TELECOM ParisTech

Institut Mines-Telecom

STMicroelectronics

Taylor Expansion of Maximum Likelihood Attacks

Nicolas Bruneau[1,2], Sylvain Guilley[1,3], Annelie Heuser[1], Olivier Rioul[1], François-Xavier Standaert[4], Yannick Teglia[2]

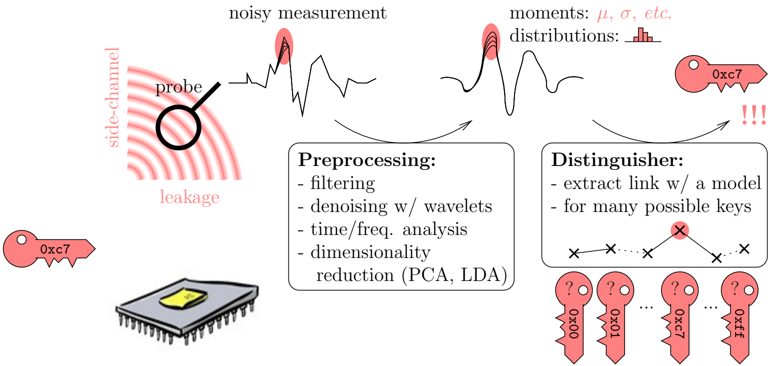
[1] Télécom-ParisTech, Crypto & ComNum Group, Paris, FRANCE
[2] STMicroelectronics, AST division, Rousset, FRANCE
[3] Secure-IC S.A.S., Rennes, FRANCE
[4] Université Catholique de Louvain, Louvain-la-Neuve, BELGIQUE

Cryptarchi 2016 — La Grande Motte, France

Outline

Outline

Side-Channel Analysis on Embedded Systems [GMN+11]



$(d-1)$th-Order Masking: Principle

Aim

The sensitive variable Z is randomly split into Ω shares:
$\Rightarrow$ need random masks M_i, $0 < i < \Omega$

Z

$Z \perp M_1 \perp \ldots \perp M_{\Omega-1}$ M_1 $\cdots$ $M_{\Omega-1}$

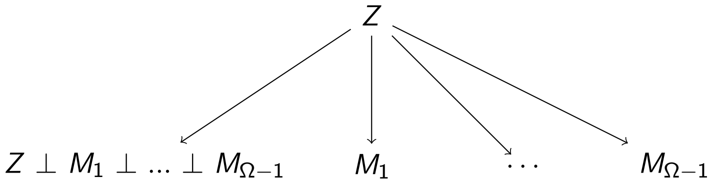
Consequence

Increases the minimum key-dependent statistical moment

$(d-1)$th-Order Masking: Principle

Aim

The sensitive variable Z is randomly split into Ω shares:
$\Rightarrow$ need random masks M_i, $0 < i < \Omega$



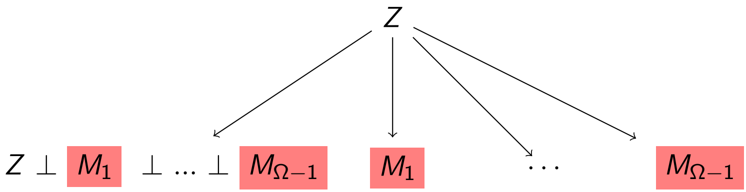
Consequence

Increases the minimum key-dependent statistical moment

$(d-1)$th-Order Masking: Principle

Aim

The sensitive variable Z is randomly split into Ω shares:
$\Rightarrow$ need random masks M_i, $0 < i < \Omega$



Consequence

Increases the minimum key-dependent statistical moment

Shuffling: Principle

Aim

Randomize the order of execution
$\Rightarrow$ need a random permutation π

Z_1 Z_2 Z_3 Z_4

Shuffling: Principle

Aim

Randomize the order of execution

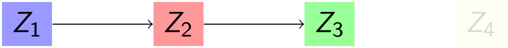
$\Rightarrow$ need a random permutation π

Shuffling: Principle

Aim

Randomize the order of execution

$\Rightarrow$ need a random permutation π

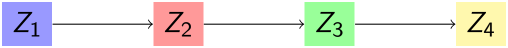


Shuffling: Principle

Aim

Randomize the order of execution

$\Rightarrow$ need a random permutation π



Shuffling: Principle

Aim

Randomize the order of execution
$\Rightarrow$ need a random permutation π

Z_1 Z_2 Z_3 Z_4

Shuffling: Principle

Aim

Randomize the order of execution

$\Rightarrow$ need a random permutation π

Shuffling: Principle

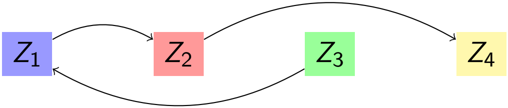
Aim

Randomize the order of execution
$\Rightarrow$ need a random permutation π

Shuffling: Principle

Aim

Randomize the order of execution
$\Rightarrow$ need a random permutation π

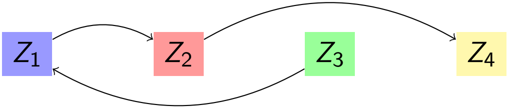


Shuffling: Principle

Aim

Randomize the order of execution

$\Rightarrow$ need a random permutation π



Consequences

Increase the noise in the attacks.

Summary of the Protection Parameters

The security level of the protections depends on these parameters:

Masking

- Ω: the number of shares (link to the numbers of masks)
- O: the order (i.e. the minimal key dependent statistical moment)

Shuffling

- Π the size of the permutation

Template Attacks

Template attacks are the most powerful in a information-theoretic sense [CRR02].

Off-line Profiling

The leakage model is learned:

- non-parametric methods (e.g. histogram, kernel methods...)
- parametric methods (e.g. mixture models)

Online Attack

Recover the key using the models by applying a maximum likelihood (ML) attack

Template Attacks

Template attacks are the most powerful in a information-theoretic sense [CRR02].

Off-line Profiling

The leakage model is learned:

- non-parametric methods (e.g. histogram, kernel methods...)
- parametric methods (e.g. mixture models)

Online Attack

Recover the key using the models by applying a maximum likelihood (ML) attack

Parametric or Non-Parametric ?

Parametric

The only random part is the noise with known distribution.

- easy to estimate;
- shuffle and mask are known;
- many templates are learned.

Non-Parametric

Shuffle and masks are part of the noise.

- can be hard to estimate $\Rightarrow$ curse of dimensionality;
- shuffle and mask are unknown.

Notations for the Online attack

The attack are applied on:

- D leakage points;
- Q traces.

For each trace the leakage model is $X = y(t, k^*, R) + N$ where:

- X is the leakage measurement;
- $y = y(t, k^*, R)$ is the deterministic part of the model that depends on the correct key k^*, some known text t, and the unknown random values (masks and permutations) R;
- N is a random noise, which follows a Gaussian distribution $p_N(z) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{z^2}{2\sigma^2}\right)$.

We let $\gamma = \frac{1}{2\sigma^2}$ be the SNR parameter.

Maximum Likelihood Attacks

Theorem (Maximum Likelihood [BGHR14a])

When the $y(t,k,R)$ are known then the optimal distinguisher (OPT) is given by

$$\begin{aligned} \mathbb{R}^{DQ} \times \mathbb{R}^{DQ} &\to \mathbb{F}_2^n \\ (\mathbf{x}, y(\mathbf{t},k,R)) &\mapsto \underset{k \in \mathbb{F}_2^n}{\operatorname{argmax}} \sum_{q=1}^{Q} \log \mathbb{E} \exp \frac{-\|x^{(q)} - y(t^{(q)},k,R)\|^2}{2\sigma^2} \end{aligned}$$

where expectation $\mathbb{E}$ is applied to the random variable $R \in \mathcal{R}$ and $\|\cdot\|$ is the Euclidean norm:

$$\left\| x^{(q)} - y(t^{(q)},k,R) \right\|^2 = \sum_{d=1}^{D} \left(x_d^{(q)} - y_d(t^{(q)},k,R) \right)^2.$$

Complexity

$$\mathcal{O}\left(Q \cdot D \cdot (2^n)^{\Omega-1} \cdot \Pi!\right)$$

- number of traces
- dimension of the attack
- number of possible share values
- number of possible permutations

Complexity

$$\mathcal{O}\left(Q \cdot D \cdot (2^n)^{\Omega-1} \cdot \Pi! \right)$$

- number of traces
- dimension of the attack
- number of possible share values
- number of possible permutations

Complexity

$$\mathcal{O}\left(Q \cdot D \cdot (2^n)^{\Omega-1} \cdot \Pi! \right)$$

- number of traces
- dimension of the attack
- number of possible share values
- number of possible permutations

Complexity

$$\mathcal{O}\left(Q \cdot D \cdot (2^n)^{\Omega-1} \cdot \Pi!\right)$$

- number of traces (Q)
- dimension of the attack (D)
- number of possible share values ($(2^n)^{\Omega-1}$)
- number of possible permutations ($\Pi!$)

Complexity

$$\mathcal{O}\left(Q \cdot D \cdot (2^n)^{\Omega-1} \cdot \Pi! \right)$$

- number of traces (Q)
- dimension of the attack (D)
- number of possible share values ($(2^n)^{\Omega-1}$)
- number of possible permutations ($\Pi!$)

Not computable for large Π !

Outline

Taylor Expansion of Optimal Attacks in Gaussian Noise

The optimal attack consists in maximizing the sum over all traces $q = 1, \ldots, Q$ of the log-likelihood:

$$\mathrm{LL} = \sum_{\ell=1}^{+\infty} \frac{\kappa_\ell}{\ell!}(-\gamma)^\ell$$

where

- κ_ℓ is the ℓth-order cumulant of $\|x - y(t,k,R)\|^2$

$$\kappa_\ell = \mu_\ell - \sum_{\ell'=1}^{\ell-1} \binom{\ell-1}{\ell'-1} \kappa_{\ell'} \mu_{\ell-\ell'} \qquad (\ell \geq 1).$$

- $\mu_\ell = \mathbb{E}_R\left(\|x - y(t,k,R)\|^{2\ell}\right)$

Rounded Optimal Attack

Rounded Optimal Attack (ROPT_L)

The rounded optimal *Lth-degree attack* consists in maximizing over the key hypothesis the sum over all traces of the Lth-order Taylor expansion LL_L in the SNR of the log-likelihood :

$$\begin{array}{rccl} \mathrm{ROPT}_L : & \mathbb{R}^{DQ} \times \mathbb{R}^{DQ} & \longrightarrow & \mathbb{F}_2^n \\ & (\mathbf{x}, y(\mathbf{t}, k, R)) & \longmapsto & \underset{k \in \mathbb{F}_2^n}{\operatorname{argmax}} \, \mathrm{LL}_L . \end{array}$$

where $\mathrm{LL}_L = \sum_{\ell=1}^{L} (-1)^{\ell} \kappa_{\ell} \frac{\gamma^{\ell}}{\ell!}$.

And we have

$$\boxed{\mathrm{LL} = \mathrm{LL}_L + o(\gamma^L)}$$

Complexity

- number of possible share values
- number of traces

$$\mathcal{O}\left(Q \cdot L \cdot \binom{D+L-1}{L} \cdot 2^{(\Omega-1)n} \cdot \binom{\Pi}{\min(\lceil \frac{\Pi}{2} \rceil, L)} \right)$$

- Factorial terms
  - dimension of the attack
  - degree of the Taylor Expansion
  - size of the permutation

Complexity

- number of possible share values
- number of traces

$$\mathcal{O}\left(Q \cdot L \cdot \binom{D+L-1}{L} \cdot 2^{(\Omega-1)n} \cdot \binom{\Pi}{\min(\lceil \frac{\Pi}{2} \rceil, L)} \right)$$

- Factorial terms
  - dimension of the attack
  - degree of the Taylor Expansion
  - size of the permutation

Complexity

- number of possible share values
- number of traces

$$\mathcal{O}\left(Q \cdot L \cdot \binom{D+L-1}{L} \cdot 2^{(\Omega-1)n} \cdot \binom{\Pi}{\min(\lceil \frac{\Pi}{2} \rceil, L)} \right)$$

- Factorial terms
  - dimension of the attack
  - degree of the Taylor Expansion
  - size of the permutation

Complexity

- number of possible share values
- number of traces

$$\mathcal{O}\left(Q \cdot L \cdot \binom{D+L-1}{L} \cdot 2^{(\Omega-1)n} \cdot \binom{\Pi}{\min(\lceil \frac{\Pi}{2} \rceil, L)} \right)$$

- Factorial terms
  - dimension of the attack
  - degree of the Taylor Expansion
  - size of the permutation

Complexity

- number of possible share values
- number of traces

$$\mathcal{O}\left(Q \cdot L \cdot \binom{D+L-1}{L} \cdot 2^{(\Omega-1)n} \cdot \binom{\Pi}{\min(\lceil \frac{\Pi}{2} \rceil, L)} \right)$$

- Factorial terms
  - dimension of the attack
  - degree of the Taylor Expansion
  - size of the permutation

Complexity

- number of possible share values
- number of traces

$$\mathcal{O}\left(Q \cdot L \cdot \binom{D+L-1}{L} \cdot 2^{(\Omega-1)n} \cdot \binom{\Pi}{\min\left(\left\lceil \frac{\Pi}{2} \right\rceil, L\right)} \right)$$

- Factorial terms
  - dimension of the attack
  - degree of the Taylor Expansion
  - size of the permutation

Complexity

- number of possible share values
- number of traces

$$\mathcal{O}\left(Q \cdot L \cdot \binom{D+L-1}{L} \cdot 2^{(\Omega-1)n} \cdot \binom{\Pi}{\min(\lceil \frac{\Pi}{2} \rceil, L)} \right)$$

- Factorial terms
  - dimension of the attack
  - degree of the Taylor Expansion
  - size of the permutation

Reduces to small constants when $L \ll D$

Outline

Implementation of Masking Schemes

In masking schemes, while the implementation of the linear parts is obvious, that of the non linear parts is more difficult.

- algebraic methods [BGK04, RP10];
- global look-up table method [PR07, SVCO$^+$10];
- table recomputation methods which precompute a masked S-box stored in a table [CJRR99, Mes00, AG01].

Recently, Coron presented at EUROCRYPT 2014 [Cor14] a table recomputation scheme secure against dth-order attacks.

Table Recomputation Algorithm

input : t, one byte of plaintext, and k, one byte of key
output: The application of AddRoundKey and SubBytes on t, i.e., $S(t \oplus k)$

$m \leftarrow_{\mathcal{R}} \mathbb{F}_2^n$, $m' \leftarrow_{\mathcal{R}} \mathbb{F}_2^n$ `// Draw of random input and output masks` ;
for $\omega \in \{0, 1, \ldots, 2^n - 1\}$ **do** `// Sbox masking`
    $z \leftarrow \omega \oplus m$ `// Masked input` ;
    $z' \leftarrow S[\omega] \oplus m'$ `// Masked output` ;
    $S'[z] \leftarrow z'$ `// Creating the masked Sbox entry` ;
end
$t \leftarrow t \oplus m$ `// Plaintext masking` ;
$t \leftarrow t \oplus k$ `// Masked AddRoundKey` ;
$t \leftarrow S'[t]$ `// Masked SubBytes` ;
$t \leftarrow t \oplus m'$ `// Demasking` ;
return t

- usual 2-variate 2nd-order attack;
- 2-stage CPA attack [PdHL09, TWO13];
- improved $(2^n + 1)$-variate 2nd-order attack on the input [BGHR14b].

Classical Countermeasure

Make the index of the loop unknown
➔ compute the loop in a random order.

Use some random permutation φ:

- random start index;
- LFSR;
- etc.

Protected Table Recomputation Algorithm

input : t, one byte of plaintext, and k, one byte of key
output: The application of AddRoundKey and SubBytes on t

$m \leftarrow_{\mathcal{R}} \mathbb{F}_2^n$, $m' \leftarrow_{\mathcal{R}} \mathbb{F}_2^n$ `// Draw of random input and output masks ;`
$\varphi \leftarrow_{\mathcal{R}} \mathbb{F}_2^n \rightarrow \mathbb{F}_2^n$ `// Draw of random permutation of` $\mathbb{F}_2^n$ `;`
for $\varphi(\omega) \in \{\varphi(0), \varphi(1), \ldots, \varphi(2^n - 1)\}$ **do** `// S-box masking`
    $z \leftarrow \varphi(\omega) \oplus m$ `// Masked input ;`
    $z' \leftarrow S[\varphi(\omega)] \oplus m'$ `// Masked output ;`
    $S'[z] = z'$ `// Creating the masked S-box entry ;`
end
$t \leftarrow t \oplus m$ `// Plaintext masking ;`
$t \leftarrow t \oplus k$ `// Masked AddRoundKey ;`
$t \leftarrow S'[t]$ `// Masked SubBytes ;`
$t \leftarrow t \oplus m'$ `// Demasking ;`
return t

Leakages

input : t, one byte of plaintext, and k, one byte of key
output: The application of AddRoundKey and SubBytes on t

$m \leftarrow_{\mathcal{R}} \mathbb{F}_2^n$, $m' \leftarrow_{\mathcal{R}} \mathbb{F}_2^n$ `// Draw of random input and output masks ;`
$\varphi \leftarrow_{\mathcal{R}} \mathbb{F}_2^n \rightarrow \mathbb{F}_2^n$ `// Draw of random permutation of` $\mathbb{F}_2^n$;
for $\varphi(\omega) \in \{\varphi(0), \varphi(1), \ldots, \varphi(2^n - 1)\}$ **do** `// S-box masking`
    $z \leftarrow \varphi(\omega) \oplus m$ `// Masked input ;`
    $z' \leftarrow S[\varphi(\omega)] \oplus m'$ `// Masked output ;`
    $S'[z] = z'$ `// Creating the masked S-box entry ;`
end
$t \leftarrow t \oplus m$ `// Plaintext masking ;`
$t \leftarrow t \oplus k$ `// Masked AddRoundKey ;`
$t \leftarrow S'[t]$ `// Masked SubBytes ;`
$t \leftarrow t \oplus m'$ `// Demasking ;`
return t

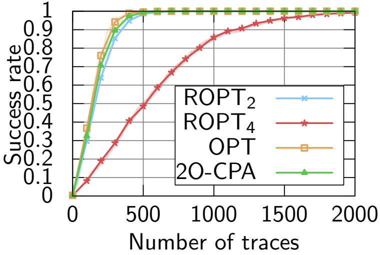
Leakages

input : t, one byte of plaintext, and k, one byte of key
output: The application of AddRoundKey and SubBytes on t

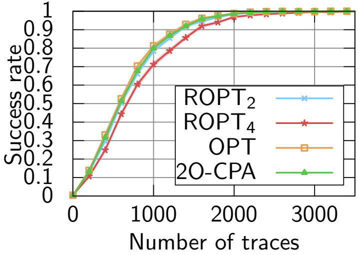
$m \leftarrow_{\mathcal{R}} \mathbb{F}_2^n$, $m' \leftarrow_{\mathcal{R}} \mathbb{F}_2^n$ `// Draw of random input and output masks ;`
$\varphi \leftarrow_{\mathcal{R}} \mathbb{F}_2^n \rightarrow \mathbb{F}_2^n$ `// Draw of random permutation of` $\mathbb{F}_2^n$;
for $\varphi(\omega) \in \{\varphi(0), \varphi(1), \ldots, \varphi(2^n - 1)\}$ **do** `// S-box masking`
    $z \leftarrow \varphi(\omega) \oplus m$ `// Masked input ;`
    $z' \leftarrow S[\varphi(\omega)] \oplus m'$ `// Masked output ;`
    $S'[z] = z'$ `// Creating the masked S-box entry ;`
end
$t \leftarrow t \oplus m$ `// Plaintext masking ;`
$t \leftarrow t \oplus k$ `// Masked AddRoundKey ;`
$t \leftarrow S'[t]$ `// Masked SubBytes ;`
$t \leftarrow t \oplus m'$ `// Demasking ;`
return t

- second-order Correlation Power Analysis 2O-CPA;
- OPTimal distinguisher OPT_2;
  - Rounded OPTimal Distinguisher ROPT_2, ROPT_4

Bi-Variate Attacks



(a) $\sigma = 1$



(b) $\sigma = 2$

Leakages, with Table Recomputation

input : t, one byte of plaintext, and k, one byte of key
output: The application of AddRoundKey and SubBytes on t

$m \leftarrow_{\mathcal{R}} \mathbb{F}_2^n$, $m' \leftarrow_{\mathcal{R}} \mathbb{F}_2^n$ `// Draw of random input and output masks ;`
$\varphi \leftarrow_{\mathcal{R}} \mathbb{F}_2^n \rightarrow \mathbb{F}_2^n$ `// Draw of random permutation of` $\mathbb{F}_2^n$;
for $\varphi(\omega) \in \{\varphi(0), \varphi(1), \ldots, \varphi(2^n - 1)\}$ **do** `// S-box masking`
  $z \leftarrow \varphi(\omega) \oplus m$ `// Masked input ;`
  $z' \leftarrow S[\varphi(\omega)] \oplus m'$ `// Masked output ;`
  $S'[z] = z'$ `// Creating the masked S-box entry ;`
end
$t \leftarrow t \oplus m$ `// Plaintext masking ;`
$t \leftarrow t \oplus k$ `// Masked AddRoundKey ;`
$t \leftarrow S'[t]$ `// Masked SubBytes ;`
$t \leftarrow t \oplus m'$ `// Demasking ;`
return t

Leakages, with Table Recomputation

input : t, one byte of plaintext, and k, one byte of key
output: The application of AddRoundKey and SubBytes on t

$m \leftarrow_{\mathcal{R}} \mathbb{F}_2^n$, $m' \leftarrow_{\mathcal{R}} \mathbb{F}_2^n$ `// Draw of random input and output masks ;`
$\varphi \leftarrow_{\mathcal{R}} \mathbb{F}_2^n \rightarrow \mathbb{F}_2^n$ `// Draw of random permutation of` $\mathbb{F}_2^n$;
for $\varphi(\omega) \in \{\varphi(0), \varphi(1), \ldots, \varphi(2^n - 1)\}$ **do** `// S-box masking`
    $z \leftarrow \varphi(\omega) \oplus m$ `// Masked input ;`
    $z' \leftarrow S[\varphi(\omega)] \oplus m'$ `// Masked output ;`
    $S'[z] = z'$ `// Creating the masked S-box entry ;`
end
$t \leftarrow t \oplus m$ `// Plaintext masking ;`
$t \leftarrow t \oplus k$ `// Masked AddRoundKey ;`
$t \leftarrow S'[t]$ `// Masked SubBytes ;`
$t \leftarrow t \oplus m'$ `// Demasking ;`
return t

- optimal distinguisher NOT computable due to the term 2^n!

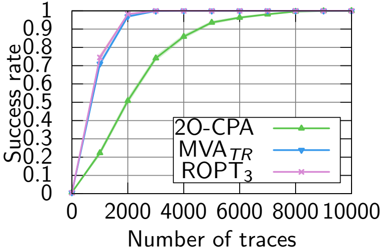
Leakages, with Table Recomputation

input : t, one byte of plaintext, and k, one byte of key
output: The application of AddRoundKey and SubBytes on t

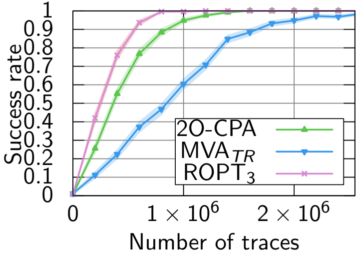
$m \leftarrow_{\mathcal{R}} \mathbb{F}_2^n$, $m' \leftarrow_{\mathcal{R}} \mathbb{F}_2^n$ // Draw of random input and output masks ;
$\varphi \leftarrow_{\mathcal{R}} \mathbb{F}_2^n \to \mathbb{F}_2^n$ // Draw of random permutation of $\mathbb{F}_2^n$;
for $\varphi(\omega) \in \{\varphi(0), \varphi(1), \ldots, \varphi(2^n - 1)\}$ **do** // S-box masking
  $z \leftarrow \varphi(\omega) \oplus m$ // Masked input ;
  $z' \leftarrow S[\varphi(\omega)] \oplus m'$ // Masked output ;
  $S'[z] = z'$ // Creating the masked S-box entry ;
end
$t \leftarrow t \oplus m$ // Plaintext masking ;
$t \leftarrow t \oplus k$ // Masked AddRoundKey ;
$t \leftarrow S'[t]$ // Masked SubBytes ;
$t \leftarrow t \oplus m'$ // Demasking ;
return t

- third order attack MVA_{TR} [BGNT15]
- Rounded Optimal Distinguisher ROPT_3

$(2^{n+1}+2)$-Variate Attacks on Shuffled Table Recomputation

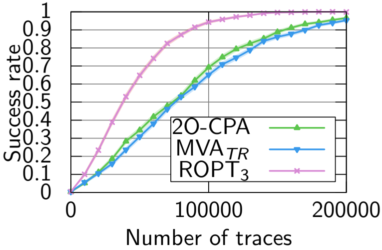


(a) $\sigma = 3$

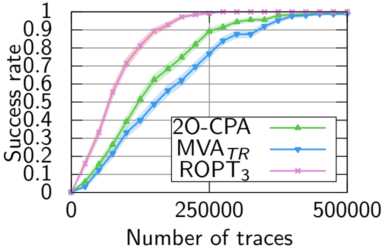


(b) $\sigma = 12$

$(2^{n+1}+2)$-Variate Attacks on Shuffled Table Recomputation

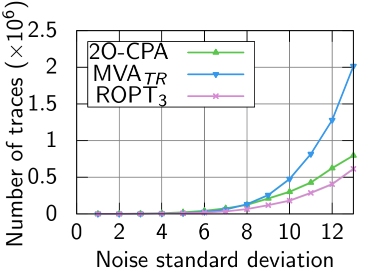


(a) $\sigma = 8$

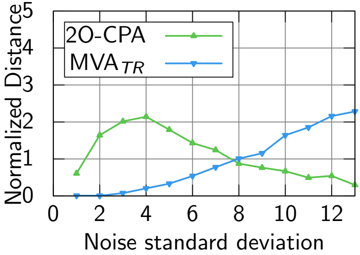


(b) $\sigma = 9$

$(2^{n+1}+2)$-Variate Attacks on Shuffled Table Recomputation



(a) Number of traces to reach 80% of success



(b) Distance with ROPT$_3$ at 80% of success

Complexity of the Case Study

Attack	Time (seconds)	Computational Complexity
2O-CPA	39	$\mathcal{O}(Q)$
ROPT_2	295	$\mathcal{O}(Q)$
OPT_{2O}	9473	$\mathcal{O}(Q \cdot 2^n)$
MVA_{TR}	130	$\mathcal{O}(Q \cdot 2^n)$
ROPT_3	2495	$\mathcal{O}\left(Q \cdot 2^{2n}\right)$
OPT	Not computable	$\mathcal{O}\left(Q \cdot 2^n \cdot 2^n! \cdot \left(2^{n+1} + 2\right)\right)$

Conclusion

Results

We have presented a practical, truncated version of the theoretical, optimal distinguisher:

- becomes effective;
- remains efficient.

Perspective

How to quantify the accuracy of the approximation?

Thank you for your attention.

[AG01] Mehdi-Laurent Akkar and Christophe Giraud.
An Implementation of DES and AES Secure against Some Attacks.
In LNCS, editor, *Proceedings of CHES'01*, volume 2162 of *LNCS*, pages 309–318. Springer, May 2001.
Paris, France.

[BGHR14a] Nicolas Bruneau, Sylvain Guilley, Annelie Heuser, and Olivier Rioul.
Masks Will Fall Off – Higher-Order Optimal Distinguishers.
In Palash Sarkar and Tetsu Iwata, editors, *Advances in Cryptology – ASIACRYPT 2014 - 20th International Conference on the Theory and Application of Cryptology and Information Security, Kaoshiung, Taiwan, R.O.C., December 7-11, 2014, Proceedings, Part II*, volume 8874 of *Lecture Notes in Computer Science*, pages 344–365. Springer, 2014.

[BGHR14b] Nicolas Bruneau, Sylvain Guilley, Annelie Heuser, and Olivier Rioul.
Masks Will Fall Off: Higher-Order Optimal Distinguishers.
In *ASIACRYPT*, volume 8874 of *LNCS*, pages 344–365. Springer, December 2014.
P. Sarkar and T. Iwata (Eds.): ASIACRYPT 2014, PART II.

[BGK04] Johannes Blömer, Jorge Guajardo, and Volker Krummel.
Provably Secure Masking of AES.
In Helena Handschuh and M. Anwar Hasan, editors, *Selected Areas in Cryptography*, volume 3357 of *Lecture Notes in Computer Science*, pages 69–83. Springer, 2004.

[BGNT15] Nicolas Bruneau, Sylvain Guilley, Zakaria Najm, and Yannick Teglia.
Multi-variate high-order attacks of shuffled tables recomputation.
In Tim Güneysu and Helena Handschuh, editors, *Cryptographic Hardware and Embedded Systems - CHES 2015 - 17th International Workshop, Saint-Malo, France, September 13-16, 2015, Proceedings*, volume 9293 of *Lecture Notes in Computer Science*, pages 475–494. Springer, 2015.

[CJRR99] Suresh Chari, Charanjit S. Jutla, Josyula R. Rao, and Pankaj Rohatgi.
Towards Sound Approaches to Counteract Power-Analysis Attacks.
In *CRYPTO*, volume 1666 of *LNCS*. Springer, August 15-19 1999. Santa Barbara, CA, USA. ISBN: 3-540-66347-9.

[Cor14] Jean-Sébastien Coron.
Higher Order Masking of Look-Up Tables.
In Phong Q. Nguyen and Elisabeth Oswald, editors, *EUROCRYPT*, volume 8441 of *Lecture Notes in Computer Science*, pages 441–458. Springer, 2014.

[CRR02] Suresh Chari, Josyula R. Rao, and Pankaj Rohatgi.
Template Attacks.
In *CHES*, volume 2523 of *LNCS*, pages 13–28. Springer, August 2002.
San Francisco Bay (Redwood City), USA.

[GMN+11] Sylvain Guilley, Olivier Meynard, Maxime Nassar, Guillaume Duc, Philippe Hoogvorst, Houssem Maghrebi, Aziz Elaabid, Shivam Bhasin, Youssef Souissi, Nicolas Debande, Laurent Sauvage, and Jean-Luc Danger.
Vade Mecum on Side-Channels Attacks and Countermeasures for the Designer and the Evaluator.
In *DTIS (Design & Technologies of Integrated Systems)*, IEEE. IEEE, March 6-8 2011.
Athens, Greece. DOI: 10.1109/DTIS.2011.5941419 ; Online version:
`http://hal.archives-ouvertes.fr/hal-00579020/en/`.

[Mes00] Thomas S. Messerges.
Securing the AES Finalists Against Power Analysis Attacks.
In *Fast Software Encryption'00*, pages 150–164. Springer-Verlag, April 2000.
New York.

[PdHL09] Jing Pan, Jerry I. den Hartog, and Jiqiang Lu.
You cannot hide behind the mask: Power analysis on a provably secure S-box implementation.
In Heung Youl Youm and Moti Yung, editors, *Information Security Applications, 10th International Workshop, WISA 2009, Busan, Korea, August 25-27, 2009, Revised Selected Papers*, volume 5932 of *Lecture Notes in Computer Science*, pages 178–192. Springer, 2009.

[PR07] Emmanuel Prouff and Matthieu Rivain.
A Generic Method for Secure SBox Implementation.
In Sehun Kim, Moti Yung, and Hyung-Woo Lee, editors, *WISA*, volume 4867 of *Lecture Notes in Computer Science*, pages 227–244. Springer, 2007.

[RP10] Matthieu Rivain and Emmanuel Prouff.
Provably Secure Higher-Order Masking of AES.
In Stefan Mangard and François-Xavier Standaert, editors, *CHES*, volume 6225 of *LNCS*, pages 413–427. Springer, 2010.

[SVCO+10] François-Xavier Standaert, Nicolas Veyrat-Charvillon, Elisabeth Oswald, Benedikt Gierlichs, Marcel Medwed, Markus Kasper, and Stefan Mangard.
The World is Not Enough: Another Look on Second-Order DPA.
In *ASIACRYPT*, volume 6477 of *LNCS*, pages 112–129. Springer, December 5-9 2010.
Singapore.
http://www.dice.ucl.ac.be/~fstandae/PUBLIS/88.pdf.

[TWO13] Michael Tunstall, Carolyn Whitnall, and Elisabeth Oswald.
Masking Tables - An Underestimated Security Risk.
In Shiho Moriai, editor, *FSE*, volume 8424 of *Lecture Notes in Computer Science*, pages 425–444. Springer, 2013.