is

$$\hat{\gamma}_{nk}^{(t)} = \frac{p(y_n|x_k)p_k}{\sum_{k=1}^{K} p(y_n|x_k)p_k}, \tag{9}$$

and the M-step is

$$\begin{aligned} \hat{x}_k^{(t+1)} &= \frac{\sum_{n=1}^{N} \hat{\gamma}_{nk}^{(t)} y_n}{\sum_{n=1}^{N} \hat{\gamma}_{nk}^{(t)}}, \ \hat{p}_k^{(t+1)} = \frac{1}{N} \sum_{n=1}^{N} \hat{\gamma}_{nk}^{(t)}, \\ \hat{\sigma}^{(t+1)} &= \{\frac{1}{N} \sum_{n=1}^{N} \sum_{k=1}^{K} \hat{\gamma}_{nk}^{(t)} (y_n - x_k)^2\}^{1/2} \ . \end{aligned} \tag{10}$$

where $n = 1, \ldots, N$, $k = 1, \ldots, K$, $\hat{\cdot}$ denotes the estimation of a parameter, t means the t-th iteration, γ_{nk} means the probability that the k-th component produces the observation y_n, and $\sum_{n=1}^{N} \sum_{k=1}^{K} \hat{\gamma}_{nk} = 1$. Because of the constraint $\sum_{k=1}^{K} p_k = 1$, a Lagrange multiplier should be added to the log likelihood first, and then setting the derivative w.r.t. p_k to zero to obtain $\hat{p}_k^{(t+1)}$. The iterations will stop when the evaluated log-likelihood becomes converged.

After obtaining the estimation of the parameter set, the average MI of the communication channel is easily computed according to information theory. The form of the average MI $I(X, Y)$ is shown as

$$I(X, Y) = H(Y) - H(Y|X) = H(Y) - H(N), \tag{11}$$

where $H(\cdot)$ denotes the information entropy of a random variable, and

$$H(N) = -\int_{-\infty}^{+\infty} p(n) log(p(n)) dn = \frac{1}{2} log 2\pi e \sigma^2, \tag{12}$$

$$\begin{aligned} H(Y) &= -\int_{-\infty}^{+\infty} p(y)log(p(y))dy \\ &= -\int_{-\infty}^{+\infty} p(y)log\{\sum_{k=1}^{K} p_k \frac{1}{\sqrt{2\pi}\sigma} exp(-\frac{(y-x_k)^2}{2\sigma^2})\}dy \ . \end{aligned} \tag{13}$$

Unfortunately, due to the logarithm of the exponential functions sum, $H(Y)$ generally has no known closed-form solution [29]. 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 in Eq. (13) (e.g. Monte Carlo sampling), and so on. In order to ensure the accuracy and computational cheapness, the Taylor-series expansion method is applied to obtain an appropriate approximation of $log(p(y))$ [29]. Expanding $log(p(y))$ around x_k of each component, the R-order Taylor-series expansion of $log(p(y))$ is acquired as follows:

$$log(p(y)) = \sum_{r=1}^{R} \frac{1}{r!} \frac{d^r}{dy} \{log(p(y))\}(y-x_k)^r|_{y=x_k} + O_R, \tag{14}$$

where O_R is Lagrange remainder term. Therefore,

$$H(Y) \approx -\int_{-\infty}^{+\infty} p(y)\{\sum_{r=1}^{R} \frac{1}{r!} \frac{d^r}{dy} \{log(p(y))\}(y-x_k)^r|_{y=x_k}\}dy, \tag{15}$$

can be solved. 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

$$H(Y) \approx -\sum_{k=1}^{K} p_k\{[log(p(y)) - \frac{1}{2}f(y)]|_{y=x_k}\}, \tag{16}$$

where

$$\begin{aligned} f(y) =& \frac{1}{p(y)\sqrt{2\pi}\sigma} \sum_{i=1}^{K} p_i[\frac{1}{p(y)}(y-x_i)\frac{d}{dy}p(y) + \\ & \frac{1}{\sigma^2}(y-x_i)^2 - 1]exp(-\frac{(y-x_i)^2}{2\sigma^2}) \ . \end{aligned} \tag{17}$$

Then the estimation of the average MI of the communication channel can be computed, that is to say, the amount of side-channel information leakage of the device is obtained.

Note that the parameter K needs to be determined before perform 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 [24], [30], [31], but in this paper this optimum K is the one which leads to the largest MI, i.e.

$$K = \underset{K \in \mathbb{N}_+}{\operatorname{argmax}} \ I(X,Y) \ . \tag{18}$$

Also note that the selection of initial values will affect 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.

Nevertheless, due to the iterations refer to the computation of the Gaussian distribution, the parameter estimation problem may occasionally become an ill-conditioned problem and leads to meaningless results. In this case, an upper bound of the average MI, which is termed the communication channel capacity, can be used to furnish a rough estimation of the leakage amount of the device.

Capacity of the Communication Channel. Denote the capacity as C, it satieties

$$C = max\, I(X,Y) = max[H(Y) - H(N)]\ . \tag{19}$$

C is the convex function of the distribution of x and determined by $\{x_k\}$ and $\{p_k\}$. In practical device, the power of output signal Y is limited. Based on information theory, the average MI of the communication channel will achieve the ultimate value when $\{x_k\}$ takes Gaussian distribution [33]. Now the additive Gaussian channel with discrete input and continuous output becomes the additive Gaussian channel with continuous input and output. The unknown parameters of the channel is also can be calculated by the EM algorithm.

Firstly, classify the observations which are corresponding to a same plain text or cipher text (i.e. T) in a group. Suppose there are m groups and each group has n_i elements, $i = 1, \ldots, m$. Denote the j-th observation in the i-th group as y_{ij}. Since each y_{ij} is acquired individually, they are independent with each other. Recall the communication channel, then we have

$$(y_{ij}|x_i, \sigma) \sim \phi(x_i, \sigma^2),\ j = 1, \ldots, n_i;\ i = 1, \ldots, m, \tag{20}$$

where ϕ means a Gaussian distribution with mean x_i and variance σ^2. Because $\{x_i\}$ takes Gaussian distribution, it can assume that $x_i \sim \phi(\mu, \tau^2)$. Denote $y = \{y_{ij}, j = 1, \ldots, n_i;\ i = 1, \ldots, m\}$, $z = (x_1, \ldots, x_m)$, $N = \sum_{i=1}^{m} n_i$ and $\theta = (\mu, log\,\sigma, log\,\tau)$ is the unknown parameter set. From the Bayesian rules, the following expressions can be found:

$$\begin{aligned} &p(z, \theta|y) = p(z, \theta, y)/p(y) = p(\theta|y, z)p(z|y), \\ &p(z, \theta, y) = p(\theta)p(z|\theta)p(y|z, \theta)\ . \end{aligned} \tag{21}$$

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

$$p(\theta|y, z) \propto p(z, \theta|y) \propto p(z, \theta, y), \tag{22}$$

and then

$$log(p(\theta|y, z)) \propto log(p(z, \theta|y)) \propto log(p(z, \theta, y))\ . \tag{23}$$

Since the prior distribution θ can be considered proportional to τ [35], Eq. (23) can be rewritten as

$$\begin{aligned} log(p(\theta|y,z)) \propto & - Nlog\,\sigma - (m-1)log\,\tau - \frac{1}{2\tau^2}\sum_{i=1}^{m}(x_i-\mu)^2 - \\ & \frac{1}{2\sigma^2}\sum_{i=1}^{m}\sum_{j=1}^{n_i}(x_i-y_{ij})^2 \ . \end{aligned} \tag{24}$$

Consequently, z can be viewed as the latent variable and the EM algorithm can be used [35]. In the E-step, the expectation of Eq. (24) 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 [35], we have

$$(x_i|\theta^{(t)},y) \sim \phi(\upsilon_i^{(t)},\nu_i^{(t)}), \tag{25}$$

where

$$\begin{aligned} \upsilon_i^{(t)} &= [\frac{\mu}{(\tau^{(t)})^2} + \frac{\sum_{j=1}^{n_i} y_{ij}}{(\sigma^{(t)})^2}]/[\frac{1}{(\tau^{(t)})^2} + \frac{n_i}{(\sigma^{(t)})^2}], \\ \nu_i^{(t)} &= [\frac{1}{(\tau^{(t)})^2} + \frac{n_i}{(\sigma^{(t)})^2}]^{-1} \ . \end{aligned} \tag{26}$$

The two formulas in Eq. (26) 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

$$\begin{aligned} \hat{\mu}^{(t+1)} &= \frac{1}{m}\sum_{i=1}^{m}\upsilon_i^{(t)}, \\ \hat{\sigma}^{(t+1)} &= \{\frac{1}{n}\sum_{i=1}^{m}\sum_{j=1}^{n_i}[(y_{ij}-\upsilon_i^{(t)})^2+\nu_i^{(t)}]\}^{1/2}, \\ \hat{\tau}^{(t+1)} &= \{\frac{1}{m-1}\sum_{i=1}^{m}(\upsilon_i^{(t)}-\mu^{(t+1)})^2+\nu_i^{(t)}\}^{1/2} \ . \end{aligned} \tag{27}$$

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 $\tau^2+\sigma^2$. Finally, the channel capacity, which is equal to $I(X,Y)$ at this time, can be obtained through Eq. (11), i.e.

$$\begin{aligned} C &= I(X,Y) = H(Y) - H(N) \\ &= \frac{1}{2}log(1+\frac{\tau^2}{\sigma^2}) \ . \end{aligned} \tag{28}$$

Note that the leakage amount of a device is constant because the leakage is determined for a target intermediate value in an implementation on a device.

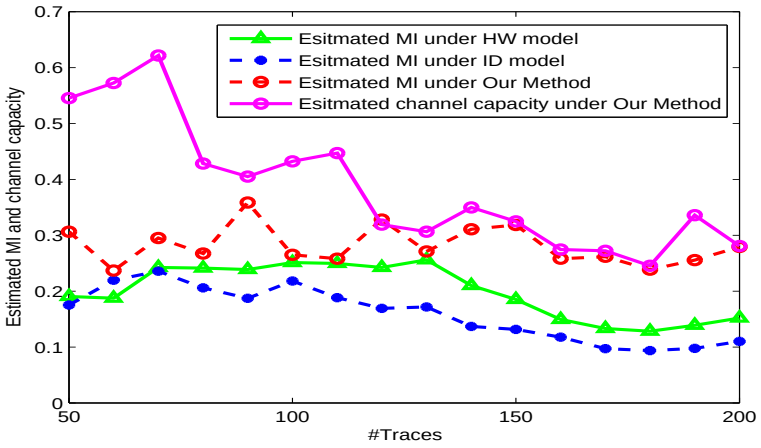


Fig. 2: The estimated MI and channel capacity with Gaussian noise.

Therefore, C just describes an upper bound of the leakage amount of a device, i.e. the leakage amount of a device at the worst case scenario. The average MI may achieve C, or maybe not. By the way, K in section 3.2, needs not to select since m is the component number. Because there is no computation of the Gaussian pdf in the iterations, the results will be always valid.

Fig. 2 shows an example of leakage characterizing about an unprotected AES-128 implementation on 8-bit Micro Controller Unit (MCU). The measured power leakage is corresponding to the 9^{th} S-box output in 1^{st} round and assumed it is contaminated by an additive Gaussian noise. Two different leakage models, Hamming Weight (HW) and Identification (ID) models, were used to estimate the information amount which the evaluator can utilize. First, suppose the leakage of the device takes HW or ID model, then computing the hypothesis leakage of a target sensitive variable as the key is known. At last, the MI value between the hypothesis and measured leakage are estimated by using kernel density estimators [32]. The MI values under HW and ID model assumption are shown in Fig. 2 (the green line and the blue line).

It can be seen that HW model outperform ID model, which is anastomotic with the attack results (see Fig. 7, Appendix A). Due to the data structure and the insufficient estimated precision with a fewer traces, the estimated MI under the two models and the capacity become a little smaller when the trace number increases. But the difference of MI between the POIs and other time samples is increasing (see Fig. 8, Appendix A). The difference of the two MI values shows that they are just reflect the utilization rate of the evaluator, rather than the real power leakage amount of the device. The average MI of the channel (the pinkish red line in Fig. 2) can characterize the real power leakage amount of the device because it is only rely on the measured leakage. And the channel capacity (the red line in Fig. 2) characterizes 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 the leakage of the device takes the Gaussian distribution and the noise are Gaussian noise. With the increasing trace number, the estimated precision is also increasing and finally MI reach C. The result confirms the assumption of the Gaussian noise is reasonable.

3.3 Analysis on the Communication Channel with Non-Gaussian Noise

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 the 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 approach the noise, then $p(n)$ can be described as

$$p(n) = \sum_{m=1}^{M} \alpha_m \frac{1}{\sqrt{2\pi}\delta_m} exp(-\frac{n^2}{2\delta_m^2}), \tag{29}$$

where δ_m^2 is the variance of the m-th component. When $M = 1$, $p(n)$ reduce to Gaussian noise. Eqs. (6) and (7) can be respectively rewritten as

$$p(y \mid x_k) = p(n)|_{n=y-x_k} = \sum_{m=1}^{M} \alpha_m \frac{1}{\sqrt{2\pi}\delta_m} exp(-\frac{(y-x_k)^2}{2\delta_m^2}), \tag{30}$$

and

$$p(y) = \sum_{k=1}^{K} p(y|x_k)p_k = \sum_{k=1}^{K} p_k \sum_{m=1}^{M} \alpha_m \frac{1}{\sqrt{2\pi}\delta_m} exp(-\frac{(y-x_k)^2}{2\delta_m^2}) \ . \tag{31}$$

Therefore, the log-likelihood function of the GMM is

$$\sum_{n=1}^{N} log(p(y_n)) = \sum_{n=1}^{N} log\{\sum_{k=1}^{K} p_k [\sum_{m=1}^{M} \alpha_m \frac{1}{\sqrt{2\pi}\delta_m} exp(-\frac{(y-x_k)^2}{2\delta_m^2})]\} \ . \tag{32}$$

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. Setting the derivative of $log(p(y))$ w.r.t. each parameter to zero, the iterations in E-step is obtained as

$$\begin{aligned} \hat{\gamma}_{nk}^{(t)} &= \frac{p(y_n|x_k)p_k}{\sum_{k=1}^{K} p(y_n|x_k)p_k}, \\ \hat{\beta}_{nm}^{(t)} &= \frac{\sum_{k=1}^{K} p_k \alpha_m \frac{1}{\sqrt{2\pi}\delta_m} exp(-\frac{(y-x_k)^2}{2\delta_m^2})}{\sum_{k=1}^{K} p(y_n|x_k)p_k}, \end{aligned} \tag{33}$$

and the M-step is

$$\begin{aligned} \hat{x}_k^{(t+1)} &= \frac{\sum_{n=1}^{N} \hat{\gamma}_{nk}^{(t)} y_n}{\sum_{n=1}^{N} \hat{\gamma}_{nk}^{(t)}}, \ \hat{\alpha}_m^{(t+1)} = \frac{1}{N} \sum_{n=1}^{N} \hat{\beta}_{nm}^{(t)}, \\ \hat{p}_k^{(t+1)} &= \frac{1}{N} \sum_{n=1}^{N} \hat{\gamma}_{nk}^{(t)}, \qquad \hat{\delta}_m^{(t+1)} = \{ \frac{1}{\sum_{n=1}^{N} \hat{\beta}_{nm}^{(t)}} \sum_{n=1}^{N} \hat{\beta}_{nm}^{(t)} (y_n - x_k)^2 \}^{1/2}, \end{aligned} \tag{34}$$

where $n = 1, \ldots, N$, $k = 1, \ldots, K$, $m = 1, \ldots, M$. Because of the constraint $\sum_{k=1}^{K} p_k = 1$, a Lagrange multiplier should be added to the log likelihood first, and then setting the derivative w.r.t. p_k to zero to obtain $\hat{p}_k^{(t+1)}$. $\hat{\alpha}_m^{(t+1)}$ is also obtained by using the same process.

Afterwards, $H(N)$ can be approximated by the Taylor-series expansion method 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. The iterations also refer to the computation of the Gaussian distribution, which may occasionally lead to meaningless results. In this case, 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. It is much easier to provide a bound of C.

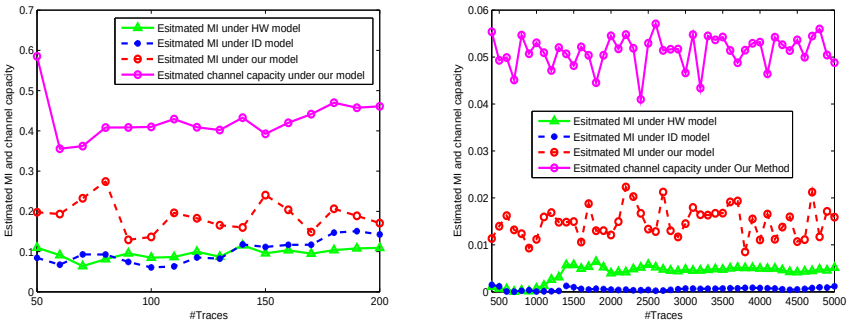
Denote σ^2 as the noise variance and σ_n^2 the entropy power of noise. The channel capacity C satisfies [36]

$$\frac{1}{2} log(1 + \frac{\sigma_x^2}{\sigma^2}) \leq C \leq \frac{1}{2} log(\frac{\sigma^2 + \sigma_x^2}{\sigma_n^2}) \tag{35}$$

if the input power $E(x^2) \leq \sigma_x^2$, where σ_n^2 satisfies $H(N) = \frac{1}{2} log(2\pi e \sigma_n^2)$. Therefore, the right term of Eq. (35) can be seen as an upper bound of the average MI of the channel.

Fig. 3(a) describes an example of leakage characterizing about an unprotected AES-128 implementation on 8-bit MCU. The measured electromagnetic emanation leakage is corresponding to the 9^{th} S-box output of 1^{st} round and it is contaminated by an additive non-Gaussian noise. HW and ID models were also used to estimate the information amount which the evaluator can employ. It can be observed that HW model outperforms ID model, which is anastomotic with the attack results (see Fig. 7, Appendix A). Due to the non-Gaussian noise, the average MI of the channel is not close to C with the increasing trace number and the leakage of the device does not achieve the worst case.

Fig. 3(b) depicts an example of leakage characterizing about an AES-128 implementation with boolean masking on a smart card. The leakage have been preprocessed [34] to make the 1^{st}-order leakage information be exposed. It has similar result as Fig. 2. Due to the inevitable loss of preprocessing [34], the information utilization rate of the evaluator are lower than the device with 1^{st}-order leakage.



(a) The estimated MI and channel capacity with non-Gaussian noise (an unprotected AES-128 implementation on 8-bit MCU)

(b) The estimated MI and channel capacity with non-Gaussian noise (a protected AES-128 implementation on smart card)

Fig. 3: The estimated MI and channel capacity with non-Gaussian noise.

4 Leakage Detection Based on Consistency Check

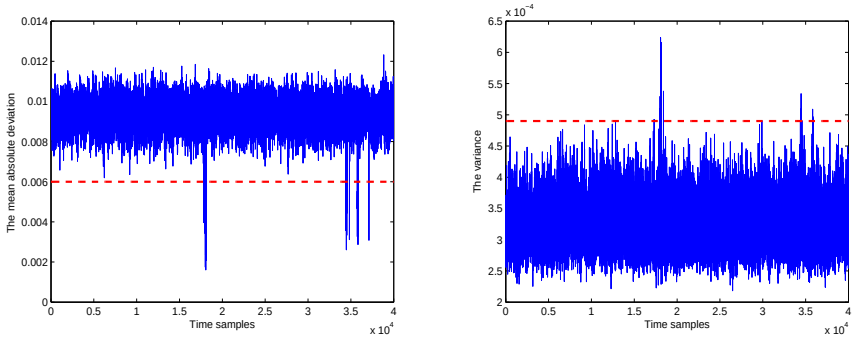
On the basis of the above analysis, a leakage detection method is developed. It is based on the concept of consistency check in statistics, i.e. a parameter estimated by the data at the leakage points, especially POIs, is a consistent estimator, but the estimation computing by the non-leakage data is not. Certainly, the proposed leakage detection method do not need to mount side channel attack. The method is detailed in the following.

Since the communication channel model is especially established for the leakage signal, the model should be not suitable for other non-leakage points. Based on this judgement, some estimated parameters of the communication channel by using the data at a leakage point should be consistent estimators. 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. Therefore, the methods for computing the average MI and capacity of the channel in Section 3.2, are both suitable for leakage detection. The estimated parameters to check consistency can be σ in Eq. (10), or σ, μ, or τ in Eq. (27). The concept of consistency [35] 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.

$$\lim_{n\to\infty} Pr\{|\hat{\theta} - \theta| < \varepsilon\} = 1, \ \forall \varepsilon > 0 \ . \tag{36}$$

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 consistency check. The scheme computes the standard deviation of the values of $\hat{\theta}$ in all iterations of the whole EM algorithm, and considers a leakage



(a) The leakage detection result of the proposed method with 1,000 traces

(b) The leakage detection result of the variance based method with 1,000 traces

Fig. 4: 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).

point should have a much less standard deviation than non-leakage points. It can reach 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.

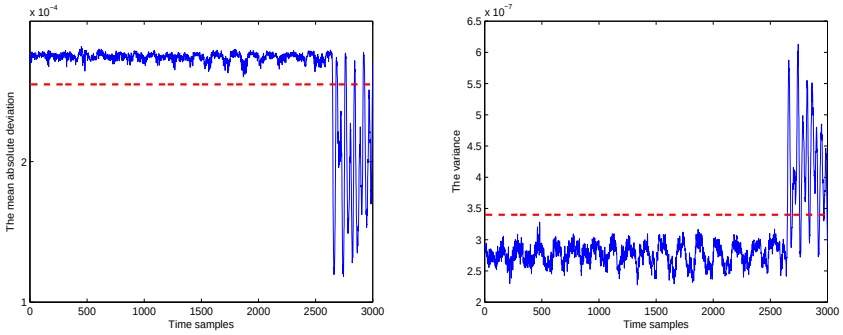
Furthermore, to obtain a more steady result, the mean absolute deviation [37] of the values of $\hat{\theta}$ is recommended to replace the standard deviation in this paper. The form of the mean absolute deviation of the samples in Z is

$$d_n = \frac{1}{n}\sum_{i=1}^{n}|Z_i - \bar{Z}|, \tag{37}$$

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 the d_n at a sample point is lower than the threshold, the sample point is viewed as a leakage point.

Some practical experiments verified the effectiveness of the proposed method (Figs. 4-6). The acquired leakage is all power consumptions of crypto devices. The compared algorithm is the variance detection technique proposed by [22], which also have no special requirements. 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. (27). In all figures, the dash line means the selected 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 the leakage points. Fig. 4 provides the power leakage points of an unprotected AES-128 implementation on an 8-bit MCU by using the proposed method and the variance detection technique [22], respectively. Fig. 5 shows the leakage points of an unprotected AES-128 implementation on FPGA by using the proposed method and the variance detection, respectively.

In Figs. 4 and 5, the leakage points found by the two methods are almost exactly the same obtained by correlation power analysis (CPA) (see Fig. 9, Ap-



(a) The leakage detection result of the proposed method with 30,000 traces

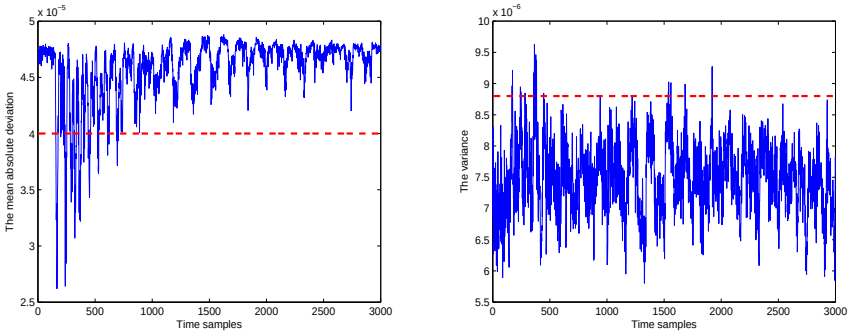
(b) The leakage detection result of the variance based method with 30,000 traces

Fig. 5: 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).

pendix A). These leakage points corresponding to the 1^{st} S-box output for MCU implementation or the exclusive or value between the 1^{st} S-box input and output for FPGA implementation. That is, the leakage points detected by the methods are POIs, which can be used to mount an attack direct, while the detection based T-test methods often detect many useless leakage points for attacking [18]. The proposed method perform better than the compared method. In Fig. 4, the proposed method find all POIs, but the compared method not. Moreover, the difference between POIs and other points in Fig. 5(a) is larger than Fig. 5(b). However, sometimes the leakage points of some bytes found by the two methods may contain few POIs. For instance, the operation "ShiftRows" will mask the POIs of some bytes in the last round of AES (see Fig. 10, Appendix A).

Fig. 6 shows the leakage points of the same FPGA implementation by using the two methods, respectively. The proposed method perform much better than the variance based method. Besides, the method cannot detect the POIs of a masking implementation (e.g. a masked AES-128 implementation, see Fig. 11, Appendix A), unless the traces have been preprocessed before detection (see Fig. 12, Appendix A). Furthermore, similar results will be obtained if other parameters (e.g. σ in Eq. (10)) are used and they are no longer shown here.

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 [36], that is, the proposed method can works even the device has the minimum bound of leakage amount. It can be seen that the empirical thresholds needs to be determined first for the two methods. It is not convenient compare with the detection based T-test methods, but the two methods have no requirements about the leakage acquisition. Besides, the two method can find the POIs in traces and introduce few leakage points cannot be utilized in an attack. And the proposed method perform better and more stable



(a) The leakage detection result of the proposed method with 30,000 traces

(b) The leakage detection result of the variance based method with 30,000 traces

Fig. 6: 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).

than the variance based method.

5 Extension

Multiple Leakage Points Analysis. If there exist multiple leakage points in the measured traces, the input becomes an extended source, and each leakage point corresponds to one communication channel. The entropy of the input and the average MI will increase, which explains why multi-points SCAs outperform the single point SCAs [38] from the view of information theory.

Leakage Modeling. It is also possible to perform leakage modeling if the key is known. It is the byproduct of the leakage characterizing. Because γ_{nk} in Eq. (10) 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 [39]. When the key is known, the intermediate value is also known, hence the leakage model can be obtained.

Collision Attack. After determining the leakage value corresponding to each T by utilizing a match algorithm, if the key is unknown, a novel collision attack can be developed to recover the key. Take AES-128 as an example, divide the whole plain text or cipher text into 16 bytes and each byte corresponds to a T. The leakage value corresponding 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.

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. (10), Eq. (27) or Eq. (34), the noise can be characterized and reduced for the measured leakage. Moreover, SNR can be estimated, too.

6 Conclusion

In this paper, we revisited the information theoretic metric in a communication channel model, and proposed two objective metrics (i.e. the average MI and capacity of the channel) to characterize the leakage amount of a crypto device and its upper bound through communication theory. Furthermore, we investigated the MI of the measured leakage under different kinds of noises and the largest leakage amount of the device. On the basis of the communication channel model, we also proposed a novel leakage detection method. In the future, we will continue the research of the four byproducts mentioned in Section 5, and to investigate the leakage detection on the measured leakage of masking.

References

1. Kocher, P.: Timing Attacks on Implementations of Diffie-Hellman, RSA, DSS and Other Systems. In: Koblltz, N. (ed.) Advances In Cryptology-CRYPTO '96. LNCS, vol. 1109, pp. 104–113. Springer, Heidelberg (1996)
2. Brier, E., Clavier, C., Olivier, F.: Correlation Power Analysis with a Leakage Model. In: Joye M., Quisquater J.-J. (eds.) CHES 2004, LNCS, vol. 3156, pp. 16–29. Springer, Heidelberg (2004)
3. Gandolfi, K., Mourtel, C., Olivier, F.: Electromagnetic Analysis: Concrete Results. In: Koç, Ç. K., Naccache, D., Paar, C. (eds.) CHES 2001. LNCS, vol. 2162, pp. 251–261. Springer, Heidelberg (2001).
4. Carlet, C., Prouff, E., Rivain, M., Roche T.: Algebraic Decomposition for Probing Security. In: Gennaro, R., Robshaw, M. (eds.) CRYPTO 2015. LNCS, vol. 9215, pp. 742–763. Springer, Heidelberg (2015)
5. Standaert, F.-X., Malkin, T.G., Yung, M.: A Formal Practice-Oriented Model For The Analysis of Side-Channel Attacks. IACR e-print Archive, 2006/134, 2006.
6. Gierlichs, B., Batina, L., Tuyls, P., Preneel, B.: Mutual Information Analysis. In: Oswald, E., Rohatgi, P. (eds.) CHES 2008. LNCS, vol. 5154, pp. 426–442. Springer, Heidelberg (2008)
7. Standaert, F.-X., Malkin, T.G., Yung, M.: A Unified Framework for The Analysis of Side-channel Key Recovery Attacks. In: Joux, A. (ed.) EUROCRYPT 2009. LNCS, vol. 5479, pp. 443C-461. Springer, Heidelberg (2009)
8. Fei, Y., Luo, Q., Ding, A. A.: A Statistical Model for DPA with Novel Algorithmic Confusion Analysis. In: Prouff, E., Schaumont, P. (eds.) CHES 2012. LNCS, vol. 7428, pp. 233–250. Springer, Heidelberg (2012)
9. Veyrat-Charvillon, N., Gérard, B., Standaert, F.-X.: Security Evaluations Beyond Computing Power. In: Johansson, T., Nguyen, P.Q. (eds.) EUROCRYPT 2013. LNCS, vol. 7881, pp. 126–141. Springer, Heidelberg (2013)
10. Duc, A., Faust, S., Standaert, F.-X.: Making Masking Security Proofs Concrete (or How to Evaluate the Security of any Leaking Device). In: Oswald, E., Fischlin, M. (eds.) Eurocrypt 2015. LNCS, vol. 9056, pp 401–429. Springer, Heidelberg (2015)
11. Martin, D.P., O'Connell, J.F., Oswald, E., Stam, M.. Counting Keys in Parallel After a Side Channel Attack. In: Iwata, T., Cheon, J.H. (eds.) ASIACRYPT 2015. LNCS, vol. 9453, pp. 313–337. Springer, Heidelberg (2015)
12. Lomné, V., Prouff, E., Rivain, M., Roche, T., Thillard, A.: How to Estimate the Success Rate of Higher-Order Side-Channel Attacks. In: Batina, L., Robshaw, M. (eds.) CHES 2014. LNCS, vol. 8731, pp. 35–54. Springer, Heidelberg (2014)
13. Mangard, S.: Hardware Countermeasures against DPA? A Statistical Analysis of Their Effectiveness. In: Okamoto, T. (ed.) CT-RSA 2004. LNCS, vol. 2964, pp. 222–235. Springer, Heidelberg (2004)
14. Durvaux, F., Standaert, F.-X., Veyrat-Charvillon, N.: How to Certify the Leakage of a Chip? In: Nguyen P. Q., Oswald, E. (eds.) EUROCRYPT 2014, LNCS, vol. 8441, pp. 459–476. Springer, Heidelberg (2014)
15. Renauld, M., Standaert, F.-X., Veyrat-Charvillon, N., Kamel, D., Flandre, D.: A Formal Study of Power Variability Issues and Side-Channel Attacks for Nanoscale Devices. In: Paterson, K.G. (ed.) EUROCRYPT 2011. LNCS, vol. 6632, pp. 109C128. Springer, Heidelberg (2011)
16. Cooper, J., DeMulder, E., Goodwill, G., Jaffe, J., Kenworthy, G., Rohatgi, P.: Test Vector Leakage Assessment (TVLA) Methodology in Practice (Extended Abstract). ICMC 2013. `http://icmc-2013.org/wp/wp-content/uploads/2013/09/goodwillkenworthtestvector.pdf`

17. Goodwill, G., Jun, B., Jaffe, J., Rohatgi, P.: A Testing Methodology for Side Channel Resistance Validation. NIST Non-Invasive Attack Testing Workshopp (2011). http://csrc.nist.gov/news_events/non-invasive-attack-testing-workshop/papers/08_Goodwill.pdf
18. Durvaux, F, Standaert, F.-X.: From Improved Leakage Detection to The Detection of Points of Interests in Leakage Traces. In: Fischlin, M. and Coron, J.-S. (eds.) EUROCRYPT 2016. LNCS, vol. 9665, pp. 240–262. Springer, Heidelberg (2013)
19. Mather, L., Oswald, E., Bandenburg, J., Wójcik, M.: Does My Device Leak Information? A Priori Statistical Power Analysis of Leakage Detection Tests. In: Sako, K., Sarkar, P. (eds.) ASIACRYPT 2013. LNCS, vol. 8269, pp. 486–505. Springer, Heidelberg (2013)
20. Schneider, T., Moradi, A.: Leakage Assessment Methodology. In: Güneysu, T., Handschuh, H. (eds.) CHES 2015. LNCS, vol. 9293, pp. 495–513. Springer, Heidelberg (2015)
21. Ding, A.A., Chen, C., Eisenbarth, T.: Simpler, Faster, and More Robust T-test Based Leakage Detection. Cryptology ePrint Archive, Report 2015/1215 (2015), http://eprint.iacr.org/2015/1215.pdf
22. Moradi, A., Guilley, S., Heuser, A.: Detecting Hidden Leakages. In: Boureanu I., Owesarski P., Vaudenay S. (eds.) ACNS 2014. LNCS, vol. 8479, pp. 324–342. Springer, Heidelberg (2014)
23. Mixture Models for the Analysis of Gene Expression, http://www.diss.fu-berlin.de/diss/receive/FUDISS_thesis_000000003441
24. Chen, X.-F., Liu, X.-B., Jia, Y.-D.: Unsupervised Selection and Discriminative Estimation of Orthogonal Gaussian Mixture Models for Handwritten Digit Recognition. In: 10th International Conference on Document Analysis and Recognition (ICDAR 2009), pp. 1151–1155.
25. McLachlan, G.J., Peel D.: Finite Mixture Models. Wiley Series in Probability and Statistics. Wiley, New York, 2000
26. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum Likelihood from Incomplete Data via the EM Algorithm. Journal of the Royal Statistical Society 39, 1–38 (1977)
27. Heuser, A., Rioul, O., Guilley, S.: Good is Not Good Enough. In: Batina, L., Robshaw, M. (eds.) CHES 2014. LNCS, vol. 8731, pp. 55–74. Springer, Heidelberg (2014)
28. Bose, R.: Information Theory, Coding and Cryptography, Second Edition. Tata McGraw-Hill Education Pvt. Ltd. (2008)
29. Huber M.F., Bailey, T., Hugh D.-W., Hanebeck, U. D.: On Entropy Approximation for Gaussian Mixture Random Vectors. In: MFI 2008, pp. 181–188. IEEE (2008)
30. Zhang, M.-H., Cheng Q.-S.: Determine the Number of Components in a Mixture Model by the Extended KS Test. Pattern Recognition Letters 25, 211–216 (2004)
31. Xie, C.-H., Chang, J.-Y., Liu, Y.-J.: Estimating the Number of Components in Gaussian Mixture Models Adaptively. Journal of Information & Computational Science 10, 4453–4460 (2013)
32. Moon, Y.I., Rajagopalan, B., Lall, U.: Estimation of Mutual Information Using Kernel Density Estimators. Phys Rev E Stat Phys Plasmas Fluids Relat Interdiscip Topics 52, 2318–2321 (1995)
33. Wei, L.: 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
34. Dabosville, G., Doget, J., Prouff, E.: A New Second-order Side Channel Attack Based on Linear Regression. IEEE Transactions on Computers 62, 1629–1640 (2013)
35. Mao, S.-S., Wang, J.-L., Pu, X.-L.: Advanced Mathematical Statistics (in Chinese), Second Edition. Higher Education Press, Beijing (2009)

36. Mceliece, R.J.: The Theroy of Information and Coding, Second Edition. Cambridge University Press (2002)
37. Tanner, M. A.: Tools for Statistical Inference: Methods for the Exploration of Posterior Distributions and Likelihood Functions. Springer-Verlag, NewYork (1993)
38. Mather, L., Oswald, E., Whitnall, C.: Multi-target DPA Attacks: Pushing DPA Beyond the Limits of a Desktop Computer. In: P. Sarkar and T. Iwata (eds.) ASIACRYPT 2014. LNCS 8873, pp. 243–261, Springer, Heidelberg (2014)
39. Kuhn, H.W.: The Hungarian Method for the assignment problem. Naval Research Logistics Quarterly 2, 83C-97 (1955)

Appendix: Additional Figures

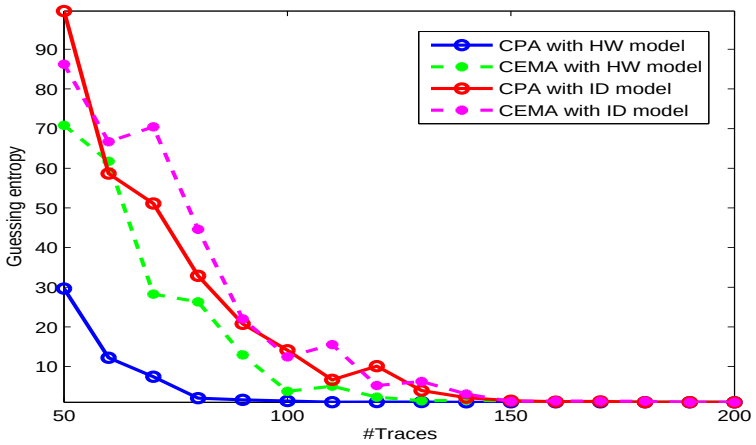
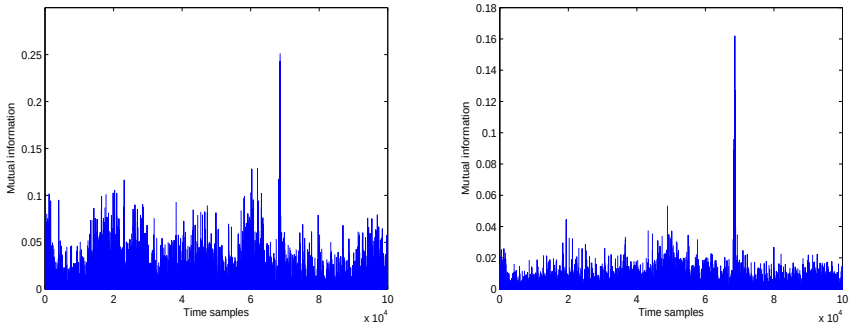
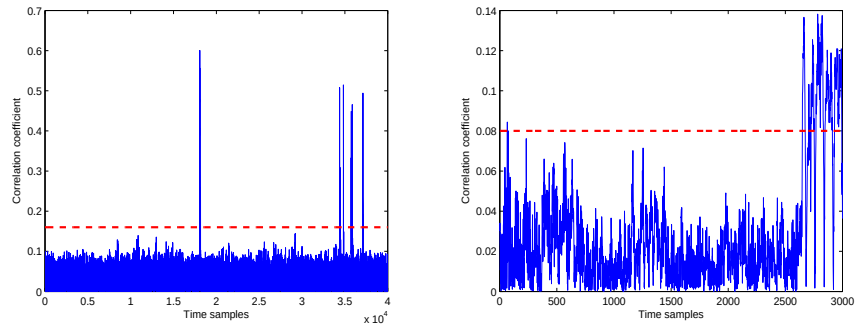


Fig. 7: The guessing entropy of CPA and CEMA under HW and ID model on an unprotected AES-128 implemented on an 8-bit MCU (the target intermediate value is 9^{th} S-box output of 1^{st} round).



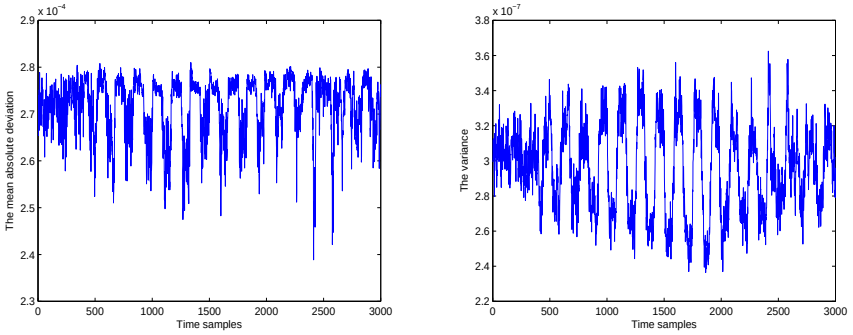
(a) The estimated MI with 100 traces (b) The estimated MI with 200 traces

Fig. 8: The estimated MI with different trace numbers of an unprotected AES-128 implementation on an 8-bit MCU (the target intermediate value is 1^{st} S-box output of 1^{st} round).



(a) POIs of the implementation on MCU with 1,000 traces (b) POIs of the implementation on FPGA with 30,000 traces

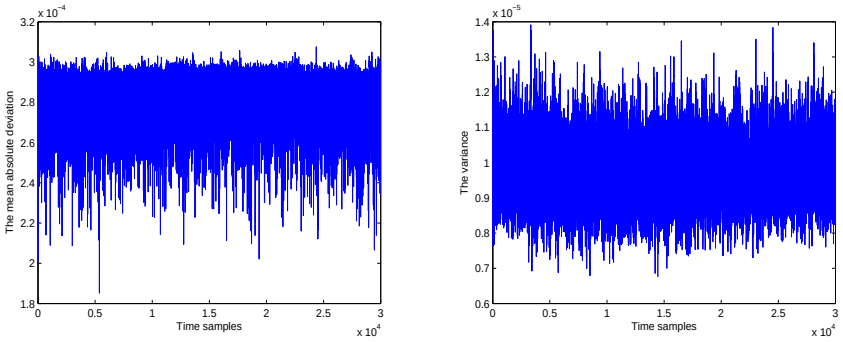
Fig. 9: POIs of an unprotected AES-128 implementation on MCU (the target intermediate value is 1^{st} S-box output of 1^{st} round) and FPGA (the target intermediate value is the exclusive or value between the 1^{st} S-box input and output of the last round) found by CPA.



(a) The leakage detection result of the proposed method with 30,000 traces

(b) The leakage detection result of the variance based method with 30,000 traces

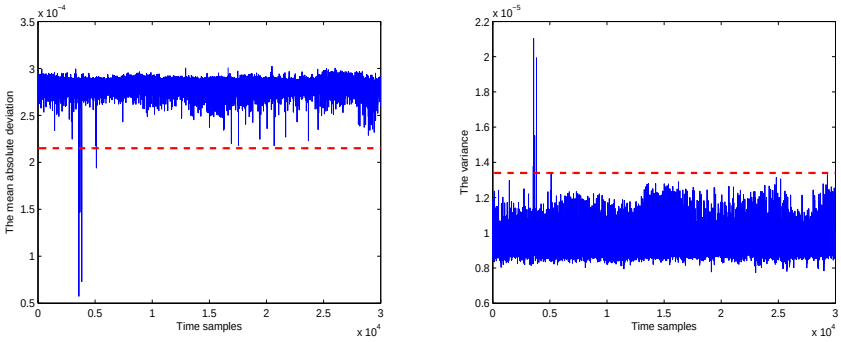
Fig. 10: The leakage detection results of an unprotected AES-128 implementation on FPGA found by the proposed method and the variance based method (classifying traces according to 2^{st} bytes of the cipher texts).



(a) The leakage detection result of the proposed method with 5,000 traces

(b) The leakage detection result of the variance based method with 5,000 traces

Fig. 11: 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 leakage detection result of the proposed method with 5,000 traces

(b) The leakage detection result of the variance based method with 5,000 traces

Fig. 12: The leakage detection results of a masked AES-128 implementation on a smart card found by the proposed method and the variance based method (after preprocessing, and classifying traces according to 1^{st} bytes of the plain texts).