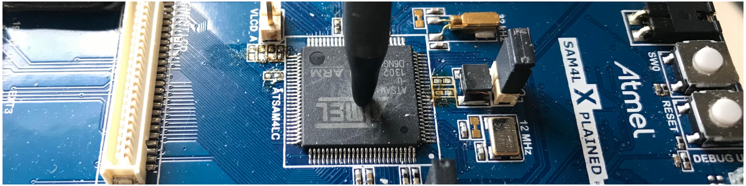


A Decade of Masking Security Proofs

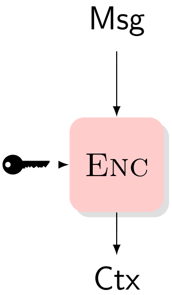
Loïc Masure

PROOFS, 18 September 2025, Kuala Lumpur

Agenda

Content

Context : Side-Channel Analysis (SCA)

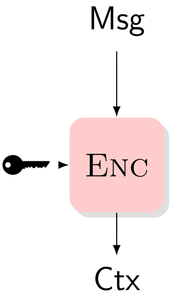


: N bits

Black-box cryptanalysis: 2^N

Context : Side-Channel Analysis (SCA)

"Cryptographic algorithms don't run on paper,

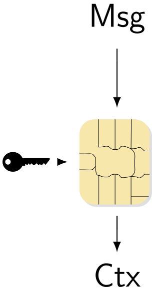


🔑: N bits

Black-box cryptanalysis: 2^N

Context : Side-Channel Analysis (SCA)

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

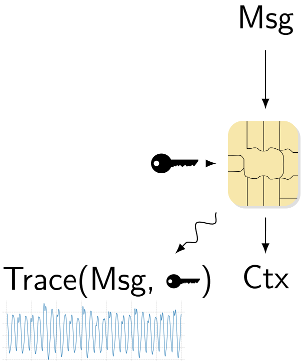


: N bits

Black-box cryptanalysis: 2^N

Context : Side-Channel Analysis (SCA)

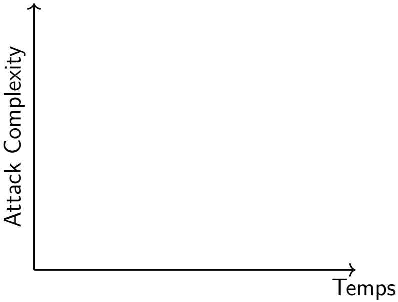
"Cryptographic algorithms don't run on paper, they run on physical devices"



: N bits

Black-box cryptanalysis: 2^N

Evaluate Security against Side-Channel Attacks

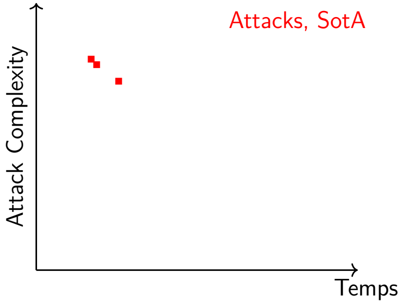


a

Attack approach (industry):

[a] Shamelessly stolen to O. Bronchain

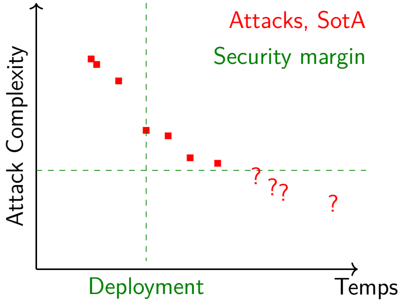
Evaluate Security against Side-Channel Attacks



Attack approach (industry):
Current security level ✓

[a] Shamelessly stolen to O. Bronchain

Evaluate Security against Side-Channel Attacks



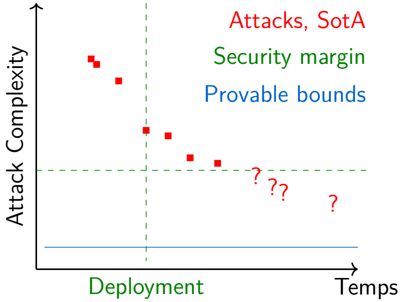
a

Attack approach (industry):

- Current security level ✓
- Future improvement → reevaluation ✗

[a] Shamelessly stolen to O. Bronchain

Evaluate Security against Side-Channel Attacks



a

Attack approach (industry):

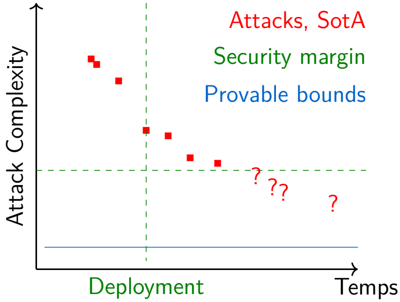
- Current security level ✓
- Future improvement → reevaluation ✗

Approach by *proofs* (academia):

- Rigorous approach ✓
- Potentially conservative ✗

[a] Shamelessly stolen to O. Bronchain

Evaluate Security against Side-Channel Attacks



a

Attack approach (industry):

- Current security level ✓
- Future improvement → reevaluation ✗

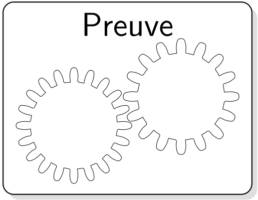
Approach by *proofs* (academia):

- Rigorous approach ✓
- Potentially conservative ✗

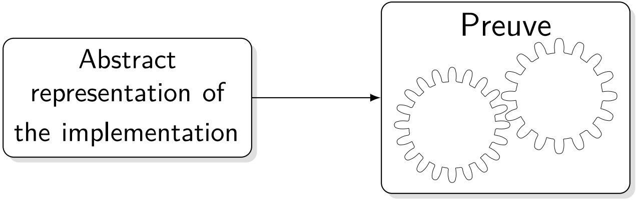
[a] Shamelessly stolen to O. Bronchain

Today's agenda: evaluation by proofs

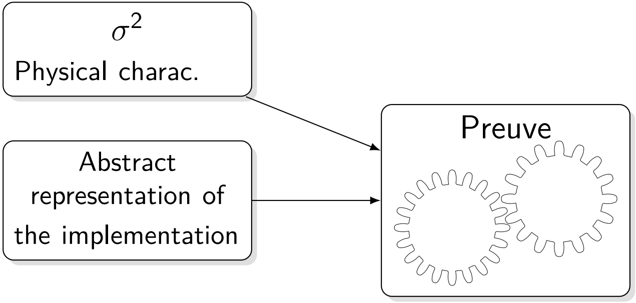
Security Proof



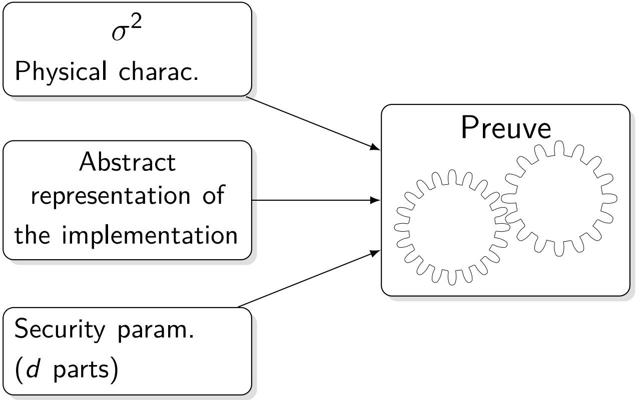
Security Proof



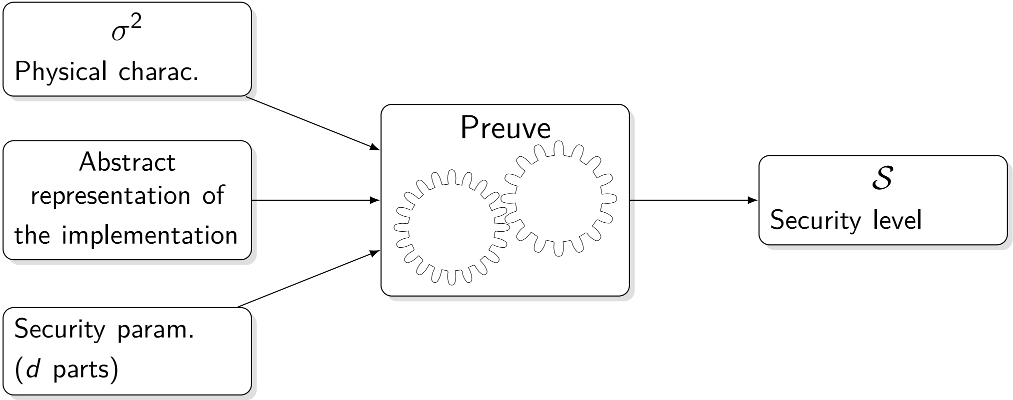
Security Proof



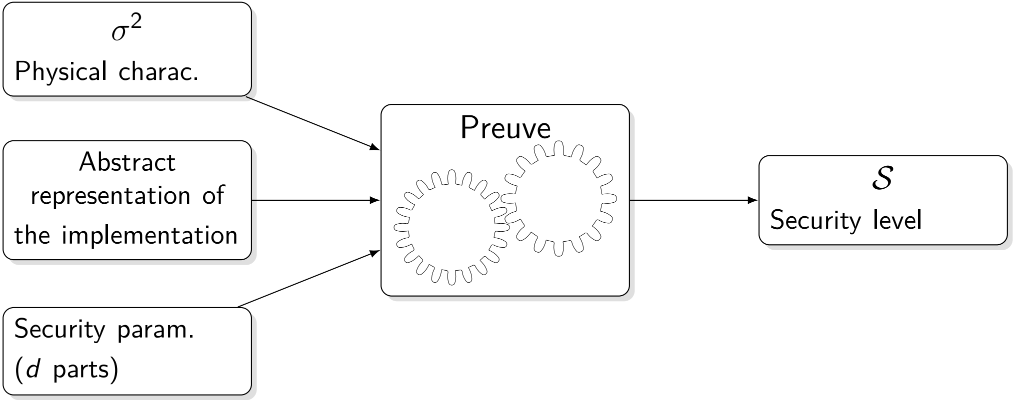
Security Proof



Security Proof



Security Proof

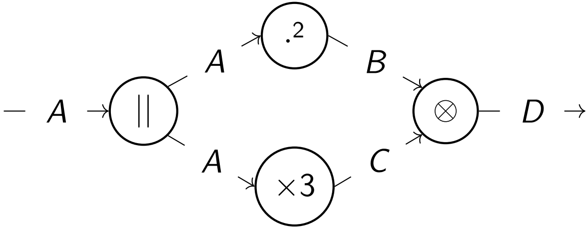


"Any successful attack requires $\mathcal{S}$ observations"

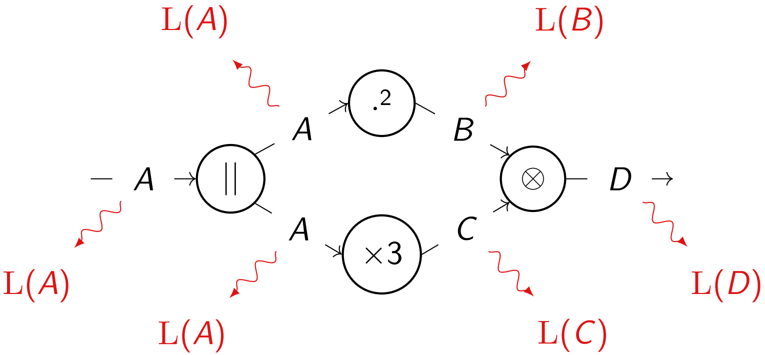
Content

Content

Statement of the Problem

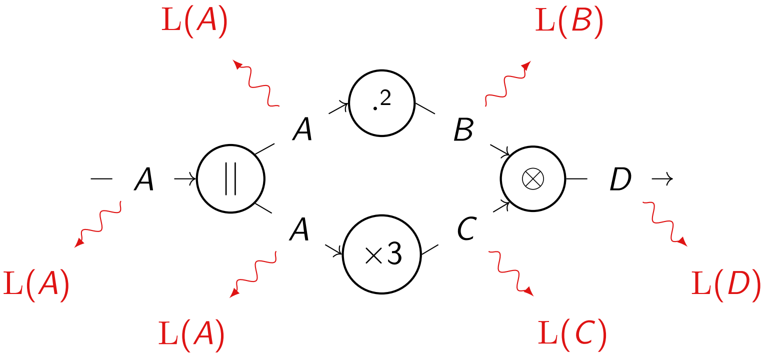


Statement of the Problem



For each wire X, a leakage function $\mathrm{L}(X)$ is revealed to the adversary.

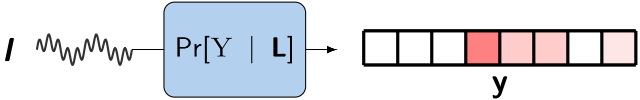
Statement of the Problem



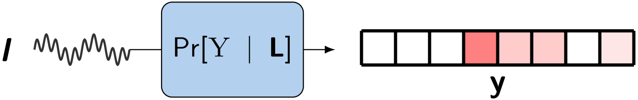
For each wire X, a leakage function $\mathrm{L}(X)$ is revealed to the adversary.

How informative **L** about A?

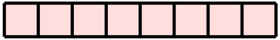
The Noisy Leakage Model



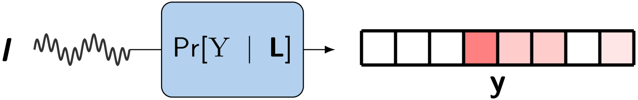
The Noisy Leakage Model



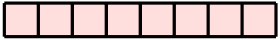
If, the adversary gets:



The Noisy Leakage Model



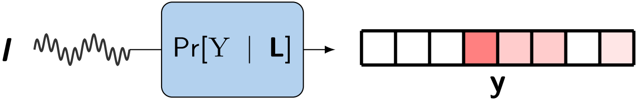
If, the adversary gets:



Very noisy leakage

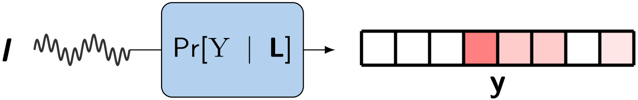
Y indistinguishable from blind guess

The Noisy Leakage Model

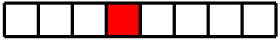


If, the adversary gets:

The Noisy Leakage Model

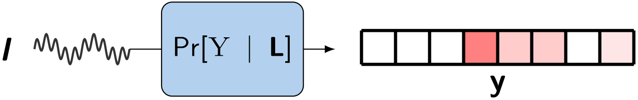


If, the adversary gets:



Low-noise leakage
Exact prediction for Y

The Noisy Leakage Model

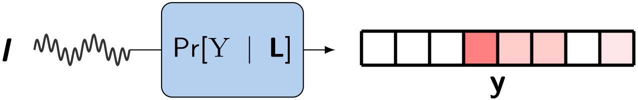


δ-noisy adversary

Any intermediate computation Y leaks $\mathrm{L}(\mathrm{Y})$ such that:

$$\mathsf{SD}(\mathrm{Y};\mathrm{L}) = \mathop{\mathbb{E}}_{\mathrm{L}}\left[\mathsf{TV}\left(\underbrace{\Pr[\mathrm{Y} \mid \mathrm{L}]}_{\Pr[\mathrm{Y} \mid \mathrm{L}]}, \underbrace{\Pr[\mathrm{Y}]}_{\Pr[\mathrm{Y}]}\right)\right] \leq \delta$$

The Noisy Leakage Model



δ-noisy adversary

Any intermediate computation Y leaks $\mathrm{L}(\mathrm{Y})$ such that:

$$\mathsf{SD}\left(\mathrm{Y};\mathrm{L}\right) = \mathop{\mathbb{E}}_{\mathrm{L}}\left[\mathsf{TV}\left(\underbrace{\Pr[\mathrm{Y} \mid \mathrm{L}]}_{\Pr[\mathrm{Y} \mid \mathrm{L}]}, \underbrace{\Pr[\mathrm{Y}]}_{\Pr[\mathrm{Y}]}\right)\right] \leq \delta$$

Main assumption: every observed leakage is δ-noisy

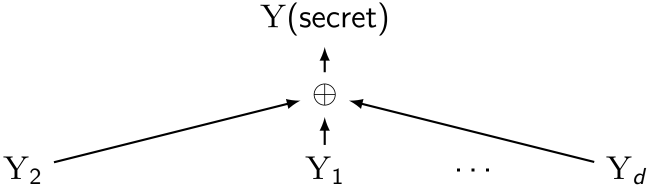
Masking: what is that ?

Masking, a.k.a. *MPC on silicon*:[12] secret sharing over a finite field $(\mathbb{F}, \oplus, \otimes)$

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

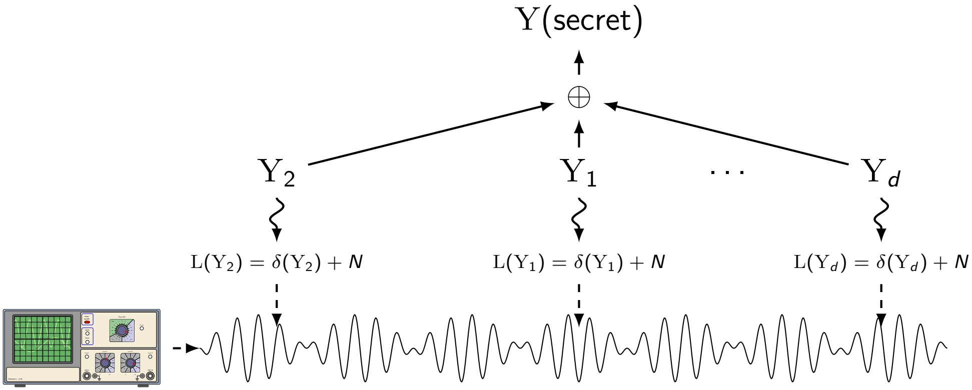
Masking: what is that ?

Masking, a.k.a. *MPC on silicon*:[12] secret sharing over a finite field $(\mathbb{F}, \oplus, \otimes)$



Masking: what is that ?

Masking, a.k.a. *MPC on silicon*:[1,2] secret sharing over a finite field $(\mathbb{F}, \oplus, \otimes)$

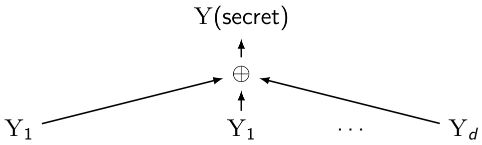


[1] Chari et al., "Towards Sound Approaches to Counteract Power-Analysis Attacks".
[2] Goubin and Patarin, "DES and Differential Power Analysis (The "Duplication" Method)".

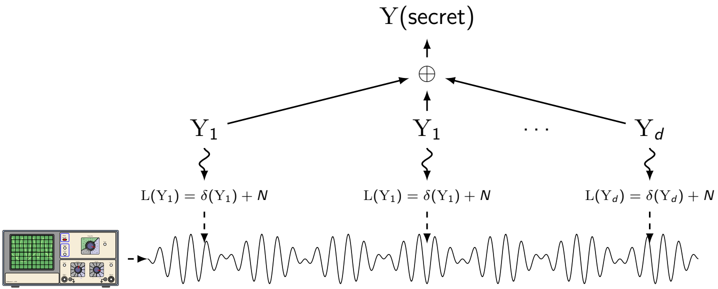
The Effect of Masking

Y(secret)

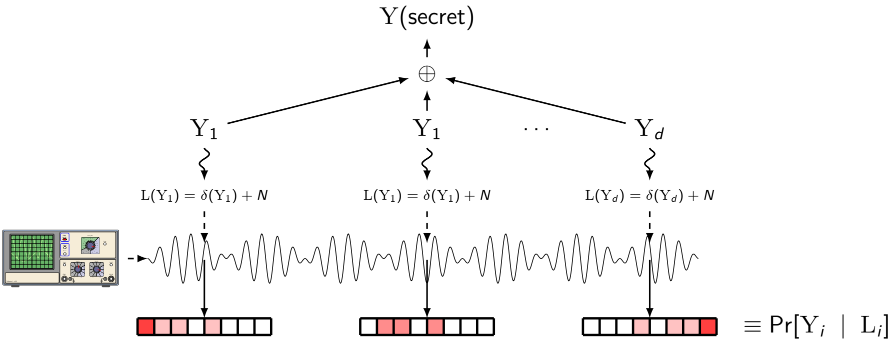
The Effect of Masking



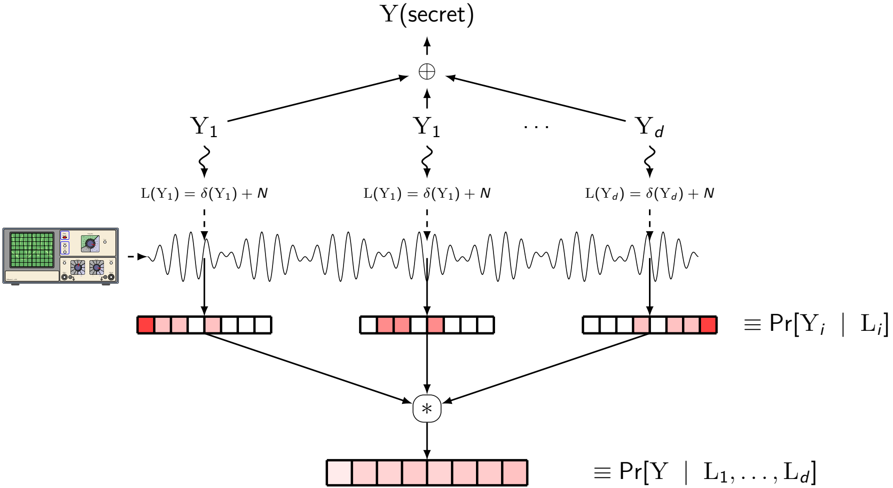
The Effect of Masking



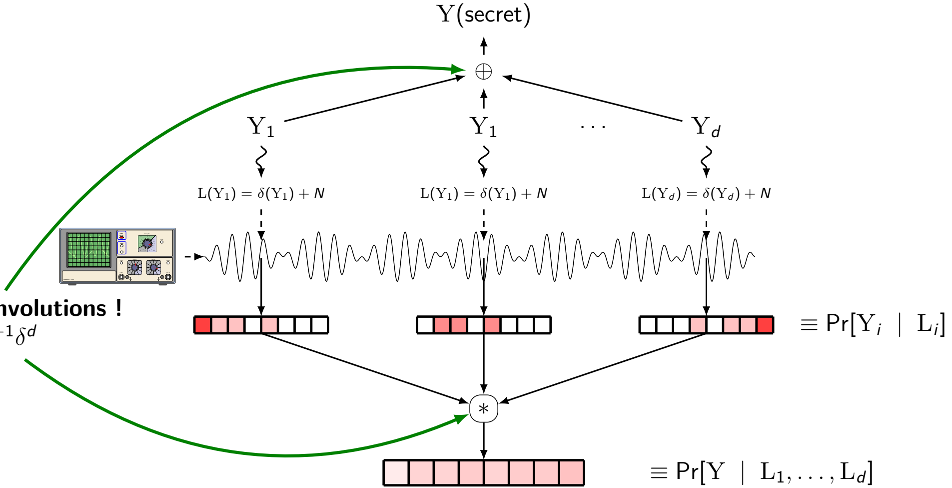
The Effect of Masking



The Effect of Masking



The Effect of Masking



Masking ≡ convolutions !

$\mathrm{SD}(\mathbf{Y}; \mathbf{L}) \leq 2^{d-1}\delta^d$

Content

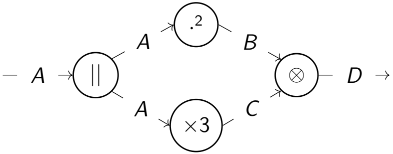
Computing over Masked Secrets

Idea to make a masked circuit

[4] Ishai, Sahai, and Wagner, "Private Circuits: Securing Hardware against Probing Attacks".
[4] Rivain and Prouff, "Provably Secure Higher-Order Masking of AES".

Computing over Masked Secrets

Idea to make a masked circuit

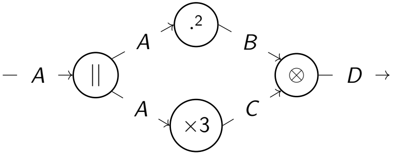


- View your algorithm as a circuit

[4]Ishai, Sahai, and Wagner, "Private Circuits: Securing Hardware against Probing Attacks".
[4]Rivain and Prouff, "Provably Secure Higher-Order Masking of AES".

Computing over Masked Secrets

Idea to make a masked circuit



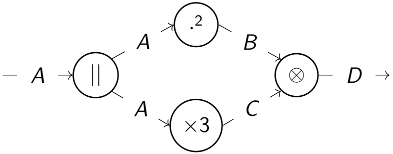
- View your algorithm as a circuit

→ Made of `not`, `and` gates [3]

[4] Ishai, Sahai, and Wagner, "Private Circuits: Securing Hardware against Probing Attacks".
[4] Rivain and Prouff, "Provably Secure Higher-Order Masking of AES".

Computing over Masked Secrets

Idea to make a masked circuit



- View your algorithm as a circuit

$\rightarrow$ Made of `not`, and gates [3]

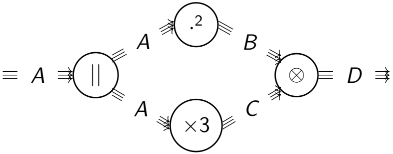
$\rightarrow$ Made of $\oplus, \otimes$ gates [4]

[4] Ishai, Sahai, and Wagner, "Private Circuits: Securing Hardware against Probing Attacks".

[4] Rivain and Prouff, "Provably Secure Higher-Order Masking of AES".

Computing over Masked Secrets

Idea to make a masked circuit

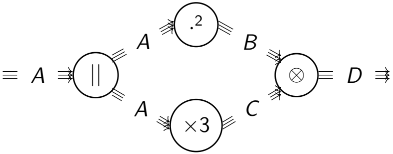


- View your algorithm as a circuit
  - → Made of `not`, `and` gates [3]
  - → Made of $\oplus$, $\otimes$ gates [4]
- Replace each gate by a masked *gadget*

[4] Ishai, Sahai, and Wagner, "Private Circuits: Securing Hardware against Probing Attacks".
[4] Rivain and Prouff, "Provably Secure Higher-Order Masking of AES".

Computing over Masked Secrets

Idea to make a masked circuit



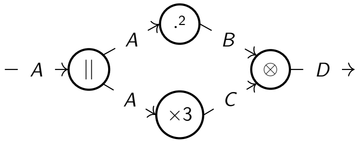
- View your algorithm as a circuit
→ Made of `not`, `and` gates [3]
→ Made of $\oplus$, $\otimes$ gates [4]
- Replace each gate by a masked *gadget*
- Et voilà !**

For now, let's assume the whole circuit to be *probing secure*: every subset of $d-1$ wires is independent from the secret.

[4] Ishai, Sahai, and Wagner, "Private Circuits: Securing Hardware against Probing Attacks".
[4] Rivain and Prouff, "Provably Secure Higher-Order Masking of AES".

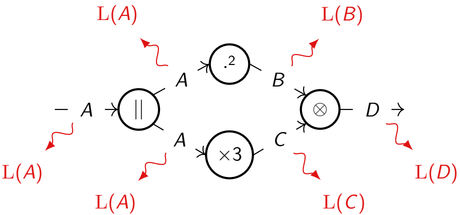
Security Proof for a Gadget

Consider a gadget with ℓ intermediate computations:



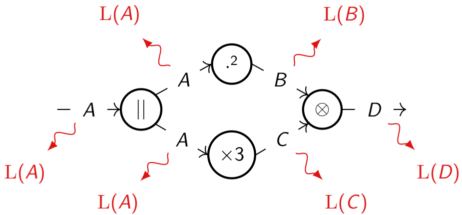
Security Proof for a Gadget

Consider a gadget with ℓ δ-noisy intermediate computations:



Security Proof for a Gadget

Consider a gadget with ℓ δ-noisy intermediate computations:

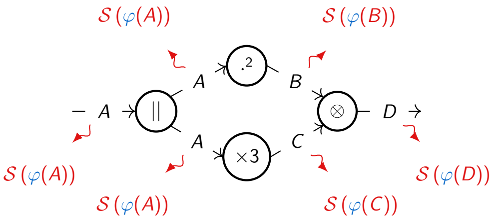


Data-processing Inequality

If for any x the leakage function $\mathrm{L}(x)$

Security Proof for a Gadget

Consider a gadget with ℓ δ-noisy intermediate computations:

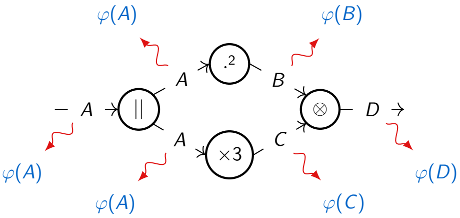


Data-processing Inequality

If for any x the leakage function $\mathrm{L}(x)$ may be expressed as $\mathcal{S}(\varphi(x))$,

Security Proof for a Gadget

Consider a gadget with ℓ δ-noisy intermediate computations:



Data-processing Inequality

If for any x the leakage function $\mathrm{L}(x)$ may be expressed as $\mathcal{S}(\varphi(x))$, then:
advantage from $\mathrm{L}(x) \leq$ advantage from $\varphi(x)$

Reduction from Noisy Leakage to Random Probing

Lemma (Simulatability by Random Probing)

The leakage function L *can be simulated from a random probing adversary:* $\varphi(x)$ *reveals* x *with probability* $\epsilon = 1 - \sum_l \min_x \Pr[\mathrm{L}(x) = l] \leq \delta \cdot |\mathbb{F}|$.[5]

Random probing model: easier to analyze for leakage from computations

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

Security against a Random Probing Adversary

To succeed, at least d out of ℓ wires must be revealed to the adversary:

$$\Pr[\text{Adv. learns sth}] \leq \Pr[\text{At least } d \text{ wires revealed}]$$

[6] Boucheron, Lugosi, and Massart, *Concentration Inequalities: A Nonasymptotic Theory of Independence*, P.24, and Ex. 2.11.

Security against a Random Probing Adversary

To succeed, at least d out of ℓ wires must be revealed to the adversary:

$$\Pr[\text{Adv. learns sth}] \leq \Pr[\text{At least } d \text{ wires revealed}]$$

Theorem (Chernoff Concentration Inequality[6])

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

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

[6] Boucheron, Lugosi, and Massart, *Concentration Inequalities: A Nonasymptotic Theory of Independence*, P.24, and Ex. 2.11.

Putting all Together

In our context, $\ell \leq \mathcal{O}\left(d^2\right)$ (for $\otimes$ gadget), and $\epsilon \leq \delta \cdot |\mathbb{F}|$:

Theorem (Security Bound)

For a single gadget with $\ell \leq \mathcal{O}\left(d^2\right)$ intermediate computations:

$$\mathsf{SD}\left(k; \mathbf{L}\right) \leq \left(\mathcal{O}\left(d\right) \cdot \delta \cdot |\mathbb{F}|\right)^d$$

Putting all Together

In our context, $\ell \leq \mathcal{O}\left(d^2\right)$ (for $\otimes$ gadget), and $\epsilon \leq \delta \cdot |\mathbb{F}|$:

Theorem (Security Bound)

For a single gadget with $\ell \leq \mathcal{O}\left(d^2\right)$ intermediate computations:

$$\mathsf{SD}(k; \mathbf{L}) \leq (\mathcal{O}(d) \cdot \delta \cdot |\mathbb{F}|)^d$$

For the whole circuit $\mathbb{C}$,

$$\mathsf{SD}(k; \mathbf{L}) \leq (|\mathbb{C}| \cdot \mathcal{O}(d) \cdot \delta \cdot |\mathbb{F}|)^d$$

Reduction from Noisy Leakage to Random Probing (I)

Assume there exists such a simulator $\mathcal{S}$,

Reduction from Noisy Leakage to Random Probing (I)

Assume there exists such a simulator $\mathcal{S}$, we need to construct it for all inputs:

$$\begin{aligned} \Pr[\mathcal{S}(x) = l] &= \ldots, \text{ for all } x \\ \Pr[\mathcal{S}(\perp) = l] &= \ldots \end{aligned}$$

Constraints:

Reduction from Noisy Leakage to Random Probing (I)

Assume there exists such a simulator $\mathcal{S}$, we need to construct it for all inputs:

$$\Pr[\mathcal{S}(x) = l] = \ldots, \text{ for all } x$$
$$\Pr[\mathcal{S}(\perp) = l] = \ldots$$

Constraints:

$\rightarrow$ For all input x, $\Pr[\mathcal{S}(x)]$ should be a p.m.f.

Reduction from Noisy Leakage to Random Probing (I)

Assume there exists such a simulator $\mathcal{S}$, we need to construct it for all inputs:

$$\begin{aligned} \Pr[\mathcal{S}(x) = l] &= \ldots, \text{ for all } x \\ \Pr[\mathcal{S}(\perp) = l] &= \ldots \end{aligned}$$

Constraints:

→ For all input x, $\Pr[\mathcal{S}(x)]$ should be a p.m.f.

→ For the input $\perp$, $\Pr[\mathcal{S}(\perp)]$ should be a p.m.f.

Reduction from Noisy Leakage to Random Probing (I)

Assume there exists such a simulator $\mathcal{S}$, we need to construct it for all inputs:

$$\begin{aligned} \Pr[\mathcal{S}(x) = l] &= \ldots, \text{ for all } x \\ \Pr[\mathcal{S}(\perp) = l] &= \ldots \end{aligned}$$

Constraints:

→ For all input x, $\Pr[\mathcal{S}(x)]$ should be a p.m.f.

→ For the input $\perp$, $\Pr[\mathcal{S}(\perp)]$ should be a p.m.f.

→ For any x, l, $\Pr[\mathcal{S}(\varphi(x)) = l] = \Pr[\mathrm{L}(x) = l]$

Reduction from Noisy Leakage to Random Probing (II)

Let us start from the last constraint. For any x and any l:

$$\Pr[\mathrm{L}(x) = l] \quad = \quad \Pr[\mathcal{S}\left(\varphi(x)\right) = l]$$

Reduction from Noisy Leakage to Random Probing (II)

Let us start from the last constraint. For any x and any l:

$$\begin{aligned}\Pr[\mathrm{L}(x) = l] &= \Pr[\mathcal{S}(\varphi(x)) = l] \\ &= \Pr[\varphi(x) = x] \cdot \Pr[\mathcal{S}(x) = l] + \Pr[\varphi(x) = \perp] \cdot \Pr[\mathcal{S}(\perp) = l]\end{aligned}$$

Reduction from Noisy Leakage to Random Probing (II)

Let us start from the last constraint. For any x and any l:

$$\begin{aligned}\Pr[\mathrm{L}(x) = l] &= \Pr[\mathcal{S}\left(\varphi(x)\right) = l] \\ &= \Pr[\varphi(x) = x] \cdot \Pr[\mathcal{S}\left(x\right) = l] + \Pr[\varphi(x) = \bot] \cdot \Pr[\mathcal{S}\left(\bot\right) = l] \\ &= \epsilon \cdot \Pr[\mathcal{S}\left(x\right) = l] + (1 - \epsilon) \cdot \Pr[\mathcal{S}\left(\bot\right) = l]\end{aligned}$$

Reduction from Noisy Leakage to Random Probing (II)

Let us start from the last constraint. For any x and any l:

$$\begin{aligned}\Pr[\mathrm{L}(x) = l] &= \Pr[\mathcal{S}(\varphi(x)) = l] \\ &= \Pr[\varphi(x) = x] \cdot \Pr[\mathcal{S}(x) = l] + \Pr[\varphi(x) = \bot] \cdot \Pr[\mathcal{S}(\bot) = l] \\ &= \epsilon \cdot \Pr[\mathcal{S}(x) = l] + (1 - \epsilon) \cdot \Pr[\mathcal{S}(\bot) = l]\end{aligned}$$

Hence,

$$0 \leq \Pr[\mathcal{S}(\bot) = l] = \frac{\overbrace{\Pr[\mathrm{L}(x) = l] - \epsilon \cdot \Pr[\mathcal{S}(x) = l]}^{\text{Should not depend on } x}}{1 - \epsilon}$$

Reduction from Noisy Leakage to Random Probing (II)

Let us start from the last constraint. For any x and any l:

$$\begin{aligned}\Pr[\mathrm{L}(x) = l] &= \Pr[\mathcal{S}\left(\varphi(x)\right) = l] \\ &= \Pr[\varphi(x) = x] \cdot \Pr[\mathcal{S}\left(x\right) = l] + \Pr[\varphi(x) = \bot] \cdot \Pr[\mathcal{S}\left(\bot\right) = l] \\ &= \epsilon \cdot \Pr[\mathcal{S}\left(x\right) = l] + (1 - \epsilon) \cdot \Pr[\mathcal{S}\left(\bot\right) = l]\end{aligned}$$

Hence,

$$0 \leq \Pr[\mathcal{S}\left(\bot\right) = l] = \frac{\overbrace{\Pr[\mathrm{L}(x) = l] - \epsilon \cdot \Pr[\mathcal{S}\left(x\right) = l]}^{\text{Should not depend on } x}}{1 - \epsilon} = \frac{\pi(l)}{1 - \epsilon} \quad (1)$$

Reduction from Noisy Leakage to Random Probing (II)

Let us start from the last constraint. For any x and any l:

$$\begin{aligned}
\Pr[\mathrm{L}(x) = l] &= \Pr[\mathcal{S}\left(\varphi(x)\right) = l] \\
&= \Pr[\varphi(x) = x] \cdot \Pr[\mathcal{S}\left(x\right) = l] + \Pr[\varphi(x) = \perp] \cdot \Pr[\mathcal{S}\left(\perp\right) = l] \\
&= \epsilon \cdot \Pr[\mathcal{S}\left(x\right) = l] + (1 - \epsilon) \cdot \Pr[\mathcal{S}\left(\perp\right) = l]
\end{aligned}$$

Hence,

$$0 \leq \Pr[\mathcal{S}\left(\perp\right) = l] = \frac{\overbrace{\Pr[\mathrm{L}(x) = l] - \epsilon \cdot \Pr[\mathcal{S}\left(x\right) = l]}^{\text{Should not depend on } X}}{1 - \epsilon} = \frac{\pi(l)}{1 - \epsilon} \quad (1)$$

$$0 \leq \Pr[\mathcal{S}\left(x\right) = l] = \frac{\Pr[\mathrm{L}(x) = l] - \pi(l)}{\epsilon} \quad (2)$$

Reduction from Noisy Leakage to Random Probing (II)

Let us start from the last constraint. For any x and any l:

$$\begin{aligned} \Pr[\mathrm{L}(x) = l] &= \Pr[\mathcal{S}(\varphi(x)) = l] \\ &= \Pr[\varphi(x) = x] \cdot \Pr[\mathcal{S}(x) = l] + \Pr[\varphi(x) = \perp] \cdot \Pr[\mathcal{S}(\perp) = l] \\ &= \epsilon \cdot \Pr[\mathcal{S}(x) = l] + (1 - \epsilon) \cdot \Pr[\mathcal{S}(\perp) = l] \end{aligned}$$

Hence,

$$0 \leq \Pr[\mathcal{S}(\perp) = l] = \frac{\overbrace{\Pr[\mathrm{L}(x) = l] - \epsilon \cdot \Pr[\mathcal{S}(x) = l]}^{\text{Should not depend on } X}}{1 - \epsilon} = \frac{\pi(l)}{1 - \epsilon} \qquad (1)$$

$$0 \leq \Pr[\mathcal{S}(x) = l] = \frac{\Pr[\mathrm{L}(x) = l] - \pi(l)}{\epsilon} \qquad (2)$$

Is there any ϵ such that $\geq$ and $\geq$ are valid?

Reduction from Noisy Leakage to RP (III)

Is there any ϵ such that $\geq$ and $\geq$ are valid?

Reduction from Noisy Leakage to RP (III)

Is there any ϵ such that $\geq$ and $\geq$ are valid? From (6), and (7), we get

$$0 \leq \pi(l) \leq \Pr[\mathrm{L}(x) = l] \text{ for any } x$$

Reduction from Noisy Leakage to RP (III)

Is there any ϵ such that $\geq$ and $\geq$ are valid? From (6), and (7), we get

$$0 \leq \pi(l) \leq \Pr[\mathrm{L}(x) = l] \text{ for any } x$$

In other words,

$$0 \leq \pi(l) \leq \min_{x} \Pr[\mathrm{L}(x) = l]$$

Reduction from Noisy Leakage to RP (III)

Is there any ϵ such that $\geq$ and $\geq$ are valid? From (6), and (7), we get

$$0 \leq \pi(l) \leq \Pr[\mathrm{L}(x) = l] \text{ for any } x$$

In other words,

$$0 \leq \pi(l) \leq \min_x \Pr[\mathrm{L}(x) = l]$$

Furthermore, summing (6) over l, by definition of probability distributions,

$$\sum_l \pi(l) = \underbrace{\sum_l \Pr[\mathrm{L}(x) = l]}_{=1} - \epsilon \cdot \underbrace{\sum_l \Pr[\mathcal{S}(x) = l]}_{=1}$$

Reduction from Noisy Leakage to RP (III)

Is there any ϵ such that $\geq$ and $\geq$ are valid? From (6), and (7), we get

$$0 \leq \pi(l) \leq \Pr[\mathrm{L}(x) = l] \text{ for any } x$$

In other words,

$$0 \leq \pi(l) \leq \min_x \Pr[\mathrm{L}(x) = l]$$

Furthermore, summing (6) over l, by definition of probability distributions,

$$\sum_l \pi(l) = \underbrace{\sum_l \Pr[\mathrm{L}(x) = l]}_{=1} - \epsilon \cdot \underbrace{\sum_l \Pr[\mathcal{S}(x) = l]}_{=1} = 1 - \epsilon$$

Reduction from Noisy Leakage to RP (III)

Is there any ϵ such that $\geq$ and $\geq$ are valid? From (6), and (7), we get

$$0 \leq \pi(l) \leq \Pr[\mathrm{L}(x) = l] \text{ for any } x$$

In other words,

$$0 \leq \pi(l) \leq \min_x \Pr[\mathrm{L}(x) = l]$$

Furthermore, summing (6) over l, by definition of probability distributions,

$$\sum_l \pi(l) = \underbrace{\sum_l \Pr[\mathrm{L}(x) = l]}_{=1} - \epsilon \cdot \underbrace{\sum_l \Pr[\mathcal{S}(x) = l]}_{=1} = 1 - \epsilon$$

Hence,

$$\epsilon = 1 - \sum_l \pi(l) \geq 1 - \sum_l \min_x \Pr[\mathrm{L}(x) = l]$$

Reduction from Noisy Leakage to RP (III)

Is there any ϵ such that $\geq$ and $\geq$ are valid? From (6), and (7), we get

$$0 \leq \pi(l) \leq \Pr[\mathrm{L}(x) = l] \text{ for any } x$$

In other words,

$$0 \leq \pi(l) \leq \min_x \Pr[\mathrm{L}(x) = l]$$

Furthermore, summing (6) over l, by definition of probability distributions,

$$\sum_l \pi(l) = \underbrace{\sum_l \Pr[\mathrm{L}(x) = l]}_{=1} - \epsilon \cdot \underbrace{\sum_l \Pr[\mathcal{S}(x) = l]}_{=1} = 1 - \epsilon$$

Hence, to have the smallest ϵ,

$$\epsilon = 1 - \sum_l \pi(l) = 1 - \sum_l \min_x \Pr[\mathrm{L}(x) = l]$$

Main Challenge

For the whole circuit $\mathbb{C}$,

$$\mathsf{SD}\left(k; \mathbf{L}\right) \leq \left(|\mathsf{C}| \cdot \mathcal{O}\left(d\right) \cdot |\mathbb{F}| \cdot \delta\right)^{d}$$

Main challenge: get rid of the three factors d, $|\mathsf{C}|$, and $|\mathbb{F}|$

Main Challenge

For the whole circuit $\mathbb{C}$,

$$\mathsf{SD}(k; \mathbf{L}) \leq \left(|\mathsf{C}| \cdot \mathcal{O}(d) \cdot |\mathbb{F}| \cdot \delta\right)^{d}$$

Main challenge: get rid of the three factors d, $|\mathsf{C}|$, and $|\mathbb{F}|$

d: Abdel's thesis

Main Challenge

For the whole circuit $\mathbb{C}$,

$$\mathsf{SD}(k; \mathbf{L}) \leq \left(|\mathsf{C}| \cdot \mathcal{O}(d) \cdot |\mathbb{F}| \cdot \delta\right)^d$$

Main challenge: get rid of the three factors d, $|\mathsf{C}|$, and $|\mathbb{F}|$

- d: Abdel's thesis
- $|\mathsf{C}|$: this talk (a bit)

Main Challenge

For the whole circuit $\mathbb{C}$,

$$\mathsf{SD}(k; \mathbf{L}) \leq \left(|\mathsf{C}| \cdot \mathcal{O}(d) \cdot |\mathbb{F}| \cdot \delta\right)^{d}$$

Main challenge: get rid of the three factors d, $|\mathsf{C}|$, and $|\mathbb{F}|$

- d: Abdel's thesis
- $|\mathsf{C}|$: this talk (a bit)
- $|\mathbb{F}|$: Monday's talk (a bit) at CHES, and this talk (a bit)

Main Challenge

For the whole circuit $\mathbb{C}$,

$$\mathsf{SD}\left(k; \mathbf{L}\right) \leq \left(|\mathsf{C}| \cdot \mathcal{O}\left(d\right) \cdot |\mathbb{F}| \cdot \delta\right)^{d}$$

Main challenge: get rid of the three factors d, $|\mathsf{C}|$, and $|\mathbb{F}|$

d: Abdel's thesis

$|\mathsf{C}|$: this talk (a bit)

$|\mathbb{F}|$: Monday's talk (a bit) at CHES, and this talk (a bit)

A few numbers:

$$d(2, 3, 4, ..., 16) \ll |\mathsf{C}|(\approx 10^3, 10^5), |\mathbb{F}|(256, 2^{23}, 2^{50})$$

Bad News

Without further assumption on the circuit, the previous bound is *tight*:

d: horizontal attacks ✗

Bad News

Without further assumption on the circuit, the previous bound is *tight*:

- d: horizontal attacks ✗
- $|C|$: horizontal attacks ✗

Bad News

Without further assumption on the circuit, the previous bound is *tight*:

- d: horizontal attacks ✗
- $|\mathcal{C}|$: horizontal attacks ✗
- $|\mathbb{F}|$: non-uniform wires ✗

Why $|\mathbb{F}|$ is Tight

Consider the following leakage model (with $\delta \approx \frac{2}{|\mathbb{F}|} \cdot \alpha$):

$$L(x) = \begin{cases} x, & \text{with probability } \alpha, \text{ if } x = 0, \\ \bot, & \text{otherwise} \end{cases} . \tag{3}$$

Leakage from a uniform encoding: $\epsilon \leq \frac{|\mathbb{F}|}{2} \cdot (2\delta)^d$ ✓

Why $|\mathbb{F}|$ is Tight

Consider the following leakage model (with $\delta \approx \frac{2}{|\mathbb{F}|} \cdot \alpha$):

$$L(x) = \begin{cases} x, & \text{with probability } \alpha, \text{ if } x = 0, \\ \perp, & \text{otherwise} \end{cases} . \tag{3}$$

Leakage from a uniform encoding: $\epsilon \leq \frac{|\mathbb{F}|}{2} \cdot (2\delta)^d$ ✓
Leakage from computation ?

$\rightarrow$ Share-wise computation of LSB to get LSB of secret (in binary field)

Why $|\mathbb{F}|$ is Tight

Consider the following leakage model (with $\delta \approx \frac{2}{|\mathbb{F}|} \cdot \alpha$):

$$L(x) = \begin{cases} x, & \text{with probability } \alpha, \text{ if } x = 0, \\ \perp, & \text{otherwise} \end{cases} . \qquad (3)$$

Leakage from a uniform encoding: $\epsilon \leq \frac{|\mathbb{F}|}{2} \cdot (2\delta)^d$ ✓
Leakage from computation ?

→ Share-wise computation of LSB to get LSB of secret (in binary field)

→ Each output share uniform over $\{0, 1\}$ instead of $\mathbb{F}$

Why $|\mathbb{F}|$ is Tight

Consider the following leakage model (with $\delta \approx \frac{2}{|\mathbb{F}|} \cdot \alpha$):

$$L(x) = \begin{cases} x, & \text{with probability } \alpha, \text{ if } x = 0, \\ \perp, & \text{otherwise} \end{cases} . \tag{3}$$

Leakage from a uniform encoding: $\epsilon \leq \frac{|\mathbb{F}|}{2} \cdot (2\delta)^d$ ✓
Leakage from computation ?

→ Share-wise computation of LSB to get LSB of secret (in binary field)

→ Each output share uniform over $\{0, 1\}$ instead of $\mathbb{F}$

→ Successful recovery of all shares with probability at least $\alpha^d \approx \left(\frac{\delta|\mathbb{F}|}{2}\right)^d$ ✗

Content

Setting

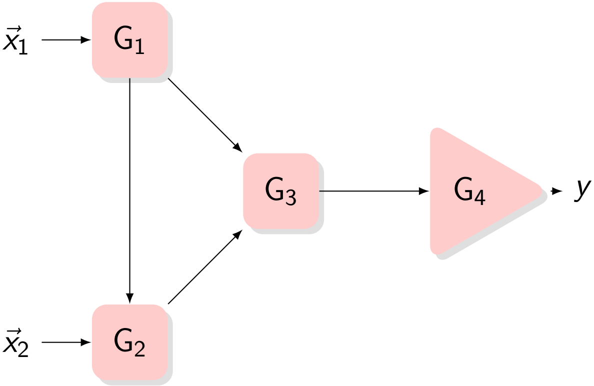
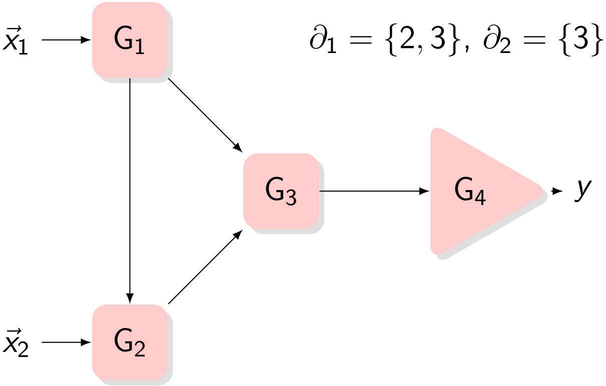


Figure: G_1: SNI copy gadget, G_2, G_3: SNI gadgets, G_4: NIo gadget.

Setting



$\partial_1 = \{2, 3\}$, $\partial_2 = \{3\}$, $\partial_3 = \{4\}$, $\partial_4 = \emptyset$

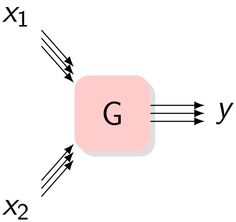
Figure: G_1: SNI copy gadget, G_2, G_3: SNI gadgets, G_4: NIo gadget.

∂_i: set of all subsequent gadgets linked to G_i

Strong Non-Interference[8]

Definition (t-Strong Non-Interference)

A gadget G is t-SNI



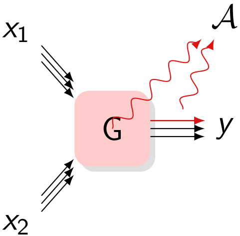
[7] Must be connected to different gadgets ✓

[8] Barthe et al., "Strong Non-Interference and Type-Directed Higher-Order Masking".

Strong Non-Interference[8]

Definition (t-Strong Non-Interference)

A gadget G is t-SNI if any set W^{G} of internal probes and any set J^{G} of output probes such that $\left|W^{\mathsf{G}}\right| + \left|J^{\mathsf{G}}\right| \leq t$

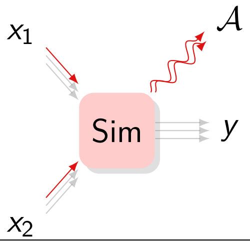


[7] Must be connected to different gadgets ✓
[8] Barthe et al., "Strong Non-Interference and Type-Directed Higher-Order Masking".

Strong Non-Interference[8]

Definition (t-Strong Non-Interference)

A gadget G is t-SNI if any set W^{G} of internal probes and any set J^{G} of output probes such that $\left|W^{\mathsf{G}}\right| + \left|J^{\mathsf{G}}\right| \leq t$ can be simulated with at most $\left|I^{\mathsf{G}}\right| \leq \left|W^{\mathsf{G}}\right|$ shares of each input sharing



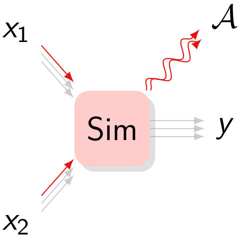
[7] Must be connected to different gadgets ✓

[8] Barthe et al., "Strong Non-Interference and Type-Directed Higher-Order Masking".

Strong Non-Interference[8]

Definition (t-Strong Non-Interference)

A gadget G is t-SNI if any set W^{G} of internal probes and any set J^{G} of output probes such that $\left|W^{\mathsf{G}}\right| + \left|J^{\mathsf{G}}\right| \leq t$ can be simulated with at most $\left|I^{\mathsf{G}}\right| \leq \left|W^{\mathsf{G}}\right|$ shares of each input sharing



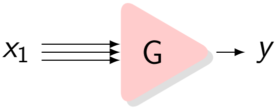
→ Composable : circ. SNI iff all gadgets SNI
→ SNI $\implies$ probing security
→ Extends to multiple outputs[7]

[7] Must be connected to different gadgets ✓
[8] Barthe et al., “Strong Non-Interference and Type-Directed Higher-Order Masking”.

Non-Interference with Public Outputs[10]

Definition (t-NIo)

A gadget is t-NIo



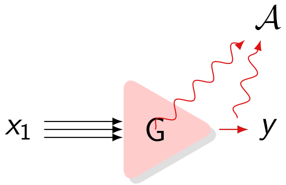
[9] Coron et al., *High-order Polynomial Comparison and Masking Lattice-based Encryption*

[10] Barthe et al., "Masking the GLP Lattice-Based Signature Scheme at Any Order".

Non-Interference with Public Outputs[10]

Definition (t-NIo)

A gadget is t-NIo if any set of $t_1 \leq t$ internal probes and the output

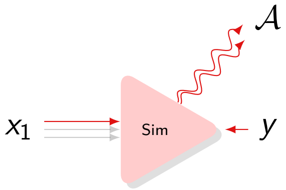


[9] Coron et al., *High-order Polynomial Comparison and Masking Lattice-based Encryption*
[10] Barthe et al., "Masking the GLP Lattice-Based Signature Scheme at Any Order".

Non-Interference with Public Outputs[10]

Definition (t-NIo)

A gadget is t-NIo if any set of $t_1 \leq t$ internal probes and the output can be jointly simulated from the output and at most t_1 input shares

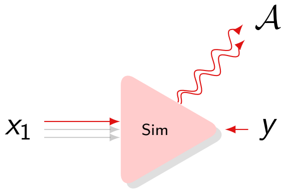


[9] Coron et al., *High-order Polynomial Comparison and Masking Lattice-based Encryption*
[10] Barthe et al., "Masking the GLP Lattice-Based Signature Scheme at Any Order".

Non-Interference with Public Outputs[10]

DEFINITION (t-NIO)

A gadget is t-NIo if any set of $t_1 \leq t$ internal probes and the output can be jointly simulated from the output and at most t_1 input shares



→ Output assumed to be public anyway
→ Built from strong Refreshing [9]

[9] Coron et al., *High-order Polynomial Comparison and Masking Lattice-based Encryption*
[10] Barthe et al., "Masking the GLP Lattice-Based Signature Scheme at Any Order".

Composition Theorem

Theorem

Assume: (1) Each output gadget $(d-1)$*-NIo;*

Composition Theorem

Theorem

Assume: (1) Each output gadget $(d-1)$-NIo; (2) Each internal gadget t_i-SNI;

Composition Theorem

Theorem

Assume: (1) Each output gadget $(d-1)$-NIo; (2) Each internal gadget t_i-SNI; (3) Each copy gadget connected to different gadgets;

Composition Theorem

Theorem

Assume: (1) Each output gadget $(d-1)$-NIo; (2) Each internal gadget t_i-SNI; (3) Each copy gadget connected to different gadgets; then, C is secure with proba $\geq 1 - \eta$ such that:

$$\eta \leq \sum_{\substack{i=1 \\ G_i \text{ not output}}}^{|C|} \left(e \cdot \frac{|G_i| + \sum_{j \in \partial_i} |G_j|}{t_i + 1} \cdot \epsilon \right)^{t_i+1} .$$

Composition Theorem

Theorem

Assume: (1) Each output gadget $(d-1)$-NIo; (2) Each internal gadget t_i-SNI; (3) Each copy gadget connected to different gadgets; then, C is secure with proba $\geq 1-\eta$ such that:

$$\eta \leq \sum_{\substack{i=1 \\ G_i \text{ not output}}}^{|C|} \left(e \cdot \frac{|G_i| + \sum_{j \in \partial_i} |G_j|}{t_i + 1} \cdot \epsilon \right)^{t_i+1} .$$

Corollary

The d-share ISW compiler is $|C| \cdot \left(\mathcal{O}(d) \cdot |\mathbb{F}| \cdot \delta\right)^d$-noisy leakage secure

Proof Sketch

Apply SNI simulator gadget-wise, in reversed order, until complete or failure

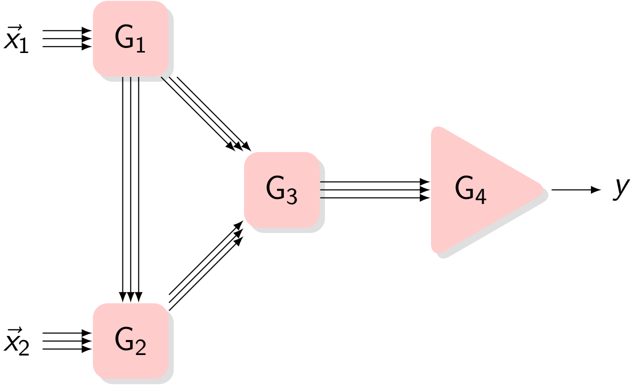


Figure: G_1: SNI copy gadget, G_2, G_3: SNI gadgets, G_4: NIo gadget. $\partial_1 = \{2, 3\}$, $\partial_2 = \{3\}$, $\partial_3 = \{4\}$, $\partial_4 = \emptyset$

Proof Sketch

Apply SNI simulator gadget-wise, in reversed order, until complete or failure

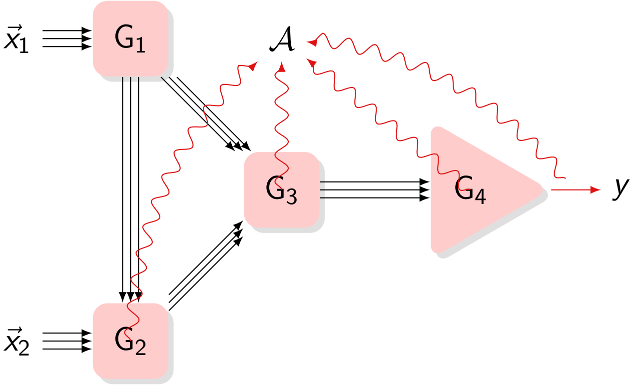


Figure: G_1: SNI copy gadget, G_2, G_3: SNI gadgets, G_4: NIo gadget. $\partial_1 = \{2, 3\}$, $\partial_2 = \{3\}$, $\partial_3 = \{4\}$, $\partial_4 = \emptyset$

Proof Sketch

Apply SNI simulator gadget-wise, in reversed order, until complete or failure

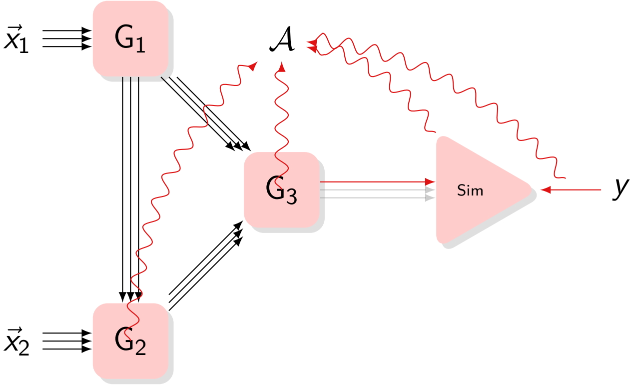


Figure: G_1: SNI copy gadget, G_2, G_3: SNI gadgets, G_4: NIo gadget. $\partial_1 = \{2, 3\}$, $\partial_2 = \{3\}$, $\partial_3 = \{4\}$, $\partial_4 = \emptyset$

Proof Sketch

Apply SNI simulator gadget-wise, in reversed order, until complete or failure

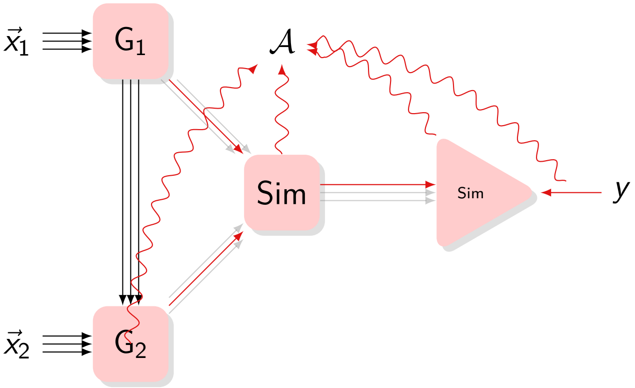


Figure: G_1: SNI copy gadget, G_2, G_3: SNI gadgets, G_4: NIo gadget. $\partial_1 = \{2, 3\}$, $\partial_2 = \{3\}$, $\partial_3 = \{4\}$, $\partial_4 = \emptyset$

Proof Sketch

Apply SNI simulator gadget-wise, in reversed order, until complete or failure

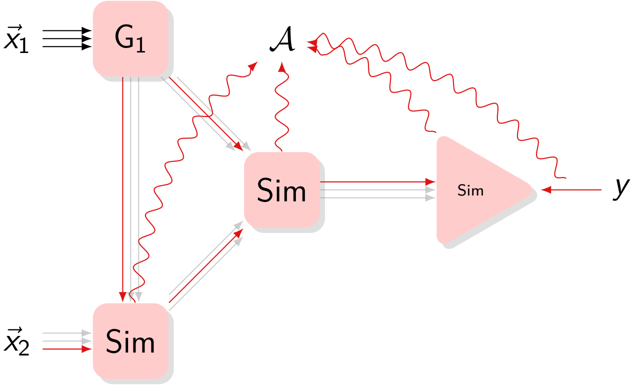


Figure: G_1: SNI copy gadget, G_2, G_3: SNI gadgets, G_4: NIo gadget. $\partial_1 = \{2, 3\}$, $\partial_2 = \{3\}$, $\partial_3 = \{4\}$, $\partial_4 = \emptyset$

Proof Sketch

Apply SNI simulator gadget-wise, in reversed order, until complete or failure

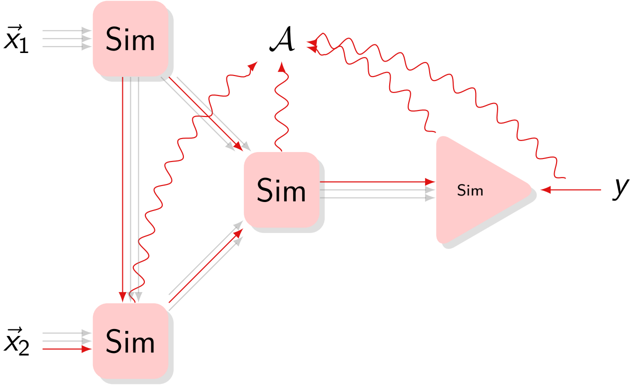


Figure: G_1: SNI copy gadget, G_2, G_3: SNI gadgets, G_4: NIo gadget. $\partial_1 = \{2, 3\}$, $\partial_2 = \{3\}$, $\partial_3 = \{4\}$, $\partial_4 = \emptyset$

Proof Sketch

Failure may happen (simulation with abort)

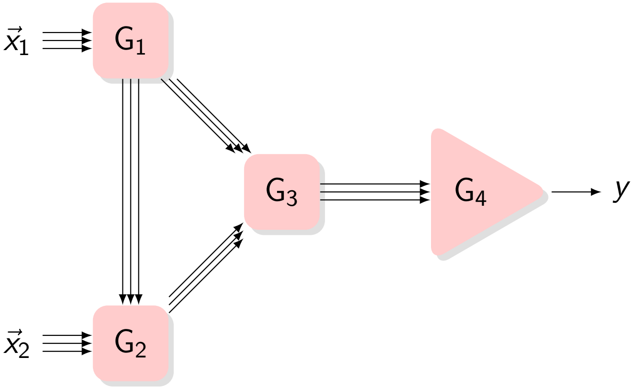


Figure: G_1: SNI copy gadget, G_2, G_3: SNI gadgets, G_4: NIo gadget. $\partial_1 = \{2, 3\}$, $\partial_2 = \{3\}$, $\partial_3 = \{4\}$, $\partial_4 = \emptyset$

Proof Sketch

Failure may happen (simulation with abort)

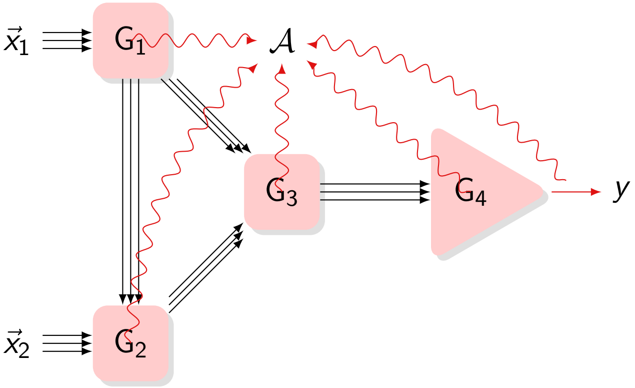


Figure: G_1: SNI copy gadget, G_2, G_3: SNI gadgets, G_4: NIo gadget. $\partial_1 = \{2, 3\}$, $\partial_2 = \{3\}$, $\partial_3 = \{4\}$, $\partial_4 = \emptyset$

Proof Sketch

Failure may happen (simulation with abort)

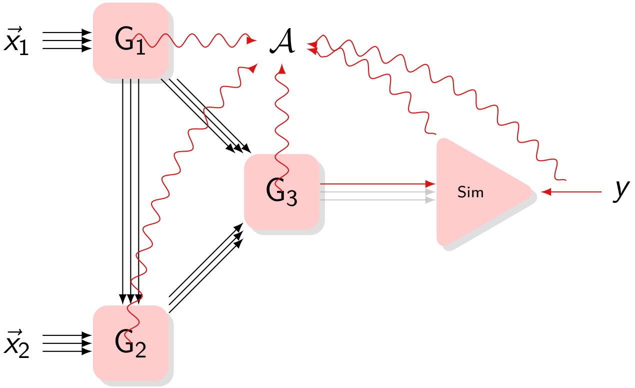


Figure: G_1: SNI copy gadget, G_2, G_3: SNI gadgets, G_4: NIo gadget. $\partial_1 = \{2, 3\}$, $\partial_2 = \{3\}$, $\partial_3 = \{4\}$, $\partial_4 = \emptyset$

Proof Sketch

Failure may happen (simulation with abort)

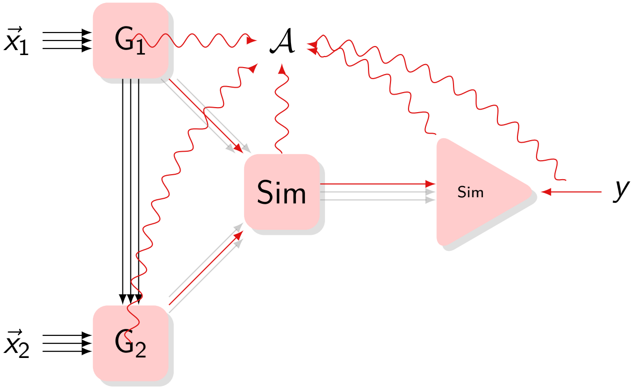


Figure: G_1: SNI copy gadget, G_2, G_3: SNI gadgets, G_4: NIo gadget. $\partial_1 = \{2, 3\}$, $\partial_2 = \{3\}$, $\partial_3 = \{4\}$, $\partial_4 = \emptyset$

Proof Sketch

Failure may happen (simulation with abort)

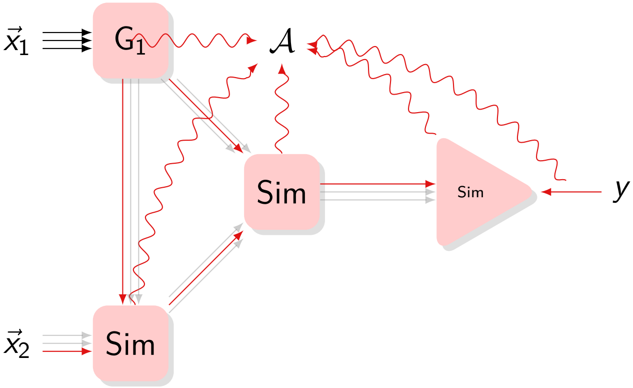


Figure: G_1: SNI copy gadget, G_2, G_3: SNI gadgets, G_4: NIo gadget. $\partial_1 = \{2, 3\}$, $\partial_2 = \{3\}$, $\partial_3 = \{4\}$, $\partial_4 = \emptyset$

Proof Sketch

Failure may happen (simulation with abort)

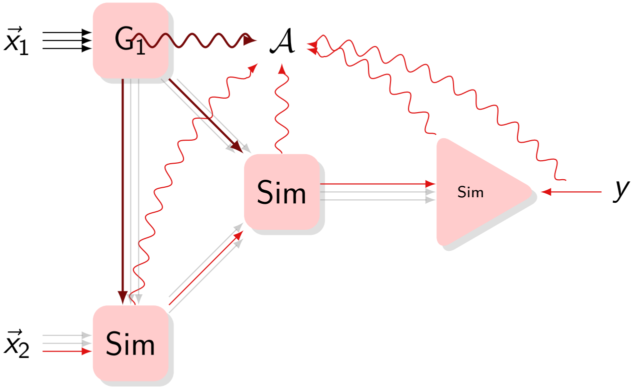


Figure: G_1: SNI copy gadget, G_2, G_3: SNI gadgets, G_4: NIo gadget. $\partial_1 = \{2, 3\}$, $\partial_2 = \{3\}$, $\partial_3 = \{4\}$, $\partial_4 = \emptyset$

How Often Does It Fail?

Let bad_i: "simulation failure at step i". This implies:

[11] If G_j is an NIo output gadget, this is also verified.

How Often Does It Fail?

Let bad_i: "simulation failure at step i". This implies:

$\rightarrow$ t_i-SNI assumption of G_i not verified: $\left|W^{G_i}\right| + \sum_{j \in \partial_i} \left|J_j^{G_i}\right| \geq t_i$

[11] If G_j is an NIo output gadget, this is also verified.

How Often Does It Fail?

Let bad_i: “simulation failure at step i”. This implies:

$\rightarrow$ t_i-SNI assumption of G_i not verified: $\left|W^{G_i}\right| + \sum_{j \in \partial_i} \left|J_j^{G_i}\right| \geq t_i$

$\rightarrow$ $\forall j > i$, t_j-SNI assumption of G_j verified, thereby $\left|J_j^{G_i}\right| = \left|I_i^{G_j}\right| \leq \left|W^{G_j}\right|$[11]

[11] If G_j is an NIo output gadget, this is also verified.

How Often Does It Fail?

Let bad_i: "simulation failure at step i". This implies:

→ t_i-SNI assumption of G_i not verified: $\left|W^{G_i}\right| + \sum_{j \in \partial_i} \left|J_j^{G_i}\right| \geq t_i$

→ $\forall j > i$, t_j-SNI assumption of G_j verified, thereby $\left|J_j^{G_i}\right| = \left|I_i^{G_j}\right| \leq \left|W^{G_j}\right|$[11]

Hence,

$$\Pr[\mathsf{bad}_i] \leq \Pr\left[\left|W^{G_i}\right| + \sum_{j \in \partial_i} \left|W^{G_j}\right| \geq t_i\right]$$

Using the union bound:

$$\eta = \sum_{\substack{i=1 \\ G_i \text{not output}}}^{|\mathsf{C}|} \Pr[\mathsf{bad}_i]$$

[11] If G_j is an NIo output gadget, this is also verified.

Concluding the Proof

Using Chernoff:

$$\Pr\left[\left|W^{G_i}\right| + \sum_{j\in\partial_i}\left|W^{G_j}\right| > t_i\right] = \Pr\left[\left|W^{G_i} \cup \left(\bigcup_{j\in\partial_i} W^{G_j}\right)\right| \geq t_i + 1\right]$$

$$\leq \left(e \cdot \frac{|G_i| + \sum_{j\in\partial_i}|G_j|}{t_i + 1} \cdot \epsilon\right)^{t_i+1} .$$

Comparison with Previous Works

So far, trade-off was needed:

→ Duc *et al.*:[12] $|C| \cdot \left(\mathcal{O}(d) \cdot |\mathbb{F}| \cdot \delta\right)^{d/2}$

→ Belaïd *et al.*:[13] $|C| \cdot \left(\mathcal{O}(1) \cdot |\mathbb{F}| \cdot \delta\right)^{\approx d/3}$

→ Eurocrypt'25, Asiacrypt'25: tighter composition (yet more complex)

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

[13] Taleb, "Secure and Verified Cryptographic Implementations in the Random Probing Model. (Implémentations cryptographiques sûres et vérifiées dans le modèle random probing)".

References I

Barthe, G. et al. “Masking the GLP Lattice-Based Signature Scheme at Any Order”. In: *J. Cryptol.* 37.1 (2024), p. 5. DOI: 10.1007/S00145-023-09485-Z. URL: https://doi.org/10.1007/s00145-023-09485-z.

Barthe, G. et al. “Strong Non-Interference and Type-Directed Higher-Order Masking”. In: *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. CCS ’16. Vienna, Austria: Association for Computing Machinery, 2016, 116–129. ISBN: 9781450341394. DOI: 10.1145/2976749.2978427. URL: https://doi.org/10.1145/2976749.2978427.

References II

Boucheron, S., G. Lugosi, and P. Massart. *Concentration Inequalities: A Nonasymptotic Theory of Independence*. Oxford University Press, 2013. ISBN: 9780191747106. URL: https://books.google.fr/books?id=O3yoAQAACAAJ.

Chari, S. et al. "Towards Sound Approaches to Counteract Power-Analysis Attacks". In: *Advances in Cryptology - CRYPTO '99, 19th Annual International Cryptology Conference, Santa Barbara, California, USA, August 15-19, 1999, Proceedings*. Ed. by M. J. Wiener. Vol. 1666. Lecture Notes in Computer Science. Springer, 1999, pp. 398–412. ISBN: 3-540-66347-9. DOI: 10.1007/3-540-48405-1_26. URL: https://doi.org/10.1007/3-540-48405-1_26.

References III

- Coron, J.-S. et al. *High-order Polynomial Comparison and Masking Lattice-based Encryption*. Cryptology ePrint Archive, Paper 2021/1615. 2021. URL: https://eprint.iacr.org/2021/1615.
- 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.

References IV

Goubin, L. and J. Patarin. “DES and Differential Power Analysis (The "Duplication" Method)”. In: *Cryptographic Hardware and Embedded Systems, First International Workshop, CHES’99, Worcester, MA, USA, August 12-13, 1999, Proceedings*. Ed. by Ç. K. Koç and C. Paar. Vol. 1717. Lecture Notes in Computer Science. Springer, 1999, pp. 158–172. DOI: 10.1007/3-540-48059-5_15. URL: https://doi.org/10.1007/3-540-48059-5_15.

References V

Ishai, Y., A. Sahai, and D. A. Wagner. "Private Circuits: Securing Hardware against Probing Attacks". In: *Advances in Cryptology - CRYPTO 2003, 23rd Annual International Cryptology Conference, Santa Barbara, California, USA, August 17-21, 2003, Proceedings*. Ed. by D. Boneh. Vol. 2729. Lecture Notes in Computer Science. Springer, 2003, pp. 463–481. DOI: 10.1007/978-3-540-45146-4_27. URL: https://doi.org/10.1007/978-3-540-45146-4_27.

References VI

Rivain, M. and E. Prouff. “Provably Secure Higher-Order Masking of AES”. In: *Cryptographic Hardware and Embedded Systems, CHES 2010, 12th International Workshop, Santa Barbara, CA, USA, August 17-20, 2010. Proceedings*. Ed. by S. Mangard and F. Standaert. Vol. 6225. Lecture Notes in Computer Science. Springer, 2010, pp. 413–427. DOI: 10.1007/978-3-642-15031-9_28. URL: https://doi.org/10.1007/978-3-642-15031-9_28.

Taleb, A. R. “Secure and Verified Cryptographic Implementations in the Random Probing Model. (Implémentations cryptographiques sûres et vérifiées dans le modèle random probing)”. PhD thesis. Sorbonne University, Paris, France, 2023. URL: https://tel.archives-ouvertes.fr/tel-04457258.

Reduction from Noisy Leakage to Random Probing (I)

Assume there exists such a simulator $\mathcal{S}$,

Reduction from Noisy Leakage to Random Probing (I)

Assume there exists such a simulator $\mathcal{S}$, we need to construct it for all inputs:

$$\begin{aligned} \Pr[\mathcal{S}(x) = l] &= \ldots, \text{ for all } x \\ \Pr[\mathcal{S}(\perp) = l] &= \ldots \end{aligned}$$

Constraints:

Reduction from Noisy Leakage to Random Probing (I)

Assume there exists such a simulator $\mathcal{S}$, we need to construct it for all inputs:

$$\begin{aligned}\Pr[\mathcal{S}(x) = l] &= \ldots, \text{ for all } x \\ \Pr[\mathcal{S}(\perp) = l] &= \ldots\end{aligned}$$

Constraints:

$\rightarrow$ For all input x, $\Pr[\mathcal{S}(x)]$ should be a p.m.f.

Reduction from Noisy Leakage to Random Probing (I)

Assume there exists such a simulator $\mathcal{S}$, we need to construct it for all inputs:

$$\begin{aligned} \Pr[\mathcal{S}(x) = l] &= \ldots, \text{ for all } x \\ \Pr[\mathcal{S}(\perp) = l] &= \ldots \end{aligned}$$

Constraints:

$\rightarrow$ For all input x, $\Pr[\mathcal{S}(x)]$ should be a p.m.f.

$\rightarrow$ For the input $\perp$, $\Pr[\mathcal{S}(\perp)]$ should be a p.m.f.

Reduction from Noisy Leakage to Random Probing (I)

Assume there exists such a simulator $\mathcal{S}$, we need to construct it for all inputs:

$$
\begin{aligned}
\Pr[\mathcal{S}(x) = l] &= \ldots, \text{ for all } x \\
\Pr[\mathcal{S}(\bot) = l] &= \ldots
\end{aligned}
$$

Constraints:

→ For all input x, $\Pr[\mathcal{S}(x)]$ should be a p.m.f.

→ For the input $\bot$, $\Pr[\mathcal{S}(\bot)]$ should be a p.m.f.

→ For any x, l, $\Pr[\mathcal{S}(\varphi(x)) = l] = \Pr[\mathrm{L}(x) = l]$

Reduction from Noisy Leakage to Random Probing (II)

Let us start from the last constraint. For any x and any l:

$$\Pr[\mathrm{L}(x) = l] \;=\; \Pr[\mathcal{S}\left(\varphi(x)\right) = l]$$

Reduction from Noisy Leakage to Random Probing (II)

Let us start from the last constraint. For any x and any l:

$$\begin{aligned} \Pr[\mathrm{L}(x) = l] &= \Pr[\mathcal{S}(\varphi(x)) = l] \\ &= \Pr[\varphi(x) = x] \cdot \Pr[\mathcal{S}(x) = l] + \Pr[\varphi(x) = \perp] \cdot \Pr[\mathcal{S}(\perp) = l] \end{aligned}$$

Reduction from Noisy Leakage to Random Probing (II)

Let us start from the last constraint. For any x and any l:

$$\begin{aligned}
\Pr[\mathrm{L}(x) = l] &= \Pr[\mathcal{S}(\varphi(x)) = l] \\
&= \Pr[\varphi(x) = x] \cdot \Pr[\mathcal{S}(x) = l] + \Pr[\varphi(x) = \perp] \cdot \Pr[\mathcal{S}(\perp) = l] \\
&= \epsilon \cdot \Pr[\mathcal{S}(x) = l] + (1 - \epsilon) \cdot \Pr[\mathcal{S}(\perp) = l]
\end{aligned}$$

Reduction from Noisy Leakage to Random Probing (II)

Let us start from the last constraint. For any x and any l:

$$\begin{aligned}
\Pr[\mathrm{L}(x) = l] &= \Pr[\mathcal{S}\left(\varphi(x)\right) = l] \\
&= \Pr[\varphi(x) = x] \cdot \Pr[\mathcal{S}\left(x\right) = l] + \Pr[\varphi(x) = \bot] \cdot \Pr[\mathcal{S}\left(\bot\right) = l] \\
&= \epsilon \cdot \Pr[\mathcal{S}\left(x\right) = l] + (1 - \epsilon) \cdot \Pr[\mathcal{S}\left(\bot\right) = l]
\end{aligned}$$

Hence,

$$0 \leq \Pr[\mathcal{S}\left(\bot\right) = l] = \frac{\overbrace{\Pr[\mathrm{L}(x) = l] - \epsilon \cdot \Pr[\mathcal{S}\left(x\right) = l]}^{\text{Should not depend on } x}}{1 - \epsilon}$$

Reduction from Noisy Leakage to Random Probing (II)

Let us start from the last constraint. For any x and any l:

$$\begin{aligned}
\Pr[\mathrm{L}(x) = l] &= \Pr[\mathcal{S}(\varphi(x)) = l] \\
&= \Pr[\varphi(x) = x] \cdot \Pr[\mathcal{S}(x) = l] + \Pr[\varphi(x) = \perp] \cdot \Pr[\mathcal{S}(\perp) = l] \\
&= \epsilon \cdot \Pr[\mathcal{S}(x) = l] + (1 - \epsilon) \cdot \Pr[\mathcal{S}(\perp) = l]
\end{aligned}$$

Hence,

$$0 \leq \Pr[\mathcal{S}(\perp) = l] = \frac{\overbrace{\Pr[\mathrm{L}(x) = l] - \epsilon \cdot \Pr[\mathcal{S}(x) = l]}^{\text{Should not depend on } x}}{1 - \epsilon} = \frac{\pi(l)}{1 - \epsilon} \qquad (4)$$

Reduction from Noisy Leakage to Random Probing (II)

Let us start from the last constraint. For any x and any l:

$$\begin{aligned}
\Pr[\mathrm{L}(x) = l] &= \Pr[\mathcal{S}(\varphi(x)) = l] \\
&= \Pr[\varphi(x) = x] \cdot \Pr[\mathcal{S}(x) = l] + \Pr[\varphi(x) = \bot] \cdot \Pr[\mathcal{S}(\bot) = l] \\
&= \epsilon \cdot \Pr[\mathcal{S}(x) = l] + (1 - \epsilon) \cdot \Pr[\mathcal{S}(\bot) = l]
\end{aligned}$$

Hence,

$$0 \leq \Pr[\mathcal{S}(\bot) = l] = \frac{\overbrace{\Pr[\mathrm{L}(x) = l] - \epsilon \cdot \Pr[\mathcal{S}(x) = l]}^{\text{Should not depend on } x}}{1 - \epsilon} = \frac{\pi(l)}{1 - \epsilon} \quad (4)$$

$$0 \leq \Pr[\mathcal{S}(x) = l] = \frac{\Pr[\mathrm{L}(x) = l] - \pi(l)}{\epsilon} \quad (5)$$

Reduction from Noisy Leakage to Random Probing (II)

Let us start from the last constraint. For any x and any l:

$$\begin{aligned}
\Pr[\mathrm{L}(x) = l] &= \Pr[\mathcal{S}(\varphi(x)) = l] \\
&= \Pr[\varphi(x) = x] \cdot \Pr[\mathcal{S}(x) = l] + \Pr[\varphi(x) = \bot] \cdot \Pr[\mathcal{S}(\bot) = l] \\
&= \epsilon \cdot \Pr[\mathcal{S}(x) = l] + (1 - \epsilon) \cdot \Pr[\mathcal{S}(\bot) = l]
\end{aligned}$$

Hence,

$$0 \leq \Pr[\mathcal{S}(\bot) = l] = \overbrace{\frac{\Pr[\mathrm{L}(x) = l] - \epsilon \cdot \Pr[\mathcal{S}(x) = l]}{1 - \epsilon}}^{\text{Should not depend on } X} = \frac{\pi(l)}{1 - \epsilon} \quad (4)$$

$$0 \leq \Pr[\mathcal{S}(x) = l] = \frac{\Pr[\mathrm{L}(x) = l] - \pi(l)}{\epsilon} \quad (5)$$

Is there any ϵ such that $\geq$ and $\geq$ are valid?

Reduction from Noisy Leakage to RP (III)

Is there any ϵ such that $\geq$ and $\geq$ are valid?

Reduction from Noisy Leakage to RP (III)

Is there any ϵ such that $\geq$ and $\geq$ are valid? From (6), and (7), we get

$$0 \leq \pi(l) \leq \Pr[\mathrm{L}(x) = l] \text{ for any } x$$

Reduction from Noisy Leakage to RP (III)

Is there any ϵ such that $\geq$ and $\geq$ are valid? From (6), and (7), we get

$$0 \leq \pi(l) \leq \Pr[\mathrm{L}(x) = l] \text{ for any } x$$

In other words,

$$0 \leq \pi(l) \leq \min_{x} \Pr[\mathrm{L}(x) = l]$$

Reduction from Noisy Leakage to RP (III)

Is there any ϵ such that $\geq$ and $\geq$ are valid? From (6), and (7), we get

$$0 \leq \pi(l) \leq \Pr[\mathrm{L}(x) = l] \text{ for any } x$$

In other words,

$$0 \leq \pi(l) \leq \min_x \Pr[\mathrm{L}(x) = l]$$

Furthermore, summing (6) over l, by definition of probability distributions,

$$\sum_l \pi(l) = \underbrace{\sum_l \Pr[\mathrm{L}(x) = l]}_{=1} - \epsilon \cdot \underbrace{\sum_l \Pr[\mathcal{S}(x) = l]}_{=1}$$

Reduction from Noisy Leakage to RP (III)

Is there any ϵ such that $\geq$ and $\geq$ are valid? From (6), and (7), we get

$$0 \leq \pi(l) \leq \Pr[\mathrm{L}(x) = l] \text{ for any } x$$

In other words,

$$0 \leq \pi(l) \leq \min_x \Pr[\mathrm{L}(x) = l]$$

Furthermore, summing (6) over l, by definition of probability distributions,

$$\sum_l \pi(l) = \underbrace{\sum_l \Pr[\mathrm{L}(x) = l]}_{=1} - \epsilon \cdot \underbrace{\sum_l \Pr[\mathcal{S}(x) = l]}_{=1} = 1 - \epsilon$$

Reduction from Noisy Leakage to RP (III)

Is there any ϵ such that $\geq$ and $\geq$ are valid? From (6), and (7), we get

$$0 \leq \pi(l) \leq \Pr[\mathrm{L}(x) = l] \text{ for any } x$$

In other words,

$$0 \leq \pi(l) \leq \min_x \Pr[\mathrm{L}(x) = l]$$

Furthermore, summing (6) over l, by definition of probability distributions,

$$\sum_l \pi(l) = \underbrace{\sum_l \Pr[\mathrm{L}(x) = l]}_{=1} - \epsilon \cdot \underbrace{\sum_l \Pr[\mathcal{S}(x) = l]}_{=1} = 1 - \epsilon$$

Hence,

$$\epsilon = 1 - \sum_l \pi(l) \geq 1 - \sum_l \min_x \Pr[\mathrm{L}(x) = l]$$

Reduction from Noisy Leakage to RP (III)

Is there any ϵ such that $\geq$ and $\geq$ are valid? From (6), and (7), we get

$$0 \leq \pi(l) \leq \Pr[\mathrm{L}(x) = l] \text{ for any } x$$

In other words,

$$0 \leq \pi(l) \leq \min_x \Pr[\mathrm{L}(x) = l]$$

Furthermore, summing (6) over l, by definition of probability distributions,

$$\sum_l \pi(l) = \underbrace{\sum_l \Pr[\mathrm{L}(x) = l]}_{=1} - \epsilon \cdot \underbrace{\sum_l \Pr[\mathcal{S}(x) = l]}_{=1} = 1 - \epsilon$$

Hence, to have the smallest ϵ,

$$\epsilon = 1 - \sum_l \pi(l) = 1 - \sum_l \min_x \Pr[\mathrm{L}(x) = l]$$

Reduction to Average Random Probing (I)

For any x and any l:

$$\Pr[\mathrm{L}(x) = l] = \Pr[\mathcal{S}\left(\varphi(x)\right) = l]$$

Reduction to Average Random Probing (I)

For any x and any l:

$$
\begin{aligned}
\Pr[\mathrm{L}(x) = l] &= \Pr[\mathcal{S}(\varphi(x)) = l] \\
&= \Pr[\varphi(x) = x] \cdot \Pr[\mathcal{S}(x) = l] + \Pr[\varphi(x) = \bot] \cdot \Pr[\mathcal{S}(\bot) = l]
\end{aligned}
$$

Reduction to Average Random Probing (I)

For any x and any l:

$$\begin{aligned}
\Pr[\mathrm{L}(x) = l] &= \Pr[\mathcal{S}(\varphi(x)) = l] \\
&= \Pr[\varphi(x) = x] \cdot \Pr[\mathcal{S}(x) = l] + \Pr[\varphi(x) = \bot] \cdot \Pr[\mathcal{S}(\bot) = l] \\
&= \epsilon_x \cdot \Pr[\mathcal{S}(x) = l] + (1 - \epsilon_x) \cdot \Pr[\mathcal{S}(\bot) = l]
\end{aligned}$$

Reduction to Average Random Probing (I)

For any x and any l:

$$\begin{aligned}\Pr[\mathrm{L}(x) = l] &= \Pr[\mathcal{S}\left(\varphi(x)\right) = l] \\ &= \Pr[\varphi(x) = x] \cdot \Pr[\mathcal{S}\left(x\right) = l] + \Pr[\varphi(x) = \perp] \cdot \Pr[\mathcal{S}\left(\perp\right) = l] \\ &= \epsilon_x \cdot \Pr[\mathcal{S}\left(x\right) = l] + (1 - \epsilon_x) \cdot \Pr[\mathcal{S}\left(\perp\right) = l]\end{aligned}$$

Hence, provided that $\epsilon_x < 1$,

$$0 \leq \Pr[\mathcal{S}\left(\perp\right) = l] = \frac{\overbrace{\Pr[\mathrm{L}(x) = l] - \epsilon_x \cdot \Pr[\mathcal{S}\left(x\right) = l]}^{\text{Should not depend on } x}}{1 - \epsilon_x}$$

Reduction to Average Random Probing (I)

For any x and any l:

$$\begin{aligned}
\Pr[\mathrm{L}(x) = l] &= \Pr[\mathcal{S}(\varphi(x)) = l] \\
&= \Pr[\varphi(x) = x] \cdot \Pr[\mathcal{S}(x) = l] + \Pr[\varphi(x) = \perp] \cdot \Pr[\mathcal{S}(\perp) = l] \\
&= \epsilon_x \cdot \Pr[\mathcal{S}(x) = l] + (1 - \epsilon_x) \cdot \Pr[\mathcal{S}(\perp) = l]
\end{aligned}$$

Hence, provided that $\epsilon_x < 1$,

$$0 \leq \Pr[\mathcal{S}(\perp) = l] = \frac{\overbrace{\Pr[\mathrm{L}(x) = l] - \epsilon_x \cdot \Pr[\mathcal{S}(x) = l]}^{\text{Should not depend on } x}}{1 - \epsilon_x} = \frac{\pi(l, x)}{1 - \epsilon_x} \quad (6)$$

Reduction to Average Random Probing (I)

For any x and any l:

$$\begin{aligned}\Pr[\mathrm{L}(x) = l] &= \Pr[\mathcal{S}(\varphi(x)) = l] \\ &= \Pr[\varphi(x) = x] \cdot \Pr[\mathcal{S}(x) = l] + \Pr[\varphi(x) = \perp] \cdot \Pr[\mathcal{S}(\perp) = l] \\ &= \epsilon_x \cdot \Pr[\mathcal{S}(x) = l] + (1 - \epsilon_x) \cdot \Pr[\mathcal{S}(\perp) = l]\end{aligned}$$

Hence, provided that $\epsilon_x < 1$,

$$0 \leq \Pr[\mathcal{S}(\perp) = l] = \frac{\overbrace{\Pr[\mathrm{L}(x) = l] - \epsilon_x \cdot \Pr[\mathcal{S}(x) = l]}^{\text{Should not depend on } x}}{1 - \epsilon_x} = \frac{\pi(l, x)}{1 - \epsilon_x} \quad (6)$$

$$0 \leq \Pr[\mathcal{S}(x) = l] = \frac{\Pr[\mathrm{L}(x) = l] - \pi(l, x)}{\epsilon_x} \quad (7)$$

Reduction to Average Random Probing (I)

For any x and any l:

$$\begin{aligned}\Pr[\mathrm{L}(x) = l] &= \Pr[\mathcal{S}(\varphi(x)) = l] \\ &= \Pr[\varphi(x) = x] \cdot \Pr[\mathcal{S}(x) = l] + \Pr[\varphi(x) = \perp] \cdot \Pr[\mathcal{S}(\perp) = l] \\ &= \epsilon_x \cdot \Pr[\mathcal{S}(x) = l] + (1 - \epsilon_x) \cdot \Pr[\mathcal{S}(\perp) = l]\end{aligned}$$

Hence, provided that $\epsilon_x < 1$,

$$0 \leq \Pr[\mathcal{S}(\perp) = l] = \frac{\overbrace{\Pr[\mathrm{L}(x) = l] - \epsilon_x \cdot \Pr[\mathcal{S}(x) = l]}^{\text{Should not depend on } X}}{1 - \epsilon_x} = \frac{\pi(l, x)}{1 - \epsilon_x} \quad (6)$$

$$0 \leq \Pr[\mathcal{S}(x) = l] = \frac{\Pr[\mathrm{L}(x) = l] - \pi(l, x)}{\epsilon_x} \quad (7)$$

Is there any ϵ such that $\geq$ and $\geq$ are valid?

Reduction from Noisy Leakage to RP (III)

Is there any ϵ such that $\geq$ and $\geq$ are valid?

Reduction from Noisy Leakage to RP (III)

Is there any ϵ such that $\geq$ and $\geq$ are valid? From (6), and (7), we get

$$0 \leq \pi(l, x) \leq \Pr[\mathrm{L}(x) = l] \text{ for any } x$$

Reduction from Noisy Leakage to RP (III)

Is there any ϵ such that $\geq$ and $\geq$ are valid? From (6), and (7), we get

$$0 \leq \pi(l, x) \leq \Pr[\mathrm{L}(x) = l] \text{ for any } x$$

So (3) gives

$$\Pr[\mathcal{S}(\perp) = l] \leq \frac{\Pr[\mathrm{L}(x) = l]}{1 - \epsilon_x} \text{ for any } x \text{ s.t. } \epsilon_x < 1$$

Reduction from Noisy Leakage to RP (III)

Is there any ϵ such that $\geq$ and $\geq$ are valid? From (6), and (7), we get

$$0 \leq \pi(l, x) \leq \Pr[\mathrm{L}(x) = l] \text{ for any } x$$

So (3) gives

$$\Pr[\mathcal{S}(\perp) = l] \leq \frac{\Pr[\mathrm{L}(x) = l]}{1 - \epsilon_x} \text{ for any } x \text{ s.t. } \epsilon_x < 1$$

In other words,

$$0 \leq \Pr[\mathcal{S}(\perp) = l] \leq \min_{x' : \epsilon_{x'} < 1} \left\{ \frac{\Pr[\mathrm{L}(x') = l]}{1 - \epsilon_{x'}} \right\}$$

Reduction from Noisy Leakage to RP (III)

Is there any ϵ such that $\geq$ and $\geq$ are valid? From (6), and (7), we get

$$0 \leq \pi(l, x) \leq \Pr[\mathrm{L}(x) = l] \text{ for any } x$$

So (3) gives

$$\Pr[\mathcal{S}(\bot) = l] \leq \frac{\Pr[\mathrm{L}(x) = l]}{1 - \epsilon_x} \text{ for any } x \text{ s.t. } \epsilon_x < 1$$

In other words,

$$0 \leq \Pr[\mathcal{S}(\bot) = l] \leq \min_{x' : \epsilon_{x'} < 1} \left\{ \frac{\Pr[\mathrm{L}(x') = l]}{1 - \epsilon_{x'}} \right\}$$

And (3) also gives

$$0 \leq \pi(l, x) \leq (1 - \epsilon_x) \cdot \min_{x' : \epsilon_{x'} < 1} \left\{ \frac{\Pr[\mathrm{L}(x') = l]}{1 - \epsilon_{x'}} \right\}$$

Characterization of ARP-simulable Leakages

Furthermore, summing (6) over l, by definition of probability distributions,

$$\sum_l \pi(l, x) = \underbrace{\sum_l \Pr[\mathrm{L}(x) = l]}_{=1} - \epsilon_x \cdot \underbrace{\sum_l \Pr[\mathcal{S}(x) = l]}_{=1}$$

[14] One needs at least one $\epsilon_x < 1$ for non-trivial simulation

Characterization of ARP-simulable Leakages

Furthermore, summing (6) over l, by definition of probability distributions,

$$\sum_l \pi(l, x) = \underbrace{\sum_l \Pr[\mathrm{L}(x) = l]}_{=1} - \epsilon_x \cdot \underbrace{\sum_l \Pr[\mathcal{S}(x) = l]}_{=1} = 1 - \epsilon_x$$

[14] One needs at least one $\epsilon_x < 1$ for non-trivial simulation

Characterization of ARP-simulable Leakages

Furthermore, summing (6) over l, by definition of probability distributions,

$$\sum_{l} \pi(l, x) = \underbrace{\sum_{l} \Pr[\mathrm{L}(x) = l]}_{=1} - \epsilon_x \cdot \underbrace{\sum_{l} \Pr[\mathcal{S}(x) = l]}_{=1} = 1 - \epsilon_x$$

Hence, the following result

Theorem (ARP-Simulability)

L *is simulable in the* $\{\epsilon_x\}_x$ *average random probing model iff*[14]

$$1 \leq \sum_{l} \min_{x' : \epsilon_{x'} < 1} \left\{ \frac{\Pr[\mathrm{L}(x') = l]}{1 - \epsilon_{x'}} \right\}$$

[14] One needs at least one $\epsilon_x < 1$ for non-trivial simulation