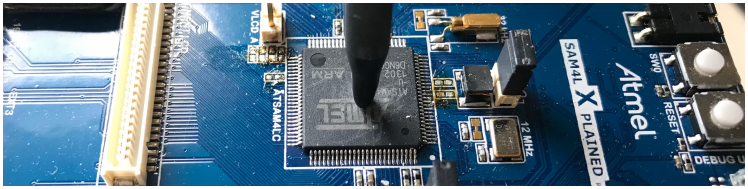


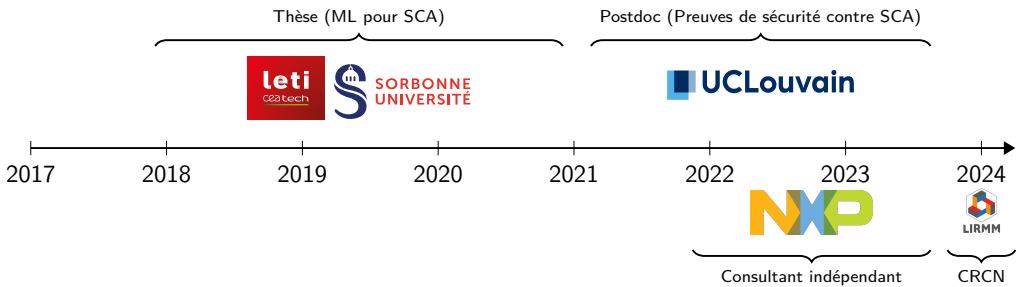
Au Bal des Implémentations Masquées

Comment protéger une implémentation cryptographique contre les attaques par canaux auxiliaires

Loïc Masure

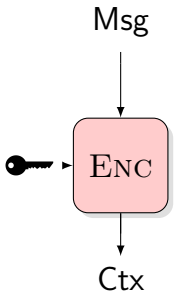
CIEL, Montpellier, 15 Janvier 2024

Qui suis-je ?


Thèse (ML pour SCA)
Postdoc (Preuves de sécurité contre SCA)
leti
CEA tech
SORBONNE UNIVERSITÉ
UCLouvain
2017
2018
2019
2020
2021
2022
2023
2024
NXP
LIRMM
Consultant indépendant
CRCN

Content

Context : Side-Channel Analysis (SCA)

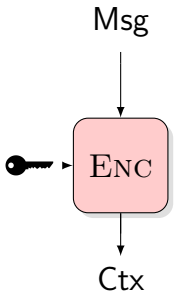


: N bits

Black-box cryptanalysis: 2^N

Context : Side-Channel Analysis (SCA)

"Cryptographic algorithms don't run on paper,

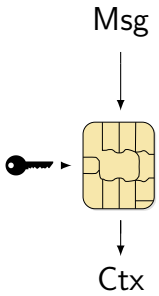


: N bits

Black-box cryptanalysis: 2^N

Context : Side-Channel Analysis (SCA)

"Cryptographic algorithms don't run on paper, they run on physical devices"

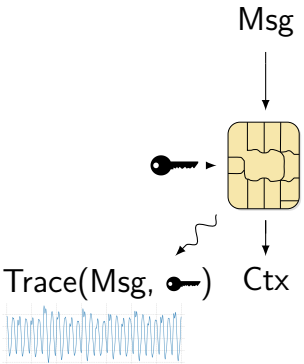


: N bits

Black-box cryptanalysis: 2^N

Context : Side-Channel Analysis (SCA)

"Cryptographic algorithms don't run on paper, they run on physical devices"

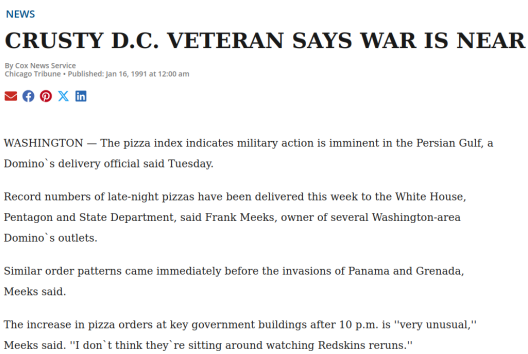


: N bits

Black-box cryptanalysis: 2^N

Side Channel = Unintended Communication Channel

Example: the Washington Pizza Index[1]



NEWS

CRUSTY D.C. VETERAN SAYS WAR IS NEAR

By Cox News Service
Chicago Tribune • Published: Jan 16, 1991 at 12:00 am

WASHINGTON — The pizza index indicates military action is imminent in the Persian Gulf, a Domino`s delivery official said Tuesday.

Record numbers of late-night pizzas have been delivered this week to the White House, Pentagon and State Department, said Frank Meeks, owner of several Washington-area Domino`s outlets.

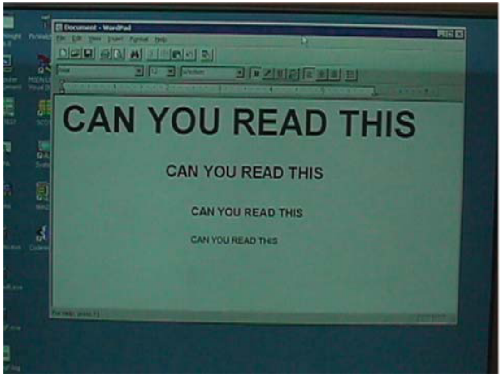
Similar order patterns came immediately before the invasions of Panama and Grenada, Meeks said.

The increase in pizza orders at key government buildings after 10 p.m. is "very unusual," Meeks said. "I don`t think they`re sitting around watching Redskins reruns."

Figure: Chicago Tribune, **Jan. 16 1991**, the day before *Desert Storm* operation began.

[1] Reality questioned: `http://home.xnet.com/~warinner/pizzacites.html`

What is a Side Channel? A First Example



(a) A good old monitor

(b) Reconstruction from EM field

Figure: An example from Koç, *Cryptographic Engineering*.

An Exemplary SCA on Crypto

RSA: Modular exponentiation over large ($\approx$ 2000-bit wide) integers

An Exemplary SCA on Crypto

RSA: Modular exponentiation over large ($\approx$ 2000-bit wide) integers

Square-and-Multiply:

$$M^k = M^{\sum_i k_i \cdot 2^i} = \prod_i (M^{k_i})^{2^i}$$

An Exemplary SCA on Crypto

RSA: Modular exponentiation over large ($\approx$ 2000-bit wide) integers

Square-and-Multiply:

$$M^k = M^{\sum_i k_i \cdot 2^i} = \prod_i (M^{k_i})^{2^i}$$

Step 0: M^{k_N} then square
Step 1: $\times M^{k_{N-1}}$ then square
...
Step i: $\times M^{k_{N-i}}$ then square

An Exemplary SCA on Crypto

RSA: Modular exponentiation over large ($\approx$ 2000-bit wide) integers

Square-and-Multiply:

$$M^k = M^{\sum_i k_i \cdot 2^i} = \prod_i (M^{k_i})^{2^i}$$

Step 0: M^{k_N} then square
Step 1: if $k_{N-1} = 1$, $\times M$ then square
...
Step i: if $k_{N-i} = 1$, $\times M$ then square

An Exemplary SCA on Crypto

RSA: Modular exponentiation over large ($\approx$ 2000-bit wide) integers

Square-and-Multiply:

$$M^k = M^{\sum_i k_i \cdot 2^i} = \prod_i (M^{k_i})^{2^i}$$

Step 0: M^{k_N} then square
Step 1: if $k_{N-1} = 1$, $\times M$ then square
...
Step i: if $k_{N-i} = 1$, $\times M$ then square

Op.	Guess
$\times M$	
square	
square	
square	
$\times M$	
square	
$\times M$	
square	

An Exemplary SCA on Crypto

RSA: Modular exponentiation over large ($\approx$ 2000-bit wide) integers

Square-and-Multiply:

$$M^k = M^{\sum_i k_i \cdot 2^i} = \prod_i (M^{k_i})^{2^i}$$

Step 0: M^{k_N} then square
Step 1: if $k_{N-1} = 1$, $\times M$ then square
...
Step i: if $k_{N-i} = 1$, $\times M$ then square

Op.	Guess
$\times M$	$k_N = 1$
square	
square	
square	
$\times M$	
square	
$\times M$	
square	

An Exemplary SCA on Crypto

RSA: Modular exponentiation over large ($\approx$ 2000-bit wide) integers

Square-and-Multiply:

$$M^k = M^{\sum_i k_i \cdot 2^i} = \prod_i (M^{k_i})^{2^i}$$

Step 0: M^{k_N} then square
Step 1: if $k_{N-1} = 1$, $\times M$ then square
...
Step i: if $k_{N-i} = 1$, $\times M$ then square

Op.	Guess
$\times M$	$k_N = 1$
square	
square	$k_{N-1} = 0$
square	
$\times M$	
square	
$\times M$	
square	

An Exemplary SCA on Crypto

RSA: Modular exponentiation over large ($\approx$ 2000-bit wide) integers

Square-and-Multiply:

$$M^k = M^{\sum_i k_i \cdot 2^i} = \prod_i (M^{k_i})^{2^i}$$

Step 0: M^{k_N} then square
Step 1: if $k_{N-1} = 1$, $\times M$ then square
...
Step i: if $k_{N-i} = 1$, $\times M$ then square

Op.	Guess
$\times M$	$k_N = 1$
square	
square	$k_{N-1} = 0$
square	$k_{N-2} = 0$
$\times M$	
square	
$\times M$	
square	

An Exemplary SCA on Crypto

RSA: Modular exponentiation over large ($\approx$ 2000-bit wide) integers

Square-and-Multiply:

$$M^k = M^{\sum_i k_i \cdot 2^i} = \prod_i (M^{k_i})^{2^i}$$

Step 0: M^{k_N} then square
Step 1: if $k_{N-1} = 1$, $\times M$ then square
...
Step i: if $k_{N-i} = 1$, $\times M$ then square

Op.	Guess
$\times M$	$k_N = 1$
square	
square	$k_{N-1} = 0$
square	$k_{N-2} = 0$
$\times M$	$k_{N-3} = 1$
square	
$\times M$	
square	

An Exemplary SCA on Crypto

RSA: Modular exponentiation over large ($\approx$ 2000-bit wide) integers

Square-and-Multiply:

$$M^k = M^{\sum_i k_i \cdot 2^i} = \prod_i (M^{k_i})^{2^i}$$

Step 0: M^{k_N} then square
Step 1: if $k_{N-1} = 1$, $\times M$ then square
...
Step i: if $k_{N-i} = 1$, $\times M$ then square

Op.	Guess
$\times M$	$k_N = 1$
square	
square	$k_{N-1} = 0$
square	$k_{N-2} = 0$
$\times M$	$k_{N-3} = 1$
square	
$\times M$	$k_{N-4} = 1$
square	

Can you guess the key from the Oscilloscope?

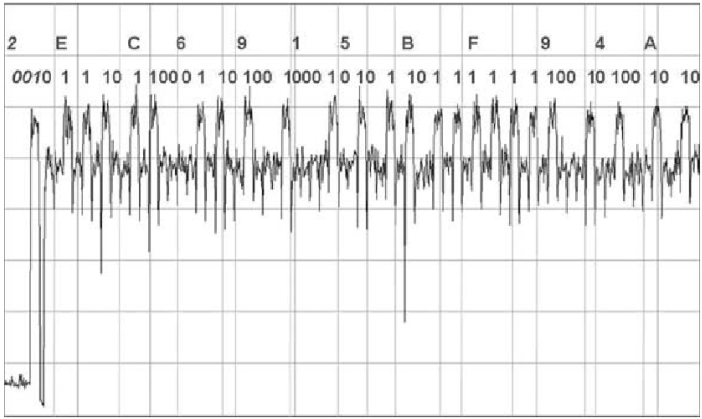
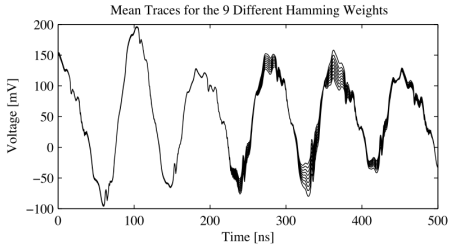


Figure: Power consumption. Illustration from Koç, *Cryptographic Engineering*

Power Analysis on Symmetric Key

Power consumption: each bit x_i of a data chunk X stored in a register[2]

$x_i = 0 \implies$ register voltage $= 0$
$x_i = 1 \implies$ register voltage $\neq 0$
Overall consumption of X is $\propto \sum_i x_i$

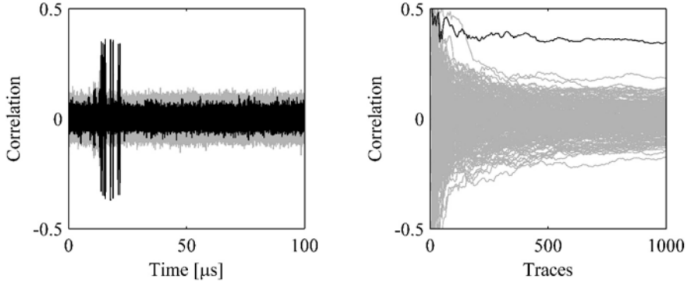


[2]Mangard, Oswald, and Popp, *Power analysis attacks - revealing the secrets of smart cards.*

Power Analysis on Symmetric Key

Power consumption: each bit x_i of a data chunk X stored in a register[2]

Key guessed by a statistical test leveraging the correlation between the Hamming weight of data and the power consumption



[2]Mangard, Oswald, and Popp, *Power analysis attacks - revealing the secrets of smart cards*.

Content

Masking: what is that ?

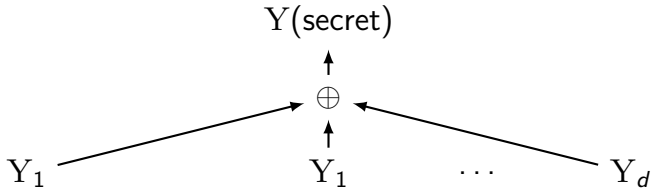
Masking, aka *MPC on silicon*: linear secret sharing over a finite field $(\mathbb{F}, \star, \cdot)$

$$\mathrm{Y}(\text{secret})$$

Introduced by Chari *et al.*, Goubin & Patarin (Crypto, Ches 99)

Masking: what is that ?

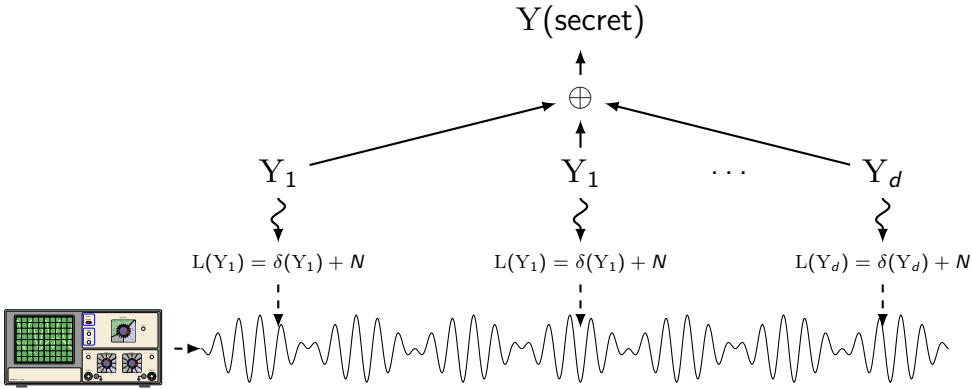
Masking, aka *MPC on silicon*: linear secret sharing over a finite field $(\mathbb{F}, \star, \cdot)$



Introduced by Chari *et al.*, Goubin & Patarin (Crypto, Ches 99)

Masking: what is that ?

Masking, aka *MPC on silicon*: linear secret sharing over a finite field $(\mathbb{F}, \star, \cdot)$



Introduced by Chari *et al.*, Goubin & Patarin (Crypto, Ches 99)

How to Calculate over Masked Data? Outline

Over finite field $\mathbb{F}$, write each operation as a polynomial (Lagrange)
One polynomial is made of:

How to Calculate over Masked Data? Outline

Over finite field $\mathbb{F}$, write each operation as a polynomial (Lagrange)
One polynomial is made of:

- $\mathbb{F}$-affine functions (*e.g.*, $\oplus$): $f(\sum_i \mathrm{Y}_i) = \sum_i f(\mathrm{Y}_i)$;

How to Calculate over Masked Data? Outline

Over finite field $\mathbb{F}$, write each operation as a polynomial (Lagrange)
One polynomial is made of:

- $\mathbb{F}$-affine functions (*e.g.*, $\oplus$): $f(\sum_i \mathrm{Y}_i) = \sum_i f(\mathrm{Y}_i)$;

$\rightarrow$ Trivial transformation

How to Calculate over Masked Data? Outline

Over finite field $\mathbb{F}$, write each operation as a polynomial (Lagrange)
One polynomial is made of:

- $\mathbb{F}$-affine functions (*e.g.*, $\oplus$): $f(\sum_i \mathrm{Y}_i) = \sum_i f(\mathrm{Y}_i)$;

$\rightarrow$ Trivial transformation

- $\mathbb{F}$-bilinear (*e.g.* $\otimes$) mappings: $f(\sum_i \mathrm{A}_i, \sum_j \mathrm{B}j) = \sum_i \sum_j f(\mathrm{A}_i, \mathrm{B}_j)$.

How to Calculate over Masked Data? Outline

Over finite field $\mathbb{F}$, write each operation as a polynomial (Lagrange)
One polynomial is made of:

- $\mathbb{F}$-affine functions (*e.g.*, $\oplus$): $f(\sum_i \mathrm{Y}_i) = \sum_i f(\mathrm{Y}_i)$;

  $\rightarrow$ Trivial transformation

- $\mathbb{F}$-bilinear (*e.g.* $\otimes$) mappings: $f(\sum_i \mathrm{A}_i, \sum_j \mathrm{B}j) = \sum_i \sum_j f(\mathrm{A}_i, \mathrm{B}_j)$.

  $\rightarrow$ Spans d^2 shares

  $\rightarrow$ Needs to *securely* compress into d shares.

  $\rightarrow$ Introduce *fresh* randomness somewhere.

How to Calculate over Masked Data? Outline

Over finite field $\mathbb{F}$, write each operation as a polynomial (Lagrange)
One polynomial is made of:

- $\mathbb{F}$-affine functions (*e.g.*, $\oplus$): $f(\sum_i \mathrm{Y}_i) = \sum_i f(\mathrm{Y}_i)$;

  $\rightarrow$ Trivial transformation

- $\mathbb{F}$-bilinear (*e.g.* $\otimes$) mappings: $f(\sum_i \mathrm{A}_i, \sum_j \mathrm{B}j) = \sum_i \sum_j f(\mathrm{A}_i, \mathrm{B}_j)$.

  $\rightarrow$ Spans d^2 shares

  $\rightarrow$ Needs to *securely* compress into d shares.

  $\rightarrow$ Introduce *fresh* randomness somewhere.

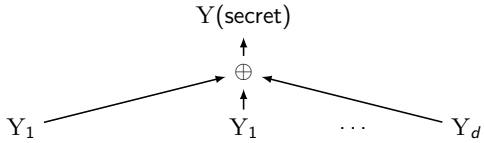
In this talk we only focus on the leakage of *one* d-sharing only

Content

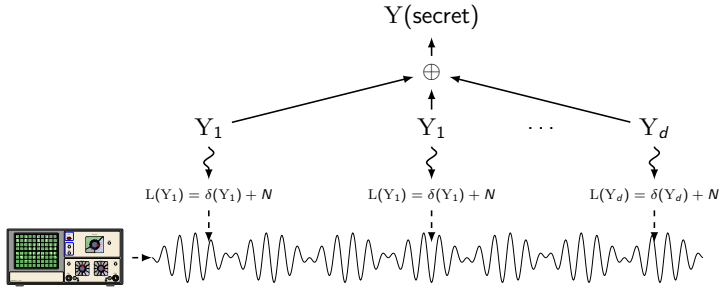
Why these Observations?

$Y(\text{secret})$

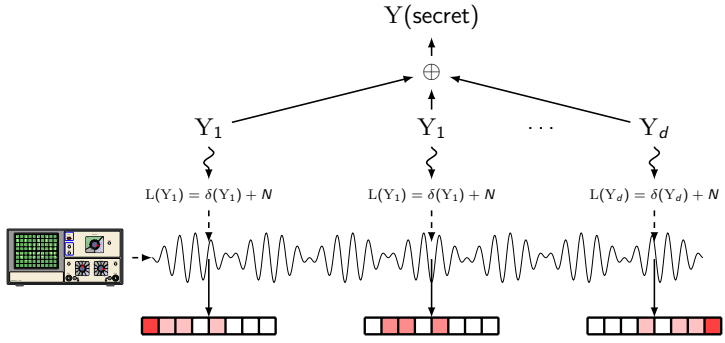
Why these Observations?



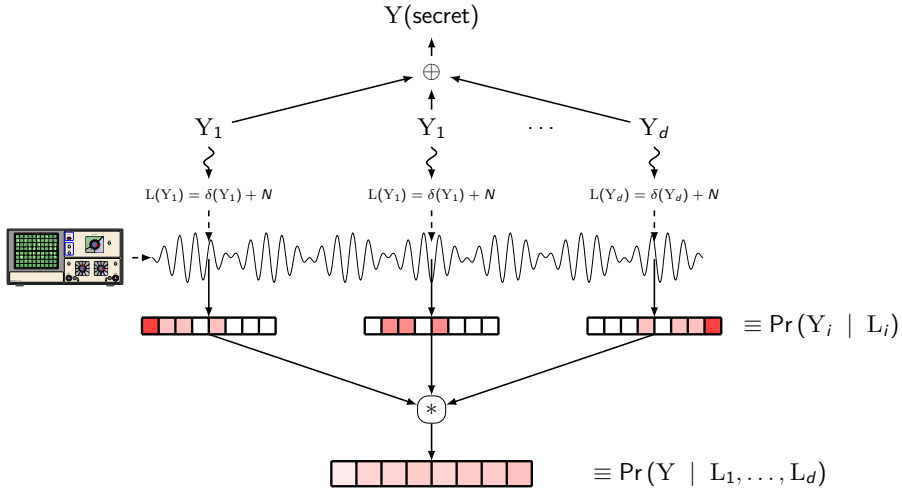
Why these Observations?



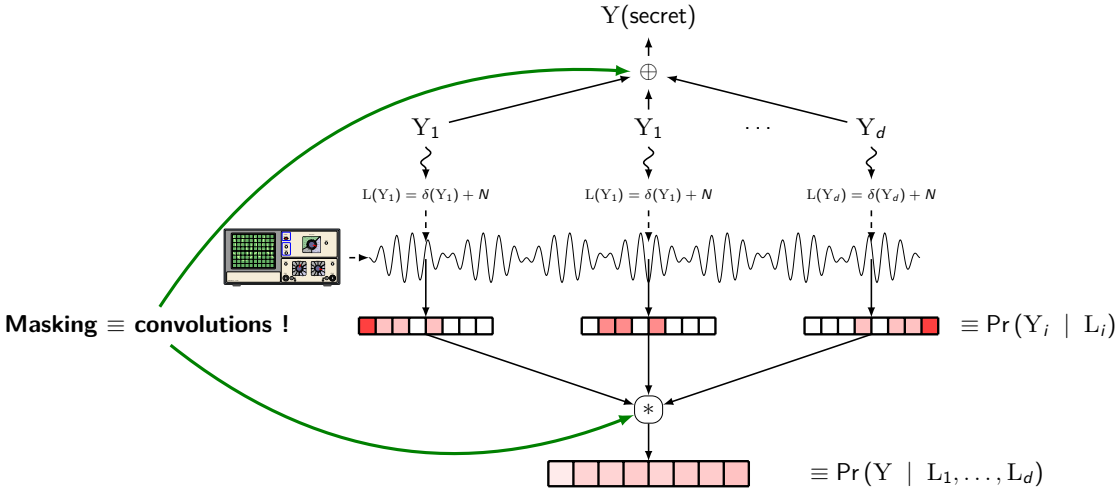
Why these Observations?



Why these Observations?

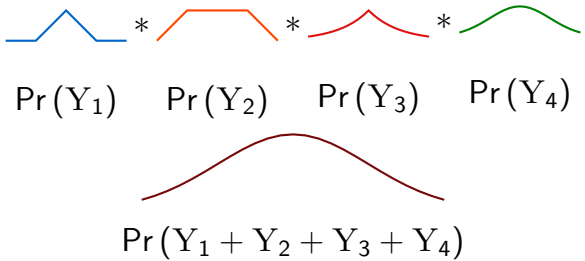


Why these Observations?



The Secret Power of Convolutions

Central Limit Theorem: Assume real-valued random variables Y_i

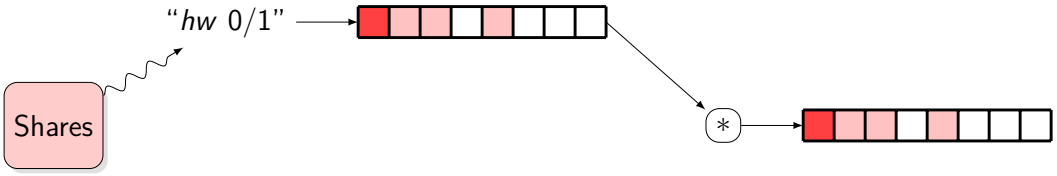


Then the sum is (approximately) distributed like a Gaussian[3]
Interesting property of Gaussian: maximizes the entropy (*i.e.*, uncertainty)[4]

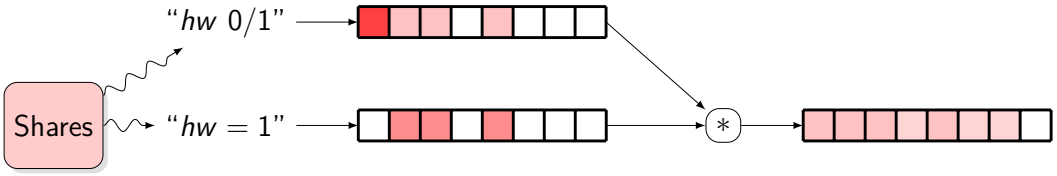
[3] With mild assumptions, but we'll get back to that ...

[4] Out of all Probability Density Functions (p.d.f.s) of same mean and variance

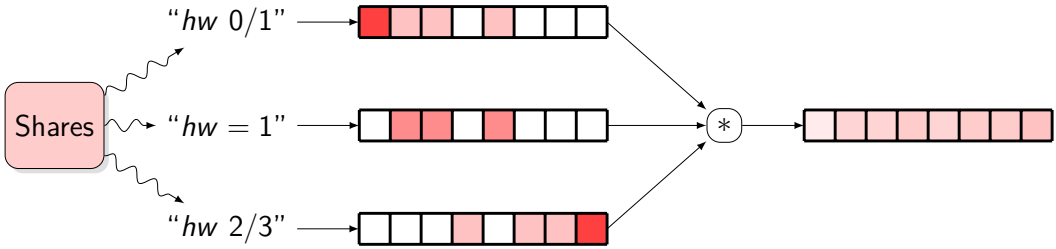
CLT also Works in Finite Groups/Fields !



CLT also Works in Finite Groups/Fields !



CLT also Works in Finite Groups/Fields !



	$Y_i \in \mathbb{R}$	$Y_i \in \mathbb{F}$
Limit Entropy	Gaussian Maximized	Uniform Maximized

Fast Fourier Transform also apply over finite fields !

Quantitative version of CLT

Theorem (Mrs. Gerber's Lemma[5]*)*

Given $\mathrm{Y} = \mathrm{Y}_1 \oplus \ldots \oplus \mathrm{Y}_d$*, and each* Y_i *with (indep.) side information* $\mathrm{L}_1, \ldots, \mathrm{L}_d$*, then for* $\eta^{-1} = 2\log(2)$*:*

[5] Béguinot et al., "Removing the Field Size Loss from Duc et al.'s Conjectured Bound for Masked Encodings".

Quantitative version of CLT

Theorem (Mrs. Gerber's Lemma[5])

Given $\mathrm{Y} = \mathrm{Y}_1 \oplus \ldots \oplus \mathrm{Y}_d$*, and each* Y_i *with (indep.) side information* $\mathrm{L}_1, \ldots, \mathrm{L}_d$*, then for* $\eta^{-1} = 2\log(2)$*:*

$$\mathsf{MI}(\mathrm{Y}; \mathbf{L}) \leq \prod_{i=1}^{d} \frac{\mathsf{MI}(\mathrm{Y}_i; \mathrm{L}_i)}{\eta} + \mathcal{O}\left(\prod_{i=1}^{d} \mathsf{MI}(\mathrm{Y}_i; \mathrm{L}_i) \right) \quad \textit{in } \mathbb{F}_{2^n}$$

[5] Béguinot et al., "Removing the Field Size Loss from Duc et al.'s Conjectured Bound for Masked Encodings".

Quantitative version of CLT

Theorem (Mrs. Gerber's Lemma[5])

Given $\mathrm{Y} = \mathrm{Y}_1 \oplus \ldots \oplus \mathrm{Y}_d$*, and each* Y_i *with (indep.) side information* $\mathrm{L}_1, \ldots, \mathrm{L}_d$*, then for* $\eta^{-1} = 2\log(2)$*:*

$$\mathsf{MI}(\mathrm{Y}; \mathbf{L}) \leq \prod_{i=1}^{d} \frac{\mathsf{MI}(\mathrm{Y}_i; \mathrm{L}_i)}{\eta} + \mathcal{O}\left(\prod_{i=1}^{d} \mathsf{MI}(\mathrm{Y}_i; \mathrm{L}_i) \right) \quad \textit{in } \mathbb{F}_{2^n}$$

$\rightarrow$ Security $\propto \frac{1}{\mathsf{MI}(\mathrm{Y};\mathbf{L})} \implies$ increases **exponentially fast** with d ✓

[5] Béguinot et al., "Removing the Field Size Loss from Duc et al.'s Conjectured Bound for Masked Encodings".

Quantitative version of CLT

Theorem (Mrs. Gerber's Lemma[5])

Given $\mathrm{Y} = \mathrm{Y}_1 \oplus \ldots \oplus \mathrm{Y}_d$*, and each* Y_i *with (indep.) side information* $\mathrm{L}_1, \ldots, \mathrm{L}_d$*, then for* $\eta^{-1} = 2\log(2)$*:*

$$\mathsf{MI}(\mathrm{Y}; \mathbf{L}) \leq \prod_{i=1}^{d} \frac{\mathsf{MI}(\mathrm{Y}_i; \mathrm{L}_i)}{\eta} + \mathcal{O}\left(\prod_{i=1}^{d} \mathsf{MI}(\mathrm{Y}_i; \mathrm{L}_i) \right) \quad \textit{in } \mathbb{F}_{2^n}$$

$\rightarrow$ Security $\propto \frac{1}{\mathsf{MI}(\mathrm{Y};\mathbf{L})} \implies$ increases **exponentially fast** with d ✓

$\rightarrow$ Independent of the adversary ✓

[5] Béguinot et al., "Removing the Field Size Loss from Duc et al.'s Conjectured Bound for Masked Encodings".

Convolution = Noise Amplification

Simulation, for $\mathbb{F}_{2^n}$: $L(Y_i) = hw(Y_i) + \mathcal{N}(0; \sigma^2)$, hw = Hamming weight

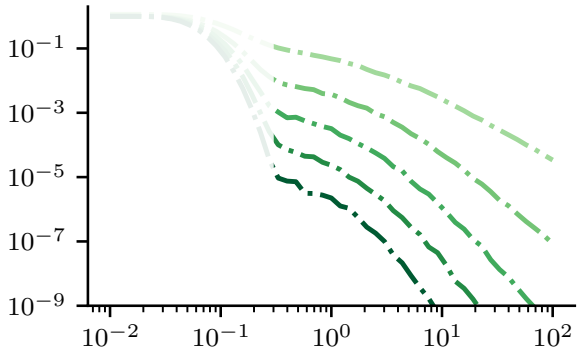


Figure: $\mathrm{MI}(\mathrm{Y}; \mathbf{L})$ vs. σ^2, $2 \leq d \leq 6$

Masking in a Low-Noise Setting

Does masking always work in a low-noise setting ?

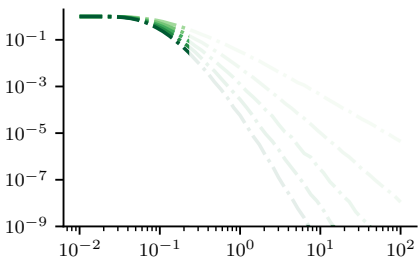


Figure: $MI(Y; Trace)$ vs. σ^2, $2 \leq d \leq 6$

Observation:
Secret always leaks > 1 bit, regardless of d

Explanation:
$\mathsf{lsb}(\mathrm{Y}_1 \oplus \ldots \oplus \mathrm{Y}_d) = \mathsf{lsb}(\mathrm{Y}_1) \oplus \ldots \oplus \mathsf{lsb}(\mathrm{Y}_d)$

Masking in a Low-Noise Setting

Does masking always work in a low-noise setting ?

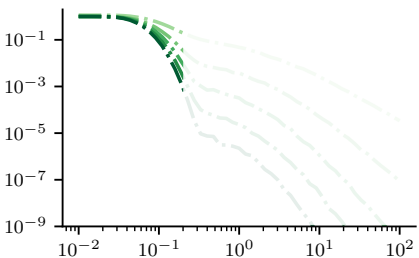


Figure: $MI(Y; Trace)$ vs. σ^2, $2 \leq d \leq 6$

Observation:
Secret always leaks > 1 bit, regardless of d
Explanation:
$\mathrm{hw}(\mathrm{Y}_1 \oplus \ldots \oplus \mathrm{Y}_d) = \sum_i \mathrm{hw}(\mathrm{Y}_i) - 2 \cdot \left(\ldots \right)$
Parity of $\mathrm{hw}(\mathrm{Y})$: **cosets of** $\mathbb{F}_{2^n}$
Corollary: parallelism is no cure either

Masking in a Low-Noise Setting

Does masking always work in a low-noise setting ?

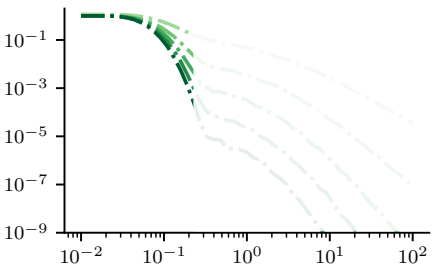
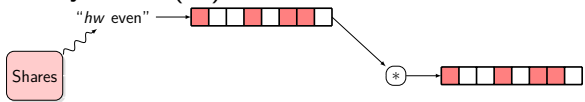


Figure: $MI(Y; Trace)$ vs. σ^2, $2 \leq d \leq 6$

Explanation:

Parity of $\mathrm{hw}(Y)$: **cosets of** $\mathbb{F}_{2^n}$



Masking in a Low-Noise Setting

Does masking always work in a low-noise setting ?

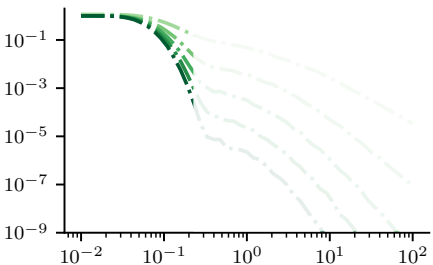
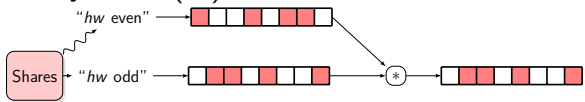


Figure: $MI(Y; Trace)$ vs. σ^2, $2 \leq d \leq 6$

Explanation:

Parity of $\mathrm{hw}(Y)$: **cosets of** $\mathbb{F}_{2^n}$



Masking in a Low-Noise Setting

Does masking always work in a low-noise setting ?

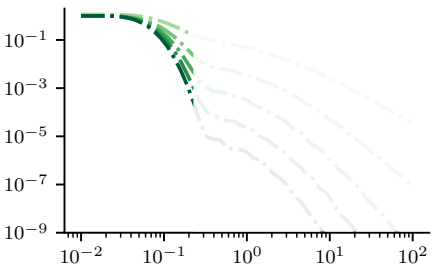
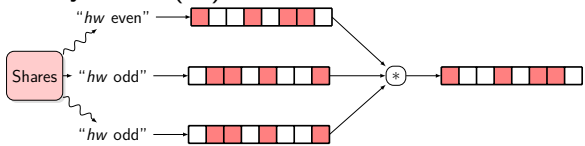


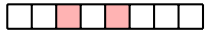
Figure: $MI(Y; Trace)$ vs. σ^2, $2 \leq d \leq 6$

Explanation:

Parity of $\mathrm{hw}(Y)$: **cosets of** $\mathbb{F}_{2^n}$

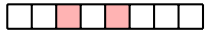


Conditions for Sound Masking

What conditions the distributions  of each share must fit?

[6] Stromberg, "Probabilities on a Compact Group".

Conditions for Sound Masking

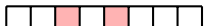
What conditions the distributions  of each share must fit?

"Central Limit Theorem" (qualitative)[6]

Conv. to uniform $\iff$ support *not* contained in any non-trivial coset of $\mathbb{F}$

[6] Stromberg, "Probabilities on a Compact Group".

Conditions for Sound Masking

What conditions the distributions  of each share must fit?

"Central Limit Theorem" (qualitative)[6]

Conv. to uniform $\iff$ support *not* contained in any non-trivial coset of $\mathbb{F}$

In $\mathbb{R}$: mild assumption

$\rightarrow$ Only $\mathbb{Z}$ and $\mathbb{Q}$ (and their respective subgroups)

$\rightarrow$ Negligible measure over $\mathbb{R}$

In finite $\mathbb{F}$: no longer mild in finite fields ...

[6] Stromberg, "Probabilities on a Compact Group".

Two Solutions

Two Solutions

Solution 1: Make sure to leak < 1 bit per share:

- Support of PMF always larger than any coset
- Work with any $\mathbb{F}$ (usually chosen to fit the cipher) ✓
- **Leakage-dependent: not always verified ✗**

Two Solutions

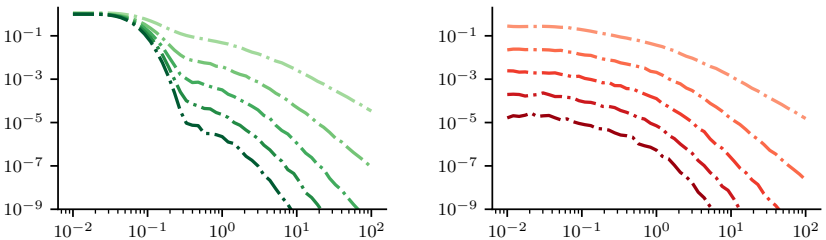
Solution 2: Choose $\mathbb{F}$ without any non-trivial subgroup, *i.e.*, $\mathbb{F}_p$, p prime:

- No assumption on the leakage ✓
- Major change of paradigm:

  Fix $\mathbb{F}$ masking-friendly first,

  Then build crypto upon it ✓

Comparing Binary and Prime Fields: a Simulation



(a) $\mathbb{F}_{2^7}$ (b) $\mathbb{F}_{2^7-1}$

Figure: Comparing binary and prime fields.

Content

How to leverage?

Q: How can we make use of masking in $\mathbb{F}_p$ to effectively and efficiently protect crypto implementations?

A: Ideally, we need algorithms that work in implementation-friendly prime fields, such as **small-Mersenne-prime fields** ($\mathbb{F}_{2^n-1}$), and use only simple field arithmetic ($+,\ -,\ \cdot$)

Complex in Software? Not really!

Field Addition in $\mathbb{F}_{2^n-1}$ in C/C++ and ARM Assembly ($c = a + b \bmod p$)

```
c = a+b;                          ADD r0,r0,r1
                                  UBFX r1,r0,#0,#n
c = (c & p) + (c >> n);           ADD r0,r1,r0,ASR #n
```

Field Multiplication in $\mathbb{F}_{2^n-1}$ in C/C++ and ARM Assembly ($c = a \cdot b \bmod p$)

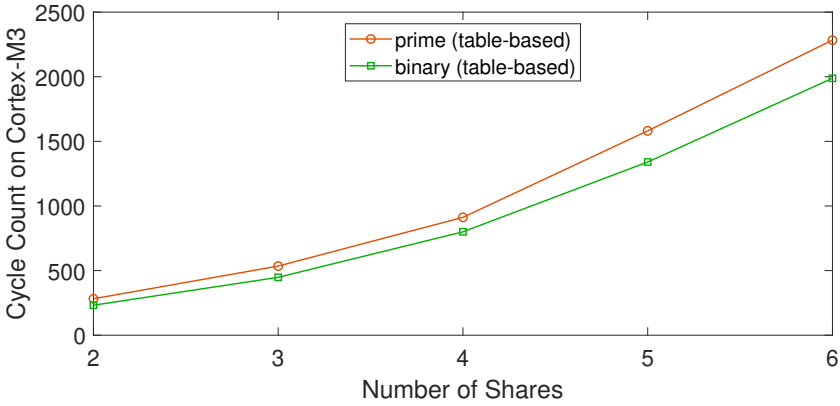
```
c = a*b;                          MUL r0,r1,r0
                                  UBFX r1,r0,#0,#n
c = (c & p) + (c >> n);           ADD r0,r1,r0,ASR #n
                                  UBFX r1,r0,#0,#n
c = (c & p) + (c >> n);           ADD r0,r1,r0,ASR #n
```

→ Only works for sufficiently small integers (< 16 bit for multiplication operands on ARM Cortex-M3)

→ If $c < p$ is strictly needed for the addition result, then $c \stackrel{?}{=} p$ needs to be checked after reduction

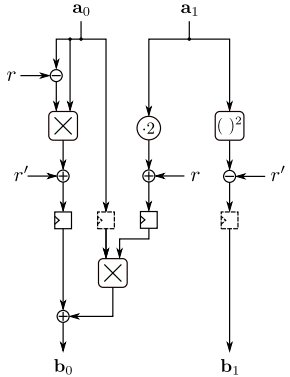
Software Case Study: Masked S-box

Naive implementation of masked $x^5 + 2$ using 3 consecutive ISW multiplications:



Dealing with Non-Linearity

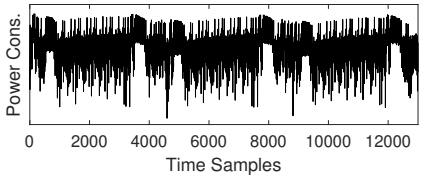
In $\mathbb{F}_p$, every $\mathbb{F}_2$-linear mapping, *e.g.* $\cdot^2$, becomes non-linear ✗
Ches 2023: new gadgets more efficient than multiplication gadgets[7]



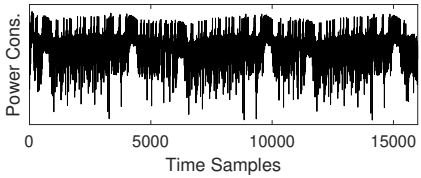
In $\mathbb{F}_{2^n-1}, 2 \cdot x$: cyclic shift of the bits
- Almost free in hardware
- Interesting property for later ...

[7] Cassiers et al., "Prime-Field Masking in Hardware and its Soundness against Low-Noise SCA Attacks".

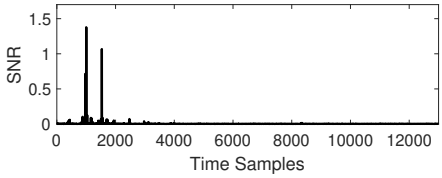
Masked $x^5 + 2$ (naive) in Software, Log/Alog tables



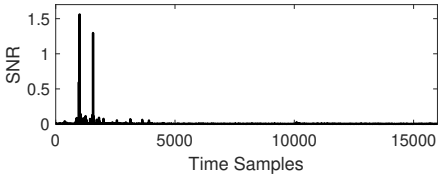
(a) Cortex-M3 sample trace, $\mathbb{F}_{2^7}$.



(b) Cortex-M3 sample trace, $\mathbb{F}_{2^7-1}$.

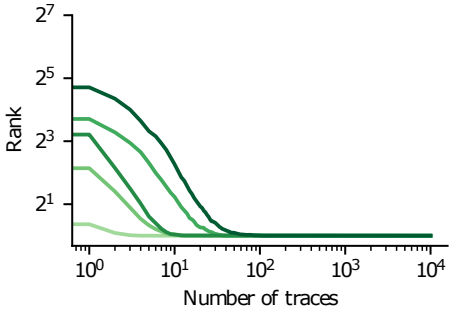


(c) SNR of input share 0, $\mathbb{F}_{2^7}$.

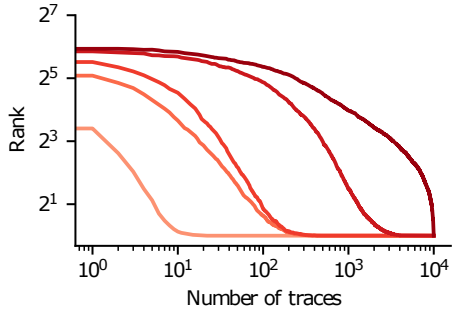


(d) SNR of input share 0, $\mathbb{F}_{2^7-1}$.

Software, Horizontal SASCA Attack for 2-6 Shares

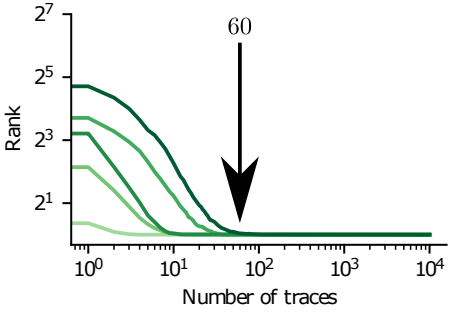


(a) $\mathbb{F}_{2^7}$

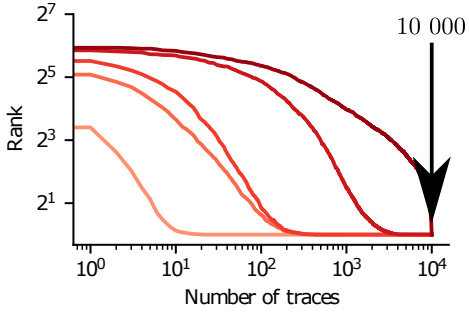


(b) $\mathbb{F}_{2^7-1}$

Software, Horizontal SASCA Attack for 2-6 Shares



(a) $\mathbb{F}_{2^7}$



(b) $\mathbb{F}_{2^7-1}$

Recap

What we know so far about a masking friendly finite field:

Recap

What we know so far about a masking friendly finite field:

- Prime characteristic, for leakage resilience

Recap

What we know so far about a masking friendly finite field:

- Prime characteristic, for leakage resilience
- Size of a Mersenne number $2^n - 1$ for implementation efficiency

$\rightarrow$ Largest encoding within n bits

$\rightarrow$ Nice implementation for modulo reductions, for $\times 2$, ...

Recap

What we know so far about a masking friendly finite field:

- Prime characteristic, for leakage resilience
- Size of a Mersenne number $2^n - 1$ for implementation efficiency

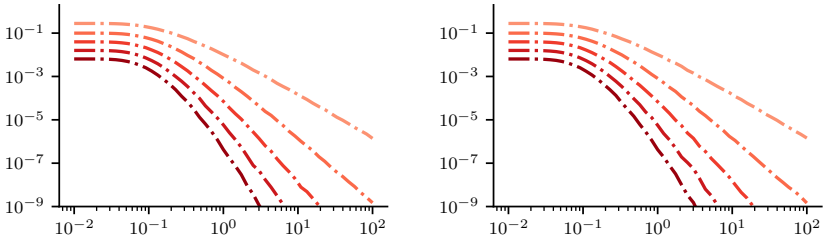
$\rightarrow$ Largest encoding within n bits

$\rightarrow$ Nice implementation for modulo reductions, for $\times 2$, ...

- **What about the size of Mersenne prime p?**

What is the Effect of Field Size ?

LSB = Least Significant Bit. One bit leaked on every share.



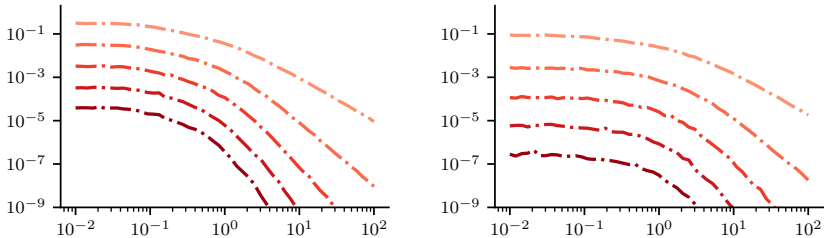
(a) LSB, $n = 7$. (b) LSB, $n = 13$.

Figure: MI vs. σ^2, for LSB.

Observation: no effect of the field size ✗

What is the Effect of Field Size ?

HW = Hamming Weight. $\approx \log(n)$ bits leaked on every share.



(a) HW, $n = 7$. (b) HW, $n = 13$.

Figure: MI vs. σ^2, for HW.

Observation: increasing the field size helps resilience ✓

Wait a Minute ...

"Central Limit Theorem" (quantitative)[8]

If each share is δ-leaky, for $\delta < 1$, then the secret is $\mathcal{O}\left(\delta^d\right)$-leaky.

First Intuition: "*the leakier the shares, the leakier the masked secret*"

[8] Béguinot et al., "Removing the Field Size Loss from Duc et al.'s Conjectured Bound for Masked Encodings"; Dziembowski, Faust, and Skórski, "Optimal Amplification of Noisy Leakages".

Wait a Minute ...

"Central Limit Theorem" (quantitative)[8]

If each share is δ-leaky, for $\delta < 1$, then the secret is $\mathcal{O}\left(\delta^d\right)$-leaky.

First Intuition: "*the leakier the shares, the leakier the masked secret*"
Counter-example: HW leaks more than LSB on each share ...

[8] Béguinot et al., "Removing the Field Size Loss from Duc et al.'s Conjectured Bound for Masked Encodings"; Dziembowski, Faust, and Skórski, "Optimal Amplification of Noisy Leakages".

Wait a Minute ...

"Central Limit Theorem" (quantitative)[8]

If each share is δ-leaky, for $\delta < 1$, then the secret is $\mathcal{O}\left(\delta^d\right)$-leaky.

First Intuition: "*the leakier the shares, the leakier the masked secret*"
Counter-example: HW leaks more than LSB on each share ... but less on the secret !

[8] Béguinot et al., "Removing the Field Size Loss from Duc et al.'s Conjectured Bound for Masked Encodings"; Dziembowski, Faust, and Skórski, "Optimal Amplification of Noisy Leakages".

Wait a Minute ...

"Central Limit Theorem" (quantitative)[8]

If each share is δ-leaky, for $\delta < 1$, then the secret is $\mathcal{O}\left(\delta^d\right)$-leaky.

First Intuition: "*the leakier the shares, the leakier the masked secret*"
Counter-example: HW leaks more than LSB on each share ... but less on the secret !

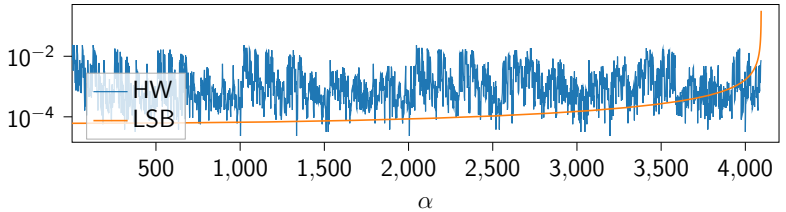
Why ?

[8] Béguinot et al., "Removing the Field Size Loss from Duc et al.'s Conjectured Bound for Masked Encodings"; Dziembowski, Faust, and Skórski, "Optimal Amplification of Noisy Leakages".

Masking $\equiv$ Convolution

Masking ≡ Convolution ≡ Fourier Analysis

"The leakage-resilience can be read in the maximum amplitude of the Fourier spectrum"



Fourier Analysis for LSB

Related works[9] and ours show secret to be $\Theta\left(\left(\frac{2}{\pi}\right)^d\right)$-leaky

Independent of p !

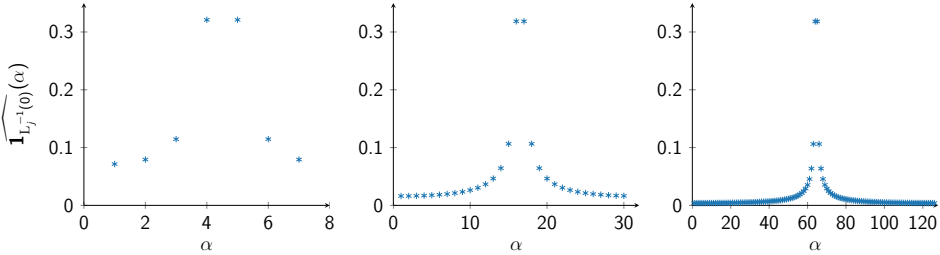


Figure: α vs $\widehat{\mathbf{1}_{\mathrm{L}_j^{-1}(0)}}(\alpha)$ for $\alpha \in \mathbb{F}^*$ in the LSB leakage model, for $p = 7, 31, 127$

[9] Benhamouda et al., "On the Local Leakage Resilience of Linear Secret Sharing Schemes".

Fourier Analysis for HW

At first glance, messier spectrum than for LSB — *i.e.* harder to analyze ...

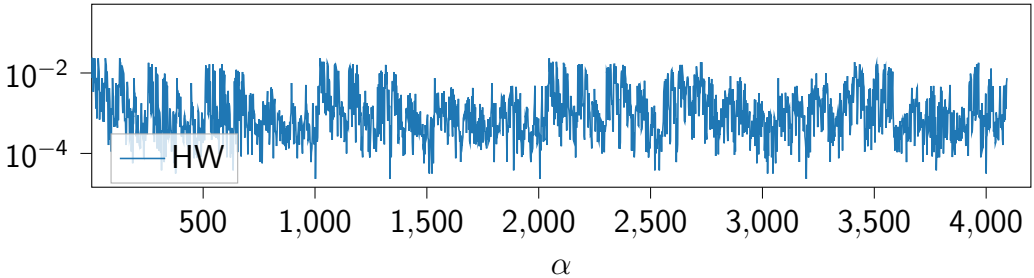


Figure: Fourier spectrum (1st half) of $\mathbf{1}_{\mathrm{hw}^{-1}(n/2)}$ and for $n = 17, p = 2^n - 1$.

Fourier Analysis for HW

More regular patterns in log scale

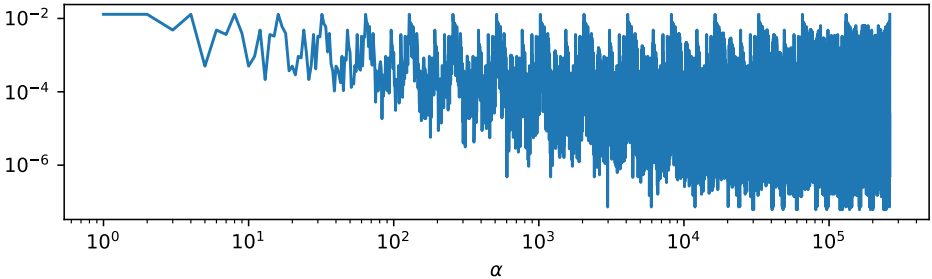


Figure: Fourier spectrum (1st half) of $\mathbf{1}_{\mathrm{hw}^{-1}(n/2)}$ and for $n = 17, p = 2^n - 1$.

Explanation

Remember that in $\mathbb{F}_{2^n-1}$, $\cdot 2$ is a cyclic shift of the bits

Explanation

Remember that in $\mathbb{F}_{2^n-1}$, $\cdot 2$ is a cyclic shift of the bits: keeps hw unchanged

Explanation

Remember that in $\mathbb{F}_{2^n-1}$, $\cdot 2$ is a cyclic shift of the bits: keeps hw unchanged
As a result: for all $\alpha \neq 0$ and for all k,

$$\left|\widehat{\mathbf{1}_h}\left(2^k\alpha\right)\right| = \left|\widehat{\mathbf{1}_h}(\alpha)\right| \quad .$$

Explanation

Remember that in $\mathbb{F}_{2^n-1}$, $\cdot 2$ is a cyclic shift of the bits: keeps hw unchanged
As a result: for all $\alpha \neq 0$ and for all k,

$$\left|\widehat{\mathbf{1}_h}\left(2^k\alpha\right)\right| = \left|\widehat{\mathbf{1}_h}(\alpha)\right| \quad .$$

Corollary: the secret is $\mathcal{O}\left(n^{1-\frac{d}{4}}\right)$-leaky $\implies$ **larger field size help !**

Explanation

Remember that in $\mathbb{F}_{2^n-1}$, $\cdot 2$ is a cyclic shift of the bits: keeps hw unchanged
As a result: for all $\alpha \neq 0$ and for all k,

$$\left|\widehat{\mathbf{1}_h}\left(2^k\alpha\right)\right| = \left|\widehat{\mathbf{1}_h}(\alpha)\right| \quad .$$

Corollary: the secret is $\mathcal{O}\left(n^{1-\frac{d}{4}}\right)$-leaky $\implies$ **larger field size help !**

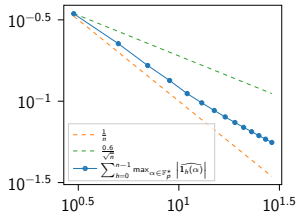


Figure: Even tighter empirically

Content

Conclusion

Working over binary fields: prone to attacks in low-noise

Working over prime fields: more leakage resilient

Opens perspectives for *Post-Quantum* crypto (works over prime fields) and new primitives for symmetric crypto

References I

Béguinot, J. et al. "Removing the Field Size Loss from Duc et al.'s Conjectured Bound for Masked Encodings". In: *Constructive Side-Channel Analysis and Secure Design - 14th International Workshop, COSADE 2023, Munich, Germany, April 3-4, 2023, Proceedings*. Ed. by E. B. Kavun and M. Pehl. Vol. 13979. Lecture Notes in Computer Science. Springer, 2023, pp. 86–104. DOI: 10.1007/978-3-031-29497-6_5. URL: https://doi.org/10.1007/978-3-031-29497-6_5.

References II

- Benhamouda, F. et al. "On the Local Leakage Resilience of Linear Secret Sharing Schemes". In: *Advances in Cryptology - CRYPTO 2018 - 38th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2018, Proceedings, Part I*. Ed. by H. Shacham and A. Boldyreva. Vol. 10991. Lecture Notes in Computer Science. Springer, 2018, pp. 531–561. DOI: 10.1007/978-3-319-96884-1_18. URL: https://doi.org/10.1007/978-3-319-96884-1_18.
- Cassiers, G. et al. "Prime-Field Masking in Hardware and its Soundness against Low-Noise SCA Attacks". In: *IACR Trans. Cryptogr. Hardw. Embed. Syst.* 2023.2 (2023), pp. 482–518. DOI: 10.46586/TCHES.V2023.I2.482-518. URL: https://doi.org/10.46586/tches.v2023.i2.482-518.

References III

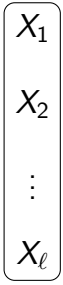
- Duc, A., S. Dziembowski, and S. Faust. "Unifying Leakage Models: From Probing Attacks to Noisy Leakage". In: *J. Cryptology* 32.1 (2019), pp. 151–177. DOI: 10.1007/s00145-018-9284-1. URL: https://doi.org/10.1007/s00145-018-9284-1.
- Dziembowski, S., S. Faust, and M. Skórski. "Optimal Amplification of Noisy Leakages". In: *Theory of Cryptography - 13th International Conference, TCC 2016-A, Tel Aviv, Israel, January 10-13, 2016, Proceedings, Part II*. Ed. by E. Kushilevitz and T. Malkin. Vol. 9563. Lecture Notes in Computer Science. Springer, 2016, pp. 291–318. DOI: 10.1007/978-3-662-49099-0_11. URL: https://doi.org/10.1007/978-3-662-49099-0_11.

References IV

- Koç, Ç. K., ed. *Cryptographic Engineering*. Springer, 2009. ISBN: 978-0-387-71816-3. DOI: 10.1007/978-0-387-71817-0. URL: https://doi.org/10.1007/978-0-387-71817-0.
- Mangard, S., E. Oswald, and T. Popp. *Power analysis attacks - revealing the secrets of smart cards*. Springer, 2007. ISBN: 978-0-387-30857-9.
- Stromberg, K. "Probabilities on a Compact Group". In: *Transactions of the American Mathematical Society* 94.2 (1960), pp. 295–309. ISSN: 00029947. URL: http://www.jstor.org/stable/1993313.

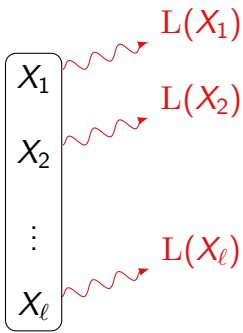
Provable Masking over the whole Circuit

Consider a leaky gadget (*e.g.*, emulating a $\otimes$) with ℓ wires:



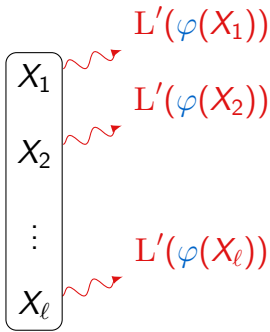
Provable Masking over the whole Circuit

Consider a leaky gadget (*e.g.*, emulating a $\otimes$) with ℓ wires:



Provable Masking over the whole Circuit

Consider a leaky gadget (*e.g.*, emulating a $\otimes$) with ℓ wires:

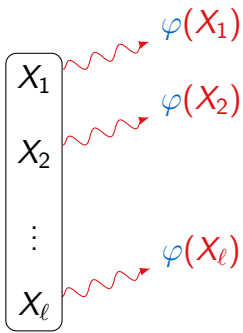


If $\mathrm{L}(\cdot)$ noisy enough, can be *simulated* by a random probing adversary: $\varphi(x)$ reveals x with some probability ϵ[10]

[10] Duc, Dziembowski, and Faust, "Unifying Leakage Models: From Probing Attacks to Noisy Leakage".

Provable Masking over the whole Circuit

Consider a leaky gadget (*e.g.*, emulating a $\otimes$) with ℓ wires:



Upper bound by removing L' (Data Processing Inequality)

Security against a Random Probing Adversary

Assuming the gadget to be "properly" implemented, at least d out of ℓ wires must be revealed to leak the secret for a successful attack:

Theorem (Chernoff)

If ℓ wires, each independently revealed with proba. ϵ:

$$\Pr\left(\textit{At least } d \textit{ wires revealed}\right) \leq \left(\frac{2e \cdot \ell \cdot \epsilon}{d}\right)^d$$

Still exponential security ✓

Strong requirements on the noise level to get non-trivial values of ϵ ✗