Analysis of Operation Errors for Fault Injection Attack

Masaya Yoshikawa and Hikaru Goto
Dept. of information engineering Meijo University, Nagoya, Japan
Email: dpa_cpa@uahoo.co.jp;

***Abstract*—Although the encryption standards are theoretically safe, it has been recently reported that confidential information could be illegally revealed when the encryption standards are embedded in the electronic devices. In particular, the menace posed by fault injection attacks has become extremely serious. To guarantee the safety of electronic devices in the future, into which cryptographic circuits have been incorporated, fault injection attacks must be thoroughly studied. This study elucidates the tendency of fault injection.**

***Index Terms*—fault injection attack, tendency of an operation error, glitch generation, tamper resistance, advanced encryption standard**

I. Introduction

Cryptographic circuits are used to protect confidential information, and encryption standards used in cryptographic circuits have been sufficiently confirmed that their decryption is computationally impossible. The advanced encryption standard (AES) is the most popular encryption standard and used in all over the world. However, it was recently reported that when a theoretically safe encryption algorithm was embedded in the hardware, confidential information could be illegally revealed by side-channel attacks [1]-[12]. The side-channel attacks are classified into two categories. One is power analysis attack [7]-[12]. And the other is fault injection attack [1]-[6]. Power analysis attack utilizes dynamic power consumption. It occurs when circuit is operating. Static power consumption which is leak power consumption is noise in this attack. Therefore, the power analysis attack is hard to apply in deep-sub micron technologies, because the dynamic power consumption is reduced in comparison the static power consumption.

By contrast, fault injection attack is one of the most dreadful attacks in side-channel attacks. In fault injection attacks, secret keys are revealed using a fault, which has been intentionally generated during the operation of a cryptographic circuit, and using a pair of cipher text containing data errors (cipher text with the fault) and a correct cipher text.

A fault can be generated by the following three methods [3], [4]: (1) using laser irradiation, (2) lowering the power supply voltage, and (3) inserting a glitch in a clock. The method of using laser irradiation is ineffective since it needs circuit information in LSI and a laser irradiation apparatus is expensive. The method of lowering the power supply voltage induces an abnormal circuit operation by applying the voltage which is smaller than the reference voltage. However, this method may destroy the circuit since it manipulates the power supply voltage. By contrast, the glitch is easy to generate [4]. A glitch is an abnormal pulse with an extremely short period. Generally, the round processing of the AES is performed in every clock. When a glitch is mixed in the AES during encryption processing and recognized as a clock, the round processing is initiated by the glitch. However, since the period of a glitch is very short, the computational time necessary for the round processing cannot be secured. By this, the setup time constraint of a flip-flop is violated. Subsequently, an operation error (fault) may occur.

In order to guarantee the safety of electronic devices [5] in the future, into which cryptographic circuits have been incorporated, fault injection attacks must be thoroughly studied. Especially, it is important to examine the characteristics of circuit in case of inserting a glitch. This study elucidates the tendency of an operation error due to a glitch using the AES. The experiments using FPGA clarify the tendency of operation errors by the glitch.

II. Preliminaries

AES consists of 128-bit block ciphers, in which a round is composed of SubBytes, ShiftRows, MixColumns, and AddRoundKey processes, and data are transformed by repeating the round processing for multiple times. The number of rounds is determined according to the key length. The present study adopts the key length of 128 bits, which is often used.

In this case, the number of rounds is 10. MixColumns is omitted only at round 10, the final round. Key values used at each round are repeatedly calculated using the process called KeySchedule to use for the round processing. SubBytes is used for numeric transformation in the form of a byte unit.

When the input byte value is assumed to be x, SubBytes can be expressed in the form of function S(x). ShiftRows is used for the shift of a byte location. In the case where the value of byte location, *i*, is shifted to *j*, this shift can be expressed using function ShiftRow(x) and Formula (1):

Manuscript received March 2, 2014; revised May 6, 2014.

doi: 10.12720/ijsps.2.1.74-77

$$j = \text{shiftrow}(i) \quad (1)$$

Since the process of round 10 is independent every byte, the value of the jth byte of R10(I, K10), which is the process of round 10 and in which round key K10 and cryptographic intermediate value I are to be input, can be defined using Formula (2):

$$R10(I, K10)[j] = S(I[i])\text{^}K10[j] \quad (2)$$

where ^ represents XOR operation and X[i] represents the value of byte location i, of X. When the key scheduling process is expressed by function Ks(x), the key value of round 9 (K9) and that of round 10 (K10) can be expressed using Formula (3):

$$K10 = Ks(K9) \quad (3)$$

In the key scheduling process, RotWord, SubWord, and XOR are used, and these three operations are performed one by one. When a 32-bit value, which is output from SubWord at round 10, is assumed to be w, Formula (4) can be obtained:

$$w[i] = S(K9[j]) \quad (4)$$

When the round constant of round 10 is assumed to be Rcon10, Formula (5) can be obtained:

$$Ks(K9)[i] = Rcon10[i]\text{^}w[i]\text{^}K9[i] \quad (5)$$

III. Proposed Method

In a fault generation method using a glitch, a tendency was predicted to be shown in the fault occurrence point due to signal propagation delay. In the AES, many processes were performed in every state. The processing time of each state differed from each other due to signal propagation delay. In addition, the delay time differed from one state to the other in bit unit due to wiring delay and the difference in logical step. To elucidate the difference in the delay time in bit unit, a simulation of delay in an AES encryption circuit was performed. Fig. 1 shows the simulation results.

In Fig. 1, CLK denotes operation clock, [115]-[127] denotes a timing chart from the 116th bit to the 128th bit in the 128-bit cryptographic intermediate value, and the dotted line denotes the time when the bit-wise theoretical value is determined (stabilized).

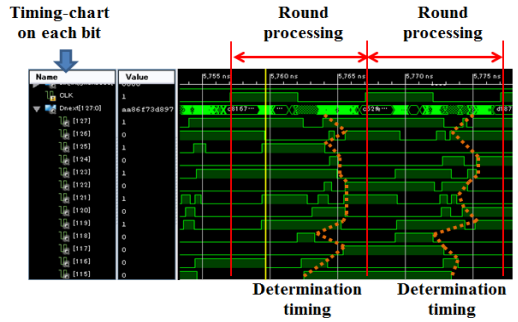


Figure 1. Example of a logic simulation of delay time in an AES encryption

This figure reveals that the time the theoretical value is determined differs according to the bit. Since the round processing ends up with the AddRoundKey process, the time when the theoretical value is determined can be reworded as the time when the AddRoundKey process is completed.

In the case where all bit values are simultaneously determined, if the setup time constraint is violated due to a glitch before the values are determined, a fault due to the operation error will occur in all the bits. In contrast, if the setup time constraint is not violated, a fault will not occur in all the bits. In the case where the time when the theoretical value is determined differs according to the bit, when a fault due to a glitch occurs during the AddRoundKey process, a bit in which a fault occurs and a bit in which a fault does not occur are predicted to coexist. Therefore, the number of fault bits is biased to be small and a fault does not occur in some states, as shown in Fig. 2.

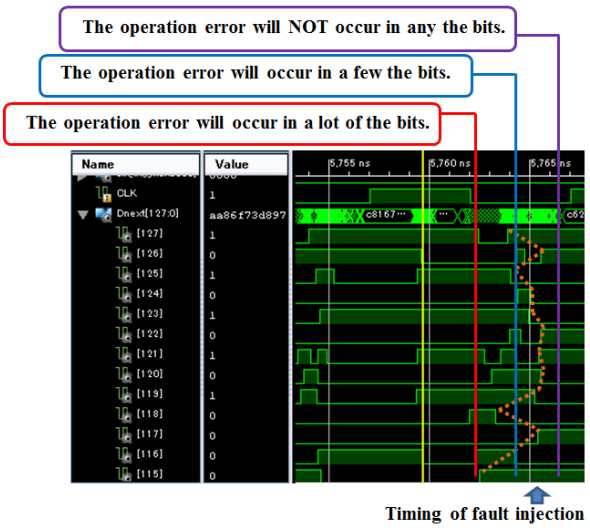


Figure 2. Example of fault generation timing in an AES encryption

The fault generation method using a glitch cannot specify the fault occurrence point. Moreover, the analytical efficiency of the fault injection attacks is low. If the processing time is terminated before the MixColumns process due to a glitch, the AddRoundKey process cannot be performed. Consequently, a fault occurs in almost all states and the fault occurrence point cannot be specified. Therefore, when a fault due to a glitch occurs before the MixColumns process, the efficiency of fault injection attacks decreases.

However, when a fault due to a glitch occurs during the AddRoundKey process, the number of fault bits trend to be small and a fault can be predicted to not occur in some states. When a fault due to a glitch is mixed in the AES after round nine, the MixColumns process cannot be applied to later processes and the fault cannot be diffused to other states.

Hence, in an output cipher containing a fault, the number of states where a fault is mixed decreases. Consequently, only the point of the fault that has occurred during the AddRoundKey process can be specified from a cipher containing a fault, as shown in Fig. 3.

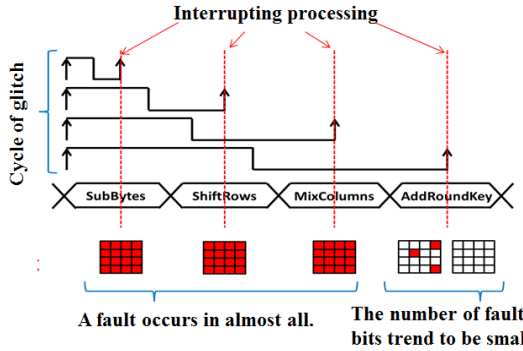


Figure 3. Relationship between interrupting points and the number of faults

Moreover, a characteristic where the number of fault bits is small can be obtained. Using this characteristic, the fault generation method using a glitch will be able to achieve a high analytical efficiency of fault injection attacks.

IV. Experiments and Discussions

A. Glitch Generation Experiment

To generate a glitch, two clocks of a base clock and a clock, which is obtained by shifting the phase of the base clock (phase-shift clock), are prepared, as shown in Fig. 4. In this experiment, DSO1024A (Agilent Technologies) is used as an oscilloscope, and N2863A 10.1(Agilent Technologies) is used as a probe.

By changing from the base clock to the phase-shift clock at a specific timing, a glitch is artificially generated. Fig. 5(1) shows the results obtained when a glitch is generated. Fig. 5(2) shows the results obtained when a glitch is not generated. As shown in Fig. 5(2), when a glitch could not be generated, a glitch could not be recognized as a clock. The reason for this is that since the timing of changing from the base clock to the phase-shift clock was too early, the value of a glitch could not exceed the threshold value of a fall.

B. Evaluation of the Characteristic of an Operation Error

To examine the tendency of a fault in the AES, an experiment was performed using an FPGA. Fig. 6 shows the experimental environment.

The experimental procedure was as follows:

- A plain text was arbitrarily created in a PC.
- The created plain text was encoded onto an FPGA board. Simultaneously, the created plain text was enciphered in a PC to obtain a normal cipher.
- The plain text encoded on the board was enciphered using the AES, which had been incorporated into a cryptographic circuit. At this moment, an operation error was generated using a glitch supplied from a control circuit to obtain a cipher containing a fault.
- The cipher containing a fault was received from the board.
- The tendency of the number of fault bits in each state was analyzed.

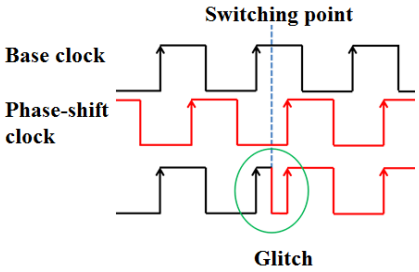
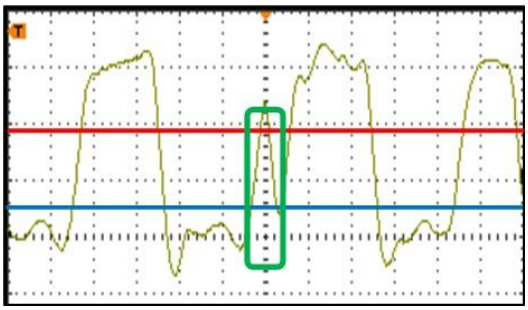
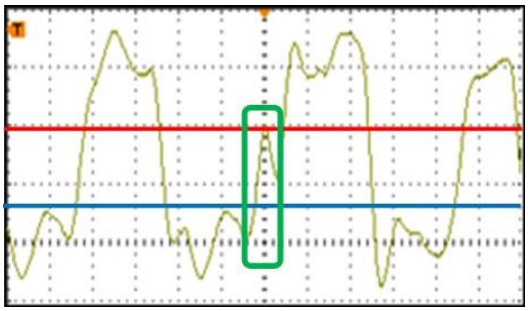


Figure 4. Fault generation procedure



(1) Example of success for a glitch generation



(2) Example of failure for a glitch generation

Figure 5. Generated fault

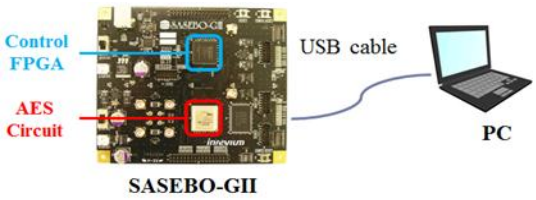


Figure 6. Experimental environment

Fig. 7 shows the details of the faults that have occurred using 100,000 cipher pairs. In this figure, 16 (4 x 4) graphs corresponded to the 16 states of the cryptographic intermediate value when round nine was completed. The vertical axis of each graph represents the number of faults and the horizontal axis represents the number of fault bits (1-8). For the number of fault bits to be easily observed, the memories of the vertical axes were not uniformed.

As shown in Fig. 7, the number of fault bits is small in any state. In particular, the number of fault bits is one in many cases. Based on this result, the characteristic of an operation error where the number of fault bits is biased to be small can be verified. The reason for this small number of fault bits is probably that even when the setup time constraint is violated using a glitch, the value of a normal cipher is sometimes the same as that of a cipher containing a fault.

Fig. 8 shows the results obtained by equalizing the memories of the vertical axes in Fig. 7. As shown in Fig. 8, the tendency of a fault differed according to the state. The number of faults is particularly large at the first and third columns and is small at the 0th and second columns (when the leftmost column is expressed as the 0th column). It is believed that the reason behind this is that the processing time differed according to the state, and that the processing time is particularly long for the first and third columns, so many faults tend to occur at these columns.

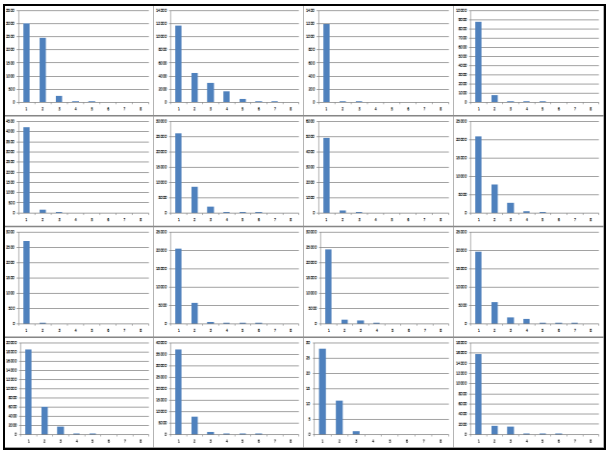


Figure 7. Details of the faults that have occurred using 100,000 cipher pairs

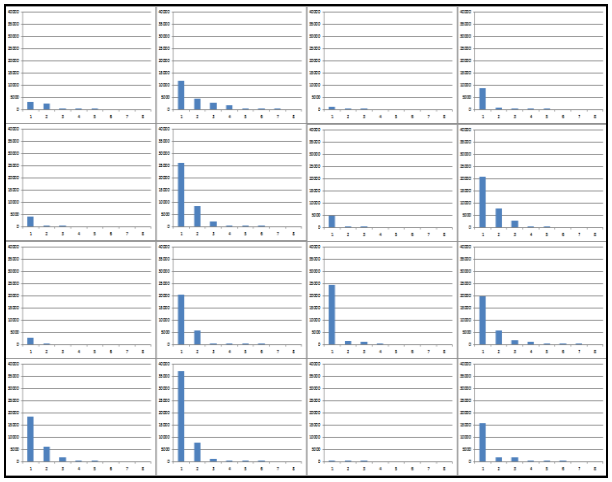


Figure 8. Results obtained by equalizing the memories of the vertical axes

V. Conclusion

This study clarified the tendency of an operation error due to a glitch using the AES. The analysis performed by the proposed method found the characteristic of an operation error, which stated that the probability of the number of fault bits being one was high.

In the future, we will examine the tendency of other illegal attacks such as power analysis attacks.

Acknowledgment

This study was supported by Japan Science and Technology Agency (JST), Core Research for Evolutional Science and Technology (CREST).

References

[1] E. Brier, C. Clavier, and F. Olivier, "Correlation power analysis with a leakage model," *Lecture Notes in Computer Science*, vol. 3156, pp. 16-29, 2004.

[2] G. Piret and J.-J. Quisquater, "A differential fault attack technique against SPN structure, with application to the AES and Khazad," *Lecture Notes in Computer Science*, vol. 2779, pp. 77-88, 2003.

[3] C. H. Kim and J. J. Quisquater, "Faults, injection methods, and fault attacks," *IEEE Design and Test of Computers*, vol. 24, no. 6, pp. 544-545, 2007.

[4] M. Ono, M. Katsube, M. Shiozaki, T. Fujino, and M. Yoshikawa, "Architecture aware fault analysis based on differential presumption for multiple errors and its evaluation," *IEEJ Trans. EIS*, vol. 132, no. 12, pp. 1888-1896, 2012.

[5] P. Maistri, R. Leveugle, "Double-data-rate computation as a countermeasure against fault analysis," *IEEE Trans. on Computers*, vol. 57, no. 11, pp. 1528-1539, 2008.

[6] L. Yang, K. Ohta, and K. Sakiyama, "New fault-based side-channel attack using fault sensitivity," *IEEE Trans. on Information Forensics and Security*, vol. 7, no. 1, pp. 88-97, 2012.

[7] P. C. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," in *Advances in Cryptology—CRYPTO'99*, Springer Berlin Heidelberg, 1999, pp. 388-397.

[8] E. Brier, C. Clavier, and F. Olivier, "Correlation power analysis with a leakage model," in *Cryptographic Hardware and Embedded Systems*, Springer Berlin Heidelberg, 2004, pp. 16-29.

[9] S. Nikova, C. Rechberger, and V. Rijmen, "Threshold implementations against side-channel attacks and glitches," in *Proc. ICICS*, 2006, pp. 529-545.

[10] T. Asai and M. Yoshikawa, "Efficient acquisition of the side-channel information using event model simulation methods," in *Proc. 30th Symposium on Cryptography and Information Security*, 2013.

[11] R. Satoh, D. Matsushima, and M. Yoshikawa, "Subkey driven power analysis attack in frequency domain against cryptographic LSIs," in *Proc. 17th Workshop on Synthesis And System Integration of Mixed Information Technologies*, 2012, pp. 262 - 267.

[12] T. Pop and S. Mangard, "Masked dual-rail pre-charge logic: DPA-Resistance without routing constraints," in *Cryptographic Hardware and Embedded Systems*, Springer Berlin Heidelberg, 2005, pp. 172-186.

Masaya Yoshikawa is a professor in the Department of Information Engineering at Meijo University, Nagoya, Japan.
He received his B.E., M.E. and Ph.D degrees from Ritsumeikan University, Shiga, Japan, in 1996, 1998, and 2001, respectively.
His research interests include LSI design methodology, and cryptographic hardware. He is a member of IEEE, IEEJ, IPSJ, and ISCIE.

Hikaru Goto received his B.E.and M.E. degrees from Meijo University, Nagoya, Japan, in 2012 and 2014, respectively.
He was engaged in research on tamper-resistance device, especially, against fault injection attacks.