New Single-Trace Side-channel Attacks on a Specific Class Elgamal

N. Mahdion, Hadi Soleimany, Pouya Habibi, Farokhlagha Moazami
Cyberspace Research Institute, Shahid Beheshti University, Tehran, Iran
p.mahdion@sbu.ac.ir, h_soleimany@sbu.ac.ir, f_moazemi@sbu.ac.ir

1 Introduction
The main purpose of SCAs on RSA and Elgamal public key encryptions is modular exponentiation algorithms that involve multiplication and squaring operations. However most implementation of multiplication and square algorithm use the similar sequence instructions, it is very difficult to distinguish between multiplication and square process for random input message[7]. In order to overcome this problem many researchers proposed some attacks such that distinguishing between square and multiply is depend on the chosen input message[8]. SPA with adaptively chosen messages are proposed by Novak[8], applicable on RSA implementation. There are some chosen message attacks on public key encryption in [4, 5, 11, 5]. In 2005, Yen et al. proposed the first N − 1 attack on the modular exponentiation algorithms where N is the modulus in RSA or Elgamal and all its powers are either 1 or −1 [13]. The N − 1 as input makes collisions in internal state of modular exponential which make it possible to distinguish between squaring and multiplying operations in only single-trace. Table 1 shows brief of this attack on SMA algorithm. Fig. 1 is an example for The N − 1 attack against SMA algorithm in one power trace.

2 Main Objectives
Several researches presented many different countermeasures to guard the implementation of exponentiation functions against the N − 1 attack. However, a strong countermeasure cannot be established for free. It has an impact on the latency, power consumption and size of the implementation. Consequently, simple countermeasures in real-world applications are preferred as they utilize a small overhead, rather than complicated solutions with notable extra cost. In this direction, to protect the implementation against the N − 1 attack, several attacks propose the simplest solution, i.e. “block the special message −1”.

This research demonstrates that SMA and ML algorithms are vulnerable by N −1-type attacks even if the special message N −1 is blocked. We illustrate the practical feasibility of our attacks in real-life experiments using SPA attack.

3 Mathematical Section
3.1 Our chosen ciphertext
Basic idea of our attack is similar to the original N − 1 attack. We select a ciphertext that enhances the differences between executed operations during the modular exponentiation algorithms according to the bit pattern of the secret key. We utilize a chosen message c such that \(c^2 \equiv 1 \pmod{p}\) where p = dk. In the following, we show how create our ciphertext by using Little Fermat Theorem.

Little Fermat Theorem: If p is a prime number and a is any number not divisible by p, then:

\[a^{p-1} \equiv 1 \pmod{p}\]

Based on the Little Fermat theorem, we can conclude a \(a^{p-1} \equiv 1 \pmod{p}\). In mathematics, if \(a^{p-1} = 1 \pmod{p}\), a is called a quadratic residue mod p and if \(a^{p-1} \neq 1 \pmod{p}\), a is called a non-residue mod p. Let c be a non-residue mod p so that \(c^p \equiv 1 \pmod{p}\) when \(p = 4d + 1\). We have:

\[c^{2d} \equiv c \pmod{p}\]

Therefore, we have \(c^{2d-1} \equiv 1 \pmod{p}\) and c is the appropriate number to choose as a ciphertext.

3.2 Attack on Square and Multiply Always Algorithm
One of the efficient algorithms for modular exponentiation is the Square and Multiply Always algorithm (SMA). Algorithm 1 illustrates an implementation of the SMA algorithm.

Algorithm 1 Square and Multiply Always (SMA)

Require: \(a, d = (d_0, d_1, \ldots, d_k), c\)
Ensure: \(m = c^d \pmod{p}\)
1: \(m \leftarrow 1\)
2: for \(i = 0\) to \(d_k - 1\) do
3: \(m \leftarrow m^2 \pmod{p}\)
4: if \(d_i \equiv 1 \pmod{2}\) then
5: \(m \leftarrow m \times c \pmod{p}\)
6: end if
7: end for
8: return \(m\)

By crafting the ciphertext \(c\) such that \(c^2 \equiv 1 \pmod{p}\), the output of squaring (line 3 of Algorithm 1) is always 1 or 3. These values inter as input of the multiplication operation (line 4). Hence, when the bit of the secret key is \(d_i = 1\) then the input of the squaring operation is either \(c^2 \pmod{p}\) and when the bit of the secret key is \(d_i = 0\) then the input of squaring operation is either 1 or \(-p\).

Table 1: Summary of our attack on SMA Algorithm

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>SMA</td>
<td>(c^2 \equiv 1 \pmod{p})</td>
</tr>
</tbody>
</table>

Since \((p - 1)\) is a large random-looking number in comparison to the value of \(p\), we expect that the differences in the patterns of two multiply operations \((p - 1) \times c \pmod{p}\) and \((1 \times c) \pmod{p}\). These differences can be easily perceived in a single-trace of side-channel information by visual observation.

3.3 Attack on Montgomery Ladder Algorithm
Another popular algorithm for modular exponentiation is the Montgomery Ladder algorithm (ML) which is demonstrated in Algorithm 2.

Algorithm 2 Montgomery Ladder (ML)

Require: \(c, d = (d_0, d_1, \ldots, d_k), c \pmod{p}\)
Ensure: \(m = c^d \pmod{p}\)
1: \(R_0 \leftarrow c\)
2: \(R_1 \leftarrow R_0 \times R_0 \pmod{p}\)
3: for \(i = 1\) to \(d_k - 1\) do
4: \(R_2 = R_1 \times R_1 \pmod{p}\)
5: \(R_1 \leftarrow R_1 \times R_0 \pmod{p}\)
6: end for
7: return \(R_1\)

Let us assume that the ciphertext \(c\) such that \(c^2 \equiv 1 \pmod{p}\) is given as an input to the ML algorithm. By computing the executed operations on two consecutive secret exponent bits \(d_i\) and \(d_{i+1}\), we find out that the modular squaring operation can be exploited solely to reveal the secret exponent bits, as Table 3. In this Table, \(N\) denotes squaring operation at the \((s - 1)\)-th iteration.

Table 3: Summary of our attack on ML Algorithm

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>ML</td>
<td>(c^2 \equiv 1 \pmod{p})</td>
</tr>
</tbody>
</table>

The operations in Table 3 can easily perceived in a single-trace of side-channel information by visual observation. Notice, our attack is also applicable on Montgomery Powering Ladder algorithm (Modified ML algorithm).

4 Experimental results
We verified our theoretical model by implementing SMA and ML algorithms on an Attmel ATXMEGA128D8 8-bit micro-controller which was located on the TARGET Board of the ChipWhisperer CW1173 [9]. Fig. 2 illustrates our proposed attacks on SMA and ML algorithms.

5 Conclusions
In this research, we proposed a new chosen ciphertext attack on Elgamal encryption which implements by using SMA and ML algorithms. Our ciphertext is \(c\), such that \(c^2 \equiv 1 \pmod{p}\), where p is the prime module and the public key in Elgamal cryptosystem. We exploited the leakage of power consumption to confirm the practicality of the proposed attack during the decryption execution of a specific ciphertext \(c\).

References

Figure 1: The N − 1 attack on SMA algorithm in single power trace.

Figure 2: Experimental results of our proposed attacks.