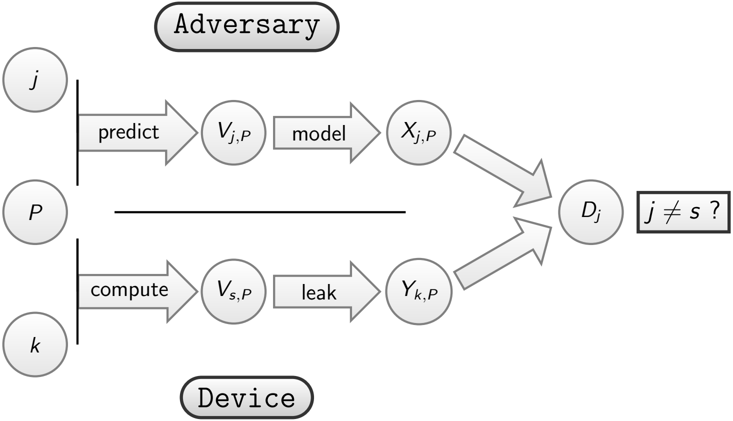
Generic Side-Channel Distinguishers: Improvements and Limitations

N. Veyrat-Charvillon and F-X. Standaert

UCL Crypto Group, Université catholique de Louvain

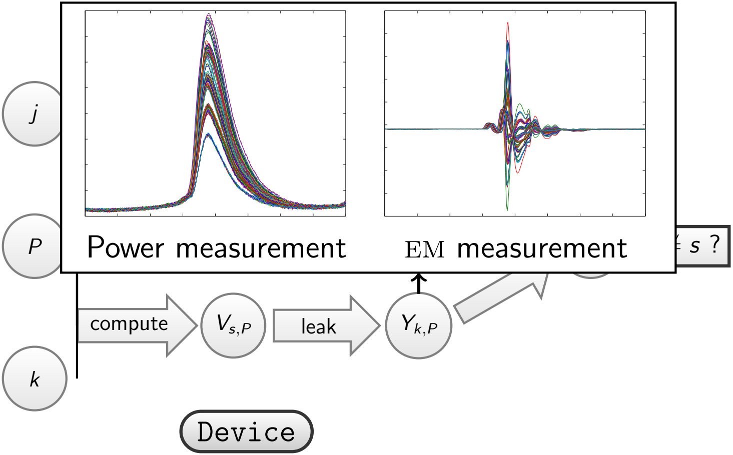
CRYPTO 2011, August 16

Evaluating Implementations With DPA Attacks



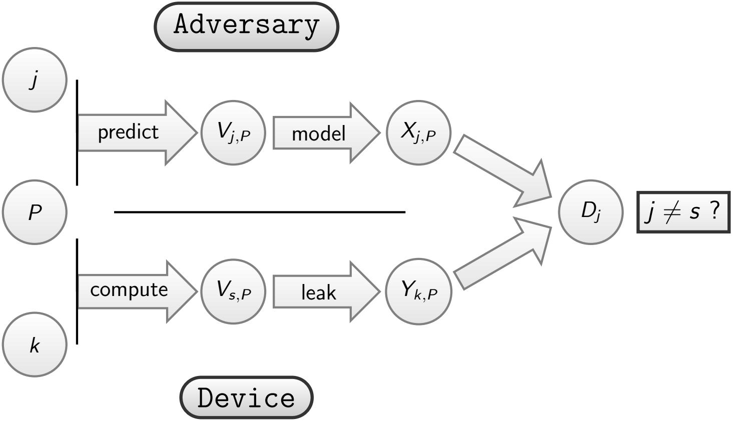
Main ingredients: leakage model & dependency test

Evaluating Implementations With DPA Attacks



Main ingredients: leakage model & dependency test

Evaluating Implementations With DPA Attacks



Main ingredients: leakage model & dependency test

Ingredient 1: Leakage Models

Two adversarial scenarios:

- Profiled case: preliminary estimation of the leakage pdf
  - Gaussian distribution
  - Mixture model
  - . . .
- Non-profiled case: assumption on the leakages pdf (based on engineering intuition)
  - Hamming weight/distance
  - Linear (or quadratic, . . .) function of bits
  - Identity function
  - . . .

Ingredient 2: Dependency Test

Different adversarial choices depending on:

- Number of samples used: univariate or multivariate
- Moment of the pdf exploited: mean, variance, ...
- Type of dependency tested: linear, monotonic, ...

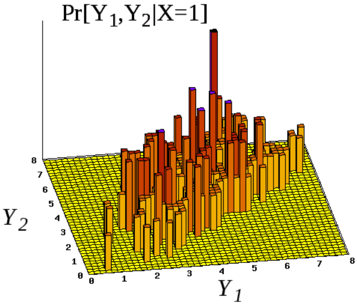
Existing Tests: Efficiency vs. Genericity

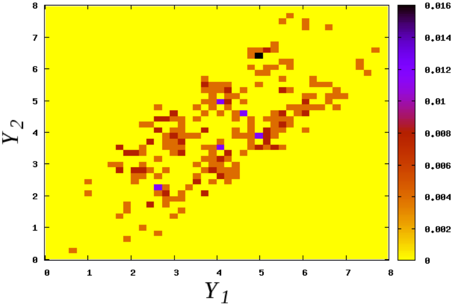
Pearson correlation	univariate mean linear
Spearman correlation	univariate mean monotonic
Least Square Regression	multivariate mean MV linear
Mutual information	multivariate all moments any dependency

↑ Efficient

↓ Generic

Additional Concern: Choice of Parameters





- e.g. number of histogram bins
- (or kernel bandwidth, number of mixture components)

Open questions

- Question 1: can we design a generic side-channel distinguisher that is free of parameters?
- Question 2: can we evaluate side-channel attacks with non-profiled distinguishers only?

Our Contributions

w.r.t. question 1, a new distinguisher based on:

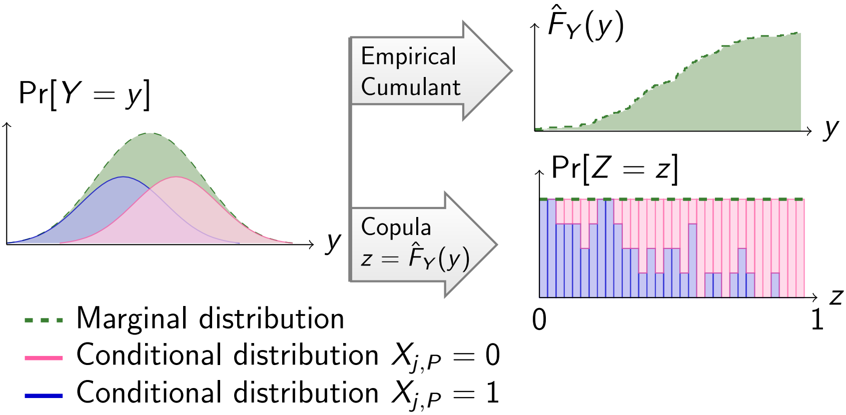
1. leakage space reduction through copulas
2. dimensionality reduction using spacings
3. non-parametric uniformity test

w.r.t. question 2: empirical evaluations showing:

1. the efficiency of the new generic test
2. the necessity of profiled security evaluations

The new distinguisher

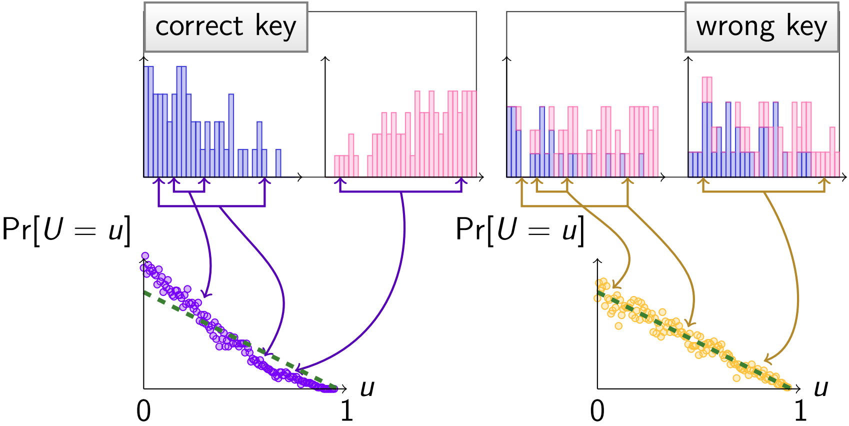
Tool 1: Leakage Space Reduction



╌ Marginal distribution
— Conditional distribution $X_{j,P} = 0$
— Conditional distribution $X_{j,P} = 1$

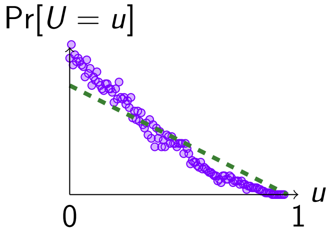
+ Cumulants are easier to estimate than pdfs
+ Projected marginal distribution is uniform

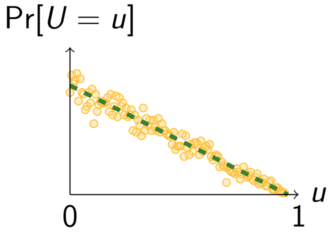
Tool 2: Leakage Partition and Distance Sampling

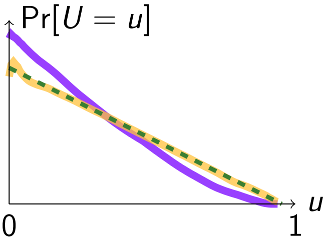


- Wrong key candidates should behave like uniform
- All model values contribute to the estimation

Tool 3: Smoothing and Evaluation







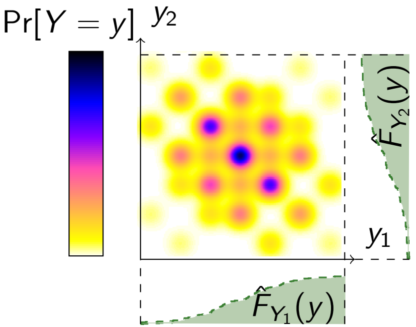
- - - Theoretical distribution

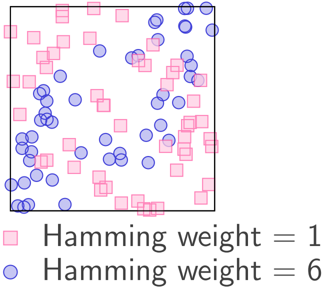
● — Correct key

● — Wrong key

+ No parameters

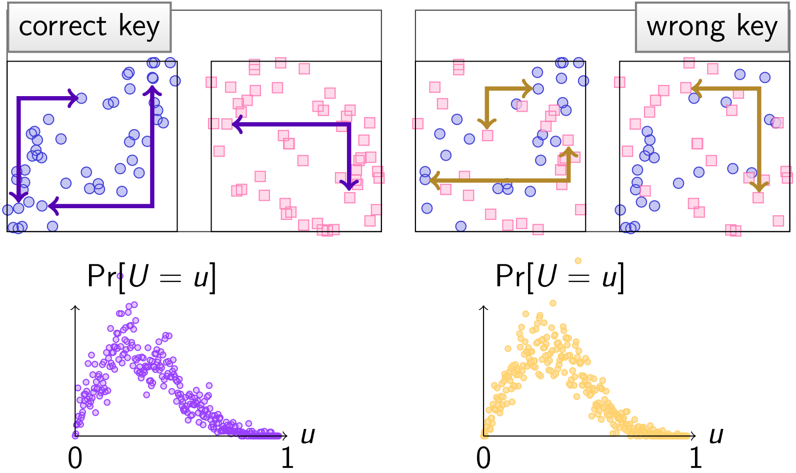
2D case: Leakage Space Reduction





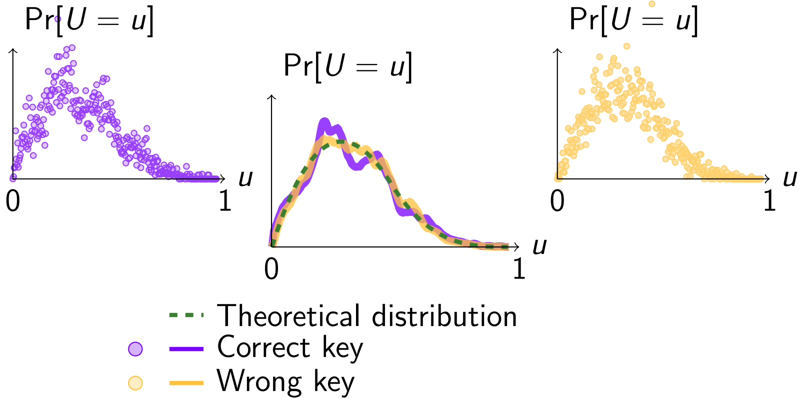
+ Copula transform preserves multivariate dependencies

2D case: Leakage Partition and Distance Sampling



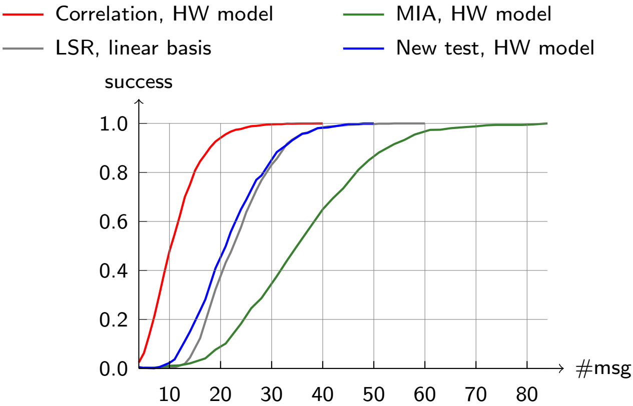
- Univariate pdf of a multidimensional distance

2D case: Smoothing and Evaluation



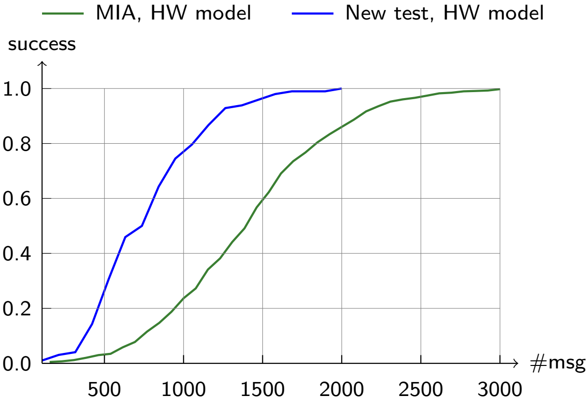
Experimental Results

Univariate Hamming Weight Leakages



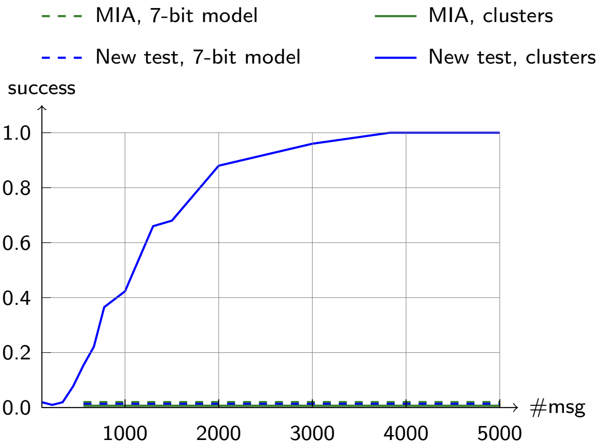
- Specific distinguishers are more efficient

Hamming Weight Leakage, Bivariate Dependency



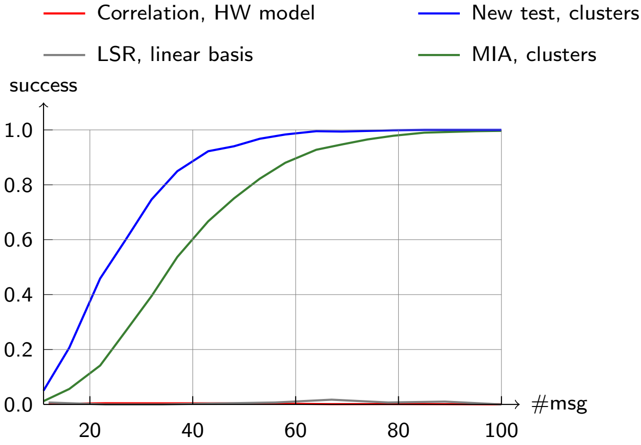
- New test exploits samples efficiently (compared to MIA)

CMOS 65 nm Measurements, Bivariate Dependency



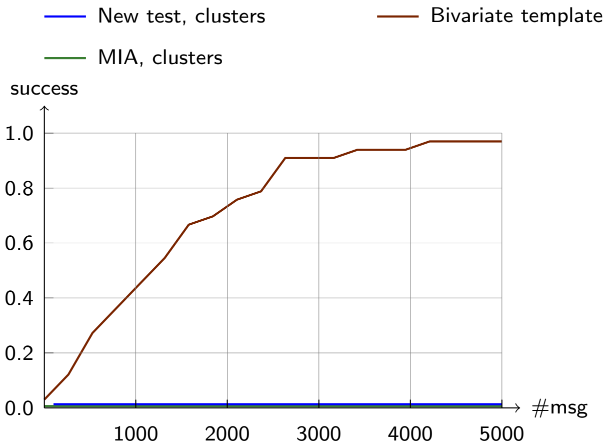
- Leakage model hard to infer from engineering intuition

Dual-Rail Simulations, Univariate Dependency



- Non-linear leakage functions can be exploited

Dual-Rail Simulations, Bivariate Dependency



- Profiling is needed to evaluate protected implementations

Conclusions

1. SCAs = efficiency vs. genericity tradeoff ('simple' dependencies are easier to exploit)
   - New generic test completely free of parameters
2. Profiling is needed for security evaluations
   - Dependency tests can be generic
   - ... but not leakage models (so far)
   - (Eurocrypt 2009 evaluation framework)

Open question: do highly non-linear leakage functions exist in practice? (or can non-linearity be used as a design criteria)