Exploiting Algebraic Structures in Probing Security

Maxime Plançon[1,2]

[1]IBM Research Zurich [2]ETH Zurich

Outline of the talk

→ Introduction to masking and side-channel attacks

→ Technical and result overview

→ Performance comparison

Side-channel attacks

Side-channel attacks refer to all attacks extracting information from the physical device running a cryptographic algorithm.

- Timing attacks
- Power analysis
- Electromagnetic radiation analysis
- etc

Security game

We have a circuit $\mathcal{C}$ that manipulates secret random variables $(x_1, \ldots, x_n)$. The adversary $\mathcal{A}$ plays the following game:

1. $\mathcal{A}$ learns some information on the wires $\mathcal{W}$
2. $\mathcal{A}$ outputs a guess $(y_1, \ldots, y_n)$
3. $\mathcal{A}$ wins if $(y_1, \ldots, y_n) = (x_1, \ldots, x_n)$

Security game

We have a circuit $\mathcal{C}$ that manipulates secret random variables $(x_1, \ldots, x_n)$. The adversary $\mathcal{A}$ plays the following game:

1. $\mathcal{A}$ learns some information on the wires $\mathcal{W}$
2. $\mathcal{A}$ outputs a guess $(y_1, \ldots, y_n)$
3. $\mathcal{A}$ wins if $(y_1, \ldots, y_n) = (x_1, \ldots, x_n)$

We say that $\mathcal{C}$ is secure when $\mathcal{A}$ has no advantage over an adversary that skips step 1.

Adversary model

t-threshold probing model:
The adversary picks and learns t wires of the circuit [ISW03]

r-region probing model:
Let $\mathcal{C}_1, \ldots, \mathcal{C}_m$ be a partition of $\mathcal{C}$ into subcircuits. The adversary picks and learns $r|\mathcal{C}_i$ wires in each of the subcircuits[ADF16]

Masking: Secret sharing the sensitive variables

Arithmetic encoding: $x_0 + ... + x_{d-1} = x$ with $\mathbf{x} \in \mathbb{F}^d$ distributed uniformly conditioned on $\mathbf{x}^T \mathbf{1} = x$.

Masking: Secret sharing the sensitive variables

Arithmetic encoding: $x_0 + ... + x_{d-1} = x$ with $\mathbf{x} \in \mathbb{F}^d$ distributed uniformly conditioned on $\mathbf{x}^T\mathbf{1} = x$.

Geometric interpretation: the encodings of 0 are the hyperplane H orthogonal to $\mathbf{1}^\perp$

Masking: Secret sharing the sensitive variables

Arithmetic encoding: $x_0 + ... + x_{d-1} = x$ with $\mathbf{x} \in \mathbb{F}^d$ distributed uniformly conditioned on $\mathbf{x}^T \mathbf{1} = x$.

Geometric interpretation: the encodings of 0 are the hyperplane H orthogonal to $\mathbf{1}^\perp$

Algebraic interpretation: the encodings of 0 are the ideal $(X-1) \cdot R$ with $R = \mathbb{F}[X]/X^d$, i.e the polynomials $\mathbf{x} \in \mathbb{F}^d$ such that $\mathbf{x}(1) = 0$.

Generalization: ω-encoding

Fix an element $\omega \in \mathbb{F}$, $x_0 + \omega x_1 + \ldots + \omega^{d-1} x_{d-1} = x$ with $\mathbf{x} \in \mathbb{F}^d$ distributed uniformly conditioned on $\mathbf{x}^T \boldsymbol{\omega}_d = x$ [GJR18].

Geometric interpretation: the encodings of 0 are the hyperplane H orthogonal to $\boldsymbol{\omega}_d^{\perp}$ with $\boldsymbol{\omega}_d = (1 \; \omega \; \ldots \; \omega^{d-1})$.

Algebraic interpretation: the encodings of 0 are the ideal $(X - \omega) \cdot R$ with $R = \mathbb{F}[X]/X^d$, i.e the polynomials $\mathbf{x} \in \mathbb{F}^d$ such that $\mathbf{x}(\omega) = 0$.

Masked compiler

The idea of masking to protect a circuit $\mathcal{C}$ is:

1. Replace each secret random variable with an encoding

Masked compiler

The idea of masking to protect a circuit $\mathcal{C}$ is:

1. Replace each secret random variable with an encoding
2. Replace each gate with a secure gadget

Masked compiler

The idea of masking to protect a circuit $\mathcal{C}$ is:

1. Replace each secret random variable with an encoding
2. Replace each gate with a secure gadget
3. (If needed) Refresh the randomness of the encodings every now and then with a refresh gadget

$\underline{/!\backslash}$ Even if $\mathcal{C}_1$ and $\mathcal{C}_2$ are probing secure, their composition in general is not $\underline{/!\backslash}$

Gadgets for arithmetic circuits

Let $\mathbf{a}, \mathbf{b}$ be encodings of respectively a, b.

- Addition gadget: $\mathbf{c} = (a_0 + b_0, \ldots, a_{d-1} + b_{d-1})$ is $d-1$-probing secure, and $\mathbf{c}(1) = \mathbf{a}(1) + \mathbf{b}(1)$.

Gadgets for arithmetic circuits

Let $\mathbf{a}, \mathbf{b}$ be encodings of respectively a, b.

- Addition gadget: $\mathbf{c} = (a_0 + b_0, \ldots, a_{d-1} + b_{d-1})$ is $d-1$-probing secure, and $\mathbf{c}(1) = \mathbf{a}(1) + \mathbf{b}(1)$.

- Multiplication gadget: Need more work to compute $\mathbf{c}$ s.t $\mathbf{c}(1) = \mathbf{a}(1)\mathbf{b}(1)$ *securely*. Overwhelmingly most used is [ISW03].

Gadgets for arithmetic circuits

Let $\mathbf{a}, \mathbf{b}$ be encodings of respectively a, b.

- Addition gadget: $\mathbf{c} = (a_0 + b_0, \ldots, a_{d-1} + b_{d-1})$ is $d-1$-probing secure, and $\mathbf{c}(1) = \mathbf{a}(1) + \mathbf{b}(1)$.

- Multiplication gadget: Need more work to compute $\mathbf{c}$ s.t $\mathbf{c}(1) = \mathbf{a}(1)\mathbf{b}(1)$ *securely*. Overwhelmingly most used is [ISW03].

$\rightarrow$ ISW computes all the $a_i b_j$ and recombines these products with $d(d-1)/2$ random elements

Outline of the talk

→ Introduction to masking and side-channel attacks

→ Technical and result overview

→ Performance comparison

Opening claim

Let $\mathbb{F}$ be a field, K be a subfield of $\mathbb{F}$ and $\omega \in \mathbb{F}$. Let $\mathcal{C}$ be a circuit taking as input a uniform $\boldsymbol{\omega}_d$-encoding $\mathbf{x}$.

Opening claim

Let $\mathbb{F}$ be a field, K be a subfield of $\mathbb{F}$ and $\omega \in \mathbb{F}$. Let $\mathcal{C}$ be a circuit taking as input a uniform $\boldsymbol{\omega}_d$-encoding $\mathbf{x}$.

Assume $\mathcal{C}$ is such that for all set of probes $P \subset \mathcal{W}$, all the probes in P are of the form $p(\mathbf{x}) = \mathbf{p}^T\mathbf{x}$, for some vector $\mathbf{p} \in K^d$.

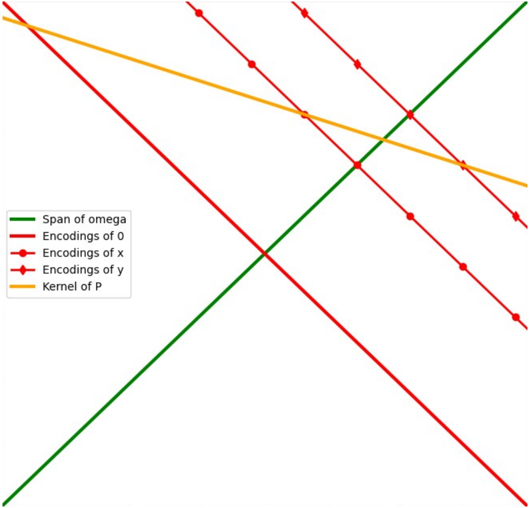
Opening claim

Let $\mathbb{F}$ be a field, K be a subfield of $\mathbb{F}$ and $\omega \in \mathbb{F}$. Let $\mathcal{C}$ be a circuit taking as input a uniform $\boldsymbol{\omega}_d$-encoding $\mathbf{x}$.

Assume $\mathcal{C}$ is such that for all set of probes $P \subset \mathcal{W}$, all the probes in P are of the form $p(\mathbf{x}) = \mathbf{p}^T\mathbf{x}$, for some vector $\mathbf{p} \in K^d$.

Then, if $\deg_K(\omega) \geqslant d$, we have that $\mathcal{C}$ is $d-1$-probing secure.

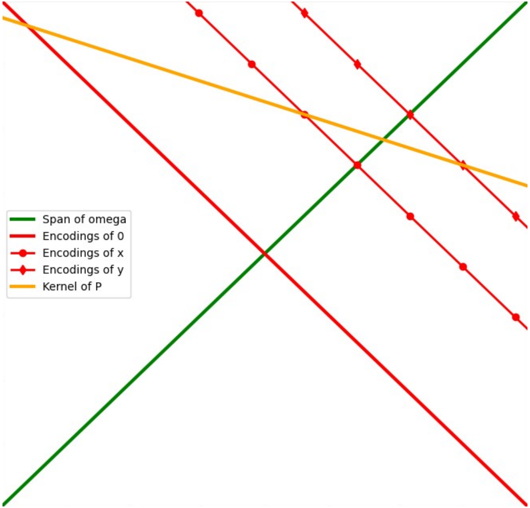
Proof sketch



Adversary's view:

· $\mathbf{x}$ is uniform over a shifted copy of $\ker \mathbf{P}$.

Proof sketch

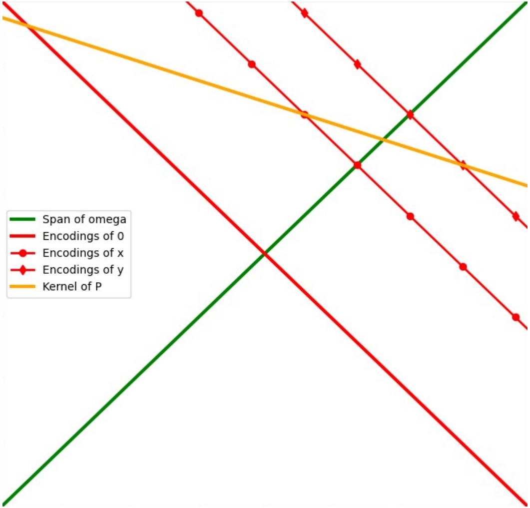


Adversary's view:

· $\mathbf{x}$ is uniform over a shifted copy of $\ker \mathbf{P}$.

· $\boldsymbol{\omega}_d$ is NOT orthogonal to $\ker \mathbf{P}$

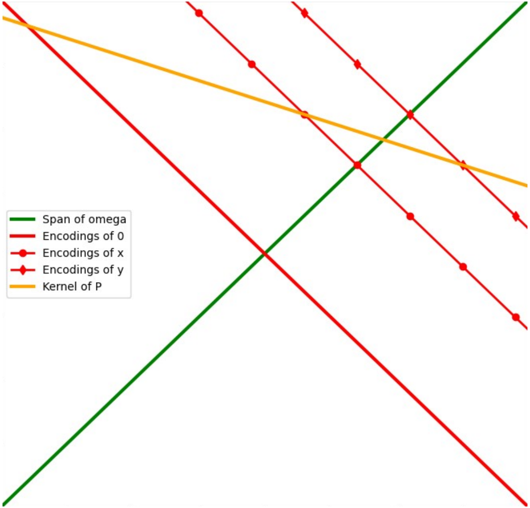
Proof sketch



Adversary's view:

- $\mathbf{x}$ is uniform over a shifted copy of $\ker \mathbf{P}$.
- $\boldsymbol{\omega}_d$ is NOT orthogonal to $\ker \mathbf{P}$
- $\mathbb{P}(\omega^T \mathbf{x} = x) =$ the volume of the intersection $H \cap \ker \mathbf{P}$

Proof sketch



Adversary's view:

· $\mathbf{x}$ is uniform over a shifted copy of $\ker \mathbf{P}$.

· $\boldsymbol{\omega}_d$ is NOT orthogonal to $\ker \mathbf{P}$

· $\mathbb{P}(\omega^T \mathbf{x} = x) =$ the volume of the intersection $H \cap \ker \mathbf{P}$

· Therefore $\boldsymbol{\omega}_d^T \mathbf{x}$ is uniform

Abstract security notion

Reducible-To-Independent-K-Linear

Let $\mathcal{C}$ be a circuit and $\mathbf{x}_1, \ldots, \mathbf{x}_n$ be n uniform and independent encodings. We say that $\mathcal{C}$ is RTIK when for all set of probes P, there exists a set of probes $Q = (Q_i)_{1 \leqslant i \leqslant n}$ such that

1. Q contains more information than P
2. $|Q_i| \leqslant |P|$
3. Every probe $q \in Q_i$ is of the form $\mathbf{q}^T \mathbf{x}_i$ for some vector $\mathbf{q} \in K^d$.

Important properties of RTIK circuits

Composition

If $\mathcal{C}_1$ and $\mathcal{C}_2$ are RTIK, so is their composition (in all known examples*).

Important properties of RTIK circuits

Composition

If $\mathcal{C}_1$ and $\mathcal{C}_2$ are RTIK, so is their composition (in all known examples*).

Security

If $\mathcal{C}$ is RTIK, then $\mathcal{C}$ is secure in the region-probing model.

Security notion for refresh gadgets

The security notion to refresh the randomness of ω-encodings between RTIK circuits is weaker.

Examples of randomness-optimal refresh gadgets that use $d-1$ random field elements

Outline of the talk

→ Introduction to masking and side-channel attacks

→ Technical and result overview

→ Performance comparison

Performance comparison of multiplication gadgets

	ISW	GPRV	This work
Bilinear mul	d^2	$2d$	$d^{\log 3}$
Randomness	$\frac{d(d-1)}{2}$	$d\log(2d)^*$	d
t-threshold	$d-1$	$d/2-1$	$d-1$

*: The input and output of GPRV must refreshed, which implies a bigger cost in randomness not taken into account in the table.

Performance comparison of multiplication gadgets in the AES field

$d = 2$	ISW	GPRV	This work
Bilinear mul	4	4	3
Randomness	1	4	1

$d = 4$	ISW	GPRV	This work
Bilinear mul	16	8	9
Randomness	6	12	4

$d = 8$	ISW	GPRV	This work
Bilinear mul	64	16	27
Randomness	28	32	8

Take away

→ We propose an efficient arithmetic circuit masked compiler in the region-probing model

Take away

$\rightarrow$ We propose an efficient arithmetic circuit masked compiler in the region-probing model

$\rightarrow$ Number of shares bounded by the algebraic structure available $d \leqslant [\mathbb{F} : K]$

Take away

→ We propose an efficient arithmetic circuit masked compiler in the region-probing model

→ Number of shares bounded by the algebraic structure available $d \leqslant [\mathbb{F} : K]$

→ Extra efficiency when $d | [\mathbb{F} : K]$

Take away

$\rightarrow$ We propose an efficient arithmetic circuit masked compiler in the region-probing model

$\rightarrow$ Number of shares bounded by the algebraic structure available $d \leqslant [\mathbb{F} : K]$

$\rightarrow$ Extra efficiency when $d | [\mathbb{F} : K]$

$\rightarrow$ Find out about cool techniques: eprint 2022/1540, or come chat with me !

Open questions and future work

- Implementation to determine whether this work is an improvement in practice

Open questions and future work

- Implementation to determine whether this work is an improvement in practice

- Prove the security in more realistic models

Open questions and future work

- Implementation to determine whether this work is an improvement in practice
- Prove the security in more realistic models
- Formal verification of security for implementations

Open questions and future work

- Implementation to determine whether this work is an improvement in practice
- Prove the security in more realistic models
- Formal verification of security for implementations
- Efficient gadgets for equality/inequality test, conversions

Open questions and future work

- Implementation to determine whether this work is an improvement in practice
- Prove the security in more realistic models
- Formal verification of security for implementations
- Efficient gadgets for equality/inequality test, conversions
- Lift the upper bound on the number of probes

Thank you for your attention !

References I

Marcin Andrychowicz, Stefan Dziembowski, and Sebastian Faust, *Circuit compilers with o (1/\log (n)) o (1/log (n)) leakage rate*, Advances in Cryptology–EUROCRYPT 2016: 35th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Vienna, Austria, May 8-12, 2016, Proceedings, Part II 35, Springer, 2016, pp. 586–615.

References II

- Dahmun Goudarzi, Antoine Joux, and Matthieu Rivain, *How to securely compute with noisy leakage in quasilinear complexity*, International Conference on the Theory and Application of Cryptology and Information Security, Springer, 2018, pp. 547–574.
- Yuval Ishai, Amit Sahai, and David Wagner, *Private circuits: Securing hardware against probing attacks*, Advances in Cryptology-CRYPTO 2003: 23rd Annual International Cryptology Conference, Santa Barbara, California, USA,

References III

August 17-21, 2003. Proceedings 23, Springer, 2003, pp. 463–481.