Evaluating the Security of Implementations Against Side Channel Attacks

Activities from ANSSI Laboratory for Embedded Security

Emmanuel Prouff
emmanuel.prouff@ssi.gouv.fr

Agence Nationale de la Sécurité des Systèmes d'Information

Summer School, Šibenik, Croatia – June, 17-21, 2019

ANSSI Core Missions

- Prevent threats by supporting the development of trusted products and services for Governmental entities and economic actors
- Provide reliable advice and support to Governmental entities and operators of Critical Infrastructure
- Keep companies and the general public informed about information security threats and the related means of protection through an active communication policy
- Give support to security evaluation labs (ITSEF) and to the french national certification center (CCN).

Certification Body

- Certification Body: 10 agents
- List of certified products available on the ANSSI website: www.ssi.gouv.fr
- Some statistics about french Common Criteria evaluations:
  - 50% smartcard evaluations
  - 35% microcontroller evaluations
  - 15% softwares, network, misc ...

Certification Labels for Security Products

- CSPN - certification for first level security *black-box*
  - fast and easy procedure (for ex. allow to label freewares)
  - evaluation made by ITSEFs
  - compliance with security target
  - efficiency of security functionalities
  - 25-35 man/day
- Common Criteria - CC certification *white-box*
  - longer procedure, recognized outside of France
  - evaluation made by ITSEFs
  - compliance with security target
  - eval. of each security functionality
  - different assurance levels: EAL1, ..., EAL7

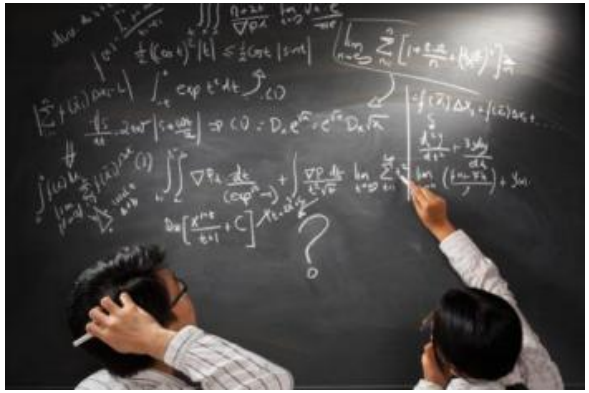
Security Evaluation in the Industry

Context

- Mandatory for some security products (e.g. banking cards, ePassport or secure platforms for embedded systems)
- Not always Mandatory but Economical Advantage for many others (e.g. USIM or access control)
- General Framework
  - Developers implement countermeasures against SoA attacks (passive, semi-invasive or invasive)
  - Independent Labs evaluate the security w.r.t SoA attacks (e.g. listed by the JHAS group)
  - Certification authorities (e.g. ANSSI or BSI or EMV-CO) validate the evaluation and deliver the certificates.

Security Evaluation in the Industry

Facts



- Attacks are each year more and more powerful
  - Semi-invasive attacks with multiple faults
  - Template Attacks, Second-Order SCA or Horizontal SCA
  - Use of HPC
- ... each year more numerous ($\simeq$ 100 publications / year)
- Security is costly: development/testing time, decreasing of the performances, loss of genericity for the codes, expertise cost
- How to increase coverage and accuracy of the evaluation while decreasing the cost?

Security Evaluation in the Industry

Some needs...

- Automatize evaluations without quality loss
- Increase trust in evaluation results
  - Failure due to countermeasures or to evaluator weakness?
- Quantify the security instead of testing a set of attacks
  - Too many attacks, too many parametrizations, etc.
  - Need to always stay up-to-date
  - Failure with 10^6 measurements but what if 10^7 are available?
- Measure the information leakage
  - Portability gain
  - Allow for comparison between evaluations
- Identify Points of Interest
  - Exchange "experts How-To" for sound and repeatable techniques

To sum-up: Estimate the efficiency of the most powerful attacks in a minimum of time.

Security Evaluation in the Industry

Some needs...

- Automatize evaluations without quality loss
- Increase trust in evaluation results
  - Failure due to countermeasures or to evaluator weakness?
- Quantify the security instead of testing a set of attacks
  - Too many attacks, too many parametrizations, etc.
  - Need to always stay up-to-date
  - Failure with 10^6 measurements but what if 10^7 are available?
- Measure the information leakage
  - Portability gain
  - Allow for comparison between evaluations
- Identify Points of Interest
  - Exchange "experts How-To" for sound and repeatable techniques

To sum-up: Estimate the efficiency of the most powerful attacks in a minimum of time.

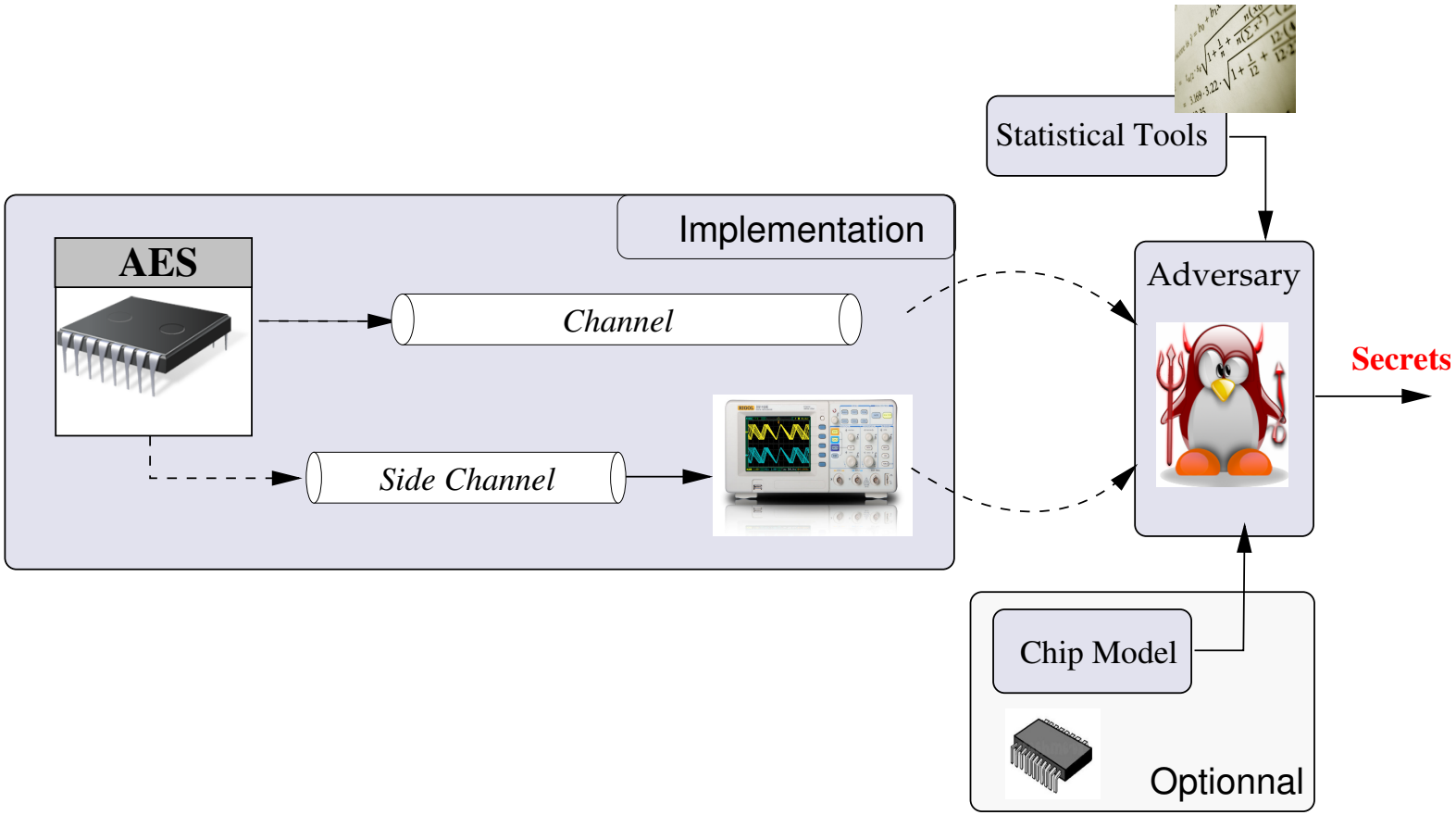
What we do at ANSSI related to these subjects?

Some Examples...

- Define generic frameworks to encompass most of the SoA attacks
- Use the latter frameworks to build generic and modular testing libraries
- Define methods to accurately measure the information leakage from a chip
- Define methods to evaluate the success rate of SoA attacks based on the latter measure
- Adapt methods from Machine Learning to identify PoI

Advanced Side Channel Attacks (DPA like attacks)

Side Channel Analysis: General Framework.



Advanced Side Channel Attacks

Side Channel Analysis: General Framework (Theoretical)

Context: attack during the manipulation of $S(X + k)$.

1. Measurement :
   - get a leakages sample $(\ell_{k,i})_i$ related to a sample $(x_i)_i$ of plaintexts.
2. Model Selection :
   - Design/Select a function $\mathbf{m}(\cdot)$.
3. Prediction :
   - For every $\hat{k}$, compute $m_{\hat{k},i} = \mathbf{m}(S(x_i + \hat{k}))$.
4. Distinguisher Selection :
   - Choose a statistical distinguisher Δ.
5. Key Discrimination :
   - For every $\hat{k}$, compute the distinguishing value $\Delta_{\hat{k}}$:

$$\Delta_{\hat{k}} = \Delta\left((\ell_{k,i})_i, (m_{\hat{k},i})_i\right) \quad .$$

6. Key Candidate Selection :
   - Deduce $\hat{k}$ from all the values $\Delta_{\hat{k}}$.

Advanced Side Channel Attacks

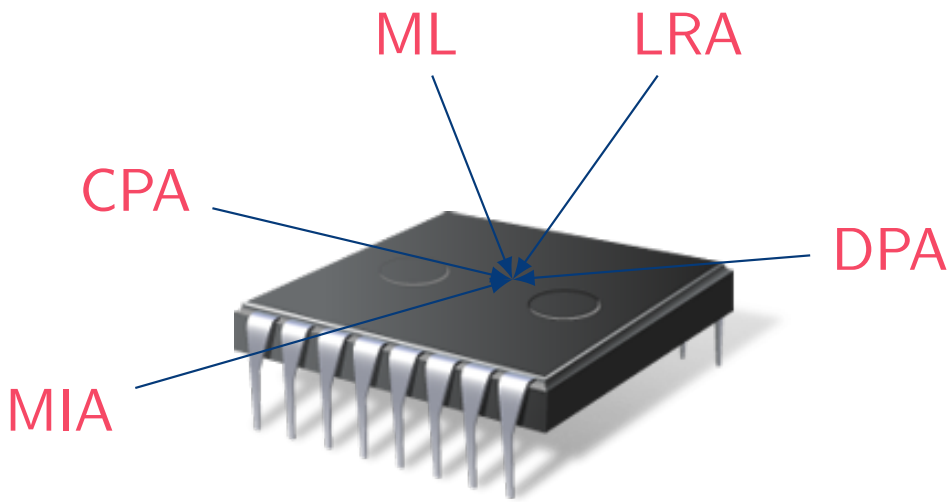
Side Channel Analysis: attack Description Sheet

Attack Description Sheet

Type of Leakage: *e.g. power consumption or electromagnetic emanation*
Model Function: *e.g. one bit of Z or its Hamming weight*
Statistical Distinguisher: *e.g. difference of means, correlation or entropy*
Key Candidate Selection: *e.g. the candidate the maximizes the scores*

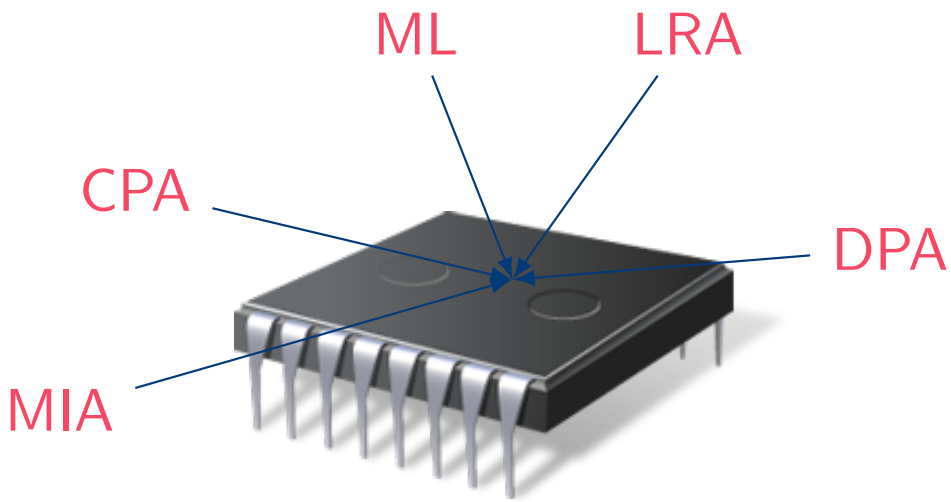
A designer/evaluator POV

- Security of a device against SCA is tested by designers/evaluators.
- Large set of SCA to test: CPA, MIA, LRA, DPA, ML, etc.
- Little time, limited means, constrained resources.
- Strong knowledge of my device.



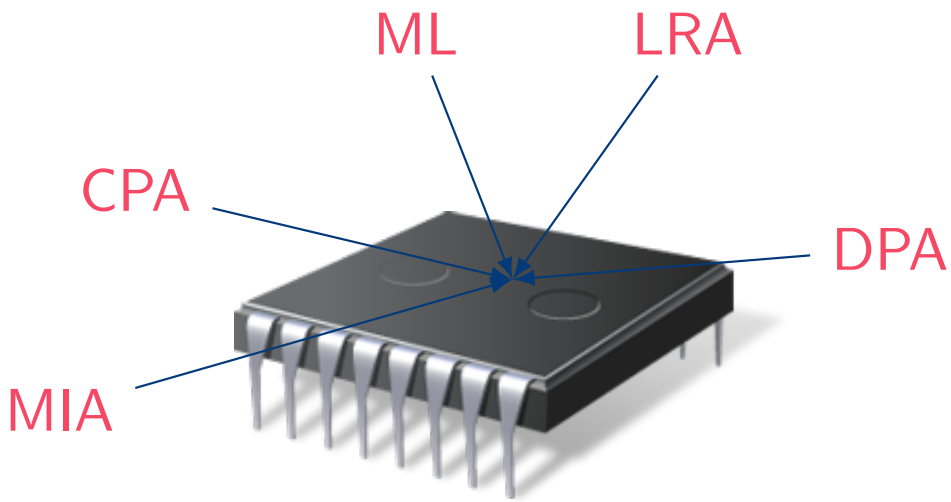
A designer/evaluator POV

- Security of a device against SCA is tested by designers/evaluators.
- Large set of SCA to test: CPA, MIA, LRA, DPA, ML, etc.
- Little time, limited means, constrained resources.
- Strong knowledge of my device.



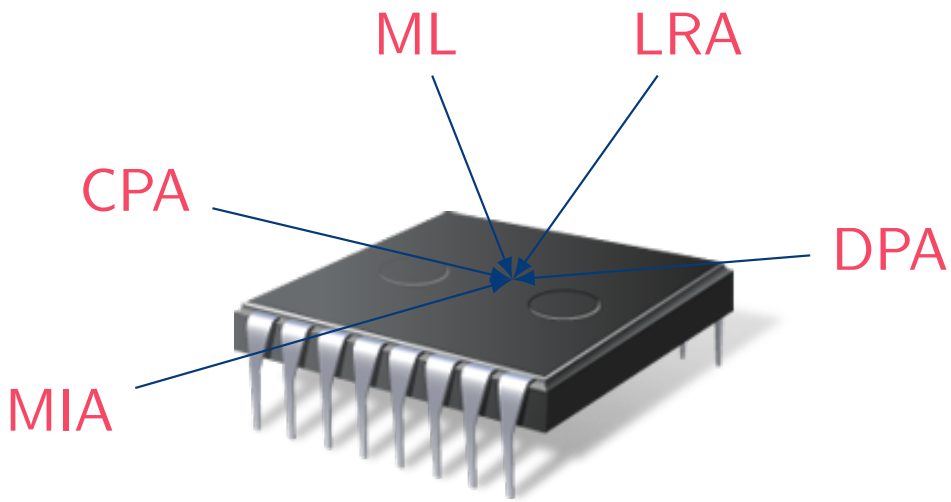
A designer/evaluator POV

- Security of a device against SCA is tested by designers/evaluators.
- Large set of SCA to test: CPA, MIA, LRA, DPA, ML, etc.
- Little time, limited means, constrained resources.
- Strong knowledge of my device.



A designer/evaluator POV

- Security of a device against SCA is tested by designers/evaluators.
- Large set of SCA to test: CPA, MIA, LRA, DPA, ML, etc.
- Little time, limited means, constrained resources.
- Strong knowledge of my device.



Leakage assessment: is there information in the traces?

- must be very efficient in the number of traces.
- must be as generic as possible: any kind information must be revealed.
  - ↪ independent from leakage functions.
  - ↪ takes into account as many intermediate variables as possible.

Intuitions

- First focus on first-order leakages, *i.e.* the information is contained in the conditional mean of the traces.

$$E[T \mid Z = z] \neq E[T]$$

- A secure implementation would behave as manipulating random values.

Test Vector Leakage Assessment (TVLA) [Becker *et al.* White Paper CRI]

Acquire some sets of traces:

- S_1: Plaintexts and Keys are both fixed to well chosen values.
- S_2: Plaintexts are randomly chosen and Keys are fixed.
- S_3: Plaintexts are fixed and Keys are randomly chosen.
- ...

Welch t-test

- between S_i and S_1 compute , for each time sample t,

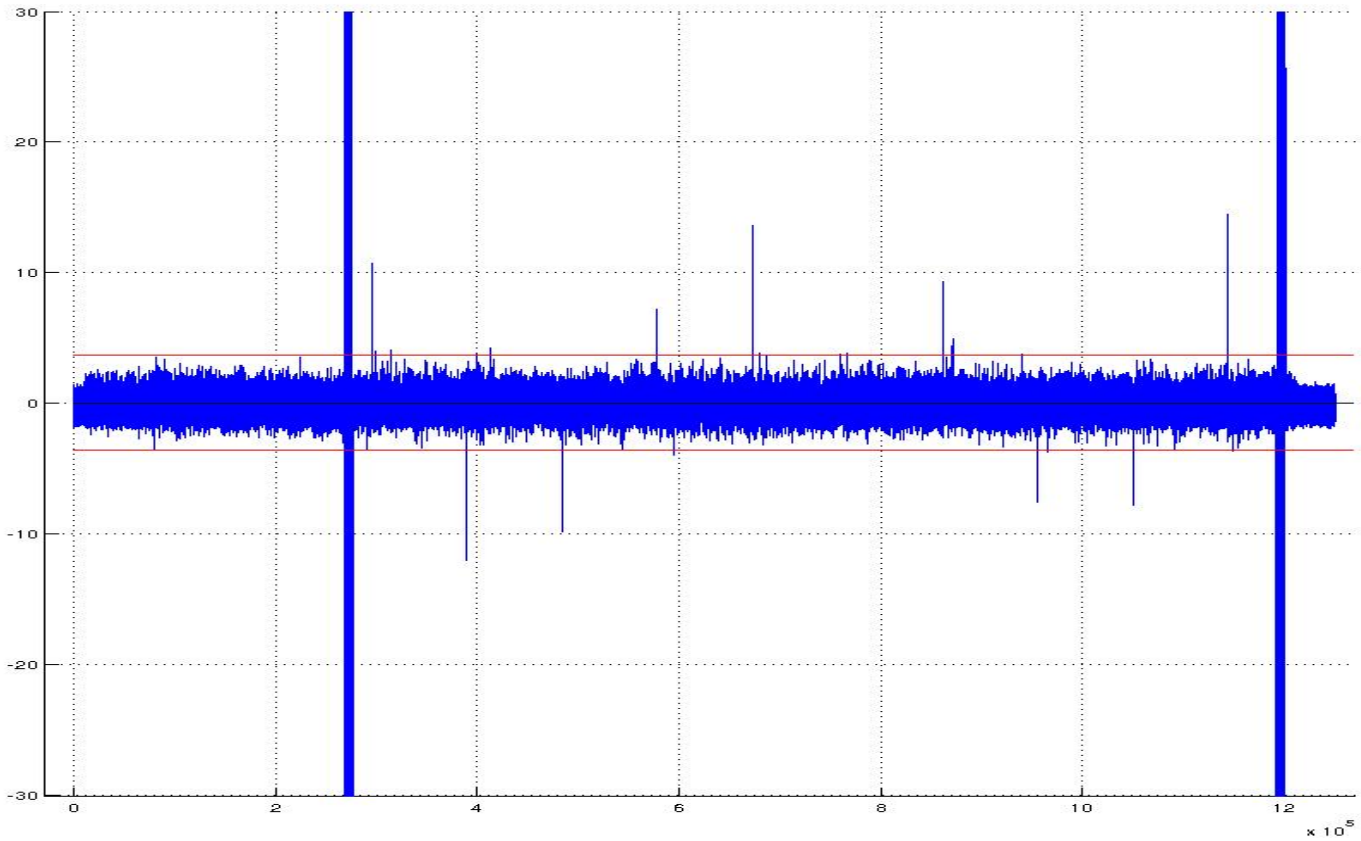
$$score(t) = \frac{\hat{E}[S_i] - \hat{E}[S_1]}{\sqrt{\left(\frac{\hat{V}[S_i]}{N_i} + \frac{\hat{V}[S_1]}{N_1}\right)}}$$

  where $\hat{E}$ and $\hat{V}$ are estimations of the mean and of the variance respectively.
- if $score(t) > \text{threshold}$ then there is a leakage...

An example on AES implem. (8-bit ATMega)

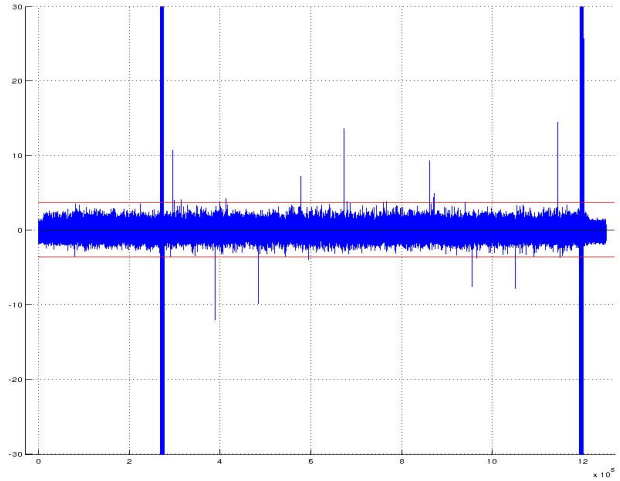
200000 observations, Random plaintexts vs. Fixed Set.

$\text{threshold} = 4.5 std(score) + mean.$



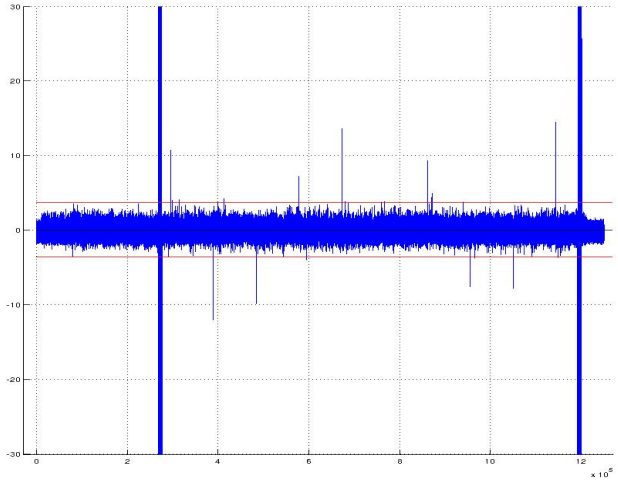
TVLA output...

There are first-order leakages!



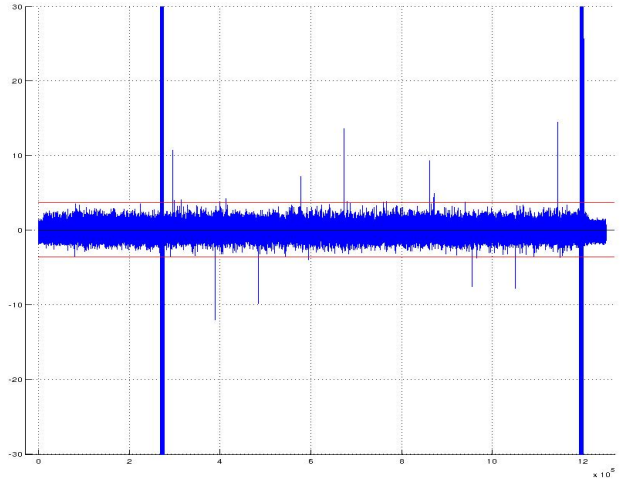
TVLA output...

There are first-order leakages!
But which sensitive value is leaking?



TVLA output...

There are first-order leakages!
But which sensitive value is leaking?



\+ There is some first-order leakages and their time samples.

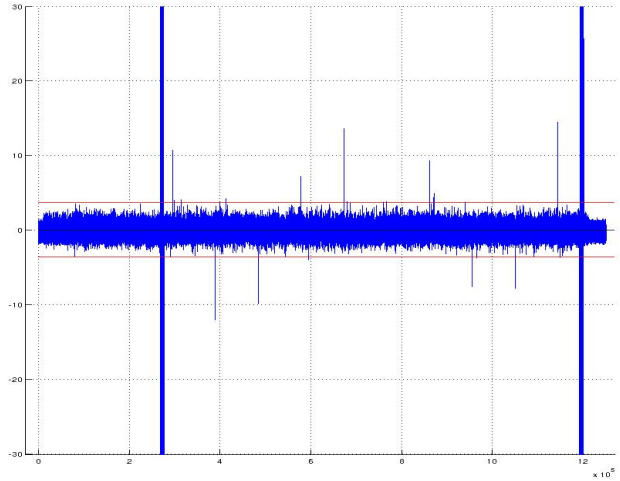
\- These leakages may not be sensitive... $\hookrightarrow$ use S_3

\- These leakages may depend on sensitive values in any ways:

- several bytes of plaintext/key may be involved.
- relationship between these leakages and intermediate variables may be tricky.

TVLA output...

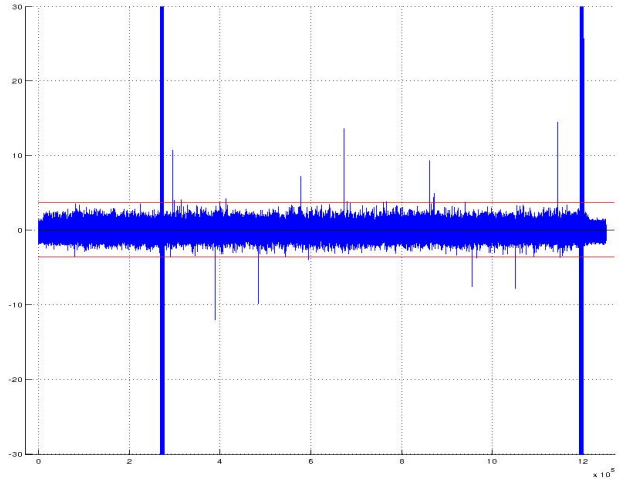
There are first-order leakages!
But which sensitive value is leaking?



+ There is some first-order leakages and their time samples.
- These leakages may not be sensitive... $\hookrightarrow$ use S_3
- These leakages may depend on sensitive values in any ways:
  - several bytes of plaintext/key may be involved.
  - relationship between these leakages and intermediate variables may be tricky.

TVLA output...

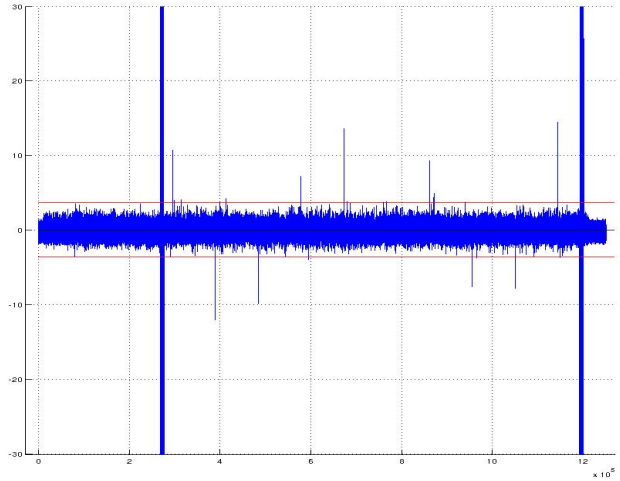
There are first-order leakages!
But which sensitive value is leaking?



+ There is some first-order leakages and their time samples.
- These leakages may not be sensitive... $\hookrightarrow$ use S_3
- These leakages may depend on sensitive values in any ways:
  - several bytes of plaintext/key may be involved.
  - relationship between these leakages and intermediate variables may be tricky.

TVLA output...

There are first-order leakages!
But which sensitive value is leaking?



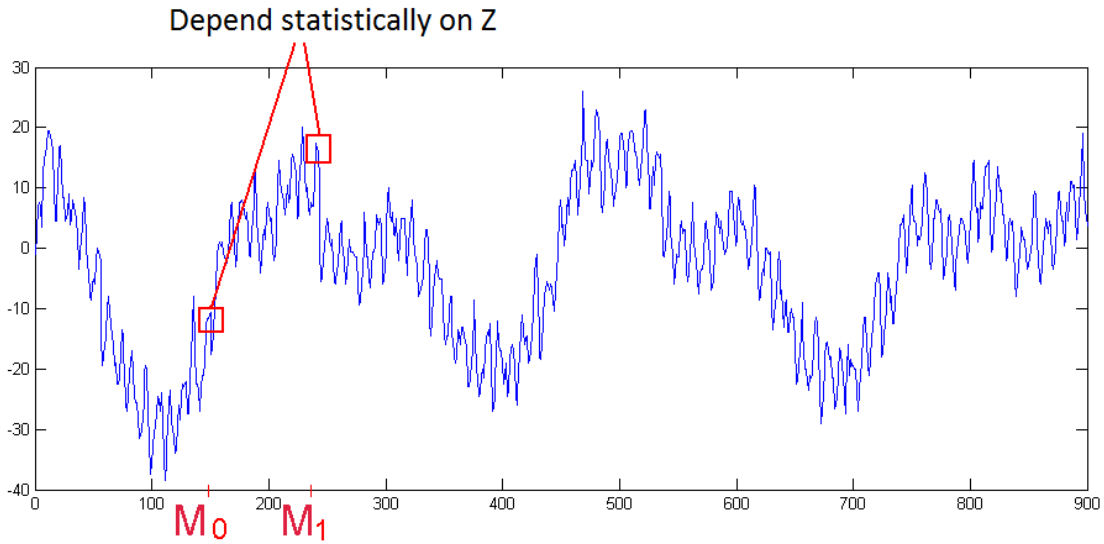
Find strategies to identify the plaintext/key bytes involved

- minimize the number of subsequent acquisition campaigns.
- use generic tools to observe leakages: T-test, SNR, etc...

Higher Order Side Channel Attacks

Core Principle

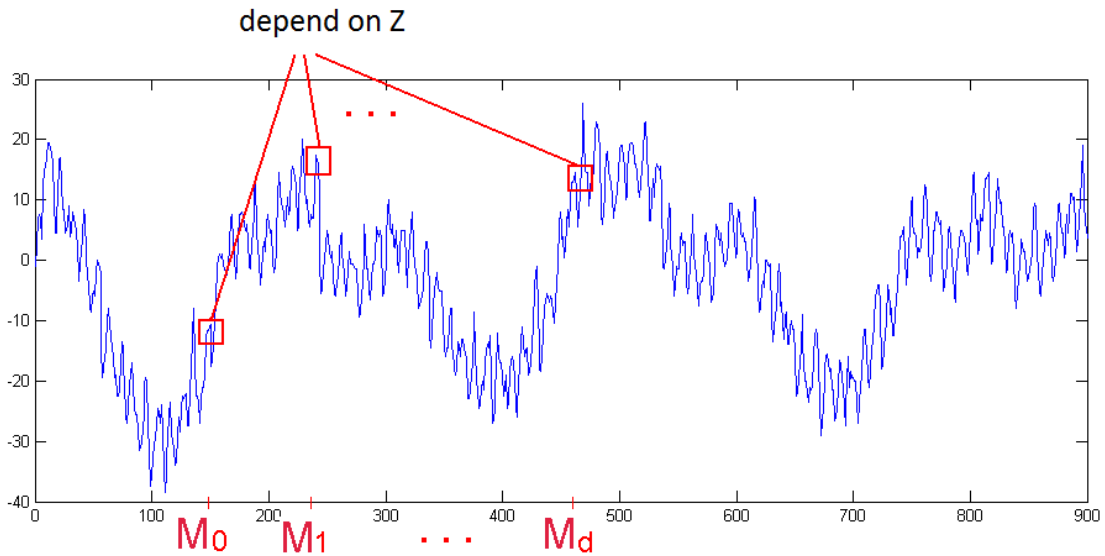
- First Order Masking: $M_0 = Z \oplus M_1$
- $\Longrightarrow$ Second Order SCA:



Higher Order Side Channel Attacks

Core Principle

- Masking of order d: $M_0 = Z \oplus M_1 \oplus \cdots \oplus M_d$
- Attack of order $d + 1$:



Higher-Order SCA

Context: **attack during the manipulation of $s_0, s_1, \cdots, s_d$ with $S(X + k) = \sum_{i=0}^{d} m_i$.**

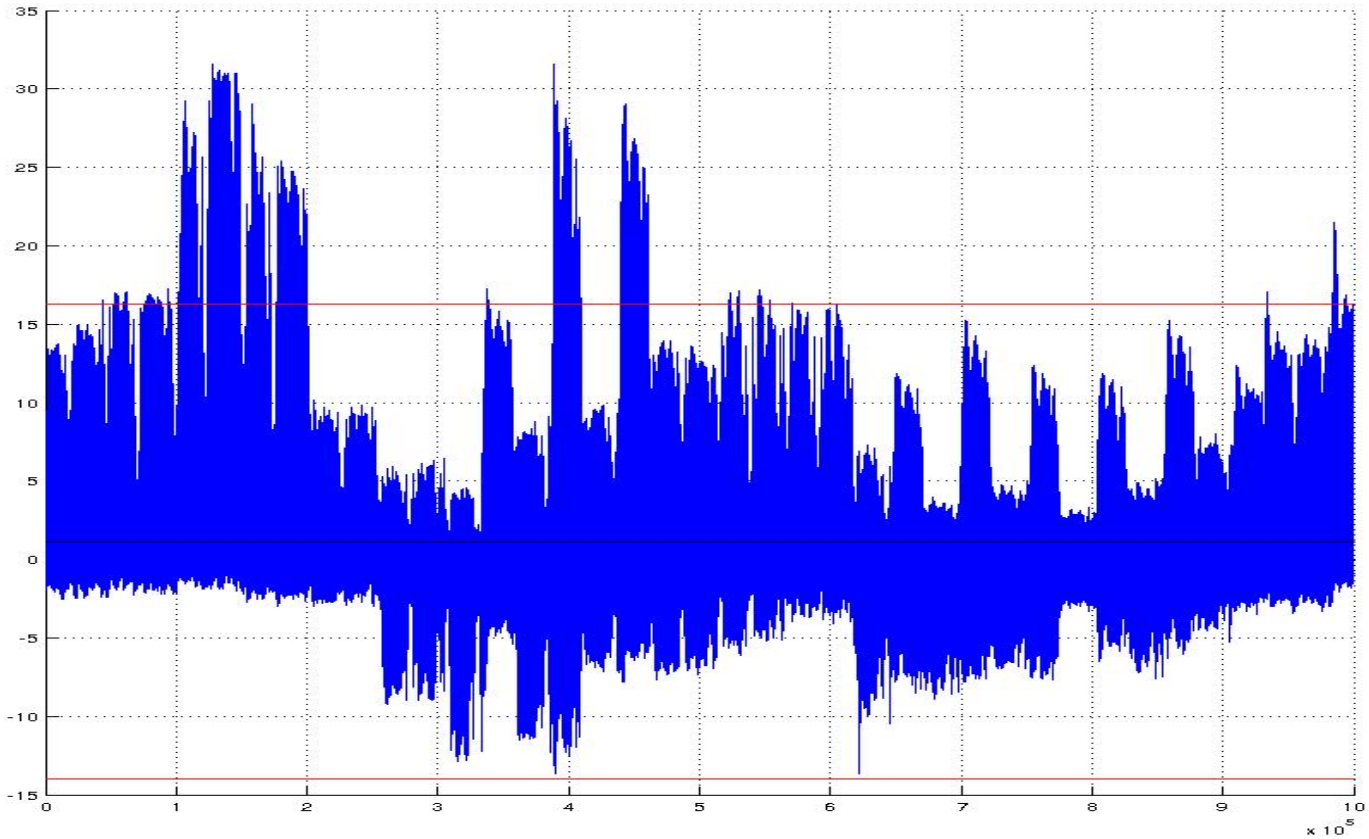
1. Measurement :
   - get a leakages sample $(\vec{\ell}_{k,i})_i$ related to a sample $(x_i)_i$ of plaintexts.
2. Pre-processing and Model Selection :
   - Select a combination function $\mathbf{f}(\vec{\ell})$, Design/Select a function $\mathbf{m}(s)$.
3. Prediction :
   - For every $\hat{k}$, compute $m_{\hat{k},i} = \mathbf{m}(S(x_i + \hat{k}))$.
4. Distinguisher Selection :
   - Choose a statistical distinguisher Δ.
5. Key Discrimination :
   - For every $\hat{k}$, compute the distinguishing value $\Delta_{\hat{k}}$:

$$\Delta_{\hat{k}} = \Delta\left((f(\vec{\ell}_{k,i}))_i, (m_{\hat{k},i})_i\right) \quad .$$

6. Key Candidate Selection :
   - Deduce $\hat{k}$ from all the values $\Delta_{\hat{k}}$.

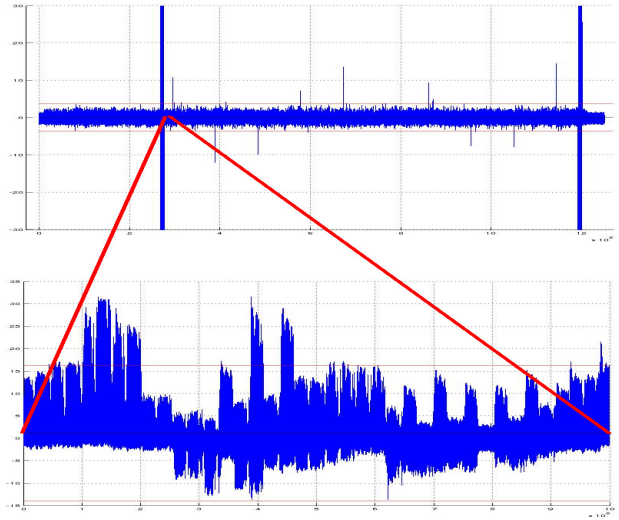
An example on AES implem. (8-bit ATMega)

200000 observations, Random plaintexts vs. Fixed Set.
Combination function: Centered product.



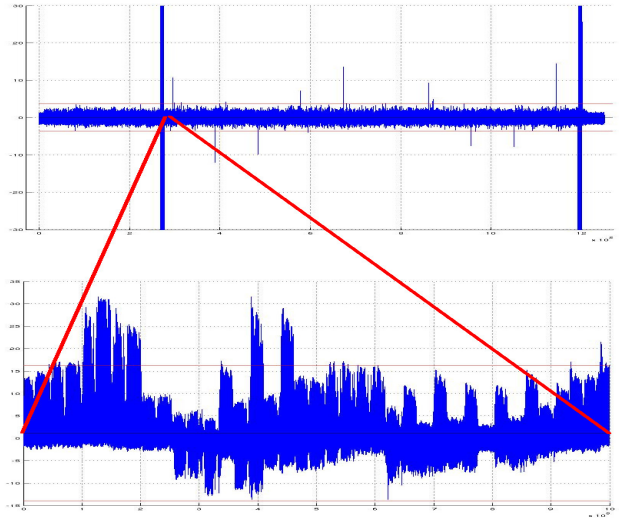
TVLA 2nd-order output...

There are second-order leakages!



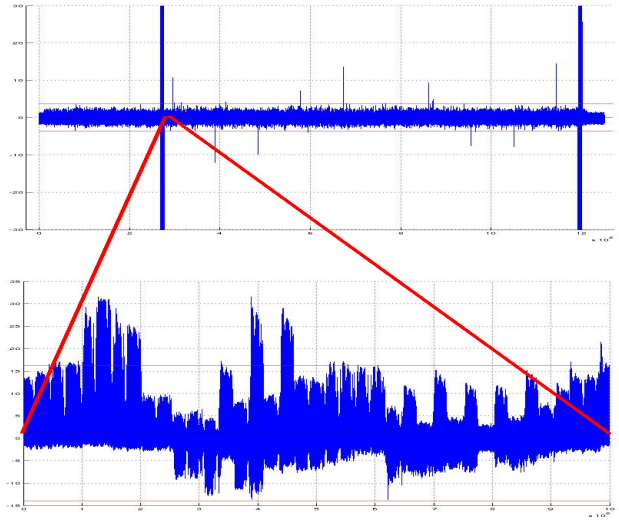
TVLA 2nd-order output...

There are second-order leakages!
The trace sizes are squared...



TVLA 2nd-order output...

There are second-order leakages!
The trace sizes are squared...



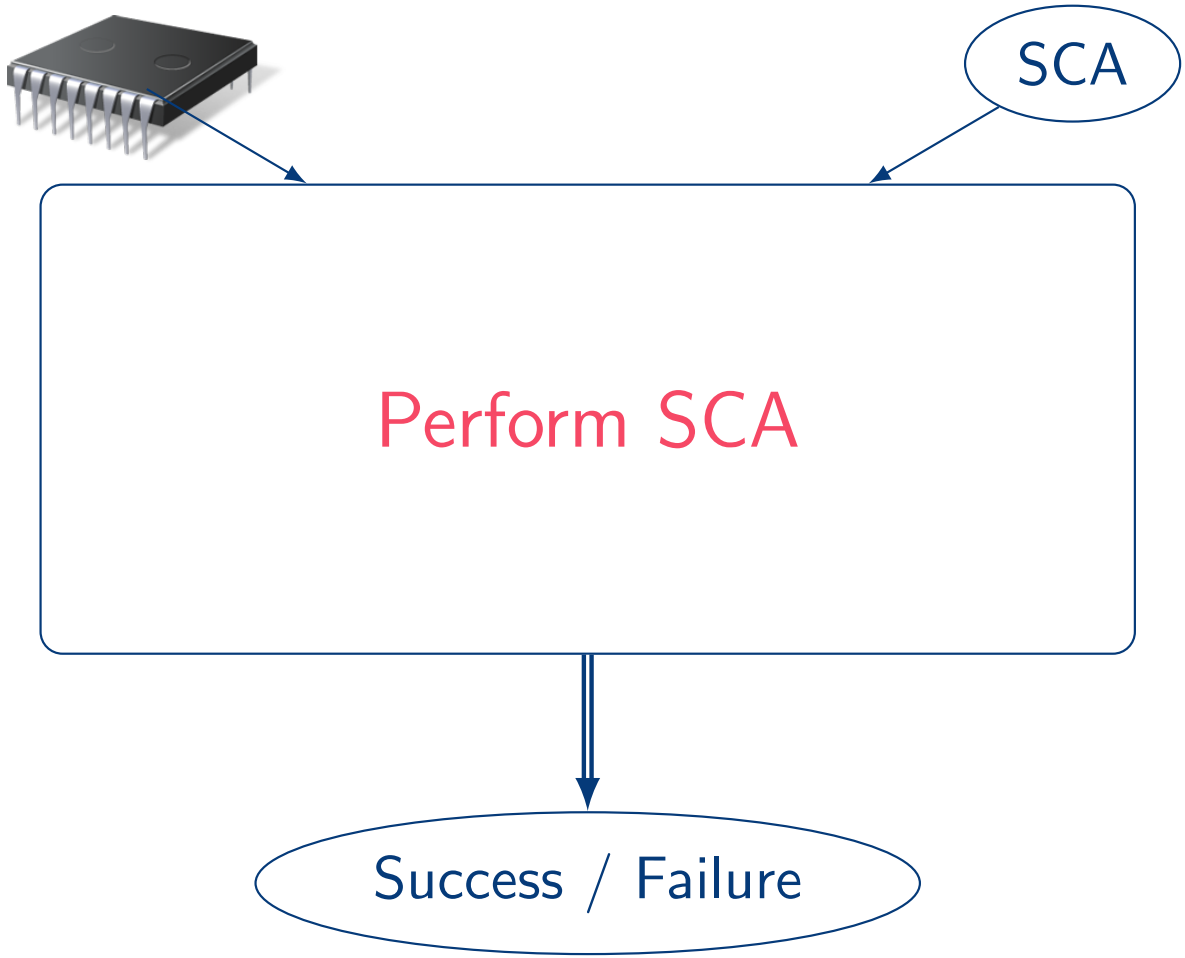
+ The centred product + TVLA allow to identify second order leakages.
- The treatment complexity increases exponentially with the order.
- The number of traces increases exponentially with the order.

Open Issues with TVLA

- Difficulty to deal with false positives: how to specify efficient and smart acquisition campaigns?
- Adversary may be considered too strong for some contexts: e.g. knowledge of the masking material, ability to profile the device, etc. Is it an issue? How to deal with it?
- Relation between SNR peaks amplitude and attacks efficiency.

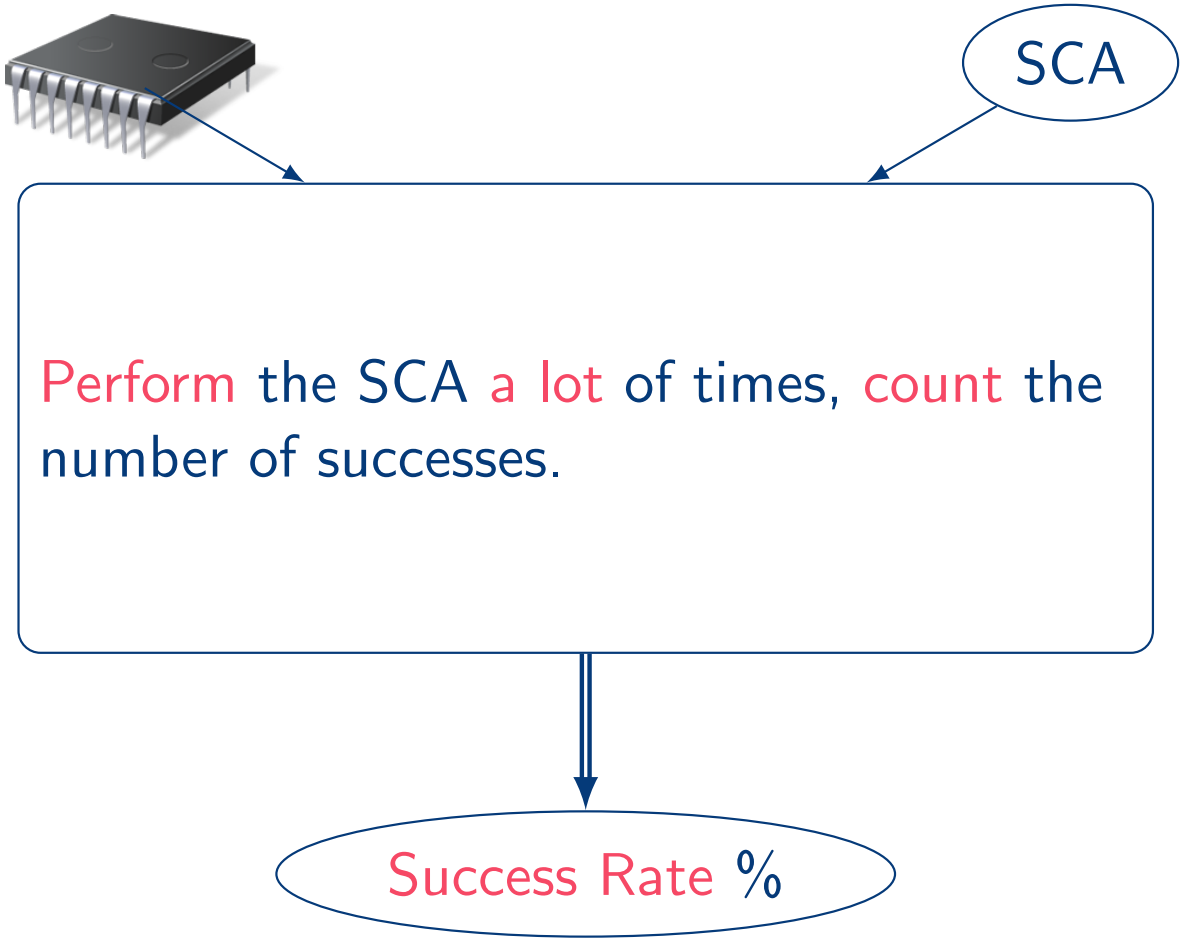
Security of a device: practice

Problem: Is my device secure against an attack ?



Security of a device: much better

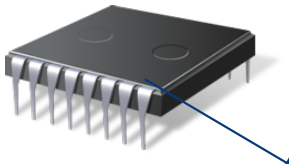
Problem: Is my device secure against an attack ?



Issue: Might be too expensive (acquisitions, computations ...).

Security of a device: much better

Problem: Is my device secure against an attack ?



SCA

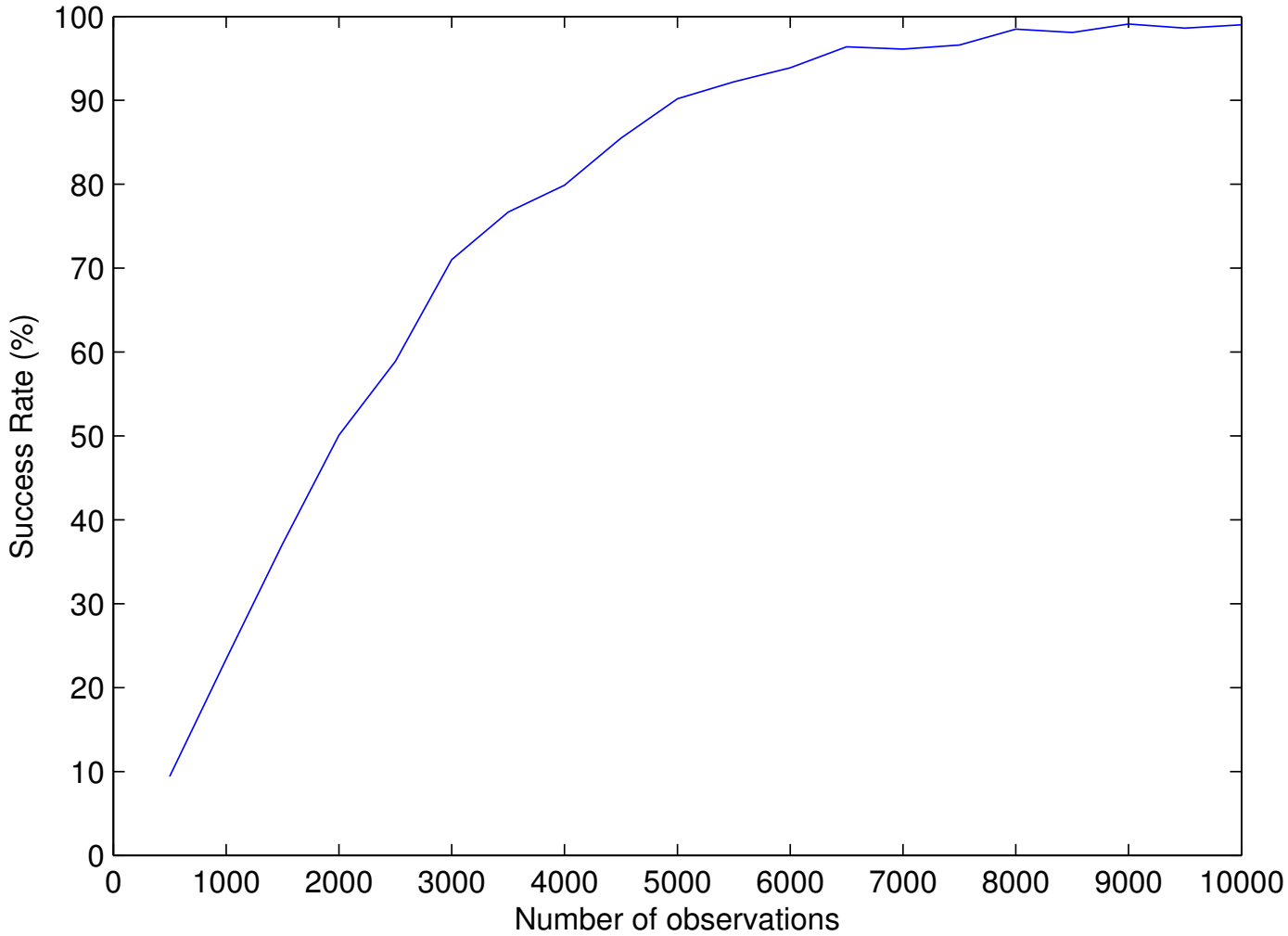
Perform the SCA a lot of times, count the number of successes.

Success Rate %

Issue: Might be too expensive (acquisitions, computations ...).

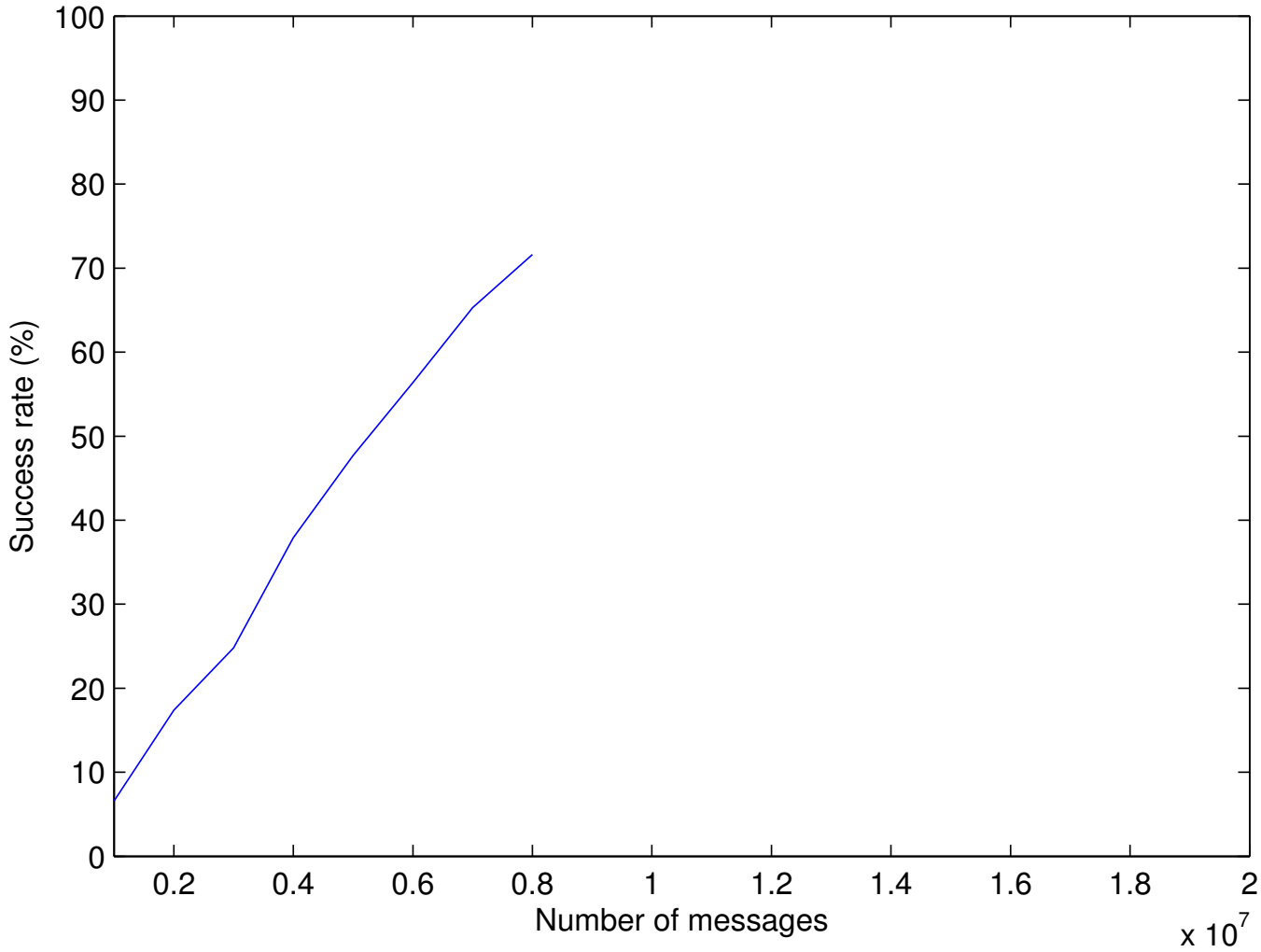
An example : 2O-CPA

$1000 \times 10000 = 10$ Millions of observations.



An example : 4O-CPA

$1000 \times 10^7 = 10$ Billions of observations.



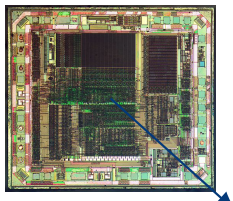
Knowledge of the device

- The designer/evaluator has a strong knowledge of the device:
  - Leakage functions.
  - Noises distributions.
  - ...
- Total control over inputs:
  - Plaintext.
  - Key.
  - Randoms.

credit:ZeptoBars

Security of a device: designer

Problem: Is my device secure against an attack ?



Costly SCA

Methodology

Success Rate %

PCD Methodology

Outlines:

1. Profile the device parameters.
2. Compute some formulas using these parameters.
3. Deduce the success rate.

Purpose:

- Work for any additive distinguisher st. DPA, CPA, Maximum Likelihood, etc.
- Generalize to HO versions.
- Enable to clearly identify the impact of each device's parameter on its security.

PCD Methodology

Outlines:

1. Profile the device parameters.
2. Compute some formulas using these parameters.
3. Deduce the success rate.

Purpose:

- Work for any additive distinguisher st. DPA, CPA, Maximum Likelihood, etc.
- Generalize to HO versions.
- Enable to clearly identify the impact of each device's parameter on its security.

PCD Methodology

Outlines:

1. Profile the device parameters.
2. Compute some formulas using these parameters.
3. Deduce the success rate.

Purpose:

- Work for any additive distinguisher st. DPA, CPA, Maximum Likelihood, etc.
- Generalize to HO versions.
- Enable to clearly identify the impact of each device's parameter on its security.

Distribution of the Scores Vector

Score vector

Vector of scores given by the SCA (when targeting 8-bit secrets)

$$\vec{d} = (d_{k_0}, d_{k_1}, \cdots, d_{k_{255}})$$

Comparison vector

Vector of differences of scores between $k^\star$ and k.

$$\vec{c} = (d_{k^\star} - d_{k_0}, d_{k^\star} - d_{k_1}, \cdots, d_{k^\star} - d_{k_{255}})$$

Attack success $\Leftrightarrow \vec{c} > \vec{0}$.

$$SR = P[\vec{c} > \vec{0}].$$

Distribution of the Scores Vector

Score vector

Vector of scores given by the SCA (when targeting 8-bit secrets)

$$\vec{d} = (d_{k_0}, d_{k_1}, \cdots, d_{k_{255}})$$

Comparison vector

Vector of differences of scores between $k^\star$ and k.

$$\vec{c} = (d_{k^\star} - d_{k_0}, d_{k^\star} - d_{k_1}, \cdots, d_{k^\star} - d_{k_{255}})$$

Attack success $\Leftrightarrow \vec{c} > \vec{0}$.

$$SR = P[\vec{c} > \vec{0}].$$

Distribution of the Scores Vector

Distribution of score vector

- # observations $\to \infty \implies \vec{d} \sim \mathcal{N}(m_d, \Sigma_d)$ (mCLT).
- m_d, Σ_d can be computed/deduced from leakage profiling
  - even when masking material is unknown(by averaging the combining of leakage points)!

Distribution of comparison vector

- # observations$\infty \implies \vec{c} \sim \mathcal{N}(m_c, \Sigma_c)$ (mCLT).
- m_c, Σ_c can be computed/deduced from leakage profiling

$$SR = P[\vec{c} > \vec{0}] = \Phi_{m_c, \Sigma_c}(\vec{0}, \vec{\infty}).$$

Distribution of the Scores Vector

Distribution of score vector

- # observations $\to \infty \implies \vec{d} \sim \mathcal{N}(m_d, \Sigma_d)$ (mCLT).
- m_d, Σ_d can be computed/deduced from leakage profiling
  - even when masking material is unknown(by averaging the combining of leakage points)!

Distribution of comparison vector

- # observations$\infty \implies \vec{c} \sim \mathcal{N}(m_c, \Sigma_c)$ (mCLT).
- m_c, Σ_c can be computed/deduced from leakage profiling

$$SR = P[\vec{c} > \vec{0}] = \Phi_{m_c, \Sigma_c}(\vec{0}, \vec{\infty}).$$

Evaluation of SR: PCD Methodology

1. Profile the leakage of every share.
2. Compute the parameters m_d and Σ_d of the score vector.
3. Deduce the parameters m_c and Σ_c and evaluate the success rate thanks to the multivariate normal cdf.

Evaluation of SR: PCD Methodology

1. Profile the leakage of every share.
2. Compute the parameters m_d and Σ_d of the score vector.
3. Deduce the parameters m_c and Σ_c and evaluate the success rate thanks to the multivariate normal cdf.

Evaluation of SR: PCD Methodology

1. Profile the leakage of every share.
2. Compute the parameters m_d and Σ_d of the score vector.
3. Deduce the parameters m_c and Σ_c and evaluate the success rate thanks to the multivariate normal cdf.

Evaluation of SR: PCD Methodology

1. Profile the leakage of every share.
2. Compute the parameters m_d and Σ_d of the score vector.
3. Deduce the parameters m_c and Σ_c and evaluate the success rate thanks to the multivariate normal cdf.

Validation of the approach

Context

- Two 'real life' devices: 130nm and 350nm architectures.
- Masked AES, output of S-box.
- EM radiations.

Methodology

- Estimation of leakage parameters using linear regression techniques on 200.000 samples.
- HO-CPAs using normalized product combination function, and `HW` model function.

Validation of the approach

Context

- Two 'real life' devices: 130nm and 350nm architectures.
- Masked AES, output of S-box.
- EM radiations.

Methodology

- Estimation of leakage parameters using linear regression techniques on 200.000 samples.
- HO-CPAs using normalized product combination function, and `HW` model function.

Results

Figure: 130nm, 2OCPA

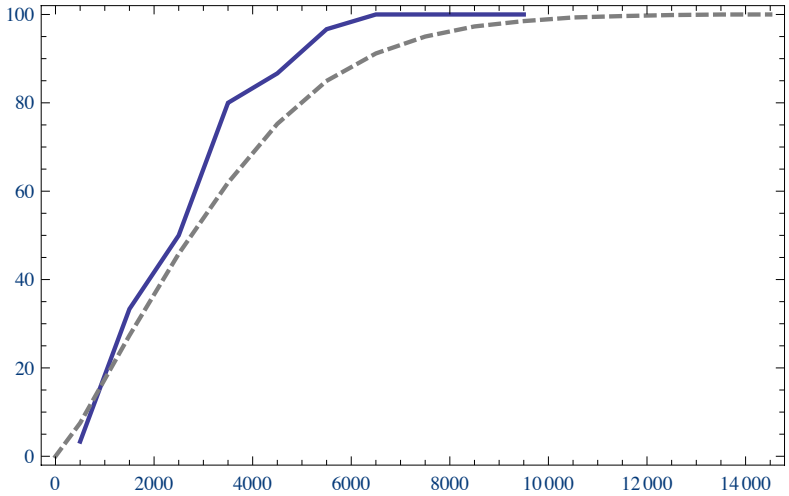


Figure: 350nm, 2OCPA

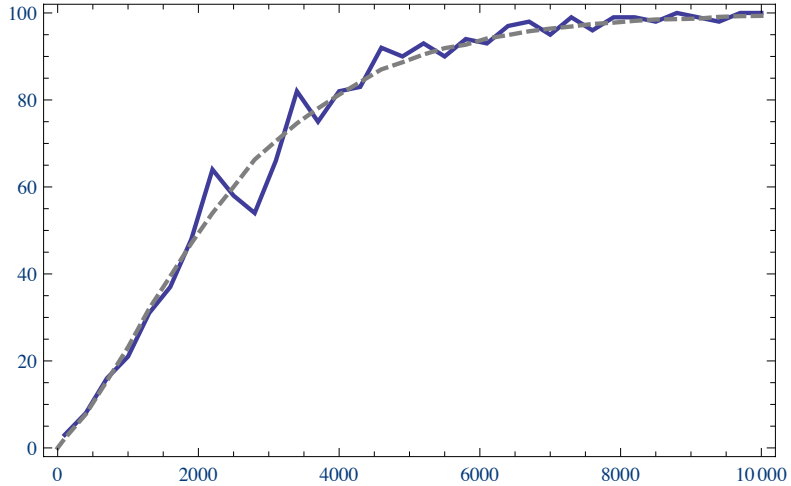


Figure: 350 nm, 3OCPA

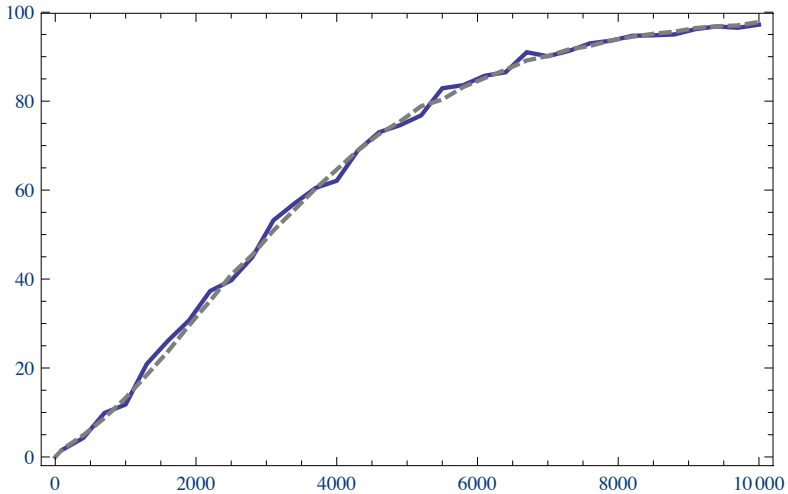
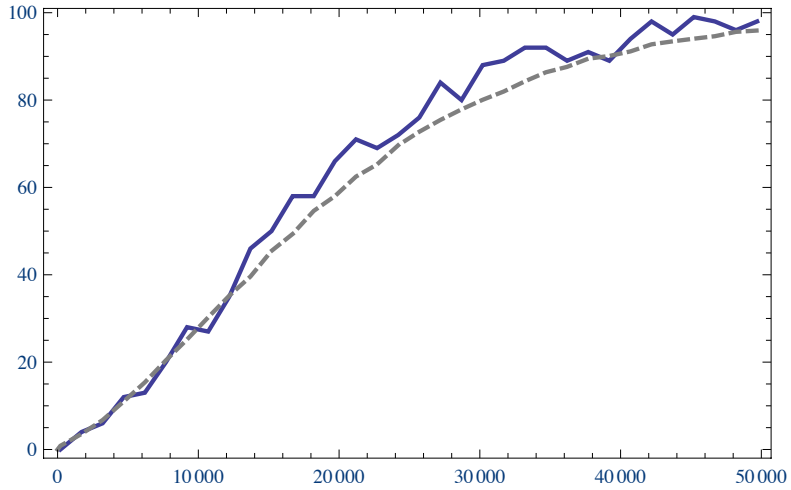


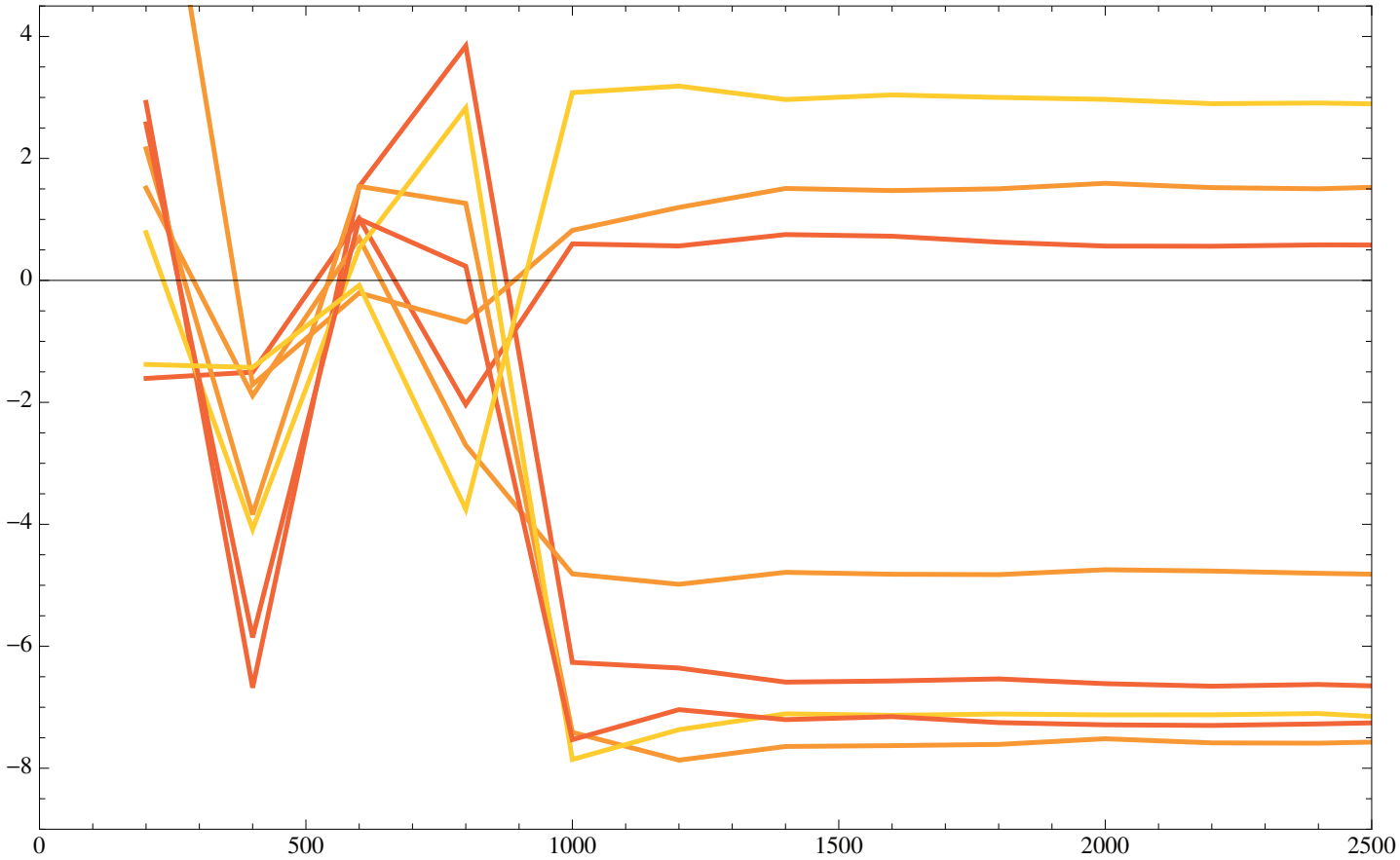
Figure: 350 nm, 4OCPA



Impact of leakage profiling

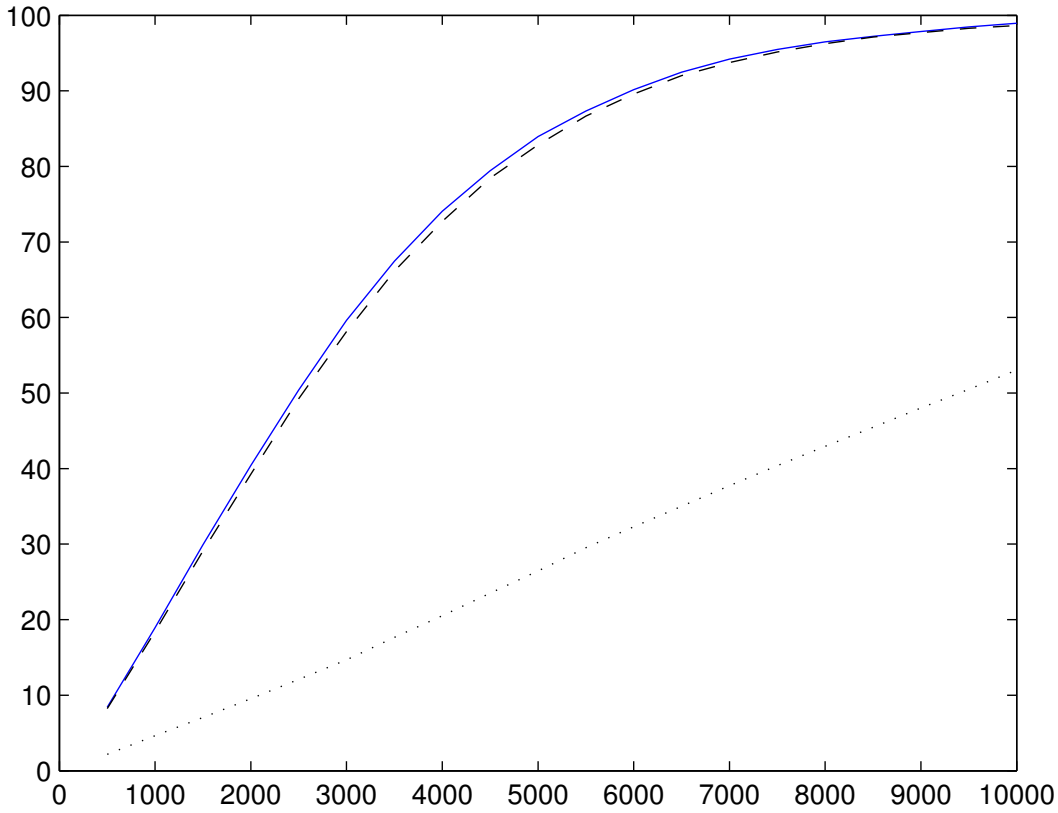
What happens when we regress/profile with less samples ?

$$Leak(X) = c_0 + c_1 X_1 + c_2 X_2 + c_3 X_3 + c_4 X_4 + c_5 X_5 + c_6 X_6 + c_7 X_7 + \text{Noise}$$



Results

Figure: 350 nm, 2OCPA



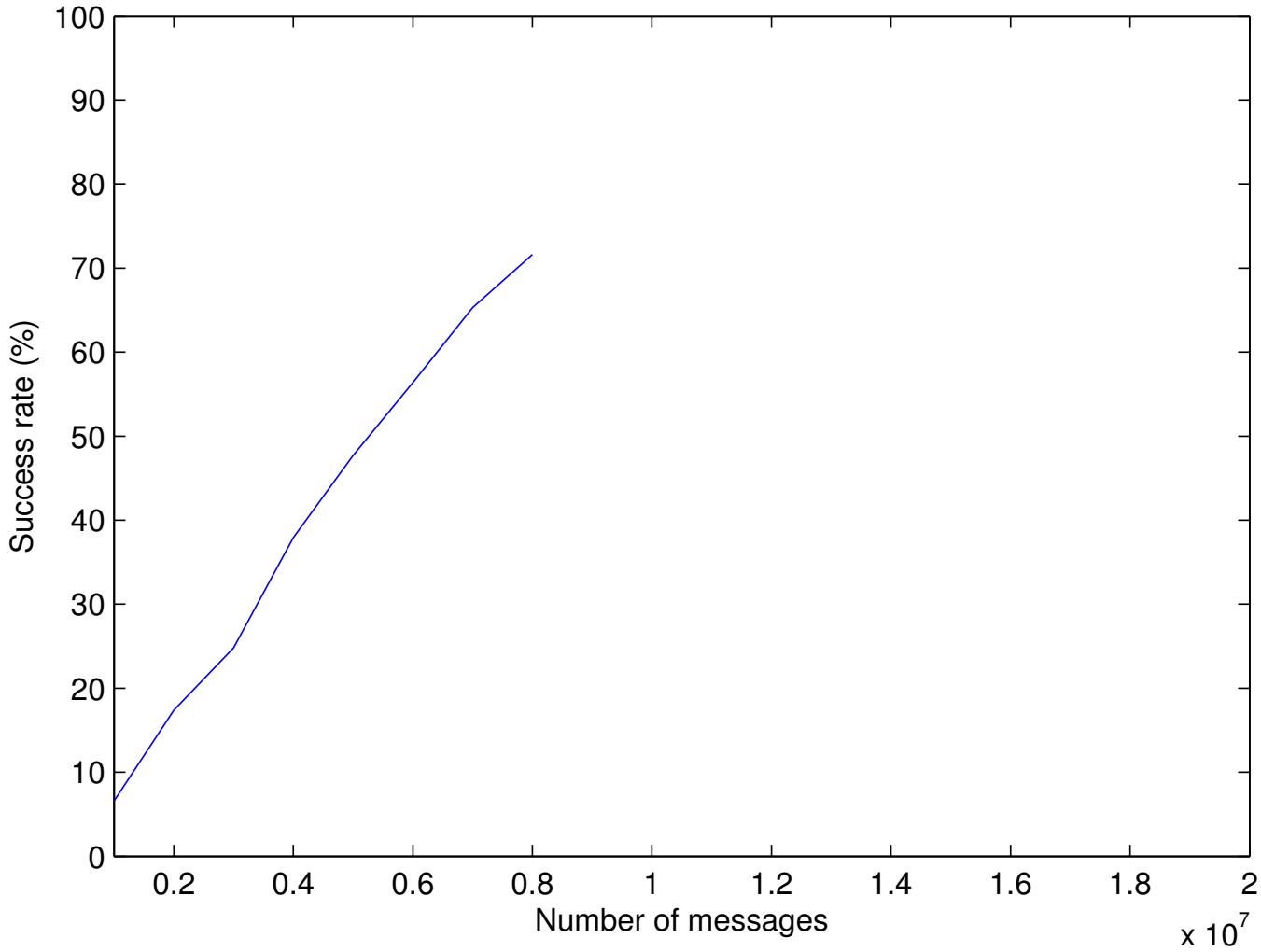
Conclusion: 1500 samples are enough to accurately assess the efficiency of this attack (instead of 10 Millions !).

Increasing the order

- Number of observations: constant (instead of exponential).
- Number of operations: linear (instead of exponential).

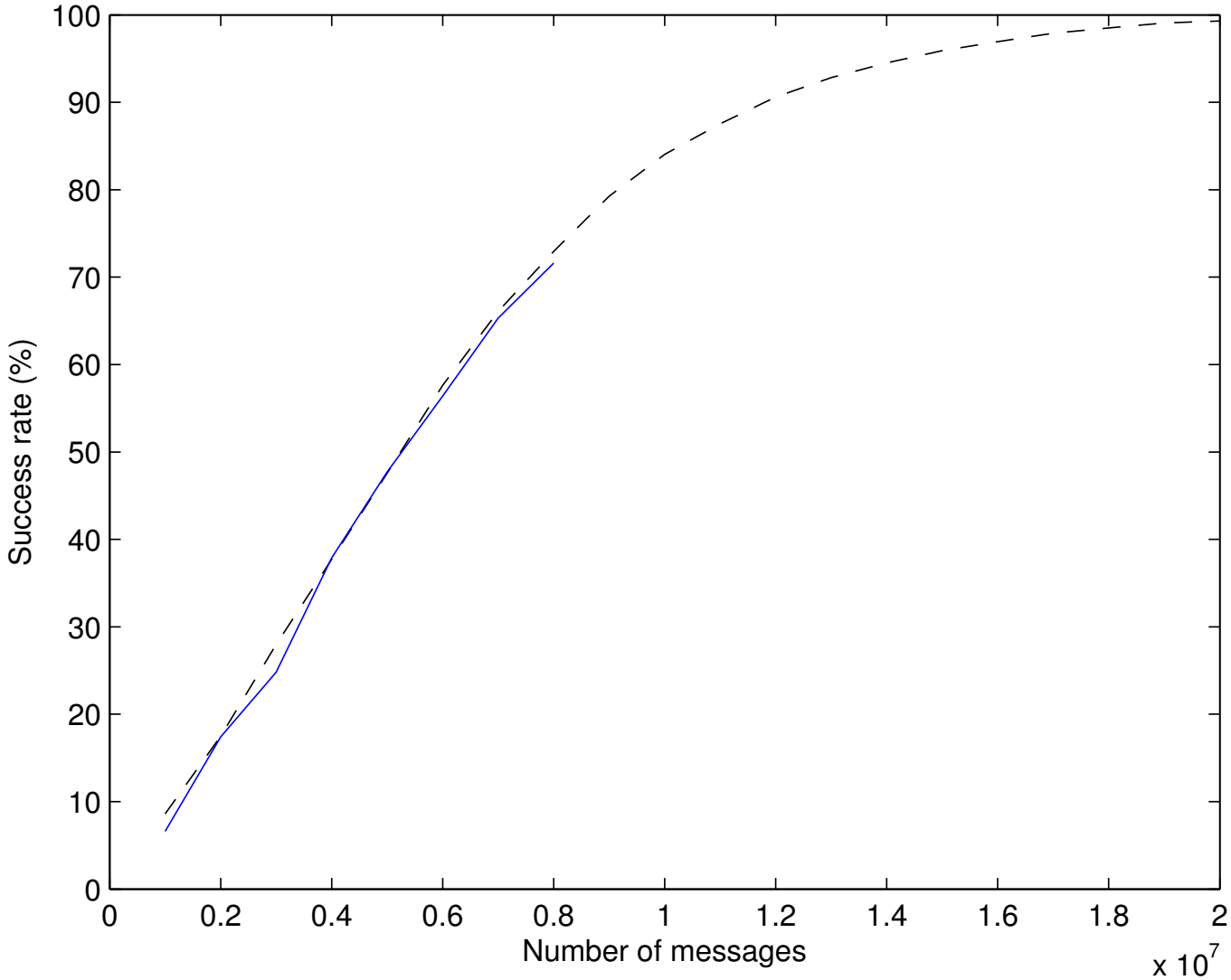
What about our 4O-CPA ?

$1000 \times 10^7 = 10$ Billions of observations.

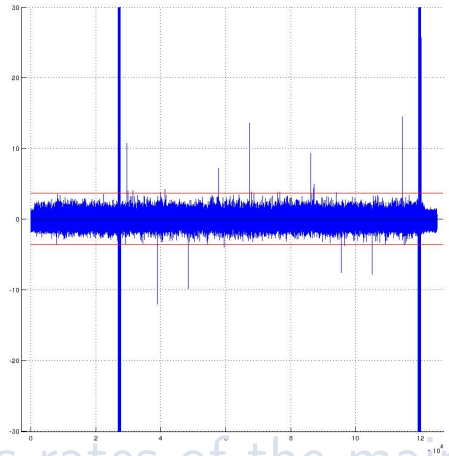


What about our 4O-CPA ?

15 hundreds of observations.

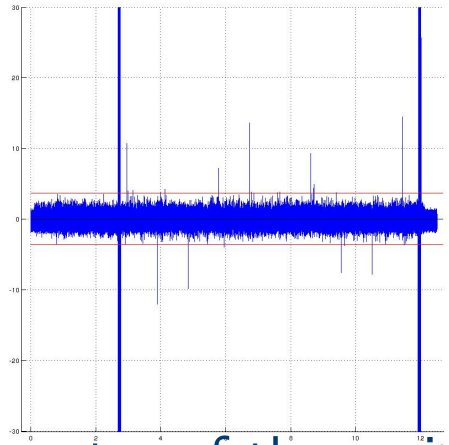


Conclusion about SR and Profiling



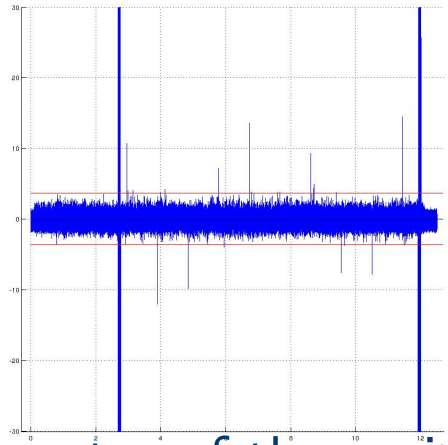
- From good leakage assessment, the success rates of the main side-channel attacks can be deduced.
  - Do we still need to perform the attacks?
- Formulas moreover indicate the impact of device's parameters on the SR.
  - Related to recent works by Bruneau, Heuser, Guilley and Rioul.
- Possibility to precisely know the SR of attacks requiring a lot of observations, using only a very limited number of acquisitions!
- Open an important issue: specify sound campaigns to compute SNR that can be easily related to information leakage.

Conclusion about SR and Profiling



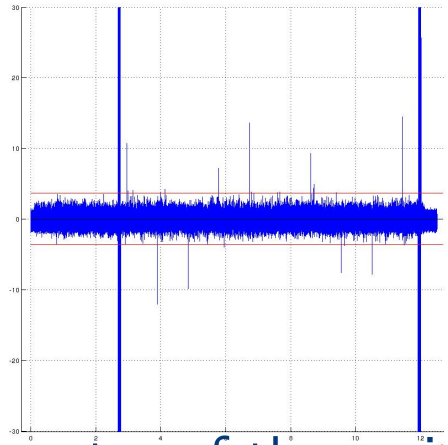
- From good leakage assessment, the success rates of the main side-channel attacks can be deduced.
  - Do we still need to perform the attacks?
- Formulas moreover indicate the impact of device's parameters on the SR.
  - Related to recent works by Bruneau, Heuser, Guilley and Rioul.
- Possibility to precisely know the SR of attacks requiring a lot of observations, using only a very limited number of acquisitions!
- Open an important issue: specify sound campaigns to compute SNR that can be easily related to information leakage.

Conclusion about SR and Profiling



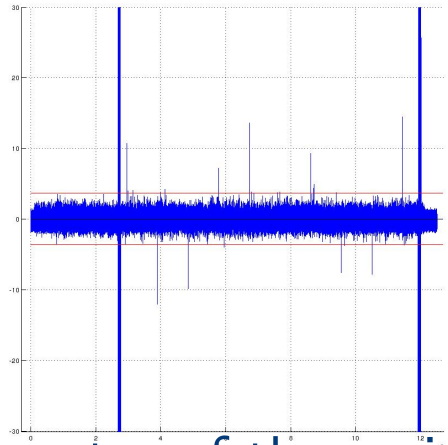
- From good leakage assessment, the success rates of the main side-channel attacks can be deduced.
  - Do we still need to perform the attacks?
- Formulas moreover indicate the impact of device's parameters on the SR.
  - Related to recent works by Bruneau, Heuser, Guilley and Rioul.
- Possibility to precisely know the SR of attacks requiring a lot of observations, using only a very limited number of acquisitions!
- Open an important issue: specify sound campaigns to compute SNR that can be easily related to information leakage.

Conclusion about SR and Profiling



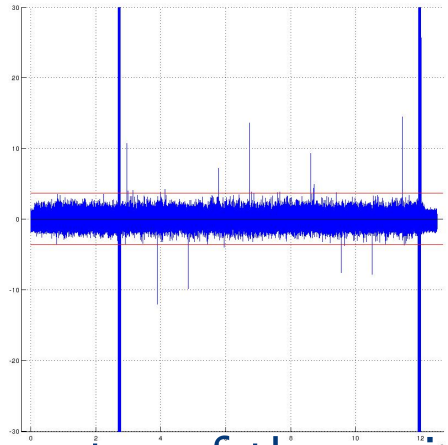
- From good leakage assessment, the success rates of the main side-channel attacks can be deduced.
  - Do we still need to perform the attacks?
- Formulas moreover indicate the impact of device's parameters on the SR.
  - Related to recent works by Bruneau, Heuser, Guilley and Rioul.
- Possibility to precisely know the SR of attacks requiring a lot of observations, using only a very limited number of acquisitions!
- Open an important issue: specify sound campaigns to compute SNR that can be easily related to information leakage.

Conclusion about SR and Profiling



- From good leakage assessment, the success rates of the main side-channel attacks can be deduced.
  - Do we still need to perform the attacks?
- Formulas moreover indicate the impact of device's parameters on the SR.
  - Related to recent works by Bruneau, Heuser, Guilley and Rioul.
- Possibility to precisely know the SR of attacks requiring a lot of observations, using only a very limited number of acquisitions!
- Open an important issue: specify sound campaigns to compute SNR that can be easily related to information leakage.

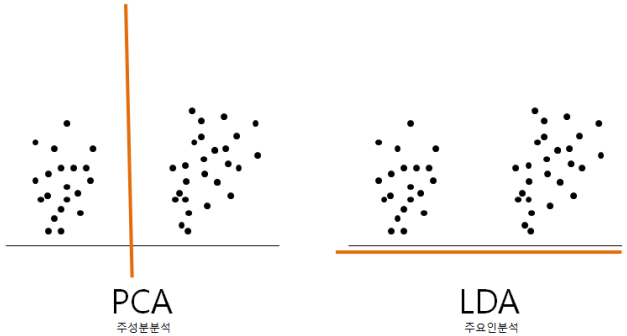
Conclusion about SR and Profiling



- From good leakage assessment, the success rates of the main side-channel attacks can be deduced.
  - Do we still need to perform the attacks?
- Formulas moreover indicate the impact of device's parameters on the SR.
  - Related to recent works by Bruneau, Heuser, Guilley and Rioul.
- Possibility to precisely know the SR of attacks requiring a lot of observations, using only a very limited number of acquisitions!
- Open an important issue: specify sound campaigns to compute SNR that can be easily related to information leakage.

Machine Learning and PoI

Some Open Issues



- Sound choice among the first components returned by Principal Component Analysis (PCA) or Linear Discriminant analysis
  - maximizing intra/inter class variances is not equivalent to maximizing information leakage!
- Combine use of Kernel functions with PCA/LDA to get efficient methods to identify PoI (or tuples of PoI against masked implementations)
- Compare PCA/LDA + Kernel functions with recent methods based on Projection Pursuits.

Conclusion

- Leakage assessment is a sound alternative to the "attacks testing" approach (at least for designers/developers)
- Profiling + leakage assessment enables to soundly estimate the efficiency of the main SCA
- Machine Learning provides us with many tools to answer our questions (key extraction, dimension reduction,) BUT adaptations and further studies are needed!
- Genericity and automitzation is our Graal
  - it does not go against human expertise, it helps to identify the central problems!

Acknowledgement: part of the slides come from presentations given by Eleonora Cagli, Thomas Roche, Adrian Thillard