Key Recovery Against 3DES in CPU Smart Card Based on Improved Correlation Power Analysis

Yanting Ren, Liji Wu*, Hexin Li, Xiangyu Li, Xiangmin Zhang, An Wang, and Hongyi Chen

Abstract: The security of CPU smart cards, which are widely used throughout China, is currently being threatened by side-channel analysis. Typical countermeasures to side-channel analysis involve adding noise and filtering the power consumption signal. In this paper, we integrate appropriate preprocessing methods with an improved attack strategy to generate a key recovery solution to the shortcomings of these countermeasures. Our proposed attack strategy improves the attack result by combining information leaked from two adjacent clock cycles. Using our laboratory-based power analysis system, we verified the proposed key recovery solution by performing a successful correlation power analysis on a Triple Data Encryption Standard (3DES) hardware module in a real-life 32-bit CPU smart card. All 112 key bits of the 3DES were recovered with about 80 000 power traces.

Key words: Triple Data Encryption Standard (3DES); CPU smart cards; power analysis; key recovery; side-channel analysis

1 Introduction

Smart cards are among the most widely used tamper-resistant hardware options and can provide a variety of security mechanisms because of their embedded CPUs. The CPU processing power makes it possible to implement cryptographic algorithms in the card. However, the security of smart cards is threatened by Side-Channel Analysis (SCA).

SCA is a passive and non-invasive attack that exploits side-channel information (e.g., power consumption, electro-magnetic radiation) leaked by a cryptographic device to recover secret data (e.g., the key). Compared with invasive and semi-invasive attacks, SCA is more efficient and powerful, and is less dependent on the attackers’ equipment and knowledge of the target devices. Since Kocher et al. [1] first introduced Differential Power Analysis (DPA) in 1999, SCA has received much attention. Theoretical and experimental research results have shown that mathematical security is not sufficient to ensure the security of real-world cryptographic devices. Unfortunately, the designers of cryptographic devices often underestimate the risks of SCA. For economic and technical reasons, simple countermeasures such as adding noise and filtering the power consumption are currently the most common approaches taken [2]. In recent years, many commercial cryptographic devices have been reported as being vulnerable to SCA [3–5].

The Data Encryption Standard (DES) is a symmetric cryptographic algorithm proposed in 1970s. Today, its key length (56 bits) is too short to meet security requirements. Therefore, the 3DES, which has a longer key length (112 bits or 168 bits), has taken the place...
of DES and become one of the most widely used block ciphers in smart cards.

The additive Gaussian noise model is the most commonly used noise model in power signal processing. Many methods have been proposed to increase its Signal-to-Noise Ratio (SNR), including a fourth-order cumulant\cite{6}, power (second-order moment), and integration\cite{2}. In Ref. [6], taking advantage of the fact that the high-order statistics of Gaussian noise is zero, the authors introduced a fourth-order cumulant to eliminate noise. They also compared the performance of the three methods mentioned above and concluded that the fourth-order cumulant achieved the best result. However, this method has a limitation in that it is only effective when the signal has high kurtosis. If the power consumption signal is filtered, the kurtosis of the power signal is rather small, and this method is rendered ineffective.

In this paper, we focus on recovering the 3DES keys in smart cards following SCA attacks, and propose a key recovery solution that integrates appropriate preprocessing methods with an improved attack strategy. Using a power analysis system in our laboratory, we mount successful attacks on the 3DES in a real-life 32-bit CPU smart card. The experimental results show that our key recovery solution is more efficient than both the traditional Correlation Power Analysis (CPA) and CPA using other high-order preprocessing methods. In addition, we discuss our improved attack strategy. Our work reveals that smart cards equipped only with simple countermeasures such as increasing the noise and filtering the power consumption are vulnerable to black-box SCA.

The remainder of this paper is arranged as follows. In Section 2, we introduce the measurement setup and the target smart card. In Section 3, we describe in detail the proposed key recovery solution. In Section 4, we recover a 3DES key and compare the experimental results of our solution with those of traditional CPA as well as CPA using other high-order preprocessing methods. We discuss the proposed strategy in Section 5, and make our conclusion in Section 6.

2 Measurement Setup and Object of Attack

2.1 Power analysis system

We designed and implemented a power analysis system in our lab. As shown in Fig. 1, we used a PC as the control center, a programmable card reader as the device environment, and a commercial mixed-signal oscilloscope (Agilent Infiniium 54830) to measure the power consumption of the target smart card.

The programmable card reader, developed on an FPGA board, can be configured as a reader, a card, or an eavesdropper between the card and reader. In our case, we used it as an ISO7816 compliant card reader. The reader communicates with the PC and passes commands and data (e.g., plaintexts) to the target card. It also generates a trigger signal for the oscilloscope to begin a measurement. To acquire power consumption information, we put a small resistor in the Vcc path. With a differential probe and a sampling rate of \(2 \times 10^8\) s\(^{-1}\), the oscilloscope measures the voltage difference of the resistance, which indicates the power consumption of the smart card. The acquired power traces were then sent to the PC to be analyzed.

2.2 Object of attack

The target card is a dual-interface smart card, which can be used as a bank card, an access card, or a student ID card. It integrates a 32-bit CPU and a hardware-implemented 3DES engine. Figure 2 shows a block diagram of the smart card. Information provided by the card manufacturer indicates that the target smart card is equipped with countermeasures against SCA, such as increasing the noise and filtering the power consumption signal.

3 Key Recovery Solution

We propose a key recovery solution that integrates preprocessing methods with an improved attack strategy and which can reveal the 3DES keys of a smart card in a black-box scenario in which the standard countermeasures have been used, as described above. The key recovery solution can be summarized in Fig. 3.

3.1 Preprocessing of raw power traces

Rationale for preprocessing A misalignment of
power traces and noise reduces the SNR, and thereby increases the difficulty of making an attack. In this section, we focus on the preprocessing methods used.

**Preprocessing: alignment** An unstable clock, a random delay, or inaccurate triggering in measurement[2] usually causes misalignment of the power traces. Many alignment methods have been proposed, including pattern matching[2], sliding window DPA[7], wavelet transformation[8], elastic alignment[9], and Differential Frequency Analysis (DFA)[10].

Of these methods, pattern matching is an effective option because of its low computational complexity. In our case, the most critical issue in pattern matching is to find a suitable pattern, namely, a template for aligning the power traces, which should have distinctive features and be as close as possible to the 3DES encryption.

As shown in Fig. 4, we locate the bus transfers of plaintext and ciphertext by calculating correlation coefficients of the plaintext/ciphertext bytes’ Hamming Weights and power traces from 4000 power traces. The results also indicate that there is an unprotected 8-bit data bus in the smart card.

A zoomed-in view of the intervals of the 3DES encryption is shown in Fig. 5, and we can see that...
there are six equidistant peaks after the encryption. We used these peaks as our template to align the power traces because they are close to the intervals of 3DES encryption, and are very distinct from other peaks in the trace.

Preprocessing: noise reduction The authors in Ref. [6] used a fourth-order cumulant to reduce the noise, as shown in Eq. (1):

$$\kappa_4 = E[(x - E(x))^4] - 3E[(x - E(x))^2]^2$$  \hspace{1cm} (1)

However, as shown in Fig. 5, the peaks in the power trace are not as sharp as they should be because of power filtering. Thus, the fourth-order cumulant is not suitable for use in this case.

We introduce a fourth-order moment, Eq. (2), to deal with this countermeasure. This method retains the merits of high-order statistics in reducing the Gaussian noise, and has no requirement regarding the kurtosis of the signal to be processed. As shown in Fig. 6, in this case, our method achieves a much better result than that of the fourth-order cumulant.

$$M_4 = E[(x - E(x))^4]$$  \hspace{1cm} (2)

We also considered other high-order statistics, including a third-order cumulant and a third-order moment. While the third-order cumulant can also reduce the Gaussian noise, it is not suitable for a symmetrically distributed signal. The experimental results presented in Section 4 show that a fourth-order moment works best in our case.

We note that these methods also help to correct the misalignment of the power traces, since they use a sliding window.

3.2 Improved attack strategy

Correlation Power Analysis CPA\(^{[11]}\) is one of the most widely used SCA attack methods. Based on the premise that the side-channel information is data dependent, CPA uses Pearson’s correlation coefficient to determine the linear relationship between the side-channel information and the intermediate value corresponding to a guessed key. Based on our measurement setup, we implemented a CPA attack as follows.

We recorded the power traces while the target device was repeatedly executing 3DES encryption. Our power analysis system guarantees the transparency of the communication between the card and reader, so that known plaintext attacks are made available. For every possible key, we can calculate hypothetical intermediate values of all the plaintexts. Generally, the intermediate value should depend on both the plaintext and the key, and be sensitive to any change in them. In the 3DES algorithm, the value of \(R_0\) (see Fig. 1 of Ref. [12]) is a reasonable choice.

To calculate the correlation, the hypothetical intermediate values are mapped to the hypothetical power consumptions based on a given power model, and are then compared to the actual power consumption. In a black-box attack, the implementation details of the target cryptographic algorithm are not usually available to the attackers, so typically, trial-and-error is used to choose between power models. In 3DES encryption, the outputs of each round are stored in two registers (see the DES algorithm in Ref. [12]), and the Hamming-Distance (HD) model is commonly used to simulate the register power consumption\(^{[2]}\). As such, here we chose HD as our power model.

**Improved Attack Strategy** To improve the results of the CPA, we propose an attack strategy based on the DES algorithm (cf. Ref. [12]) and its features.

In the DES algorithm, the value of \(R\) is always passed to \(L\) in the next round. Considering the hardware implementation, this means that the \(L\) register always flips in the same way as the \(R\) register does in the previous round. Thus, similar side-channel leakage can appear in two adjacent rounds. For example, at the beginning of the second round, the value of the \(R\) register changes from \(R_0\) to \(R_1\), and the Hamming distance of the \(R\) register is \(R_0 \oplus R_1\). At the beginning of the third round, the value of the \(L\) register changes from \(L_1 = R_0\) to \(L_2 = R_1\), and the Hamming distance of \(L\) equals \(R_0 \oplus R_1\).

Sometimes, the power consumption of the \(L\) register is not considered, perhaps because the \(R\) register and...
its subsequent logic circuit consume more power than the \( L \) register and its subsequent logic circuit. This is due to the greater complexity of the logic circuit connected to the \( R \) register, for example, the “\( f \)” function. However, if the power consumption of the \( L \) register and its subsequent logic circuit is not neglected, it can be used to enhance the result of the attack. Our attack strategy improves the attack result by combining the side-channel information of adjacent rounds, and comparing the hypothetical power consumption with the summed power consumption of two adjacent clock cycles rather than that of just one cycle.

**Applications to Feistel Structure** Our improved attack strategy can be applied to all standard Feistel ciphers, such as Blowfish and Twofish. Let \( F \) be the round function, and \( K_i \) be the sub-key for round \( i \). The operation of a Feistel cipher in each round \( i = 0, 1, \ldots, n \) computes \( L \) and \( R \) as follows:

\[
\begin{align*}
L_{i+1} &= R_i, \\
R_{i+1} &= L_i \oplus F(R_i, K_i)
\end{align*}
\]  

(3)

From Eq. (3), we can deduce that \( R_i \oplus R_{i-1} = L_{i+1} \oplus L_i \). So, the Hamming distance of the \( R \) register in round \( i \) equals that of the \( L \) register in round \( i + 1 \). This improved attack strategy is effective for Feistel ciphers, and can be extended to generalized Feistel structures.

### 4 Experimental Results

In this section, we compare the experimental results of our proposed strategy with those of traditional CPA\cite{11} and CPAs with different high-order preprocessing methods. We conducted all attacks under the same conditions—in the same experimental environments, on the same smart card, and with the same number of power traces. We used 80,000 aligned power traces to mount these attacks. In our case, each raw trace contained 10,000 sample points, and it took about 1 second to acquire a trace. Therefore, the total time of the online stage was about 22 hours.

#### 4.1 Traditional CPA

Since the plaintexts are known, we applied our first attack on the S-Box output of the first round of the first DES. Corresponding to eight S-Boxes, we divided the 32-bit intermediate variable \( R_0 \oplus R_1 \) into eight parts, with each part being the output of an S-Box. We carried out attacks on each part separately. Taking S-Box 8 as an example, we illustrate the workflow of the CPA attack in Algorithm 1.

**Algorithm 1 Workflow of CPA Attack on S-Box 8**

| Input: | Preprocessed power traces \( \{T^1, \ldots, T^{n_r}\} \) and corresponding plaintexts \( \{P^1, \ldots, P^{n_r}\} \). |
| Output: | The guessed key for S-Box 8: \( K_8 \). |
| 1: | for (eachGuessedKey \( \in \text{GF}(2^6) \)) |
| 2: | for \((0 < i \leq n_t)\) |
| 3: | \( x(i) = \text{HD}(F(S_8(\text{GuessedKey} \oplus P^i), P^i)) \) |
| 4: | end for |
| 5: | for \((0 < t \leq l)\) |
| 6: | \( y_t = \{T^{i_1}_t, \ldots, T^{i_n}_t\} \) |
| 7: | Correlation\_GuessedKey\( (t) = p(x, y_t) \) |
| 8: | end for |
| 9: | end for |
| 10: | \( K_8 = \text{SearchPeak} (\{\text{Correlation}\_\text{GuessedKey}\}) \) |
| 11: | \( \text{GuessedKey} \in \text{GF}(2^6) \) |

**Notation** To ensure a clear understanding, the notations used in Algorithm 1 are detailed as follows.

We used the letters \( T \) and \( P \), respectively, for the power trace and plaintext. The number of power traces used in the CPA attack is denoted as \( n_t \), and the corresponding plaintexts for S-Box 8 are written as \( \{P^1, \ldots, P^{n_r}\} \). Each power trace is composed of \( l \) sample points. Hence, the \( i \)-th power trace is denoted as \( T^i = \{T^{i_1}_1, T^{i_2}_2, \ldots, T^{i_n}_n\} \). Furthermore, we used \( K_8 \) to denote the six key bits corresponding to S-Box 8, and \( F_8 \) to denote the round function corresponding to S-Box 8.

In Steps 1–7 of Algorithm 1, we can get 64 correlation coefficient traces corresponding to 64 possible values of \( K_8 \). In Step 8, we search these correlation traces to determine whether a peak that is higher than the noise level exists\cite{2}. If the CPA attack is successful, the highest peak will indicate the correct guess of \( K_8 \).

We conducted a traditional CPA attack and obtained 64 correlation traces for each S-Box, as shown in Fig. 7. The correlation coefficient peaks of the correct key bits of five S-boxes (S-Boxes 2, 3, 6, 7, and 8) exceeded the interval \( \pm4/\sqrt{n_t}[2] \) at the correct time (beginning of the second round), where \( n_t \) is the number of power traces used in the attack. For S-Boxes 4 and 5, there were no obvious peaks. For S-Box 1, a small peak appeared at a wrong time (beginning of the third round).

#### 4.2 CPA with preprocessing methods

We introduced several kinds of high-order preprocessing methods in Section 3.1. Here we conduct CPA attacks using different preprocessing methods,
including a third-order cumulant, a third-order moment, a fourth-order cumulant, and a fourth-order moment. We used the same number (80,000) of power traces as in the CPA attack, and the results are presented in Figs. 8a–8d. As shown in Figs. 8b–8d, the best result is achieved after noise reduction with the fourth-order moment. In Fig. 8d, peaks have become higher for all the S-Boxes. Moreover, for S-Boxes 4 and 5, the peaks of the correct key exceed the interval $\pm 4/\sqrt{n}$.

For S-Box 1, peaks arise in both the second and third rounds, but only the latter exceeds $\pm 4/\sqrt{n}$.

With 80,000 traces, we recovered the key bits of at most seven S-Boxes using the conventional CPA. Using the fourth-order moment method to preprocess the power traces can improve the attack results.

### 4.3 Proposed solution

By integrating the preprocessing methods of the fourth-order moment (cf. Section 3.1) and the improved attack strategy (cf. Section 3.2), we conducted an improved attack on the target card. The workflow of our attack is presented in Algorithm 2. The differences between Algorithms 1 and 2 are in the preprocessing (Step 1) and Combine (Step 2) steps. The Combine step is detailed in Algorithm 3. Let $d$ be the number of sample points in one round. We shift a preprocessed power trace $T^i$ by $d$ points, and add the shifted one to the original. Hence, the information of two adjacent rounds is combined, yielding a combined power trace $TC^i$. Using the same number of power traces as in the traditional CPA attack, we achieved much better results, as shown in Fig. 9. Compared with Fig. 8d, the peaks of the right key bits of S-Boxes 1, 4, and 5 are higher and exceed $\pm 4/\sqrt{n}$, as expected.

We applied our solution on the second round of the first DES in the same way, and then on the second DES. The results are shown in Figs. 10–12.

### 5 Discussion of the Improved Attack Strategy

The effects of the attack strategy on different S-Boxes were not the same. The peaks of S-Boxes 1, 4, and
Fig. 8a Correlation coefficients (CPA with third-order cumulant).

Fig. 8b Correlation coefficients (CPA with third-order moment).

Fig. 8c Correlation coefficients (CPA with fourth-order cumulant).

Fig. 8d Correlation coefficients (CPA with fourth-order moment).
5 rise, while those of S-Boxes 3, 6, and 8 fall. Thus, the strategy conditions must be discussed.

The power consumption of the \( i \)-th round \( C_i \) can be written as in Eq. (4):

\[
C_i = CR_i + CL_i + CN_i = CR_i + \alpha CR_{i-1} + CN_i \quad (4)
\]

where CR is the power consumption of the \( R \) register, CL is the power consumed by the \( L \) register, and CN is the noise. Since the \( L \) register always flips in the same
way as did $R$ in the previous round, $CL_i$ is proportional to $CR_{i-1}$ with a scale factor $\alpha$.

CR can be divided into two parts: the power consumption of the four target bits $CR_{\text{tar}}$, and that of the other bits $CR_{\text{other}}$:

$$CR = CR_{\text{tar}} + CR_{\text{other}}$$  \hspace{1cm} (5)

If we let $H$ be the hypothetical power consumption, the correlation coefficient $\rho(C_i, H)$ can be written as

$$\rho(C_i, H) = \frac{\text{cov}(C_i, H)}{\sqrt{\text{var}(C_i) \cdot \text{var}(H)}} = \frac{\text{cov}((CR_{\text{tar}} + \alpha CR_{i-1} + CN_i), H)}{\sqrt{\text{var}(C_i) \cdot \text{var}(H)}}$$  \hspace{1cm} (6)

In Eq. (6), $H$ is only relative to $CR_{\text{tar}}$, so $\text{cov}((CR_{\text{other}} + \alpha CR_{i-1} + CN_i), H) = 0$.

Combining two adjacent rounds, we have:

$$C_{i}\text{comb} = (1 + \alpha)CR_{\text{tar}} + (1 + \alpha)CR_{\text{other}} + \alpha CR_{i-1} + CR_{i+1} + CN_{i+1} + CN_i$$  \hspace{1cm} (7)

$$\rho(C_{i}\text{comb}, H) = \rho(C_i, H) \cdot \sqrt{1 - \frac{\text{var}(CR_{\text{other}}) + \alpha^2 \text{var}(CR_{i-1}) + \text{var}(CN_i)}{\text{var}(CR_{\text{tar}})}}$$

$$\sqrt{1 + \frac{\text{var}(CR_{\text{other}}) + \alpha^2 \text{var}(CR_{i+1}) + 2 \text{var}(CN_i)}{\text{var}(CR_{\text{tar}})}}$$  \hspace{1cm} (8)

By combining these rounds, we increase the SNR, and Eq. (8) clarifies the effect of this combination on the noise: After the combination, the effect of $CR_{\text{other}}$ remains the same, and the effects of $CN_i$ and $\alpha CR_{i-1}$ (power consumption of the $L$ register) reduce by the factor $(1 + \alpha)^2$. However, this also brings new noise terms as follows: $CR_{i+1}$ (power consumption of the $R$ register in the next round) and $CN_{i+1}$ (noise in the next round). To compare $\rho(C_i, H)$ and $\rho(C_{i}\text{comb}, H)$, we made the following hypothesis to simplify Eq. (8):

Hypothesis: $\text{var}(CR_{i-1}) = \text{var}(CR_{i+1})$.

With this hypothesis, Eq. (8) can be deduced as follows:

$$\rho(C_{i}\text{comb}, H) > \rho(C_i, H) \Rightarrow \alpha^2 \text{var}(CR_{i-1}) + \text{var}(CN_i) > \frac{(1 + \alpha^2) \text{var}(CR_{i+1}) + \text{var}(CN_i)}{(1 + \alpha)^2}$$

$$\Rightarrow \frac{2}{(1 + \alpha)^2} \text{var}(CN_i) > (1 + \alpha^2) \text{var}(CN_i)$$  \hspace{1cm} (9)

For $\alpha > \sqrt{2} - 1$, the noise term on the right side of Formula (9) will be greater than that on the left. Then, this inequality can be further simplified:

$$\rho(C_{i}\text{comb}, HD) > \rho(C_i, HD) \Rightarrow \alpha^2 \text{var}(CR_{i-1}) > \frac{(1 + \alpha^2) \text{var}(CR_{i-1})}{(1 + \alpha)^2}$$

$$\Rightarrow \alpha^4 + 2 \alpha^3 - 1 > 0$$  \hspace{1cm} (10)

For $\alpha > 0$, the solution set is $\alpha > 0.7167$, meaning when $\alpha > 0.7167$, combining two adjacent rounds will enhance the CPA result. The results show that while the information leaked by the $L$ register is comparable to that of the $R$ register, the proposed combination strategy is worth trying.

As shown in Figs. 8d and 9, for $S$-Boxes 1, 4, and 5, the correlation coefficient peaks of the second and third rounds are similar, which means they have a large $\alpha$, so the peaks of the correct key bits become higher after the combination.

6 Conclusion

In this work, we presented a key recovery solution that integrates preprocessing methods with an improved attack strategy. We then conducted black-box attacks on the 3DES in a real-life 32-bit CPU smart card. In the attack, we recovered 112 3DES key bits with 80 000 power traces.

Our results show that if proper preprocessing methods are applied, it is feasible for an attacker to obtain useful information from cryptographic devices that are equipped with no specialized countermeasures in a “black-box” scenario. The unprotected data bus leaks information about when the plaintext and ciphertext are loaded. Hence, the encryption operation can be quickly pinpointed. The power traces of the smart card are not adequately uniform, and as such, a proper “pattern” can be used to align them. Designers should avoid such obvious patterns in encryption/decryption operations. Adding noise and filtering the power consumption are insufficient for hiding security information, because various noise reduction techniques can efficiently increase the SNR.

Based on features of the Feistel cipher, we proposed an improved attack strategy that takes advantage of the information leaked in two adjacent rounds. In the 3DES algorithm, $L$ always changes in the same way as $R$ did in the previous round. Therefore, the power consumption of the $L$ register can be used to improve the attack result. This strategy can increase the SNR and enhance the power analysis result.

The proposed attack strategy does not rely on hardware implementation, as it is based on the structure
of a Feistel cipher. It can be used effectively when the intermediate results, $L$ and $R$, are kept in two different registers and are operated separately and simultaneously, which is natural and reasonable in the hardware implementation of a Feistel cipher. Therefore, the proposed method has good universality. To avoid leakage of the $L$ register and resist the improved attack strategy, further countermeasures must be adopted.

Adding noise and filtering the power consumption are not sufficient to resist SCA. In order to improve the security level of secret information stored in smart cards, designers must use multiple countermeasures to defend these cards against SCA.

Acknowledgment

This work was supported by the Major Program “Core of Electronic Devices, High-End General Chips, and Basis of Software Products” of the Ministry of Industry and Information Technology of China (No. 2014ZX01032205), and the Key Technologies Research and Development Program of the Twelfth Five-Year Plan of China (No. MMJJ201401009).

References

[1] P. Kocher, J. Jaffe, and B. Jun, Differential power analysis, in Proceedings of the 19th Annual International Cryptology Conference on Advances in Cryptology (CRYPTO), Santa Barbara, CA, USA, 1999, pp. 388–397.

[2] S. Mangard, E. Oswald, and T. Popp, Power Analysis Attacks: Revealing the Secrets of Smart Cards, Boston, MA, USA: Springer, 2007.

[3] A. Moradi, M. Kasper, and C. Paar, Black-box side-channel attacks highlight the importance of countermeasures—An analysis of the Xilinx Virtex-4 and Virtex-5 bitstream encryption mechanism, Lecture Notes in Computer Science, vol. 7178, pp. 1–18, 2012.

[4] T. Eisenbarth, T. Kasper, A. Moradi, C. Paar, M. Salmasizadeh, and M. M. Shalmani, On the power of power analysis in the real world: A complete break of the KEELOQ code hopping scheme, in Proceedings of the 28th Annual International Cryptology Conference on Advances in Cryptology (CRYPTO), Santa Barbara, CA, USA, 2008, pp. 203–220.

[5] D. Oswald and C. Paar, Breaking Mifare DESFire MF3ICD40: Power analysis and templates in the real world, Lecture Notes in Computer Science, vol. 6917, pp. 207–222, 2011.

[6] T. H. Le, J. Clediere, C. Serviere, and J. L. Lacoume, Noise reduction in side channel attack using fourth-order cumulant, IEEE Transactions on Information Forensics & Security (TIFS), vol. 2, no. 4, pp. 710–720, 2007.

[7] C. Clavier, J. S. Coron, and N. Dabbous, Differential power analysis in the presence of hardware countermeasures, Lecture Notes in Computer Science, vol. 1965, pp. 252–263, 2000.

[8] X. Charvet and H. Pelletier, Improving the DPA attack using wavelet transform, in NIST Physical Security Testing Workshop, Honolulu, HI, USA, 2005.

[9] J. J. Woudenberg, M. Witteman, and B. Bakker, Improving differential power analysis by elastic alignment, Lecture Notes in Computer Science, vol. 6558, pp. 104–119, 2011.

[10] T. Plos, M. Hutter, and M. Feldhofer, Evaluation of side-channel preprocessing techniques on cryptographic-enabled HF and UHF RFID tag prototypes, in Workshop on RFID Security, Budapest, Hungary, 2008, pp. 114–127.

[11] E. Brier, C. Clavier, and F. Olivier, Correlation power analysis with a leakage model, Lecture Notes in Computer Science, vol. 219, pp. 114–127, 2008.

[12] U.S. Department of Commerce and NIST, Fips pub 46-3: data encryption standard (DES), 1999.
Xiangmin Zhang received the BS degree from Peking University in 1988, and MEng degree in electronic engineering from Tsinghua University in 1991. Now, he works in the Institute of Microelectronics, Tsinghua University. His main research interests are information security and automotive electronics.

An Wang currently works as a lecturer in Beijing Institute of Technology. He received the PhD degree from Shandong University in 2011. His main research interests include side-channel attack, embedded system, and fast implementation in cryptography. He is also the webmaster of a cryptographic website (http://www.mathmagic.cn).

Hongyi Chen is a professor of Department of Micro-nanoelectronics in Tsinghua University, a senior member of Chinese Institute of Electronics (CIE), a senior member of Institute of Electric and Electronic Engineers (IEEE). He got the BEng and MEng degrees from Tsinghua University in 1965 and 1981, respectively. His research interests include ASIC and SOC (system on a chip) design methodology, algorithm mapping to hardware architecture and realization, VLSI-DSP and applications in multimedia signal (e.g., speech, audio, image and video) processing, and information security fields, etc. He has published more than 300 academic articles, co-published two books and co-translated two textbook, claimed 21 items of US innovation patents, among which 13 items have been awarded.