Efficient countermeasure against fault injection attacks on modular exponentiation algorithms

O. Sho Endo, Naofumi Homma and Takafumi Aoki
Graduate School of Information Sciences, Tohoku University

1. Introduction
Modular exponentiation is a major arithmetic operation for public-key cryptosystems such as the RSA scheme. In recent years, fault injection attacks such as Bellcore attack and safe-error attack [1] have been proposed against the modular exponentiation algorithms. In this paper, we present an efficient fault detection scheme as a countermeasure against such fault injection attacks on modular exponentiation algorithms.

2. Fault injection attack on modular exponentiation
The square-and-multiply always method[2] performs modular exponentiation as shown in Fig.1, where the dummy multiplication is processed for the zero bits of the exponent in order to perform both squaring and multiplication for each bit. This algorithm is resistant to simple power analysis attacks, but is vulnerable to the safe-error attack which induces a carefully timed fault during the multiplication process. If the returned result is correct, the attacker can find that the multiplication is a dummy and the secret key bit is zero since the result of the dummy multiplication is never used in the following process.

3. Proposed countermeasure
The Montgomery multiplication [3] is widely used for calculating modular multiplication in the square-and-multiply always method because the algorithm can compute it without division. Fig. 2 shows the algorithm of the Montgomery multiplication, where the division on the line 3 is substituted with a shift operation because \( R \) is the power of 2. The intermediate variable \( Z_t \) at the line 2 is described as follows:

\[
Z_t = XY + TN \\
= XY + \{(XY \cdot WN) \mod R\} \\
= XY + \{(XY \cdot (-N^{-1} \mod R))N \mod R\} \\
= XY - (XY \mod R) \cdot n \cdot R, \tag{1}
\]

where \( n \) is an arbitrary integer. Thus, \( Z_t \) is always a multiple of \( R \). Based on the feature, we propose a fault detection scheme during the Montgomery multiplication in which the lower half bits of an intermediate variable is always zero. The proposed scheme is to check whether the lower half bits of \( Z_t \) is zero or not. If one or more of those bits are one, we can find that the fault is injected.

4. Experiment
We have implemented the experimental setup on SASEBO-W and have confirmed the validity of the proposed countermeasure. Fig. 3 illustrates the glitchy-clock generator implemented on SASEBO-W. The generator can inject faults timely to cryptographic operations. Fig. 4 shows the overview of the experimental environment. We have used a SASEBO-W, an oscilloscope, and a PC. Fig. 5 depicts an example of power trace obtained from the IC card on the SASEBO-W.

5. Conclusions
We have proposed an efficient fault detection scheme using a feature of the Montgomery multiplication algorithm. Further experiments are being conducted to show the applicable scope of proposed method against a variety of fault injection attacks.

References