Side Channel Cryptanalysis of a Higher Order Masking Scheme

J.-S. Coron[1] E. Prouff[2] M. Rivain[1,2]

[1]University of Luxembourg

[2]Oberthur Card Systems

CHES 2007

Contents

Differential Power Analysis (DPA)

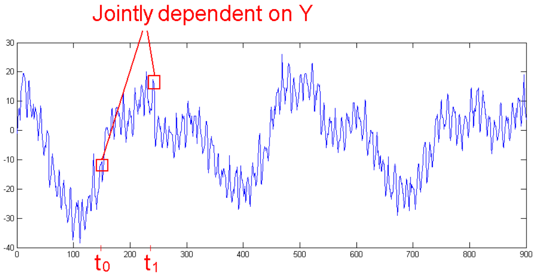
- The **physical leakage** of the execution of any algorithm depends on the **intermediate variables**
- DPA exploits leakage on **sensitive variables** that depends on the secret key
- Common countermeasure: **masking**
  - A random value is added to every sensitive variable
  - $\Rightarrow$ Instantaneous leakage independent of sensitive variables

Higher Order DPA (HO-DPA)

Against First Order Masking

- Y: sensitive variable, M: mask
  - $Y \oplus M$ processed at t_0
  - M processed at t_1

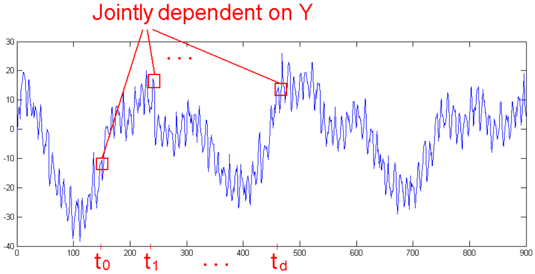
- First order DPA attack not feasible
- Second order DPA attack feasible



Higher Order DPA (HO-DPA)

Against d-th Order Masking

- Y: sensitive variable, M_i's: masks
  - $Y \oplus M_1 \oplus \cdots \oplus M_d$ processed at t_0
  - M_i's processed at t_i
- d-th order DPA attack not feasible
- $(d+1)$-th order DPA attack feasible



Higher Order DPA (HO-DPA)

- The complexity of an HO-DPA is **exponential** with its order (Chari *et al.* in CRYPTO'99)
- The order d is a good security parameter
- A generic masking scheme must
  - involve d random masks per sensitive variable
  - thwart d-th order DPA

Higher Order Masking Schemes

Formalizing the security:

- **sensitive variable**: depends on both the plaintext and the secret key

Higher Order Masking Schemes

Formalizing the security:

- **sensitive variable**: depends on both the plaintext and the secret key
- d**-th order flaw**: a d-tuple of intermediate variables statistically dependent on a sensitive variable

Higher Order Masking Schemes

Formalizing the security:

- **sensitive variable**: depends on both the plaintext and the secret key
- **d-th order flaw**: a d-tuple of intermediate variables statistically dependent on a sensitive variable
- **security against d-th order DPA**: no d-th order flaw

Higher Order Masking Schemes

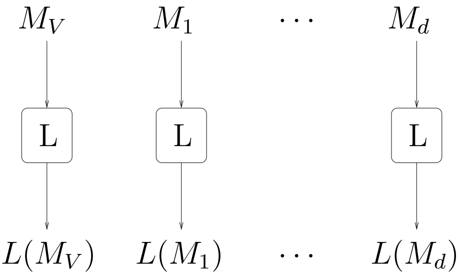
- Each sensitive variable Y is masked with d masks M_i's
- **completeness**: the masked variable M_V and the masks M_i's must always satisfy:

$$M_V \oplus M_1 \oplus \cdots \oplus M_d = Y$$

- **security**: M_V and all the M_i's must be processed separately

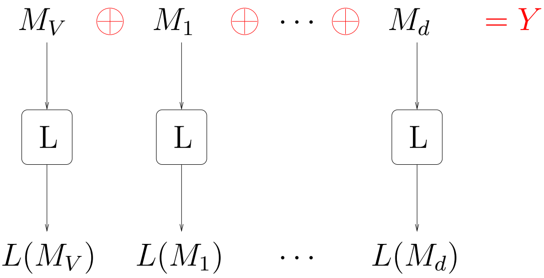
Higher Order Masking Schemes

- In network of linear layers and non-linear SBoxes
  - Propagation through a **linear layer**



Higher Order Masking Schemes

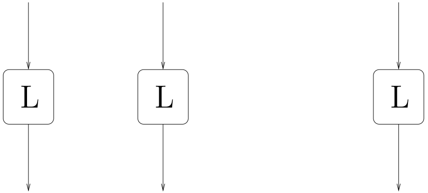
- In network of linear layers and non-linear SBoxes
  - Propagation through a **linear layer**



Higher Order Masking Schemes

- In network of linear layers and non-linear SBoxes
  - Propagation through a **linear layer**

$$M_V \oplus M_1 \oplus \cdots \oplus M_d = Y$$



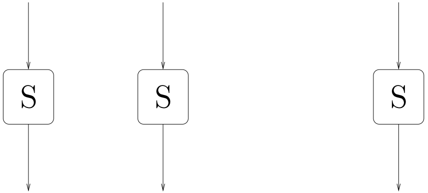
$$L(M_V) \oplus L(M_1) \oplus \cdots \oplus L(M_d) = L(Y)$$

Oberthur Card Systems
The Key to an Open World

Higher Order Masking Schemes

- In network of linear layers and non-linear SBoxes
  - Propagation through a **non-linear SBox**

$$M_V \oplus M_1 \oplus \cdots \oplus M_d = Y$$



$$S(M_V) \oplus S(M_1) \oplus \cdots \oplus S(M_d) \neq S(Y)$$

Higher Order Masking Schemes

- In network of linear layers and non-linear SBoxes
  - Propagation through a **non-linear SBox**

$$M_V \oplus M_1 \oplus \cdots \oplus M_d = Y$$

$$\downarrow \quad \downarrow \quad \downarrow$$

$$?? \quad ?? \quad ??$$

$$\downarrow \quad \downarrow \quad \downarrow$$

$$N_V \oplus N_1 \oplus \cdots \oplus N_d = S(Y)$$

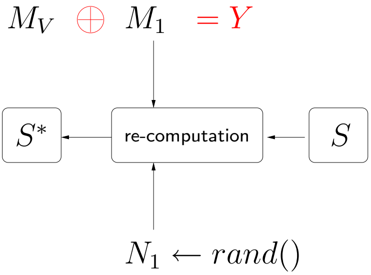
Problem

How to **securely** *compute* $(N_V, N_i's)$ *from* $(M_V, M_i's)$ *?*

Higher Order Masking Schemes

- Problem widely investigated for 1-st order masking
  - Efficient and widely used method: the **table re-computation**
- For d-th order masking: one single proposal in the Literature
  - [SP06] - K. Schramm and C. Paar, “Higher Order Masking of the AES” in CT-RSA 2006.
  - Principle: adapt the table re-computation method to d-th order masking
- Our paper: [SP06] is **broken by 3-rd order DPA** for any value of the masking order d

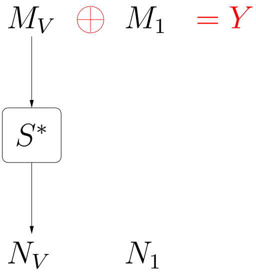
Table re-computation method
For 1-st order masking



- For all x: $S^*(x) \leftarrow S(x \oplus M_1) \oplus N_1$

Table re-computation method

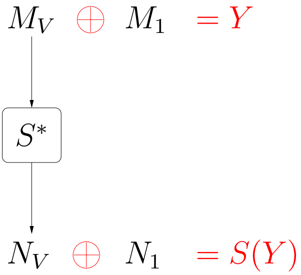
For 1-st order masking



- For all x: $S^*(x) \leftarrow S(x \oplus M_1) \oplus N_1$
- $N_V \leftarrow S^*(M_V)$

Table re-computation method

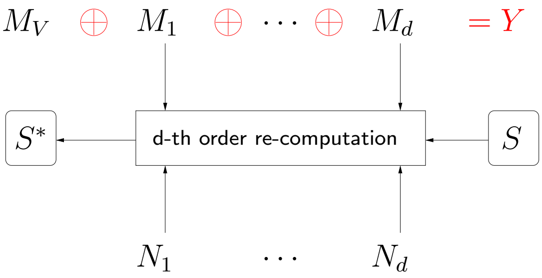
For 1-st order masking



- For all x: $S^*(x) \leftarrow S(x \oplus M_1) \oplus N_1$
- $N_V \leftarrow S^*(M_V) = S(M_V \oplus M_1) \oplus N_1 = S(Y) \oplus N_1$

Table re-computation method

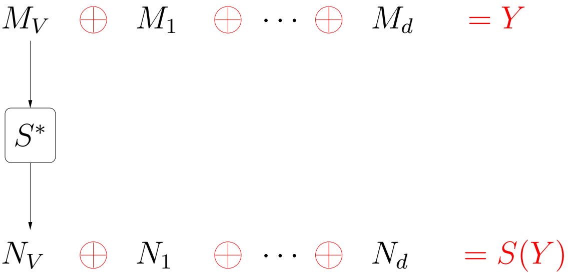
For d-th order masking [SP06]



- For every x: $S^*(x) = S\left(x \oplus \bigoplus_{i=1}^{d} M_i\right) \oplus \bigoplus_{i=1}^{d} N_i$

Table re-computation method

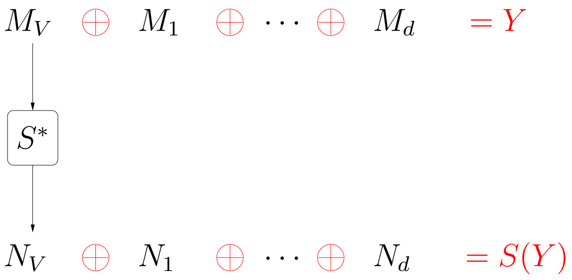
For d-th order masking [SP06]



- For every x: $S^*(x) = S\left(x \oplus \bigoplus_{i=1}^{d} M_i\right) \oplus \bigoplus_{i=1}^{d} N_i$

Table re-computation method

For d-th order masking [SP06]



- For every x: $S^*(x) = S\left(x \oplus \bigoplus_{i=1}^{d} M_i\right) \oplus \bigoplus_{i=1}^{d} N_i$

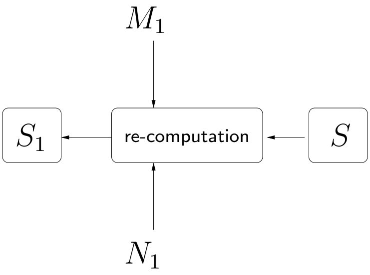
Problem

How to **securely** *compute* S^* *from* $(S, M_i's, N_i's)$.

Generic Scheme

Process d successive table re-computations:

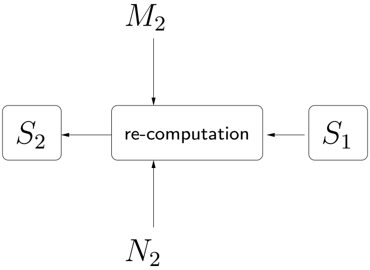
- $S_1(x) = S(x \oplus M_1) \oplus N_1$



Generic Scheme

Process d successive table re-computations:

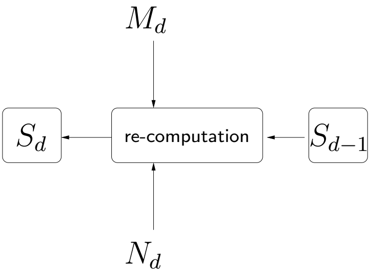
- $S_1(x) = S(x \oplus M_1) \oplus N_1$
- $S_2(x) = S(x \oplus M_1 \oplus M_2) \oplus N_1 \oplus N_2$



Generic Scheme

Process d successive table re-computations:

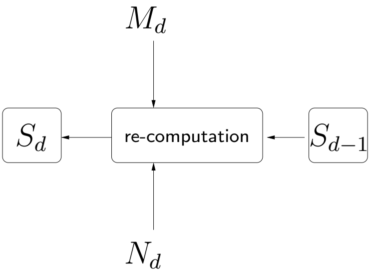
- $S_1(x) = S(x \oplus M_1) \oplus N_1$
- $S_2(x) = S(x \oplus M_1 \oplus M_2) \oplus N_1 \oplus N_2$
- ...
- $S_d(x) = S(x \oplus M_1 \oplus M_2 \oplus \cdots \oplus M_d) \oplus N_1 \oplus N_2 \oplus \cdots \oplus N_d$



Generic Scheme

Process d successive table re-computations:

- $S_1(x) = S(x \oplus M_1) \oplus N_1$
- $S_2(x) = S(x \oplus M_1 \oplus M_2) \oplus N_1 \oplus N_2$
- ...
- $S_d(x) = S^*(x)$



3-rd order flaw

- Let $M = \bigoplus_{i=1}^{d} M_i$ and $N = \bigoplus_{i=1}^{d} N_i$
- The masked variable M_V satisfies:
  1) $M_V = Y \oplus M$
- During the re-computation of table S_d:
  2) $S_d(0) = S(0 \oplus M) \oplus N$
  3) $S_d(1) = S(1 \oplus M) \oplus N$

3-rd order flaw

- Let $M = \bigoplus_{i=1}^{d} M_i$ and $N = \bigoplus_{i=1}^{d} N_i$
- The masked variable M_V satisfies:
  1) $M_V = Y \oplus M$
- During the re-computation of table S_d:
  2) $S_d(0) = S(0 \oplus M) \oplus N$
  3) $S_d(1) = S(1 \oplus M) \oplus N$
- The distribution of $(M_V, S_d(0), S_d(1))$ depends on Y
  - 3-rd order flaw!
  - thus a 3-rd order DPA theoretically feasible!

3-rd order flaw

Pinpointing the dependency

- We have:
  1) $M_V = Y \oplus M$
  2) $S_d(0) = S(0 \oplus M) \oplus N$
  3) $S_d(1) = S(1 \oplus M) \oplus N$
- $S_d(0) \oplus S_d(1) = S(M) \oplus S(M \oplus 1)$
  - depends on M
- Hence, $S_d(0) \oplus S_d(1)$ and M_V jointly depend on Y
- Hence, the 3-tuple $(M_V, S_d(0), S_d(1))$ depends on Y

3-rd order flaw

- The attack also works for any 3-tuple ($a \neq b$):

$$\tau_{a,b} = (M_V, S_d(a), S_d(b))$$

iff $x \mapsto S(x) \oplus S(x \oplus a \oplus b)$ **is not constant**

- $\tau_{a,b}$ is independent of Y for every (a, b) iff S is affine
- Hence, S is non-affine $\Rightarrow \exists(a, b)$: $\tau_{a,b}$ depends of Y

3-rd order flaw

- The attack also works for any 3-tuple ($a \neq b$):

$$\tau_{a,b} = (M_V, S_d(a), S_d(b))$$

iff $x \mapsto S(x) \oplus S(x \oplus a \oplus b)$ **is not constant**

- $\tau_{a,b}$ is independent of Y for every (a, b) iff S is affine
- Hence, S is non-affine $\Rightarrow \exists (a, b) : \tau_{a,b}$ depends of Y
- For every non-affine SBox, the generic scheme [SP06] admits a 3-rd order flaw!

3-rd order flaw

- The attack also works for any 3-tuple ($a \neq b$):

$$\tau_{a,b} = (M_V, S_d(a), S_d(b))$$

iff $x \mapsto S(x) \oplus S(x \oplus a \oplus b)$ **is not constant**

- $\tau_{a,b}$ is independent of Y for every (a, b) iff S is affine
- Hence, S is non-affine $\Rightarrow \exists (a, b) : \tau_{a,b}$ depends of Y
- The generic scheme [SP06] is broken by 3-rd order DPA for any masking order d!

Conclusion

The approach of processing d table re-computations is not sound to thwart d-th order DPA.

Improved Scheme

- Generic scheme very costly
  - d table re-computations per S-Box access
- Proposed improvement [SP06]:
  - d table re-computations for the first SBox access
  - **1 single table re-computation** for each next SBox access
- How ?
  - each new masked SBox is derived from the previous one

Improved Scheme

- Let M_V and M'_V be two consecutive masked SBox inputs
  - $M_V = Y \oplus M_1 \oplus \cdots \oplus M_d$
  - $M'_V = Y' \oplus M'_1 \oplus \cdots \oplus M'_d$

Improved Scheme

- Let M_V and M'_V be two consecutive masked SBox inputs
  - $M_V = Y \oplus M_1 \oplus \cdots \oplus M_d$
  - $M'_V = Y' \oplus M'_1 \oplus \cdots \oplus M'_d$
- Let S^* and S^*_{new} be the masked SBoxes:
  - $S^*(x) = S\left(x \oplus \bigoplus_{i=1}^{d} M_i\right) \oplus \bigoplus_{i=1}^{d} N_i$
  - $S^*_{new}(x) = S\left(x \oplus \bigoplus_{i=1}^{d} M'_i\right) \oplus \bigoplus_{i=1}^{d} N'_i$

Improved Scheme

- From:
  - $S^*(x) = S\left(x \oplus \bigoplus_{i=1}^{d} M_i\right) \oplus \bigoplus_{i=1}^{d} N_i$
  - $S^*_{new}(x) = S\left(x \oplus \bigoplus_{i=1}^{d} M'_i\right) \oplus \bigoplus_{i=1}^{d} N'_i$
- we have:

$$S^*_{new}(x) = S^*\left(x \oplus \bigoplus_{i=1}^{d} M_i \oplus \bigoplus_{i=1}^{d} M'_i\right) \oplus \bigoplus_{i=1}^{d} N_i \oplus \bigoplus_{i=1}^{d} N'_i$$

- $S^*_{new} \leftarrow$ re-computation $\left(S^*, \bigoplus_{i=1}^{d} M_i \oplus \bigoplus_{i=1}^{d} M'_i, \bigoplus_{i=1}^{d} N_i \oplus \bigoplus_{i=1}^{d} N'_i\right)$

Improved Scheme

- From:
  - $S^*(x) = S\left(x \oplus \bigoplus_{i=1}^{d} M_i\right) \oplus \bigoplus_{i=1}^{d} N_i$
  - $S^*_{new}(x) = S\left(x \oplus \bigoplus_{i=1}^{d} M'_i\right) \oplus \bigoplus_{i=1}^{d} N'_i$
- we have:

$$S^*_{new}(x) = S^*\left(x \oplus \bigoplus_{i=1}^{d} M_i \oplus \bigoplus_{i=1}^{d} M'_i\right) \oplus \bigoplus_{i=1}^{d} N_i \oplus \bigoplus_{i=1}^{d} N'_i$$

- $S^*_{new} \leftarrow$ re-computation$(S^*, \mathbf{ICM}, \mathbf{OCM})$
  - $\mathbf{ICM} = \bigoplus_{i=1}^{d} M_i \oplus \bigoplus_{i=1}^{d} M'_i$
  - $\mathbf{OCM} = \bigoplus_{i=1}^{d} N_i \oplus \bigoplus_{i=1}^{d} N'_i$

3-rd order flaws

- The processing of **ICM** (*resp.* **OCM**) introduces a 3-rd order flaw
- ICM 3-rd order flaw:
  1) $M_V = Y \oplus M_1 \oplus \cdots \oplus M_d$
  2) $M'_V = Y' \oplus M'_1 \oplus \cdots \oplus M'_d$
  3) $\mathbf{ICM} = M_1 \oplus \cdots \oplus M_d \oplus M'_1 \oplus \cdots \oplus M'_d$
- $M_V \oplus M'_V \oplus \mathbf{ICM} = Y \oplus Y'$

3-rd order flaws

- The processing of **ICM** (*resp.* **OCM**) introduces a 3-rd order flaw

- OCM 3-rd order flaw:
  1) $N_V = S(Y) \oplus N_1 \oplus \cdots \oplus N_d$
  2) $N'_V = S(Y') \oplus N'_1 \oplus \cdots \oplus N'_d$
  3) $\mathbf{OCM} = N_1 \oplus \cdots \oplus N_d \oplus N'_1 \oplus \cdots \oplus N'_d$

- $N_V \oplus N'_V \oplus \mathbf{OCM} = S(Y) \oplus S(Y')$

3-rd order flaws

- The processing of **ICM** (*resp.* **OCM**) introduces a 3-rd order flaw

- OCM 3-rd order flaw:
  1) $N_V = S(Y) \oplus N_1 \oplus \cdots \oplus N_d$
  2) $N'_V = S(Y') \oplus N'_1 \oplus \cdots \oplus N'_d$
  3) $\mathbf{OCM} = N_1 \oplus \cdots \oplus N_d \oplus N'_1 \oplus \cdots \oplus N'_d$

- $N_V \oplus N'_V \oplus \mathbf{OCM} = S(Y) \oplus S(Y')$

- The improved scheme [SP06] is broken by 3-rd order DPA for any masking order d!

Conclusion

The improvement of the scheme – that makes it efficient in a low ressource environment – is not suitable.

Experimental Results

- Attack simulations
  - Known plaintext attacks on AES
  - Hamming weight model with (low) Gaussian noise
- Two attack strategies
  - Combining 3O-DPA:
    - correlation attack on a combination of the 3 leakages
    - classical HO-DPA attack
  - Profiling 3O-DPA:
    - Maximum likelihood test
    - strong adversarial model (requires the knowledge of the exact distribution of the 3 leakages)
- See the paper for further details on the simulations

Experimental Results

Implementation	Attack	Measurements
Generic scheme	combining 3O-DPA	6.10^6
Generic scheme	profiling 3O-DPA	2.10^3
Improved scheme	combining 3O-DPA	10^5
Improved scheme	profiling 3O-DPA	10^3

Table: Number of measurements required for a success rate of 50%.

- Our attacks are practical in a classical leakage model
- The profiling 3O-DPA is more efficient than the combining 3O-DPA
- The attacks are more efficient on the improved scheme

Conclusion

- The scheme [SP06] is vulnerable to 3-rd order DPA and is not suitable for d-th order DPA resistance
  - First attack: process d table re-computations not suitable
  - Second attack: proposed improvement not suitable
- Our attacks are practical in a weakly noisy environnement
- The design of a Higher Order Masking Scheme is still an open issue