The Eurocrypt 2009 Evaluation Framework for SCAs, Revisited

F.-X. Standaert

UCL Crypto Group, Université catholique de Louvain

LIRMM, Montpellier, France, July, 2012
SKLOIS, Bejing, China, August 2012

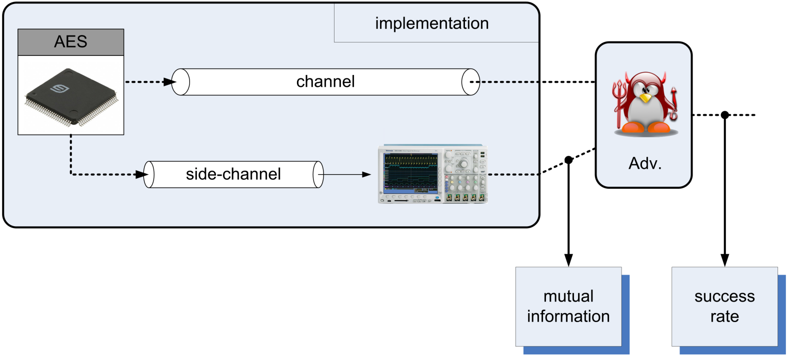
Outline

- The big picture
- Motivating worst case evaluation
- Applying the framework
  - Information theoretic analysis
    - Introduction
    - In practice
    - Main theorem
    - Examples of applications
  - Security analysis
- Which statistical tools to use ?
- Conclusion

Outline

- The big picture
- Motivating worst case evaluation
- Applying the framework
  - Information theoretic analysis
    - Introduction
    - In practice
    - Main theorem
    - Examples of applications
  - Security analysis
- Which statistical tools to use ?
- Conclusion

SCA evaluation framework [1]



Three main ingredients : *design* (e.g. AES in a μcontroller), *leakage function* (e.g. power cons. + scope), *adversary*

Definition of the adversary

- $\mathbf{Adv}(p, d, n, t, m, s)$
  - p : profiled or non-profiled attack
  - d : data complexity (excludes repetition)
  - n : number of measurements (includes repetition)
  - t : time complexity
  - m : memory complexity
  - $s \in$ unknown/known/chosen plaintexts/ciphertexts

Definition of the leakage function

- Formally, $\mathsf{L}(\delta, \Sigma, \rho)$
  - δ : configuration of the target device
    - Depends on the public input x and secret input k
    - May depend on a random (non-physical) parameter r
  - Σ : measurement setup
  - ρ : physical randomness

Definition of the leakage function

- Formally, $\mathsf{L}(\delta, \Sigma, \rho)$
  - δ : configuration of the target device
    - Depends on the public input x and secret input k
    - May depend on a random (non-physical) parameter r
  - Σ : measurement setup
  - ρ : physical randomness
- Additional informal classification :
  - Independent noise : if $\mathsf{L}(x, k, \rho) = f(x, k) + g(\rho)$
  - Variability : if $\mathsf{L}(x, k, \rho)$ is different for “similar” chips
  - Linear : if $f(x, k)$ is a linear function of x, k
  - Non-linear : if $f(x, k)$ is a non-linear function of x, k

Specification of the design

- Cryptographic algorithm
- Target device and technology
- Type of countermeasures inserted, e.g.
  - Noise addition
  - Masking
  - Time randomization
  - Dual-rail logic styles
  - Re-keying
  - ...

Message #1

- SCA depend on many parameters
- Any comparison should fix all of them but one

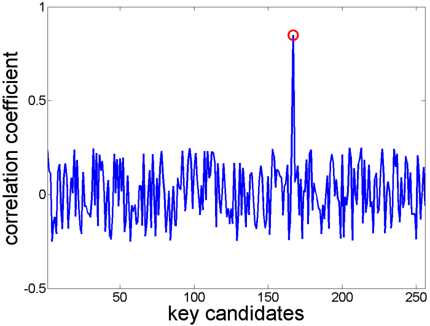
- e.g. impact of a countermeasure
  - Best analyzed on the same device & with the same setup as the unprotected implementation

Outline

- The big picture
- Motivating worst case evaluation
- Applying the framework
  - Information theoretic analysis
    - Introduction
    - In practice
    - Main theorem
    - Examples of applications
  - Security analysis
- Which statistical tools to use ?
- Conclusion

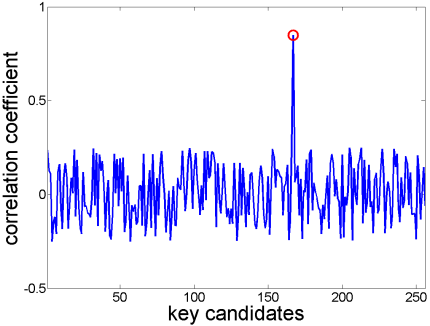
How not to evaluate

- Launch a single attack with an arbitrary distinguisher



How not to evaluate

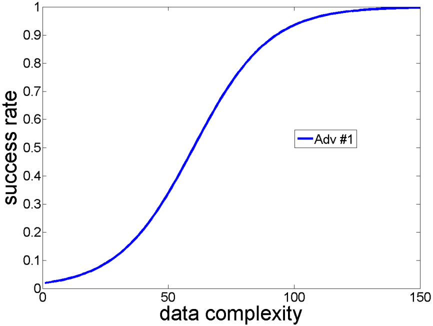
- Launch a single attack with an arbitrary distinguisher



- First issue : no statistical confidence in the evaluation

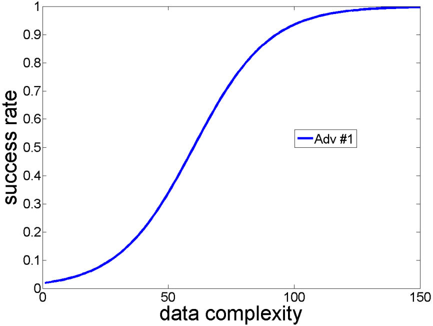
A first improvement

- Repeat the attack and estimate a success rate



A first improvement

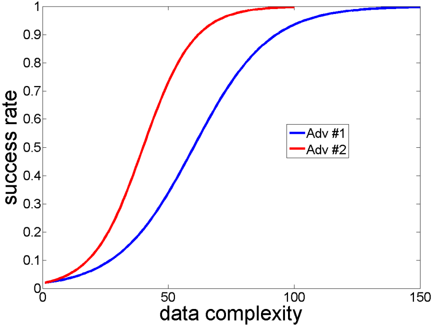
- Repeat the attack and estimate a success rate



- Second issue : arbitrary adversary (maybe suboptimal)

A first improvement

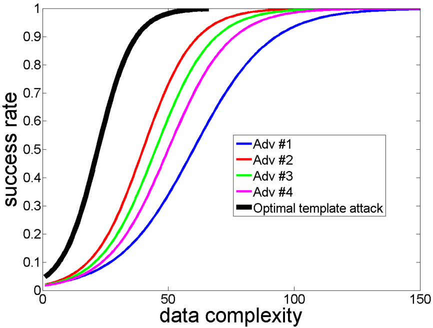
- Repeat the attack and estimate a success rate



- A stronger adversary may invalidate the evaluation

A second improvement

- Apply an "optimal" template attack



Message #2

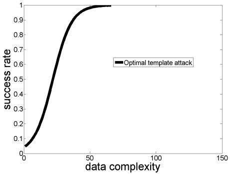
- Worst case evaluation
    - Anticipates "all" side-channel adversaries
    - Adds security margins to the implementations
        - Practical adversaries may be suboptimal
    - Represents the designer's point of view
- Profiling is (provably) needed for this purpose [2]

Outline

- The big picture
- Motivating worst case evaluation
- Applying the framework
  - Information theoretic analysis
    - Introduction
    - In practice
    - Main theorem
    - Examples of applications
  - Security analysis
- Which statistical tools to use ?
- Conclusion

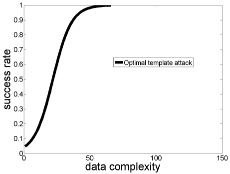
The starting point

- Why do we need it ?
  - All the quantified data of a worst case evaluation is contained in security metrics (e.g. success rates)



The starting point

- Why do we need it ?
  - All the quantified data of a worst case evaluation is contained in security metrics (e.g. success rates)



- But evaluating = quantifying + understanding
- Remaining issue : why is the attack successful ?
  - Information theoretic analysis helps understanding

Estimation issues

- Information theoretic analysis = estimating the information leakage $\perp\!\!\!\perp$ of the adversary
- But estimating the mutual information between arbitrary distributions is notoriously hard
  - Estimators are biased & distribution-dependent

Estimation issues

- Information theoretic analysis = estimating the information leakage ⫫ of the adversary
- But estimating the mutual information between arbitrary distributions is notoriously hard
  - Estimators are biased & distribution-dependent

- Good news : side-channel attacks need a model
  - i.e. an estimation of the leakage distribution
- Main idea : estimate the mutual information from the "best available" profiled model (i.e. the worst case)

Definition

- Information leakage on the secret key

$$\mathsf{H}[K] - \sum_{k \in \mathcal{K}} \Pr[k] \sum_{l \in \mathcal{L}} \Pr_{\texttt{chip}}[l|k] \cdot \log_2 \hat{\Pr}_{\texttt{model}}[k|l],$$

Definition

- Information leakage on the secret key

$$\mathsf{H}[K] - \sum_{k \in \mathcal{K}} \Pr[k] \sum_{l \in \mathcal{L}} \Pr_{\texttt{chip}}[l|k] \cdot \log_2 \hat{\Pr}_{\texttt{model}}[k|l],$$

- where $\hat{\Pr}_{\texttt{model}}[k|l]$ is obtained by profiling the target device
- where $\Pr_{\texttt{chip}}[k|l]$ is obtained by sampling the target device

$\Rightarrow$ Two cases can happen

Case #1 : ideal evaluation

Perfect profiling phase

$$\downarrow$$

$$\hat{\Pr}_{\texttt{model}} = \Pr_{\texttt{chip}}$$

$$\hat{\mathrm{MI}}(K;L) = \mathrm{H}[K] - \sum_{k\in\mathcal{K}} \Pr[k] \sum_{l\in\mathcal{L}} \Pr_{\texttt{chip}}[l|k] \log_2 \hat{\Pr}_{\texttt{model}}[k|l]$$

$\Rightarrow$ mutual information properly estimated

Case #2 : "biased" evaluation

Bounded profiling phase

Variability

Simpler model

$$\hat{\Pr}_{\texttt{model}} \cancel{=} \Pr_{\texttt{chip}}$$

$$\cancel{\hat{\mathrm{MI}}}(K;L) = \mathrm{H}[K] - \sum_{k \in \mathcal{X}} \Pr[k] \sum_{l \in \mathcal{L}} \Pr_{\texttt{chip}}[l|k] \log_2 \hat{\Pr}_{\texttt{model}}[k|l]$$

$\hat{\mathrm{PI}}$

perceived information = estimator for the mutual information biased by the adversary's model

Message #3

- In general, $\mathrm{MI}(K; L)$ cannot be exactly computed
- But we can sometime be sufficiently close
  - (see the "tools" section)

- Goal of an evaluator : be as close as possible
  - Again motivates the use of profiling

Outline

- The big picture
- Motivating worst case evaluation
- Applying the framework
  - Information theoretic analysis
    - Introduction
    - In practice
    - Main theorem
    - Examples of applications
  - Security analysis
- Which statistical tools to use ?
- Conclusion

Two-step process

- Step 1 : estimate the leakage model $\hat{\Pr}_{\texttt{model}}[k|l]$
  - e.g. with Gaussian templates, linear regression [3] (or Gaussian Mixtures, SVMs, ...)
- Step 2 : estimate $\hat{\mathrm{PI}}(K;L)$ by sampling $\hat{\Pr}_{\texttt{chip}}[k|l]$
  - i.e. by generating actual measurements

Two-step process

- Step 1 : estimate the leakage model $\hat{\Pr}_{\texttt{model}}[k|l]$
  - e.g. with Gaussian templates, linear regression [3] (or Gaussian Mixtures, SVMs, ...)
- Step 2 : estimate $\hat{\mathrm{PI}}(K;L)$ by sampling $\hat{\Pr}_{\texttt{chip}}[k|l]$
  - i.e. by generating actual measurements
- Note : measurements to estimate the leakage model and to estimate $\hat{\mathrm{PI}}(K;L)$ must be different

Example

- 4 key candidates with correct key $k = 1$
- $\sum_{l \in \mathcal{L}} \mathrm{Pr}_{\texttt{chip}}[l|k=1] \log_2 \hat{\mathrm{Pr}}_{\texttt{model}}[k=1|l]$ estimation

Example

- 4 key candidates with correct key $k = 1$
- $\sum_{l \in \mathcal{L}} \mathrm{Pr}_{\texttt{chip}}[l|k=1] \log_2 \hat{\mathrm{Pr}}_{\texttt{model}}[k=1|l]$ estimation

	$k=0$	$k=1$	$k=2$	$k=3$
l_1	$\hat{p}_0^1$	$\hat{p}_1^1$	$\hat{p}_2^1$	$\hat{p}_3^1$

Example

- 4 key candidates with correct key $k = 1$
- $\sum_{l \in \mathcal{L}} \mathrm{Pr}_{\texttt{chip}}[l|k=1] \log_2 \hat{\mathrm{Pr}}_{\texttt{model}}[k=1|l]$ estimation

	$k=0$	$k=1$	$k=2$	$k=3$
l_1	$\hat{p}_0^1$	$\hat{p}_1^1$	$\hat{p}_2^1$	$\hat{p}_3^1$
l_2	$\hat{p}_0^2$	$\hat{p}_1^2$	$\hat{p}_2^2$	$\hat{p}_3^2$

Example

- 4 key candidates with correct key $k = 1$
- $\sum_{l \in \mathcal{L}} \mathrm{Pr}_{\texttt{chip}}[l|k=1] \log_2 \hat{\mathrm{Pr}}_{\texttt{model}}[k=1|l]$ estimation

	$k=0$	$k=1$	$k=2$	$k=3$
l_1	$\hat{p}_0^1$	$\hat{p}_1^1$	$\hat{p}_2^1$	$\hat{p}_3^1$
l_2	$\hat{p}_0^2$	$\hat{p}_1^2$	$\hat{p}_2^2$	$\hat{p}_3^2$
l_3	$\hat{p}_0^3$	$\hat{p}_1^3$	$\hat{p}_2^3$	$\hat{p}_3^3$

Example

- 4 key candidates with correct key $k = 1$
- $\sum_{l \in \mathcal{L}} \mathrm{Pr}_{\texttt{chip}}[l|k = 1] \log_2 \hat{\mathrm{Pr}}_{\texttt{model}}[k = 1|l]$ estimation

	$k=0$	$k=1$	$k=2$	$k=3$
l_1	$\hat{p}_0^1$	$\hat{p}_1^1$	$\hat{p}_2^1$	$\hat{p}_3^1$
l_2	$\hat{p}_0^2$	$\hat{p}_1^2$	$\hat{p}_2^2$	$\hat{p}_3^2$
l_3	$\hat{p}_0^3$	$\hat{p}_1^3$	$\hat{p}_2^3$	$\hat{p}_3^3$
$\dots$	$\dots$	$\dots$	$\dots$	$\dots$
l_N	$\hat{p}_0^N$	$\hat{p}_1^N$	$\hat{p}_2^N$	$\hat{p}_3^N$

$$\Rightarrow \frac{1}{N} \sum_{i=1}^{N} \log_2 \hat{p}_1^i$$

Note

- MI/PI metrics $\neq$ Gierlichs et al.'s MIA [4]

Note

- MI/PI metrics $\neq$ Gierlichs et al.'s MIA [4]
- MIA is a *non-profiled* distinguisher
- MI/PI metrics are *profiled* (worst case) eval. criteria

Note

- MI/PI metrics $\neq$ Gierlichs et al.'s MIA [4]
- MIA is a *non-profiled* distinguisher
- MI/PI metrics are *profiled* (worst case) eval. criteria
- MIA requires to define a *target operation*
- MI/PI metrics are best estimated when capturing the key leakage from *all intermediate computations* [5]

Note

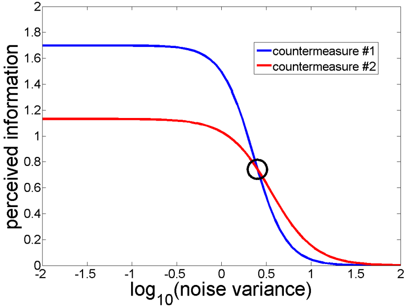
- MI/PI metrics $\neq$ Gierlichs et al.'s MIA [4]
- MIA is a *non-profiled* distinguisher
- MI/PI metrics are *profiled* (worst case) eval. criteria
- MIA requires to define a *target operation*
- MI/PI metrics are best estimated when capturing the key leakage from *all intermediate computations* [5]
- The MIA distinguisher provides a lower bound of the actual information leakage given by the MI/PI metrics

Outline

- The big picture
- Motivating worst case evaluation
- Applying the framework
  - Information theoretic analysis
    - Introduction
    - In practice
    - Main theorem
    - Examples of applications
  - Security analysis
- Which statistical tools to use ?
- Conclusion

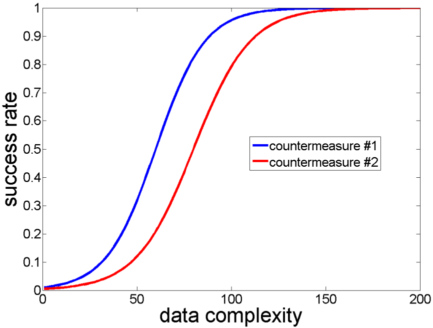
Main theorem (informal)

- $\mathrm{PI}(K;L)$ is directly proportional to the success rate of an adversary using $\hat{\Pr}_{\texttt{model}}[k|l]$ as template
- e.g. $\mathrm{PI}(K;L)$ in function of the noise variance



As a result

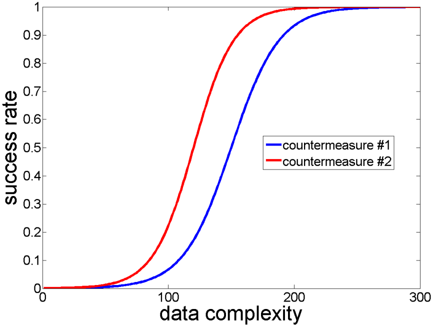
- Left of the intersection



- Countermeasure #2 more secure than first one

As a result

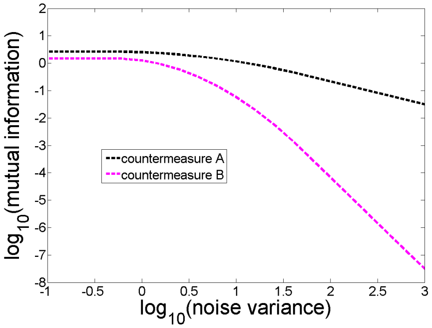
- Right of the intersection



- Countermeasure #1 more secure than first one

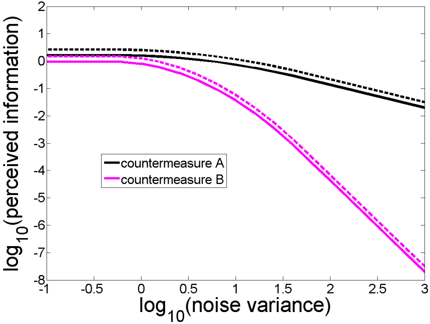
In other words

- $\mathrm{MI}(K;L)$ measures the worst case data complexity

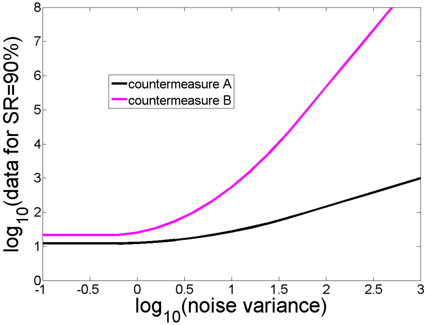


In other words

- $\mathrm{PI}(K; L)$ is the evaluator's best estimate

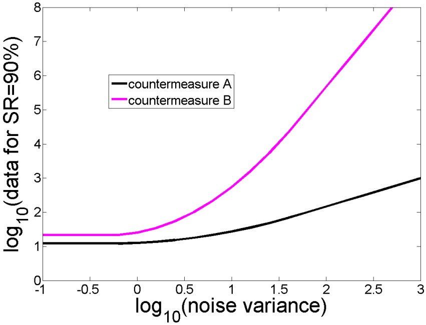


Relation with data complexity



- Theorem only proven in very specific cases

Relation with data complexity



- Theorem only proven in very specific cases
- But holds surprisingly well in all real-world settings

Message #4

- A single success rate curve does not reveal a trend nor an explanation about a leaking device
- Most intuition regarding the data complexity of of a side-channel attack can be extracted by plotting $\mathrm{PI}(K;L)$ in function of a noise variable
- $\mathrm{PI}(K;L)$ curves are easier to sample than the average data complexity to reach a given success rate

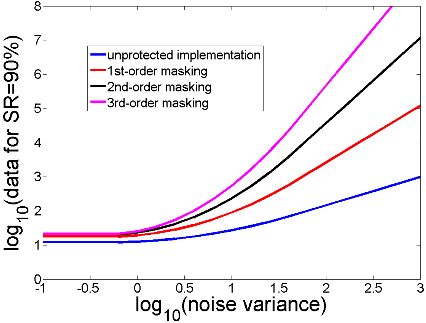
Outline

- The big picture
- Motivating worst case evaluation
- Applying the framework
  - Information theoretic analysis
    - Introduction
    - In practice
    - Main theorem
    - Examples of applications
  - Security analysis
- Which statistical tools to use ?
- Conclusion

Example #1 : masking

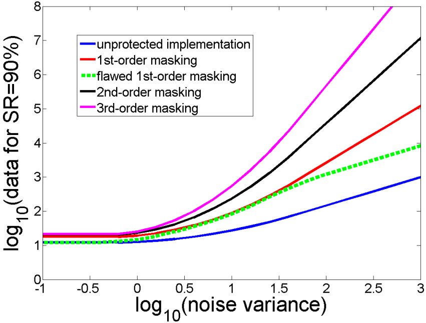
- Main idea : split the sensitive data in r shares
- If "perfect" implementation, the data complexity to break masking is proportional to $(\sigma_n^2)^r$
  - Perfect $\approx$ if the smallest-order key-dependent moment in the leakage distribution is r
  - Essentially depends on the hardware (e.g. glitches or early propagation make implementations imperfect)

Information theoretic intuition [6]



- Smallest-order key-dept. moment = slope of the curve

Information theoretic intuition [6]

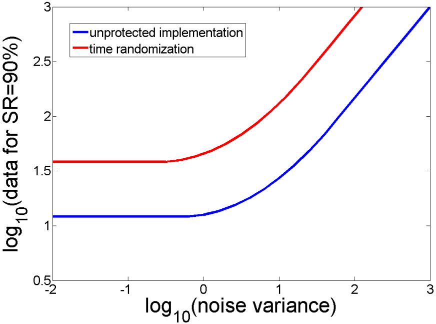


- Flaws due to physical defaults can be detected

Example #2 : time randomization

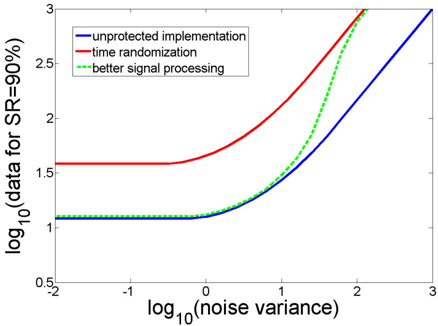
- Random delays, unstable clock, shuffling, ...
- Essentially adds noise to the implementation

Information theoretic intuition [7]



- e.g. shuffling can give rise to a Y-axis shift

Information theoretic intuition [7]



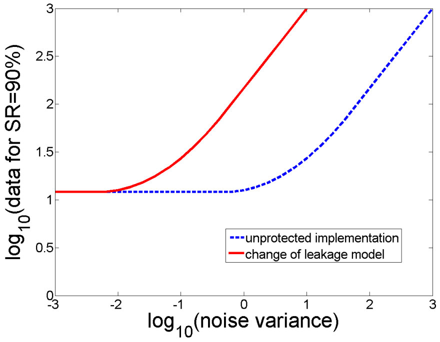
- Main issue : highly dependent on signal processing

Example #3 : dual rail logic styles

- Main idea : have constant activity within the implementation in order to
  1. Modify the leakage models (i.e. avoid simple models such as Hamming weight/distance)
  2. Reduce the data dependencies in the leakages

- Practical limitation : usually implies strong hardware constraints (i.e. need to "balance" the wires)

Information theoretic intuition [8]

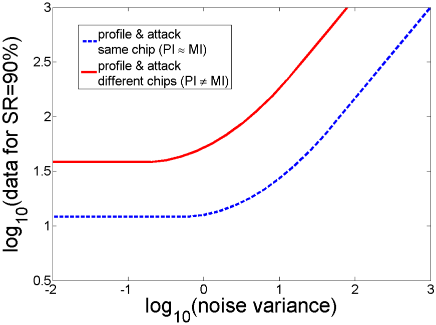


- Reduced data dependencies $\Rightarrow$ X-axis shift

Example #4 : variability

- Leakage function can be $\neq$ for $\neq$ "similar" chips
  - e.g. because of manufacturing process
- Raises new questions regarding profiled attacks
  - e.g. profile n chips, attack another chip
    - How large should n be ?
- Variability may create a gap between MI and PI

Information theoretic intuition [9]



- Worst case may be harder to exploit by adversaries...
- ...but remains the most reliable evaluation metric!

Message #5

- $\mathrm{PI}(K; l)$ provides a unifying view of countermeasures
- Only masking can lead to exponential security increase

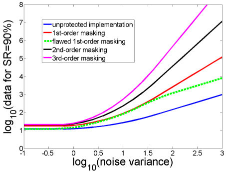
- Again, beware of "false sense of security"
  - $\mathrm{PI}(K; L) \neq \mathrm{MI}(K; L)$
  - Significant differences may be due to signal processing, bad assumptions on the leakage, ...
  - Measurement setup also matters (a lot)

Outline

- The big picture
- Motivating worst case evaluation
- Applying the framework
  - Information theoretic analysis
    - Introduction
    - In practice
    - Main theorem
    - Examples of applications
  - Security analysis
- Which statistical tools to use ?
- Conclusion

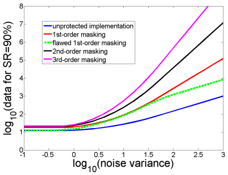
The starting point

- Why do we need it ?
  - Information theoretic curves capture most intuition about the data complexity of worst-case attacks



The starting point

- Why do we need it ?
  - Information theoretic curves capture most intuition about the data complexity of worst-case attacks

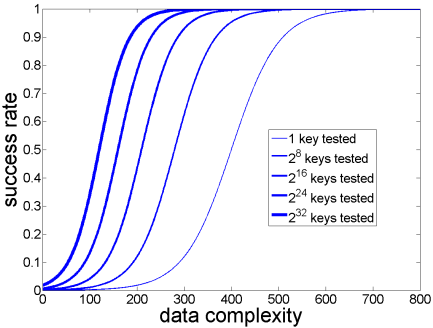


- But side-channel attacks also depend on time
- And evaluating multiple (not only worst-case) adversaries may be revealing as well [10]

Example #1 : masking

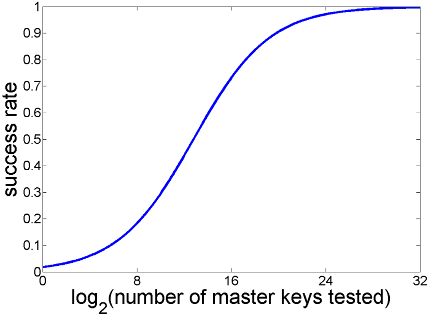
- If the r shares are manipulated in different clock cycles (i.e. in software, typically), finding these cycles requires testing N^r r-uples of time samples

Example #2 : key enumeration [11]



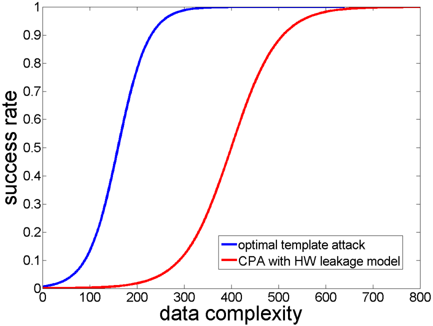
- Significant impact on the success rates

Example #2 : key enumeration [11]



- Missing data can always be traded for computations

Example #3 : other attacks



- Non-profiled attacks can be significantly less efficient

Message #6

- Security analysis : necessary complement to IT analysis
- It allows highlighting the gap between profiled and (usually more realistic) non-profiled attacks
- It incorporates time complexity in the evaluations
  - Adversaries can enumerate up to 2^{50}-2^{60} keys
  - Evaluate success rates of high orders !

Outline

- The big picture
- Motivating worst case evaluation
- Applying the framework
  - Information theoretic analysis
    - Introduction
    - In practice
    - Main theorem
    - Examples of applications
  - Security analysis
- Which statistical tools to use ?
- Conclusion

How to evaluate the metrics ?

- Implies to determine good statistical tools
  - Critical point : pdf estimation problem
- Tools are highly dependent on the contexts
- A few examples next. . .

Examples

	profiled attacks	non-profiled attacks
unprotected device, univariate leakage		
unprotected device, multivariate leakage		
dual-rail pre-charged implementation		
time randomizations		
masking		
combination of countermeasures		

- Different types of implementations & countermeasures
- Which cases are "easy to evaluate ?"

Examples

	profiled attacks	non-profiled attacks
unprotected device, univariate leakage		
unprotected device, multivariate leakage		
dual-rail pre-charged implementation		
time randomizations		
masking		
combination of countermeasures		

- Most distinguishers are asymptotically equivalent
- ... if provided with the same leakage model [12]

Examples

	profiled attacks	non-profiled attacks
unprotected device, univariate leakage		
unprotected device, multivariate leakage		
dual-rail pre-charged implementation		
time randomizations		
masking		
combination of countermeasures		

- PCA, LDA, ... useful in the profiled case
- Dimensionality reduction uneasy in non-profiled case

Examples

	profiled attacks	non-profiled attacks
unprotected device, univariate leakage		
unprotected device, multivariate leakage		
dual-rail pre-charged implementation		
time randomizations		
masking		
combination of countermeasures		

- Same tools as for an unprotected device
- Non-linear leakage functions require profiling

Examples

	profiled attacks	non-profiled attacks
unprotected device, univariate leakage		
unprotected device, multivariate leakage		
dual-rail pre-charged implementation		
time randomizations		
masking		
combination of countermeasures		

- Uneasy to evaluate for both types of attacks
- Signal proc. completely removes some countermeasures

Examples

	profiled attacks	non-profiled attacks
unprotected device, univariate leakage		
unprotected device, multivariate leakage		
dual-rail pre-charged implementation		
time randomizations		
masking		
combination of countermeasures		

- Becomes measurement intensive as r increases
- No solution is always optimal in the non-profiled case

Examples

	profiled attacks	non-profiled attacks
unprotected device, univariate leakage		
unprotected device, multivariate leakage		
dual-rail pre-charged implementation		
time randomizations		
masking		
combination of countermeasures		

- Specially hard if the design is unknown
- Large distance btw. profiled & non-profiled cases

Outline

- The big picture
- Motivating worst case evaluation
- Applying the framework
  - Information theoretic analysis
    - Introduction
    - In practice
    - Main theorem
    - Examples of applications
  - Security analysis
- Which statistical tools to use ?
- Conclusion

Conclusions (I)

- Evaluation of DPA quite well understood in theory
  - Which metrics to use and why
    - Perceived information quantifies implementations
    - Success rates quantify adversaries
- But $\exists$ many open question related to the best statistical tools needed to estimate the metrics

Conclusions (II)

- Evaluators should always try to understand from where a "false sense of security" could come from
  - Perceived information can also be used to compare different laboratories (i.e. how good are they in extracting information from an implementation ?)

Conclusions (III)

- Side-channel attacks are more than divide-and-conquer
- Next challenge : combinations with cryptanalysis
  - Collision attacks
  - Algebraic attacks
  - ...

THANKS

e-mail : fstandae@uclouvain.be
web page : http ://perso.uclouvain.be/fstandae/

Bibliography

1. F.-X. Standaert, T.G. Malkin, M. Yung, *A Unified Framework for the Analysis of Side-Channel Key Recovery Attacks*, in the proceedings of Eurocrypt 2009, Lecture Notes in Computer Science, vol 5479, pp 443-461, Cologne, Germany, April 2009, Springer.
2. C. Whitnall, E. Oswald, F.-X. Standaert, *The Myth of Generic DPA... and the Magic of Learning*, cryptology e-Print archive, report 2012/038.
3. F.-X. Standaert, F. Koeune, W. Schindler, *How to Compare Profiled Side-Channel Attacks*, in the proceedings of ACNS 2009, Lecture Notes in Computer Science, vol 5536, pp 485-498, Paris, France, June 2009, Springer.
4. N. Veyrat-Charvillon, F.-X. Standaert, *Mutual Information Analysis : How, When and Why ?*, in the proceedings of CHES 2009, Lecture Notes in Computer Science, vol 5747, pp 429-443, Lausanne, Switzerland, September 2009, Springer.
5. F.-X. Standaert, C. Archambeau, *Using Subspace-Based Template Attacks to Compare and Combine Power and Electromagnetic Information Leakages*, in the proceedings of CHES 2008, Lecture Notes in Computer Science, vol 5154, pp 411-425, Washington DC, USA, August 2008, Springer.
6. F.-X. Standaert, N. Veyrat-Charvillon, E. Oswald, B. Gierlichs, M. Medwed, M. Kasper, S. Mangard, *The World is Not Enough : Another Look on Second-Order DPA*, in the proceedings of Asiacrypt 2010, Lecture Notes in Computer Science, vol 6477, pp 112-129, Singapore, December 2010, Springer.

Bibliography

7. N. Veyrat-Charvillon, M. Medwed, S. Kerckhof, F.-X. Standaert, *Shuffling Against Side-Channel Attacks : a Comprehensive Study with Cautionary Note*, preprint, 2012.
8. M. Renauld, D. Kamel, F.-X. Standaert, D. Flandre, *Information Theoretic and Security Analysis of a 65-nanometer DDSLL AES S-box*, in the proceedings of CHES 2011, Lecture Notes in Computer Science, vol 6917, pp 223-239, Nara, Japan, September 2011, Springer.
9. M. Renauld, F.-X. Standaert, N. Veyrat-Charvillon, D. Kamel, D. Flandre, *A Formal Study of Power Variability Issues and Side-Channel Attacks for Nanoscale Devices*, in the proceedings of Eurocrypt 2011, Lecture Notes in Computer Science, vol 6632, pp 109-128, Tallinn, Estonia, May 2011, Springer.
10. F.-X. Standaert, B. Gierlichs, I. Verbauwhede, *Partition vs. Comparison Side-Channel Distinguishers : an Empirical Evaluation of Statistical Tests for Univariate Side-Channel Attacks against Two Unprotected CMOS Devices*, in the proceedings of ICISC 2008, Lecture Notes in Computer Science, vol 5461, pp 253-267, Seoul, Korea, December 2008, Springer.
11. N. Veyrat-Charvillon, B. Gerard, M. Renauld, F.-X. Standaert, *An Optimal Key Enumeration Algorithm and its Application to Side-Channel Attacks*, cryptology e-Print archive, report 2011/610.
12. S. Mangard, E. Oswald, F.-X. Standaert, *One for All - All for One : Unifying Standard DPA Attacks*, in IET Information Security, vol 5, issue 2, pp 100-110, June 2011.