How to Choose Interesting Points for Template Attacks?

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 to be the most powerful side-channel attacks from an information theoretic point of view. For classical Template Attacks, several papers suggested an accepted guideline of choosing interesting points. The accepted guideline is that one should only choose one point as the interesting point per clock cycle. Up to now, many methods of choosing interesting points were introduced. However, it is still *unclear* that which method will lead to the best classification performance for Template Attacks. In this paper, we comprehensively compare the classification performance of Template Attacks when using different known methods of choosing interesting points. Evaluation results show that CPA based method and SOST will lead to the best classification performance. Moreover, we find that some of the methods of choosing interesting points are essentially equivalent. In additional, we give out a more reasonable proof about the accepted guideline of choosing interesting points for Template Attacks by presenting a new way of conducting Template Attacks.

Keywords: Side-Channel Attacks, Power Analysis Attacks, Template Attacks, Interesting Points.

1 Introduction

As an important method of Power Analysis Attacks, Template Attacks were firstly proposed by S. Chari et al. in 2002 [1]. Under the assumption that one has a reference device identical or similar to the targeted device, and thus be well capable of characterizing power leakages of the targeted device, Template Attacks are widely accepted to be the strongest side-channel attacks from an information theoretic point of view [1]. We note that, Template Attacks are also important tools to evaluate the physical security of a cryptographic device.

Contributions Depending on the measurement setup and the data acquisition strategy, captured traces can be quite big (i.e. the number of sampled points is high). For Template Attacks to be practical, it is paramount that not all points of a power trace are part of the templates. To reduce the number of points, one needs to choose some special points as the interesting points in traces. The interesting points are those time samples that contain the most information about the characterized key-dependent operations. For classical Template Attacks, many papers [2, 3, 5, 10, 12] suggested an accepted guideline of choosing interesting points. The accepted guideline is that one should *only* choose *one* point as the interesting point per clock cycle. When more points are used as the interesting points per clock cycle, numerical problems will arise and the classification performance of classical Template Attacks will be poorer [2, 4].

Up to now, many methods of choosing interesting points were introduced. They are *Difference Of Means based method* [1] (DOM), *Sum Of Squared Differences based method* [10] (SOSD), *Correlation Power Analysis based method* [11] (CPA), *Sum Of Squared pairwise T-differences based method* [10] (SOST), *Signal to Noise Ratio based method* [11] (SNR), *Variance based method* [16] (VAR), *Mutual Information Analysis based method* [17] (MIA). *Kolmogorov-Smirnov Analysis based method* [18] (KSA), and *Principal Component Analysis based method* [3][1] (PCA). However, an important question for Template Attacks is still not solved. It is still *unclear* that which method will lead to the best classification performance for Template Attacks.

Another question is that the accepted guideline of choosing interesting points for classical Template Attacks is still not proved in a reasonable way. Previous papers [2, 4] only said that, for classical Template Attacks, numerical problems will arise when inverting the covariance matrix if one uses more than one point as the interesting point per clock cycle and more points lead to poorer classification performance. On one hand, no one gives out what the numerical problems accurately are. On the other hand, when the numerical problems do not exist, whether using more points per clock cycle will lead to better classification performance is still *unknown*. Therefore, we think that the accepted guideline of choosing interesting points is still not proved in a reasonable way.

In this paper, we try to answer the above two questions. Main contributions of this paper are two-fold. Firstly, we comprehensively compare the classification performance of Template Attacks when using different known methods of choosing interesting points. Our results show that the Correlation Power Analysis based method and the Sum Of Squared pairwise T-differences based method will lead to the best classification performance and some methods of choosing interesting points are essentially equivalent. Secondly, we more reasonably prove that the accepted guideline of choosing interesting points for Template Attacks is correct by presenting a new way of conducting Template Attacks.

Related Work Template Attacks were firstly introduced in [1]. The paper [2] provided answers to some basic and practical issues of Template Attacks, such as how to choose interesting points in an efficient way and how to preprocess noisy data. The paper [4] proposed efficient methods to avoid several possible numerical obstacles when implementing Template Attacks. The paper [12] presented a variant of Template Attacks that can be applied to block ciphers when

[1] The space is limited to show all the methods completely in this paper.

the plaintext and ciphertext used are unknown. In [8], Template Attacks were used to attack a masking protected implementation of a block cipher. Recently, a simple pre-processing technique of Template Attacks, normalizing the sample values using the means and variances was evaluated for various sizes of test data [7]. LDA-based Template Attacks were introduced in [9]. However, this kind of Template Attacks depends on the condition of equal covariances [4] (Please see Section 2.1.1 for more details.), which does not hold in most settings. Therefore, it is not a better choice compared with PCA-based Template Attacks in most settings [4] and we ignore this kind of attacks here.

Organization of This Paper The rest of this paper is organized as follows. In Section 2, we review classical Template Attacks and PCA-based Template Attacks. In Section 3, we introduce the classical way and our new way of conducting Template Attacks. The two ways are used to show our contributions. In Section 4, we comprehensively compare the classification performance of Template Attacks when using different known methods of choosing interesting points. We also prove the accepted guideline of choosing interesting points for Template Attacks by our new way in this section. In Section 5, we conclude the whole paper.

2 Preliminaries

In this section, we briefly review classical Template Attacks as well as PCA-based Template Attacks.

2.1 Classical Template Attacks

Template Attacks consist of two stages. The first stage is the profiling stage and the second stage is the extraction stage. We will introduce the two stages of classical Template Attacks. In the profiling stage, one has a reference device identical or similar to the targeted device and builds templates for each key-dependent operation with the reference device. In the extraction stage, one can exploit a small number of power traces measured from the targeted device and the templates to classify the correct (sub)key.

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 $(P_0, P_1, \ldots, P_{N-1})$. Each template is composed of a mean vector and a covariance matrix. The mean vector is used to estimate the data-dependent portion 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 noises 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 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:

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

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. We know that the matrix $\mathbf{C}_i$ is the estimate of the true covariance $\mathbf{\Sigma}_i$. The condition of equal covariances [4] means that the leakages from different key-dependent operations have the same true covariance $\mathbf{\Sigma} = \mathbf{\Sigma}_0 = \mathbf{\Sigma}_1 = \cdots = \mathbf{\Sigma}_{K-1}$. In most settings, the condition of equal covariances does not hold. Therefore, in this paper, we only consider the devices in which the condition of equal covariances does not hold.

2.1.2 The Extraction Stage Assume that one obtains t traces (denoted by $\mathbf{S}_1, \mathbf{S}_2, \ldots, \mathbf{S}_t$) from the targeted device in the extraction stage. For example, when the traces are statistically independent, one will apply maximum likelihood approach on the product of conditional probabilities [11], 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. For example, when one attacks the output of a S-box (denoted by *Sbox*) in the first round of AES-128, one builds templates for each output of the S-box. In this case, $f(\mathbf{S}_j, key_i) = Sbox(m_j \oplus key_i)$, where m_j is the plaintext corresponding to the power trace $\mathbf{S}_j$.

2.2 PCA-based Template Attacks

PCA-based Template Attacks [3] which were considered to provide the best results before [4] exploit a continual point fragment correspond to the targeted intermediate value in traces (We assume the length of the continual point fragment is N.). One first computes the empirical covariance matrix, which is given by

$$\mathbf{ECM} = \frac{1}{K}\sum_{i=0}^{K-1}(\mathbf{M}_i - \overline{\mathbf{M}})(\mathbf{M}_i - \overline{\mathbf{M}})^T.$$

The quantity $\overline{\mathbf{M}} = \sum_{i=0}^{K-1}\mathbf{M}_i/K$ is the average of the mean vectors. Let us denote the matrixes of eigenvectors and eigenvalues of $\mathbf{ECM}$ by $\mathbf{U}$ and $\mathit{\Delta}$, i.e.

$\mathbf{ECM} = \mathbf{U}\varDelta\mathbf{U}^T$. The principal directions $\{w_i\}_{i=1}^{L}$ are the columns of $\mathbf{U}$ that correspond to the L largest eigenvalues of $\varDelta$. The corresponding matrix of principal directions is denoted $\mathbf{W} \in \mathbb{R}^{N\times L}$. The *Cumulative Percentage of Total Variation* [13] is often used to determine how many principal directions should be exploited (i.e. to determine the concrete value of L). One uses projected mean vectors $\{\mathbf{W}^T\mathbf{M}_i^T\}_{i=0}^{K-1}$ and projected covariance matrices $\{\mathbf{W}^T\mathbf{C}_i\mathbf{W}\}_{i=0}^{K-1}$ to conduct the attacks. Specifically speaking, the probability of observing a noise vector when one assumes the key-dependent operation is O_i is computed by

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

One classifies the correct (sub)key based on the probability computed by equation (2). One can use the method introduced in paper [4] to compute the projected mean vectors and the projected covariance matrices. Using this method, one can avoid both numerical problems and the computation of large covariance matrices. We uses this advanced method to conduct PCA-based Template Attacks in this paper.

3 Strategies to Conduct Template Attacks

In this section, for the purpose of comparison, we will introduce three different strategies to conduct Template Attacks. The first strategy and the second strategy are the classical way of conducting Template Attacks but with different number of interesting points per clock cycle. The third strategy is our new way of conducting Template Attacks.

Assume that, in one trace, there is a *continual* point fragment $(P_0, P_1, \ldots, P_{N-1})$ which is correspond to the key-dependent operations and has length N. We also assume that these N points are in c continual clock cycles. Therefore, there are N/c points per clock cycle. Let the symbol $P_{(i,j)}$ denotes the j^{th}, $j \in \{1, \ldots, N/c\}$ interesting point in the i^{th}, $i \in \{1, \ldots, c\}$ clock cycle. For interesting points in the same clock cycle, their orders are determined by the method of choosing interesting points. For example, when one uses Correlation Power Analysis based method, he computes the coefficient of correlation of each point in the clock cycle. The point with the highest coefficient of correlation is set to be $P_{(i,1)}$ and the point with the lowest coefficient of correlation is set to be $P_{(i,N/c)}$.

3.1 Strategy 1

In this strategy, one only uses one point as the interesting point per clock cycle and chooses c points $\{P_{(1,1)}, P_{(2,1)}, \ldots, P_{(c,1)}\}$ from the N continual points as the c interesting points. Then, one conducts classical Template Attacks with templates which are built with the c interesting points. We call the attack with this strategy as “ATTACK-1”.

3.2 Strategy 2

In this strategy, one uses more than one point as the interesting points per clock cycle. In order to show this strategy more clearly, we take the simplest case as an example (i.e. We assume that one uses two points as the interesting points per clock cycle). Therefore, $2c$ points are chosen from the N continual points as the interesting points: $\{(P_{(1,1)}, P_{(1,2)}), (P_{(2,1)}, P_{(2,2)}), \ldots, (P_{(c,1)}, P_{(c,2)})\}$.

Then, one conducts classical Template Attacks with templates which are built with the $2c$ interesting points. This means that one needs to compute a $(1 \times 2c)$ mean vector and a $(2c \times 2c)$ covariance matrix for each template. We call the attack with Strategy 2 as "ATTACK-2". Note that, the success rate of Strategy 2 will reduce when one uses more points as the interesting points per clock cycle to conduct Template Attacks [2,4]. Our experiments in the next section also verify this fact.

3.3 Strategy 3 (Our New Way)

Strategy 3 is our new way of conducting Template Attacks. In our new way, during the profiling stage, one uses more than one point as the interesting points per clock cycle. We also take the simplest case as an example. Therefore, $2c$ points are chosen from the N continual points as the interesting points:

$$\big\{(P_{(1,1)}, P_{(1,2)}), (P_{(2,1)}, P_{(2,2)}), \ldots, (P_{(c,1)}, P_{(c,2)})\big\}.$$

One divides the $2c$ interesting points into two sets. In the first set, there are c interesting points: $\mathsf{Set1} = \{P_{(1,1)}, P_{(2,1)}, \ldots, P_{(c,1)}\}$. The rest c interesting points are in the second set: $\mathsf{Set2} = \{P_{(1,2)}, P_{(2,2)}, \ldots, P_{(c,2)}\}$. Note that, in each set, any 2 points of the c interesting points are not in the same clock cycle. But the 2 points $(P_{(i,1)}, P_{(i,2)}),\ i \in \{1, 2, \ldots, c\}$ are in the same clock cycle. In the following, one builds templates in the same way as classical Template Attacks with the c interesting points in $\mathsf{Set1}$ and obtains a group of templates denoted by $\mathsf{G1}$. Similarly, with the same traces used for obtaining $\mathsf{G1}$, one builds templates with the c interesting points in $\mathsf{Set2}$ and obtains another group of templates $\mathsf{G2}$. At this point, the profiling stage is finished.

In the extraction stage, one first computes a sequence

$$\big\{\mathsf{Pr}(1,0), \mathsf{Pr}(1,1), \ldots, \mathsf{Pr}(1,K-1)\big\},$$

where the value $\mathsf{Pr}(1,i)$ represents the probability of the i^{th} (sub)key is the correct (sub)key (In the example of Section 2.2, $\mathsf{Pr}(1,i)$ equals to $\prod_{j=1}^{t} \mathsf{Pr}(S_j|key_i)$.) using $\mathsf{G1}$ with some traces obtained from the targeted device in the same way as classical Template Attacks. Then, one sorts the sequence in decreasing order and computes $\mathsf{Index}(1,i),\ i = 0, 1, \ldots, K-1$ for each (sub)key. The value $\mathsf{Index}(1,i)$ represents the sequence number of $\mathsf{Pr}(1,i)$ in the sorted sequence. Similarly, one computes another sequence $\{\mathsf{Pr}(2,0), \mathsf{Pr}(2,1), \ldots, \mathsf{Pr}(2,K-1)\}$ using $\mathsf{G2}$ with the same traces obtained from the targeted device. Then, he computes

$\mathsf{Index}(2,i)$, $i = 0,1,\ldots,K-1$ for each (sub)key. The candidate value of the correct key is computed by

$$key_{ck} := argmin_i\big\{\mathsf{Index}(1,i) + \mathsf{Index}(2,i), i \in \{0,1,\ldots,K-1\}\big\}.$$

We call the attack with our new way as "ATTACK-3".

Discussions There is a kind of generalization about our new way of conducting Template Attacks. The generalization is to use more points as the interesting points per clock cycle to conduct our new way of Template Attacks. Assume that one uses s ($2 < s \leq N/c$) points as the interesting points per clock cycle, he will divide the cs interesting points into s sets and build s groups of templates in the profiling stage similarly to the way introduced above. In the extraction stage, one classifies the correct (sub)key by computing

$$key_{ck} := argmin_i\big\{\mathsf{Index}(1,i) + \ldots + \mathsf{Index}(s,i), i \in \{0,1,\ldots,K-1\}\big\}.$$

Note that, in our new way, we do not build templates with interesting points in the same clock cycle simultaneously. Therefore, the new way avoids the numerical problems when inverting the covariance matrix if one uses more than one point per clock cycle. If the accepted guideline of choosing interesting points for Template Attacks is incorrect, the new way will has better classification performance when more points are used. If the accepted guideline of choosing interesting points for Template Attacks is correct, the classification performance of the new way will remain almost unchanged when more points are used. Therefore, we can more reasonably prove the accepted guideline of choosing interesting points for Template Attacks by using this new way.

4 Experimental Evaluations

In this section, we will introduce two groups of experiments. In the first group of experiments (denoted by Group 1), we comprehensively compare the classification performance of Template Attacks when using different known methods of choosing interesting points. In the second group of experiments (denoted by Group 2), we more reasonably prove that the accepted guideline of choosing interesting points for Template Attacks is correct by using our new way.

For implementation of a cryptographic algorithm with countermeasures, one usually first uses some methods to delete the countermeasures and then tries to attack the implementation using classical attacks against unprotected implementation. For example, if one has traces with random delays [15]. He may first use the method proposed in [14] to remove the random delays and then uses classical attacks to recover the correct (sub)key. The methods of deleting countermeasures are beyond the scope of this paper. Therefore, we take unprotected AES-128 implementation as example. We tried to attack the output of the first S-box in the first round of an unprotected AES-128 software implementation on an typical 8-bit microcontroller STC89C58RD+ (This 8-bit microcontroller was also exploited by other papers [19, 20].) whose operating frequency is 11MHz.

The real power traces were acquired with a sampling rate of 50MS/s. The average number of real power traces during the sampling process was 10 times.

We used the same device for both the profiling stage and the extraction stage, which provides a good setting for the focuses of our research. For our device, the condition of equal covariances does not hold. This means that the differences between different covariance matrixes $\mathbf{C}_i$ are very evident (can easily be observed from visual inspection). In our traces, we chose three continual clock cycles about the output of the first S-box in the first round of the unprotected AES-128 software implementation. In each clock cycle, there are 4 points. Therefore, there are 12 points (denoted by $\{P_0, P_1, \ldots, P_{11}\}$) totally. We implemented all the methods of choosing interesting points including DOM, SOSD, CPA, SOST, SNR, VAR, MIA, KSA, and PCA. Let "PCA-TA" denote PCA-based Template Attacks using the first 6 principal directions (i.e. $L = 6$ Please see Section 2.2 for more details.). For our device, the first 6 principal directions are sufficient to ensure the classification performance of PCA-based Template Attacks. The Cumulative Percentage of Total Variation is larger than 0.997 when the first 6 principal directions are used. The next few principal directions only *slightly* increase the power of this kind of attacks and it is not necessary to use all the principal directions [3, 4].

For simplicity, 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 success rate [6] as a metric about classification performance of Template Attacks. For other S-boxes in the first round of the unprotected AES-128 software implementation, similar evaluation results were obtained by us for both the two groups of experiments.

4.1 Group 1

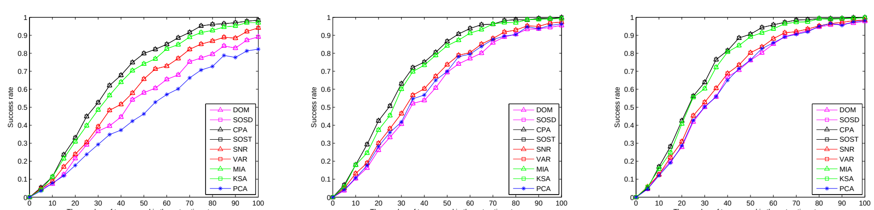
We generated 40,000 traces which were used for choosing interesting points with a fixed main key and random plaintext inputs. In Table 1, we show the interesting points chosen by different methods (DOM, SOSD, CPA, SOST, SNR, VAR, MIA, and KSA) using the 40,000 traces. In Table 1, the symbol "(i, j)" denotes the j^{th} interesting point in the i^{th} clock cycle (i.e. $P_{(i,j)}$). From Table 1, we find that some of the methods of choosing interesting points provide same results. For example, Difference Of Means based method and Sum Of Squared Differences based method provide same results. Correlation Power Analysis based method and Sum Of Squared pairwise T-differences based method provide same results. Signal to Noise Ratio based method and Variance based method provide same results. Therefore, we believe that the methods providing same results are essentially equivalent.

We will show the success rates of Template Attacks using different methods of choosing interesting points. According to the accepted guideline of choosing interesting points, for the above eight methods, we built templates with points $\{P_{(1,1)}, P_{(2,1)}, P_{(3,1)}\}$, one in each clock cycle. We conducted Template Attacks using different methods of choosing interesting points with same traces both in the profiling stage and the extraction stage. Specifically speaking, we used

Table 1. The interesting points chosen by different methods

	(1,1)	(1,2)	(1,3)	(1,4)	(2,1)	(2,2)	(2,3)	(2,4)	(3,1)	(3,2)	(3,3)	(3,4)
DOM	P_2	P_3	P_0	P_1	P_7	P_6	P_4	P_5	P_{11}	P_9	P_{10}	P_8
SOSD	P_2	P_3	P_0	P_1	P_7	P_6	P_4	P_5	P_{11}	P_9	P_{10}	P_8
CPA	P_3	P_2	P_1	P_0	P_6	P_5	P_4	P_7	P_9	P_{11}	P_{10}	P_8
SOST	P_3	P_2	P_1	P_0	P_6	P_5	P_4	P_7	P_9	P_{11}	P_{10}	P_8
SNR	P_0	P_2	P_3	P_1	P_7	P_4	P_6	P_5	P_{11}	P_9	P_{10}	P_8
VAR	P_0	P_2	P_3	P_1	P_7	P_4	P_6	P_5	P_{11}	P_9	P_{10}	P_8
MIA	P_2	P_3	P_1	P_0	P_6	P_5	P_7	P_4	P_9	P_{11}	P_{10}	P_8
KSA	P_2	P_3	P_1	P_0	P_6	P_5	P_7	P_4	P_9	P_{11}	P_8	P_{10}

10,000, 15,000, and 20,000 traces to build the 256 templates respectively. The traces were generated with a fixed main key and random plaintext inputs. We generated additional 20,000 traces which were used in the extraction stage with another fixed main key and random plaintext inputs. We tested the success rates of Template Attacks using different methods of choosing interesting points when one uses ne traces in the extraction stage as follows. We repeated the attacks 1,000 times. For each time, we chose ne traces from the 20,000 traces uniformly at random and the attacks were conducted with the same ne traces. We respectively recorded how many times the attacks can successfully recover the correct subkey. The success rates of Template Attacks using different methods of choosing interesting points are shown in Figure 1.



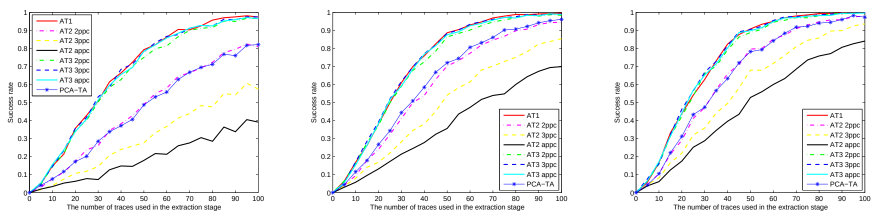
(a) np =10,000 (b) np =15,000 (c) np =20,000

Fig. 1. Experiment results of different methods of choosing interesting points

From Figure 1, we find that Correlation Power Analysis based method and Sum Of Squared pairwise T-differences based method lead to the highest success rates in all cases. When np is small (e.g. $np = 10,000$), PCA-based Template Attacks lead to the lowest success rates. When np is large, Difference Of Means based method and Sum Of Squared Differences based method lead to the lowest success rates.

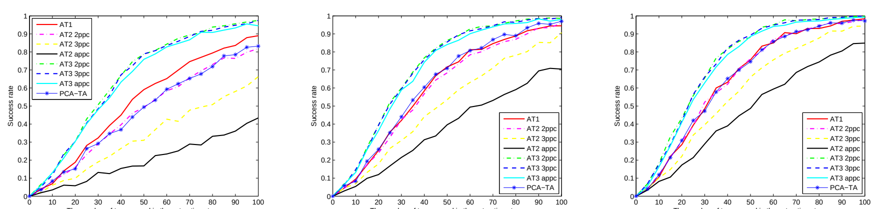
4.2 Group 2

We prove the accepted guideline of choosing interesting points for Template Attacks with both the best and the worst methods of choosing interesting points. Therefore, we chose Correlation Power Analysis based method as the best method and Difference Of Means based method as the worst method. We also conducted PCA-based Template Attacks (PCA-TA for short) for the purpose of comparison.



(a) np =10,000 (b) np =15,000 (c) np =20,000

Fig. 2. Experiment results of Correlation Power Analysis based method



(a) np =10,000 (b) np =15,000 (c) np =20,000

Fig. 3. Experiment results of Difference Of Means based method

Let “AT1” denote ATTACK-1. Let “AT2 2ppc” and “AT3 2ppc” respectively denote the case of ATTACK-2 and ATTACK-3 using the same two points as the interesting points per clock cycle. Let “AT2 3ppc” and “AT3 3ppc” respectively denote the case of ATTACK-2 and ATTACK-3 using the same three points as the interesting points per clock cycle. Let “AT2 appc” and “AT3 appc” respectively denote the case of ATTACK-2 and ATTACK-3 using all the points as the interesting points per clock cycle. We conducted the eight attacks (plus PCA-based Template Attacks) with same traces both in the profiling stage and the extraction stage. We also used 10,000, 15,000, and 20,000 traces to build the 256 templates respectively. The traces were generated with a fixed main key and random plaintext inputs. Additional 20,000 traces which were used in the extraction stage were generated like those of the first group of experiments. We tested the success rates of the eight attacks when one uses ne traces in the extraction stage as follows. We repeated the attacks 1,000 times. For each time, we chose ne traces from the 20,000 traces uniformly at random and the attacks were conducted with the same ne traces. We respectively recorded how many times the attacks can successfully recover the correct subkey. The success rates of the eight attacks when Correlation Power Analysis based method was used are shown in Figure 2. The success rates of the eight attacks when Difference Of Means based method was used are shown in Figure 3.

Let the symbol "A>B" denotes the case that Attack A has obvious higher success rate than Attack B. Let the symbol "A≈B" denotes the case that Attack A has almost the same success rate as Attack B. From Figure 2, we find that AT1≈AT3 2ppc≈AT3 3ppc≈AT3 appc>AT2 2ppc≈PCA-TA>AT2 3ppc>AT2 appc. Figure 2 shows that, when more points are used, the success rates of our new way (AT3 2ppc, AT3 3ppc, and AT3 appc) are almost unchanged as the success rate of ATTACK-1 while classical Template Attacks (AT2 2ppc, AT2 3ppc, and AT2 appc) achieve lower success rates. This discovery shows that the accepted guideline of choosing interesting points for Template Attacks is correct. Therefore, it is not necessary to use more than one point as the interesting points per clock cycle when one uses the best method of choosing interesting points. From Figure 3, we find that AT3 2ppc≈AT3 3ppc≈AT3 appc>AT1≈AT2 2ppc≈PCA-TA>AT2 3ppc>AT2 appc. Figure 3 shows that, when more points are used, our new way achieve higher success rates than that of ATTACK-1. This discovery shows that Difference Of Means based method is not a good method of choosing interesting points for Template Attacks once more. Because the rest points in the clock cycles (i.e. $P_{(i,2)}, P_{(i,3)}, P_{(i,4)},\ i = 1, 2, 3$) also contain valuable information which can be exploited to achieve higher success rate.

Using our new way, more experiments showed that different methods of choosing interesting points (except PCA) lead to almost the same success rates when more than one point are used as the interesting points. Therefore, we suggest that one should obey the accepted guideline of choosing interesting points and uses the best method of choosing interesting points to conduct Template Attacks.

5 Conclusion and Future Work

In this paper, we show that Correlation Power Analysis based method and Sum Of Squared pairwise T-differences based method are the best choices of choosing interesting points for Template Attacks. This discovery contradicts with the previous result that PCA-based Template Attacks provide the best results [4]. Moreover, we find that some methods of choosing interesting points are essentially equivalent. In additional, we give out a more reasonable proof about the accepted guideline of choosing interesting points for Template Attacks by presenting a new way of conducting Template Attacks. In the future, it is necessary to further verify our results in other devices such as FPGA and ASIC.

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, 2013.
5. Bär, M., Drexler, H., Pulkus, J.: Improved Template Attacks. COSADE2010, 2010.
6. 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.
7. 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.
8. Oswald, E., Mangard, S.: Template Attacks on Masking—Resistance Is Futile. CT-RSA2007, LNCS 4377, pp.243-256, 2007.
9. 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.
10. Gierlichs, B., Lemke-Rust, K., Paar, C.: Templates vs. Stochastic Methods A Performance Analysis for Side Channel Cryptanalysis. CHES2006, LNCS4249, pp.15-29, 2006.
11. Mangard, S., Oswald, E., Popp, T.: Power Analysis Attacks: Revealing the Secrets of Smart Cards. Springer 2007.
12. Hanley, N., Tunstall, M., Marnane, W.P.: Unknown Plaintext Template Attacks. WISA2009, LNCS 5932, pp.148-162, 2009.
13. Jolliffe, I.: "Principal Component Analysis", John Wiley & Sons, Ltd, 2005.
14. 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.
15. Coron, J.-S., Kizhvatov, I.: Analysis and Improvement of the Random Delay Countermeasure of CHES 2009. CHES2010, LNCS 6225, pp.95C109, 2010.
16. Mather, L., Oswald, E., Bandenburg, J., Wójcik, M.: Does My Device Leak Information? An a *priori* Statistical Power Analysis of Leakage Detection Tests. ASIACRYPT2013 Part I, LNCS 8269, pp.486C505, 2013.
17. Gierlichs, B., Batina, L., Tuyls, P., Preneel, B.: Mutual Information Analysis. CHES2008. LNCS 5154, pp.426C442, 2008.
18. Whitnall, C., Oswald,E., Mather, L.: An Exploration of the Kolmogorov-Smirnov Test as a Competitor to Mutual Information Analysis. CARDIS2011, LNCS 7079, pp.234C251, 2011.
19. Zhang, H., Zhou, Y., Feng, D.: An Efficient Leakage Characterization Method for Profiled Power Analysis Attacks. ICISC2011, LNCS 7259, pp.61-73, 2011.
20. Feng, M., Zhou, Y., Yu, Z.: EMD-Based Denoising for Side-Channel Attacks and Relationships between the Noises Extracted with Diffierent Denoising Methods. ICICS2013, LNCS 8233, pp.259-274, 2013.