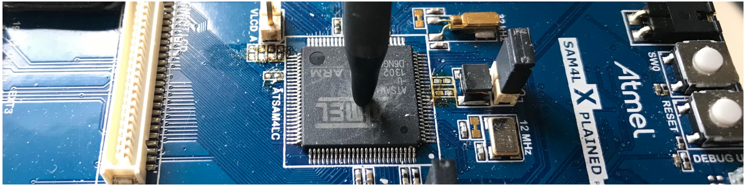


Masking Symmetric Crypto in a Low Noise Environment

Analysis and Evaluation

Loïc Masure

ASK Workshop, Guangzhou, December 1^{st}

cnrs LIRMM UNIVERSITÉ DE MONTPELLIER

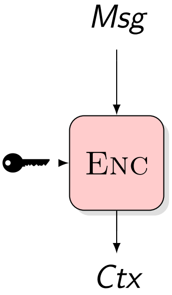
Content

Joint Work

Joint work with

- Thorben Moos, FX Standaert, Gaëtan Cassiers, Charles Momin, Pierrick Méaux (UCLouvain)
- Maximilian Orlt, Elena Micheli, Sebastian Faust (TU Darmstadt)
- Julien Béguinot, Wei Cheng, Sylvain Guilley, Yi Liu, Olivier Rioul (Télécom Paris)

Context : Side-Channel Analysis (SCA)

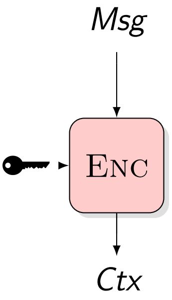


: *N bits*

Black-box cryptanalysis: 2^N

Context : Side-Channel Analysis (SCA)

"Cryptographic algorithms don't run on paper,



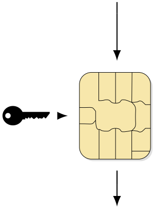
: *N bits*

Black-box cryptanalysis: 2^N

Context : Side-Channel Analysis (SCA)

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

Msg



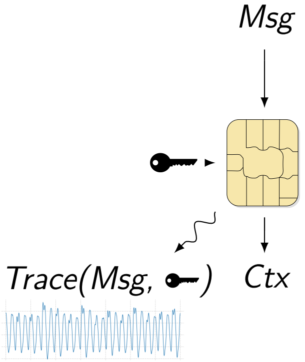
Ctx

: *N bits*

Black-box cryptanalysis: 2^N

Context : Side-Channel Analysis (SCA)

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



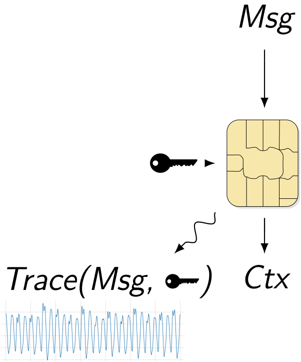
: N bits

Black-box cryptanalysis: 2^N

Trace : power, EM, acoustics, runtime, . . .

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 Analysis: $2^n \cdot \frac{N}{n}, n \ll N$

Trace : power, EM, acoustics, runtime, . . .

Masking: what is that ?

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

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

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

Masking: what is that ?

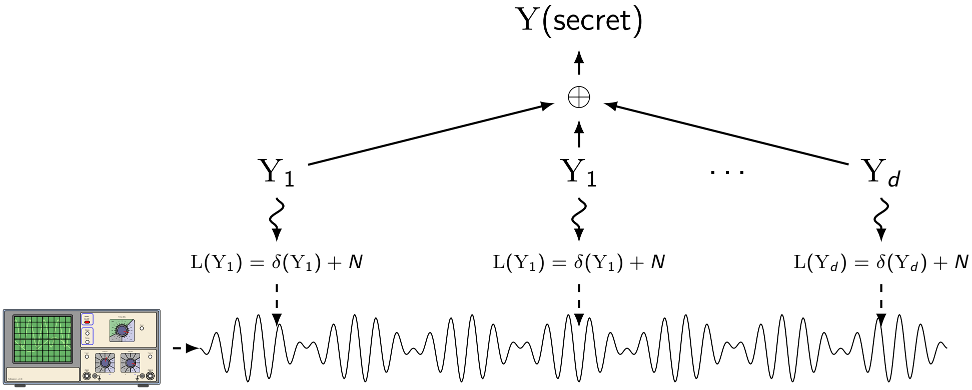
Masking, aka *MPC on silicon*: linear secret sharing over a finite field $(\mathbb{F}, \star, \cdot)$

$$\begin{tikzcd}
 & \mathrm{Y}(\text{secret}) & & \\
 & \oplus \arrow[u] & & \\
\mathrm{Y}_1 \arrow[ur] & \mathrm{Y}_1 \arrow[u] & \cdots & \mathrm{Y}_d \arrow[ull]
\end{tikzcd}$$

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

Write each operation as a polynomial (Lagrange interpolation).
One polynomial is made of:

How to Calculate over Masked Data? Outline

Write each operation as a polynomial (Lagrange interpolation).
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

Write each operation as a polynomial (Lagrange interpolation).
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

Write each operation as a polynomial (Lagrange interpolation).
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

Write each operation as a polynomial (Lagrange interpolation).
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 ; needs to *compress* into d shares.

$\rightarrow$ Introduce *fresh* randomness somewhere.

How to Calculate over Masked Data? Outline

Write each operation as a polynomial (Lagrange interpolation).
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 ; needs to *compress* into d shares.

$\rightarrow$ Introduce *fresh* randomness somewhere.

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

Content

The Effect of Masking

Simulation, for $\mathbb{F}_{2^n}$: $\mathrm{L}(\mathrm{Y}_i) = \mathsf{lsb}(\mathrm{Y}_i) + \mathcal{N}(0;\sigma^2)$, lsb = Least Sig. Bit

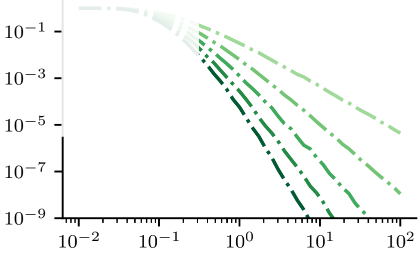


Figure: $MI(Y; Trace)$ vs. σ^2, $2 \leq d \leq 6$

Observation: "Masking amplifies noise"
Constant gap between each curve (log scale)

$\iff$

exponential security w.r.t. #shares d

The Effect of Masking

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

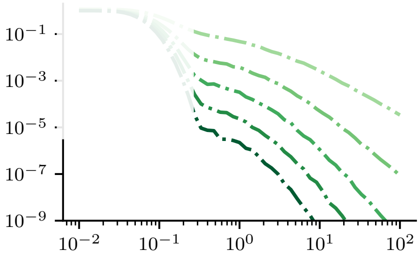


Figure: $MI(Y; Trace)$ vs. σ^2, $2 \leq d \leq 6$

Masking in a Low-Noise Setting

Does masking always work in a low-noise setting ?

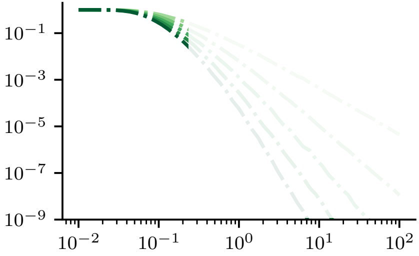


Figure: $MI(Y; Trace)$ vs. σ^2, $2 \leq d \leq 6$

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 ?

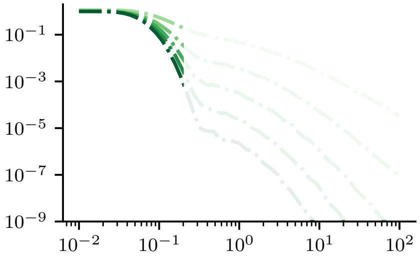


Figure: $MI(Y; Trace)$ vs. σ^2, $2 \leq d \leq 6$

Observation:

Secret always leaks > 1 bit, regardless of d

Explanation:

$\mathrm{hw}(\mathrm{Y}_1 \oplus \ldots \oplus \mathrm{Y}_d) = \sum_i \mathrm{hw}(\mathrm{Y}_i) - 2 \cdot \left(\ldots \right)$

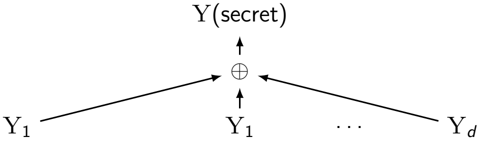
Parity of $\mathrm{hw}(\mathrm{Y})$: **cosets of** $\mathbb{F}_{2^n}$

Corollary: parallelism is no cure either

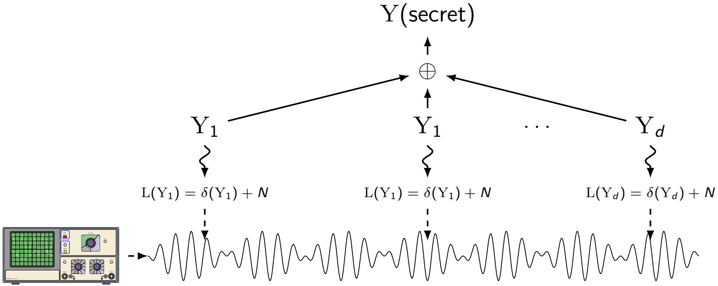
Why these Observations?

Y(secret)

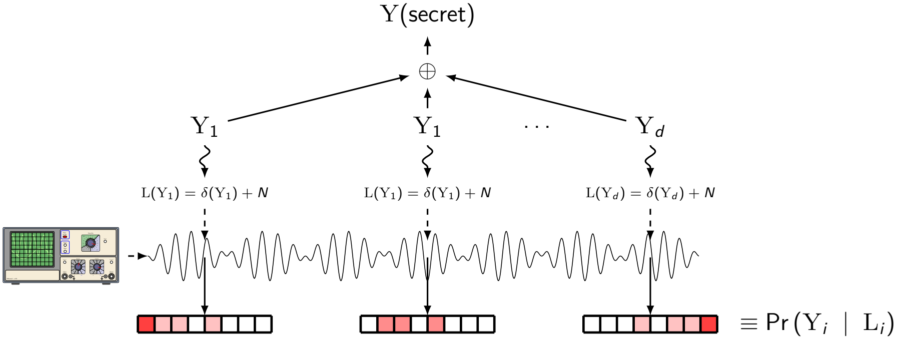
Why these Observations?



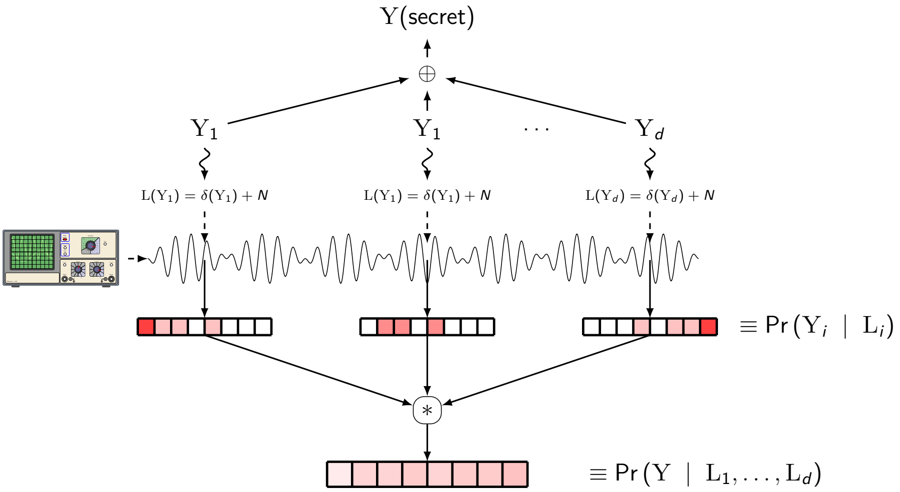
Why these Observations?



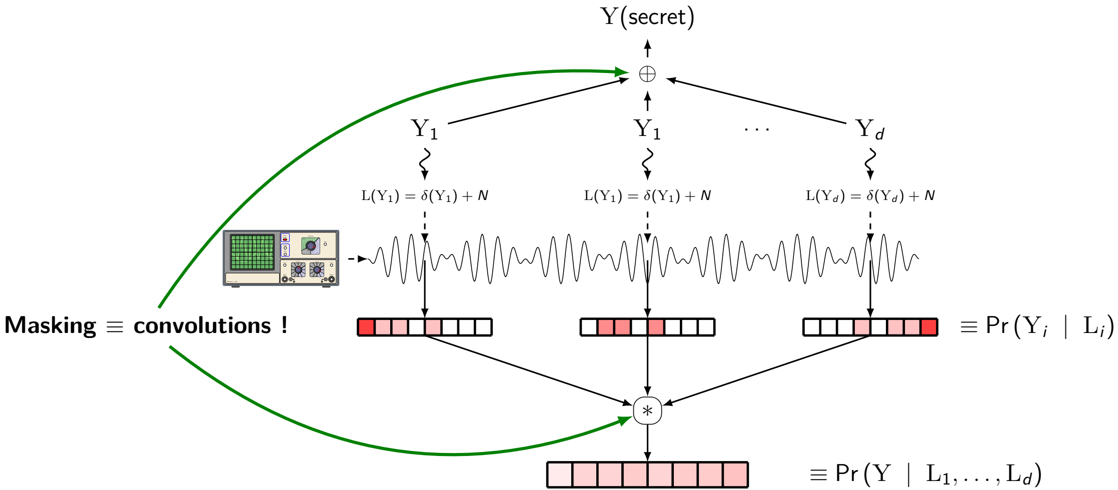
Why these Observations?



Why these Observations?



Why these Observations?



The Effect of Masking

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

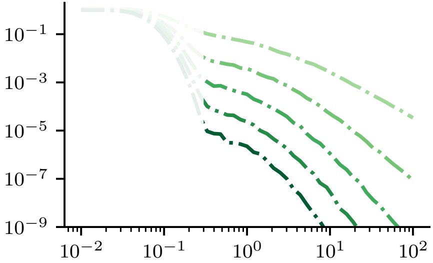
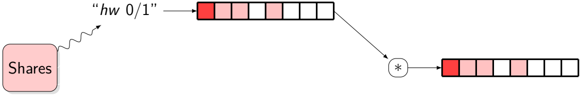


Figure: $MI(Y; Trace)$ vs. σ^2, $2 \leq d \leq 6$

Explanation: "*Masking amplifies noise*"



The Effect of Masking

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

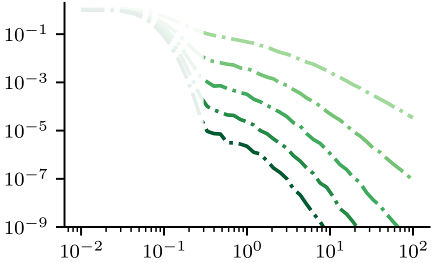
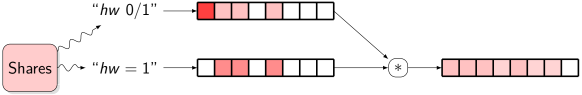


Figure: $MI(Y; Trace)$ vs. σ^2, $2 \leq d \leq 6$

Explanation: "*Masking amplifies noise*"



The Effect of Masking

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

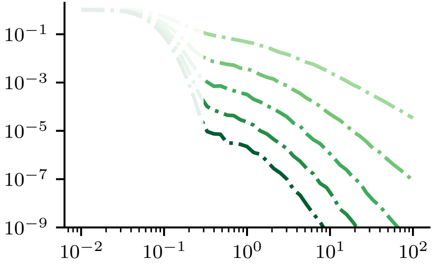
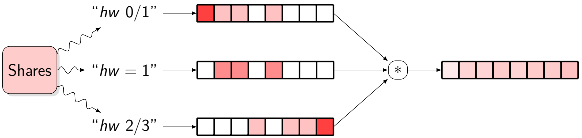


Figure: $MI(Y; Trace)$ vs. σ^2, $2 \leq d \leq 6$

Explanation: "*Masking amplifies noise*"



$\approx$ Central Limit Thm. in a finite group

$\rightarrow$ Gaussian in $\mathbb{R}$ $\equiv$ uniform in $\mathbb{F}$

Masking in a Low-Noise Setting

Does masking always work in a low-noise setting ?

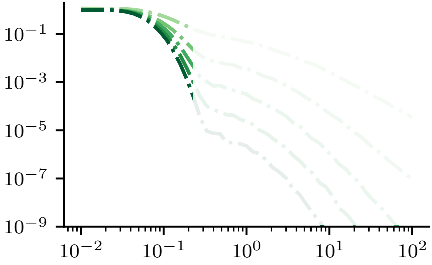
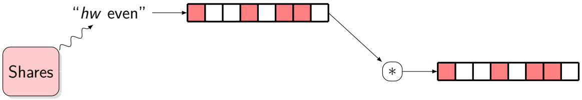


Figure: $MI(Y; Trace)$ vs. σ^2, $2 \leq d \leq 6$

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 ?

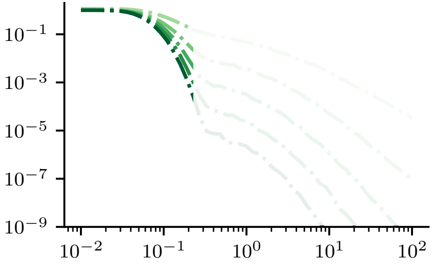
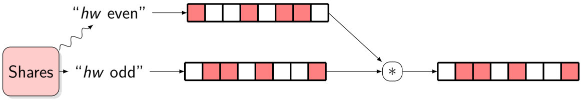


Figure: $MI(Y; Trace)$ vs. σ^2, $2 \leq d \leq 6$

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 ?

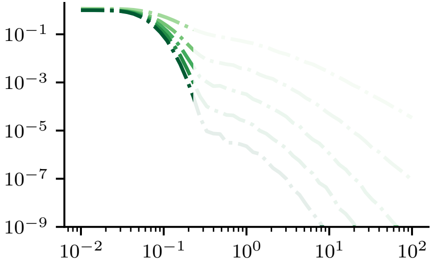
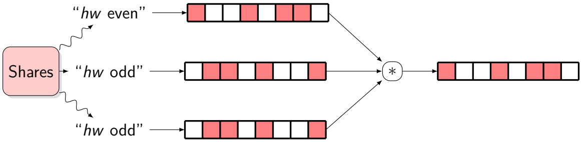


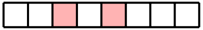
Figure: $MI(Y; Trace)$ vs. σ^2, $2 \leq d \leq 6$

Explanation:

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



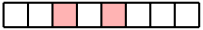
Conditions for Sound Masking

What conditions the distributions  of each share must fit?

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

[2] 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".

Conditions for Sound Masking

What conditions the distributions  of each share must fit?

"Central Limit Theorem" (qualitative)[1]

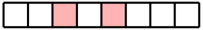
Conv. to uniform $\iff$ support *not* contained in any non-trivial coset of $\mathbb{F}$

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

[2] 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".

[3] D: KL divergence, total variation, Euclidean norm.

Conditions for Sound Masking

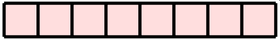
What conditions the distributions  of each share must fit?

"Central Limit Theorem" (qualitative)[1]

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

"Central Limit Theorem" (quantitative)[2]

Assume the p.m.f.s of each share to be δ-close[3] to the uniform:

D (, ) $\leq \delta < 1$,

then the p.m.f. of the secret is $\mathcal{O}\left(\delta^d\right)$-close to the uniform.

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

[2] 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".

[3] D: KL divergence, total variation, Euclidean norm.

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

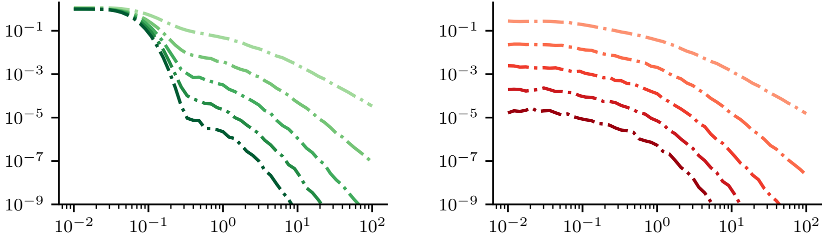
Solution 2: Choose $\mathbb{F}$ without any non-trivial subgroup, *i.e.*, $\mathbb{F}_p$, p prime:

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

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

  Then build crypto upon it ✓

title



(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 ($c = a \cdot b \bmod p$)

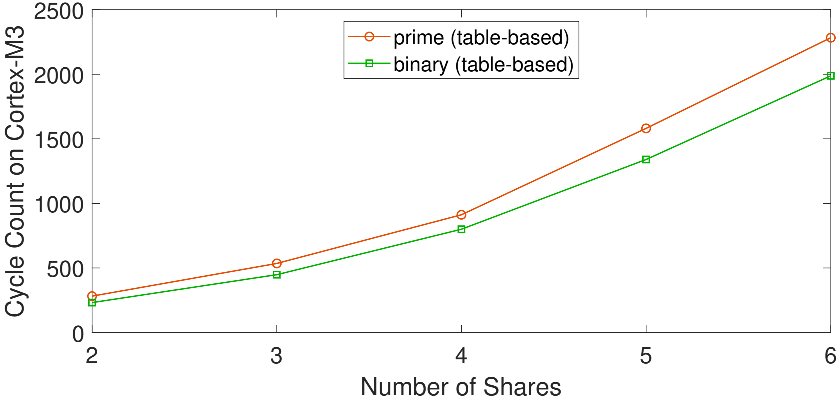
```
c = a*b;                              MUL r0,r1,r0
                                      UBFX r1,r0,#0,#n
c = (c & p) + (c >> n);               ADD r0,r1,r0,ASR #n
                                      UBFX r1,r0,#0,#n
c = (c & p) + (c >> n);               ADD r0,r1,r0,ASR #n
```

→ 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

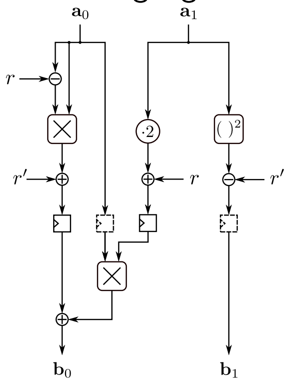
Naive implementation of masked $x^5 + 2$ using 3 consecutive ISW multiplications:



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[4]

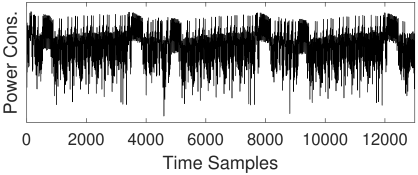


In $\mathbb{F}_{2^n-1}$, $2 \cdot x$: cyclic shift of the bits

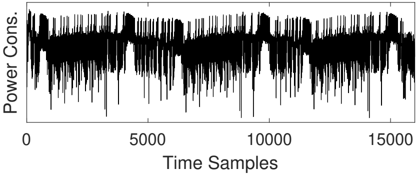
- Almost free in hardware
- Interesting property for later ...

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

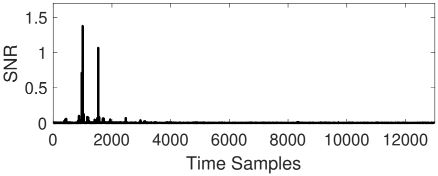
Masked $x^5 + 2$ (naive) in Software, Log/Alog tables



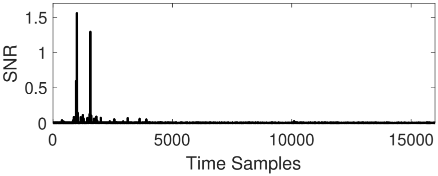
(a) Cortex-M3 sample trace, $\mathbb{F}_{2^7}$.



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

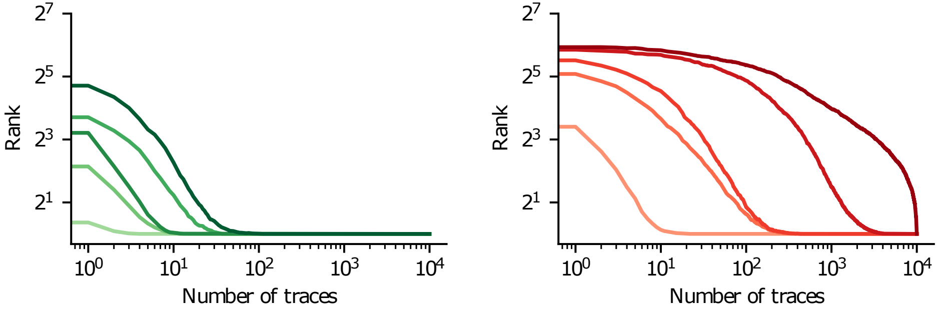


(c) SNR of input share 0, $\mathbb{F}_{2^7}$.



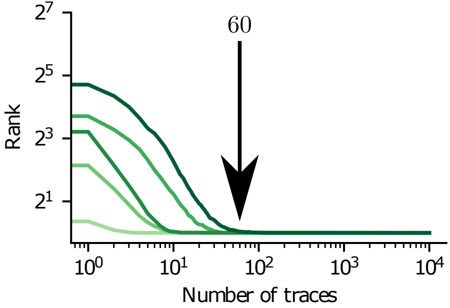
(d) SNR of input share 0, $\mathbb{F}_{2^7-1}$.

Software, Horizontal SASCA Attack for 2-6 Shares

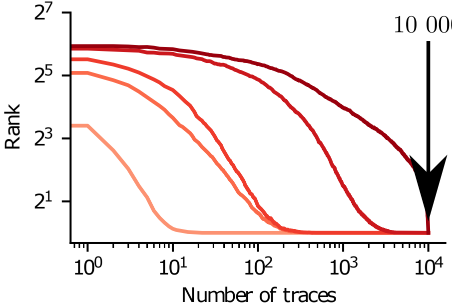


(a) $\mathbb{F}_{2^7}$ (b) $\mathbb{F}_{2^7-1}$

Software, Horizontal SASCA Attack for 2-6 Shares



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



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

Content

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

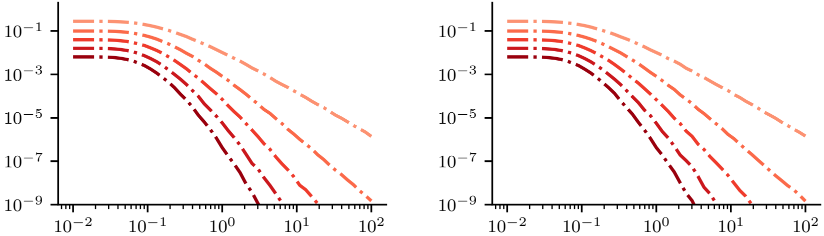
$\rightarrow$ Largest encoding within n bits

$\rightarrow$ 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.



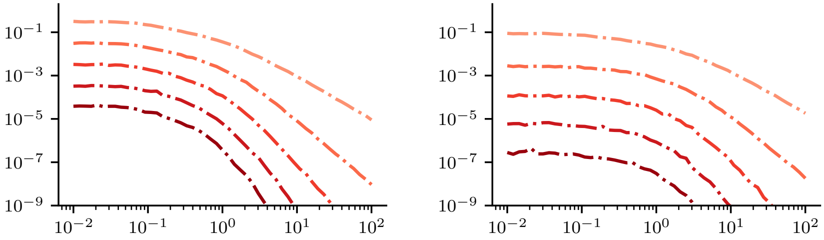
(a) LSB, $n = 7$. (b) LSB, $n = 13$.

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)[5]

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*"

[5] 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)[5]

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 . . .

[5] 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)[5]

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 !

[5] 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)[5]

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 !

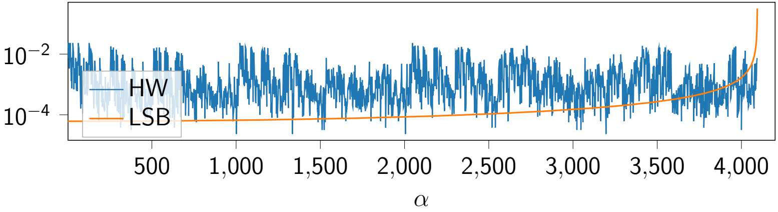
Why ?

[5] 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[6] and ours show secret to be $\Theta\left(\left(\frac{2}{\pi}\right)^d\right)$-leaky

Independent of p !

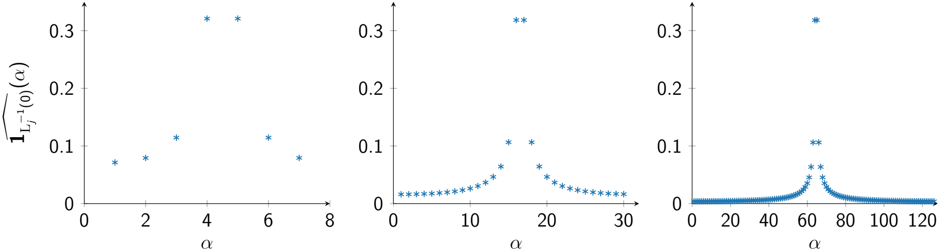


Figure: α vs $\widehat{\mathbf{1}_{\mathrm{L}_j^{-1}(0)}}(\alpha)$ for $\alpha \in \mathbb{F}^*$ in the LSB leakage model, for $p = 7, 31, 127$

[6] 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 ...

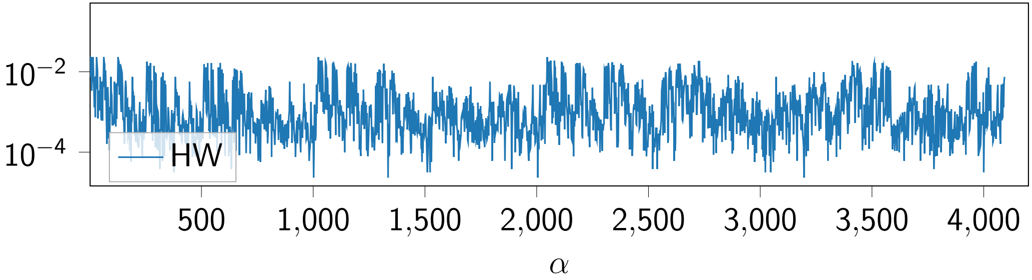


Figure: Fourier spectrum (1st half) of $\mathbf{1}_{\mathrm{hw}^{-1}(n/2)}$ and for $n = 17, p = 2^n - 1$.

Fourier Analysis for HW

More regular patterns in log scale

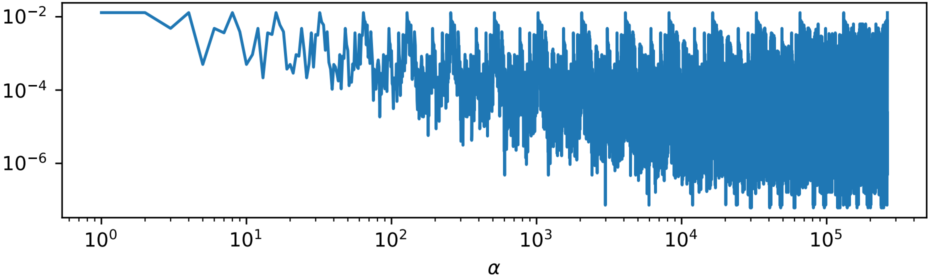


Figure: Fourier spectrum (1st half) of $\mathbf{1}_{\mathrm{hw}^{-1}(n/2)}$ and for $n = 17, p = 2^n - 1$.

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 !**

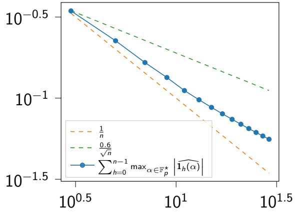


Figure: Even tighter empirically

Content

Conclusion

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

Working over prime fields: more leakage resilient

$\rightarrow$ Mersenne primes: good for implementation and for analysis

$\rightarrow$ Field size acts as a surrogate of noise $\mathcal{O}\left((\sigma^2)^d\right) \implies \mathcal{O}\left(f(n)^d\right)$

Let's build symmetric crypto over middle-size prime fields !

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

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.

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.