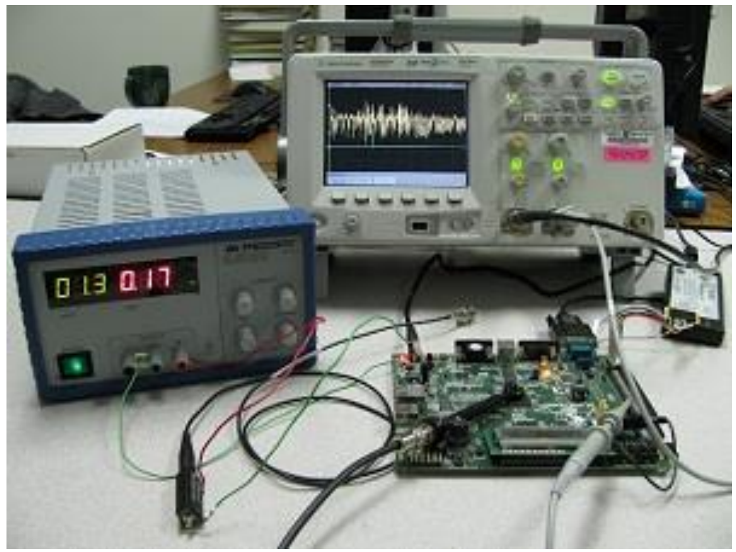
Introduction to Side-Channel Analysis



François-Xavier Standaert

UCL Crypto Group, Belgium

Summer school on real-world crypto, 2016

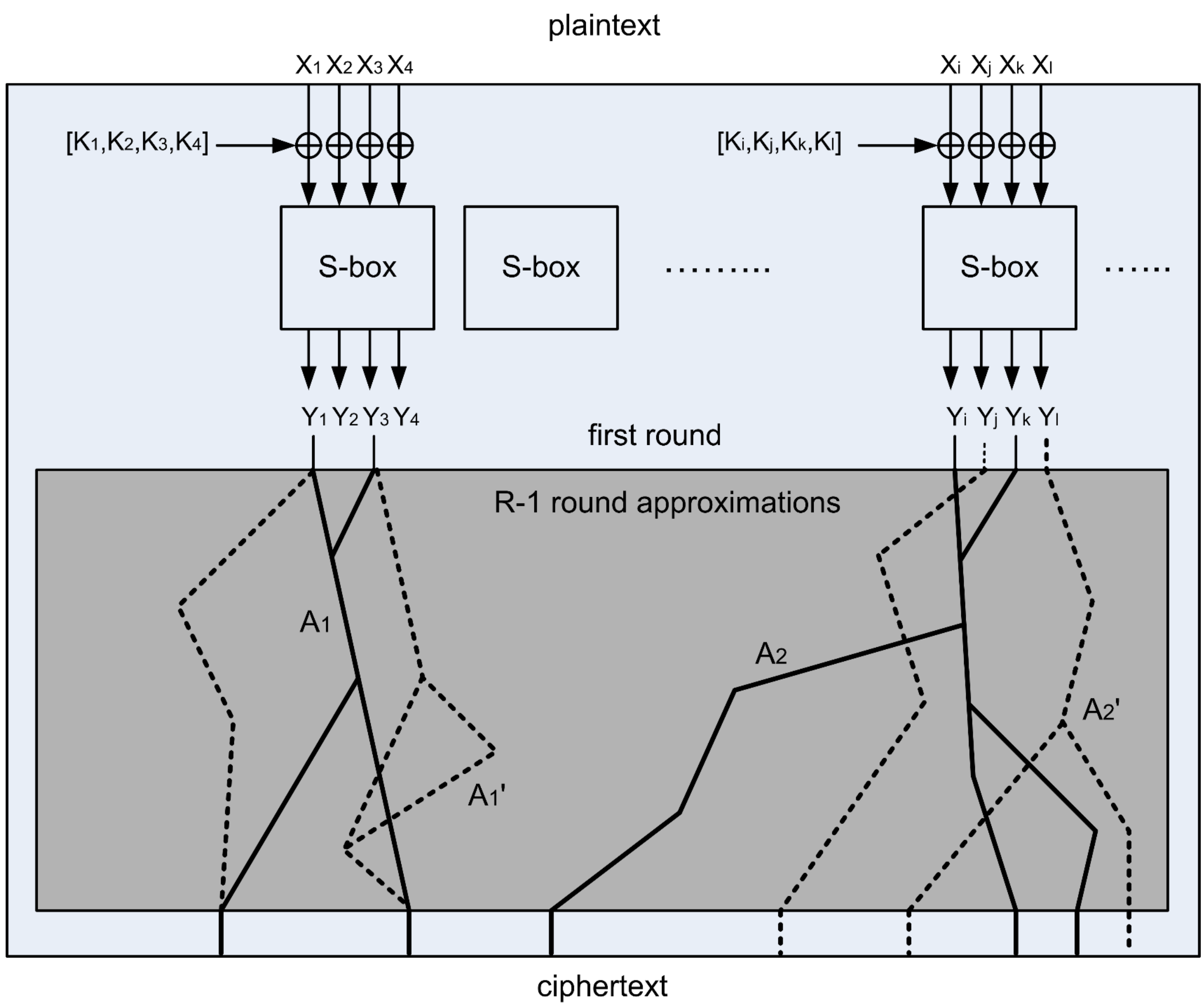
Outline

- Link with linear cryptanalysis
- Standard Differential Power Analysis
- Noise-based security (is not enough)
- *CPA vs Gaussian templates*
- Post-processing the traces
- Noise amplification (aka masking)
- Conclusions & advanced topics

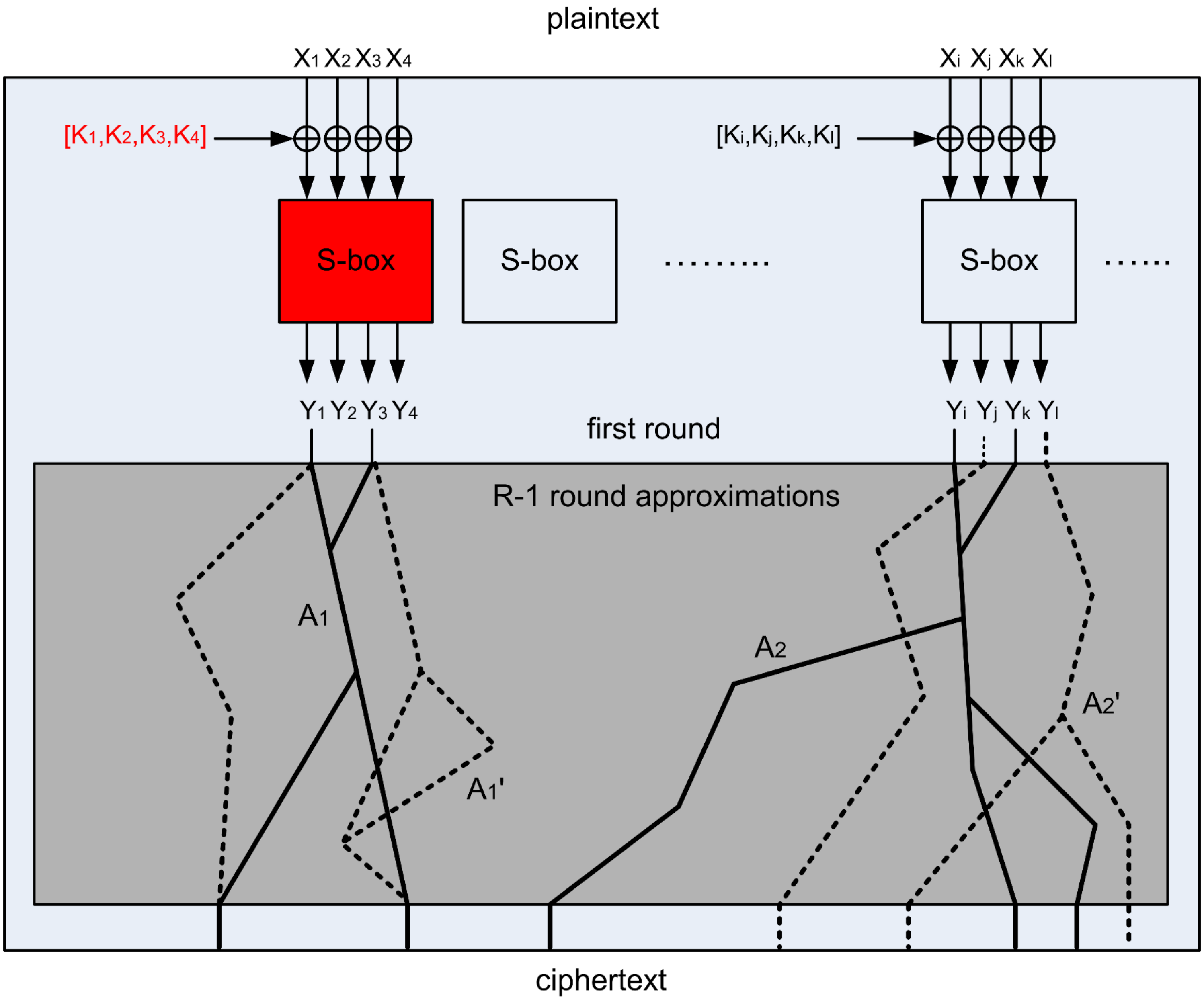
Outline

- **Link with linear cryptanalysis**
- Standard Differential Power Analysis
- Noise-based security (is not enough)
- *CPA vs Gaussian templates*
- Post-processing the traces
- Noise amplification (aka masking)
- Conclusions & advanced topics

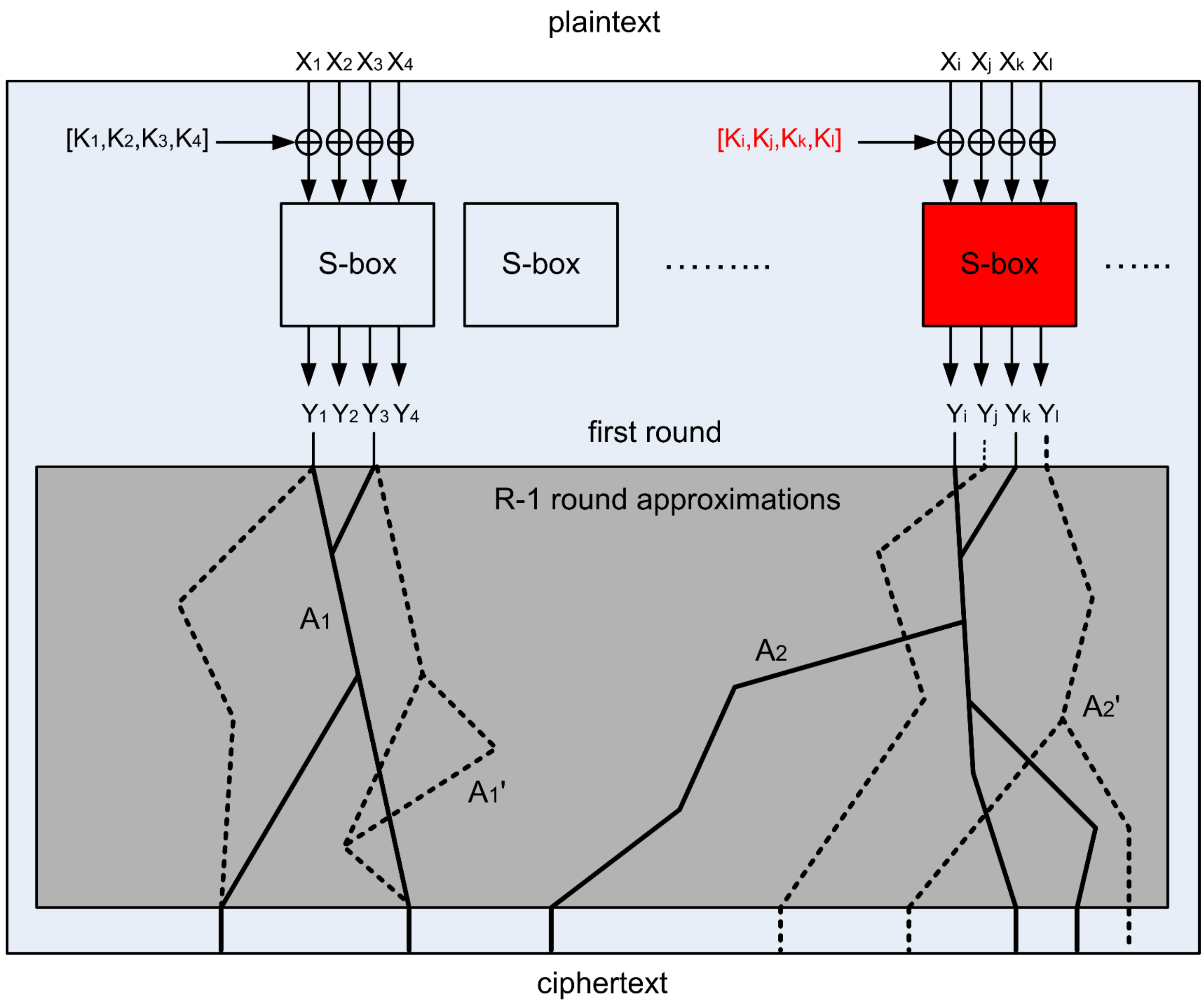
Linear cryptanalysis (I)



Linear cryptanalysis (I)



Linear cryptanalysis (I)



Linear cryptanalysis (II)

- Main characteristics
  - Divide-and-conquer attack
  - Data complexity $\propto \frac{1}{\varepsilon^2}$
    - $\varepsilon = 2^{n-1} \cdot \prod_{s=1}^{n} \varepsilon_s$ (n S-boxes in A, bias ε_s)
  - Time complexity ≈ # of active S-boxes in R1

Linear cryptanalysis (II)

- Main characteristics
  - Divide-and-conquer attack
  - Data complexity $\propto \frac{1}{\varepsilon^2}$
    - $\varepsilon = 2^{n-1} \cdot \prod_{s=1}^{n} \varepsilon_s$ (n S-boxes in A, bias ε_s)
  - Time complexity ≈ # of active S-boxes in R1

- Countermeasures
  - Data: good (non-linear) S-boxes ✓
  - Data & time: Many active S-boxes ✓
  - Data: Larger number of rounds ✓

Linear cryptanalysis (II)

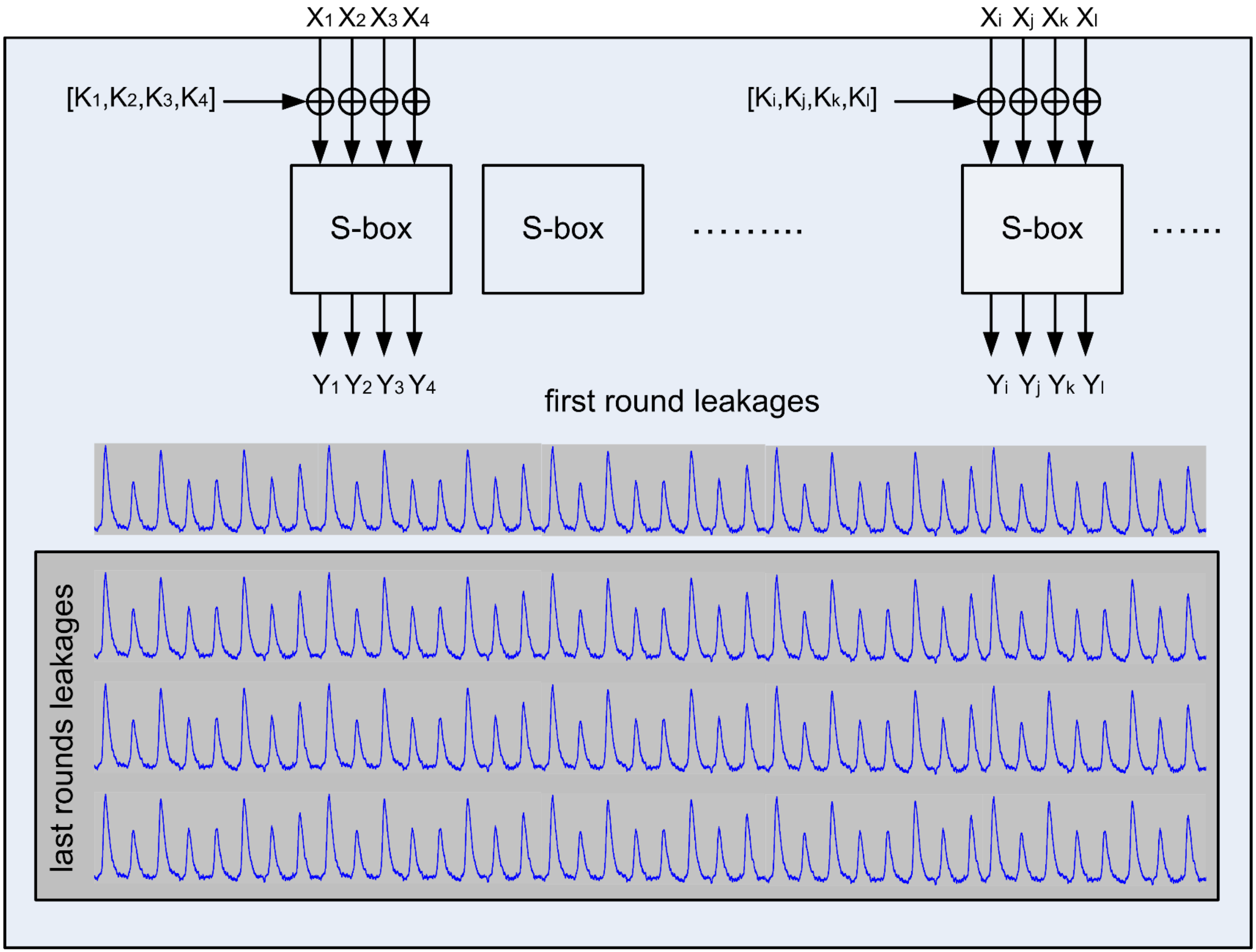
- Main characteristics
  - Divide-and-conquer attack
  - Data complexity $\propto \frac{1}{\varepsilon^2}$
    - $\varepsilon = 2^{n-1} \cdot \prod_{s=1}^{n} \varepsilon_s$ (n S-boxes in A, bias ε_s)
  - Time complexity ≈ # of active S-boxes in R1

- Countermeasures
  - Data: good (non-linear) S-boxes ✓
  - Data & time: Many active S-boxes ✓
  - Data: Larger number of rounds ✓

$\Rightarrow$ AES: $\varepsilon < 2^{-64}$ after a few of rounds

Side-channel cryptanalysis

plaintext



ciphertext

Differential Side-Channel Analysis

- Main characteristics
  - Divide-and-conquer attack
  - Data complexity $\propto \frac{1}{\mathrm{MI}(K;L,X)}$
  - Time complexity $\propto$ # of S-boxes predicted

Differential Side-Channel Analysis

- Main characteristics
  - Divide-and-conquer attack
  - Data complexity $\propto \frac{1}{\mathrm{MI}(K;L,X)}$
  - Time complexity $\propto$ # of S-boxes predicted

- Linear cryptanalysis countermeasures
  - Good (non-linear) S-boxes

- Main characteristics
  - Divide-and-conquer attack
  - Data complexity $\propto \dfrac{1}{\mathrm{MI}(K;L,X)}$
  - Time complexity ∝ # of S-boxes predicted

- Linear cryptanalysis countermeasures
  - Good (non-linear) S-boxes ✗
  - Many active S-boxes

Differential Side-Channel Analysis

- Main characteristics
  - Divide-and-conquer attack
  - Data complexity $\propto \frac{1}{\mathrm{MI}(K;L,X)}$
  - Time complexity ∝ # of S-boxes predicted

- Linear cryptanalysis countermeasures
  - Good (non-linear) S-boxes ✗
  - Many active S-boxes ?
  - Larger number of rounds

Differential Side-Channel Analysis

- Main characteristics
  - Divide-and-conquer attack
  - Data complexity $\propto \frac{1}{\mathrm{MI}(K;L,X)}$
  - Time complexity ∝ # of S-boxes predicted

- Linear cryptanalysis countermeasures
  - Good (non-linear) S-boxes ✗
  - Many active S-boxes ?
  - Larger number of rounds ✗

Differential Side-Channel Analysis

- Main characteristics
  - Divide-and-conquer attack
  - Data complexity $\propto \frac{1}{\mathrm{MI}(K;L,X)}$
  - Time complexity $\propto$ # of S-boxes predicted

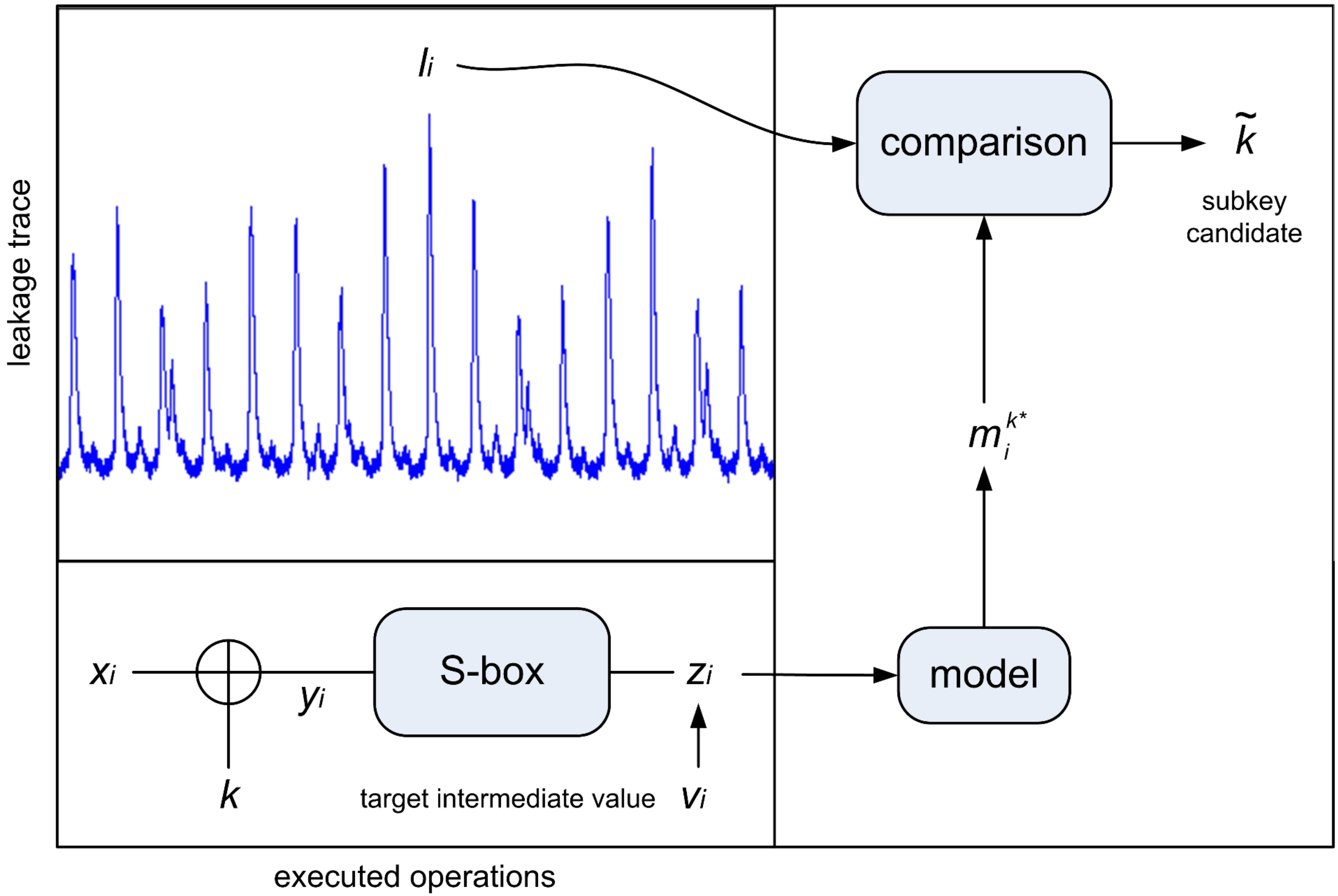
- Linear cryptanalysis countermeasures
  - Good (non-linear) S-boxes ✗
  - Many active S-boxes ?
  - Larger number of rounds ✗

$\Rightarrow$ Unprotected implem: $\mathrm{MI}(K;L,X) > 0.01$

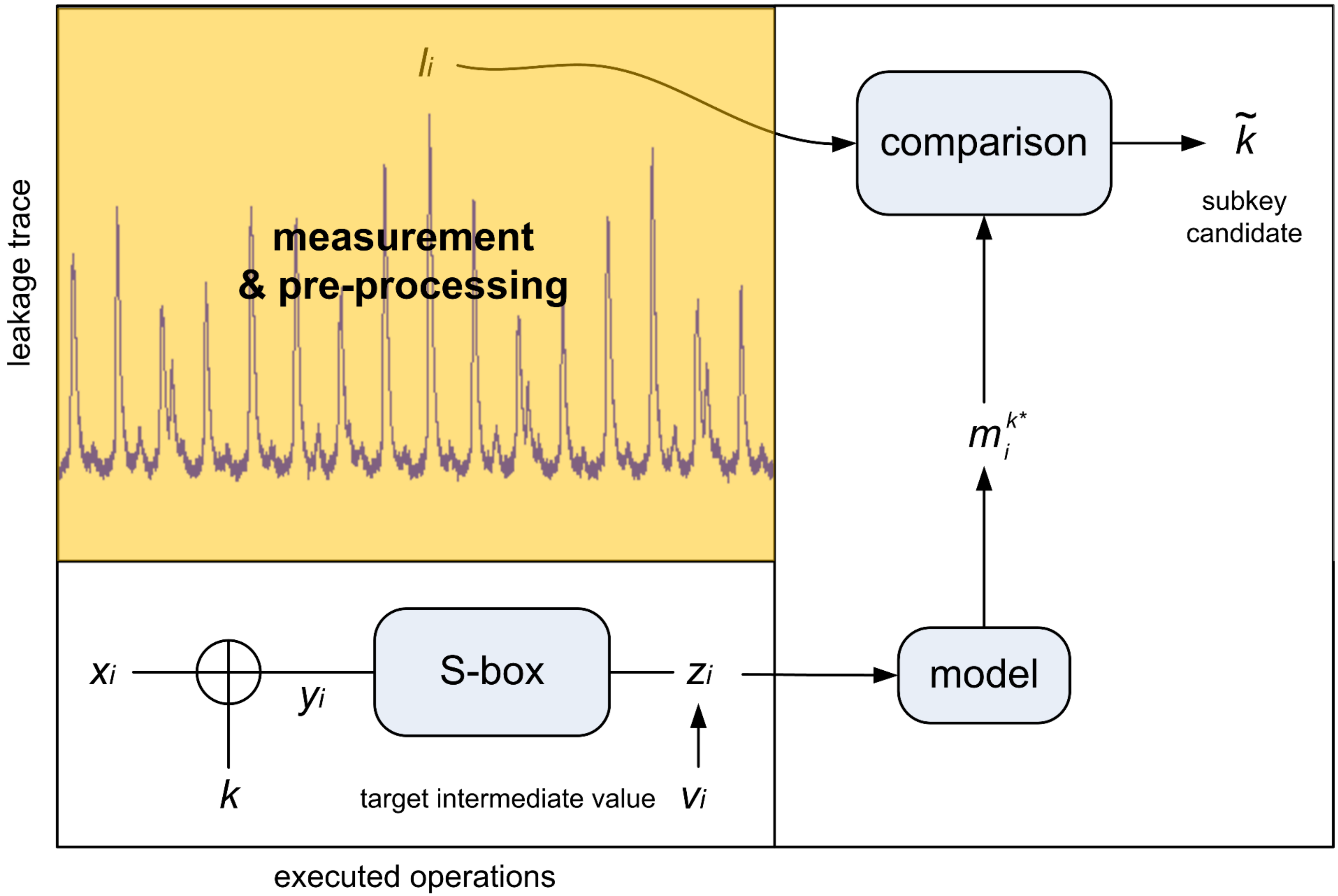
Outline

- Link with linear cryptanalysis
- **Standard Differential Power Analysis**
- Noise-based security (is not enough)
- *CPA vs Gaussian templates*
- Post-processing the traces
- Noise amplification (aka masking)
- Conclusions & advanced topics

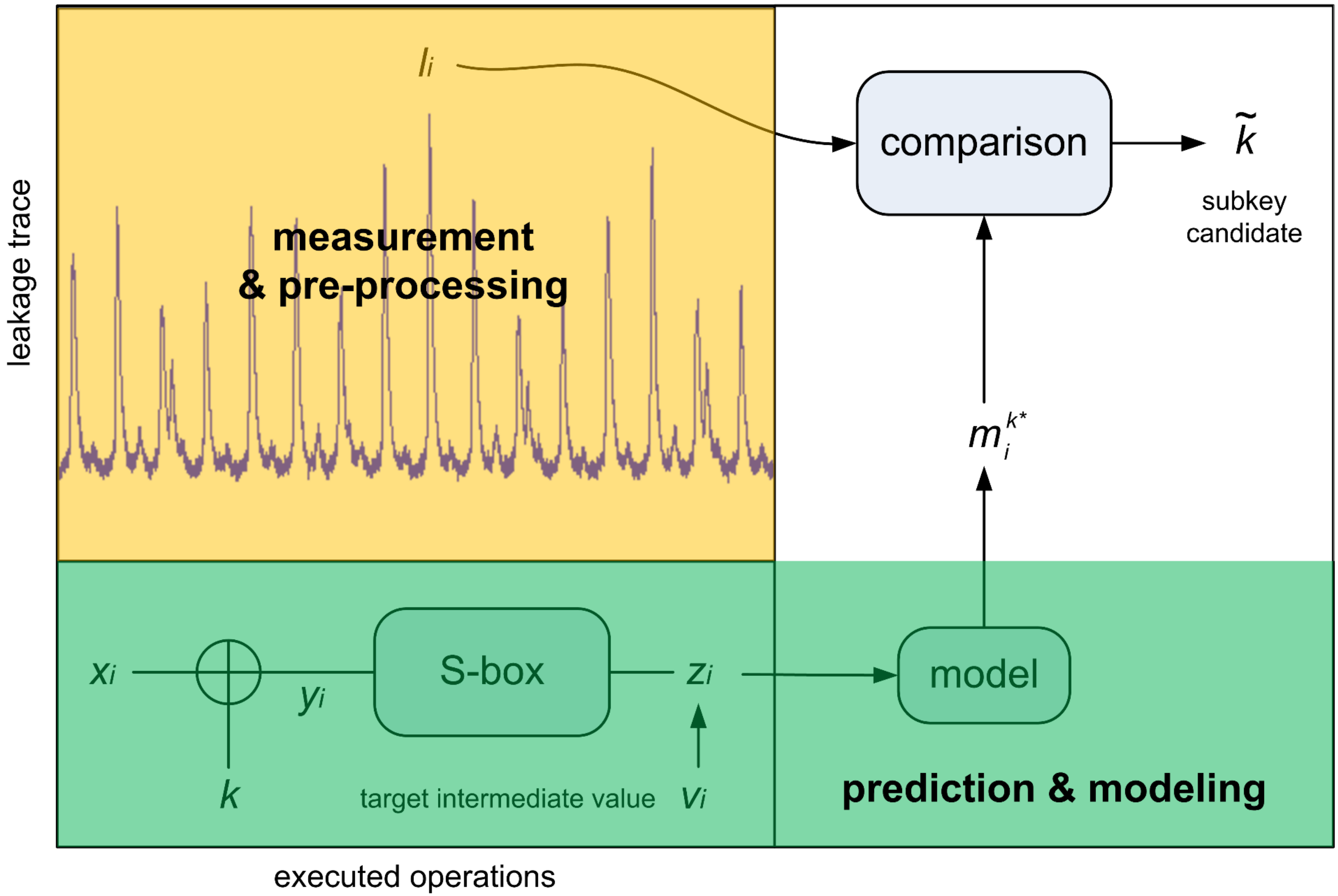
Standard DPA



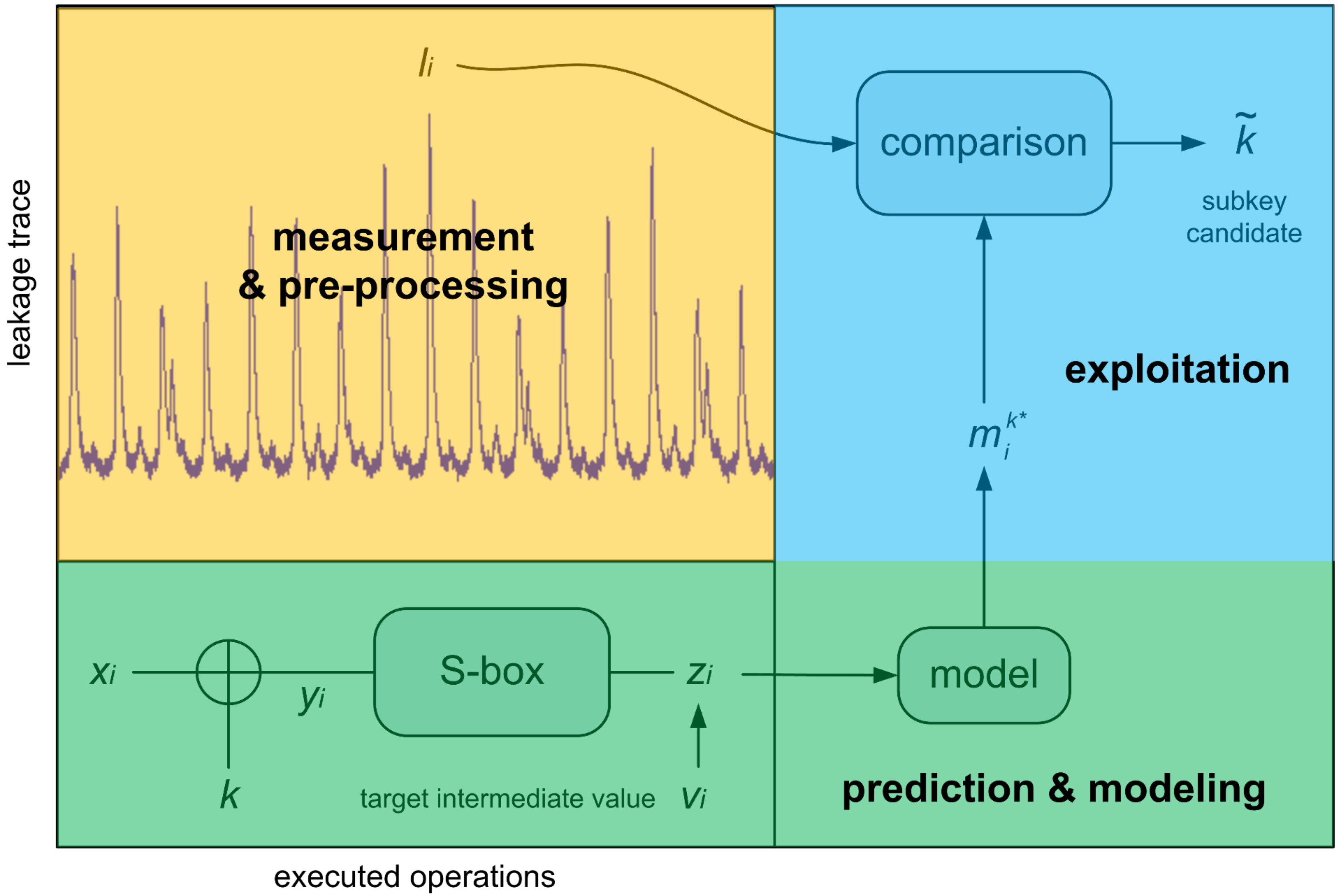
Standard DPA



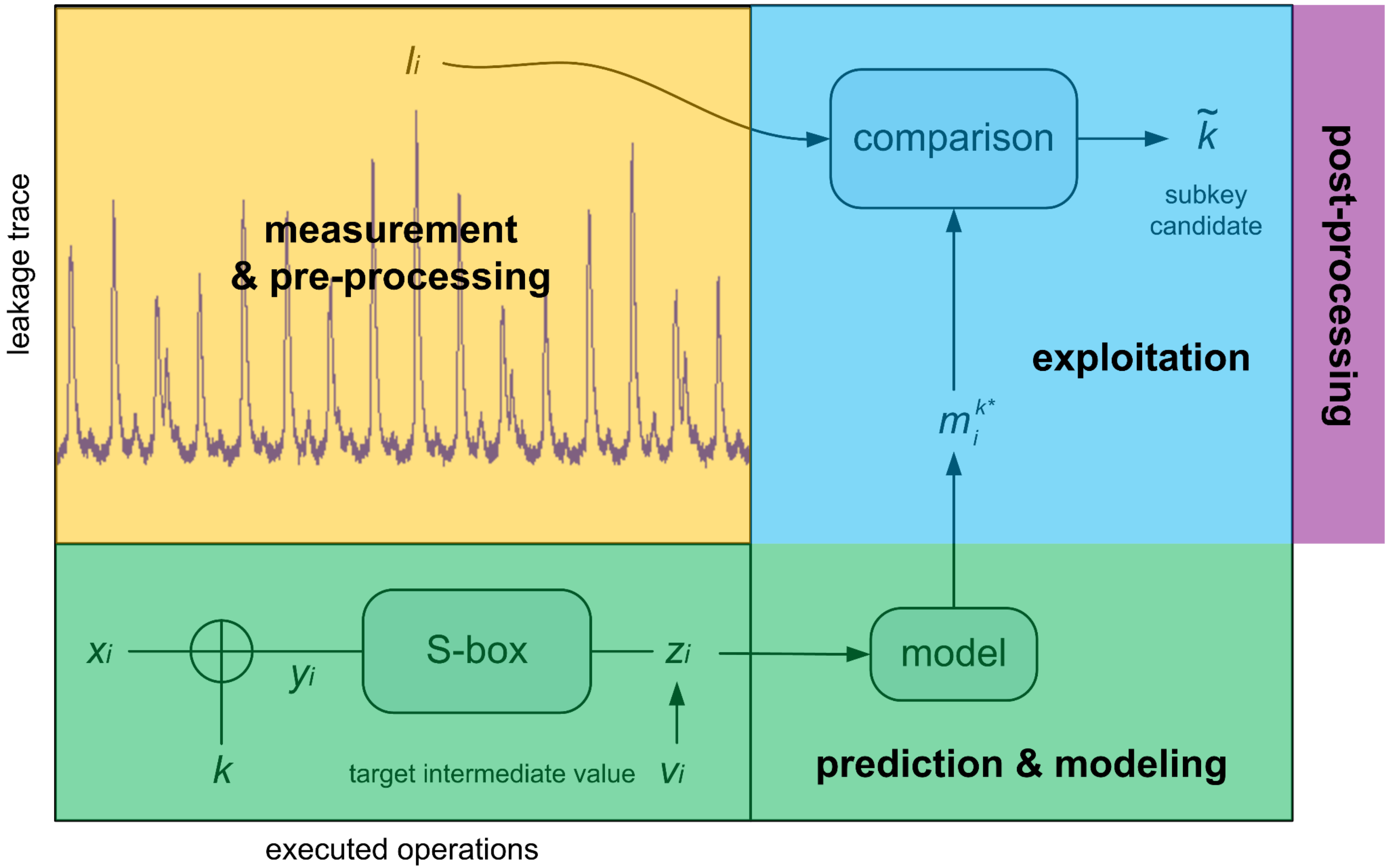
Standard DPA



Standard DPA



Standard DPA



Measurement & pre-processing

- Noise reduction via good setups (!)
- Filtering, averaging (FFT, SSA, …)
- Detection of Points-Of-Interest (POI)
- Dimensionality reduction (PCA, LDA,…)
- …

Prediction and modeling

- General case: profiled DPA
  - Build "*templates*", i.e. $\hat{f}(l_i|k,x_i)$
    - e.g. Gaussian, regression-based
  - Which directly leads to $\widehat{\Pr}[k|l_i,x_i]$

- General case: profiled DPA
  - Build "*templates*", i.e. $\hat{f}(l_i|k, x_i)$
    - e.g. Gaussian, regression-based
  - Which directly leads to $\widehat{\Pr}[k|l_i, x_i]$

- "Simplified" case: non-profiled DPA
  - Just assumes some model
  - e.g. $m_i^{k^*} = \mathrm{HW}(z_i)$

Prediction and modeling

- General case: profiled DPA
  - Build "*templates*", i.e. $\hat{f}(l_i|k, x_i)$
    - e.g. Gaussian, regression-based
  - Which directly leads to $\widehat{\Pr}[k|l_i, x_i]$

- "Simplified" case: non-profiled DPA
  - Just assumes some model
  - e.g. $m_i^{k^*} = \mathrm{HW}(z_i)$

- Separation: only profiled DPA is guaranteed to succeed against any leaking device (!)

Exploitation

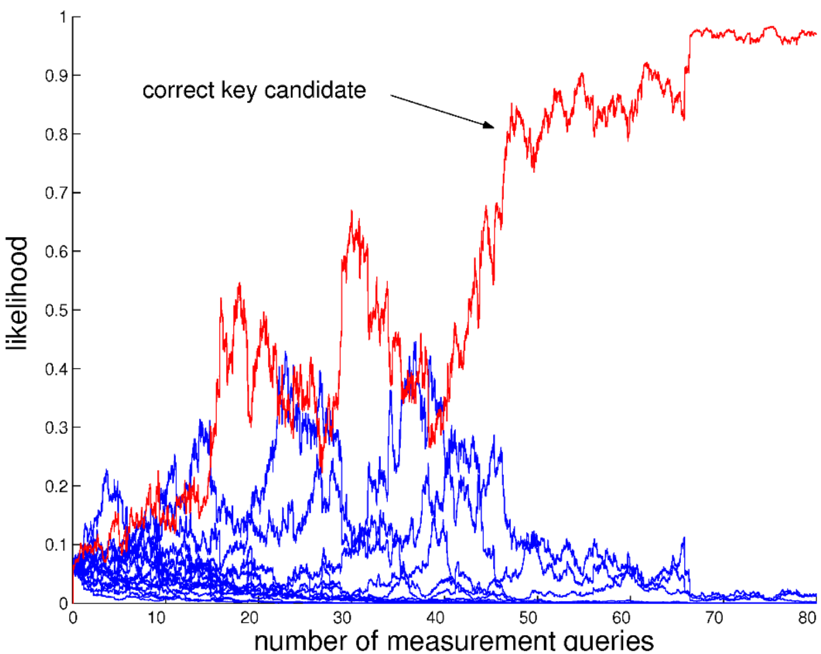
- Profiled case: maximum likelihood

Exploitation

- Profiled case: maximum likelihood

- Unprofiled case:
  - Difference-of-Means
  - Correlation (CPA)
  - « On-the-fly » regression
  - Mutual Information Analysis (MIA)
  - [...]

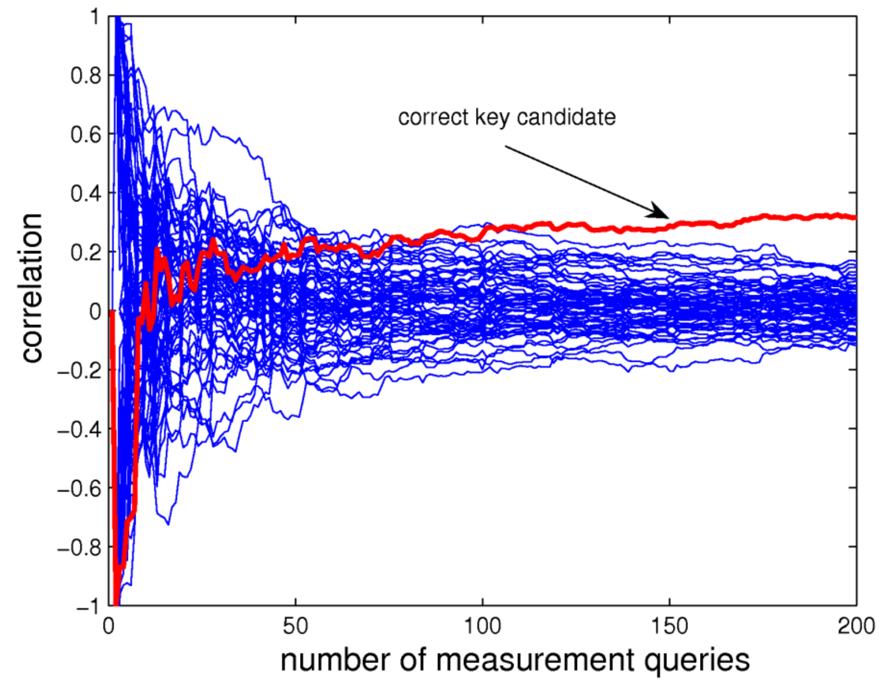
Gaussian templates



$$\tilde{k} = \underset{k^*}{\operatorname{argmax}} \prod_{i=1}^{q} \frac{1}{\sqrt{2 \cdot \pi} \cdot \sigma(L)} \cdot \exp\left(-\frac{1}{2} \cdot \left(\frac{l_i - m_i^{k^*}}{\sigma(L)}\right)^2\right)$$

- More efficient (why?)
- Outputs probabilities

CPA



$$\tilde{k} = \underset{k^*}{\operatorname{argmax}} \frac{\mathrm{E}\left(L \cdot M^{k^*}\right) - \mathrm{E}(L) \cdot \mathrm{E}\left(M^{k^*}\right)}{\sigma(L) \cdot \sigma\left(M^{k^*}\right)}$$

- Less efficient (why?)
- Outputs scores

Outline

- Link with linear cryptanalysis
- Standard Differential Power Analysis
- **Noise-based security (is not enough)**
- *CPA vs Gaussian templates*
- Post-processing the traces
- Noise amplification (aka masking)
- Conclusions & advanced topics

- **Lemma 1.** The mutual information between two normally distributed random variables X, Y with means μ_X, μ_Y and variances σ_X^2, σ_Y^2 equals:

$$\mathrm{MI}(X;Y) = -\frac{1}{2}\log_2(1 - \rho(X,Y)^2)$$

First-order CPA (I)

- **Lemma 1.** The mutual information between two normally distributed random variables X, Y with means μ_X, μ_Y and variances σ_X^2, σ_Y^2 equals:

$$\mathrm{MI}(X;Y) = -\frac{1}{2}\log_2(1-\rho(X,Y)^2)$$

- **Lemma 2.** In a CPA, the number of samples required to distinguish the corrrect key with model M_k from the other key candidates with models M_{k*} is $\propto \frac{c}{\rho(M_k,L)^2}$ (*with c a small constant depending on the SR & # of key candidates*)

First-order CPA (II)

- **Lemma 3.** Let X, Y and L be three random variables s.t. $Y = X + N_1$and $L = Y + N_2$ with N_1 and N_2 two additive noise variables. Then:

$$\rho(X, L) = \rho(X, Y) \cdot \rho(Y, L)$$

- **Lemma 3.** Let X, Y and L be three random variables s.t. $Y = X + N_1$ and $L = Y + N_2$ with N_1 and N_2 two additive noise variables. Then:

$$\rho(X, L) = \rho(X, Y) \cdot \rho(Y, L)$$

- **Lemma 4.** The correlation coefficient between the sum of n independent and identically distributed random variables and the sum of the first $m < n$ of these equals $\sqrt{m/n}$

Paper & pencil estimations (I)

- FPGA implementation of the AES
- Adversary targeting the 1st byte of key
- Hamming weight leakage function/model
- 8-bit loop architecture broken in 10 traces

Paper & pencil estimations (I)

- FPGA implementation of the AES
- Adversary targeting the 1st byte of key
- Hamming weight leakage function/model
- 8-bit loop architecture broken in 10 traces

- How does the attack data complexity scale
  - For a 32-bit architecture?
    - i.e. with 24 bits of « algorithmic noise »
  - For a 128-bit architecture?
    - i.e. with 120 bits of « algorithmic noise »

Paper & pencil estimations (II)

- Hint: $L = \mathrm{M} + \mathrm{N} = (M_P + M_U) + N$

Paper & pencil estimations (II)

- Hint: $L = \mathrm{M} + \mathrm{N} = (M_P + M_U) + N$
- Lemma 3: $\rho(M_P, L) =$

Paper & pencil estimations (II)

- Hint: $L = \mathrm{M} + \mathrm{N} = (M_P + M_U) + N$
- Lemma 3: $\rho(M_P, L) = \rho(M_P, M) \cdot \rho(M, L)$
- Lemma 4: $\rho(M_P, M) = ?$
  - For the 8-bit architecture:
  - For the 32-bit architecture:
  - For the 128-bit architecture:

Paper & pencil estimations (II)

- Hint: $L = \mathrm{M} + \mathrm{N} = (M_P + M_U) + N$
- Lemma 3: $\rho(M_P, L) = \rho(M_P, M) \cdot \rho(M, L)$
- Lemma 4: $\rho(M_P, M) = ?$
  - For the 8-bit architecture: $\sqrt{8/8}$
  - For the 32-bit architecture:
  - For the 128-bit architecture:

Paper & pencil estimations (II)

- Hint: $L = \mathrm{M} + \mathrm{N} = (M_P + M_U) + N$
- Lemma 3: $\rho(M_P, L) = \rho(M_P, M) \cdot \rho(M, L)$
- Lemma 4: $\rho(M_P, M) = ?$
  - For the 8-bit architecture: $\sqrt{8/8}$
  - For the 32-bit architecture: $\sqrt{8/32}$
  - For the 128-bit architecture: $\sqrt{8/128}$

Paper & pencil estimations (II)

- Hint: $L = \mathrm{M} + \mathrm{N} = (M_P + M_U) + N$
- Lemma 3: $\rho(M_P, L) = \rho(M_P, M) \cdot \rho(M, L)$
- Lemma 4: $\rho(M_P, M) = ?$
  - For the 8-bit architecture: $\sqrt{8/8}$
  - For the 32-bit architecture: $\sqrt{8/32}$
  - For the 128-bit architecture: $\sqrt{8/128}$
- Lemma 2: $\frac{c}{(\sqrt{8/8} \cdot \rho(M,L))^2} = 10$

Paper & pencil estimations (III)

- Data complexity for the 32-bit case:
- Data complexity for the 128-bit case:

Paper & pencil estimations (III)

- Data complexity for the 32-bit case: 40
- Data complexity for the 128-bit case: 160

- Is noise an efficient countermeasure?

Paper & pencil estimations (III)

- Data complexity for the 32-bit case: 40
- Data complexity for the 128-bit case: 160

- Is noise an efficient countermeasure?
  - 32-bit case: security $\times\ 4$, cost $\times\ ?$

Paper & pencil estimations (III)

- Data complexity for the 32-bit case: 40
- Data complexity for the 128-bit case: 160

- Is noise an efficient countermeasure?
  - 32-bit case: security $\times 4$, cost $\times 4$

- How to trade data for time?

Paper & pencil estimations (III)

- Data complexity for the 32-bit case: 40
- Data complexity for the 128-bit case: 160

- Is noise an efficient countermeasure?
  - 32-bit case: security $\times 4$, cost $\times 4$

- How to trade data for time?
  - Target more than 8 bits at once
  - Cancels (a part of) the « algorithmic noise »
  - e.g. 32-bit architecture: $\rho(M_P, M) =$

Paper & pencil estimations (III)

- Data complexity for the 32-bit case: 40
- Data complexity for the 128-bit case: 160

- Is noise an efficient countermeasure?
  - 32-bit case: security $\times 4$, cost $\times 4$

- How to trade data for time?
  - Target more than 8 bits at once
  - Cancels (a part of) the « algorithmic noise »
  - e.g. 32-bit architecture: $\rho(M_P, M) = \sqrt{32/32}$
  - (10 < data complexity < 40 because of *c*)

Outline

- Link with linear cryptanalysis
- Standard Differential Power Analysis
- Noise-based security (is not enough)
- ***CPA vs Gaussian templates***
- Post-processing the traces
- Noise amplification (aka masking)
- Conclusions & advanced topics

CPA vs. Gaussian templates

- CPA: $$\tilde{k} = \underset{k^*}{\operatorname{argmax}} \frac{\mathrm{E}\left(L \cdot M^{k^*}\right) - \mathrm{E}(L) \cdot \mathrm{E}(M^{k^*})}{\sigma(L) \cdot \sigma(M^{k^*})}$$

CPA vs. Gaussian templates

= 0 (normalization)

- CPA: $\tilde{k} = \underset{k^*}{\operatorname{argmax}} \dfrac{\mathrm{E}\left(L \cdot M^{k^*}\right) - \mathrm{E}(L) \cdot \mathrm{E}(M^{k^*})}{\sigma(L) \cdot \sigma(M^{k^*})}$

CPA vs. Gaussian templates

= 0 (normalization)

- CPA: $$\tilde{k} = \underset{k^*}{\operatorname{argmax}} \frac{\mathrm{E}\left(L \cdot M^{k^*}\right) - \mathrm{E}(L) \cdot \mathrm{E}(M^{k^*})}{\sigma(L) \cdot \sigma(M^{k^*})}$$

independent of *k**

CPA vs. Gaussian templates

= 0 (normalization)

- CPA: $$\tilde{k} = \underset{k^*}{\operatorname{argmax}} \frac{\mathrm{E}\left(L \cdot M^{k^*}\right) - \mathrm{E}(L) \cdot \mathrm{E}(M^{k^*})}{\sigma(L) \cdot \sigma(M^{k^*})}$$

independent of *k**

asymptotivally independent of k*

CPA vs. Gaussian templates

- CPA: $\tilde{k} \propto \underset{k^*}{\operatorname{argmax}} \operatorname{E}\left(L \cdot M^{k^*}\right)$

CPA vs. Gaussian templates

- CPA: $\tilde{k} \propto \underset{k^*}{\operatorname{argmax}} \operatorname{E}\left(L \cdot M^{k^*}\right)$

- Gaussian templates:

$$\tilde{k} = \underset{k^*}{\operatorname{argmax}} \prod_{i=1}^{q} \frac{1}{\sqrt{2 \cdot \pi} \cdot \sigma(L)} \cdot \exp\left(-\frac{1}{2} \cdot \left(\frac{l_i - m_i^{k^*}}{\sigma(L)}\right)^2\right)$$

CPA vs. Gaussian templates

- CPA: $\tilde{k} \propto \underset{k^*}{\operatorname{argmax}} \operatorname{E}\left(L \cdot M^{k^*}\right)$

- Gaussian templates:

$$\tilde{k} = \underset{k^*}{\operatorname{argmax}} \prod_{i=1}^{q} \frac{1}{\sqrt{2 \cdot \pi} \cdot \sigma(L)} \cdot \exp\left(-\frac{1}{2} \cdot \left(\frac{l_i - m_i^{k^*}}{\sigma(L)}\right)^2\right)$$

independent of $k*$

CPA vs. Gaussian templates

- CPA: $\tilde{k} \propto \underset{k^*}{\operatorname{argmax}} \, \mathrm{E}\left(L \cdot M^{k^*}\right)$

- Gaussian templates:

$$\tilde{k} \propto \underset{k^*}{\operatorname{argmax}} \prod_{i=1}^{q} \exp\left(-\frac{1}{2} \cdot \left(\frac{l_i - m_i^{k^*}}{\sigma(L)}\right)^2\right)$$

CPA vs. Gaussian templates

- CPA: $\tilde{k} \propto \underset{k^*}{\operatorname{argmax}} \mathrm{E}\left(L \cdot M^{k^*}\right)$

- Gaussian templates:

$$\tilde{k} \propto \underset{k^*}{\operatorname{argmin}} \mathrm{E}(L^2) - 2 \cdot \mathrm{E}\left(L \cdot M^{k^*}\right) + \mathrm{E}(\left(M^{k^*}\right)^2)$$

CPA vs. Gaussian templates

- CPA: $\tilde{k} \propto \underset{k^*}{\operatorname{argmax}} \, \mathrm{E}\left(L \cdot M^{k^*}\right)$

- Gaussian templates:

$$\tilde{k} \propto \underset{k^*}{\operatorname{argmin}} \, \mathrm{E}(L^2) - 2 \cdot \mathrm{E}\left(L \cdot M^{k^*}\right) + \mathrm{E}(\left(M^{k^*}\right)^2)$$

independent of *k**

CPA vs. Gaussian templates

- CPA: $\tilde{k} \propto \underset{k^*}{\operatorname{argmax}} \, \mathrm{E}\left(L \cdot M^{k^*}\right)$

- Gaussian templates:

$$\tilde{k} \propto \underset{k^*}{\operatorname{argmin}} \, \mathrm{E}(L^2) - 2 \cdot \mathrm{E}\left(L \cdot M^{k^*}\right) + \mathrm{E}(\left(M^{k^*}\right)^2)$$

independent of *k**

asymptotivally independent of k*

CPA vs. Gaussian templates

- CPA: $\tilde{k} \propto \underset{k^*}{\operatorname{argmax}} \mathrm{E}\left(L \cdot M^{k^*}\right)$
- Gaussian templates: $\tilde{k} \propto \underset{k^*}{\operatorname{argmax}} \mathrm{E}\left(L \cdot M^{k^*}\right)$

CPA vs. Gaussian templates

- CPA: $\tilde{k} \propto \underset{k^*}{\operatorname{argmax}} \mathrm{E}\left(L \cdot M^{k^*}\right)$
- Gaussian templates: $\tilde{k} \propto \underset{k^*}{\operatorname{argmax}} \mathrm{E}\left(L \cdot M^{k^*}\right)$

$\Rightarrow$ Both attacks are asymtotically equivalent

- For 1st-order leakages
  - i.e. unprotected implementations
- Given they exploit the same model

CPA vs. Gaussian templates

- CPA: $\tilde{k} \propto \underset{k^*}{\operatorname{argmax}} \operatorname{E}\left(L \cdot M^{k^*}\right)$
- Gaussian templates: $\tilde{k} \propto \underset{k^*}{\operatorname{argmax}} \operatorname{E}\left(L \cdot M^{k^*}\right)$

$\Rightarrow$ Both attacks are asymtotically equivalent
- For 1st-order leakages
  - i.e. unprotected implementations
- Given they exploit the same model

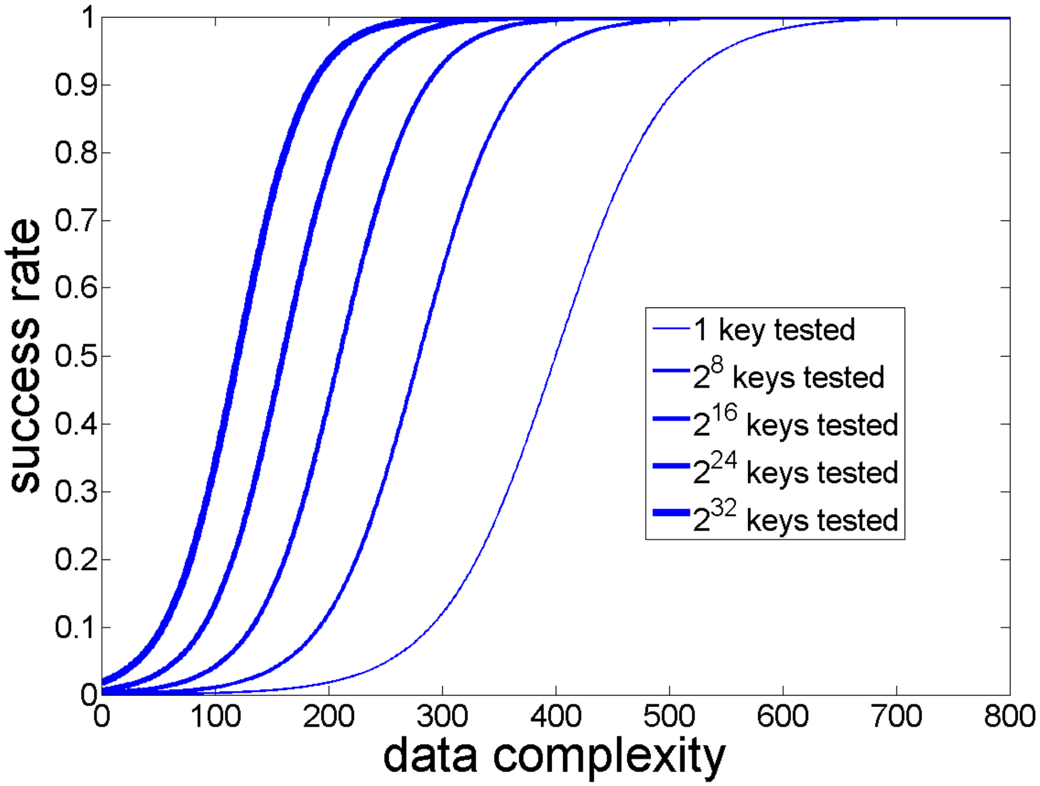
$\Rightarrow$ Gaussian templates outperforms CPA because it (usually) exploits a better (profiled) **model**

Outline

- Link with linear cryptanalysis
- Standard Differential Power Analysis
- Noise-based security (is not enough)
- *CPA vs Gaussian templates*
- **Post-processing the traces**
- Noise amplification (aka masking)
- Conclusions & advanced topics

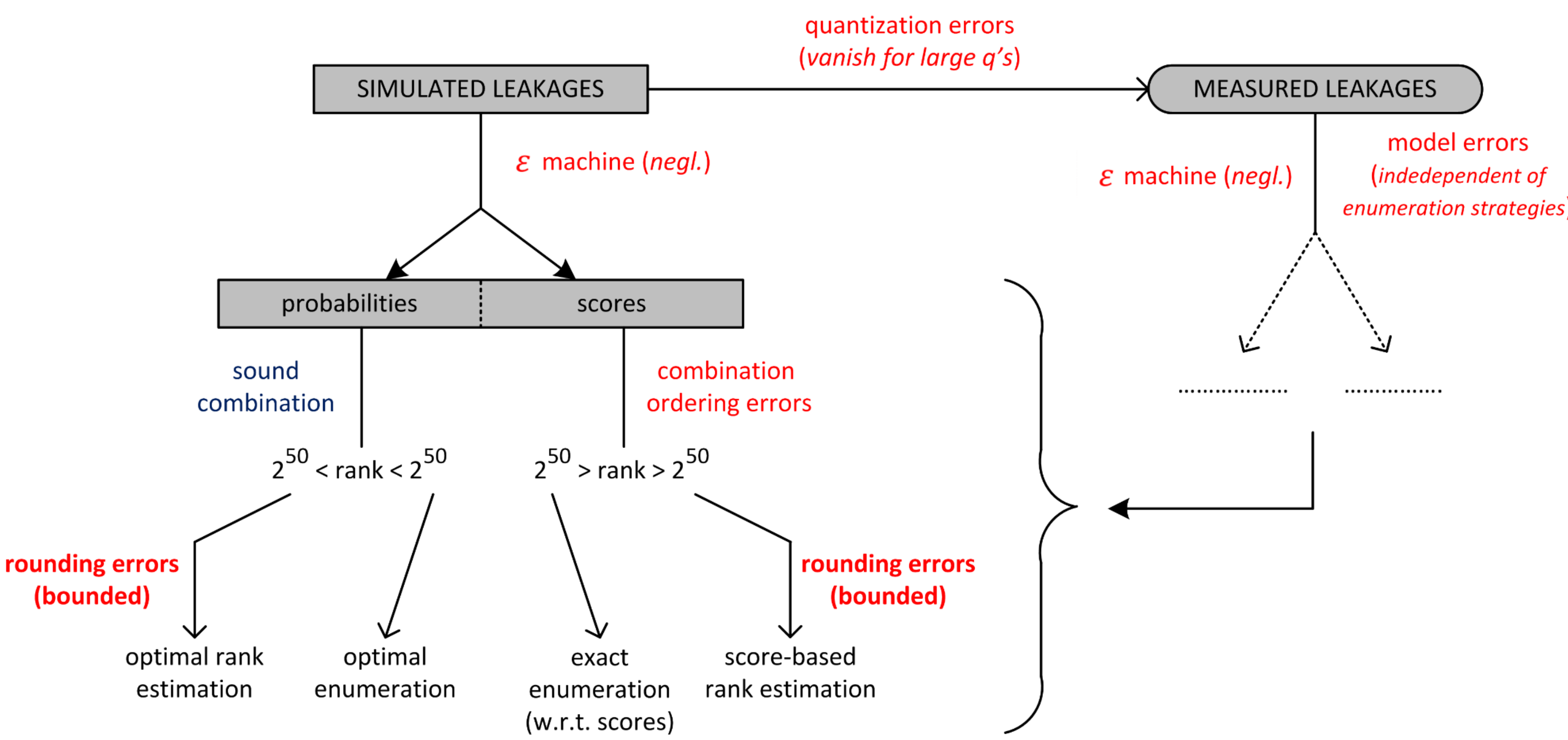
Exploiting offline computing power

- Key enumeration



- & rank estimation if key is beyond enumeration

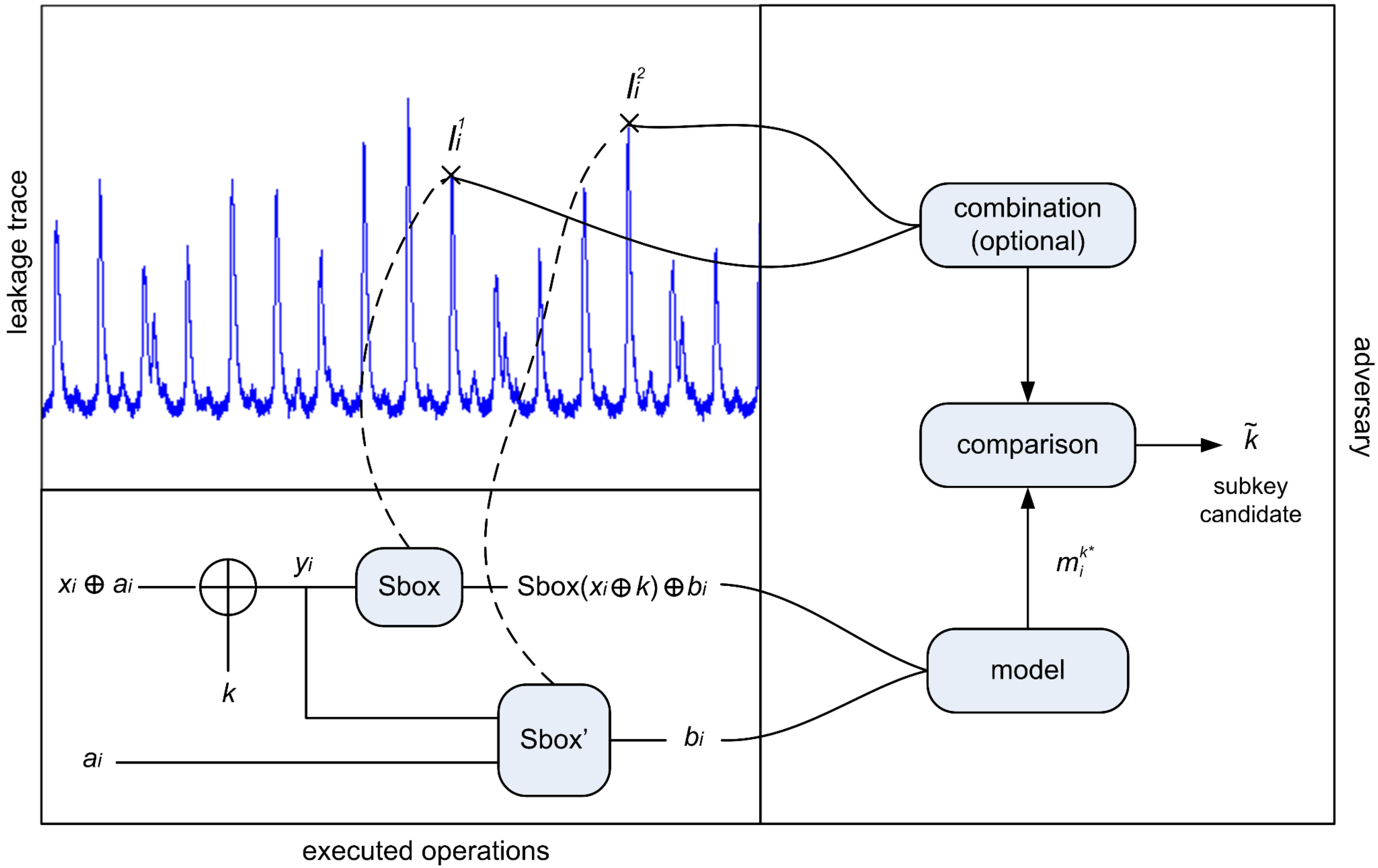
- Enumeration / rank estimation errors



Outline

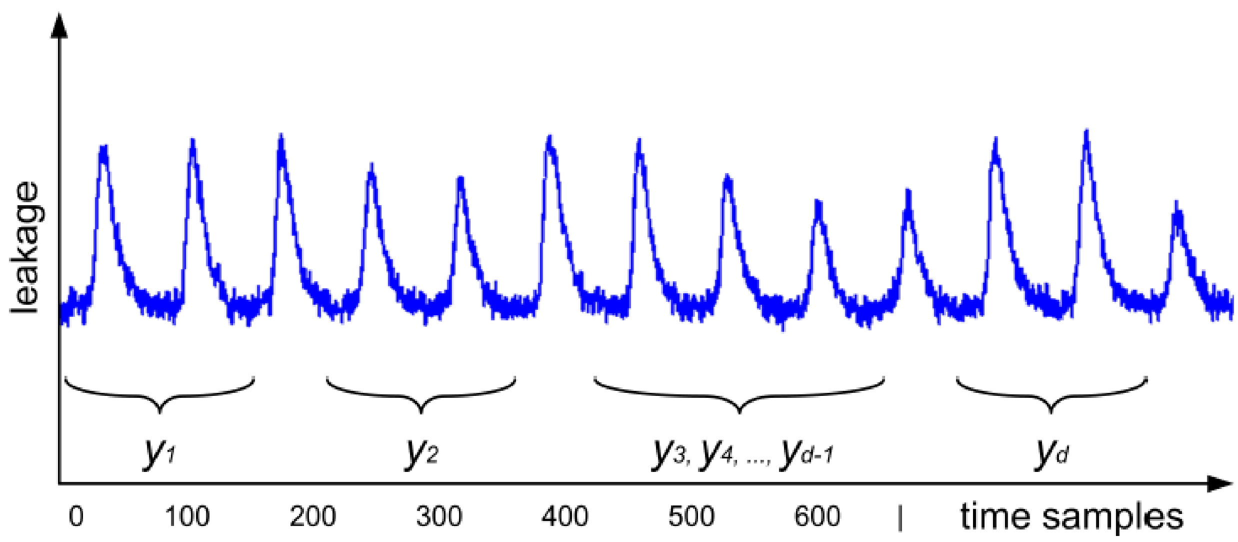
- Link with linear cryptanalysis
- Standard Differential Power Analysis
- Noise-based security (is not enough)
- *CPA vs Gaussian templates*
- Post-processing the traces
- **Noise amplification (aka masking)**
- Conclusions & advanced topics

Masking & 2nd-order DPA



More generally (I)

- Let $z = S(x \oplus k) = S(y)$ be a leaking S-box
- Let $y = y_1 \oplus y_2 \oplus \cdots \oplus y_d$ be a sharing of y



- Perform computations on "shared" variables

More generally (II)

- Linear operations: $\mathrm{f}(\mathrm{a}) = \mathrm{f}(a_1) \oplus \mathrm{f}(a_2) \oplus \cdots \oplus \mathrm{f}(a_d)$

More generally (II)

- Linear operations: $\mathrm{f(a)} = \mathrm{f}(a_1) \oplus \mathrm{f}(a_2) \oplus \cdots \oplus \mathrm{f}(a_d)$

- Multiplications: $\mathrm{c} = a \times b$ in three steps

More generally (II)

- Linear operations: $\mathrm{f(a)} = \mathrm{f}(a_1) \oplus \mathrm{f}(a_2) \oplus \cdots \oplus \mathrm{f}(a_d)$

- Multiplications: $\mathrm{c} = a \times b$ in three steps

$$\begin{bmatrix} a_1 b_1 & a_1 b_2 & a_1 b_3 \\ a_2 b_1 & a_2 b_2 & a_2 b_3 \\ a_3 b_1 & a_3 b_2 & a_3 b_3 \end{bmatrix}$$

partial products

More generally (II)

- Linear operations: $\mathrm{f(a)} = \mathrm{f}(a_1) \oplus \mathrm{f}(a_2) \oplus \cdots \oplus \mathrm{f}(a_d)$

- Multiplications: $\mathrm{c} = a \times b$ in three steps

$$\begin{bmatrix} a_1 b_1 & a_1 b_2 & a_1 b_3 \\ a_2 b_1 & a_2 b_2 & a_2 b_3 \\ a_3 b_1 & a_3 b_2 & a_3 b_3 \end{bmatrix} + \begin{bmatrix} 0 & r_1 & r_2 \\ -r_1 & 0 & r_3 \\ -r_2 & r_3 & 0 \end{bmatrix}$$

partial products refreshing

- Linear operations: $\mathrm{f(a)} = \mathrm{f}(a_1) \oplus \mathrm{f}(a_2) \oplus \cdots \oplus \mathrm{f}(a_d)$

- Multiplications: $\mathrm{c} = a \times b$ in three steps

$$\underbrace{\begin{bmatrix} a_1b_1 & a_1b_2 & a_1b_3 \\ a_2b_1 & a_2b_2 & a_2b_3 \\ a_3b_1 & a_3b_2 & a_3b_3 \end{bmatrix}}_{\text{partial products}} + \underbrace{\begin{bmatrix} 0 & r_1 & r_2 \\ -r_1 & 0 & r_3 \\ -r_2 & r_3 & 0 \end{bmatrix}}_{\text{refreshing}} \Rightarrow \underbrace{\begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix}}_{\text{compression}}$$

More generally (II)

- Linear operations: $\mathrm{f(a)} = \mathrm{f}(a_1) \oplus \mathrm{f}(a_2) \oplus \cdots \oplus \mathrm{f}(a_d)$

- Multiplications: $\mathrm{c} = a \times b$ in three steps

$$\begin{bmatrix} a_1 b_1 & a_1 b_2 & a_1 b_3 \\ a_2 b_1 & a_2 b_2 & a_2 b_3 \\ a_3 b_1 & a_3 b_2 & a_3 b_3 \end{bmatrix} + \begin{bmatrix} 0 & r_1 & r_2 \\ -r_1 & 0 & r_3 \\ -r_2 & r_3 & 0 \end{bmatrix} \Rightarrow \begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix}$$

partial products refreshing compression

$\Rightarrow$ Quadratic overheads & randomness

Main theorem (informal)

- Assume leakage variables $L_{Z_i} = \delta(Z_i) + N$ s.t.
  - $\mathrm{MI}(Z_i; L_{Z_i}) \leq \frac{c}{d^2}$ (why d^2?)
  - The leakages of the shares are independent
- For a masking scheme with *d* shares
- And an adversary using *m* measurements

- Then: $\mathrm{SR} \leq 1 - \left(1 - \mathrm{MI}(Z_i; L_{Z_i})^d\right)^m$

Main theorem (informal)

- Assume leakage variables $L_{Z_i} = \delta(Z_i) + N$ s.t.
  - $\mathrm{MI}(Z_i; L_{Z_i}) \leq \frac{c}{d^2}$ (multiplications)
  - The leakages of the shares are independent
- For a masking scheme with *d* shares
- And an adversary using *m* measurements

- Then: $\mathrm{SR} \leq 1 - \left(1 - \mathrm{MI}(Z_i; L_{Z_i})^d\right)^m$

Main theorem (informal)

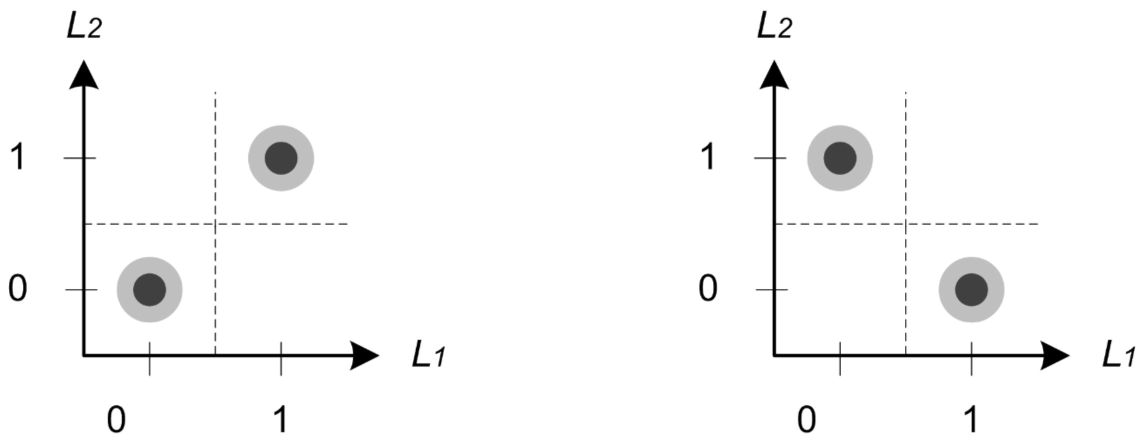
- Assume leakage variables $L_{Z_i} = \delta(Z_i) + N$ s.t.
  - $\mathrm{MI}(Z_i; L_{Z_i}) \leq \frac{c}{d^2}$ (multiplications)
  - The leakages of the shares are independent
- For a masking scheme with *d* shares
- And an adversary using *m* measurements

- Then: $\mathrm{SR} \leq 1 - \left(1 - \mathrm{MI}(Z_i; L_{Z_i})^d\right)^m$

- For $m = 1$, $\mathrm{SR} \leq \mathrm{MI}(Z_i; L_{Z_i})^d \propto (\sigma_N^2)^d$
- (Intuitively ≈ "noisy" piling up lemma)

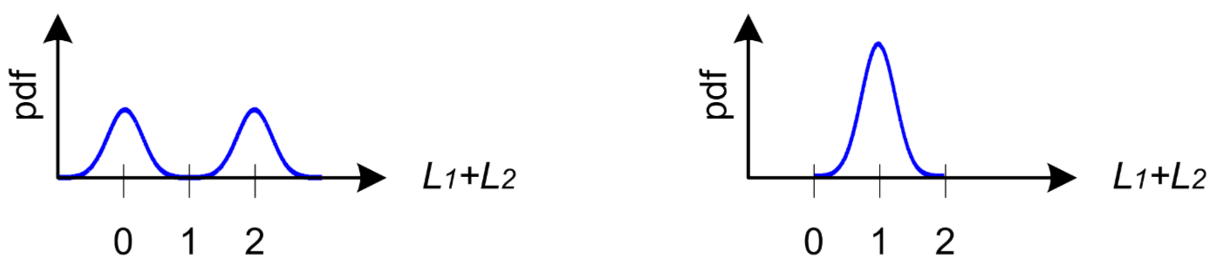
Statistical intuition

- 1-bit, 2-shares example



(a) $Z = 0$, serial.

(b) $Z = 1$, serial.



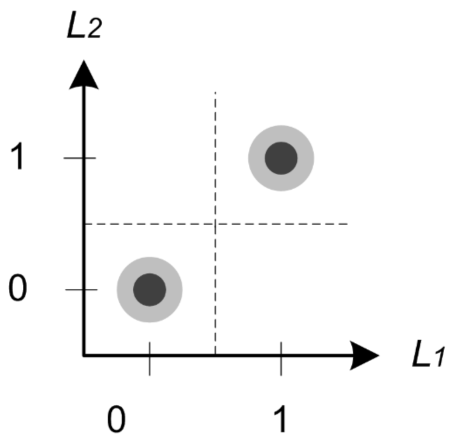
(c) $Z = 0$, parallel.

(d) $Z = 1$, parallel.

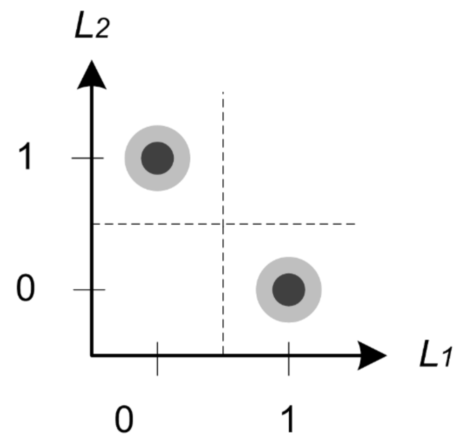
Statistical intuition

- 1-bit, 2-shares example

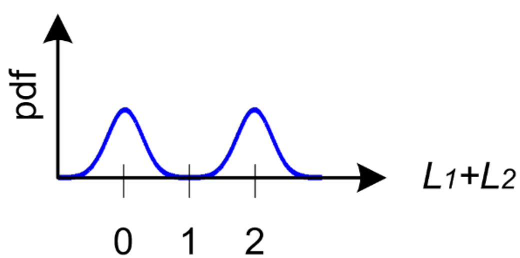
key-independent means



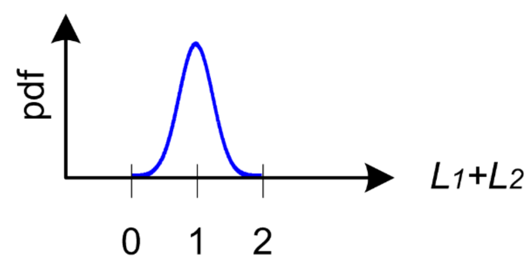
(a) Z = 0, serial.



(b) Z = 1, serial.



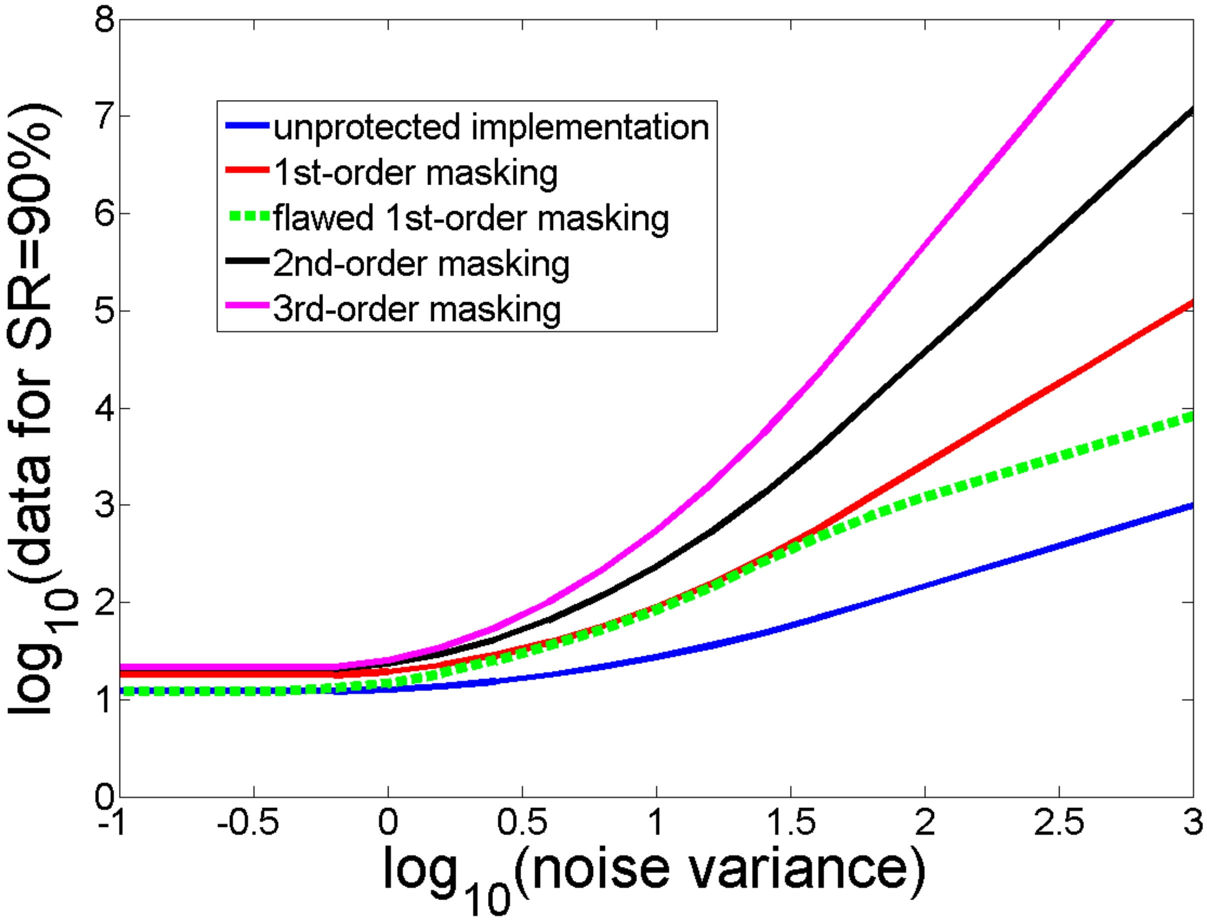
(c) Z = 0, parallel.



(d) Z = 1, parallel.

Information theoretic intuition

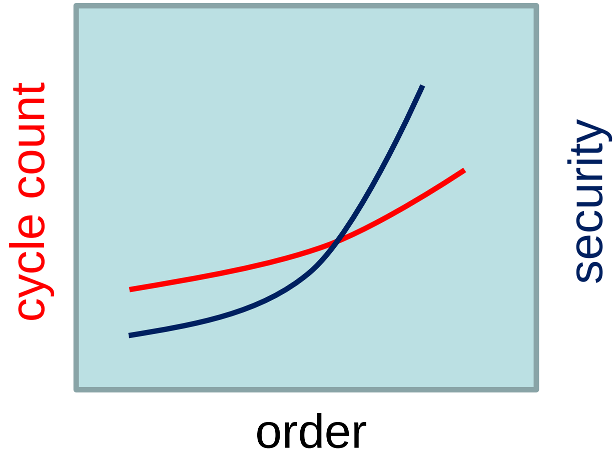
- Slope of the IT curves = d (*if independent leaks*)



Wrapping up

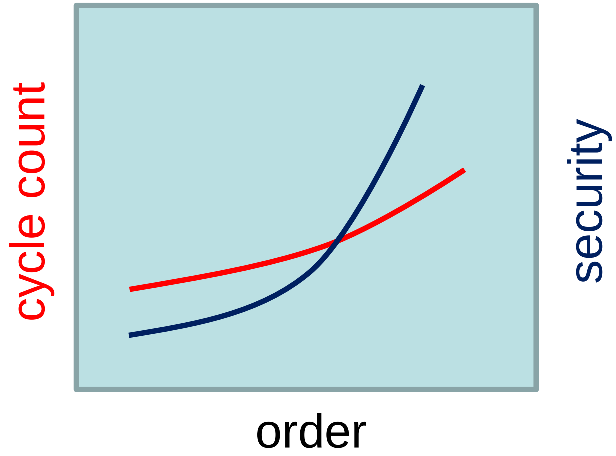
- Is masking an efficient countermeasure?
  - Security (data) is exponential in d
  - Cost is [...]

- Is masking an efficient countermeasure?
  - Security (data) is exponential in d
  - Cost is [...] quadratic in d



Wrapping up

- Is masking an efficient countermeasure?
  - Security (data) is exponential in d
  - Cost is [...] quadratic in d



- If the leakages are noisy and independent (!)

Wrapping up

- Is masking an efficient countermeasure?
  - Security (data) is exponential in d
  - Cost is [...] quadratic in d

- If the leakages are noisy and independent (!)
- How does the time complexity scale in d?

- Is masking an efficient countermeasure?
  - Security (data) is exponential in d
  - Cost is [...] quadratic in d

- If the leakages are noisy and independent (!)
- How does the time complexity scale in d?
  - Depends on the implem. (e.g. serial or //)

Outline

- Link with linear cryptanalysis
- Standard Differential Power Analysis
- Noise-based security (is not enough)
- *CPA vs Gaussian templates*
- Post-processing the traces
- Noise amplification (aka masking)
- **Conclusions & advanced topics**

Conclusions

- Unprotected implementations are easy targets
  - Physical biases are usually large
- Noise is an ingredient – not the solution
- Noise amplification is possible (via masking)
  - But is hard to implement securely

Conclusions

- Unprotected implementations are easy targets
  - Physical biases are usually large
- Noise is an ingredient – not the solution
- Noise amplification is possible (via masking)
  - But is hard to implement securely

- More generally, efficient countermeasures against side-channel attacks always combine two ingredients: sound (*falsifiable*) hardware assumptions & mathematical amplification

Advanced topics

- More elaborate/powerful attacks
  - Algebraic/analytical SCA
- Simpler/cheaper evaluations
  - Leakage detection
- Worst-case evaluations
  - Model certification
- Secure & efficient masking
  - Inner product masking
  - Threshold implementations (HW)
  - Formal verification (SW)
- Security by design (leakage-resilience)

THANKS

http://perso.uclouvain.be/fstandae/

Related publications & further readings. Standard DPA (slide 5). Stefan Mangard, Elisabeth Oswald, François-Xavier Standaert: *One for all - all for one: unifying standard differential power analysis attacks*. IET Information Security 5(2): 100-110 (2011). **Pre-processing (slide 6).** Victor Lomné, Emmanuel Prouff, Thomas Roche: *Behind the Scene of Side Channel Attacks.* ASIACRYPT (1) 2013: 506-525. ***Filtering*.** Santos Merino Del Pozo, François-Xavier Standaert: *Blind Source Separation from Single Measurements Using Singular Spectrum Analysis*. CHES 2015: 42-59. ***POI detection*.** Oscar Reparaz, Benedikt Gierlichs, Ingrid Verbauwhede: *Selecting Time Samples for Multivariate DPA Attacks*. CHES 2012: 155-174. François Durvaux, François-Xavier Standaert, Nicolas Veyrat-Charvillon, Jean-Baptiste Mairy, Yves Deville: *Efficient Selection of Time Samples for Higher-Order DPA with Projection Pursuits.* COSADE 2015: 34-50. ***Dimensionality reduction*.** Cédric Archambeau, Eric Peeters, François-Xavier Standaert, Jean-Jacques Quisquater: *Template Attacks in Principal Subspaces*. CHES 2006: 1-14. François-Xavier Standaert, Cédric Archambeau: *Using Subspace-Based Template Attacks to Compare and Combine Power and Electromagnetic Information Leakages*. CHES 2008: 411-425. **Prediction and modeling (slide 7). *Profiled DPA*.** Suresh Chari, Josyula R. Rao, Pankaj Rohatgi: *Template Attacks*. CHES 2002: 13-28. Werner Schindler, Kerstin Lemke, Christof Paar: *A Stochastic Model for Differential Side Channel Cryptanalysis*. CHES 2005: 30-46. ***Separation result.*** Carolyn Whitnall, Elisabeth Oswald, François-Xavier Standaert: *The Myth of Generic DPA...and the Magic of Learning*. CT-RSA 2014: 183-205. **Exploitation (slide 8).** Omar Choudary, Markus G. Kuhn: *Efficient Template Attacks.* CARDIS 2013: 253-270. Paul C. Kocher, Joshua Jaffe, Benjamin Jun: *Differential Power Analysis*. CRYPTO 1999: 388-397. Eric Brier, Christophe Clavier, Francis Olivier: *Correlation Power Analysis with a Leakage Model.* CHES 2004: 16-29. Julien Doget, Emmanuel Prouff, Matthieu Rivain, François-Xavier Standaert: *Univariate side channel attacks and leakage modeling*. J. Cryptographic Engineering 1(2): 123-144 (2011). Lejla Batina, Benedikt Gierlichs, Emmanuel Prouff, Matthieu Rivain, François-Xavier Standaert, Nicolas Veyrat-Charvillon: *Mutual Information Analysis: a Comprehensive Study*. J. Cryptology 24(2): 269-291 (2011). **First-order CPA (slides 10-11).** Stefan Mangard, Elisabeth Oswald, François-Xavier Standaert: *One for all - all for one: unifying standard differential power analysis attacks*. IET Information Security 5(2): 100-110 (2011). François-Xavier Standaert, Eric Peeters, Gaël Rouvroy, Jean-Jacques Quisquater, *An Overview of Power Analysis Attacks Against Field Programmable Gate Arrays*, Proceedings of the IEEE, 94(2): 383-394 (2006). **Trading data for time (slide 14).** Luke Mather, Elisabeth Oswald, Carolyn Whitnall: *Multi-target DPA Attacks: Pushing DPA Beyond the Limits of a Desktop Computer.* ASIACRYPT (1) 2014: 243-261. **CPA vs. Gaussian templates (slide 15).** Stefan Mangard, Elisabeth Oswald, François-Xavier Standaert: *One for all - all for one: unifying standard differential power analysis attacks*. IET Information Security 5(2): 100-110 (2011). **Key enumeration/rank estimation (slide 16).** Nicolas Veyrat-Charvillon, Benoît Gérard, Mathieu Renauld, François-Xavier Standaert: *An Optimal Key Enumeration Algorithm and Its Application to Side-Channel Attacks*. Selected Areas in Cryptography 2012: 390-406. Nicolas Veyrat-Charvillon, Benoît Gérard, François-Xavier Standaert: *Security Evaluations Beyond Computing Power: How to Analyze Side-Channel Attacks you Cannot Mount?* EUROCRYPT 2013: 126-141. Cezary Glowacz, Vincent Grosso, Romain Poussier, Joachim Schüth, François-Xavier Standaert: *Simpler and More Efficient Rank Estimation for Side-Channel Security Assessment*. FSE 2015: 117-129. Daniel P. Martin, Jonathan F. O'Connell, Elisabeth Oswald, Martijn Stam: *Counting Keys in Parallel After a Side Channel Attack*. ASIACRYPT (2) 2015: 313-337.**Key enumeration/rank estimation errors (slide 17).** Romain Poussier, Vincent Grosso, François-Xavier Standaert: *Comparing Approaches to Rank Estimation for Side-Channel Security Evaluations*. CARDIS 2015: 125-142. **Masking (slides 19-20).** Yuval Ishai, Amit Sahai, David Wagner: *Private Circuits: Securing Hardware against Probing Attacks*. CRYPTO 2003: 463-481. Matthieu Rivain, Emmanuel Prouff: *Provably Secure Higher-Order Masking of AES*. CHES 2010: 413-427. **Masking proof (slide 21).** Alexandre Duc, Sebastian Faust, François-Xavier Standaert: *Making Masking Security Proofs Concrete - Or How to Evaluate the Security of Any Leaking Device*. EUROCRYPT (1) 2015: 401-429. **Advanced topics (slide 26). *Algebraic/analytical attacks.*** Mathieu Renauld, François-Xavier Standaert, Nicolas Veyrat-Charvillon: *Algebraic Side-Channel Attacks on the AES: Why Time also Matters in DPA*. CHES 2009: 97-111. Nicolas Veyrat-Charvillon, Benoît Gérard, François-Xavier Standaert: *Soft Analytical Side-Channel Attacks.* ASIACRYPT (1) 2014: 282-296. Vincent Grosso, François-Xavier Standaert: ASCA, SASCA and DPA with Enumeration: Which One Beats the Other and When? ASIACRYPT (2) 2015: 291-312. ***Leakage detection.*** Luke Mather, Elisabeth Oswald, Joe Bandenburg, Marcin Wójcik: *Does My Device Leak Information? An a priori Statistical Power Analysis of Leakage Detection Tests.* ASIACRYPT (1) 2013: 486-505. François Durvaux, François-Xavier Standaert: *From Improved Leakage Detection to the Detection of Points of Interests in Leakage Traces*. EUROCRYPT (1) 2016: 240-262. ***Model certification.*** François Durvaux, François-Xavier Standaert, Nicolas Veyrat-Charvillon: *How to Certify the Leakage of a Chip?* EUROCRYPT 2014: 459-476. **Secure and efficient masking. *Inner Product Masking.*** Josep Balasch, Sebastian Faust, Benedikt Gierlichs: *Inner Product Masking Revisited*. EUROCRYPT (1) 2015: 486-510. ***Threshold implementations.*** Svetla Nikova, Vincent Rijmen, Martin Schläffer: *Secure Hardware Implementation of Nonlinear Functions in the Presence of Glitches*. J. Cryptology 24(2): 292-321 (2011). ***Formal verification*.** Gilles Barthe, Sonia Belaïd, François Dupressoir, Pierre-Alain Fouque, Benjamin Grégoire, Pierre-Yves Strub: *Verified Proofs of Higher-Order Masking*. EUROCRYPT (1) 2015: 457-485. **Leakage-resilience.** see next talk.