LETTER

# A White-Box Cryptographic Implementation for Protecting against Power Analysis

**Seungkwang LEE**[†a)], ***Member***

**SUMMARY** Encoded lookup tables used in white-box cryptography are known to be vulnerable to power analysis due to the imbalanced encoding. This means that the countermeasures against white-box attacks can not even defend against gray-box attacks. For this reason, those who want to defend against power analysis through the white-box cryptographic implementation need to find other ways. In this paper, we propose a method to defend power analysis without resolving the problematic encoding problem. Compared with the existing white-box cryptography techniques, the proposed method has twice the size of the lookup table and nearly the same amount of computation.

***key words:*** *white-box cryptography, power analysis, countermeasure*

## 1. Introduction

There are three major layers of attacks against software cryptographic implementation [1]. First, the black-box model is located in the lowest layer and provides an attacker with only the input and output values for the cryptographic implementation. The gray-box model, often referred to as a side-channel attack, is a more powerful attack providing the attacker with additional information such as the execution time of the encryption algorithm, the power leakage value, and the electromagnetic wave in addition to the input and output values. Finally, in addition to all these, the white-box model is the most powerful form of attack that allows access to all information in the device when the cryptographic algorithm is executed, and even allows modification of internal information related to the running software. On cryptographic protecting techniques for each attack layer, the white-box cryptography is particularly to protect against the white-box attack model. It should be noted here that the countermeasures against higher-layer attacks were considered to include defense against lower-layer attacks. For this reason, white-box cryptographic implementation must be able to defend against the gray-box attack model [2]. However, it was found that there was a correlation between the encoded value and the non-encoded value due to the imbalanced encoding applied to the lookup table generation in the white-box implementation. Thus, the gray-box attack succeeded [3], [4]. Thus, white-box cryptography is vulnerable to not only algebraic analysis, but also gray-box attacks.

In this paper, we propose a method to protect against gray-box attacks, more precisely power analysis, using the existing white-box cryptographic implementation. Compared to the performance of existing white-box cryptography roughly [2], the lookup table size is two times and the computational cost required for encryption or decryption is almost the same. In Sect. 2 we briefly overview a white-box implementation and its vulnerability to power analysis. In Sect. 3, we present a method to withstand power analysis using the white-box implementation technique, and provide security and performance evaluation of our proposed method. Section 4 concludes this paper.

Manuscript received August 22, 2017.
Manuscript publicized October 19, 2017.

[†]The author is with ETRI (Electronics and Telecommunications Research Institute), Korea.
a) E-mail: skwang@etri.re.kr

DOI: 10.1587/transinf.2017EDL8186

## 2. Background

The white-box attack model allows the attacker to gain total control over the execution environment and the software implementation itself. For hiding the secret key the white-box cryptographic implementation generally follows the principle below [2].

1. Generate a series of pre-computed lookup tables for all input values and the secret key for the cryptographic algorithm.
2. Apply linear and non-linear encoding to hide the secret key combined with the lookup tables.
3. The encoding applied to a lookup value is canceled by the input decoding applied to the next lookup table.

Although the encoding randomizes the Hamming weight of the intermediate values, it is known to be imbalanced [4], and thus leads to a bit-to-bit correlation before and after the encoding. This imbalance makes it possible to perform power analysis including Differential Power Analysis (DPA) [5] or Correlation Power Analysis (CPA) [6] based on the mono-bit model. Especially Differential Computation Analysis (DCA) [3] uses a noise-free software execution trace obtained by the dynamic binary instrumentation (DBI) framework instead of collecting power traces using an oscilloscope. The accuracy and efficiency of DCA is outstanding because there is no noise in the software trace, but it may belong to a somewhat higher level of attack than the classical power analysis due to the use of DBI.

By the following Definition 1 [4], we can measure the problematic imbalance using the Walsh transform. This is because the more large the absolute value of $W_f(\omega)$, the more strong the correlation between $f(x)$ and $x \cdot \omega$.

**Definition 1:** Let $x = \langle x_1, \ldots, x_n \rangle$, $\omega = \langle \omega_1, \ldots, \omega_n \rangle$ be elements of $\{0,1\}^n$ and $x \cdot \omega = x_1\omega_1 \oplus \cdots \oplus x_n\omega_n$. Let $f(x)$ be

a Boolean function of $n$ variables. Then the Walsh transform of the function $f(x)$ is a real valued function over $\{0,1\}^n$ that can be defined as $W_f(\omega) = \sum_{x\in\{0,1\}^n}(-1)^{f(x)\oplus x\cdot\omega}$.

Taking into power analysis, let us denote the combination of a target intermediate value $x$, linear and non-linear encoding by $s$ Boolean functions $f_{i\in\{1,\ldots,s\}}(x)\colon \{0,1\}^8 \to \{0,1\}$. In the case of $\mathrm{HW}(\omega) = 1$, one can find how each encoded bit and a target hypothetical bit are correlated to each other by $W_{fi}(\omega) = \sum_{x\in\{0,1\}^n}(-1)^{f_i(x)\oplus x\cdot\omega}$, where $f_i(x)$ is an encoded lookup bit, $x\cdot\omega$ is a hypothetical bit and HW is the Hamming weight. If they are (negatively) correlated to each other, $W_{fi}(\omega)$ is (negatively) large enough. In the presence of such imbalanced encoding, we present a method of preventing power analysis for a white-box implementation and demonstrate its security using the Walsh transform in the following section.

## 3. Proposed Method

### 3.1 Key Idea

Let $\mathcal{E} = (\mathcal{T}^0, \mathcal{T}^1, \gamma)$ be the proposed method where $\mathcal{T}^0 = \{t_1^0, t_2^0, \ldots, t_l^0\}$ and $\mathcal{T}^1 = \{t_1^1, t_2^1, \ldots, t_l^1\}$, two series of $n \times s$ lookup tables for a white-box implementation of a block cipher, and $\gamma \in_R \{0,1\}$. For all $x \in \{0,1\}^n$, let us have $\mathcal{T}^0$ and $\mathcal{T}^1$ such that

$$\text{if } y = t_i^0(x) \text{ then } \bar{y} = t_i^1(\bar{x}),$$

where $i \in [1, l]$. Thus, the generation of $\mathcal{T}^1$ can be easily performed using $\mathcal{T}^0$. Our proposed method is simple: we randomly pick $\gamma$ and use $\mathcal{T}^\gamma$ as lookup tables for each execution of an encryption (or decryption). When using $\mathcal{T}^1$, we have to flip all bits of the plaintext and ciphertext.

The prevention of power analysis strongly relies on the uniformly distributed $\gamma$. If a particular bit of $t_i^0(x)$ correlates with $x$, the corresponding bit of $t_i^1(x)$ negatively correlates. Our goal is to reduce the correlation coefficient at the attacked point in the power traces with randomly picked $\gamma$.

### 3.2 Security Evaluation

Suppose that $f_i(x)$ positively correlates to $x \cdot \omega$ while $\overline{f_i(x)}$ negatively correlates to $x \cdot \omega$ for particular $i$. We then define $g_i(x)$ and $W_{gi}(\omega)$ as follows:

$$g_i(x) = \begin{cases} f_i(x) & \text{if } \gamma = 0 \\ \overline{f_i(x)} & \text{else } \gamma = 1, \text{ and} \end{cases}$$

$$W_{gi}(\omega) = \sum_{x\in\{0,1\}^n} (-1)^{g_i(x)\oplus x\cdot\omega}.$$

Because of randomly selected $\gamma$, $g_i(x)$ will be sometimes positively and sometimes negatively correlated to $x \cdot \omega$. Thus, the absolute value of $W_{gi}(\omega)$ will be small, which has an effect similar to a reduction in the encoding imbalance. In the following, we apply the proposed method to details of the DPA and CPA attacks to demonstrate the resistant to power analysis.

**Against DPA**. DPA selection function $D(C, b, k^*)$ [5] is defined to select a target intermediate bit to analyze by computing the value of bit $b$, given ciphertext $C$ and key candidate $k^*$. If $k^*$ is incorrect, $D(C, b, k^*)$ will evaluate the correct value for $b$ with probability $1/2$ for each ciphertext [7]. To launch DPA, an attacker observes $M$ encryption operations and records power traces $\mathbf{V}_{1..M}[1..\kappa]$ containing $\kappa$ samples each. Then the differential trace $\Delta_D[j]$, where $1 \le j \le \kappa$, is computed by finding the difference between the average of the traces for $D(C, b, k^*) = 1$ and the average of the traces for $D(C, b, k^*) = 0$ as follows:

$$\begin{aligned}\Delta_D[j] &= \frac{\sum_{m=1}^{M} D(C_m, b, k^*)\mathrm{V}_m[j]}{\sum_{m=1}^{M} D(C_m, b, k^*)} \\ &\quad - \frac{\sum_{m=1}^{M}(1 - D(C_m, b, k^*))\mathrm{V}_m[j]}{\sum_{m=1}^{M}(1 - D(C_m, b, k^*))} \\ &\approx 2\left(\frac{\sum_{m=1}^{M} D(C_m, b, k^*)\mathrm{V}_m[j]}{\sum_{m=1}^{M} D(C_m, b, k^*)} - \frac{\sum_{m=1}^{M}\mathrm{V}_m[j]}{M}\right)\end{aligned}$$

Because $\mathrm{V}_m[j]$ is randomized by $\gamma$ in our proposed method, $D(C_m, b, k^*)$ will not correctly matched to $\mathrm{V}_m[j]$ for about half of the ciphertext $C_m$ even with the correct key. For this reason, the difference in the average of the subsets will approach zero as the subset sizes become infinity even if $D$ correctly divides a set into two subsets. This gives us

$$\lim_{M\to\infty} \Delta_D[j] \approx 0.$$

Thus we can conclude that the correct key is unlikely to be identified from its differential trace and thus we can prevent DPA on our white-box implementation.

**Against CPA**. CPA [6] is an extension of DPA where a power consumption model including the Hamming weight and the Hamming distance is applied in the analysis phase of an attack. In this case, our power model is the bit model because other-model-based CPA attacks on the white-box implementation are unlikely to be successful due to the randomized Hamming weight by the encoding. Given $\mathbf{V}_{1..M}[1..\kappa]$, the mono-bit CPA attacker will estimate the power consumption in each trace using the hypothetical intermediate value. Let there are $K$ different subkeys that we want to analyze and let $h_{m,k^*}$ $(1 \le m \le M, 0 \le k^* < K)$ be the power estimate in trace $m$, on the assumption that the subkey is $k^*$. To measure a linear relationship between hypothetical power consumption and measured power traces, the estimator $r$ is defined as follows [7]:

$$r_{k^*,j} = \frac{\sum_{m=1}^{M}(h_{m,k^*} - \overline{h_k^*}) \cdot (V_m[j] - \overline{V[j]})}{\sqrt{\sum_{m=1}^{M}(h_{m,k^*} - \overline{h_k^*})^2 \cdot \sum_{m=1}^{M}(V_m[j] - \overline{V[j]})^2}},$$

where $\overline{h_k^*}$ and $\overline{V[j]}$ are sample means of $h_k^*$ and $V[j]$, respectively. If a correlation occurs then there will be a noticeable spike in the correlation plot for the correct subkey value.

Let's take a close look at the sum of product of deviation scores $\sum_{m=1}^{M}(h_{m,k^*} - \overline{h_k^*}) \cdot (V_m[j] - \overline{V[j]})$. Due to the

confusion and diffusion effects of a block cipher, we expect that $\overline{h_k^*} = 0.5$, where the power estimate $h_{m,k^*}$ is the target intermediate bit in the bit model. This gives us $(h_{m,k^*} - \overline{h_k^*})$ becomes 0.5 or $-0.5$ depending on the value of $h_{m,k^*}$. In addition, $(V_m[j] - \overline{V[j]})$ will have a different sign but the same absolute value depending on the value of $V_m[j]$ which is in turn decided by $\gamma$. Thus, if $\gamma$ is selected uniformly at random, the sum of product of deviation scores approaches zero. Of course, this is a theoretical analysis that differs from the real execution environment. The main point is to defend the CPA attack by making the sum of product of deviation scores much smaller through randomly selected $\gamma$.

**Other considerations**. There are several attacks on the white-box implementations including DCA and the Zero Difference Enumeration (ZDE) [8] attacks. DCA is a kind of advanced power analysis performed with noise-free software traces, and ZDE addresses the misalignment of DCA memory traces by the countermeasures such as control flow obfuscation and randomization of table location. In the process of collecting the noise-free software traces, the memory read, write and accessed address are recorded, requiring more privileges than the existing power analysis attackers. Note that our proposed method is to prevent power analysis that is unable to manipulate internal memory. If we extend the DCA attacker's privileges to have a control over $\gamma$ and fix it like other white-box attackers, our proposed method that is gray-box attack resistant can be broken. However, this attack model is out of scope in this paper.

Our proposed method is also not subject to the higher-order DPA attacks [9], [10] because an attacker can not exploit the joint leakage of several intermediate values due to the linear and non-linear encodings of the white-box implementation.

### 3.3 Experimental Result

This section provides experimental results and substantiates our claims. To show the vulnerabilities of the previous white-box implementation, we performed power analysis using SCARF [11] with fixed $\gamma$ on the non-protected white-box AES-128 implementation [2]. To do so, we generated binary traces consisting of only *0*s and *1*s of all intermediate values as shown in Fig. 1. This increases the efficiency of power analysis because it is noise-free as DCA software traces.

With 10,000 random plaintexts, we conducted CPA on the LSB of the S-box output of the first subbyte in the first round. When fixing $\gamma = 0$ and $\gamma = 1$, the CPA attacks revealed the correct key as shown in Figs. 2 and 3, respectively. It is noticeable that the correlation coefficients of the correct key have the same absolute values at the same point in the two correlation plots but have different signs.

On the other hand, Fig. 4 shows a CPA correlation plot with 10,000 random plaintexts and random $\gamma$ on the LSB of the S-box output in the first round. The correlation coefficient of the correct key steeply decreases and thus the attacker can not reveal the key. (Because the resistance to CPA means the resistance to DPA, we did not conduct DPA attacks.)

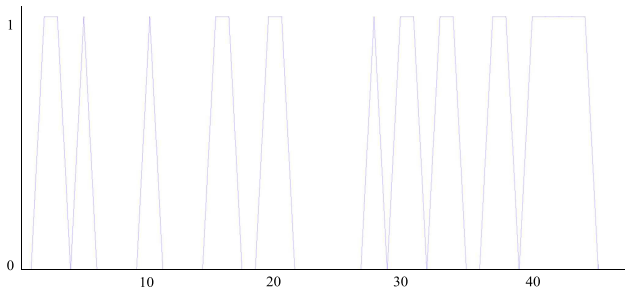


**Fig. 1** A zoomed binary trace for SCARF. Value vs. point.

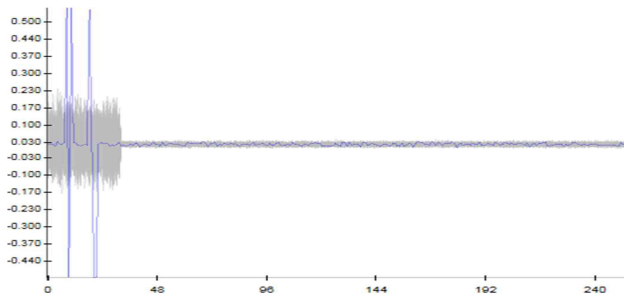


**Fig. 2** CPA result when fixing $\gamma = 0$. Correlation coefficient vs. point. Blue line: correct key, gray line: wrong key candidates.

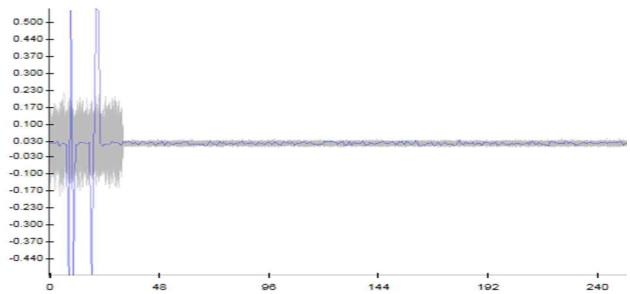


**Fig. 3** CPA result when fixing $\gamma = 1$. Correlation coefficient vs. point. Blue line: correct key, gray line: wrong key candidates.

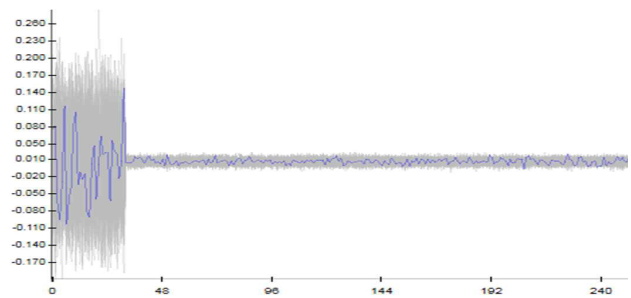


**Fig. 4** CPA result when using random $\gamma$. Correlation coefficient vs. point. Blue line: correct key, gray line: wrong key candidates.

### 3.4 Performance Evaluation

Compared to the non-protected white-box implementation, the lookup table size increases two times because we have two complement ones, $\mathcal{T}^0$ and $\mathcal{T}^1$. For example, our proposed method of the white-box AES-128 implementation requires 1,630,208 bytes (= 815,104 × 2) excluding the external encoding. When it comes to the computational cost, most operations in white-box encryption (or decryption) consist of table lookups. For example, the white-box AES-128 encryption without the external encoding can be performed with 2,032 lookups. In the case of our proposed method, the main additional operation is to randomly generate $\gamma$. In addition, when $\gamma = 1$, an operation to flip the plaintext and the ciphertext is required. Of course, the flipping can be applied in advance when generating the lookup table for $\gamma = 1$. As a result, the additional cost of our proposed method to defend power analysis is not that significant.

## 4. Conclusion

In this paper, we proposed a method to defend power analysis on the white-box cryptography. To do so, two sets of lookup tables are generated, in which the input and output values are complementary to each other. Then one set of lookup tables positively correlates with the intermediate values computed with the correct key, while the other set negatively correlates. The key idea behind the proposed method is to randomly select one of the two lookup table sets before each encryption operation to reduce the correlation. To demonstrate its security, we have shown the reduced encoding imbalance, and the resistance to DPA and CPA attacks. The additional costs of the proposed method are twice the memory space for storing the lookup tables and random bit generation compared with the existing white-box cryptographic implementation.

## Acknowledgments

This work was supported by Institute for Information & communications Technology Promotion (IITP) grant funded by the Korea government (MSIT) (No.2016-0-00399, Study on secure key hiding technology for IoT devices [KeyHAS Project]).

### References

[1] S. Lee, D. Choi, and Y.-J. Choi, "Conditional re-encoding method for cryptanalysis-resistant white-box AES," ETRI Journal, vol.37, no.5, pp.1012–1022, Oct. 2015.
[2] S. Chow, P. Eisen, H. Johnson, and P.C. Van Oorschot, "White-box cryptography and an AES implementation," Proc. Ninth Workshop on Selected Areas in Cryptography (SAC 2002), Lecture Notes in Computer Science, vol.2595, pp.250–270, Springer-Verlag, 2002.
[3] J.W. Bos, C. Hubain, W. Michiels, and P. Teuwen, "Differential computation analysis: Hiding your white-box designs is not enough," IACR Cryptology ePrint Archive, vol.2015, p.753, 2015.
[4] P. Sasdrich, A. Moradi, and T. Güneysu, "White-box cryptography in the gray box — A hardware implementation and its side channels," 23rd International Conference on Fast Software Encryption, FSE 2016, Bochum, Germany, Lecture Notes in Computer Science, vol.9783, pp.185–203, 2016.
[5] P. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," Advances in Cryptology — CRYPTO '99, Lecture Notes in Computer Science, vol.1661, pp.388–397, Springer-Verlag, 1999.
[6] E. Brier, C. Clavier, and F. Olivier, "Correlation power analysis with a leakage model," Proc. 6th International Workshop on Cryptographic Hardware and Embedded Systems, CHES 2004, Cambridge, MA, USA, Lecture Notes in Computer Science, vol.3156, pp.16–29, Springer, 2004.
[7] S. Mangard, E. Oswald, and T. Popp, Power analysis attacks: Revealing the secrets of smart cards, Springer-Verlag US, 2007.
[8] S. Banik, A. Bogdanov, T. Isobe, and M. Jepsen, "Analysis of software countermeasures for whitebox encryption," IACR Trans. Symmetric Cryptology, vol.2017, no.1, pp.307–328, 2017.
[9] T.S. Messerges, "Using second-order power analysis to attack DPA resistant software," Second International Workshop on Cryptographic Hardware and Embedded Systems, CHES 2000, Lecture Notes in Computer Science, vol.1965, pp.238–251, Springer, Berlin, Heidelberg, 2000.
[10] M. Joye, P. Paillier, and B. Schoenmakers, "On second-order differential power analysis," 7th International Workshop on Cryptographic Hardware and Embedded Systems, CHES 2005, Lecture Notes in Computer Science, vol.3659, pp.293–308, Springer, Berlin, Heidelberg, 2005.
[11] SCARF homepage, http://www.k-scarf.or.kr/