Template Attacks Based on Static Power Analysis of Block Ciphers in 45-nm CMOS Environment

Jiming Xu and Howard M. Heys
Department of Electrical and Computer Engineering
Memorial University of Newfoundland
{*jimingx, hheys*}*@mun.ca*

***Abstract*—Side-channel attacks are proven to be efficient tools in attacking cryptographic devices. Dynamic power has been used as a source for many well-known side-channel attack algorithms. As process technology size shrinks, the relative amount of static power consumption increases accordingly, and reaches a significant level in sub-100-nm chips, potentially changing the nature of side-channel analysis. In this paper we propose a type of template attack developed for static power analysis of block ciphers. In addition to the original template distinguisher, the attack is shown to work well using new distinguishers which are faster to compute.**

Keywords—side-channel attacks, static power, block ciphers, light-weight ciphers, cryptographic devices.

I. Introduction

A side-channel attack is based on the assumption that cryptographic circuits have some physical leakage related to the actual functions of the circuits. For a cryptographic device, the legitimate user provides inputs such as a plaintext and a key, and this device computes the ciphertext as output. Regarding this computation as the main channel, a device can also leak information that can reveal its behavior during computation, and such information, including power, heat, timing, and electromagnetic field, is referred to as side-channel leakage. The attacker can measure the side-channel leakage, and guess the related information in the main channel. By this means the attacker would be able to recover the input to the device. Usually we assume that the ciphertext or both plaintext and ciphertext are known to the attacker, and the key needs to be determined in order to break the cipher. The attacker would consider information from the main channel to see if it has any statistical correlation with the side-channel leakage.

Side-channel attacks using power leakage, namely power analysis, was first introduced in [1]. It is one of the major areas of side-channel attack, and has been the focus of researchers for many years. As the power analysis methods were initially proposed, they mainly focused on the dynamic behavior when the internal state of circuits change. The total power consumption of a chip consists of two parts, dynamic power and static power. The static power is the power consumption when there is no state change in the circuit. Assuming that the power supply voltage is fixed, static power is proportional to the current flowing into the circuit, also known as the static current.

It is predicted by International Technology Roadmap for Semiconductors (ITRS) that the static power consumption can be significant in technologies in sizes of 90 nm and less [2]. Additionally, with the shrinkage of technology size, the increase in static power is more significant than the increase in dynamic power, making static power a potential source for power analysis. We will refer to the power analysis attacks using static power as the measurement target as static power analysis (also referred to as leakage power analysis in [3]).

Various researches have shown how to apply existing power analysis algorithms to static power and the practicality of static power analysis has been demonstrated in different types of platforms from simulations of CMOS circuits to Field Programmable Gate Arrays (FPGA). A differential power analysis (DPA) [1] attack against the DES cipher using static power leakage was performed in [4], successfully showing that static power can give better attack results than dynamic power in 180-nm technology. More recent publications mainly use correlation power analysis (CPA) [5] as the attacking method. The static power analysis of 90-nm and 65-nm devices was considered by Alioto et. al. in [3], as well as the performance of CPA using static power on cryptographic circuits. After this, they further explored the effectiveness of static-power-based CPA against circuits enhanced with countermeasures for classical DPA attacks in [6]. The CPA in [3] and [6] are performed in a CMOS simulation environment. Moradi applied static-power-based CPA on real-world FPGA chips in [7], and static power analysis is shown to be successful against some masked AES S-box implementations.

In our paper, we apply template attacks [8] to static power of simulated CMOS devices in 45-nm technology, which is a more up-to-date environment. Our target devices are low frequency devices using light-weight block ciphers. In practice, these devices may include microcontrollers, RFID tags, and smart cards. Block ciphers usually work in specific encryption modes where the same key is repeatedly used in multiple encryptions. In such scenarios we are able to get multiple power traces for the same key in our attack process. Hence, instead of the original distinguisher used in [8], we treat different trace sets as different distributions and distinguish between these distributions. We will compare the performance of our distinguishers with the original one used in [8].

II. Template Attack Using Static Power Analysis

A template attack consists of two phases: the *profiling phase* and the *attacking phase*. Two devices, known as the profiling device and the target device, are required for these phases respectively. Similar to DPA and CPA, the target device is the device that the attacker tries to break by recovering the unknown key. Template attacks require a profiling device which is similar to the target device which the attacker has full access to. The attacker is able to modify the keys of this device and create different templates. The steps of a template attack using static power analysis are as follows [8]:

1) Use random plaintext input to generate m_T traces for each of the K possible subkeys using the profiling device. Hence there are K sets of traces, each having m_T traces of random encryption. Let there be N measurement points in each trace. For a static-power-based attack, only 1 measurement is needed to represent the power consumption when the input/output values are not changed at each different input value. That is, only 2 measurements are needed in each clock period, one when clock is high and the other when clock is low.
2) Use statistical techniques to select n $(n \leq N)$ points-of-interest (POIs) where the Gaussian distribution model applies well. In a noise-free simulation environment some of the measurement points may be highly correlated, which would cause the covariance matrix in step 3 to be singular. Principle component analysis [9] can be applied in this step to address this issue. Otherwise we could diagonalize the covariance matrix in step 3, by which we manually remove any possible correlation between the measurement points. This option addresses the issue of singularity at the cost of removing all the information about the relation between different measurement points.
3) Apply the multivariate Gaussian model to each of the K sets of traces separately. That is, regarding each POI as a univariate Gaussian variable, the vector of all POIs is regarded as a multivariate Gaussian distribution. For each possible subkey $k \in [0, K-1]$, let the trace vector generated using subkey k be $\mathbf{t}_{k,i}, 1 \leq i \leq m_T$. Note that the traces only contain chosen POIs now. Compute the mean vector $\bar{\mathbf{t}}_k$ and covariance matrix $\mathbf{\Sigma}_k$ as follows:

$$\bar{\mathbf{t}}_k = \frac{1}{m_T} \sum_{i=1}^{m_T} \mathbf{t}_{k,i}, \tag{1}$$

$$\mathbf{\Sigma}_k = \frac{1}{m_T - 1} \sum_{i=1}^{m_T} (\mathbf{t}_{k,i} - \bar{\mathbf{t}}_k)(\mathbf{t}_{k,i} - \bar{\mathbf{t}}_k)^T. \tag{2}$$

Here $\bar{\mathbf{t}}_k$ and $\mathbf{\Sigma}_k$ are referred to as template parameters. The probability density function (PDF) for this multivariate Gaussian distribution is

$$f(\mathbf{t}|\bar{\mathbf{t}}_k, \mathbf{\Sigma}_k) = 1/\left(\sqrt{(2\pi)^n |\mathbf{\Sigma}_k|}\right) \cdot \exp\left(-\frac{1}{2}(\mathbf{t} - \bar{\mathbf{t}}_k)^T \mathbf{\Sigma}_k^{-1} (\mathbf{t} - \bar{\mathbf{t}}_k)\right). \tag{3}$$

4) Generate a set of m_A traces from the target device by performing the same operation as the profiling device using random plaintexts and an unknown subkey. The measurement points and the POIs are also chosen to be the same as those in the profiling phase. Given trace $\mathbf{t}'_i$ $(1 \leq i \leq m_A)$ in the trace set, the probability that this trace is derived from subkey k, according to Bayes' rule, follows

$$Pr(k|\mathbf{t}'_i) \propto f(\mathbf{t}'_i|\bar{\mathbf{t}}_k, \mathbf{\Sigma}_k). \tag{4}$$

Hence $f(\mathbf{t}'_i|\bar{\mathbf{t}}_k, \mathbf{\Sigma}_k)$ is defined as the likelihood that $\mathbf{t}'_i$ is generated using k. Using Equation (4), we rank the likelihood computation result for all the subkeys. The subkey with the highest likelihood is assumed to be the correct key used in the target device.

Steps 1-3 are defined as the *profiling phase*, while step 4 is defined as the *attack phase*.

The original proposal in [8] uses 1 attacking trace to perform a template attack. In our scenario when targeted at block ciphers we are able to acquire multiple attacking traces. Since all the traces are acquired independently, we define the overall likelihood given all the traces as

$$Pr(k|\{\mathbf{t}'_i : 1 \leq i \leq m_A\}) = \prod_{i=1}^{m_A} Pr(k|\mathbf{t}'_i) \tag{5}$$

III. New Distinguishers For Profiled Attacks

Knowing two random variables P and Q that follow some unspecified distributions, the purpose of distinguishers is to identify how much they differ from each other. We propose to use the distinguishers that use only the parameters such as mean and covariance to distinguish between different distributions. For practical purposes, if there are multiple attacking traces, the attacker would not have to compute Equation (4) for each single attacking trace. Instead, the attacker would only need to generate the mean and covariance for the attacking set, and compute the distinguishers once for each template.

Regarding each set of template traces and the attack traces as a distribution, there are in total $K + 1$ distributions. In the static-power-based template attack, different statistical metrics can be used as the distinguisher to evaluate the difference between the distribution of attack traces and the distribution of template traces. Here we show two distinguishers we use in our experiment:

- Kullback-Leibler divergence (KL divergence) [10], also called relative entropy, is defined as

$$D_{KL}(P||Q) = \int_x p(x) \log \frac{p(x)}{q(x)}, \tag{6}$$

here $p(\cdot)$ and $q(\cdot)$ represent the PDF of distributions P and Q respectively.
- Jensen-Shannon distance (JS distance) [11], which is based on the definition of KL divergence, is defined as

$$D_{JS}(P||Q) = \sqrt{\frac{1}{2}D_{KL}(P||M) + \frac{1}{2}D_{KL}(Q||M)}, \tag{7}$$

where M is a Gaussian mixture of P and Q.

Now we apply KL divergence and JS distance to multivariate Gaussian distribution, which is used by template attack to describe the distribution of POIs. Letting $\mathbf{P}$ and $\mathbf{Q}$ be two multi-dimensional vectors that satisfy multivariate Gaussian distribution, and $\mathbf{M}$ be a Gaussian mixture of $\mathbf{P}$ and $\mathbf{Q}$, there is no closed form expression for $D_{KL}(\mathbf{P}||\mathbf{M})$ and $D_{KL}(\mathbf{Q}||\mathbf{M})$ [12]. The mean vector and covariance matrix of $\mathbf{M}$ needs to be approximated in order to compute the KL divergence between $\mathbf{M}$ and $\mathbf{P}$ and between $\mathbf{M}$ and $\mathbf{Q}$. Here we compute the mean vector and the covariance matrix for $\mathbf{M}$ as:

$$\overline{\mathbf{M}} = \frac{1}{2}(\overline{\mathbf{P}} + \overline{\mathbf{Q}}), \tag{8}$$

$$\mathbf{\Sigma_M} = \frac{1}{2}\mathbf{\Sigma_P}(\overline{\mathbf{P}} - \overline{\mathbf{M}})(\overline{\mathbf{P}} - \overline{\mathbf{M}})^T + \frac{1}{2}\mathbf{\Sigma_Q}(\overline{\mathbf{Q}} - \overline{\mathbf{M}})(\overline{\mathbf{Q}} - \overline{\mathbf{M}})^T. \tag{9}$$

The approximation method we use here is simple and quick to compute, but this method does not guarantee the best approximation result. Other alternatives are also provided in [13].

For multivariate Gaussians, the KL divergence is [14]:

$$\begin{aligned} D_{KL}(\mathbf{P}||\mathbf{Q}) =& \frac{1}{2}\left(tr(\mathbf{\Sigma_Q}^{-1}\mathbf{\Sigma_P}) - n + \log\frac{|\mathbf{\Sigma_Q}|}{|\mathbf{\Sigma_P}|} \right. \\ & \left. + (\overline{\mathbf{Q}} - \overline{\mathbf{P}})^T \mathbf{\Sigma_Q}^{-1}(\overline{\mathbf{Q}} - \overline{\mathbf{P}}) \right). \end{aligned} \tag{10}$$

Here $tr(\cdot)$ is the trace of the matrix, which is defined as the sum along diagonals of the matrix, and n is the dimension of the covariance matrices $\mathbf{\Sigma}_P$ and $\mathbf{\Sigma}_Q$, which is the number of POIs. Similar to the univariate Gaussian distribution case, the JS distance can also be computed using Equation (7).

As another approach, in order to add the symmetric property to KL divergence in our experiment, we use the average of two KL divergences, also known as the Jeffreys divergence or J-divergence [15], which is

$$D_J(\mathbf{P}||\mathbf{Q}) = \frac{1}{2}(D_{KL}(\mathbf{P}||\mathbf{Q}) + D_{KL}(\mathbf{Q}||\mathbf{P})). \tag{11}$$

Note that since we are treating the attack set and template sets as distributions, there are also other distinguishers that can be used to compare the likelihood of distributions. We leave this as potential future work.

IV. Experimental Results

In order to explore the application of template attacks on the static power leakage of cryptographic circuits, we have created simple circuits for study in 45-nm CMOS. Our circuits are implemented using RTL level Verilog with the synthesis and layout performed using the FreePDK 45nm library [16]. The generated netlists are imported into the Virtuoso environment to perform transistor level simulation. The clock period of our target circuits is set to be 200 ns. The current at the power input pin is measured as it is proportional to the power consumption. In simulation, the current sharply changes when the input values change, since at this time the dynamic power is dominant. After this the current trace would slowly approach I_{static}, representing at when the static power is dominant. Figure 1 illustrates this process. In practice we wait for a period of time which is long enough so that the contribution of dynamic power can be ignored, hence the measurement result is close enough to I_{static}. In our case, we wait for 90 ns after the clock values change.

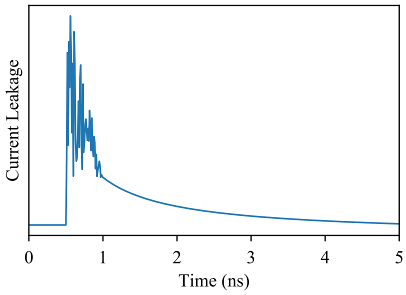


Fig. 1. An illustration of current behavior after input value changes.

Our first attack target is a simple 4-bit toy cipher which consists of an S-box from the cipher PRESENT [17] and a 4-bit XOR operation. The structure of this cipher is shown in Figure 2.

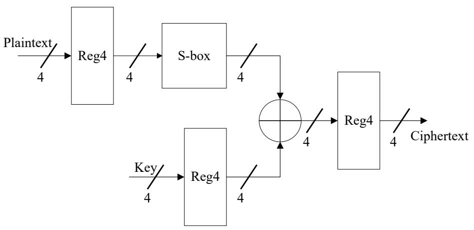
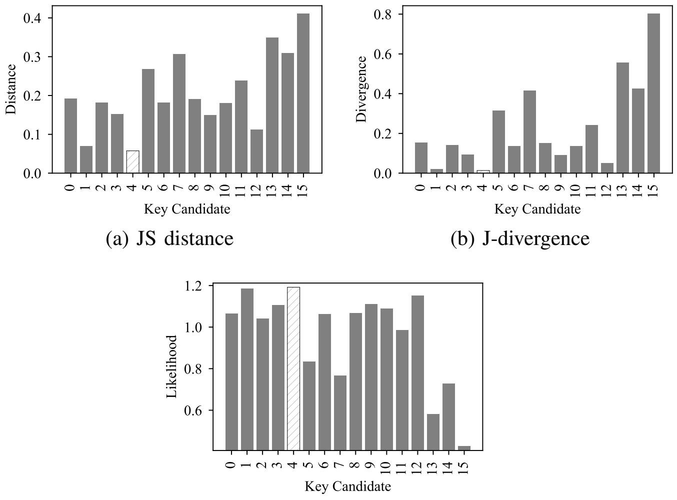


Fig. 2. Structure of a simple toy cipher.

In this circuit, the encryption is done in 1 clock period, so 2 POIs (one for the clock high and the other for the clock low) are enough for static power analysis. In order to perform a template attack on this circuit, we profiled 16 template sets using 16 different key guesses. For each key guess, we profiled using different sets of 50 random plaintext inputs. Next, assuming that we attacked on key 0100 (4 in decimal), we used this key to encrypt another set of 50 random plaintext inputs. The result of this attack is shown in Figure 3. All of the three distinguishers gain successful results as key 4 has the lowest JS distance and J-divergence, as well as highest Gaussian likelihood.

We also performed this attack using 1000 traces for the template and attack sets to verify the computation time of the distinguishers. Using Python with the Intel Math Kernel library on a computer with i7-4790 CPU, we are able to achieve our results in an elapsed time of around $0.023\,\mathrm{s}$ for J-divergence, $0.052\,\mathrm{s}$ for JS distance, and $0.599\,\mathrm{s}$ for Gaussian likelihood.

Next, we performed our attack on a simplified key schedule circuit of the PRESENT cipher. The structure of this circuit is proposed in [18]. We reduced the key size to 24 bits, and the number of rounds to 2. The 24 key bits are separated into 6 4-bit key nibbles, and we attack on the first key nibble loaded in the key register. We also used a 4-bit state register to store plaintext inputs, and this component adds noise to the power consumption of the key schedule circuit. We performed our attack following a similar attack process that we applied to the toy cipher. The difference is that in this attack we try to recover the Hamming weight of the correct key. Registers are the major source for power consumption in this circuit, and as is shown in [3], the static power consumption of registers



(a) JS distance

(b) J-divergence

(c) Gaussian likelihood

Fig. 3. Template attack results on key 0100 using 50 traces for template and 50 traces for attack.

follow the Hamming weight model, thereby making it difficult to distinguish between keys of the same Hamming weight.

We performed 16 attacks using a different correct key nibble (from 0000 to 1111) for each attack. We used the same number of traces used for profiling and attacking, and this number range from 2 to 128. Then we used the success rate defined in [19] to evaluate the result of our attacks. The result of this attack is shown in Figure 4. The success rates using different distinguishers increase as the number of traces increases, and approaches 100% with less than 60 traces. As is shown the new distinguishers gain the same level of performance as the Gaussian distinguisher. Using 128 traces for profiling and 128 traces for attack, the time for a single attack is around 0.019 s for J-divergence, 0.030 s for JS distance, and 0.051 s for Gaussian likelihood.

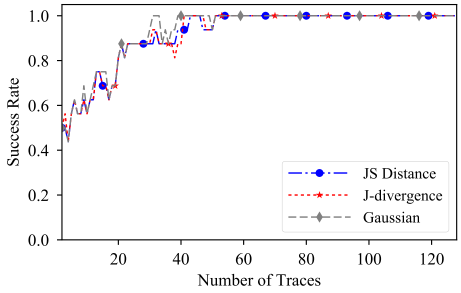


Fig. 4. Success rate of template attacks on the key schedule circuit. The success rate is the number of successful attacks in all the 16 attacks.

V. Conclusion

In this paper we extend the practicability of profiled side-channel attacks to static power consumption, and we have gained success in attacks against block cipher circuits simulated in 45-nm CMOS environment. We also introduce a set of new distinguishers to template attacks against block ciphers, which can achieve the same level of performance as the original distinguisher using less computing time.

References

[1] P. C. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," in *Proceedings of the 19th Annual International Cryptology Conference on Advances in Cryptology*, ser. CRYPTO '99. London, UK, UK: Springer-Verlag, 1999, pp. 388–397.

[2] M. Keating, D. Flynn, R. Aitken, A. Gibbons, and K. Shi, *Low Power Methodology Manual: For System-on-Chip Design*. Springer Publishing Company, Incorporated, 2007.

[3] M. Alioto, L. Giancane, G. Scotti, and A. Trifiletti, "Leakage power analysis attacks: A novel class of attacks to nanometer cryptographic circuits," *Circuits and Systems I: Regular Papers, IEEE Transactions on*, vol. 57, no. 2, pp. 355–367, Feb 2010.

[4] L. Lin and W. Burleson, "Leakage-based differential power analysis (ldpa) on sub-90nm cmos cryptosystems," in *Circuits and Systems, 2008. ISCAS 2008. IEEE International Symposium on*, May 2008, pp. 252–255.

[5] E. Brier, C. Clavier, and F. Olivier, "Correlation power analysis with a leakage model," in *Cryptographic Hardware and Embedded Systems - CHES 2004*, ser. Lecture Notes in Computer Science, M. Joye and J.-J. Quisquater, Eds. Springer Berlin Heidelberg, 2004, vol. 3156, pp. 16–29.

[6] M. Alioto, S. Bongiovanni, M. Djukanovic, G. Scotti, and A. Trifiletti, "Effectiveness of leakage power analysis attacks on dpa-resistant logic styles under process variations," *Circuits and Systems I: Regular Papers, IEEE Transactions on*, vol. 61, no. 2, pp. 429–442, Feb 2014.

[7] A. Moradi, "Side-channel leakage through static power," in *Cryptographic Hardware and Embedded Systems - CHES 2014*, ser. Lecture Notes in Computer Science, L. Batina and M. Robshaw, Eds. Springer Berlin Heidelberg, 2014, vol. 8731, pp. 562–579.

[8] S. Chari, J. R. Rao, and P. Rohatgi, "Template attacks," in *Cryptographic Hardware and Embedded Systems-CHES 2002*. Springer, 2002, pp. 13–28.

[9] C. Archambeau, E. Peeters, F.-X. Standaert, and J.-J. Quisquater, "Template attacks in principal subspaces," in *International Workshop on Cryptographic Hardware and Embedded Systems*. Springer, 2006, pp. 1–14.

[10] S. Kullback and R. A. Leibler, "On information and sufficiency," *The annals of mathematical statistics*, vol. 22, no. 1, pp. 79–86, 1951.

[11] D. M. Endres and J. E. Schindelin, "A new metric for probability distributions," *IEEE Transactions on Information theory*, 2003.

[12] K. T. Abou-Moustafa and F. P. Ferrie, "A note on metric properties for some divergence measures: The gaussian case." in *ACML*, 2012, pp. 1–15.

[13] J. R. Hershey and P. A. Olsen, "Approximating the kullback leibler divergence between gaussian mixture models," in *2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP'07*, vol. 4. IEEE, 2007, pp. IV–317.

[14] J. Duchi, "Derivations for linear algebra and optimization," *Berkeley, California*, 2007.

[15] J. Lin, "Divergence measures based on the shannon entropy," *IEEE Transactions on Information theory*, vol. 37, no. 1, pp. 145–151, 1991.

[16] FreePDK. [Online]. Available: https://www.eda.ncsu.edu/wiki/FreePDK

[17] A. Bogdanov, L. R. Knudsen, G. Leander, C. Paar, A. Poschmann, M. J. Robshaw, Y. Seurin, and C. Vikkelsoe, "Present: An ultra-lightweight block cipher," in *International Workshop on Cryptographic Hardware and Embedded Systems*. Springer, 2007, pp. 450–466.

[18] C. Rolfes, A. Poschmann, G. Leander, and C. Paar, "Ultra-lightweight implementations for smart devices–security for 1000 gate equivalents," in *International Conference on Smart Card Research and Advanced Applications*. Springer, 2008, pp. 89–103.

[19] F.-X. Standaert, T. G. Malkin, and M. Yung, "A unified framework for the analysis of side-channel key recovery attacks," in *Annual International Conference on the Theory and Applications of Cryptographic Techniques*. Springer, 2009, pp. 443–461.