A remark on a success rate model for DPA and CPA

A. Wiemers, BSI
Version 0.5
andreas.wiemers@bsi.bund.de

September 5, 2018

Abstract

The success rate is the most common evaluation metric for measuring the performance of a particular side channel attack scenario. We improve on an analytic formula for the success rate.

Keywords: Side-channel attacks, evaluation metric, success rate, DPA, CPA

1 Introduction

In [1] a general statistical model for side-channel attack analysis is proposed. Based on this model one can calculate a success rate of an attack by numerical simulation. This success rate is the most common evaluation metric for measuring the performance of a particular attack scenario. In [5] it is stated:
"Closed-form expressions of success rate are desirable because they provide an explicit functional dependence on relevant parameters such as number of measurements and signal-to-noise ratio which help to understand the effectiveness of a given attack and how one can mitigate its threat by countermeasures. However, such closed-form expressions involve high-dimensional complex statistical functions that are hard to estimate."
In the following, we will derive an analytic formula for the success rate. Simulation experiments confirm that this analytic formula is a good approximation for the success rate for a wide class of leakage functions.

2 Leakage model

We restrict ourselves to the case of a side-channel attack on AES. We further assume the simplest setting:

- The attacker tries to find the 8-bit subkey k_0 of a specific S-Box-computation in the first round of AES.
- We have m measurements. m is a multiple of 256 and all plaintext inputs p_w of this S-Box are equally distributed over these m measurements.

- The side-channel measurement is a trace of a certain number of points. We assume that the key-dependent leakage occurs in just one point of time which is known to the attacker.

- The measurement in this point of time is the sum of a deterministic signal and Gaussian noise. It can be written in the form

$$\tilde{b}_w = \tilde{h}(p_w \oplus k_0) + \tilde{\tau}_w$$

  $\tilde{h}$ is a deterministic function that only depends on the input $p_w \oplus k_0$ of the S-Box-computation. $\tilde{h}$ is completely known to the attacker. $\tilde{\tau}_w$ describes the noise of the measurement. We assume that $\tilde{\tau}_w$ are realizations of m independent random variables $\tilde{T}_w$, each one is normally distributed with known expectation and variance. We further assume

$$E(\tilde{T}_w) = 0, V(\tilde{T}_w) = \sigma^2, \sum_{z=0}^{255} \tilde{h}(z) = 0, \sum_{z=0}^{255} \tilde{h}(z)^2 = 256\tilde{\delta}^2$$

- We can calculate the mean value of all $\tilde{b}_w$ with the same p_w. In the representation of $\tilde{b}_w$ this just reduces the variance of $\tilde{T}_w$. Additionally, by applying a constant factor to each $\tilde{b}_w$ we can normalize the representation of $\tilde{b}_w$. To this end, we get a representation in the form

$$b_w = h(w \oplus k_0) + \tau_w, w = 0, \cdots, 255$$

  with

$$E(T_w) = 0, V(T_w) = 1, \sum_{z=0}^{255} h(z) = 0, \sum_{z=0}^{255} h(z)^2 = 256\delta^2$$

  If we start with the representation of $\tilde{b}_w$, the normalized representation b_w has parameter δ with

$$\delta^2 = \frac{m}{256}\frac{\tilde{\delta}^2}{\sigma^2}$$

As in [1] we now apply the maximum likehihood attack: We compute the conditional probability density function of the observations b_w under each hypothesis k. We choose as the correct key that k which maximizes the probability density function. An easy calculation shows that we have to compare the values

$$\sum_{w=0}^{255} (b_w - h(w \oplus k))^2$$

This can further be reduced to the values

$$\sum_{w=0}^{255} h(w \oplus k) b_w$$

since $\sum_{w=0}^{255} h(w \oplus k)^2$ does not depend on k. The success rate as defined in [1] is the probability that

$$\Pr(X_{k_0} > X_k \text{ for all } k \neq k_0)$$

where X_k is the random variable

$$X_k = \sum_{w=0}^{255} h(w \oplus k)(h(w \oplus k_0) + T_w)$$

This success rate can certainly be computed by numerical simulation of the T_w.

3 An approximation of the success rate

Let A be the 256×256-matrix with entries $h(w \oplus k)$. The rows of A are

$$a_k = (h(k), \ldots, h(w \oplus k), \ldots, h(255 \oplus k))$$

Let T be the random vector (as column) of length 256 with entries T_w. Let $d = A \cdot a_{k_0}^t$ with entries d_k. We define the set R of all vectors of length 256 with entries y_k that fulfill

$$y_k < y_{k_0} + 256\delta^2 - d_k \text{ for all } k \neq k_0$$

An easy calculation shows that the success rate can be written as

$$\Pr(X_{k_0} > X_k \text{ for all } k \neq k_0) = \Pr(A \cdot T \in R)$$

A is a symmetric matrix and therefore there exists a orthonormal basis of eigenvectors $v_0, \ldots, v_{255}$ with corresponding eigenvalues $\lambda_0, \cdots, \lambda_{255}$ of A. Each T can be written in the basis of eigenvectors in the form

$$T = X_0 v_0 + \cdots + X_{255} v_{255}$$

where the X_i are independent random variables with standard normal distribution. The distribution of $A \cdot T$ is the image of the standard normal distribution under A. Each vector in the distribution of T is stretched in direction of the eigenvectors of A with the corresponding eigenvalue as factor.

$$A \cdot T = \lambda_0 X_0 v_0 + \cdots + \lambda_{255} X_{255} v_{255}$$

We easily compute

$$E(||A \cdot T||^2) = 256^2\delta^2 = \lambda_0^2 + \cdots + \lambda_{255}^2$$

Since 256 is a relatively large number the typical vector in the distribution of $A \cdot T$ has square of norm $256^2\delta^2$. As a heuristic approximation for the success rate we just replace the distribution of $A \cdot T$ by the normal distribution stretched by the constant factor 16δ:

$$\text{1st approx. formula:} \qquad \Pr(16\delta \cdot T \in R)$$

In addition we omit the influence of d and get

$$\text{2nd approx. formula:} \qquad \Pr(T \in \tilde{R})$$

where $\tilde{R}$ is the set of all vectors t_k that fulfill

$$t_k < t_{k_0} + 16\delta \text{ for all } k \neq k_0$$

The last probability can be in fact computed as a two-dimensional integral

$$\Pr(T \in \tilde{R}) = \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}} \exp(-\frac{1}{2}a^2) \left[\int_{-\infty}^{a+16\delta} \frac{1}{\sqrt{2\pi}} \exp(-\frac{1}{2}t^2) dt \right]^{255} da$$

This expression only depends on δ, so that it can easily be listed for different δ by numerical methods. Figure 1 plots this approximated success rate as computed by MAPLE software.

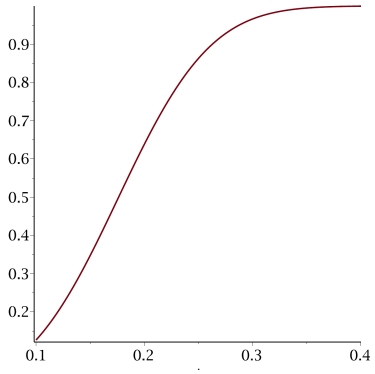

Figure 1: 2nd approx. formula. Success rate as function in δ

Remarks:

- If we start with the representation of $\tilde{b}_w$, the success rate as computed by the 2nd approximating formular only depends on

$$\delta^2 = \frac{m}{256} \frac{z^2}{\sigma^2}$$

- The approximating formulas are only valid if the eigenvalues do not vary too much. As an extreme example we can consider the case that only one eigenvalue is large whereas the others can be neglected. Let $\lambda_0 > 0$ be this large eigenvalue. Then $A \cdot T$ is roughly distributed as $\lambda_0 X_0 v_0$. $\Pr(A \cdot T \in R)$ can be written as a one-dimensional integral over the random variable X_0.

- In our approach we replaced the covariance matrix A^2 by a diagonal matrix. In effect we treated X_k as independent random variables.

- $\Pr(T \in \tilde{R}) \geq \frac{1}{256}$ with equality for $\delta = 0$. The probability of $\frac{1}{256}$ for $\delta = 0$ follows from the symmetry of the set $\tilde{R}$.

4 More on the matrix A

The properties of the matrix A are used in the context of dyadic codes, see [2]. In [3] the matrix A is called dyadic matrix. Due to the structure of A we can compute the eigenvectors of A explicitly: There are 256 GF(2)-linear functions L

$$L : \mathrm{GF}(2)^8 \longrightarrow \mathrm{GF}(2)$$

For every L, $v_L = [(-1)^{L(w)}]_w$ is a vector of length 256. For every k we have

$$\sum_w h(k \oplus w)(-1)^{L(w)} = \sum_y h(y)(-1)^{L(y \oplus k)} = (-1)^{L(k)} \sum_y h(y)(-1)^{L(y)}$$

Therefore, v_L is an eigenvector with eigenvalue $\sum_y h(y)(-1)^{L(y)}$. The rank of A is the number of non-zero eigenvalues.

5 Example: h depends on a single bit

Let S be the S-Box of the AES and G a fixed GF(2)-linear function. We assume that the leakage function h only depends on $G \circ S$, i.e. after normalization

$$h(w \oplus k) = \delta(-1)^{G(S(w \oplus k))}$$

The eigenvalues of A are now

$$\sum_y h(y)(-1)^{L(y)} = \delta \sum_y (-1)^{G(S(y))}(-1)^{L(y)}$$

With other words: The set of eigenvalues is exactly the Walsh spectrum of the boolean function $G \circ S$ multiplied by δ. Each eigenvalue is a measure how good $G \circ S$ can be approximated by a linear function L. S is the composition of the inversion over $F = \mathrm{GF}(256)$ and an affine function. The Walsh spectrum of any function of the form $G \circ S$ is well-known: It can be expressed by the so called Kloosterman sums, see [4].

$$K(a) = \sum_{y \in F^x} (-1)^{tr(y^{-1} + ay)}$$

where $tr(y)$ denotes the trace of y over F. Any GF(2)-linear function $L : F \longrightarrow \mathrm{GF}(2)$ can be written as $L(y) = tr(ly)$ for exactly one $l \in F$. Therefore, we find $c \in F$ such that

$$G(S(y)) \oplus L(y) = tr(cy^{-1} \oplus ly) \text{ for all } y \in F^x$$

or

$$G(S(y)) \oplus L(y) = tr(cy^{-1} \oplus ly) \oplus 1 \text{ for all } y \in F^x$$

Note that for $c \neq 0$

$$\sum_{y \in F^x} (-1)^{tr(cy^{-1} + ly)} = K(c \cdot l)$$

The distribution of the Kloosterman sums can be described by values of certain class numbers, see [4, Prop. 9.1], which can be interpreted in terms of the Walsh spectrum.

6 Example: h depends on the Hamming weight of the input

In this example h does not depend on the function S, but on the Hamming weight of the input. After normalization we can write

$$h(w \oplus k) = \delta g(w \oplus k) \text{ and } g(z) = \frac{1}{\sqrt{8}}((-1)^{z_1} + \cdots + (-1)^{z_8})$$

There are exactly 8 eigenvectors with eigenvalues $\neq 0$ and these are given by the 8 linear projections

$$v_j = \frac{1}{16}[(-1)^{z_j}]_{z=(z_1,\ldots,z_8)}$$

The eigenvalues of these 8 eigenvectors are equal to $\delta\frac{256}{\sqrt{8}}$. We have

$$A \cdot T = \lambda_0 X_0 v_0 + \cdots + \lambda_{255} X_{255} v_{255} = \delta\frac{256}{\sqrt{8}}(X_1 v_1 + \cdots + X_8 v_8)$$

The success rate can be described as an 8-dimensional integral. We cannot expect that the 2nd approx. formula is a good approximation in this case.
For an illustration we consider the following probability

$$\Pr(A \cdot T \leq 256\delta^2)$$

where we consider the inequality in every entry of $A \cdot T$. If we approximate $A \cdot T$ by $16\delta \cdot T$ we get as approximation

$$\Pr(A \cdot T \leq 256\delta^2) \approx \left[\int_{-\infty}^{16\delta} \frac{1}{\sqrt{2\pi}} \exp(-\frac{1}{2}t^2)dt\right]^{256}$$

The condition $A \cdot T \leq 256\delta^2$ is equivalent to

$$\delta\frac{16}{\sqrt{8}}(X_1(-1)^{k_1} + \cdots + X_8(-1)^{k_8}) \leq 256\delta^2 \text{ for all } k = (k_1, \ldots, k_8)$$

k runs over all 256 values so that any combination of signs will occur. Therefore we get equivalently the condition

$$(|X_1| + \cdots + |X_8|) \leq 16\delta\sqrt{8}$$

With other words: $\Pr(A \cdot T \leq 256\delta^2)$ can be considered as the 256 times the probability under a normal distribution of the 8-dimensional simplex given by

$$x_1 + \cdots + x_8 \leq 16\delta\sqrt{8}, x_i \geq 0$$

	$\delta = 0.1$	$\delta = 0.2$	$\delta = 0.3$
2nd approx. formula	0.13	0.64	0.97
h=$\delta(-1)^f$	$0.12\cdots 0.13$	$0.62\cdots 0.63$	0.96
h=$\delta g \circ P$	0.12	$0.62\cdots 0.63$	$0.95\cdots 0.96$
h=δg	0.07	0.33	0.69

Table 1: Comparison of success rates

7 Simulation results

We computed the success rate for different h and δ by numerical simulation of the T_w. Table 1 compares the success rate with the 2nd approximated formula. In table 1 f is chosen as a random function $\mathrm{GF}(2)^8 \longrightarrow \mathrm{GF}(2)$, but uniformly distributed. P is chosen as a random permutation on $\mathrm{GF}(2)^8$. g is the function from paragraph 6. We repeated the simulation 10 times with different f and P, so that a range is given in table 1.

We note that the 2nd approximating formula and the numerical values for $h = \delta(-1)^f$ and $h = \delta g \circ P$ match very well, which is not the case for $h = \delta g$.

8 Success rate in the case of masking

Similar to [5], we can apply the 2nd approximating formula to the case of masking. For a concrete example, we adapt our leakage model in the following way:

- We have m measurements. m is a multiple of 256 and all plaintext inputs p_w of this S-Box are equally distributed over these m measurements.
- There are exactly two points of time when meaning-full leakages occur. Both points of time are known to the attacker. One leakage is mask-dependent; the other one is key-dependent, but on the input of an S-Box-computation.
- The measurements can be written in the form

$$\begin{aligned} \tilde{b}'_w &= \mu(p_w \oplus k_0 \oplus m_w) + \tilde{\tau}'_w \\ \tilde{b}''_w &= \mu(m_w) + \tilde{\tau}''_w \end{aligned}$$

  μ is a centralized form of the Hamming weight, i.e.

$$\mu(z) = (-1)^{z_1} + \cdots + (-1)^{z_8}$$

  $\tilde{\tau}'_w$ and $\tilde{\tau}''_w$ describe the noise of the measurement. We assume that $\tilde{\tau}'_w$ and $\tilde{\tau}''_w$ are realizations of $2m$ independent random variables $\tilde{T}'_w$, $\tilde{T}''_w$, each one is normally distributed with expectation 0 and variance σ^2. m_w describes the mask. m_w are the realizations of m independent uniformly distributed random variables M_w on GF(256).

We set

$$c_\nu = \frac{256}{m} \sum_{w, p_w = \nu} \tilde{b}'_w \tilde{b}''_w$$

The sum is taken over $\frac{m}{256}$ realizations of independent random variable. For any fixed mask m_w, we compute

$$E((\mu(p_w \oplus k_0 \oplus m_w) + \tilde{T}'_w)(\mu(m_w) + \tilde{T}''_w)) = \mu(p_w \oplus k_0 \oplus m_w)\mu(m_w)$$

and

$$\begin{aligned} & V((\mu(p_w \oplus k_0 \oplus m_w) + \tilde{T}'_w)(\mu(m_w) + \tilde{T}''_w)) \\ = \;& E((\mu(p_w \oplus k_0 \oplus m_w) + \tilde{T}'_w)^2(\mu(m_w) + \tilde{T}''_w)^2) - \mu(p_w \oplus k_0 \oplus m_w)^2\mu(m_w)^2 \\ = \;& \sigma^2(\mu(p_w \oplus k_0 \oplus m_w)^2 + \mu(m_w)^2) + \sigma^4 \end{aligned}$$

If $\frac{m}{256}$ is not too small, we approximate c_ν as realizations of 256 independent normally distributed random variables, each with expectation

$$\frac{256}{m} \sum_{w, p_w = \nu} \mu(p_w \oplus k_0 \oplus m_w)\mu(m_w) = \frac{256}{m} \sum_{w, p_w = \nu} \mu(\nu \oplus k_0 \oplus m_w)\mu(m_w)$$

and variance

$$\left(\frac{256}{m}\right)^2 \sum_{w, p_w = \nu} \left(\sigma^2(\mu(\nu \oplus k_0 \oplus m_w)^2 + \mu(m_w)^2) + \sigma^4\right)$$

Again if $\frac{m}{256}$ is not too small, we approximate these sums by the expectation over the random variables M_w. An easy calulation shows

$$\frac{256}{m} \sum_{w, p_w = \nu} \mu(p_w \oplus k_0 \oplus m_w)\mu(m_w) \approx \mu(\nu \oplus k_0)$$

and

$$\left(\frac{256}{m}\right)^2 \sum_{w, p_w = \nu} \left(\sigma^2(\mu(\nu \oplus k_0 \oplus m_w)^2 + \mu(m_w)^2) + \sigma^4\right) \approx \frac{256}{m}(16\sigma^2 + \sigma^4)$$

Since $\sum_z \mu(z)^2 = 8 \cdot 256$ we can apply the leakage model of paragraph 2 with

$$\delta^2 = \frac{8m}{256(16\sigma^2 + \sigma^4)} = \frac{m}{32(16\sigma^2 + \sigma^4)}$$

Given the measurements $\tilde{b}'_w, \tilde{b}''_w$, we directly compare the values

$$\sum_\nu \mu(\nu \oplus k) c_\nu$$

for different k and decide for the k with the largest value. For large m, we can expect that the success rate of this ad-hoc attack only depends on $\delta^2 = \frac{m}{32(16\sigma^2+\sigma^4)}$. Table 2

	$\sigma = 37.6$ $\delta = 0.1$	$\sigma = 26.6$ $\delta = 0.2$	$\sigma = 21.6$ $\delta = 0.3$
Simulation	0.07	0.32	0.7
$h = \delta g$	0.07	0.33	0.69

Table 2: Success rates in the case of masking ($m = 10 \cdot 256^2$, input-dependency)

gives the success rates of this attack computed by numerical simulation. We compare this success rates with the values for the example from paragraph 6 ($h = \delta g$). Note that the values match very well.

Table 3 gives similar data, but for $m = 256^2$.

	$\sigma = 21.1$ $\delta = 0.1$	$\sigma = 14.8$ $\delta = 0.2$	$\sigma = 11.9$ $\delta = 0.3$
Simulation	0.08	0.34	0.72
$h = \delta g$	0.07	0.33	0.69

Table 3: Success rates in the case of masking, ($m = 256^2$, input-dependency)

Remark:
The leakage in $\tilde{b}'_w$ depends on the input of an S-Box-computation. We can certainly consider the case, that the leakage depends on the output of an S-Box-computation, i.e.

$$\tilde{b}'_w = \mu(S(p_w \oplus k_0) \oplus m_w) + \tilde{\tau}'_w$$

The computation is completely analog, but we expect that the 2nd approximating formula applies. Table 4 and 5 compares the numerical values for the success rate with the 2nd approximating formula.

	$\sigma = 37.6$ $\delta = 0.1$	$\sigma = 26.6$ $\delta = 0.2$	$\sigma = 21.6$ $\delta = 0.3$
Simulation	0.13	0.63	0.96
2nd approx. formula	0.13	0.64	0.97

Table 4: Success rates in the case of masking, ($m = 10 \cdot 256^2$, output-dependency)

	$\sigma = 21.1$ $\delta = 0.1$	$\sigma = 14.8$ $\delta = 0.2$	$\sigma = 11.9$ $\delta = 0.3$
Simulation	0.12	0.62	0.96
2nd approx. formula	0.13	0.64	0.97

Table 5: Success rates in the case of masking, ($m = 256^2$, output-dependency)

9 References

[1] Yunsi Fei and A. Adam Ding and Jian Lao and Liwei Zhang: A statistics-based success rate model for DPA and CPA, J. Cryptogr. Eng. (2015)
[2] Sundar Rajan and Moon Ho Lee: Quasi-Cyclic Dyadic Codes in the Walsh–Hadamard Transform Domain, IEEE Transactions on information theory, Vol. 48, No. 8, August 2002
[3] Misoczki R., Barreto P.S.L.M. (2009): Compact McEliece Keys from Goppa Codes. In: Jacobson M.J., Rijmen V., Safavi-Naini R. (eds) Selected Areas in Cryptography. SAC 2009. Lecture Notes in Computer Science, vol 5867. Springer, Berlin, Heidelberg
[4] Lachaud, Wolfmann: Weights of the orthogonals of the extended quadratic binary Goppa codes, IEEE Transactions on Information Theory, June 1990
[5] Sylvain Guilley, Annelie Heuser and Olivier Rioul: A Key to Success – Success Exponents for Side-Channel Distinguishers, Cryptology ePrint Archive: Report 2016/987