Using the Joint Distributions of a Cryptographic Function in Side Channel

Yanis LINGE[1,2], Cécile DUMAS[1], Sophie Lambert-Lacroix[2]

CEA-LETI/MINATEC

UJF-Grenoble 1 / CNRS / UPMF / TIMC-IMAG.

Introduction

Context: Side channel attacks on embedded software cryptographic algorithm.

Objective: Recovering information from traces.

Introduction

Context: Side channel attacks on embedded software cryptographic algorithm.

Objective: Recovering information from traces.

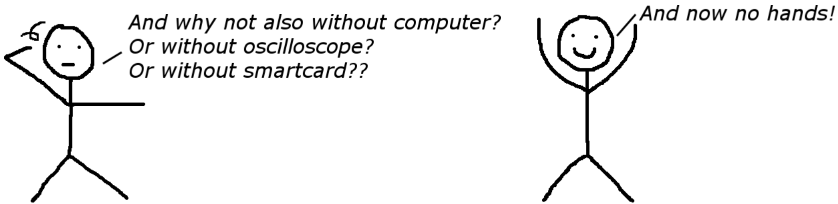
- without plaintext or ciphertext

Introduction

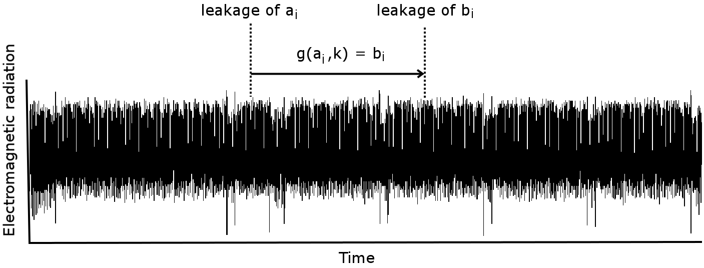
Context: Side channel attacks on embedded software cryptographic algorithm.

Objective: Recovering information from traces.

- without plaintext or ciphertext
- without profiling phase

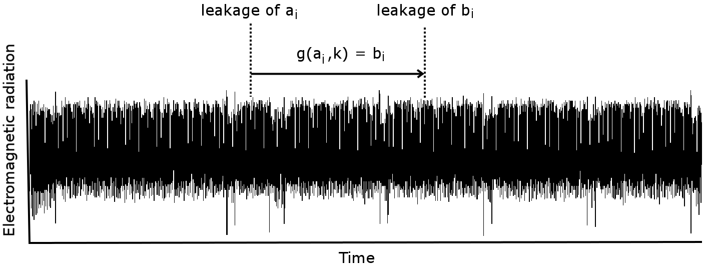


The idea



Partial trace of a cryptographic algorithm

The idea

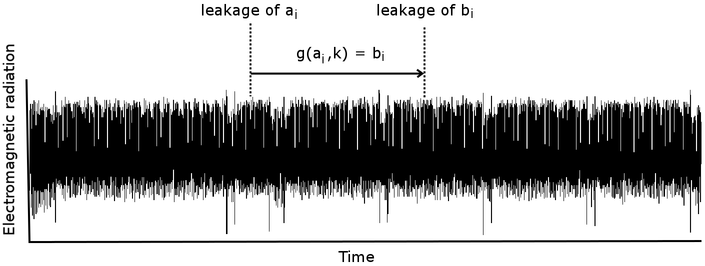


Partial trace of a cryptographic algorithm

Remarks:

- (a_i) and (b_i) have not independent distributions.
  Example: the couple (a, b) with $b = \mathtt{SB}(a)$ has impossible values.

The idea

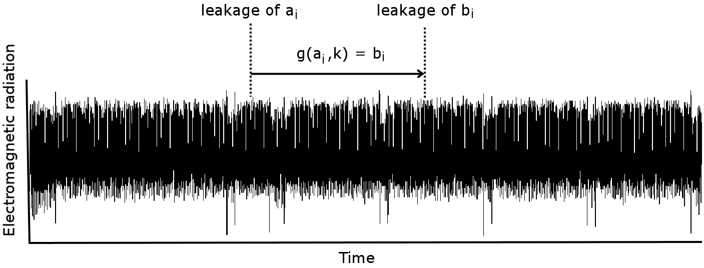


Partial trace of a cryptographic algorithm

Remarks:

- (a_i) and (b_i) have not independent distributions.
- (a_i) and (b_i) have a joint distribution that could depend on some key bits.
  Example: the couples (a, b_1) with $b_1 = \mathtt{SB}(a)$ and (a, b_2) with $b_2 = \mathtt{SB}(a \oplus \mathtt{0xff})$ have different distributions.

The idea



Partial trace of a cryptographic algorithm

Remarks:

- (a_i) and (b_i) have not independent distributions.
- (a_i) and (b_i) have a joint distribution that could depend on some key bits.

$\Rightarrow$ Choice of a targeted function.
Example: $g(a, k) = \mathtt{SB}(a \oplus k)$

The attack principle

- Acquisitions of couples (leakage of a_i , leakage of b_i).
  $\Rightarrow$ Empirical distribution S_d.

- Precomputations of theoretical distributions $S(g, k)$ of $(a_i, g(a_i, k))$ for each possible key k.

- Comparison of S_d to each $S(g, k)$.
  $\Rightarrow$ The nearest determines the correct key value.

The attack principle

- Acquisitions of couples (leakage of a_i , leakage of b_i).
  $\Rightarrow$ Empirical distribution S_d.

- Precomputations of theoretical distributions $S(g, k)$ of $(a_i, g(a_i, k))$ for each possible key k.

- Comparison of S_d to each $S(g, k)$.
  $\Rightarrow$ The nearest determines the correct key value.

The problems

- How compare two distributions from an exact value a_i and the corresponding leakage $\varphi(a_i)$?

The problems

- How compare two distributions from an exact value a_i and the corresponding leakage $\varphi(a_i)$?

- How compare two distributions: a theoretical one (exact) and an empirical one (approximate)?

The problems

- How compare two distributions from an exact value a_i and the corresponding leakage $\varphi(a_i)$?

- How compare two distributions: a theoretical one (exact) and an empirical one (approximate)?

- How find the two instants (points of interest), how synchronize the signals, ...?

The problems

- How compare two distributions from an exact value a_i and the corresponding leakage $\varphi(a_i)$?

- How compare two distributions: a theoretical one (exact) and an empirical one (approximate)?

- How find the two instants (points of interest), how synchronize the signals, ...?

Compare an exact value to a leakage one

- Intermediate data a_i and b_i are reduced to a leakage model $\varphi(a_i)$ and $\varphi(b_i)$ (Hamming weigth, identity,...)

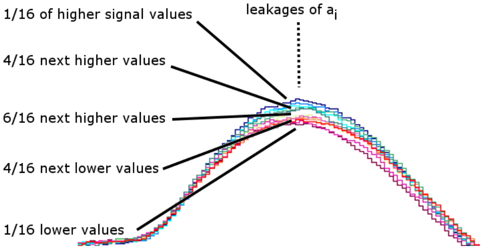
Compare an exact value to a leakage one

- Intermediate data a_i and b_i are reduced to a leakage model $\varphi(a_i)$ and $\varphi(b_i)$ (Hamming weigth, identity,...)

- Signal amplitudes are mapped to this leakage model too. Leakage estimation

Example: Classification method for a Hamming weight model of 4 bits:

HW value	number
0	1
1	4
2	6
3	4
4	1
Total	16



Compare two distributions

Notations:

- p_{ij} is the probability $\varphi(a) = i$ and $\varphi(g(a,k)) = j$ theoretical
- f_{ij} is the frequency of couple $(\varphi(a), \varphi(g(a,k^\star))) = (i,j)$ empirical

Example the χ^2 distance:

$$\chi^2(S(g,k), S_d) = \sum_i \sum_j \delta(p_{ij}, f_{ij})$$

$$\delta(p_{ij}, f_{ij}) = \begin{cases} \frac{(p_{ij} - f_{ij})^2}{p_{ij}} & , p_{ij} \neq 0 \\ 0 & , p_{ij} = f_{ij} \\ \infty & , p_{ij} = 0 \neq f_{ij} \end{cases}$$

$\Rightarrow$ The smallest distance between S_d and all the $S(g,k)$ reveals the correct key k.

But...

- Infinite distances when $p_{ij} = 0$ and $f_{ij} \neq 0$

$\Rightarrow$Instability in presence of errors.

But...

- Infinite distances when $p_{ij} = 0$ and $f_{ij} \neq 0$

  $\Rightarrow$Instability in presence of errors.

Solution: Others distances from the paper:

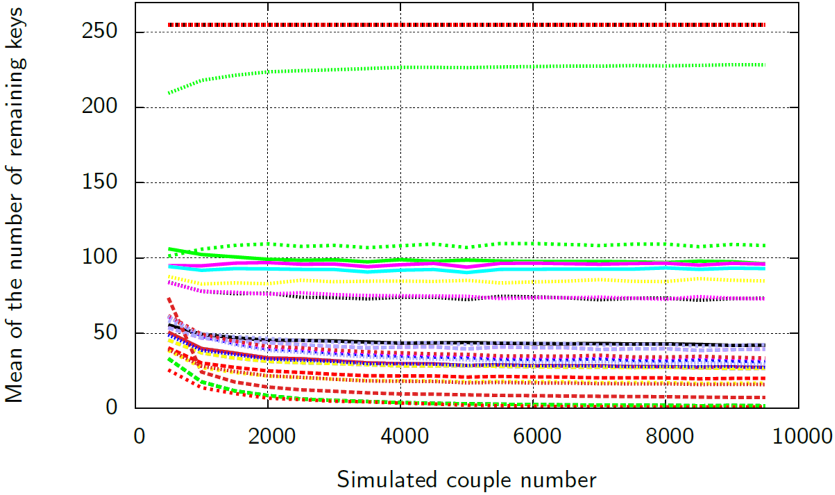
S.-H. Cha. Comprehensive survey on distance/similarity measures between probability density functions. *International Journal of Mathematical Models and Methods in Applied Sciences*, 2007.

- Classical distances over $\mathbb{R}^n$
- Distances based on scalar product
- Distances based on Shannon entropy
- ...

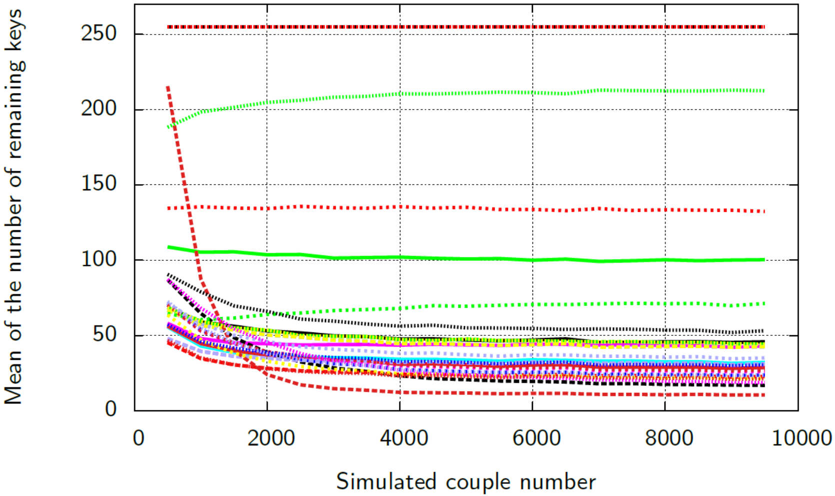
Simulations

- 100,000 simuled attacks
- Targeted function: $g(a, k) = \mathtt{SB}(a \oplus k)$
- Leakage model: Hamming weight of 8 bits
- Two kinds of error for the leakage estimation:
  - *small* errors : correct value ± 1
  - random errors : random value
- Chosen distance : 33 different distances

Simulations for different distances and 50% *small* errors



Simulations for different distances and 50% random errors



Best distances

- Pearson χ^2 distance: $\sum_i \sum_j \frac{(p_{ij}-f_{ij})^2}{f_{ij}}$
- Product scalar distance: $1 - \sum_i \sum_j p_{ij} \cdot f_{ij}$
- Kullback-Leiber distance: $\sum_i \sum_j p_{ij} \cdot ln(\frac{p_{ij}}{f_{ij}})$
- Harmonic mean distance: $1 - 2\sum_i \sum_j \frac{p_{ij} \cdot f_{ij}}{p_{ij}+f_{ij}}$

$\Rightarrow$With these distances the attack succeeds even in presence of errors.

Best distances

- Pearson χ^2 distance: $\sum_i \sum_j \frac{(p_{ij} - f_{ij})^2}{f_{ij}}$
- Product scalar distance: $1 - \sum_i \sum_j p_{ij} \cdot f_{ij}$
- Kullback-Leiber distance: $\sum_i \sum_j p_{ij} \cdot ln(\frac{p_{ij}}{f_{ij}})$
- Harmonic mean distance: $1 - 2\sum_i \sum_j \frac{p_{ij} \cdot f_{ij}}{p_{ij} + f_{ij}}$

$\Rightarrow$With these distances the attack succeeds even in presence of errors.

$\Rightarrow$The estimation may be approximative. No profiling phase is needed.

ATMega2561 : experimental conditions

- First round of a software AES-128
- Targeted function: $g(a, k) = \mathtt{SB}(a \oplus k)$
- Selection of the points of interest thanks to the variance
- Hamming weigth estimation by classification
- Chosen distance: Scalar product

ATMega2561: attack and results

- The attack is repeated on each pair of points of interest
- The first 16 results with the smaller distance

ATMega2561: attack and results

- The attack is repeated on each pair of points of interest
- The first 16 results with the smaller distance
- 4×4 instants with the higher variance:
  - The top 16 reveals 3 key bytes
  - No position information for these bytes: it remains $\approx 2^{107}$ keys to test
  - The probability for randomly finding 3 bytes is less than 2^{-24}
  - Time < 1 second

ATMega2561: attack and results

- The attack is repeated on each pair of points of interest
- The first 16 results with the smaller distance
- 4×4 instants with the higher variance:
  - The top 16 reveals 3 key bytes
  - No position information for these bytes: it remains $\approx 2^{107}$ keys to test
  - The probability for randomly finding 3 bytes is less than 2^{-24}
  - Time $<$ 1 second
- 50×50 instants with the higher variance:
  - The top 16 reveals 10 key bytes
  - No position information for these bytes: it remains $\approx 2^{70}$ keys to test
  - The probability for randomly finding 10 bytes is less than 2^{-80}
  - Time $<$ 2 minutes

Conclusion

- Without the knowledge of the plaintext or the ciphertext
- Many cryptographic functions
- Good stability in case of weak leakage estimation
- Easy and fast
- Difficulty for identifying of the position of the recovered key bytes

Perspectives

- Improve the attack thanks to the next rounds
- Apply this attack to protected implementations
- Try others methods to model and/or estimate the leakage
- Find others methods for points of interest detection without the knowledge of plaintext or ciphertext

LABORATOIRE D'ÉLECTRONIQUE
ET DE TECHNOLOGIES
DE L'INFORMATION

CEA-Leti
MINATEC Campus, 17 rue des Martyrs
38054 GRENOBLE Cedex 9
Tel. +33 4 38 78 36 25

www.leti.fr

Questions?

contact: linge.yanis@gmail.com

DPAContest V4: experimental conditions

- First round of a software AES-256 with a RSM countermeasure
- Traces with the same unknown offset i
- Targeted function: $g(a,k) = \mathtt{SB}(a \oplus k \oplus M_i) \oplus M_{i+1}$
- Selection of the points of interest thanks to the variance
- Hamming weigth estimation by classification
- Chosen distance: Scalar product

DPAContest V4: attack and results

- The attack is repeated on each pair of interest points
- Occurence number of the resulting key bytes
- Instants where the variance is 5 times the mean variance:
  - 28,000 points of interest
  - The top 16 for occurence numbers reveals 7 key bytes
  - These bytes are well-ordered: it remains $\approx 2^{92}$ keys to test
  - The probability for randomly finding these bytes is less than 2^{-40}
  - Time: 5 days