Template Attacks Based On Priori Knowledge

Guangjun Fan[1], Yongbin Zhou[2], Hailong Zhang[2], Dengguo Feng[1]

[1] Trusted Computing and Information Assurance Laboratory,
Institute of Software, Chinese Academy of Sciences
guangjunfan@163.com, feng@tca.iscas.ac.cn
[2] State Key Laboratory of Information Security,
Institute of Information Engineering, Chinese Academy of Sciences
zhouyongbin@iie.ac.cn, zhanghailong@iie.ac.cn

Abstract. Template Attacks are widely accepted as the *strongest* side-channel attacks from the information theoretic point of view, and they can be used as a very *powerful* tool to evaluate the physical security of cryptographic devices. Template Attacks consist of two stages, the profiling stage and the extraction stage. In the profiling stage, the attacker is assumed to have a large number of power traces measured from the reference device, using which he can accurately characterize signals and noises in different points. However, in practice, the number of profiling traces may not be sufficient. In this case, signals and noises are not accurately characterized, and the key-recovery efficiency of Template Attacks is significantly influenced. We show that, the attacker can still make Template Attacks powerfully enough in practice as long as the priori knowledge about the reference device be obtained. We note that, the priori knowledge is just a prior distribution of the signal of the instantaneous power consumption. The attacker can easily obtain the priori knowledge from his previous experience of conducting Template Attacks, from Internet and so on. Evaluation results show that, the priori knowledge, even if not accurate, can still help increase the power of Template Attacks, which poses a serious threat to the physical security of cryptographic devices.

Keywords: Side-Channel Attacks, Power Analysis Attacks, Template Attacks, Priori Knowledge.

1 Introduction

Template Attacks were proposed by S. Chari et al. in 2002 [1], which consist of two stages, i.e. the profiling stage and the extraction stage. In the profiling stage, the attacker has a reference device identical or similar to the target device, and he can use the reference device to characterize the leakage of the target device. In the extraction stage, the attacker can exploit a small number of power traces measured from the target device to recover the correct (sub)key. In order to make Template Attacks powerfully enough, the attacker needs to use a large number of power traces to accurately characterize signals and noises in different interesting points. However, in practice, the number of profiling traces may be limited. For example, a common countermeasure is to limit the operation times of the reference device, or the key used by the reference device will be refreshed after being used several times. In these scenarios, the attacker can only obtain a limited number of power traces in the profiling stage, and signals and noises are not accurately characterized, which significantly influences the key-recovery efficiency of Template Attacks.

1.1 Motivations

A natural question is whether or not it is possible to further increase the power of Template Attacks even if the number of profiling traces is limited? We anticipate that using the priori knowledge about the reference device may be a possible way. The priori knowledge is just a kind of prior distribution of the actual value of the signal component in the instantaneous power consumption. There are many ways that the attacker can obtain the priori knowledge in practice. We show three typical examples here.

Example 1: Assume that the attacker has characterized the power leakages of some cryptographic devices whose leakage characterizations are similar to the reference device. Then, he may obtain the priori knowledge about the reference device. For example, noises in different interesting points are usually assumed to follow the normal distribution. If the attacker can estimate the mean value and the variance of the normal distribution using power traces measured from previous cryptographic devices, then the priori knowledge about the reference device can be obtained.

Example 2: From Internet (e.g. [18,19]), the attacker may obtain some power traces or other potential useful information (e.g. Signal-to-Noise Ratio) of different devices which are similar to the reference device, using which he can infer the priori knowledge of the reference device (similarly to Example 1).

Example 3: For a sophisticated attacker, after obtaining power traces from the reference device in the profiling stage, he can use the power traces to obtain an interval estimation of the actual value of the signal component and roughly infer the prior distribution is a kind of distribution (e.g. normal distribution or uniform distribution) over the interval.

To sum up, for a seasoned attacker, it is not only reasonable but also realistic for him to possess the priori knowledge about the reference device from a practical point of view. Therefore, we need to consider the power of Template Attacks when the attacker can not obtain enough power traces from the reference device in the profiling stage *but* has the priori knowledge about the reference device. Specifically, two questions need to be answered. The first question is how can the attacker exploit the priori knowledge during the profiling stage in a theoretically correct and practically feasible way to make Template Attacks more powerful (i.e. achieve better classification performance)? The second question is whether or not the priori knowledge (even if it may not be very accurate) will make Template Attacks more powerful really?

Of course, one may ask such question: Why not the attacker exploits the power traces obtained from the similar devices (from his previous experience of conducting Template Attacks or from Internet) together with the power traces obtained from the reference device to build the templates to make Template Attacks more powerful? In fact, if one *directly* exploits power traces from the similar devices and the reference device to build the templates, the classification performance of Template Attacks will be decreased [21]. The reason is that the acquisition campaigns about the devices are different[1] even if the leakage distributions of the similar devices and the reference device are similar [21].

If we can give positive answers to the above two important questions, then in order to make Template Attacks more powerful in the above scenarios, the attacker can first extract the priori knowledge from the power traces obtained from the different but similar devices and then conduct Template Attacks with the priori knowledge as well as the limited power traces obtained from the reference device. From this point of view, these two questions are worth researching.

1.2 Contributions

Main contributions of this paper are two-folds. Firstly, based on the method of Bayes estimation [13], we give a theoretically correct and practically feasible way of exploiting the priori knowledge when the attacker conducts Template Attacks with limited power traces obtained from the reference device in the profiling stage.

Secondly, we verify our way of exploiting the priori knowledge using both simulated and practical experiments. Evaluation results show that, Template Attacks will be more powerful if the attacker can possess accurate priori knowledge. Additionally, the more accurate the priori knowledge is, the more powerful Template Attacks will be. Therefore, with the priori knowledge we can further increase the power of Template Attacks.

1.3 Related Work

Paper [2] provided answers to practical issues of Template Attacks, such as how to choose interesting points in an efficient way and how to preprocess noisy data. Paper [4] proposed efficient methods to avoid possible numerical obstacles when implementing Template Attacks. Paper [10] presented a variant of Template Attacks which can be applied to block ciphers when the plaintext and ciphertext are unknown. In [7], Template Attacks were used to attack a mask implementation. Recently, a simple pre-processing technique of Template Attacks, normalizing the sample values using the means and variances was evaluated [6]. Paper [20] showed how to best evaluate profiling and extraction of profiled attacks by using the information theoretic metric and the security metric. Principal Component Analysis (PCA)-Based Template Attacks were investigated in [3]. However, this kind of Template Attacks may not improve the classification performance [6]. Therefore, PCA-Based Template Attacks are not widely used in practice. LDA-based Template Attacks were introduced in [8]. This kind of Template Attacks

[1] For example, there exist offsets in the different acquisition campaigns.

depends on the condition of equal covariances [4] (Please see Section 2.1 for more details.), which does not hold in most settings. Therefore, it is not a better choice compared with PCA-based Template Attacks [4]. However, no previous work considered our important questions.

1.4 Organization of This Paper

The rest of this paper is organized as follows. In Section 2, we review the concept of Template Attacks and Bayes estimation. In Section 3, we give a reasonable way of exploiting the priori knowledge to make Template Attacks more powerful. In Section 4, we verify the way of exploiting the priori knowledge by both simulated and practical experiments. In Section 5, we conclude the whole paper.

2 Preliminaries

Template Attacks mainly include: classical Template Attacks [1] and reduced Template Attacks (pp. 108 in [9]). In this section, we briefly review these two kinds of Templates Attacks and the method of Bayes estimation.

2.1 Classical Template Attacks

We will introduce the two stages of classical Template Attacks: the profiling stage and the extraction stage.

2.1.1 The Profiling Stage Assume that there exist K different (sub)keys $key_i, i = 0, 1, \ldots, K-1$ which need to be classified. Also, there exist K different key-dependent operations O_i, $i = 0, 1, \ldots, K-1$. Usually, one will generate K templates, one for each key-dependent operation O_i. One can exploit some methods to choose N interesting points $(P_0, P_1, \ldots, P_{N-1})$. The interesting points are those time samples that contain the most information about the characterized key-dependent operations. Each template is composed of a mean vector and a covariance matrix. The mean vector is used to estimate the signal component of side-channel leakages. It is the average signal vector $\mathbf{M}_i = (M_i[P_0], \ldots, M_i[P_{N-1}])$ for each one of the key-dependent operations. The covariance matrix is used to estimate the probability density of the noise component at different interesting points. It is assumed that noises at different interesting points approximately follow the multivariate normal distribution. A N dimensional noise vector $\mathbf{n}_i(\mathbf{S})$ is extracted from each actual power trace $\mathbf{S} = (S[P_0], \ldots, S[P_{N-1}])$ representing the template's key dependency O_i as $\mathbf{n}_i(\mathbf{S}) = (S[P_0] - M_i[P_0], \ldots, S[P_{N-1}] - M_i[P_{N-1}])$. One computes the $(N \times N)$ covariance matrix $\mathbf{C}_i$ from these noise vectors. The probability density of the noises occurring under key-dependent operation O_i is given by the N dimensional multivariate Gaussian distribution $p_i(\cdot)$, where the probability of observing a noise vector $\mathbf{n}_i(\mathbf{S})$ is:

$$p_i(\mathbf{n}_i(\mathbf{S})) = \frac{1}{\sqrt{(2\pi)^N |\mathbf{C}_i|}} exp\Big(-\frac{1}{2}\mathbf{n}_i(\mathbf{S})\mathbf{C}_i^{-1}\mathbf{n}_i(\mathbf{S})^T \Big) \quad \mathbf{n}_i(\mathbf{S}) \in \mathbb{R}^N. \qquad (1)$$

In equation (1), the symbol $|\mathbf{C}_i|$ denotes the determinant of $\mathbf{C}_i$ and the symbol $\mathbf{C}_i^{-1}$ denotes its inverse.

2.1.2 The Extraction Stage

Assume that one obtains t actual power traces (denoted by $\mathbf{S}_1, \mathbf{S}_2, \ldots, \mathbf{S}_t$) from the target device in the extraction stage. When the power traces are statistically independent, one will apply maximum likelihood approach on the product of conditional probabilities (pp. 156 in [9]), i.e.

$$key_{ck} := argmax_{key_i}\Big\{\prod_{j=1}^{t}\mathsf{Pr}(\mathbf{S}_j|key_i), i = 0, 1, \ldots, K-1\Big\},$$

where $\mathsf{Pr}(\mathbf{S}_j|key_i) = p_{f(\mathbf{S}_j,key_i)}(n_{f(\mathbf{S}_j,key_i)}(\mathbf{S}_j))$. The key_{ck} is considered to be the correct (sub)key. The output of the function $f(\mathbf{S}_j, key_i)$ is the index of a key-dependent operation.

2.2 Reduced Template Attacks

In order to avoid numerical obstacles with the inversion of the covariance matrix $\mathbf{C}_i$, one can set the covariance matrix equal to the identity matrix. This essentially means that one does not take the cross correlation between different interesting points into consideration. A template that only consists of a mean vector is called a reduced template (pp. 108 in [9]). Correspondingly, Template Attacks based on reduced templates are called as reduced Template Attacks. In reduced Template Attacks, the probability density of the noises occurring under key-dependent operation O_i is given by the distribution $p'_i(\cdot)$, where the probability of observing a noise vector $\mathbf{n}_i(\mathbf{S})$ is:

$$p'_i(\mathbf{n}_i(\mathbf{S})) = \frac{1}{\sqrt{(2\pi)^N}}exp\Big(-\frac{1}{2}\mathbf{n}_i(\mathbf{S})\mathbf{n}_i(\mathbf{S})^T\Big) \quad \mathbf{n}_i(\mathbf{S}) \in \mathbb{R}^N.$$

2.3 Bayes Estimation

In the following, we briefly introduce the method of Bayes estimation [13]. We first introduce the definition of Bayes estimators. Then, we introduce how to compute a Bayes estimator.

Suppose an unknown parameter θ is known to have a prior distribution Λ (The prior distribution can be discrete or continuous distribution. In this paper, we only assume the prior distribution is continuous.). Quite generally, suppose that the consequences of estimating $g(\theta)$ by a value $\delta(X)$ (based on some measurements X) are measured by $L(\theta, \delta(X))$. As of the *loss function* L, we shall assume that

$$L(\theta, \delta(X)) \geq 0 \text{ for all } \theta \text{ and } \delta(X),$$

and $L[\theta, g(\theta)] = 0$ for all θ, so that the loss is zero when the correct value is estimated. The accuracy, or rather inaccuracy, of an estimator δ is then measured by the *risk function*

$$R(\theta, \delta) = E_\theta\{L[\theta, \delta(X)]\},$$

the long-term average loss resulting from the use of $\delta(X)$. This defines the risk function as a function of $\delta(X)$. An estimator $\delta(X)$ minimizing

$$r(\Lambda, \delta) = \int R(\theta, \delta) d\Lambda(\theta)$$

is called a *Bayes estimator* with respect to the prior distribution Λ. Note that, the distribution Λ is a probability distribution of the parameter θ, that is,

$$\int d\Lambda(\theta) = 1.$$

Now, we will introduce how to compute a Bayes estimator of an unknown parameter θ. Let $\lambda(\theta)$ denote the prior probability density of the parameter θ. The prior probability density of the population (or discrete probability function) is denoted by $f(X; \theta)$. If one extracts n samples $(X_1, X_2, \ldots, X_n)$ from the population, then the probability density of this group of samples is

$$f(X_1; \theta) f(X_2; \theta) \cdots f(X_n; \theta).$$

Thereby, we can compute the marginal density

$$p(X_1, X_2, \ldots, X_n) = \int \lambda(\theta) f(X_1; \theta) f(X_2; \theta) \cdots f(X_n; \theta) d\theta.$$

Then, the following posterior probability density is computed:

$$\lambda(\theta | X_1, \ldots, X_n) = \lambda(\theta) f(X_1; \theta) \cdots f(X_n; \theta) / p(X_1, X_2, \ldots, X_n). \tag{2}$$

In general, the Bayes estimator of the parameter θ is set to be the mean value of $\lambda(\theta | X_1, \ldots, X_n)$.

3 Using Priori Knowledge to Improve Template Attacks

In this section, we introduce how to use the priori knowledge about the reference device for Template Attacks. The usage of the priori knowledge for Template Attacks is the same for both classical Template Attacks and reduced Template Attacks.

It is well known that the instantaneous power consumption PC_{total} can be modeled as the sum of an operation-dependent component PC_{op}, a data-dependent component PC_{data}, the electronic noise $PC_{el.noise}$, and a constant component PC_{const} (pp. 62-65 in [9]), i.e.

$$PC_{total} = PC_{op} + PC_{data} + PC_{el.noise} + PC_{const}.$$

The value $PC_{op}+PC_{data}$ (or $PC_{op}+PC_{data}+PC_{const}$) can be viewed as the signal component and the value $PC_{el.noise}$ can be viewed as the noise component. Usually, for each point P_j in a power trace, its power consumption PC_{total} follows a normal distribution $\mathcal{N}(\mu_j, \sigma_j^2)$ and the electronic noise $PC_{el.noise}$ follows the normal distribution $\mathcal{N}(0, \sigma_j^2)$ (pp.62-65 in [9]). For fixed operation on fixed data, due to $Var(PC_{op}) = Var(PC_{data}) = Var(PC_{const}) = 0$, we have $PC_{op} + PC_{data} + PC_{el.noise} = \mu_j$. The priori knowledge is a kind of prior distribution of the actual value of the signal component μ_j. Due to the existence of the electronic noise, we can reasonably assume the prior distribution of the actual value of μ_j obtained by the attacker is a normal distribution.

There are many ways that the attacker can obtain the prior distribution and we will give out a specific one of them. Considering Example 1 in Section 1, for the same position about the target intermediate value, the attacker obtains n samples (For convenience, the samples are denoted by $X_1, \ldots, X_n$.) from power traces obtained from his previous experience of conducting Template Attacks against different devices which are similar to the reference device. Then, by computing

$$\theta_1 = \frac{1}{n} \cdot \sum_{i=1}^{n} X_i, \ \theta_2^2 = \frac{1}{n-1} \cdot \sum_{i=1}^{n} (X_i - \theta_1)^2,$$

the attacker can easily obtain the prior distribution which is the normal distribution $\mathcal{N}(\theta_1, \theta_2^2)$. Because the leakage distributions of the devices are very similar to that of the reference device, the prior distribution will be very accurate when the value of n is large and it can be used for the interesting points correspond to the same position about the target intermediate value for the reference device. We note that, compared with traditional Template Attacks, the computational price of obtaining the priori knowledge about the reference device is very small.

The more accurate the signal component (the value of μ_j) is estimated, the more accurate the noise component (the value $PC_{total} - \mu_j$) will be estimated. For an interesting point, if the signal component and the noise component are accurately estimated, accurate templates (reduced templates) will be built and Template Attacks (both classical Template Attacks and reduced Template Attacks) will be more powerful. In the classical way of building templates (reduced templates), for an interesting point, the attacker computes the mean value of the samples to estimate the actual value of the signal component μ_j. Specifically, for the key-dependent operation O_i, the point P_j is an interesting point and the attacker obtains n power traces $(\mathbf{S}_1, \mathbf{S}_2, \ldots, \mathbf{S}_n)$ from the reference device in the profiling stage. Therefore, the attacker obtains n values of the power consumption of the point P_j, one from each power trace. The n values are $S_1[P_j], S_2[P_j], \ldots, S_n[P_j]$. The actual value of μ_j is estimated by μ_j':

$$\mu_j' = M_i[P_j] = \sum_{k=1}^{n} S_k[P_j]/n.$$

However, in our scenario, the attacker not only has n power traces (The power traces are obtained from the reference device. However, the number of the power traces is limited.), but also possesses the priori knowledge about the reference device which can be used to estimate the actual value of μ_j more accurately. Let's consider the most common case. Assume that the attacker knows that the actual value of μ_j follows the normal distribution $\mathcal{N}(\theta_1, \theta_2^2)$ from priori knowledge[1] but does not know what the actual value of μ_j accurately is. The attacker can use the method of Bayes estimation to estimate the actual value of μ_j with the priori knowledge $\mathcal{N}(\theta_1, \theta_2^2)$ in the profiling stage as follows: The attacker computes the probability density of the actual value of the signal component μ_j from priori knowledge as

$$\lambda(\mu_j) = (\sqrt{2\pi}\theta_2)^{-1} exp\Big[-\frac{1}{2\theta_2^2}(\mu_j - \theta_1)^2\Big].$$

Moreover, the power consumption of the point P_j satisfies the following probability density function:

$$f(x; \mu_j, \sigma_j) = (\sqrt{2\pi}\sigma_j)^{-1} exp\Big[-\frac{1}{2\sigma_j^2}(x - \mu_j)^2\Big].$$

Although the value σ_j is not known to the attacker, it does not affect the process of computing the Bayes estimator of the actual value of μ_j. What the attacker needs is just an accurate estimation about the actual value of μ_j. From equation (2), the attacker computes the posterior probability density:

$$\lambda(\mu_j | S_1[P_j], \ldots, S_n[P_j]) = exp\Big[-\frac{1}{2\theta_2^2}(\mu_j - \theta_1)^2 - \frac{1}{2}\sum_{k=1}^{n}(S_k[P_j] - \mu_j)^2\Big]\Big/C_1,$$

the constant C_1 only has relation with $\theta_1, \theta_2, S_1[P_j], \ldots, S_n[P_j]$ and has no relation with μ_j. It has that

$$-\frac{1}{2\theta_2^2}(\mu_j - \theta_1)^2 - \frac{1}{2}\sum_{k=1}^{n}(S_k[P_j] - \mu_j)^2 = -\frac{1}{2A^2}(\mu_j - B)^2 + C_2,$$

where $A^2 = 1/(n + 1/\theta_2^2)$, $B = (nM_i[P_j] + \theta_1/\theta_2^2)/(n + 1/\theta_2^2)$, and C_2 has no relation with μ_j. Furthermore, the attacker can obtain

$$\lambda(\mu_j | S_1[P_j], \ldots, S_n[P_j]) = C_3 exp\Big[-\frac{1}{2A^2}(\mu_j - B)^2\Big],$$

where $C_3 = C_1 e^{C_2}$. Because it has that

$$\int_{-\infty}^{+\infty} \lambda(\mu_j | S_1[P_j], \ldots, S_n[P_j]) d\mu_j = 1,$$

hence $C_3 = (\sqrt{2\pi}A)^{-1}$. Up to now, the attacker obtains the Bayes estimator of the actual value of μ_j as

$$\mu_j'' = \frac{n}{n + 1/\theta_2^2}\left(\frac{\sum_{k=1}^{n} S_k[P_j]}{n}\right) + \frac{1/\theta_2^2}{n + 1/\theta_2^2}\theta_1. \tag{3}$$

[1] Note that, the normal distribution $\mathcal{N}(\theta_1, \theta_2^2)$ itself may not be very accurate. However, from the priori knowledge, the parameters θ_1, θ_2^2 are all known to the attacker.

The equation (3) shows that if the attacker does not have the priori knowledge (i.e. the prior distribution $\mathcal{N}(\theta_1, \theta_2^2)$), he can only use $\sum_{k=1}^{n} S_k[P_j]/n$ to estimate the actual value of μ_j. If the attacker does not have power traces obtained from the reference device, he can only use the priori knowledge (i.e. the value θ_1) to estimate the actual value of μ_j. If the attacker has power traces obtained from the reference device as well as the priori knowledge, by equation (3), he will use the weighted average of $\sum_{k=1}^{n} S_k[P_j]/n$ and θ_1 to estimate the actual value of μ_j under the ratio $n : 1/\theta_2^2$ in the profiling stage. This ratio is reasonable and the relevant reasons are as follows. On one hand, when more power traces are obtained from the reference device by the attacker, the proportion of $\sum_{k=1}^{n} S_k[P_j]/n$ should be larger. On the other hand, when the value θ_2^2 is smaller (This implies that the prior distribution of the actual value of μ_j is more accurate.), the proportion of θ_1 should be larger.

Other details of building templates (reduced templates) remain unchanged. Our way only exploits the priori knowledge to estimate the actual value of the signal component more accurately. We note that, due to the computational price of obtaining and exploiting the priori knowledge is very small, the priori knowledge can easily be used by practical attackers.

4 Experimental Evaluations

For the implementation of a cryptographic algorithm with countermeasures, one usually first tries his best to use some methods to delete the countermeasures from power traces. If the countermeasures can be deleted, then one tries to recover the correct (sub)key using some attack methods against unprotected implementation. For example, if one has power traces with random delays [11], he may first use the method proposed in [12] to remove the random delays from power traces and then uses some attack methods to recover the correct (sub)key. The methods of deleting countermeasures from power traces are beyond the scope of this paper. Moreover, considering power traces without any countermeasures shows the upper bound of the physical security of the target cryptographic device. Therefore, we take unprotected AES-128 implementation as an example.

We verified both classical Template Attacks and reduced Template Attacks by conducting simulated and practical experiments. In both simulated and practical experiments, we tried to attack the outputs of the S-boxes in the first round of AES-128. Before introducing the specific experiment details, we first introduce how to get the prior distribution of the actual value of the signal component for every interesting point for both simulated and practical experiments.

Paper [17] showed that reduced Template Attacks are more powerful compared with classical Template Attacks when the number of power traces used in the profiling stage is limited. Therefore, we mainly exploit reduced Template Attacks to exhibit our discoveries (Note that, our method can be used for both classical Template Attacks and reduced Template Attacks.).

For simplicity, for both simulated and practical experiments, let np denote the number of traces used in the profiling stage and let ne denote the number of traces used in the extraction stage. In this paper, we use the typical metric *Guessing Entropy* [5] as the metric about the classification performance of Template Attacks (Many other papers also used Guessing Entropy (e.g. [4,14,15]).).

4.1 How to Get The Priori Knowledge

In order to get the priori knowledge, we simulated the cases that the attacker can obtain the priori knowledge from his previous experience of conducting Template Attacks against a device identical or similar to the reference device.

For both simulated and practical experiments, we get the prior distribution of the actual value of the signal component for every interesting point using the traces which were generated in the same way as those were used in the two stages of Template Attacks. In this way, we can clearly give out an upper bound of how powerful Template Attacks will become by exploiting the priori knowledge.

In both simulated and practical experiments, for each key-dependent operation O_i and each interesting point P_j , we considered the prior distribution under four different levels of accuracy and assumed the prior distribution is a normal distribution $\mathcal{N}(\theta_1, \theta_2^2)$ (For different interesting points, the corresponding prior distributions are different.).

For each key-dependent operation O_i, we generated 400 traces (simulated traces or actual power traces). The 400 traces were used to estimate the prior distributions (i.e. the parameters θ_1 and θ_2^2 for the prior distribution) for every interesting point as follows. We repeated the following process 300 times. Every time, we chose 16 traces (Let $m = 16$ and the 16 traces are denoted by $S_1, \ldots, S_m$.) from the 400 traces uniformly at random and computed $\sum_{k=1}^{m} S_k[P_j]/m$. Therefore, there were 300 different values of $\sum_{k=1}^{m} S_k[P_j]/m$. The mean value of the 300 different values was set to be θ_1 and the variance of the 300 different values was set to be θ_2^2. In this way, the prior distribution $\mathcal{N}(\theta_1, \theta_2^2)$ was got. Note that, in practice, the attacker has many ways to get the prior distribution $\mathcal{N}(\theta_1, \theta_2^2)$. Our method which were used in this paper is just one of them.

Similarly, we additionally let $m = 32$, 64, 128 and obtained four different estimation of the prior distribution. Clearly, when the value m is larger, the estimation of θ_1 and θ_2^2 is more accurate. Therefore, we obtained the prior distribution under four different levels of accuracy, which represent the priori knowledge which the attacker can possess in practical attack scenarios.

We considered many kinds of Template Attacks and define the following symbols to denote them. In all the experiments, we let the symbol "CTA" denotes the classical Template Attacks without any priori knowledge. The symbol "CTA-16" denotes classical Template Attacks based on priori knowledge which is obtained when the value m equals to 16. Similarly, we define the symbols "CTA-32", "CTA-64", and "CTA-128" to denote the cases that the value m equals to 32, 64, and 128 respectively. We let the symbol "RTA" denotes the reduced Template Attacks without any priori knowledge. The symbol "RTA-16" denotes reduced Template Attacks based on priori knowledge which is obtained when the value m equals to 16. Similarly, we define the symbols "RTA-32", "RTA-64", and "RTA-128" to denote the cases that the value m equals to 32, 64, and 128.

4.2 Simulated Experiments

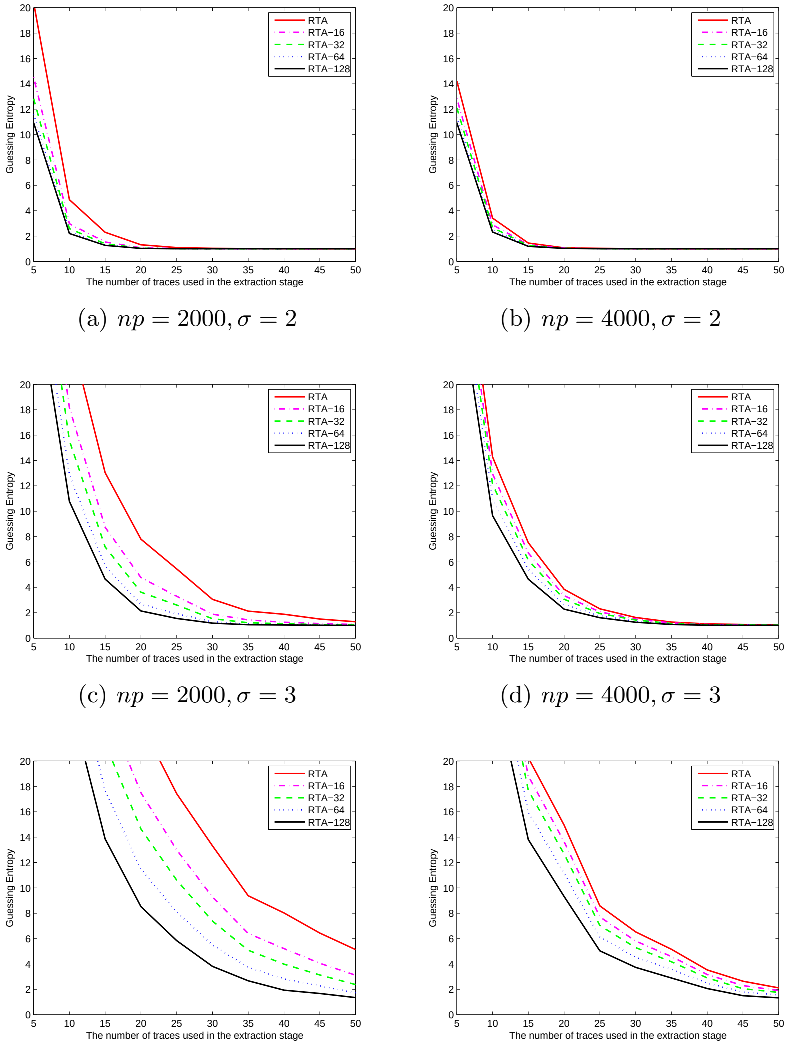
In simulated experiments, we chose 4 interesting points and the typical Hamming-Weight power model (pp. 40-41 in [9]) was adopted to describe the power consumption. The standard deviation of simulated Gaussian noise is denoted by σ. We employed three different noise levels to test the influence of noises on the classification performance of Template Attacks. The standard deviations of simulated Gaussian noise for the two noise levels were 2, 3, and 4.

For each noise level (The value of σ is fixed.), we respectively used 2,000 and 4,000 simulated traces to build the 256 reduced templates for the five kinds of reduced Template Attacks (RTA, RTA-16, RTA-32, RTA-64, and RTA-128). This means that the attacker respectively obtained 2,000 and 4,000 traces from the reference device in the profiling stage. The simulated traces were generated with a fixed subkey and random plaintext inputs. We generated additional 100,000 simulated traces with another fixed subkey and random plaintext inputs. The 100,000 simulated traces were used in the extraction stage. For fixed np and σ, we tested the Guessing Entropy of the five kinds of reduced Template Attacks when the attacker could use ne simulated traces in the extraction stage as follows. We respectively repeated the five kinds of Reduced Template Attacks 1,000 times. For each time, we chose ne simulated traces from the 100,000 simulated traces uniformly at random and the five kinds of Reduced Template Attacks were conducted with the same ne simulated traces. We respectively computed the Guessing Entropy of the five kinds of Reduced Template Attacks with the results of the 1,000 times attacks. The Guessing Entropy of the five kinds of Reduced Template Attacks for different values of np and σ is shown in Figure 1.

Table 1. The simulated experiment results for the case $np = 2,000, ne = 20, \sigma = 4$

RTA	RTA-16	RTA-32	RTA-64	RTA-128
22.94	17.50	14.65	11.49	8.53

The Guessing Entropy of the five kinds of reduced Template Attacks for the case $\{np = 2,000, ne = 20, \sigma = 4\}$ is shown in Table 1. From Figure 1 and Table 1, we find that the classification performance of reduced Template Attacks with accurate priori knowledge will be obvious better than that of reduced Template Attacks without priori knowledge. For example, in Table 1, the Guessing Entropy of RTA equals to 22.94, while the Guessing Entropy of RTA-128 equals to 8.53. Moreover, if the priori knowledge is more accurate, the classification performance of reduced Template Attacks with priori knowledge will be better. For example, in Table 1, the Guessing Entropy of RTA-16 equals to 17.50, while the Guessing Entropy of RTA-128 obviously reduces to 8.53.



(a) $np = 2000, \sigma = 2$

(b) $np = 4000, \sigma = 2$

(c) $np = 2000, \sigma = 3$

(d) $np = 4000, \sigma = 3$

(e) $np = 2000, \sigma = 4$

(f) $np = 4000, \sigma = 4$

Fig. 1. The simulated experiment results

Table 2. The simulated experiment results for different levels of noises

$np = 2,000, ne = 20$	$\sigma = 2$	$\sigma = 3$	$\sigma = 4$
RTA	1.31	7.80	22.94
RTA-128	1.03	2.14	8.53

Table 2 shows the Guessing Entropy of RTA and RTA-128 for different levels of noises when np is fixed to $2,000$ and ne is fixed to 20. From Figure 1 and Table 2, we find that, when the noise level is higher, reduced Template Attacks with priori knowledge will achieve larger advantage over Reduced Template Attacks without priori knowledge. For example, in Table 2, the Guessing Entropy of RTA and RTA-128 is almost equal when σ equals to 2 (1.31 and 1.03). However, when σ equals to 4, the Guessing Entropy of RTA-128 (8.53) is much lower than that of RTA (22.94).

When more simulated traces can be obtained from the reference device (e.g. $np = 4,000$) in the profiling stage, the advantages of reduced Template Attacks with priori knowledge over Template Attacks without priori knowledge will be smaller. For classical Template Attacks, we computed the Guessing Entropy of the five kinds of classical Template Attacks (CTA, CTA-16, CTA-32, CTA-64, and CTA-128) similarly. The simulated experiment results show that classical Template Attacks with accurate priori knowledge have advantages over classical Template Attacks without priori knowledge.

4.3 Practical Experiments

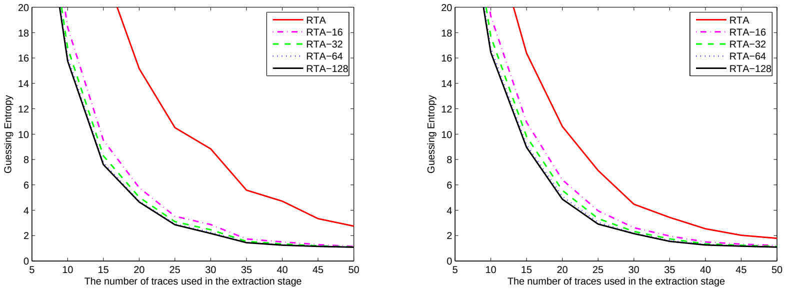
We tried to attack the outputs of all the S-boxes in the first round of an unprotected AES-128 software implementation on an typical 8-bit microcontroller STC89C58RD+ whose operating frequency is 11MHz. The actual power traces were acquired with a sampling rate of 50MS/s. The average number of actual power traces during the sampling process was 10 times. For our device, the condition of equal covariances [4] does not hold.

We generated two sets of power traces, Set A and Set B. The Set A captured 10,000 actual power traces which were generated with a fixed main key and random plaintext inputs. The Set B captured 100,000 actual power traces which were generated with another fixed main key and random plaintext inputs. The power traces in Set A were used in the profiling stage and the power traces in Set B were used in the extraction stage. We used the same device as that was used to get the prior distribution in Section 4.1 to generate the two sets of power traces, which provides a good setting for the focuses of our research. For each S-box of the unprotected AES-128 software implementation, we used CPA based method (Chapter 6 in [9]) to choose 4 interesting points in 4 continual clock cycles, one in each clock cycle[1]. We note that using CPA to choose interesting points is a popular method for Template Attacks. Both classical Template Attacks

[1] In our device, the target intermediate values only continue 4 clock cycles.

and reduced Template Attacks were conducted based on the same 4 interesting points. We only show the practical experiment results of the first and the second S-box in this paper. For other S-boxes in the first round of the unprotected AES-128 software implementation, similar evaluation results were obtained by us.

For reduced Template Attacks, we respectively chose 2,000 and 4,000 power traces from Set A to build the 256 templates for the five kinds of reduced Template Attacks (RTA, RTA-16, RTA-32, RTA-64, and RTA-128). The 100,000 power traces of Set B were used in the extraction stage for the five kinds of reduced Template Attacks. For fixed *np*, we tested the Guessing Entropy of the five kinds of reduced Template Attacks when one uses *ne* actual power traces in the extraction stage similarly to that of the simulated experiments but used actual power traces. The Guessing Entropy of the five kinds of reduced Template Attacks for the first S-box are shown in Figure 2. The Guessing Entropy of the five kinds of reduced Template Attacks for the first S-box when *np* is fixed to $2,000$ and *ne* is fixed to 20 is shown in Table 3.



(a) $np = 2,000$ (b) $np = 4,000$

Fig. 2. The experiment results of Reduced Template Attacks for the first S-box

Table 3. The experiment results of Reduced Template Attacks for the first S-box

$np = 2,000$	RTA	RTA-16	RTA-32	RTA-64	RTA-128
$ne = 20$	15.16	5.78	5.03	4.73	4.65

From Figure 3 and Table 3, we find that the classification performance of reduced Template Attacks with accurate priori knowledge will be obvious better than that of reduced Template Attacks without priori knowledge. For example, in Table 3, the Guessing Entropy of RTA equals to 15.16, while the Guessing Entropy of RTA-16 reduces to 5.78.

For classical Template Attacks, we respectively chose 5,000 and 10,000 power traces from Set A to build the 256 templates for the five kinds of classical Template Attacks (CTA, CTA-16, CTA-32, CTA-64, and CTA-128). Moreover, using power traces from Set B, we computed the Guessing Entropy of the five kinds of classical Template Attacks when one uses *ne* power traces in the extraction stage similarly. The Guessing Entropy of the five kinds of classical Template Attacks for the first S-box are shown in Figure 3 in Appendix A.

For the second S-box, we also used the power traces in Set A and Set B to compute the Guessing Entropy of the five kinds of classical Template Attacks and the five kinds of reduced Template Attacks similarly. The practical experiment results for the second S-box which can also verify our discoveries are shown in Figure 4, Figure 5, and Table 4 in Appendix B.

The practical experiment results show that, for both reduced Template Attacks and classical Template Attacks, if the priori knowledge is more accurate, the classification performance will be better. For example, in Table 3, the Guessing Entropy of RTA-16 equals to 5.78, while the Guessing Entropy of RTA-128 reduces to 4.65. When more power traces can be obtained from the reference device, the advantages of Template Attacks with priori knowledge over Template Attacks without priori knowledge will be smaller.

5 Conclusion and Future Work

In this paper, we verify that if the attacker can obtain the priori knowledge (even if may not be very accurate) about the reference device (as well as the target device), Template Attacks (both classical Template Attacks and reduced Template Attacks) will be more powerful than we previously think. Evaluation results exhibit that leaking this kind of priori knowledge poses serious threat to the physical security of cryptographic devices. Therefore, we suggest that the designers of a cryptographic device should take the priori knowledge into consideration when he uses Template Attacks to evaluate the physical security of the cryptographic device.

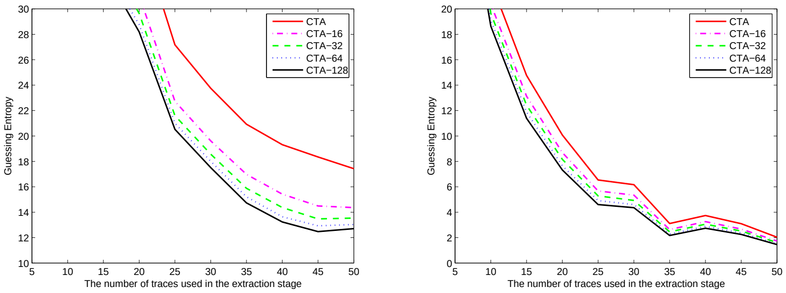
The future work is as follows. First, our discoveries show that the approach to infer (estimate) the priori knowledge as accurately as possible is crucial and is worth being researched from the attacker's view. Second, it would be interesting to research how to prevent the attacker to obtain the priori knowledge (Using countermeasures such as the method of random delays [11] may be a good choice.). We should also concern on how to exploit the priori knowledge to make other profiled side-channel attacks (such as Stochastic Model based Attacks [16], PCA-based Template Attacks etc.) become more powerful in a reasonable way. It is also necessary to further verify our discoveries in other devices such as ASIC and FPGA.

References

1. Chari, S., Rao, J.R., Rohatgi, P.: Template Attacks. CHES2002, LNCS 2523, pp. 13-28, 2003.

2. Rechberger, C., Oswald, E.: Practical Template Attacks. WISA2004, LNCS 3325, pp. 440-456, 2004.
3. Archambeau, C., Peeters, E., Standaert, F.-X., Quisquater, J.-J.: Template Attacks in Principal Subspaces. CHES2006, LNCS 4249, pp. 1-14, 2006.
4. Choudary, O., Kuhn, M.G.: Efficient Template Attacks. CARDIS2013, LNCS 8419, pp. 253-270, 2013.
5. Standaert, F.-X., Malkin, T.G., Yung, M.: A Unified Framework for the Analysis of Side-Channel Key Recovery Attacks. EUROCRYPT2009, LNCS 5479, pp. 443-461, 2009.
6. Montminy, D.P., Baldwin, R.O., Temple, M.A., Laspe, E.D.: Improving cross-device attacks using zero-mean unit-variance mormalization. Journal of Cryptographic Engineering, Volume 3, Issue 2, pp. 99-110, June 2013.
7. Oswald, E., Mangard, S.: Template Attacks on Masking—Resistance Is Futile. CT-RSA2007, LNCS 4377, pp. 243-256, 2007.
8. Standaert, F.-X., Archambeau, C.: Using Subspace-Based Template Attacks to Compare and Combine Power and Electromagnetic Information Leakages. CHES2008, LNCS 5154, pp. 411-425, 2008.
9. Mangard, S., Oswald, E., Popp, T.: Power Analysis Attacks: Revealing the Secrets of Smart Cards. Springer 2007.
10. Hanley, N., Tunstall, M., Marnane, W.P.: Unknown Plaintext Template Attacks. WISA2009, LNCS 5932, pp. 148-162, 2009.
11. Coron, J.-S., Kizhvatov, I.: Analysis and Improvement of the Random Delay Countermeasure of CHES 2009. CHES2010, LNCS 6225, pp. 95-109, 2010.
12. Durvaux, F., Renauld, M., Standaert, F.-X. et al.: Efficient Removal of Random Delays from Embedded Software Implementations Using Hidden Markov Models. CARDIS2012, LNCS 7771, pp. 123-140, 2013.
13. Lehmann, E. L., Casella, G.: Theory of Point Estimation (2nd ed.). Springer. ISBN 0-387-98502-6.
14. Standaert, F.-X., Gierlichs, B., Verbauwhede, I.: Partition vs. Comparison Side-Channel Distinguishers: An Empirical Evaluation of Statistical Tests for Univariate Side-Channel Attacks against Two Unprotected CMOS Devices. ICISC2008, LNC-S5461, pp. 253-267, 2009.
15. Medwed, M., Standaert, F.-X., Joux. A.: Towards Super-Exponential Side-Channel Security with Efficient Leakage-Resilient PRFs. CHES2012, LNCS 7428, pp. 193-212, 2012.
16. Schindler, W., Lemke, K., Paar, C.: A Stochastic Model for Differential Side Channel Cryptanalysis. CHES2005, LNCS 3659, pp. 30-46, 2005.
17. Ye, X., Eisenbarth, T.: Wide Collisions in Practice. ACNS2012, LNCS 7341, pp. 329-343, 2012.
18. The DPA Contest: http://www.dpacontest.org/home/
19. Power Analysis Attacks-Revealing the Secrets of Smartcards: http://dpabook.org/
20. Standaert, F.-X., Koeune, F., Schindler, W.: How to Compare Profiled Side-Channel Attacks? ACNS2009, LNCS 5536, pp. 485-498, 2009.
21. Choudary, O., Kuhn, M.G.: Template Attacks on Different Devices. COSADE2014, LNCS 8622, pp. 179C198, 2014.

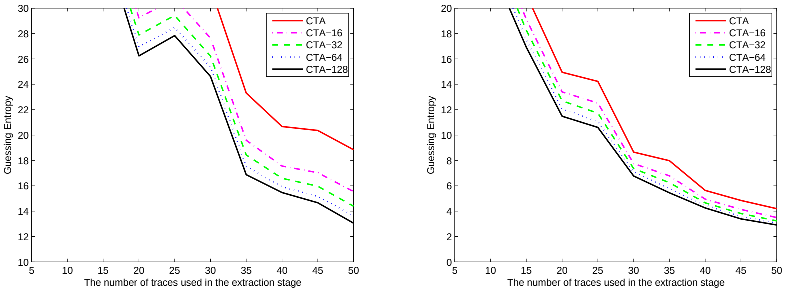
Appendix A: Practical Experiments for The 1^{st} S-box



(a) $np = 5,000$ (b) $np = 10,000$

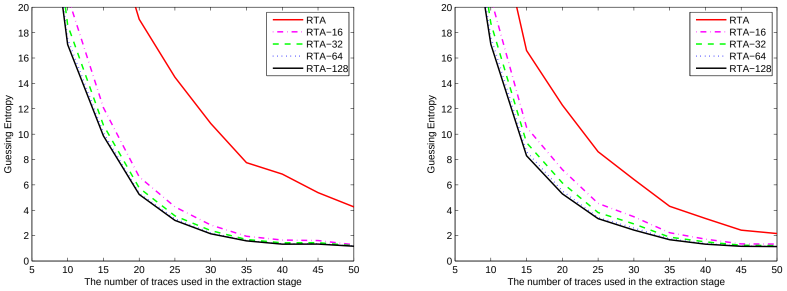
Fig. 3. The experiment results of Classical Template Attacks for the first S-box

Appendix B: Practical Experiments for The 2^{nd} S-box



(a) $np = 5,000$ (b) $np = 10,000$

Fig. 4. The experiment results of Classical Template Attacks for the second S-box



(a) $np = 2,000$

(b) $np = 4,000$

Fig. 5. The experiment results of Reduced Template Attacks for the second S-box

Table 4. The experiment results of Reduced Template Attacks for the second S-box

$np = 2,000$	RTA	RTA-16	RTA-32	RTA-64	RTA-128
$ne = 20$	19.05	6.64	5.76	5.34	5.25