Improved Protection of AES with Shamir's Secret Sharing Scheme

Jean-Sébastien Coron

March 30, 2012

Abstract

At CHES 2011 Goubin and Martinelli described a new countermeasure against side-channel analysis for AES based on Shamir's secret-sharing scheme. However their countermeasure has $\mathcal{O}(d^3)$ complexity for security against d-th order attack, instead of $\mathcal{O}(d^2)$ for the Boolean masking coutermeasure. In this paper we show a variant with complexity $\mathcal{O}(d^2)$.

1 Description

We work in a finite field $GF(2^n)$. Let $\alpha \in GF(2^n)$, with $\alpha \neq 0$. Given a sensitive variable y, we consider a random polynomial $a(x)$ of degree $\leq d$, such that $a(\alpha) = y$; therefore the polynomial $a(x)$ has $d+1$ coefficients. The goal is to be secure against a d-th order attack, or at least $d/2$.

1.1 Addition

Given two sensitive variables y and z with $a(\alpha) = y$ and $b(\alpha) = z$, we can compute $y+z$ by computing $a(x) + b(x)$.

1.2 Multiplication

To compute $y \cdot z$, we first compute the polynomial

$$c(x) = a(x) \cdot b(x)$$

Then $c(\alpha) = a(\alpha)b(\alpha)$. However $c(x)$ is of maximum degree $2d$ instead of d. Therefore we collapse some of the coefficients of $c(x)$ to obtain $c'(x)$ in the following way. We write:

$$\begin{aligned}
c(\alpha) &= \sum_{i=0}^{2d} c_i \cdot \alpha^i \\
&= \sum_{i=0}^{d-1} \left(c_i \cdot \alpha^i + c_{d+i+1} \cdot \alpha^{d+i+1} \right) + c_d \cdot \alpha^d \\
&= \sum_{i=0}^{d-1} \left(c_i + c_{d+i+1} \cdot \alpha^{d+1} \right) \cdot \alpha^i + c_d \cdot \alpha^d
\end{aligned}$$

Therefore we let $c'_i = c_i + c_{d+i+1}$ and $c'_d = c_d$. We get $c'(\alpha) = c(\alpha)$ as required.

1.2.1 Computing $c(x) = a(x) \cdot b(x)$

We write:

$$\begin{aligned} c(x) &= \left(\sum_{i=0}^{d} a_i x^i\right)\left(\sum_{j=0}^{d} b_j x^j\right) \\ &= \sum_{i=0}^{d}\sum_{j=0}^{d} a_i b_j x^{i+j} = \sum_{k=0}^{2d} c_k x^k \end{aligned}$$

Then the technique consists in computing the partial sums

$$c_k = \sum_{i+j=k} a_i b_j$$

in the same way as they are computed in the Ishai *et al.* paper of Crypto 2003 and in the Rivain-Prouff paper of CHES 2010.

In the CHES 2010 paper, one must compute the product:

$$\begin{aligned} c &= \left(\sum_{i=0}^{d} a_i\right)\left(\sum_{j=0}^{d} b_j\right) \\ &= \sum_{i=0}^{d}\sum_{j=0}^{d} a_i b_j \end{aligned}$$

and the partial sum:

$$c_i = \sum_{j=0}^{d} a_i b_j$$

is essentially computed by adding for every j a random $r_{i,j}$ to both c_i and c_j.

Similary when computing the partial sum

$$c_k = \sum_{i+j=k} a_i b_j$$

we can add for every i a random r to c_k and a random $r \cdot \alpha^{k-k'}$ to $c_{k'}$ for a well chosen (varying) k'.

2 Security Proof

The previous countermeasure is secure against $d/2$-th order masking.