Success through confidence: Evaluating the effectiveness of a side-channel attack

Adrian THILLARD, Emmanuel PROUFF, Thomas ROCHE

ANSSI (French Network and Information Security Agency)

CHES 2013, Wednesday, August 21st

Santa Barbara, CA

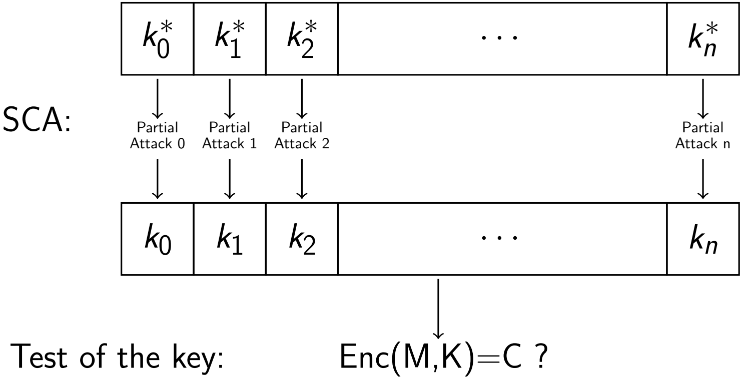
Context of this work

- Embedded Systems integrating Cryptography manipulate a secret key K.
- A Side-Channel Attack aims at recovering K through the observation of the device behavior.



Divide and conquer strategy

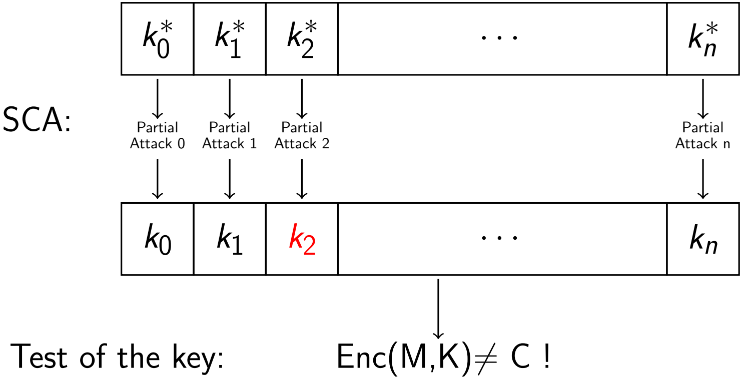
- Divide the secret in chunks and conquer each chunk separately



- Things might go wrong!

Divide and conquer strategy

- Divide the secret in chunks and conquer each chunk separately



- Some of the partial attack results are wrong! $\Longrightarrow$ all tests of full key will fail and it will be impossible to correct!

Divide and conquer strategy

- Assess a confidence to each result.

k_0	k_1	k_2	$\cdots$	k_n
90%	95%	30%	$\cdots$	92%
k_0	k_1	??	$\cdots$	k_n

- Missing chunks are retrieved using an exhaustive search or Key Enumeration Algorithm (*Veyrat-Charvillon* SAC 12).
- Issue: how to compute the confidence?

Computing confidence

	N=50msgs	N=150msgs	N=250msgs	$\cdots$	N=500msgs
k_0	94th	67th	89th	$\cdots$	1st
k_1	88th	224th	97th	$\cdots$	188th
k_2	160th	60th	77th	$\cdots$	2nd
k_3	146th	185th	58th	$\cdots$	250th
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
k_n	119th	159th	184th	$\cdots$	210th

Rankings obtained by CPAs attacks

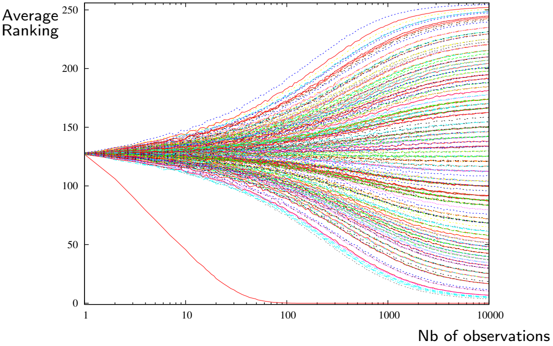
Computing confidence

- A large amount of information is not used!
- Issue: How to evaluate *a priori* the confidence ?
- Associate to each candidate a specific confidence
- Use success rate evaluations as tools to compute confidence when the correct key is unknown
    - Those evaluations are well-known and are accurate evaluation of the confidence, when the secret is known by the attacker
    - They can be adapted when the correct key is unknown
- Study of the most usual attack: CPA

CPA success rate

- CPA is based on correlation coefficient
- Formulas derived by *Mangard* (2004) and *Standaert et al.* (2006) assume that asymptotically, every wrong hypotheses leads to a null correlation
- Useful to have the general attacks trend BUT what about the accuracy of the formulas?
- Is this assumption correct?
- Experiment: Simulate several CPA on the AES Sbox output
    - assuming Hamming weight leakage
    - assuming Gaussian noise with std 3
    - average the rankings obtained for each hypothesis.

CPA success rate



- The correct hypothesis converges towards first place
- Wrong hypotheses also converge to a fixed rank!
- Hence, analyses without Mangard's assumption should be more accurate

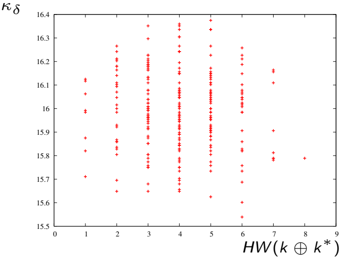
Confusion coefficient

- [*Rivain* SAC 08]: new evaluation of the CPA SR relaxing Mangard's assumption!
- [*Fei et al.* CHES 12]: DPA confusion coefficient for the DPA SR
- [This work]: Rivain's work can be rewritten using a new CPA confusion coefficient:

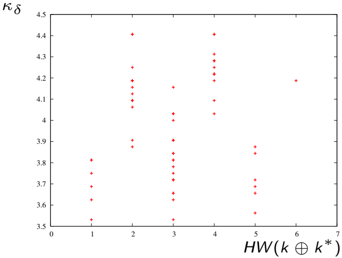
$$\kappa_\delta = \frac{1}{|\mathcal{X}|} \sum_{x \in \mathcal{X}} HW(SB[x \oplus k^*])HW(SB[x \oplus k^* \oplus \delta]),$$

where δ is the xor-distance of the hypothesis to the good hypothesis k^*

Confusion coefficient



AES Sbox coefficient



DES Sbox coefficient

- For the AES: $\kappa_0 = 18$ and for the DES: $\kappa_0 = 5$
- $\kappa_0 - MAX(\kappa_\delta)$ is higher for AES (1.67) than for DES (0.58) $\Longrightarrow$ ghost peaks are more likely to appear with DES

CPA success rate vs confusion coefficient

- Let ρ_k be the correlation coefficient for the hypothesis k
- Consider the comparison vector $\vec{c_{k^*}} = (\rho_{k^*} - \rho_k)_{k \neq k^*}$
- [*Rivain* SAC08]: the rank of k^*=# of positive coordinates in $\vec{c_{k^*}}$
- $\vec{c_{k^*}} \sim \mathcal{N}(\vec{\mu}, \Sigma)$ where:

$$\vec{\mu} = (\kappa_0 - \kappa_i)_{i>0},$$

$$\Sigma = \frac{\sigma^2}{N}(\kappa_0 - \kappa_i - \kappa_j + \kappa_{i \oplus j})_{i,j>0}.$$

Formulas' comparison

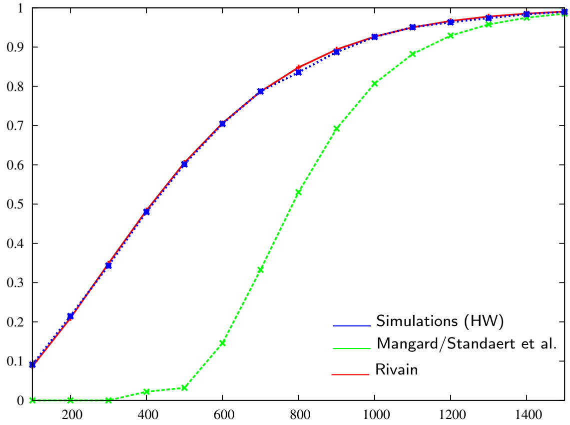


Figure : Comparison of success rate evaluations

Unknown correct key

Given a limited number N of observations, and an unknown correct key

- How to select a candidate?
- How to compute a confidence for this candidate?

Why do we need a confidence level?

- If the result of a partial attack is incorrect $\implies$ FAILURE of the whole attack
  - Even if exhaustive search is allowed
- Need for a confidence level
- Need to allow the selection rule to output no candidate
  - The cost of the hypotheses test step is increased by a factor $|\mathcal{K}|$ (e.g. by 256)
- We argue that we'd rather increase complexity than never pass the attack!

How to build confidence?

- Compute scores (correlation coefficients) for several numbers of observations, for each hypothesis in the subkey hypothesis set $\mathcal{K}$.
- Define a selection rule (usually choose the hypothesis with the highest score!).

Confidence level

For a selection rule $\mathcal{R}$ outputing an hypothesis $k^{\mathcal{R}}$, we define the confidence level $c(k^{\mathcal{R}})$:

$$c(k^{\mathcal{R}}) = \frac{P(k^{\mathcal{R}} = k^*)}{1 - P(k^{\mathcal{R}} = \emptyset)}$$

Note the difference with the success rate:

$$SR = P(k^{\mathcal{R}} = k^*)$$

$\Longrightarrow$ The success rate merges with the confidence level only when $\mathcal{R}$ always returns a candidate

Defining a selection rule

- We have 500 observations

	N=50msgs	N=150msgs	N=250msgs	$\cdots$	N=500msgs
k_0	0.35	0.39	0.39	$\cdots$	0.37
k_1	0.38	0.25	0.34	$\cdots$	0.15
k_2	0.29	0.40	0.42	$\cdots$	0.20
k_3	0.31	0.32	0.44	$\cdots$	0.09
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
k_n	0.33	0.34	0.17	$\cdots$	0.12

Average of correlation coefficients

- Averages are made on the 500 messages

Usual rule $\mathcal{R}$

- We have 500 observations

	N=50msgs	N=150msgs	N=250msgs	$\cdots$	N=500msgs
k_0	94th	67th	89th	$\cdots$	1st
k_1	88th	224th	97th	$\cdots$	188th
k_2	160th	60th	77th	$\cdots$	2nd
k_3	146th	185th	58th	$\cdots$	250th
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
k_n	119th	159th	184th	$\cdots$	210th

Rankings obtained by CPAs attacks

- Averages are made on the 500 messages

Usual rule $\mathcal{R}$

- We have 500 observations

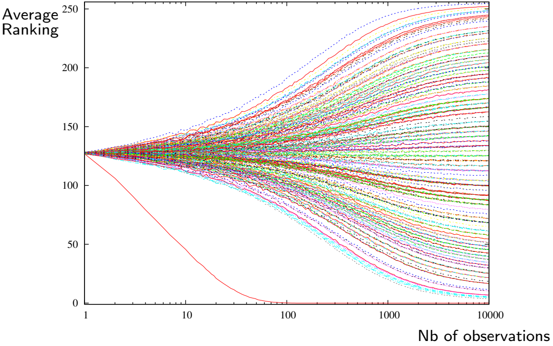
	N=50msgs	N=150msgs	N=250msgs	$\cdots$	N=500msgs
k_0	94th	67th	89th	$\cdots$	1st
k_1	88th	224th	97th	$\cdots$	188th
k_2	160th	60th	77th	$\cdots$	2nd
k_3	146th	185th	58th	$\cdots$	250th
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
k_n	119th	159th	184th	$\cdots$	210th

Rankings obtained by CPAs attacks

- Averages are made on the 500 messages

Constructing new rule

- Knowing that the candidate key is eventually ranked first, is the convergence towards this rank coherent with that of the correct hypothesis?



Constructing new rule

	N=50msgs	N=150msgs	N=250msgs	$\cdots$	N=500msgs
k_0	94th	67th	89th	$\cdots$	1st
k_1	88th	224th	97th	$\cdots$	188th
k_2	160th	60th	77th	$\cdots$	2nd
k_3	146th	185th	58th	$\cdots$	250th
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
k_n	119th	159th	184th	$\cdots$	210th

Rankings obtained by CPAs attacks

Rule $\mathcal{R}'$

	N=50msgs	N=150msgs	N=250msgs	$\cdots$	N=500msgs
k_0	94th	67th	89th	$\cdots$	1st
k_1	88th	224th	97th	$\cdots$	188th
k_2	160th	60th	77th	$\cdots$	2nd
k_3	146th	185th	58th	$\cdots$	250th
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
k_n	119th	159th	184th	$\cdots$	210th

Rankings obtained by CPAs attacks

An example - Defining a new selection rule

- Classic rule $\mathcal{R}$: select the best ranked key.
- Let's define selection rule $\mathcal{R}'$: output the best ranked hypothesis only if a CPA with the first $\frac{N}{2}$ observations also ranks this hypothesis first.
- Knowing the noise, we can extend the previous results and accurately compute the confidence level.

Comparison with "always outputs"

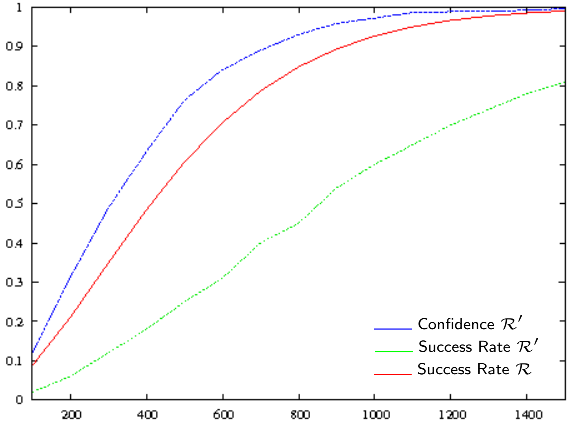


Figure : Comparison of rules, $\sigma = 10$

Interpretation

☹ The success rate associated to our rule is bad

☹ The rule induces a large number of false negatives

☺ But a small number of false positives

☺ Therefore we get a strong confidence in the result when we actually get one

Interpretation

- When the key was already ranked first, there is a better chance that it is indeed correct
- One can actually quantify this confidence
- This helps making a sound choice of which partial attacks to deem as successes, thus improving the efficiency of the whole attack

Another example

	N=50msgs	N=150msgs	N=250msgs	$\cdots$	N=500msgs
k_0	0.35	0.39	0.39	$\cdots$	0.37
k_1	0.38	0.25	0.34	$\cdots$	0.15
k_2	0.29	0.40	0.42	$\cdots$	0.20
k_3	0.31	0.32	0.44	$\cdots$	0.09
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
k_n	0.33	0.34	0.17	$\cdots$	0.12

Correlation coefficients obtained by CPAs attacks

Another example

- It has been proposed by Messerges to look at the difference between the scores of the first and second ranked keys

	N=50msgs	N=150msgs	N=250msgs	$\cdots$	N=500msgs
k_0	0.35	0.39	0.39	$\cdots$	0.37
k_1	0.38	0.25	0.34	$\cdots$	0.15
k_2	0.29	0.40	0.42	$\cdots$	0.20
k_3	0.31	0.32	0.44	$\cdots$	0.09
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
k_n	0.33	0.34	0.17	$\cdots$	0.12

Correlation coefficients obtained by CPAs attacks

Another example

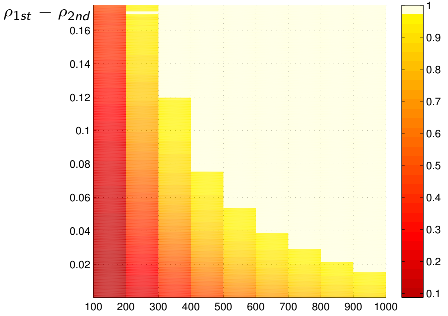


Figure : Confidence with $\sigma = 10$, given the difference between the two best ranked keys

Conclusion and perspectives

- Consider this table:

	N=50	N=150	N=250	$\cdots$	N=500
k_0	0.35	0.39	0.39	$\cdots$	0.37
k_1	0.38	0.25	0.34	$\cdots$	0.15
k_2	0.29	0.40	0.42	$\cdots$	0.20
k_3	0.31	0.32	0.44	$\cdots$	0.09
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
k_n	0.33	0.34	0.17	$\cdots$	0.12

- Define a rule exploiting more information in this table
- Are there redundant information in this table ?
- Find success rate formulas for other attacks (higher order CPAs, MIA, $\cdots$)
- Does the notion of confidence depend on the attack ?

$\ddot{\smile}$ Thank you for your attention!