Development and evaluation of a microstep DFA vulnerability estimation method

Masahiro Kaminaga^{a)}, Arimitsu Shikoda, and Hideki Yoshikawa

Faculty of Engineering, Tohoku-Gakuin University, 1–13–1 Chuo, Tagajo, Miyagi 985–8537, Japan

a) kaminaga@tjcc.tohoku-gakuin.ac.jp

Abstract: Recently, various studies of attack methods of round reduction differential fault analysis (DFA) using fault injection in block cipher-implemented microcontrollers have been reported. However, few studies have focused on the quantitative evaluation method of round reduction DFA vulnerability using detailed fault injection timing dependency of attack success rate. This is required to improve microcontroller security. Hence, we propose a quantitative evaluation method against round reduction DFA using a micro step DFA vulnerability chart and a vulnerability estimator (VE) that consists of pairs of fault injection timing and attack success rate.

Keywords: DFA, fault analysis, round reduction

Classification: Electronic instrumentation and control

References

- [1] D. Boneh, R. DeMillo, and R. Lipton, “On the Importance of Checking Cryptographic Protocols for Faults,” *EUROCRYPT’97*, LNCS 1233, pp. 37–51, 1997.
- [2] E. Biham and A. Shamir, “Differential Fault Analysis of Secret Key Cryptosystems,” *Proc. CRYPTO’97*, LNCS 1294, pp. 513–525, 1997.
- [3] S. P. Skorobogatov and R. Anderson, “Optical Fault Induction Attacks,” *Proc. CHES 2002*, Springer LNCS 2523, pp. 2–12, 2003.
- [4] H. Choukri and M. Tunstall, “Round Reduction Using Faults,” *Proc. FDTTC*, pp. 13–24, 2005.
- [5] J. Park, S. Moon, D. Choi, Y. Kang, and J. Ha, “Differential Fault Analysis for Round-Reduced AES by Fault Injection,” *ETRI Journal*, vol. 33, no. 3, pp. 434–442, 2011.

1 Introduction

Differential fault analysis (DFA) is a powerful technique used to extract secret information from a device through injection of faults. Boneh et al. [1] presented the first DFA for major public-key cryptosystems, and Biham-

Shamir [2] demonstrated a DFA for block ciphers such as the data encryption standard (DES). Some researchers have suggested that faults can be induced by abnormal power supplies, an abnormal clock signals, electromagnetic radiation, or an optical flush [3]. Various fault attacks and countermeasures have been developed for DFA against major cryptosystems. For example, when a branch operation is bypassed in block ciphers, leading to a reduction in the number of rounds, the attacker can reconstruct the secret key using fundamental cryptanalysis for block ciphers. Choukri and Tunstall [4] reported successful round reduction by bypassing the branch operations in a PIC microcontroller (Microchip Technology, USA). Their experiment was the first of its kind and provided valuable results of round reduction in real microcontrollers. Park et al. [5] obtained similar results using laser beam injection with an ATmega microcontroller (Atmel, USA). To improve the security of a microcontroller, the defender requires detailed fault injection timing dependency on the attack success rate. However, these studies did not focus on this point. The main purpose of our study is to develop a quantitative evaluation method using a microstep DFA station that is based on DFA success rates.

2 Vulnerability estimation of DFA

2.1 Principle of round reduction DFA

In this section, we explain the principle of round reduction DFA. In round reduction DFA, it is essential to consider the encryption/decryption program as a set of machine-language level operations. That is, the program code can be decomposed into operations such as MOVE, EXOR, and BRANCH. These operations are synchronized by a clock signal. The round reduction technique is based on the bypassing of branch operations. For example, we can get the result of 15 rounds by bypassing the 15th branch operation, which acts as a round counter in DES. This means that one round operation is reduced. The attacker can reconstruct the subkey for the last round by comparing the correct result to the round-reduced result on the basis of well-known cryptanalysis of one-round operations for DES. It is easy to apply this technique to other block ciphers such as the advanced encryption standard (AES).

2.2 Microstep evaluation method

To establish a quantitative evaluation method using a microstep DFA vulnerability chart, we developed a microstep DFA station as shown in Fig. 1. The timing chart shows that the branch command is responsible for counting the number of rounds of the main routine. For ordinary DES operations, this main routine must be repeated 16 times. The jmp command that follows the brne command is used to jump to the final inverse permutation. In any case, for this attack, this routine that involves the final inverse permutation must be executed without any fault. The cpi command checks whether the number of rounds has reached 16. A low-voltage period of more than 1.0 [μ s]

is used to attack the cpi and brne commands in order to skip the 15th branch command. One of the most essential steps in this experiment is to precisely determine the target command. To identify the exact attack point of the brne command in the program, the target microcontroller (ATmega168) outputs an attack request signal to the function generator prior to the 16th branch operation of the DES round program. This attack request signal is sent by the out command to the power source after the 15th main routine is completed. This signal is unnecessary when scanning the entire program execution range. After receiving the attack request, the function generator temporarily converts the power supply to a step-down voltage shortly after a configured delay time and then recovers the voltage after a specified low-voltage period. The voltage instantaneously steps down from 5.5 [V] to 1.8 [V] and then recovers instantaneously to 5.5 [V]. The drop occurs at about 1 [μ s] after the attack request signal and the delay time is increased in 100 [ns] increments. The probing range of the delay time and the low-voltage duration can each be configured using software on a PC. The DES calculation results are observed using the logic analyzer function in the oscilloscope with an 8-bit parallel digital probe. The 64 bits of the calculation results are divided into 8-bit segments and forwarded to the logic analyzer. The results are then recorded onto a USB flash memory because of data saving throughput. For example, if we set the input plaintext to be “54 65 73 74 54 65 73 74” (“TestTest” in ASCII) with a key of “13 CB 73 BE A1 C1 ED 5B,” the corresponding ciphertext is “C(16): 0F CB CE AB A2 AF 80 22.” When a bypass attack succeeds, the output of the 15th round is “C(15): 87 45 ED 7D 53 7D 48 11.” In cases where the result of the 15th round is obtained, DFA is successfully realized. Finally, the attack success rate chart and the VE value can be obtained. This experimental process is executed 400 times in 100 [ns] increments in order to determine the delay parameters and the low-voltage period needed to automate the tedious process of fault injection. Therefore, we have developed a DFA station. In this experiment, the delay between the attack request pulse and the start of the voltage step-down operation is fixed at 1000 [ns]. Moreover, the low-voltage time period from 500 to 3000 [ns] (in 100 [ns] increment). This low-voltage attack causes the program to bypass comparisons with both immediate (cpi) and branch not equal (brne) commands.

2.3 Vulnerability estimator

A comparison of the vulnerabilities of different-conditioned (clock speed and number of sampling points) target chips is not easily realized when using the microstep DFA vulnerability chart. We, therefore, define a DFA vulnerability estimator (VE) whose success rates vary depending on the timing of the attack t in the cipher operation. We denote the DFA success rate at time t by $r(t)$. Therefore, VE is defined as follows:

$$VE = \sum_{j=0}^{N-1} r(j\Delta t) \frac{\Delta t}{T} \sim \frac{1}{T} \int_0^L r(t) dt, \quad (1)$$

Fig. 1. Block Diagram of the DFA Station

where $[0, L]$ is an interval that includes the timing of the target operation executions, Δt is the sampling period, T is the clock period, and N is the number of sampling points. That is, VE indicates the size of the security hole. Here Δt must be set to be sufficiently small, then the VE defined in the left hand side of (1) can be approximated by the right hand side of (1). When comparing different-conditioned target microcontrollers, we must keep $\Delta t/T$ constant. Consequently, VE is independent of the clock speed.

3 Results

It is well known that using a low-voltage detector is an effective countermeasure against momentary voltage step-down attacks. An on-chip brown-out detection (BOD) circuit prevents the system from malfunctioning during periods of insufficient power. The microcontroller is reset when the supplied voltage V_{cc} falls below the programmed threshold voltage. The trigger level for BOD can be set to either 1.8, 2.7, or 4.3 [V] using a brown-out detec-

Fig. 2. Result of a single branch attack with BOD

tion level (BODLEVEL) fuse. In this study, we adopted a BOD threshold voltage of 2.7 [V]. With this threshold, a short power interruption of 1.8 [V] enables a command to be skipped. A voltage of 1.6 [V] is too low and causes a halt, whereas 2.0 [V] enables normal operation. We experimentally found that a voltage of around 1.8 [V] is sufficient for our experiment. To examine the effectiveness of BOD, further experiments were performed using single or multiple brne commands with BOD-ON (Figs. 2 and 3). We found that 2.4 [μs] of idle time is needed for the ATmega168 to reset the processor after detecting an insufficient voltage level. Fig. 2 shows that a single brne command is easily skipped because the low-voltage range is within 2.4 [μs]. However, the initial part of the second brne command can be attacked because it occurs within 2.4 [μs] (Fig. 3). Finally, we confirmed that as the third brne command occurs after 2.4 [μs], all the attacks failed. This means that a combination of multiple branches and BOD would be an appropriate countermeasure against operations that bypass attacks with step-down voltages. In this case, Fig. 2 shows that the use of BOD is not a sufficient countermeasure against DFA because the VE value is 99.575. When multiple brne commands are adopted together with BOD, the VE value decreases to 27.950 because the operation time of two brne commands exceeds 2.4 [μs]. Consequently, we can conclude that a combination of BOD with multiple branches can reduce the objective vulnerability.

Fig. 3. Result of a doubled branch attack with BOD

4 Conclusions

In this paper, we propose a quantitative evaluation method using a microstep DFA vulnerability chart and a VE. The aim of this study is to provide the defender with more detailed quantitative information regarding how to improve the security of the chips. To this end, we carefully observed the timing of the fault injection to each command and the round reduction DFA success rates. In this paper, the round reduction vulnerability has been quantitatively evaluated using a microstep DFA vulnerability chart and a VE. Microcontroller vendors should focus on minimizing the VE in order to reduce vulnerabilities against round reduction DFA.