A Quest for Provable Security against Side-Channel Attacks

Matthieu Rivain

AFRICACRYPT 2022

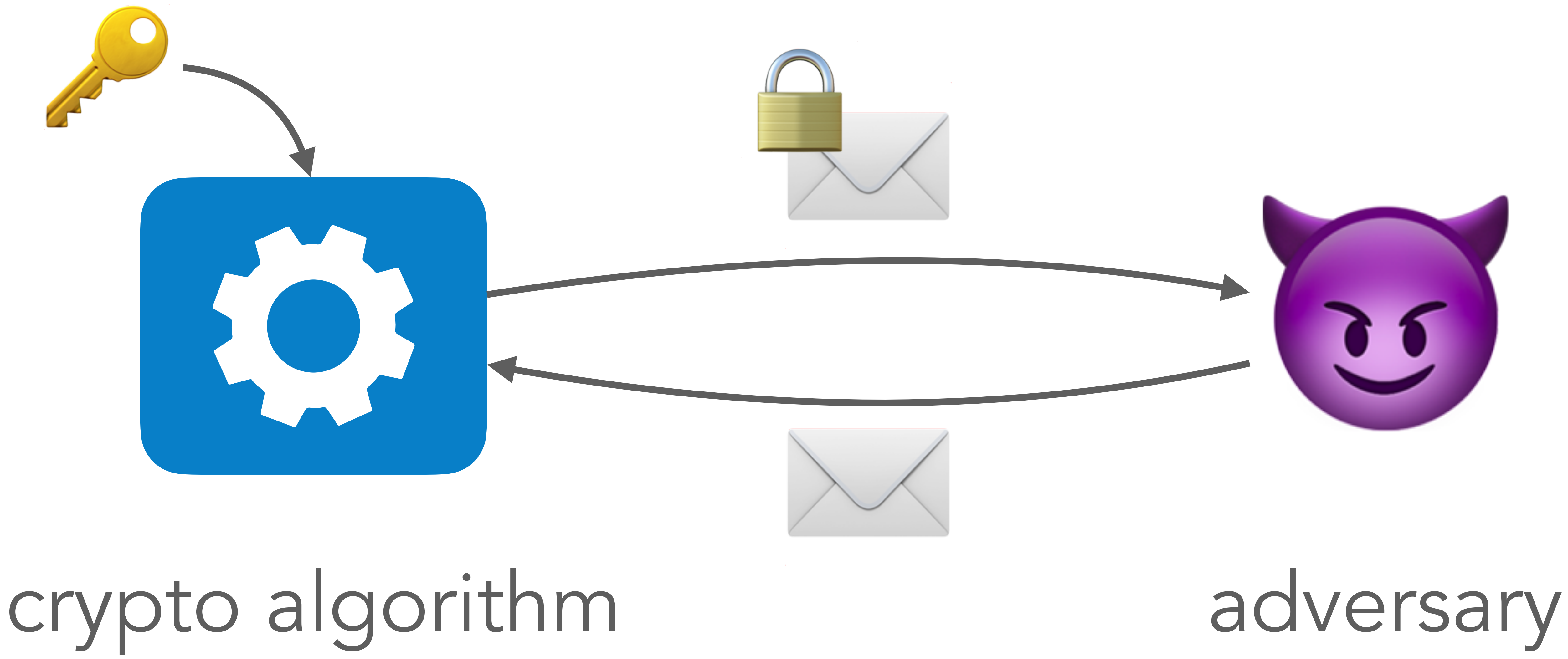
July 19, 2022, Fes, Morocco

CryptoExperts

WE INNOVATE TO SECURE YOUR BUSINESS

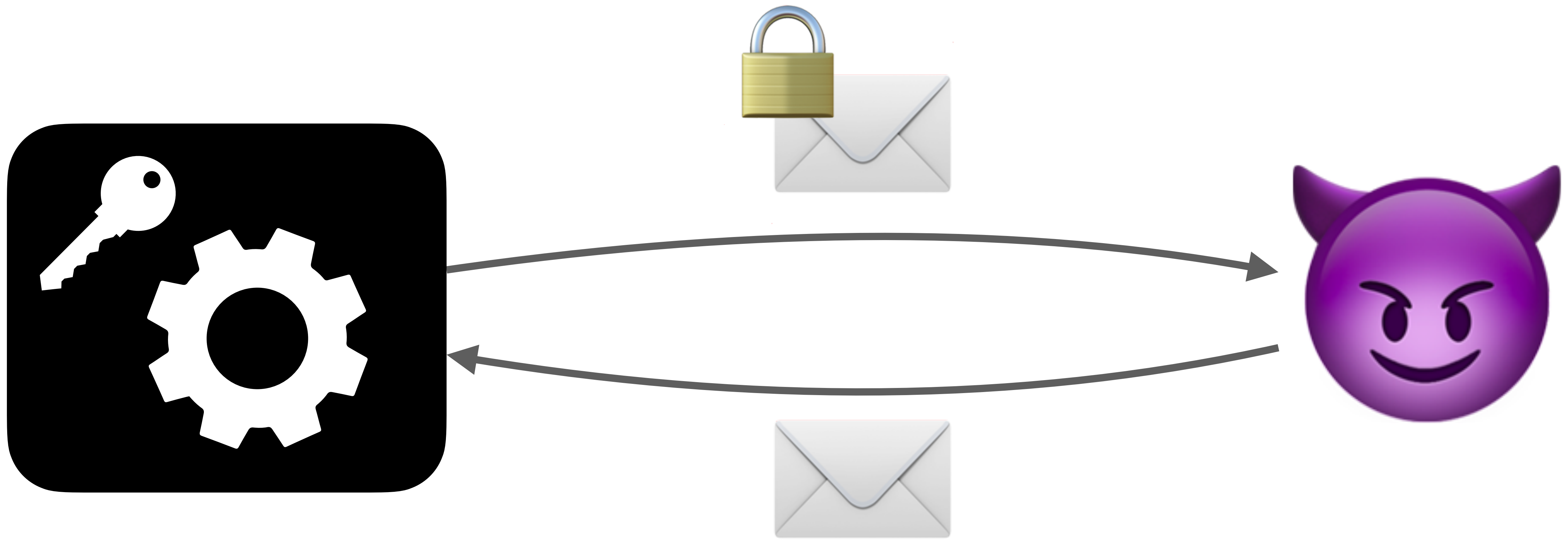
Introduction

Provable security



Provable security

Provable security

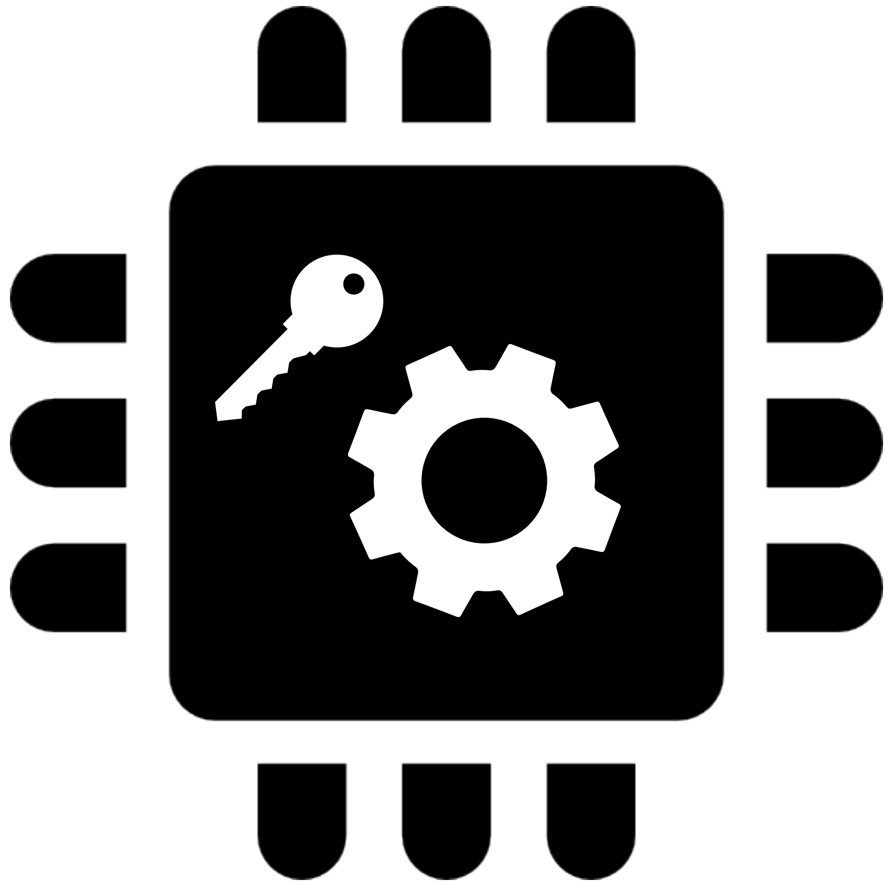


The "black-box model"

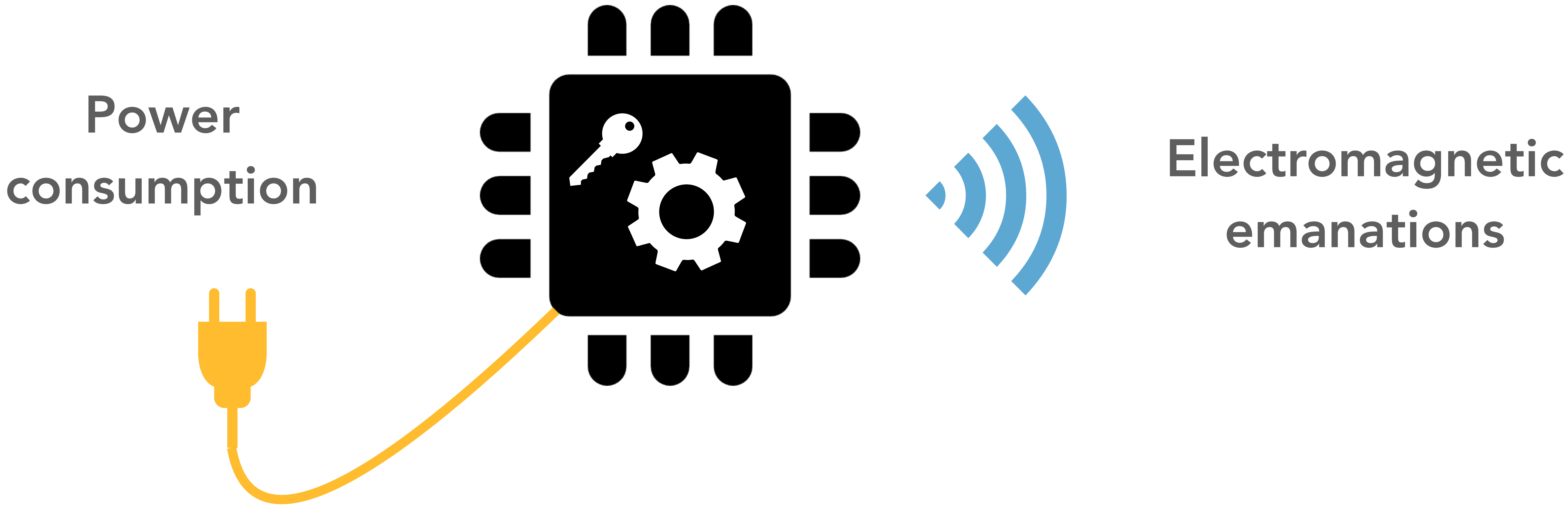
😈 needs unaffordable computing power to recover 🔑

security proof

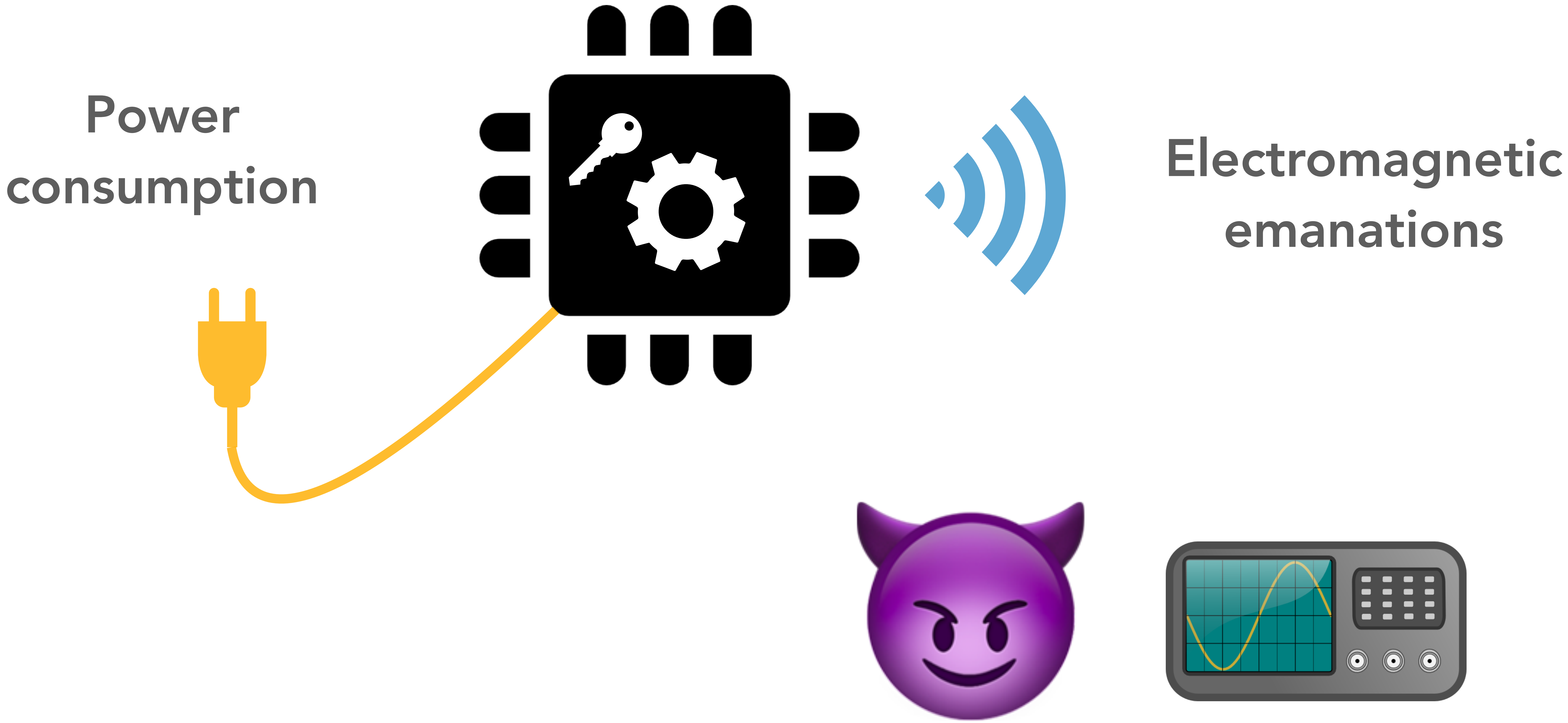
Side-channel attacks



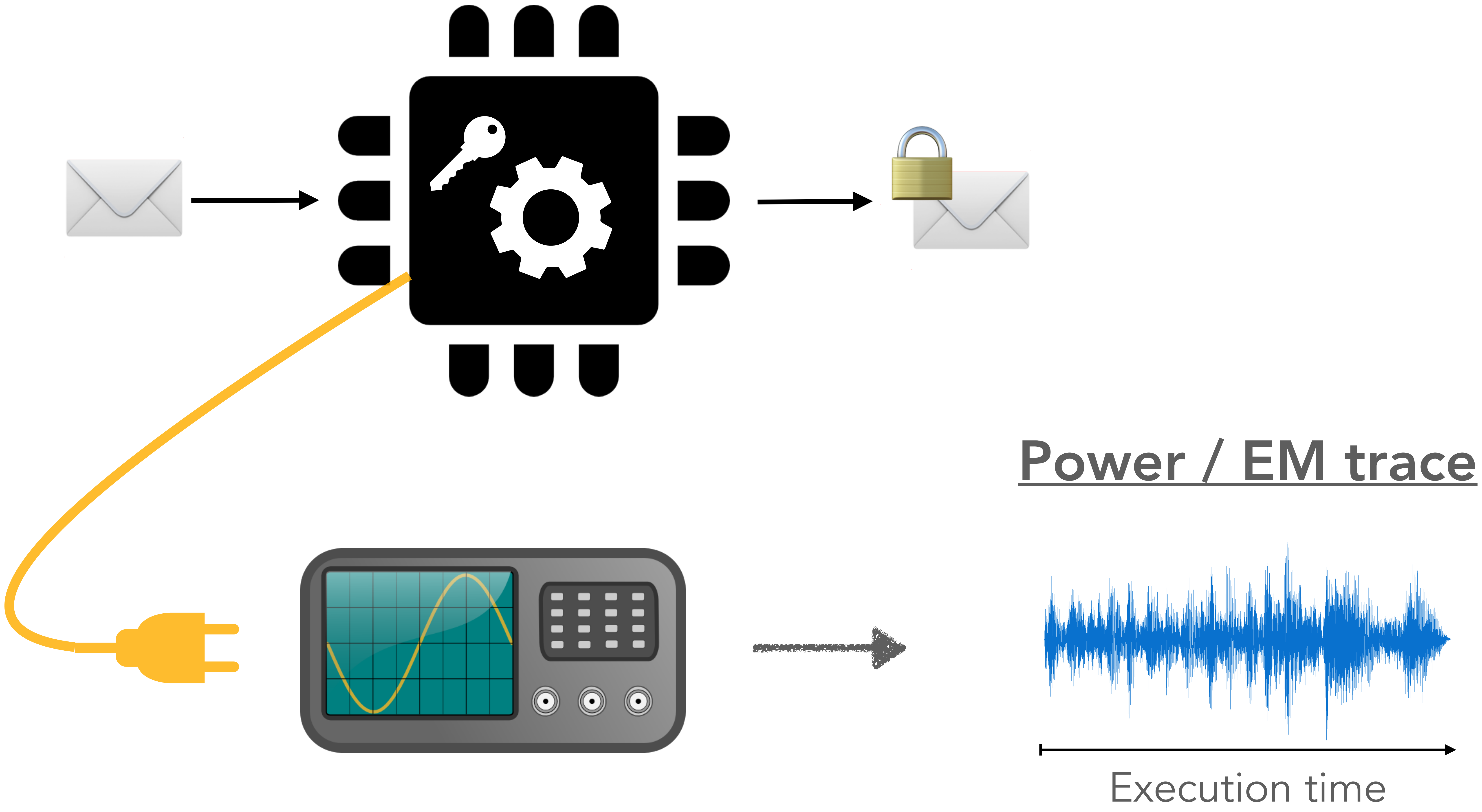
Side-channel attacks



Side-channel attacks



Differential power analysis



Masking

Apply secret sharing at the computation level

$$x = x_1 \oplus \cdots \oplus x_n$$

Masking

💡 Apply secret sharing at the computation level

$$x = x_1 \oplus \cdots \oplus x_n$$

randomly generated

Masking

Apply secret sharing at the computation level

$$x = x_1 \oplus \cdots \oplus x_n$$

randomly generated ($x_2 \oplus \cdots \oplus x_n$)

constrained to keep the correctness (x_1)

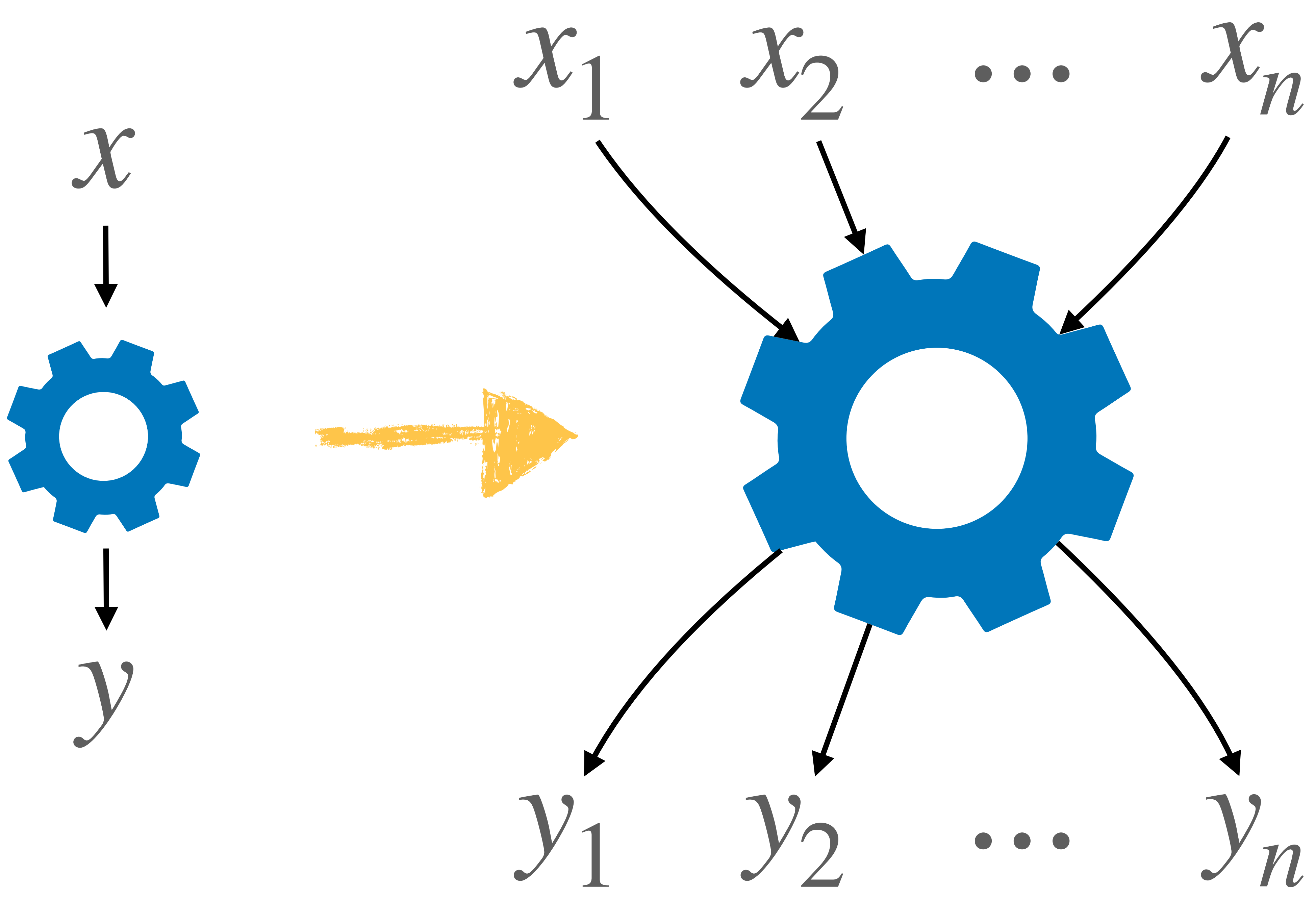
Masking

Apply secret sharing at the computation level

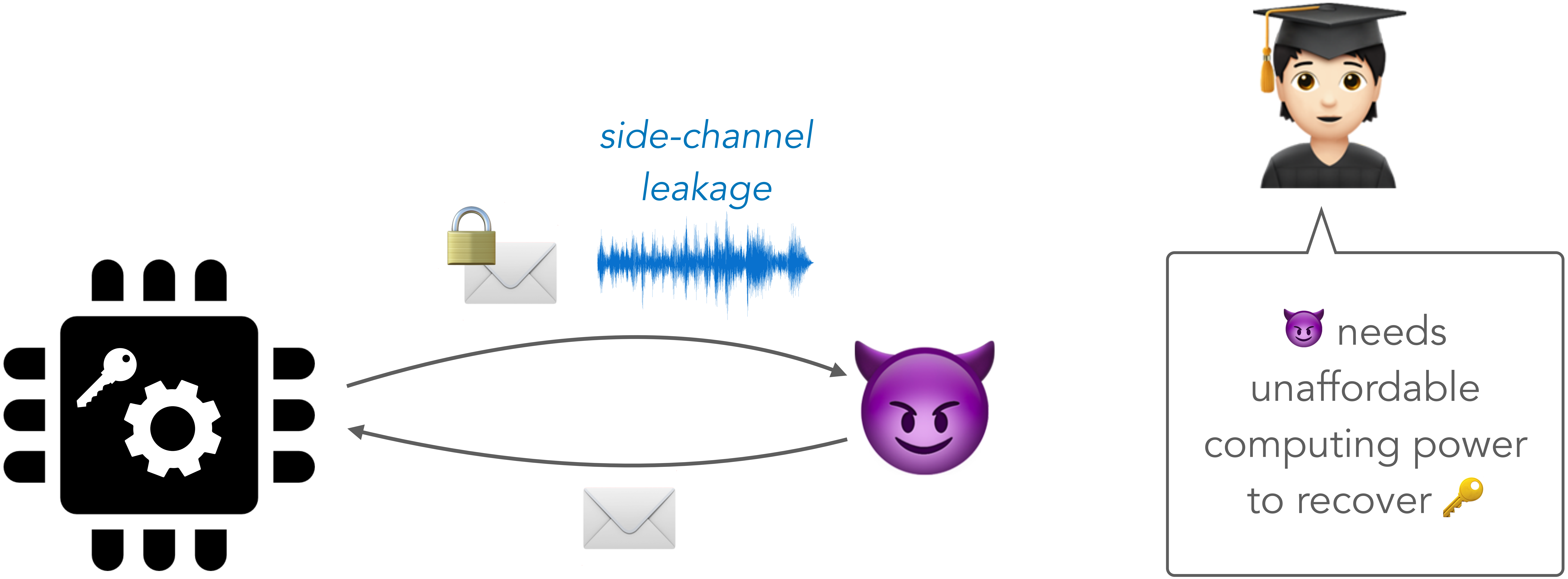
$$x = x_1 \oplus \cdots \oplus x_n$$

randomly generated

constrained to keep the correctness



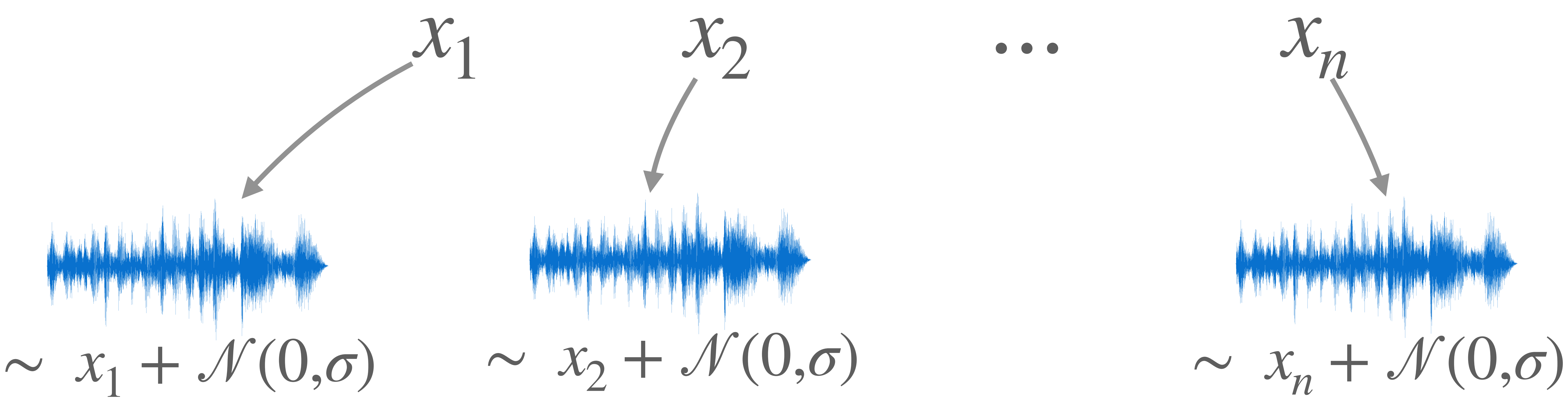
Provable security in the presence of leakage



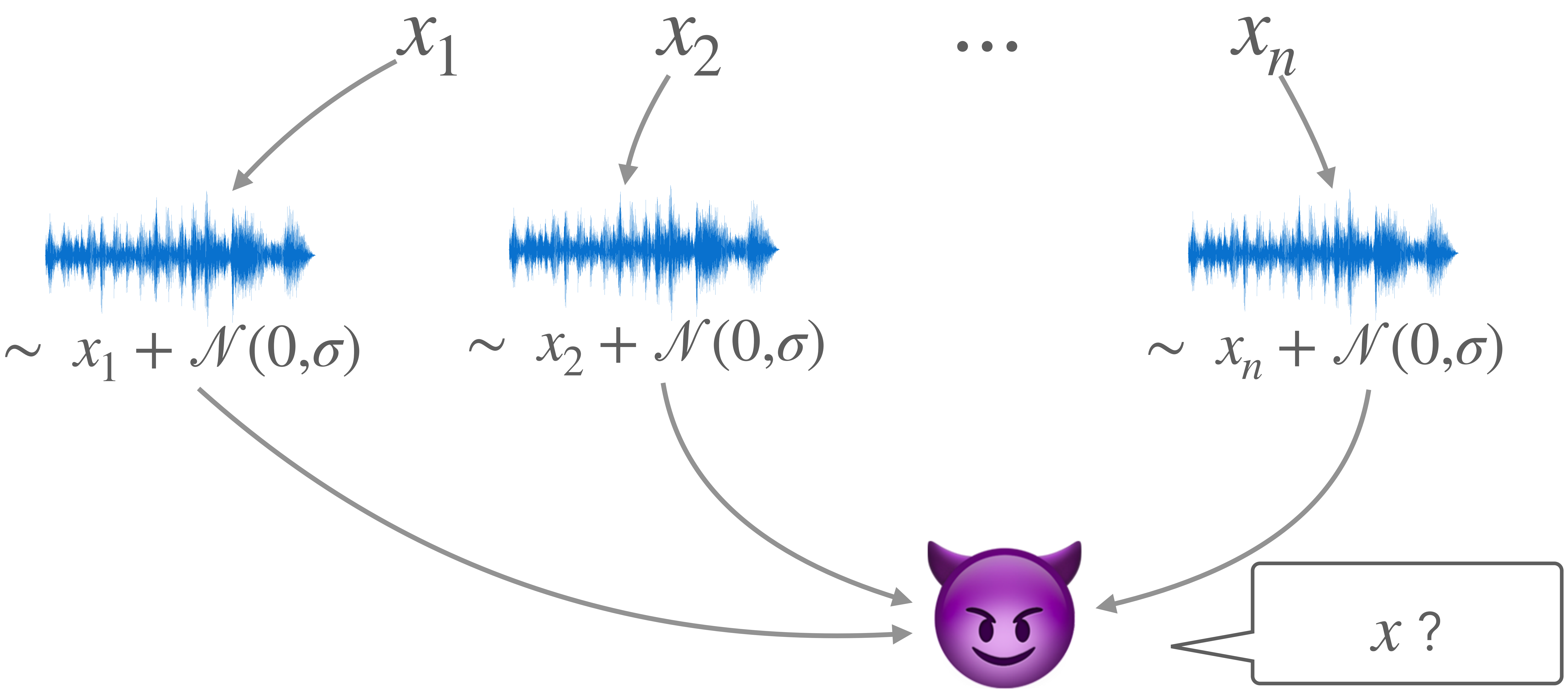
security proof

Modelling noisy leakage

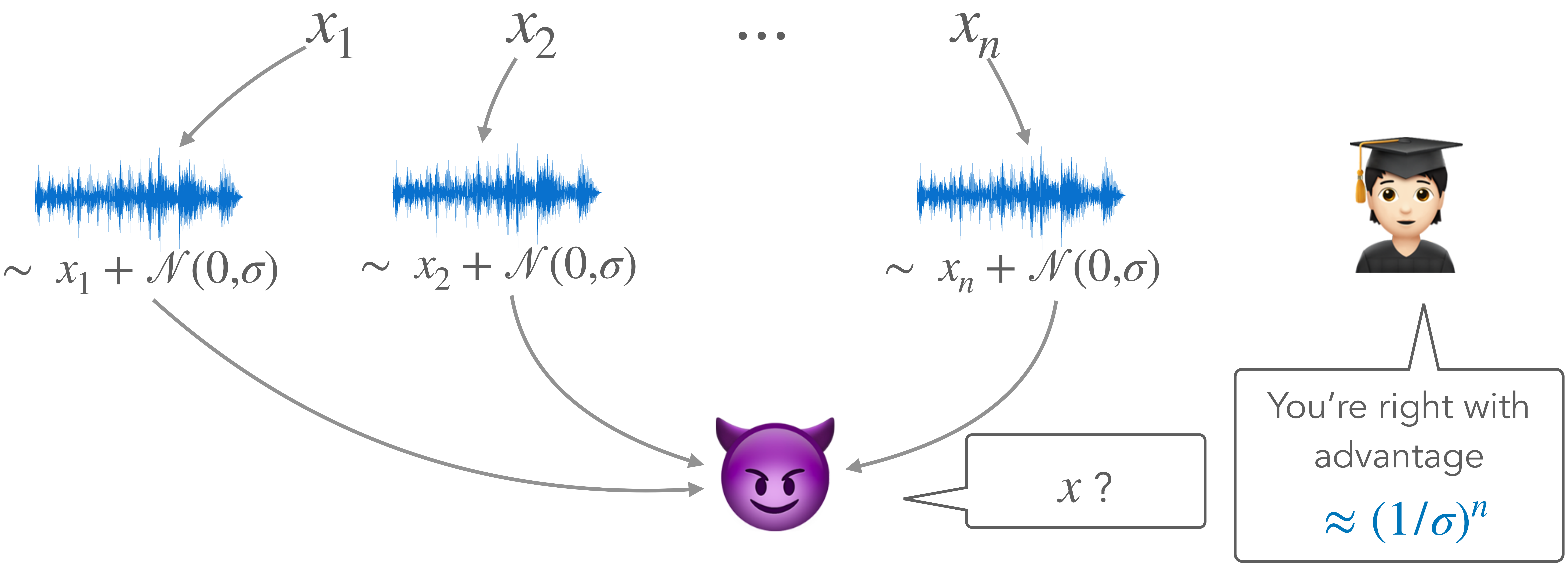
Motivation: soundness of masking with noise



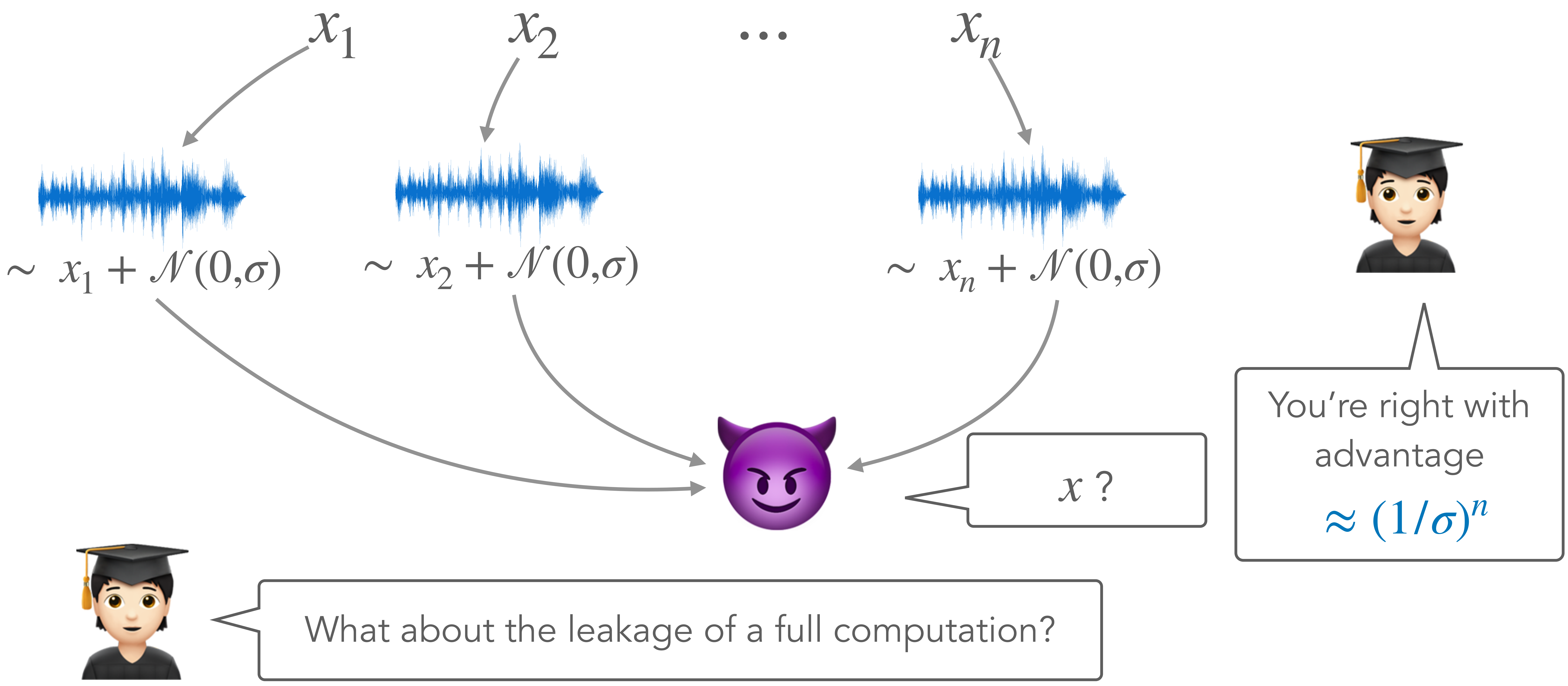
Motivation: soundness of masking with noise



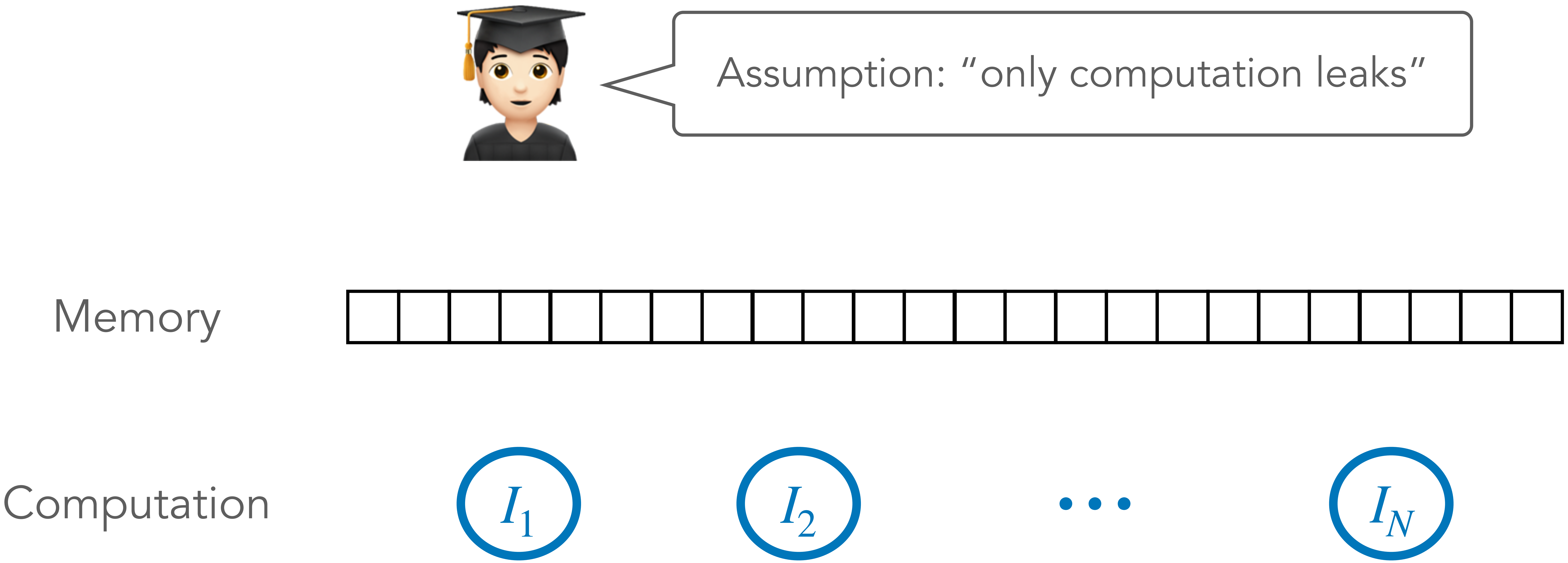
Motivation: soundness of masking with noise



Motivation: soundness of masking with noise

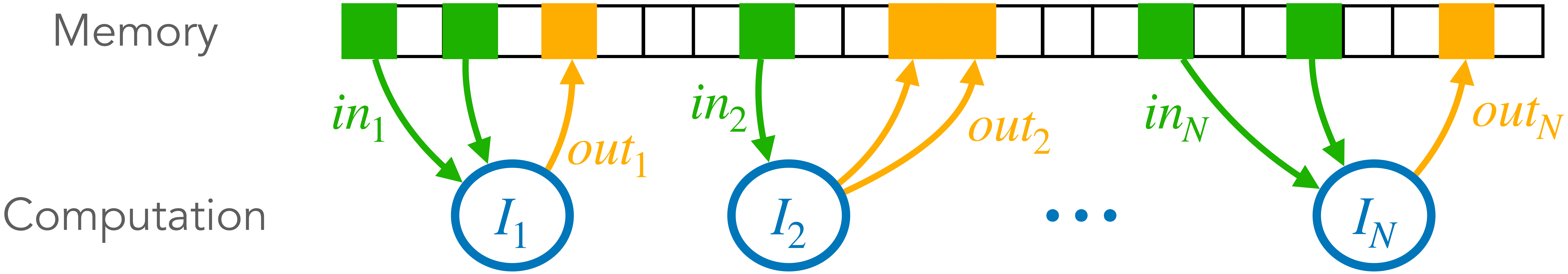


The noisy leakage model

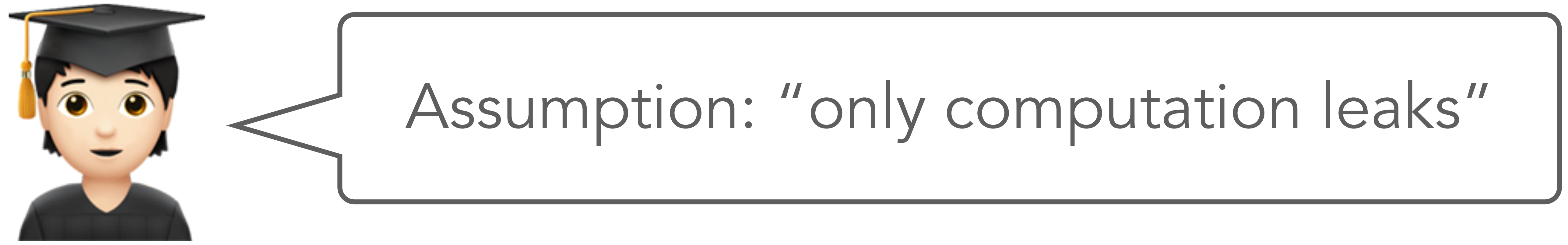


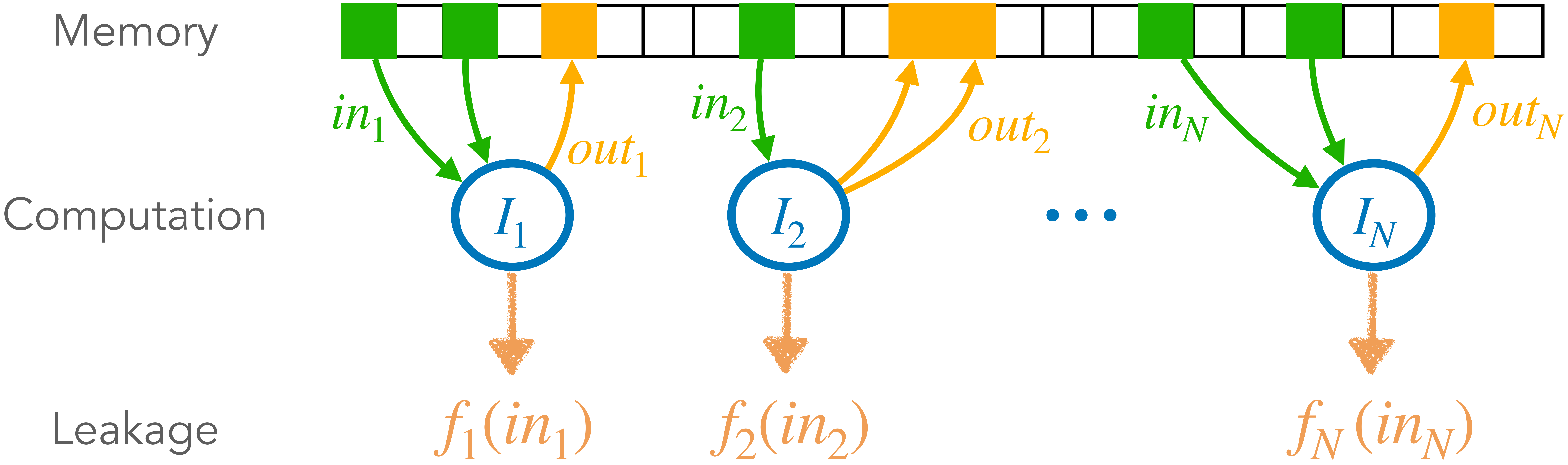
The noisy leakage model

Assumption: "only computation leaks"

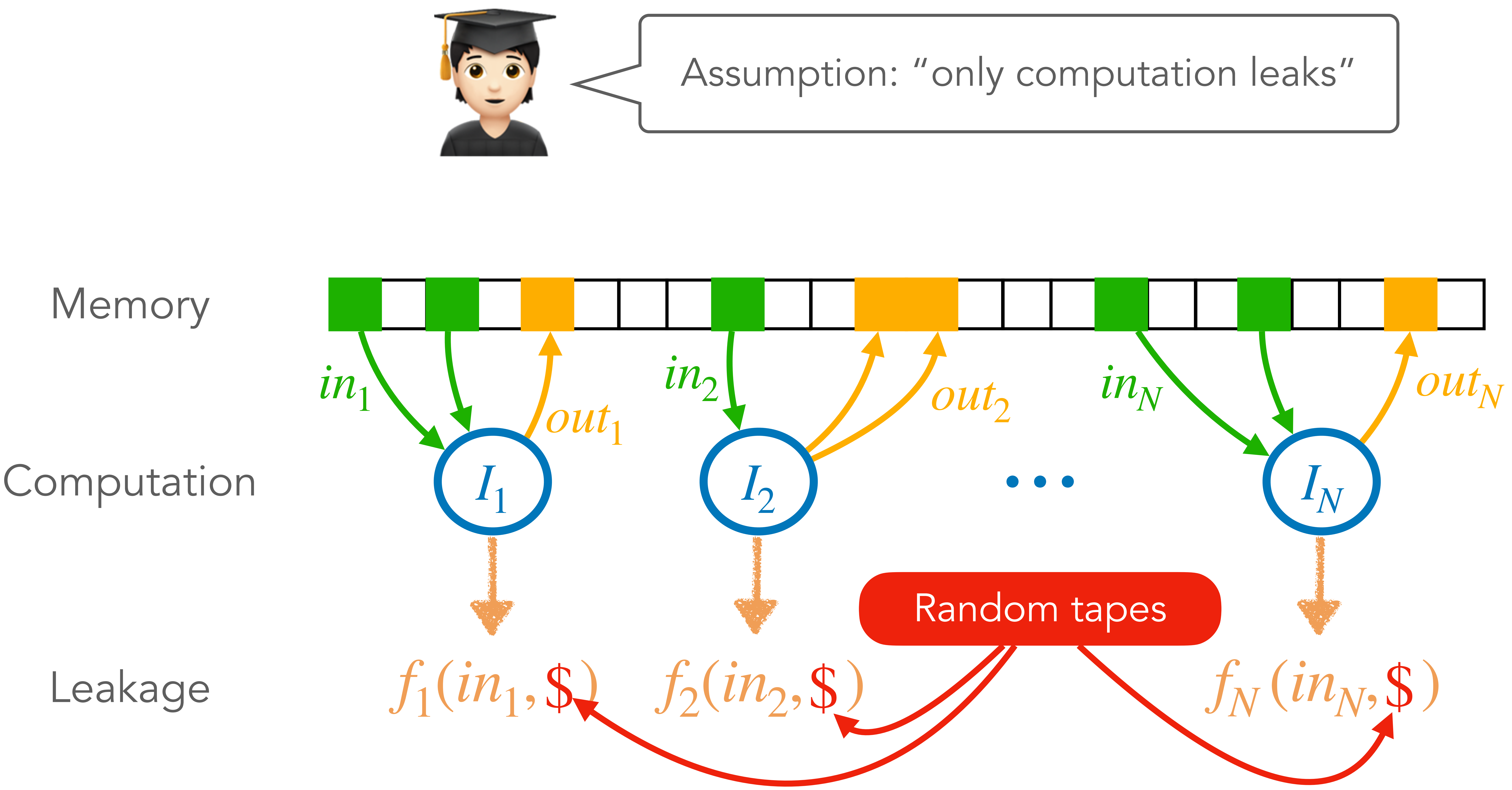


The noisy leakage model

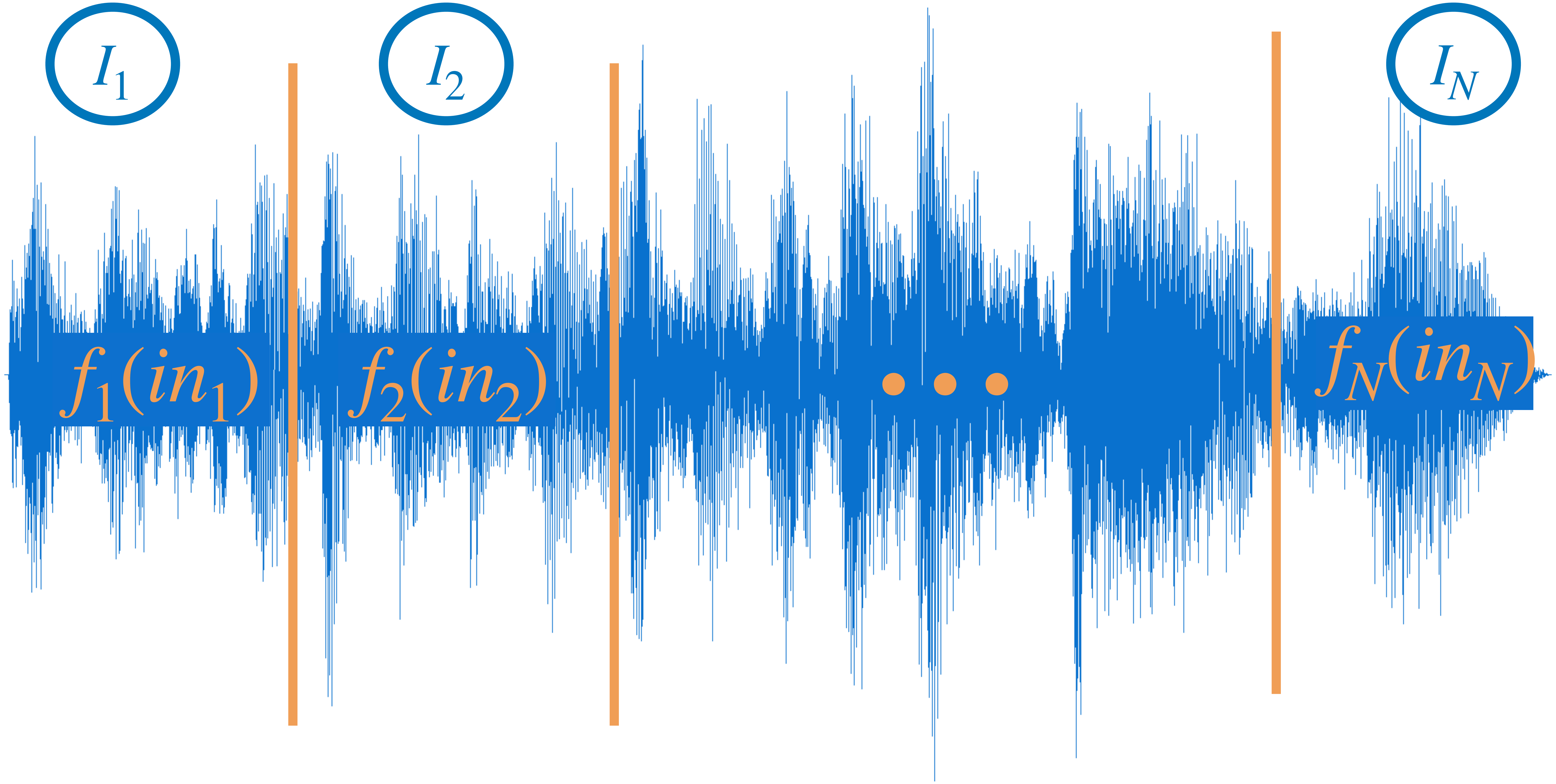




The noisy leakage model



The noisy leakage model



$f_i(in_i) \Rightarrow$ *multivariate noisy leakage*

Noisy leakage functions

A function is δ**-noisy** if (for $X \sim \mathcal{U}$):

$$\mathbb{E}_y\left[\Delta\left(X;\ \left(X \mid f(X) = y\right)\right)\right] \ \leq\ \delta$$

Noisy leakage functions

A function is δ**-noisy** if (for $X \sim \mathcal{U}$):

$$\mathbb{E}_y\left[\Delta\left(X;\ \left(X \mid f(X) = y\right)\right)\right] \ \leq\ \delta$$

statistical distance
between X and X
and given $f(X) = y$

Noisy leakage functions

A function is δ-**noisy** if (for $X \sim \mathcal{U}$):

$$\mathbb{E}_y\left[\Delta\left(X;\ \left(X \mid f(X) = y\right)\right)\right] \leq \delta$$

expectation on the possible leakage values

statistical distance between X and X and given $f(X) = y$

Noisy leakage functions

A function is **δ-noisy** if (for $X \sim \mathcal{U}$):

$$\mathbb{E}_y\left[\Delta\left(X;\ \left(X \mid f(X) = y\right)\right)\right] \leq \delta$$

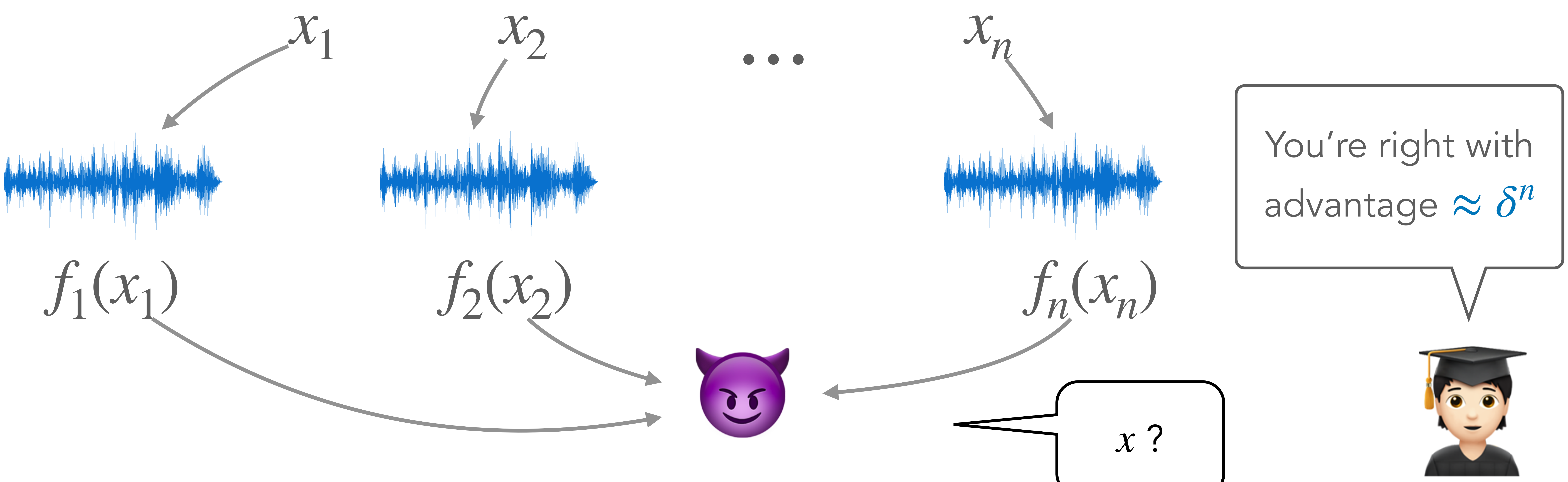
expectation on the possible leakage values

statistical distance between X and X and given $f(X) = y$

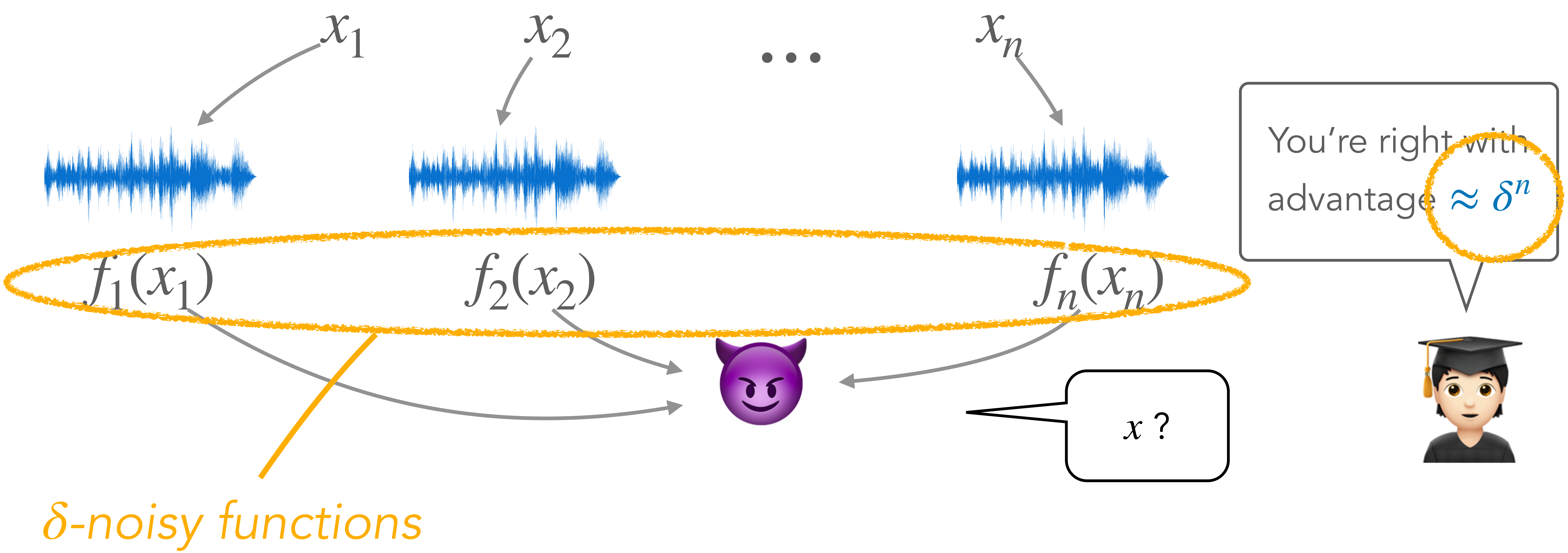
more noise $\Rightarrow$ smaller δ

$\begin{cases} 1 = \text{lot of leakage (low noise)} \\ 0 = \text{no leakage (infinite noise)} \end{cases}$

Soundness of masking with noise (generalised)

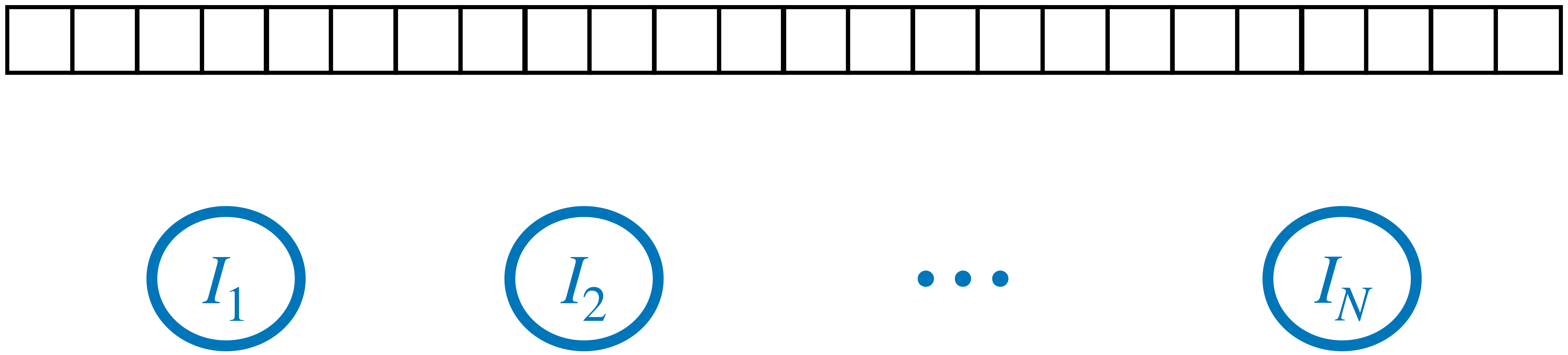


Soundness of masking with noise (generalised)

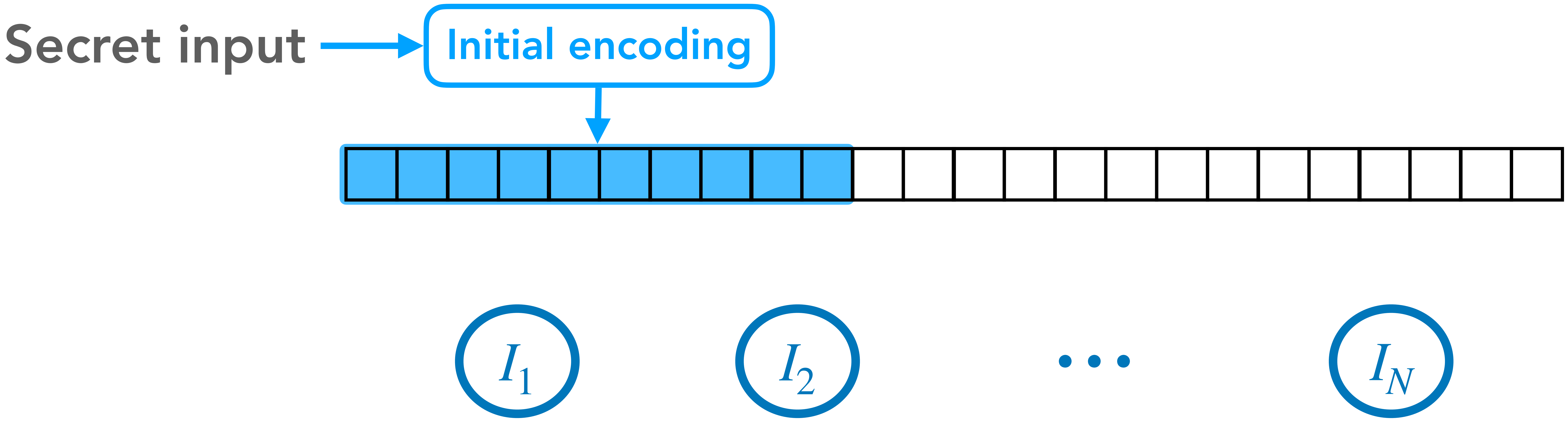


Simulation security

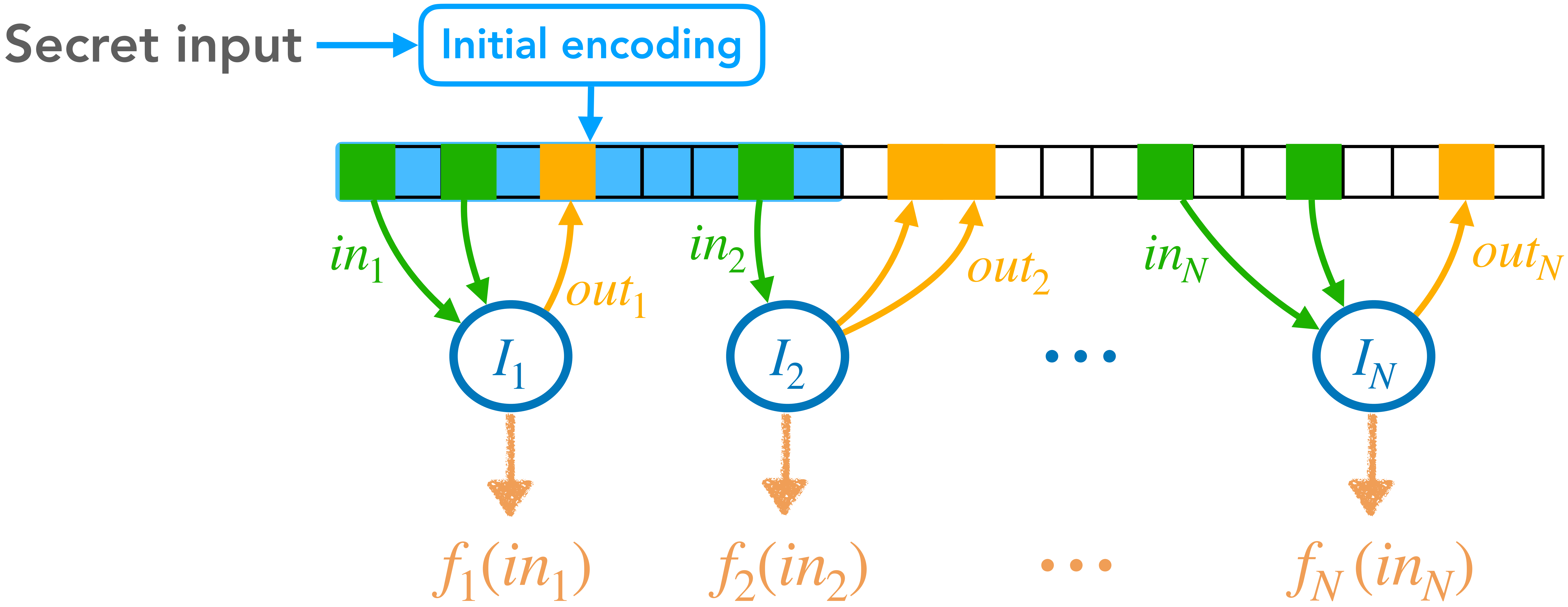
Secret input



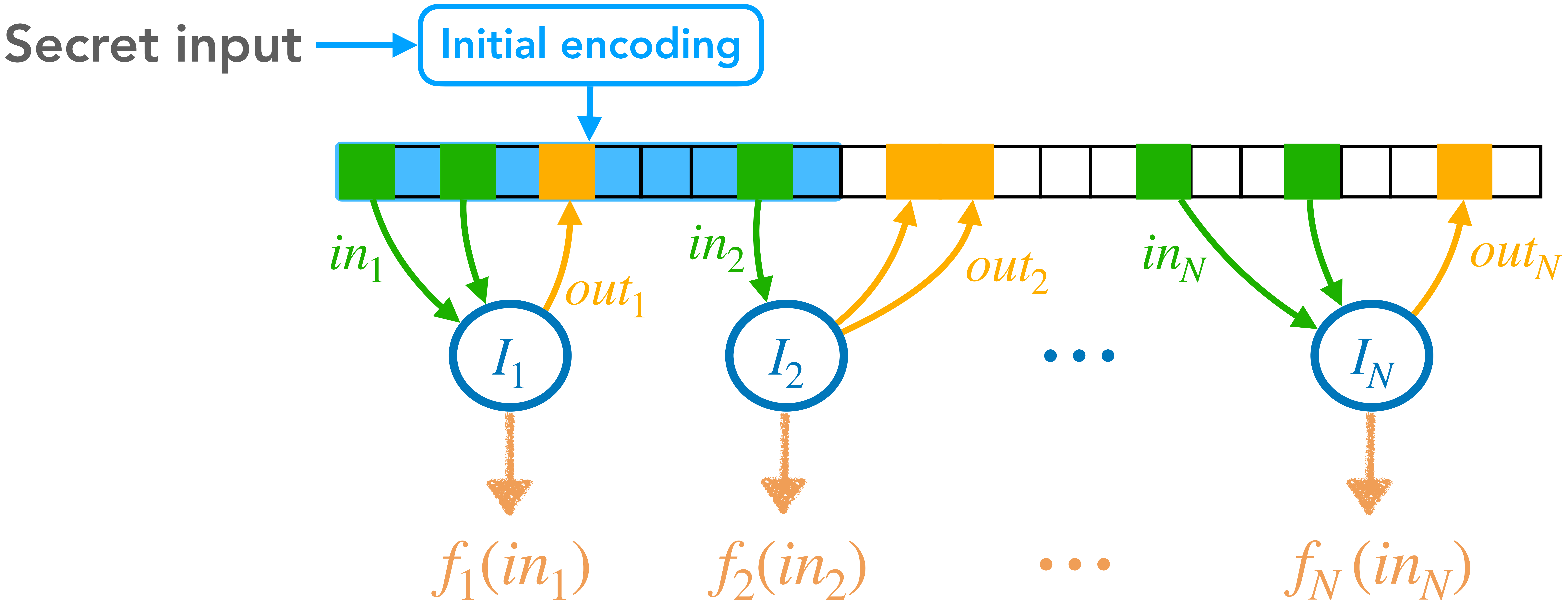
Simulation security



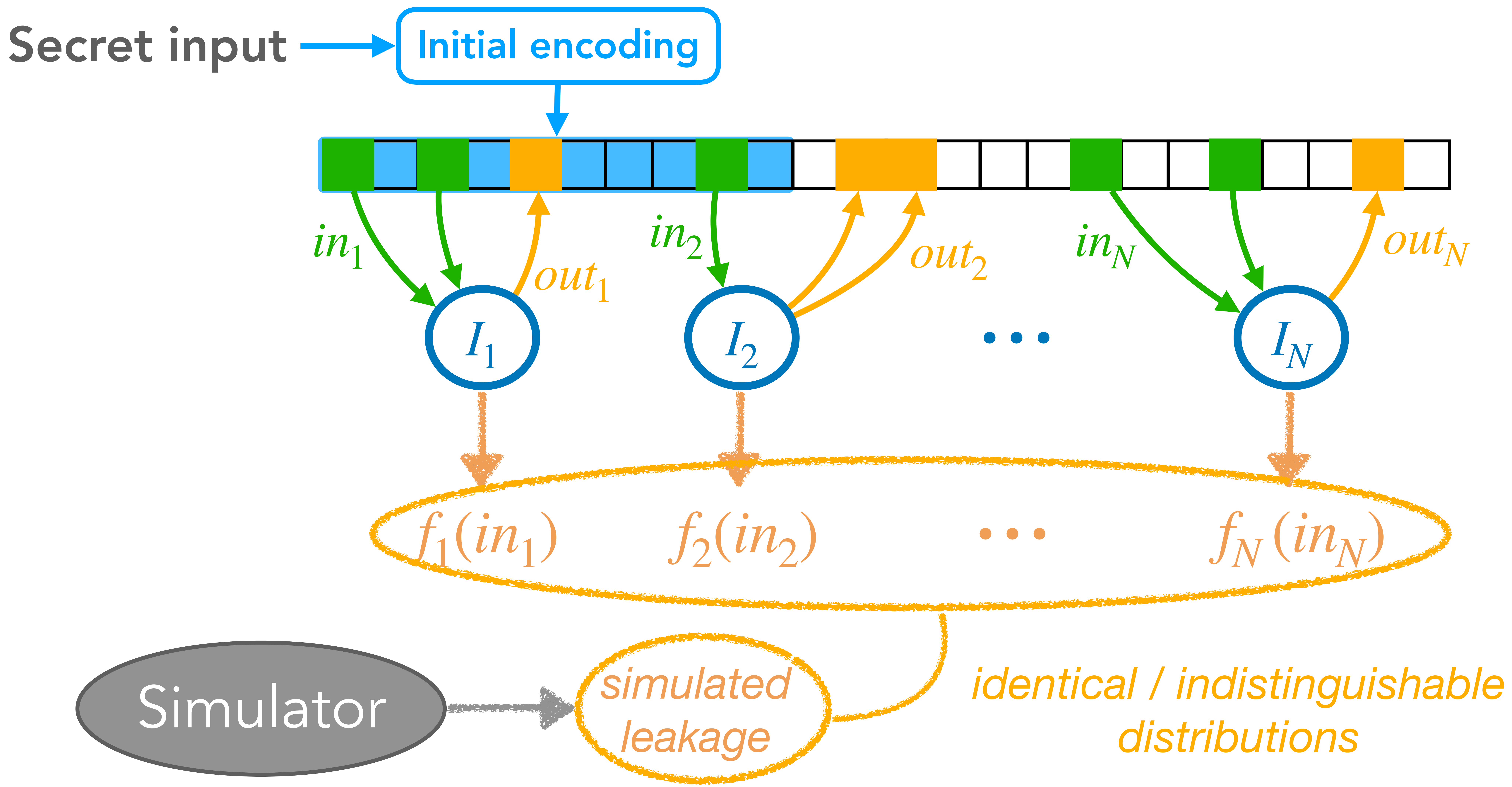
Simulation security



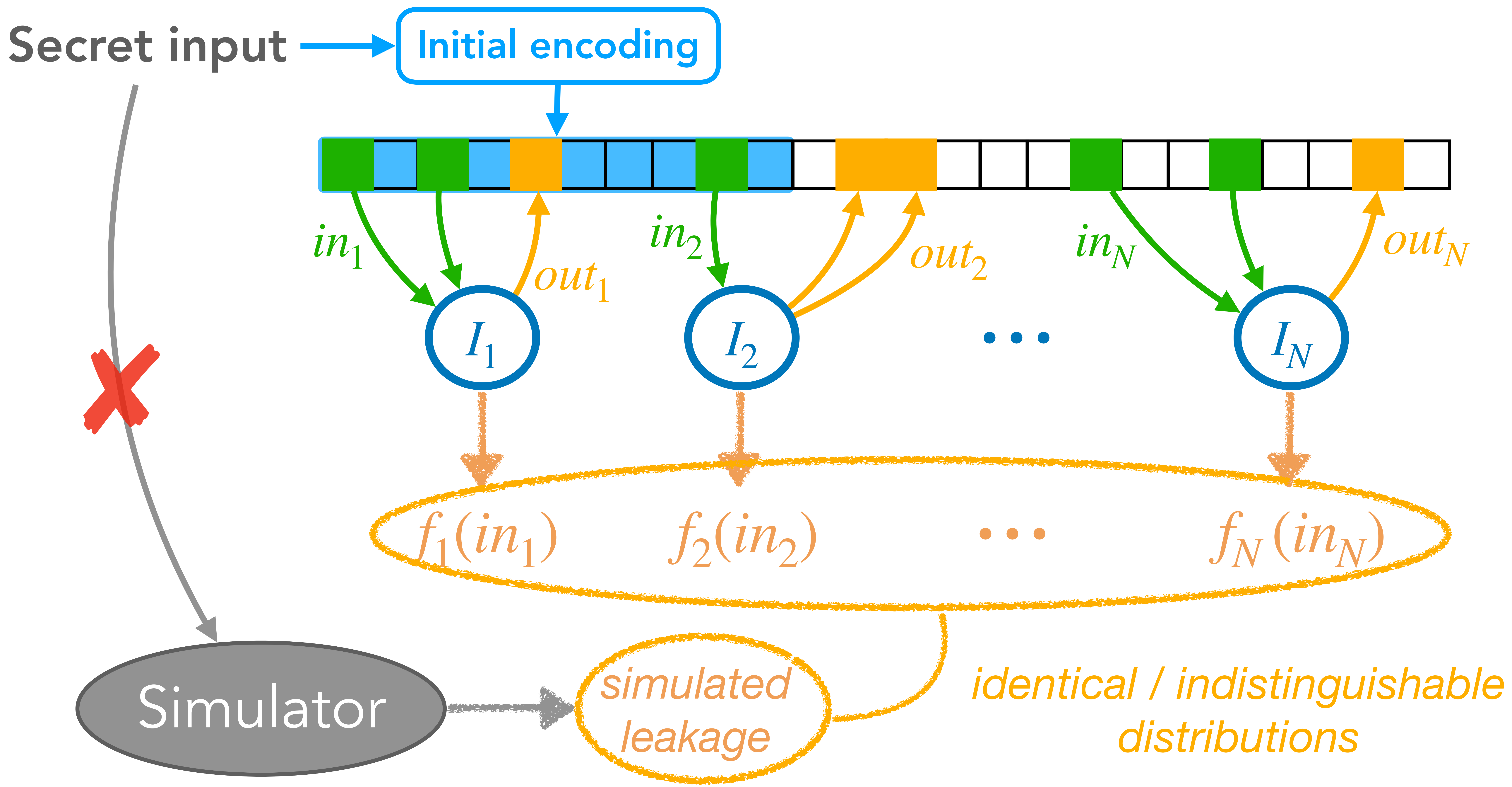
Simulation security



Simulation security



Simulation security



Masked computation

Masking

sharing

the shares

$$x \longrightarrow \vec{x} = (x_1, \ldots, x_n)$$

$$x = x_1 + \cdots + x_n$$

(on a field $\mathbb{K}$)

number of shares

Masking

$$x \longrightarrow \vec{x} = (x_1, \ldots, x_n)$$

sharing — the shares

$$x = x_1 + \cdots + x_n$$

(on a field $\mathbb{K}$)

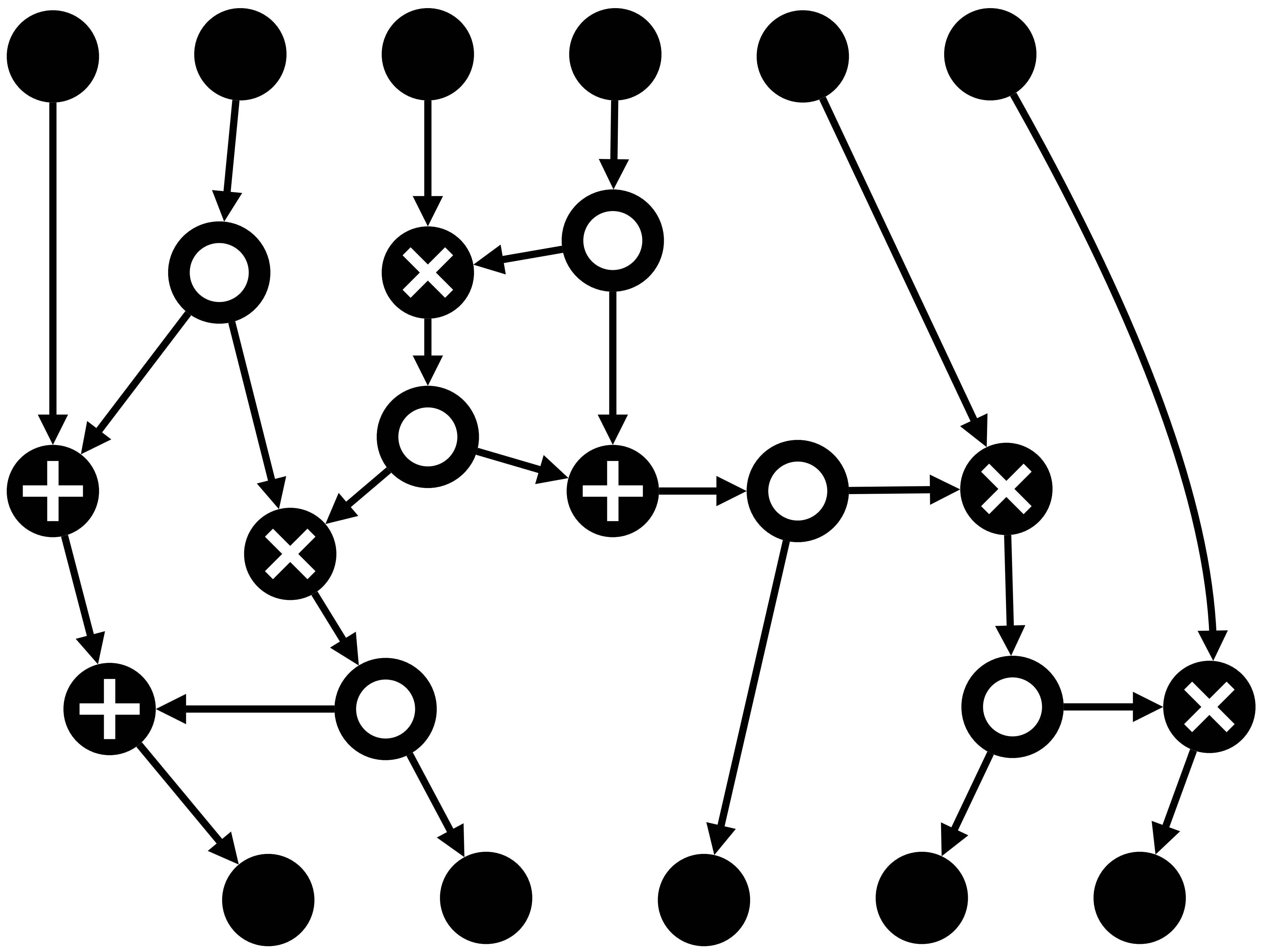
number of shares

⚠️ all the shares are necessary to recover x

🎲 any $n - 1$ shares are completely random

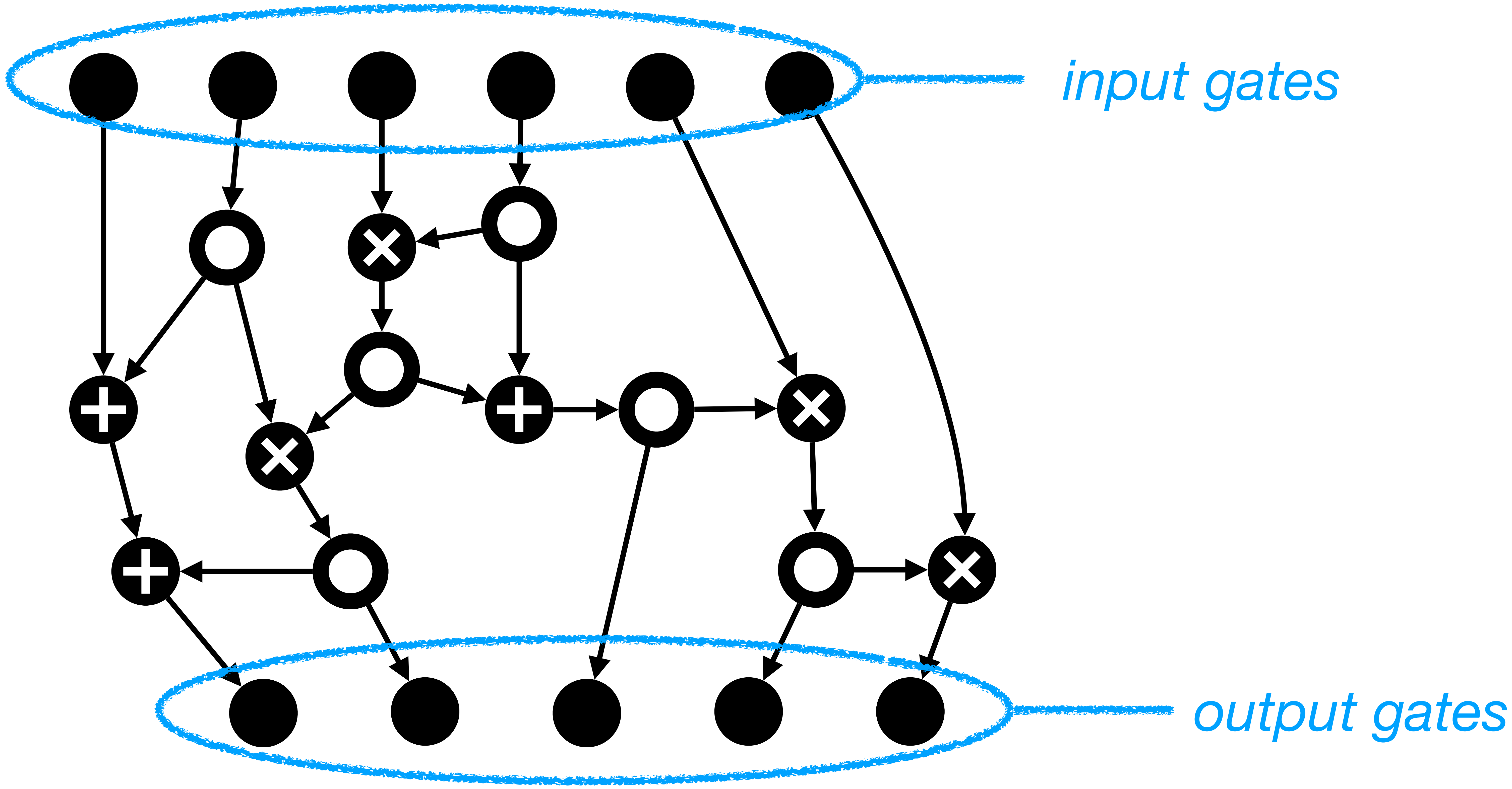
Circuit model

Crypto computation modelled as an arithmetic circuit on $\mathbb{K}$



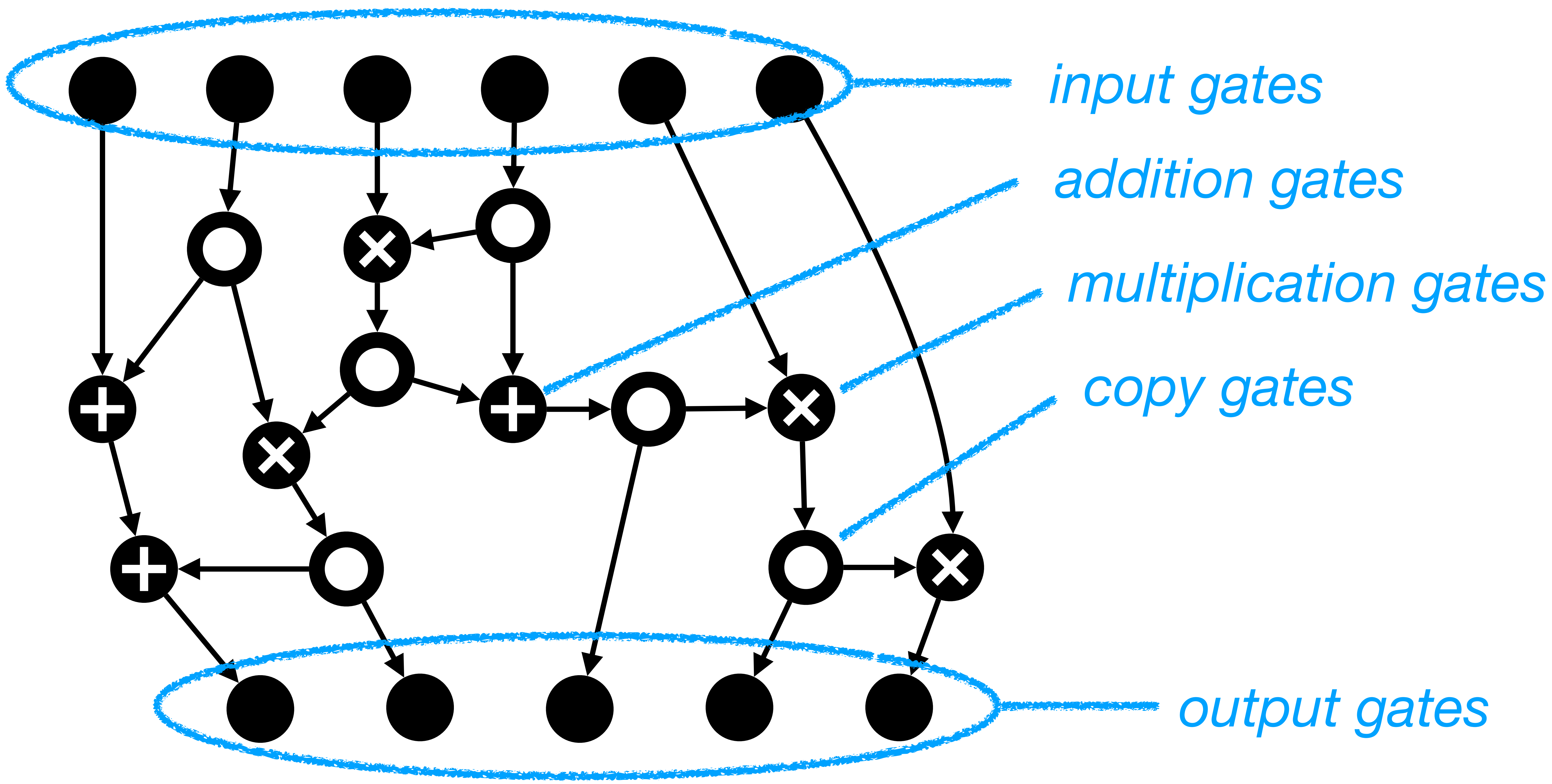
Circuit model

Crypto computation modelled as an arithmetic circuit on $\mathbb{K}$



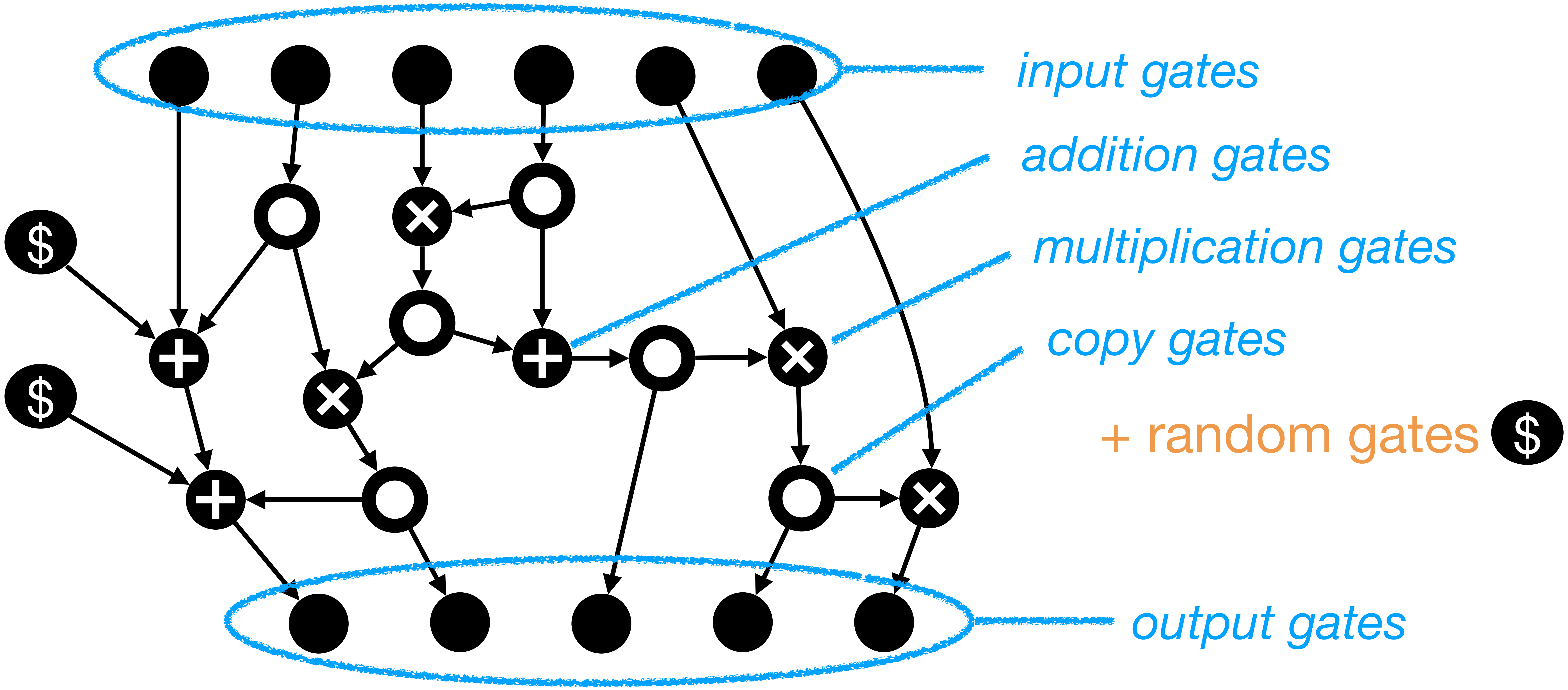
Circuit model

Crypto computation modelled as an arithmetic circuit on $\mathbb{K}$



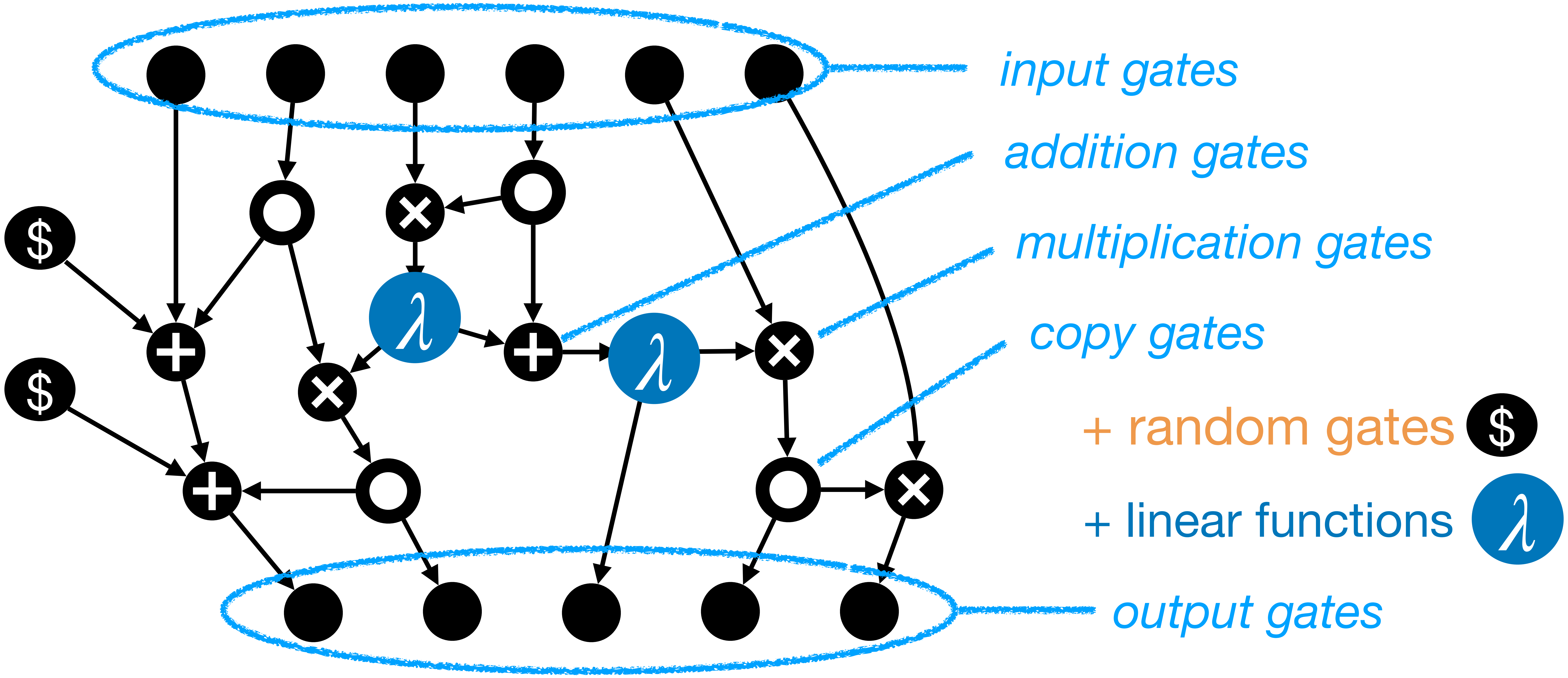
Circuit model

Crypto computation modelled as an arithmetic circuit on $\mathbb{K}$

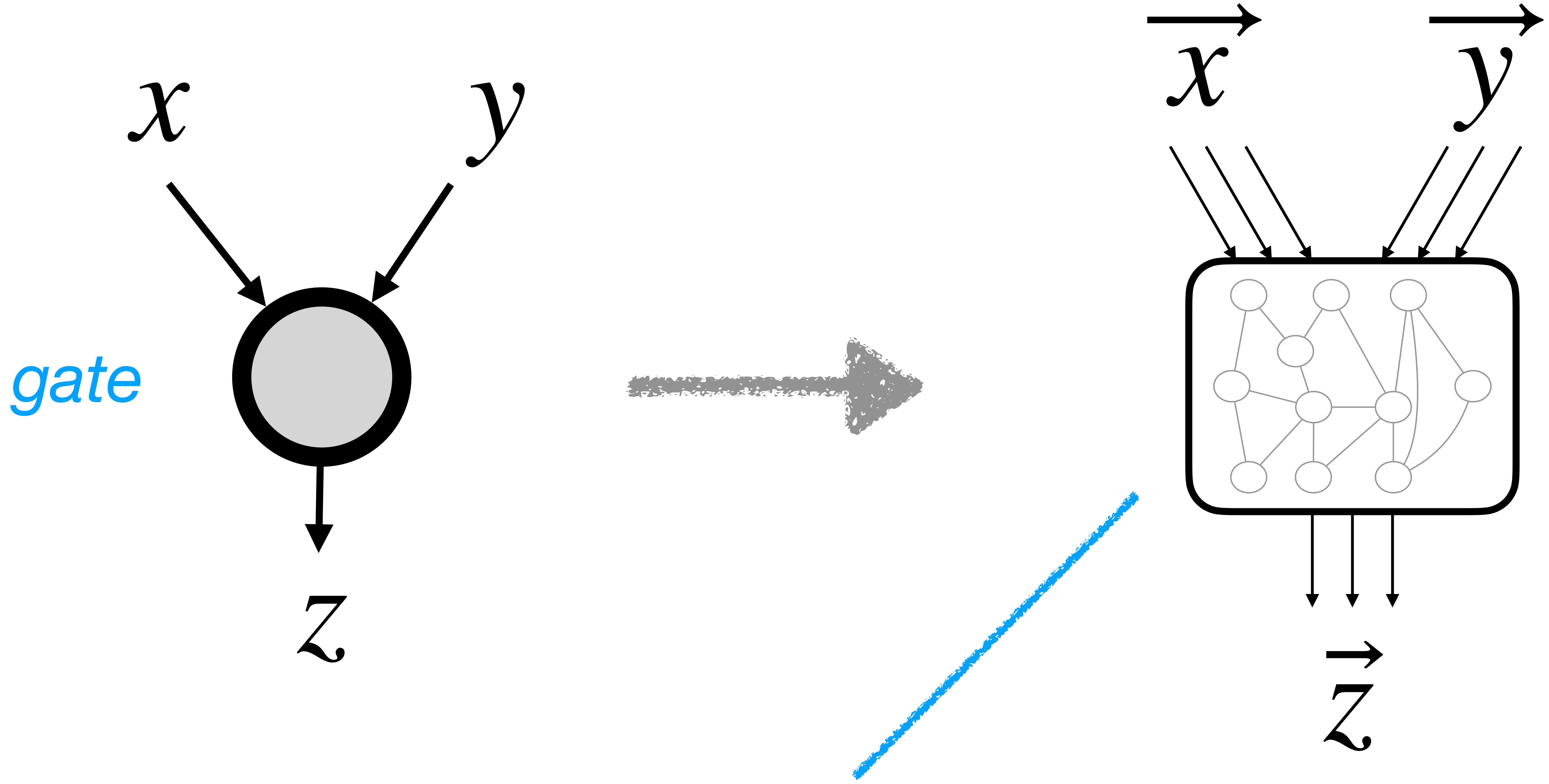


Circuit model

Crypto computation modelled as an arithmetic circuit on $\mathbb{K}$

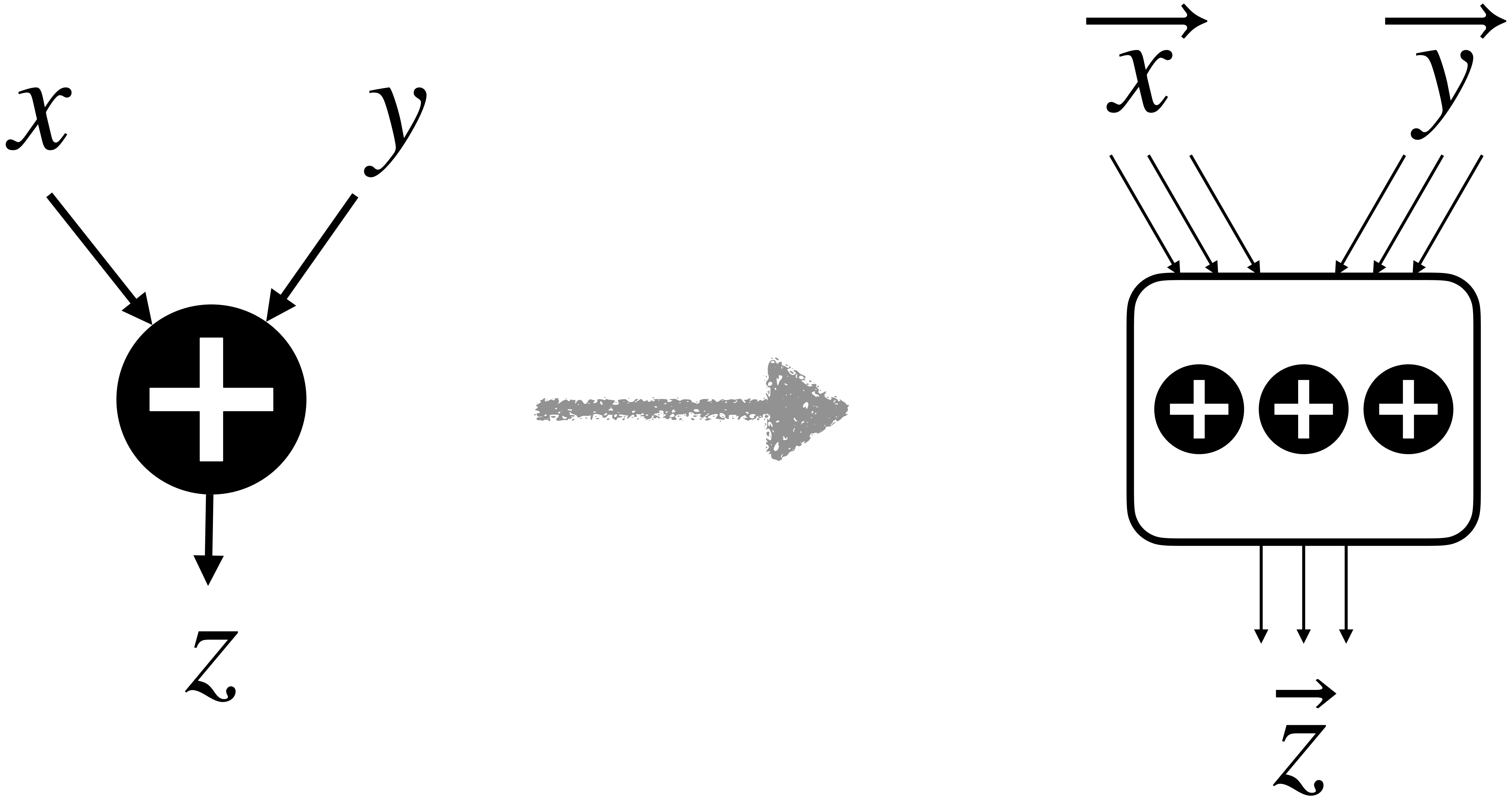


Gadgets



gadget : small circuit computing an operation on sharings

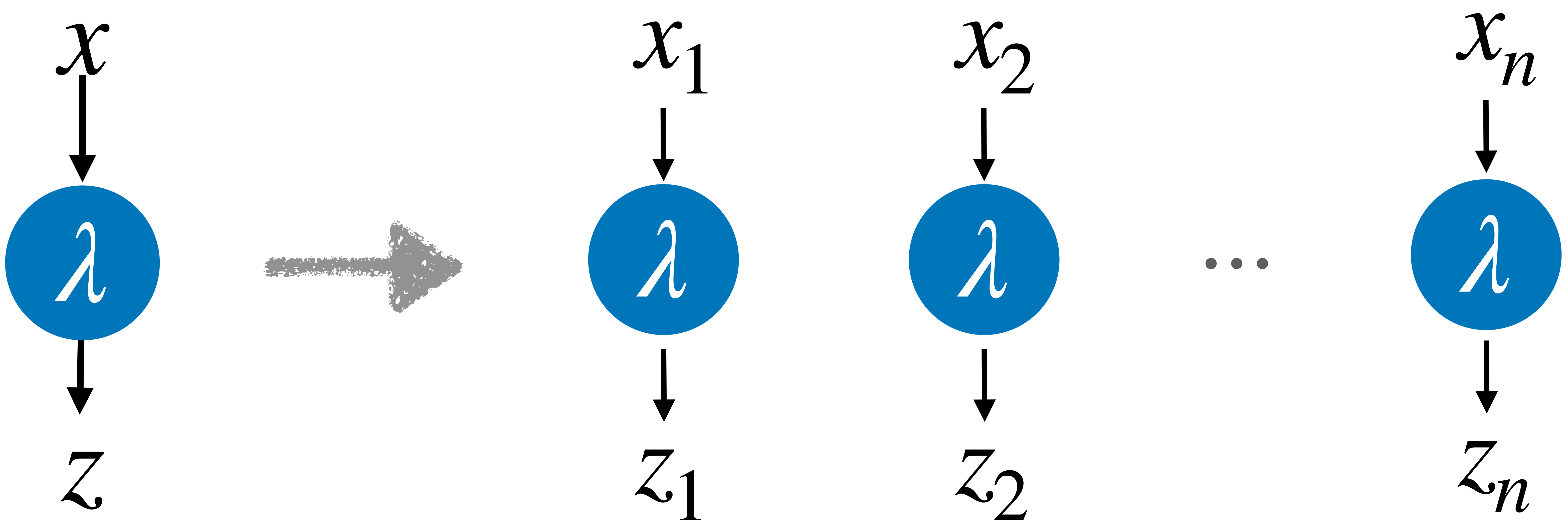
Adition gadget



sharewise computation

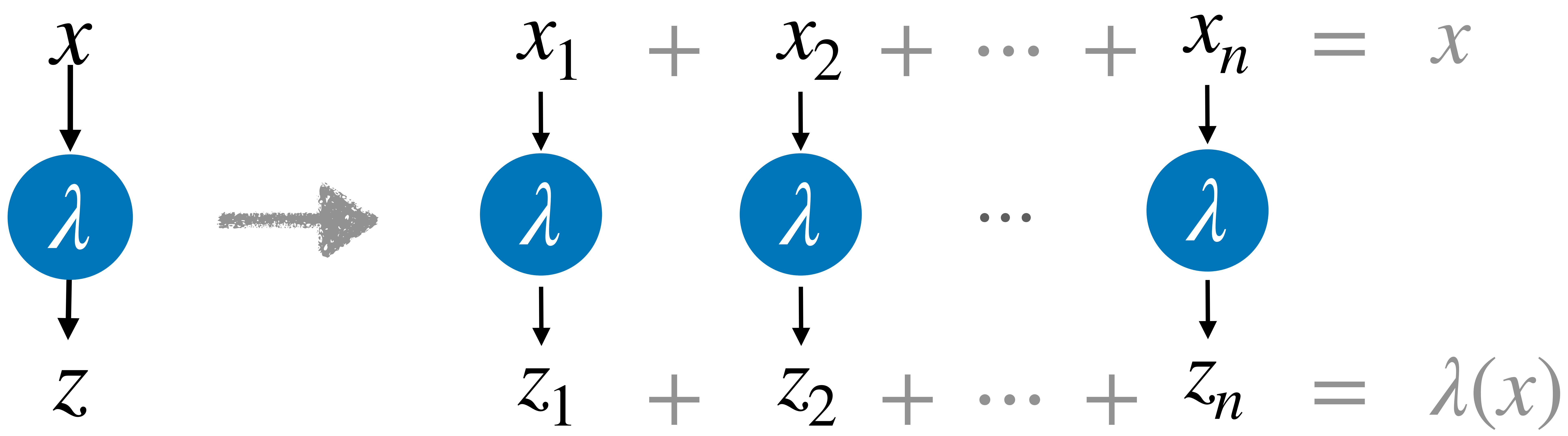
⇒ n addition gates

Linear gadget



sharewise computation $\Rightarrow$ n *evaluations of* λ

Linear gadget



sharewise computation $\Rightarrow$ *n evaluations of λ*

Multiplication gadget

$$z = x \cdot y = \left(\sum_i x_i\right)\left(\sum_i y_i\right) = \sum_{i,j} x_i y_j$$

Multiplication gadget

$$z = x \cdot y = \left(\sum_i x_i\right)\left(\sum_i y_i\right) = \sum_{i,j} x_i y_j$$

split into n shares

$$z_1$$

$$z_2$$

$$\vdots$$

$$z_n$$

Multiplication gadget

$$z = x \cdot y = \left(\sum_i x_i\right)\left(\sum_i y_i\right) = \sum_{i,j} x_i y_j$$

split into n shares

z_1

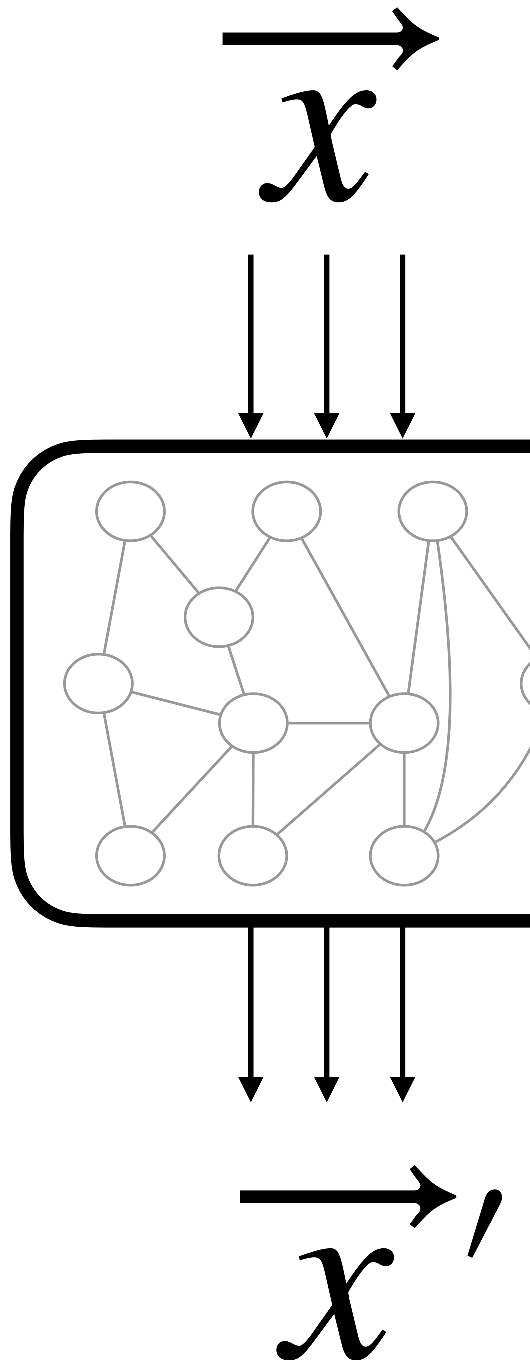
z_2

$\vdots$

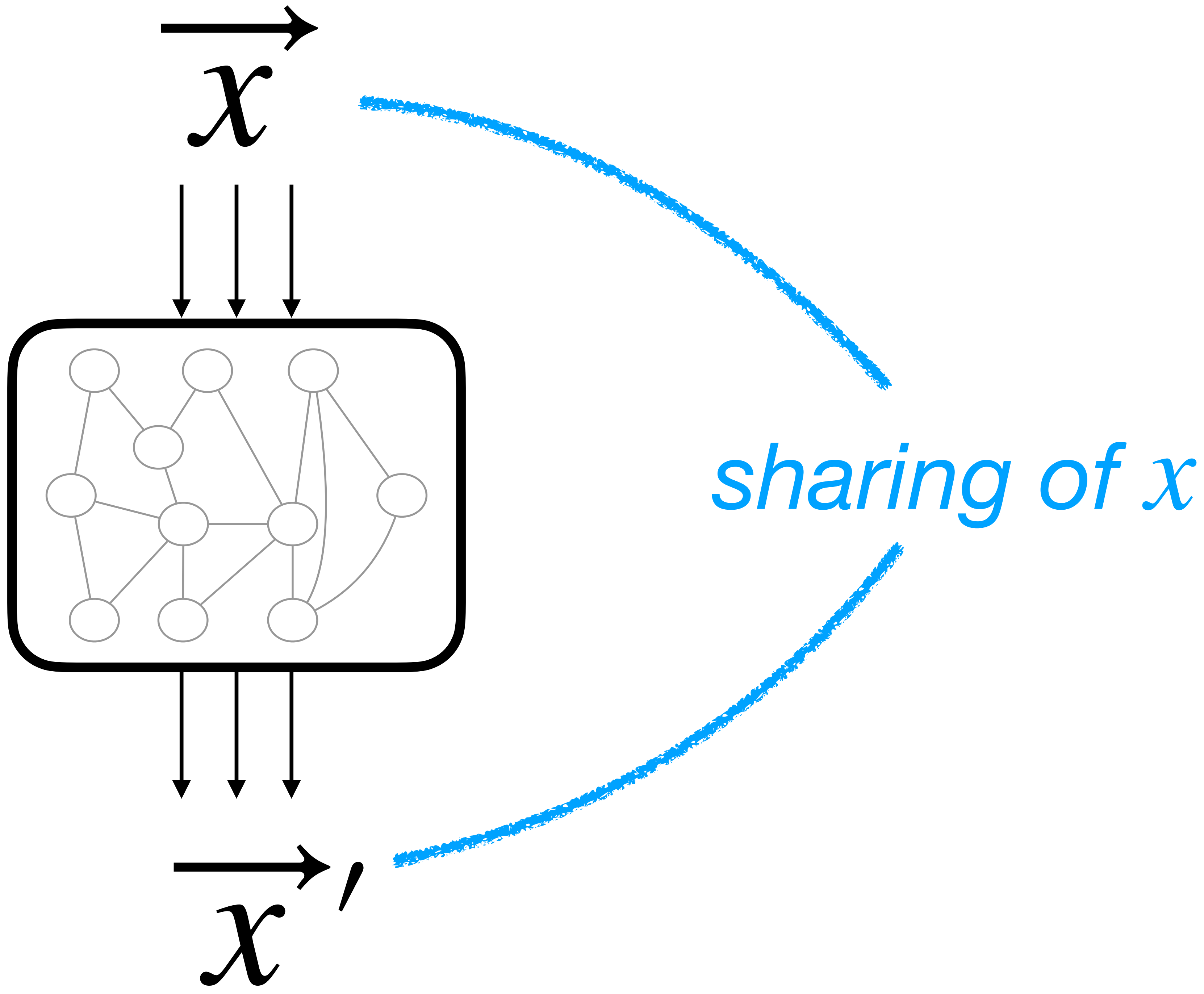
z_n

+ fresh randomness

Refresh gadget



Refresh gadget

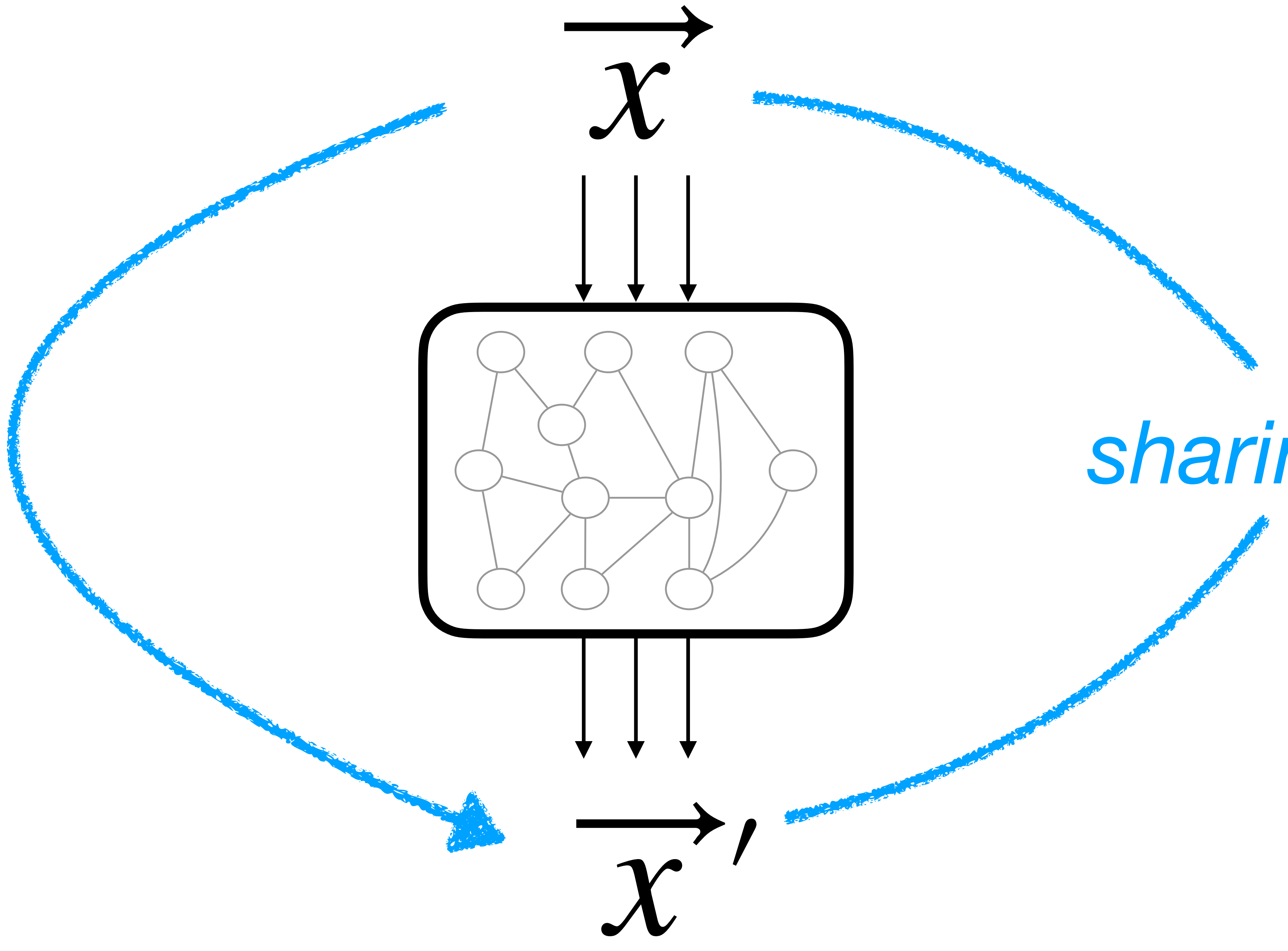


Refresh gadget

fresh randomness

$\Rightarrow$

statistical independence

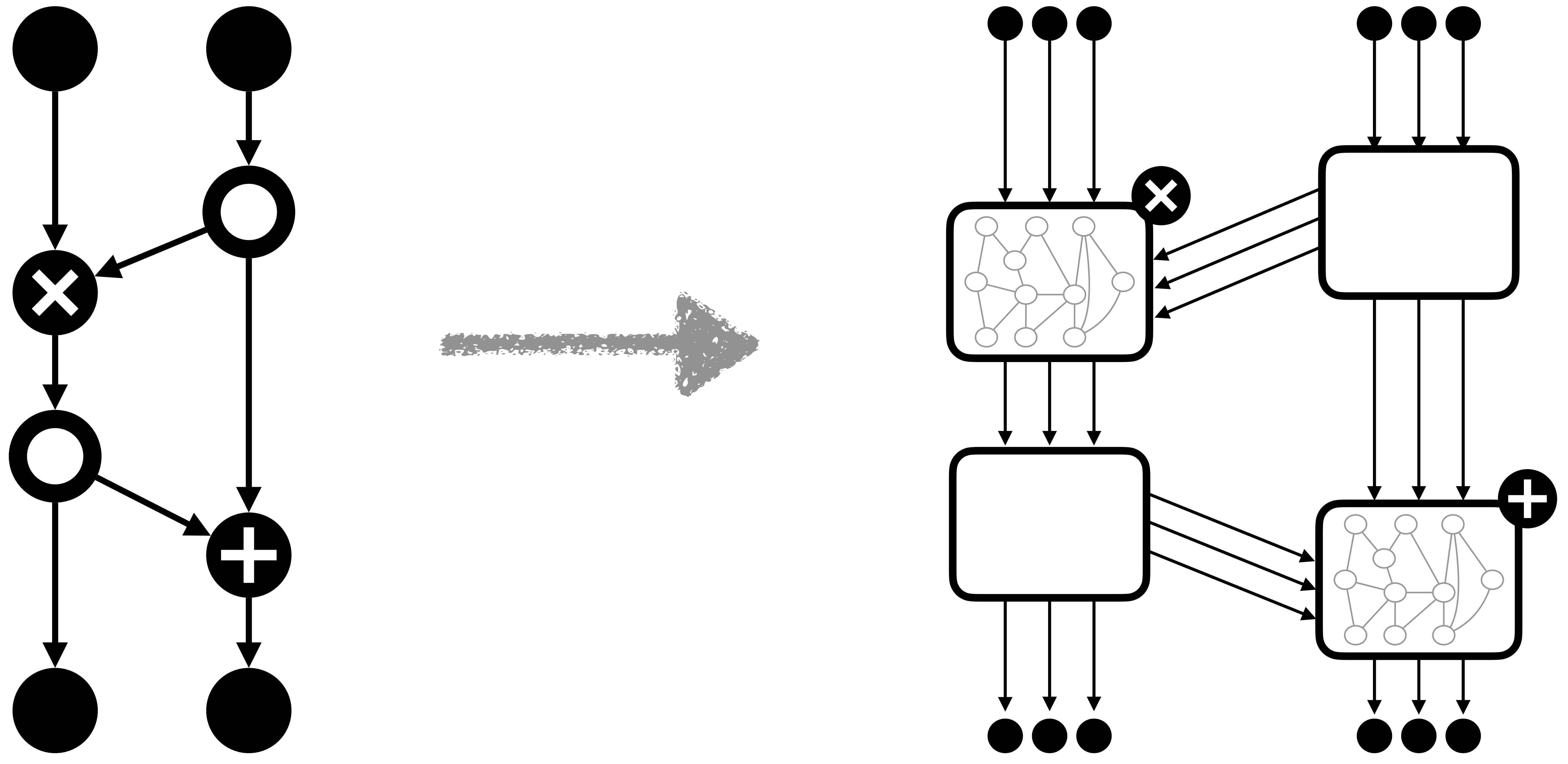


sharing of x

Standard circuit compiler

wire $\to$ n wires (sharing)

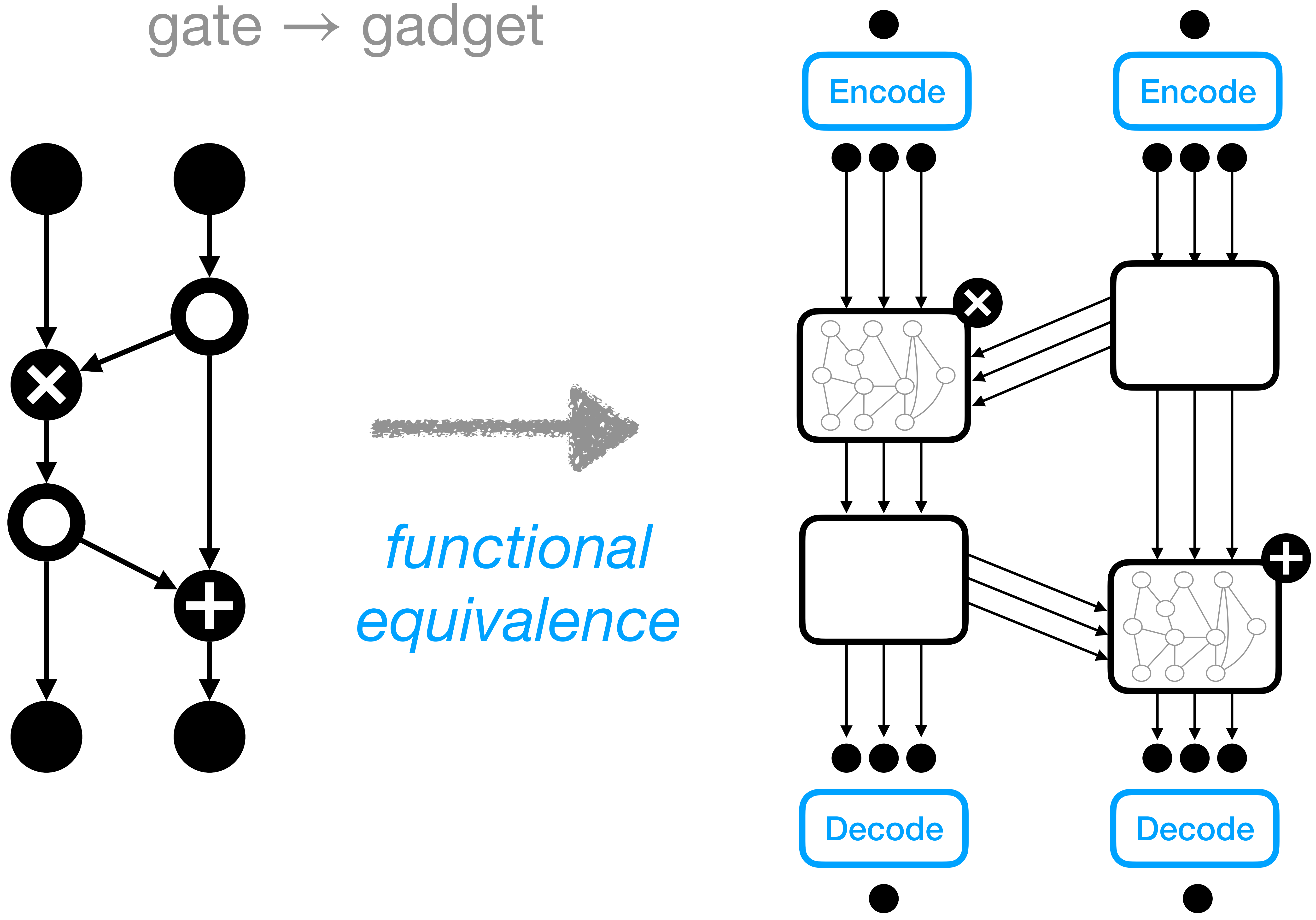
gate $\to$ gadget



Standard circuit compiler

wire $\to n$ wires (sharing)

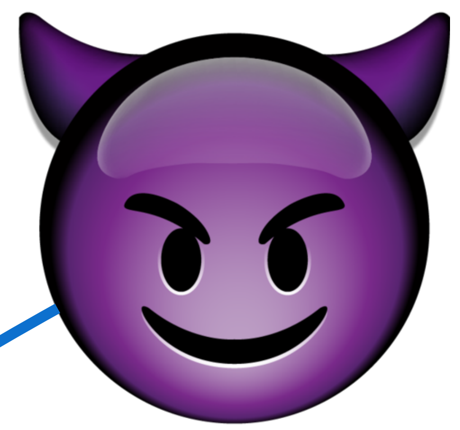
gate $\to$ gadget

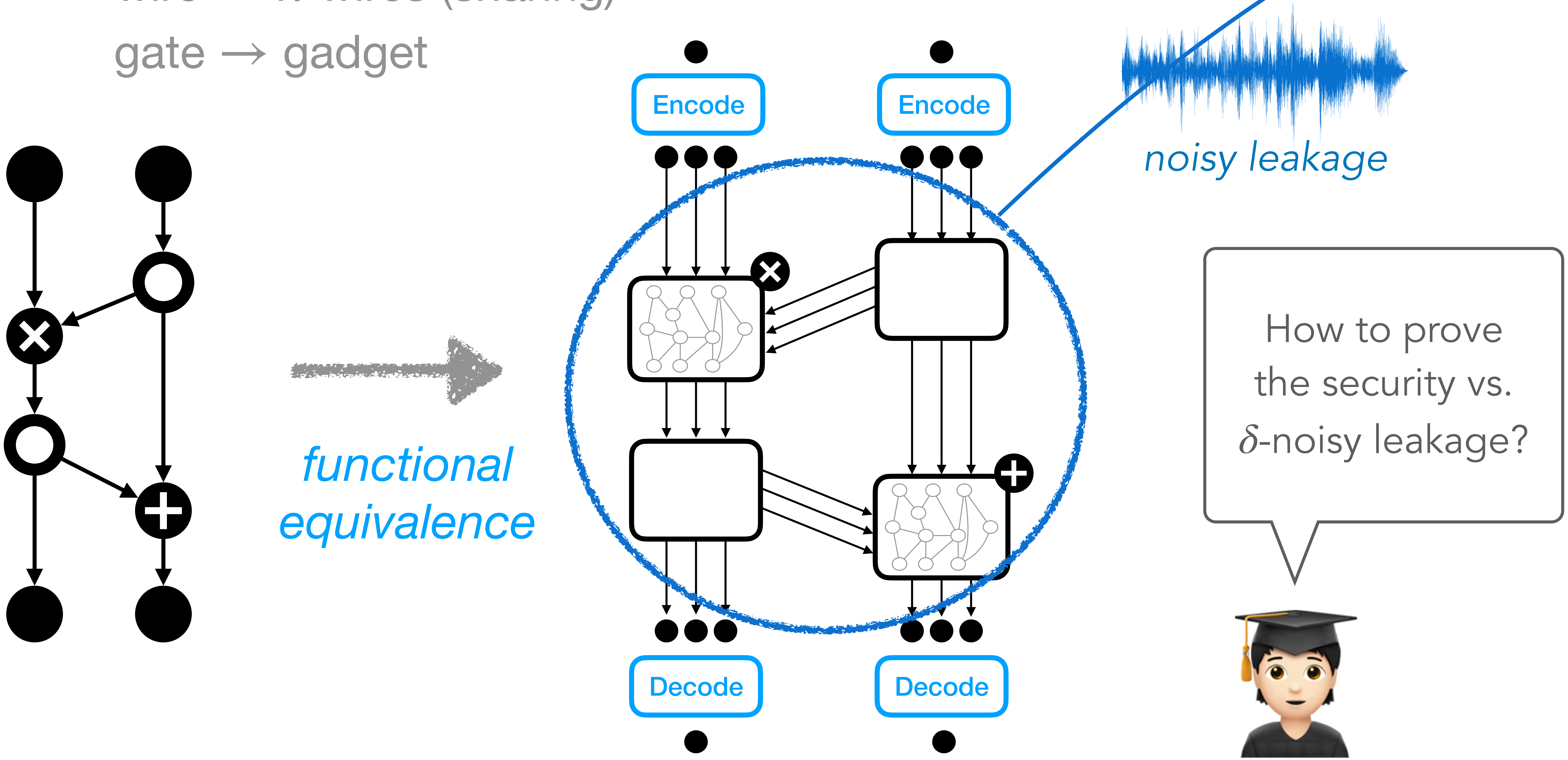


Standard circuit compiler

wire $\rightarrow$ n wires (sharing)

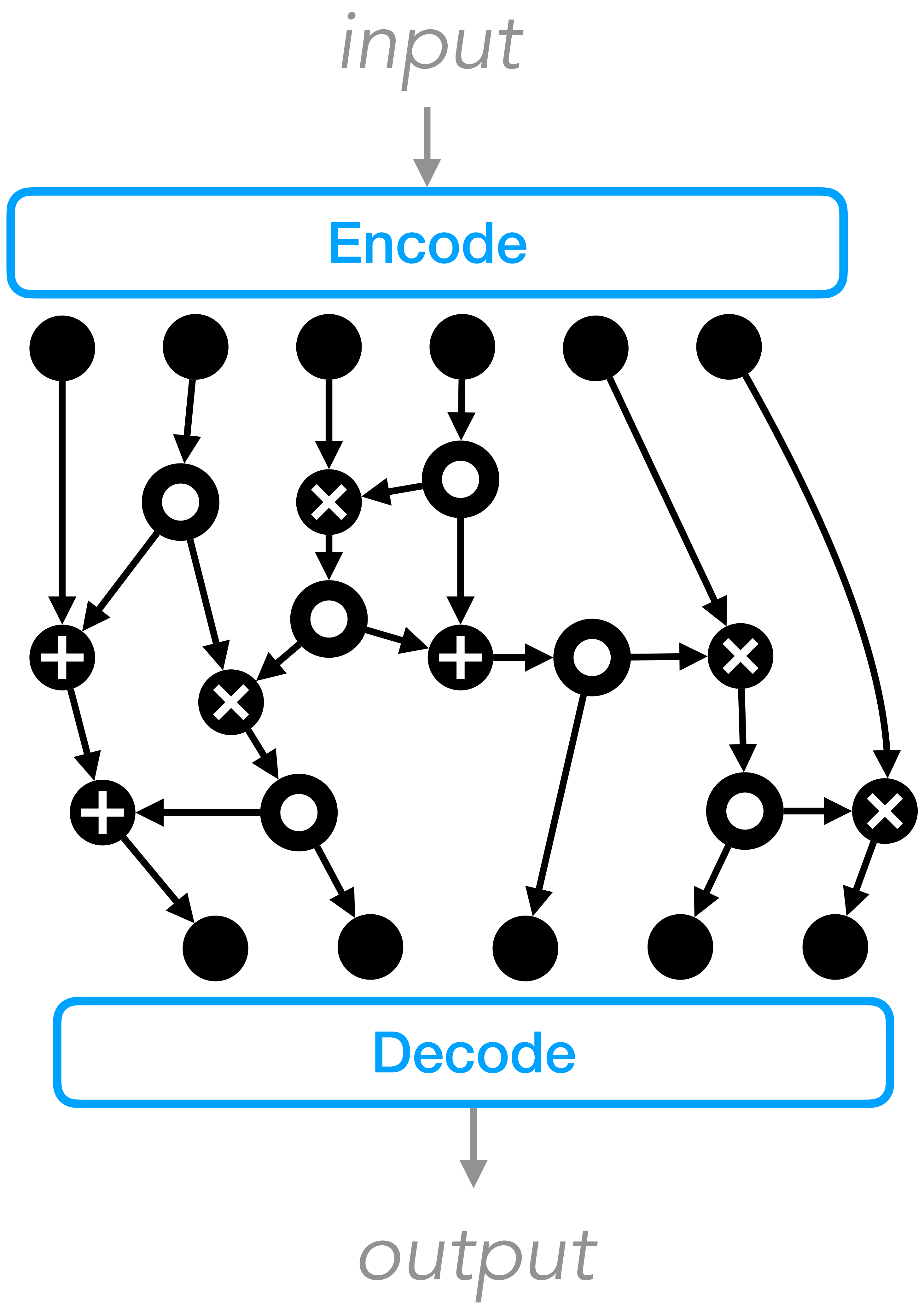
gate $\rightarrow$ gadget



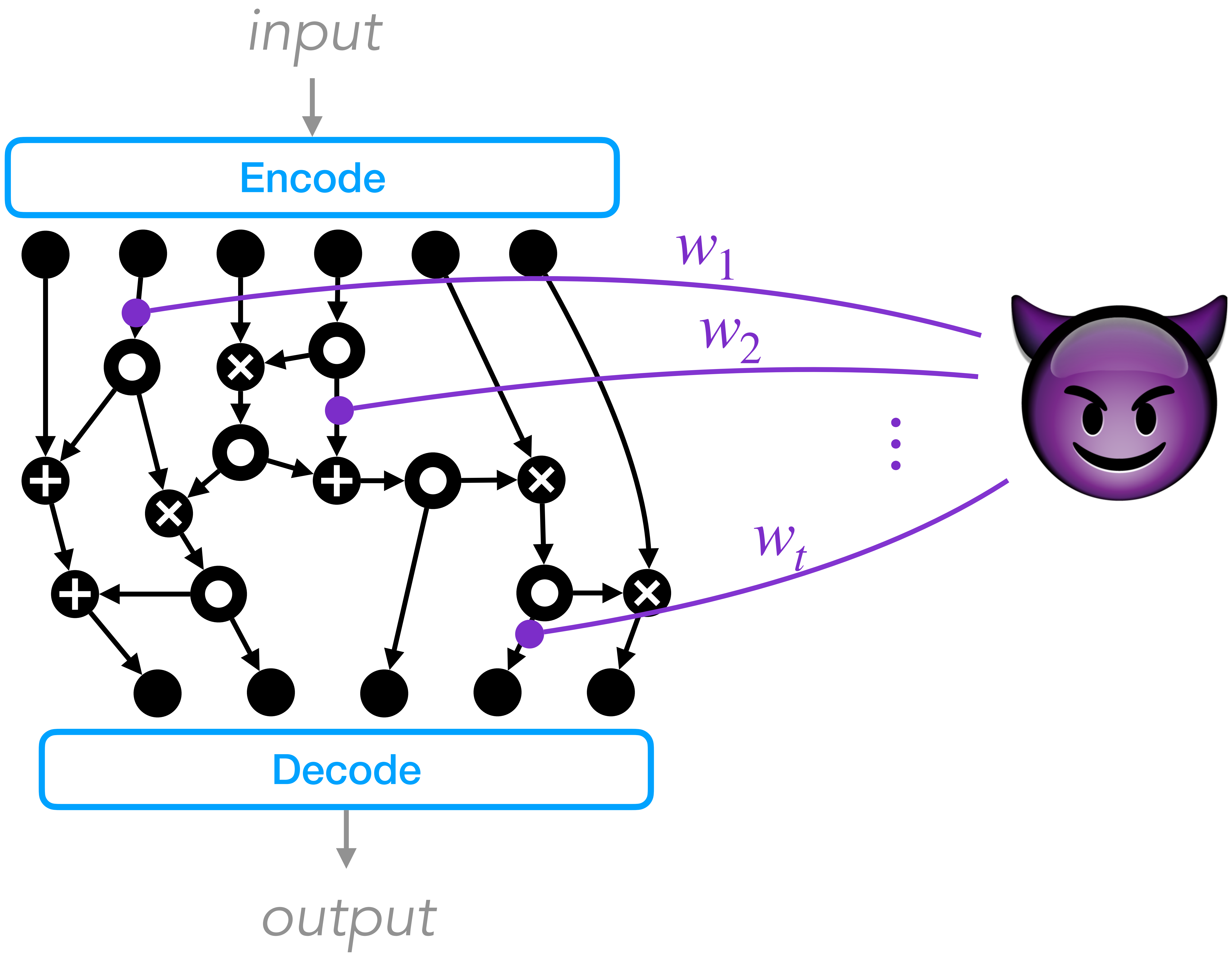


Probing models

Probing model

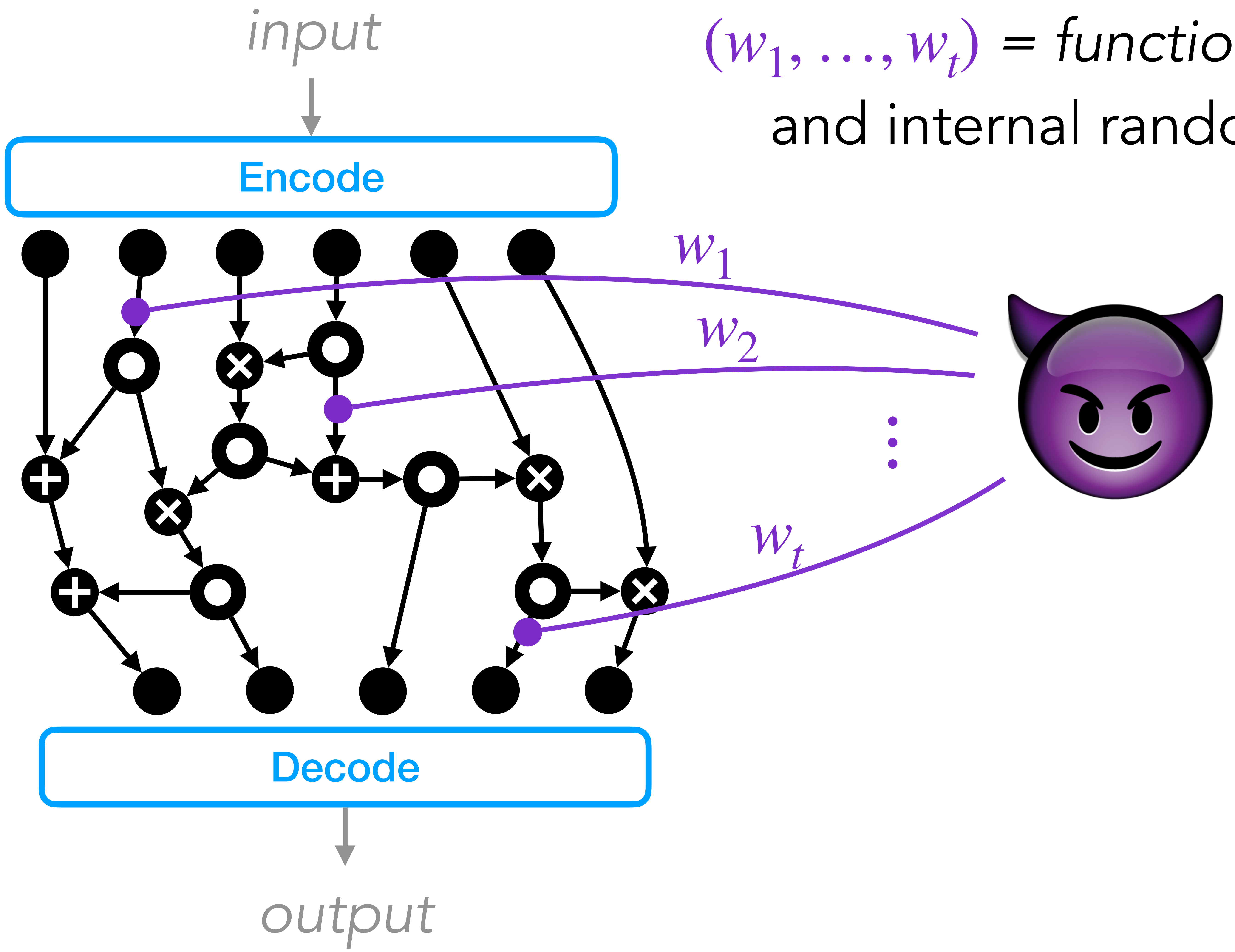


Probing model

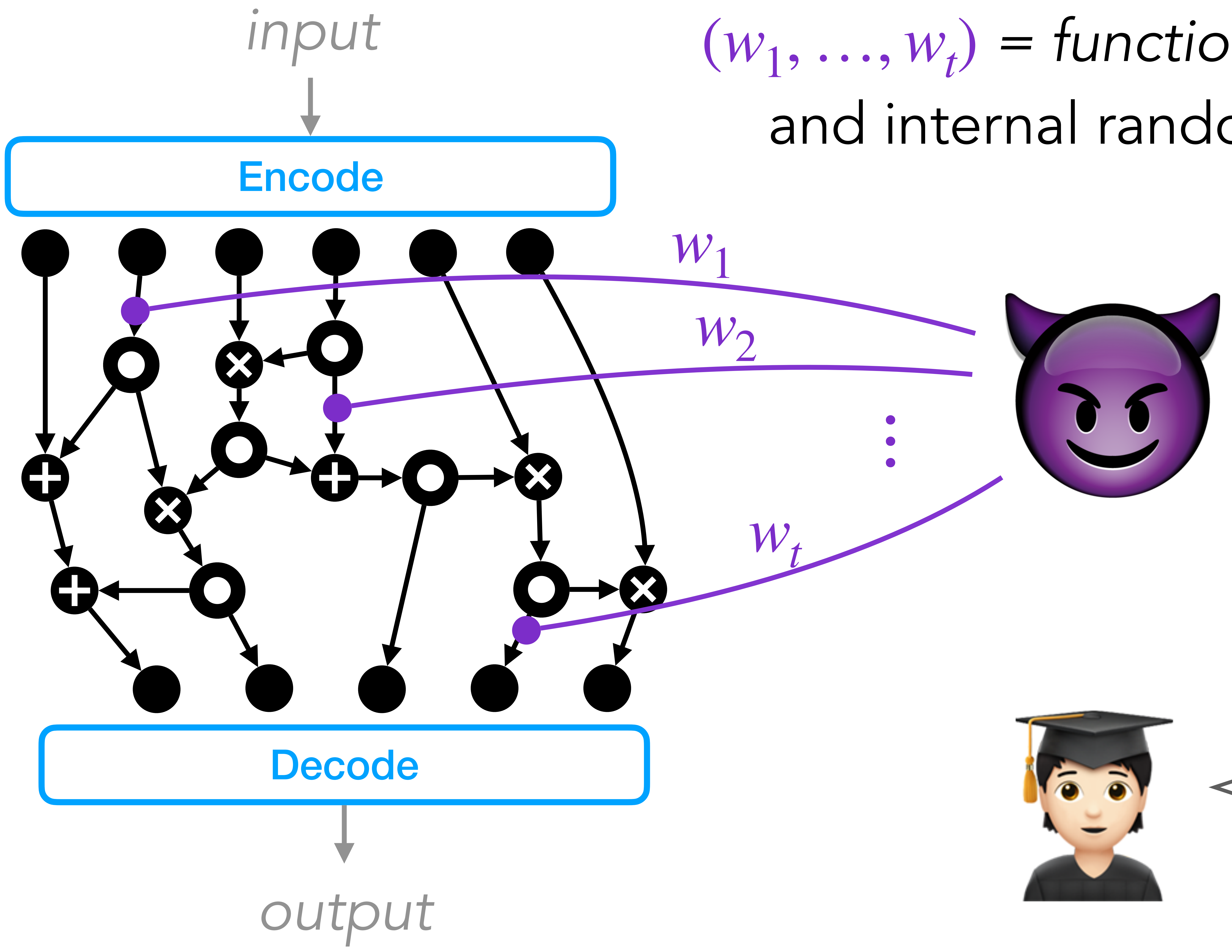


Probing model

$(w_1, \ldots, w_t)$ = *function of input*
and internal randomness



Probing model



$(w_1, \ldots, w_t)$ = *function of* *input* and internal randomness

t-probing security:

$(w_1, \ldots, w_t)$ *can be perfectly simulated w/o any knowledge about the input*

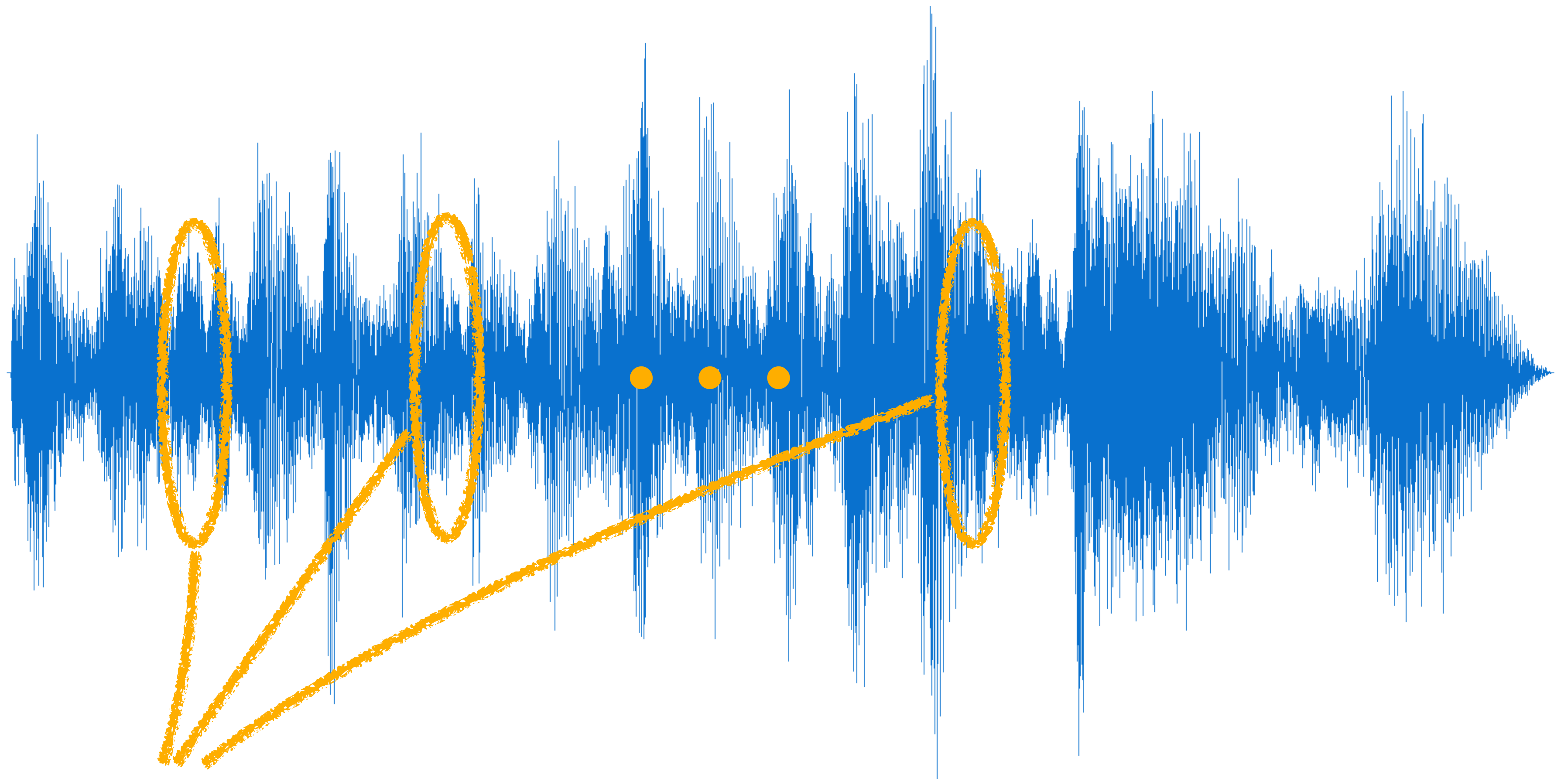
Probing model

t-probing security

$\Leftrightarrow$

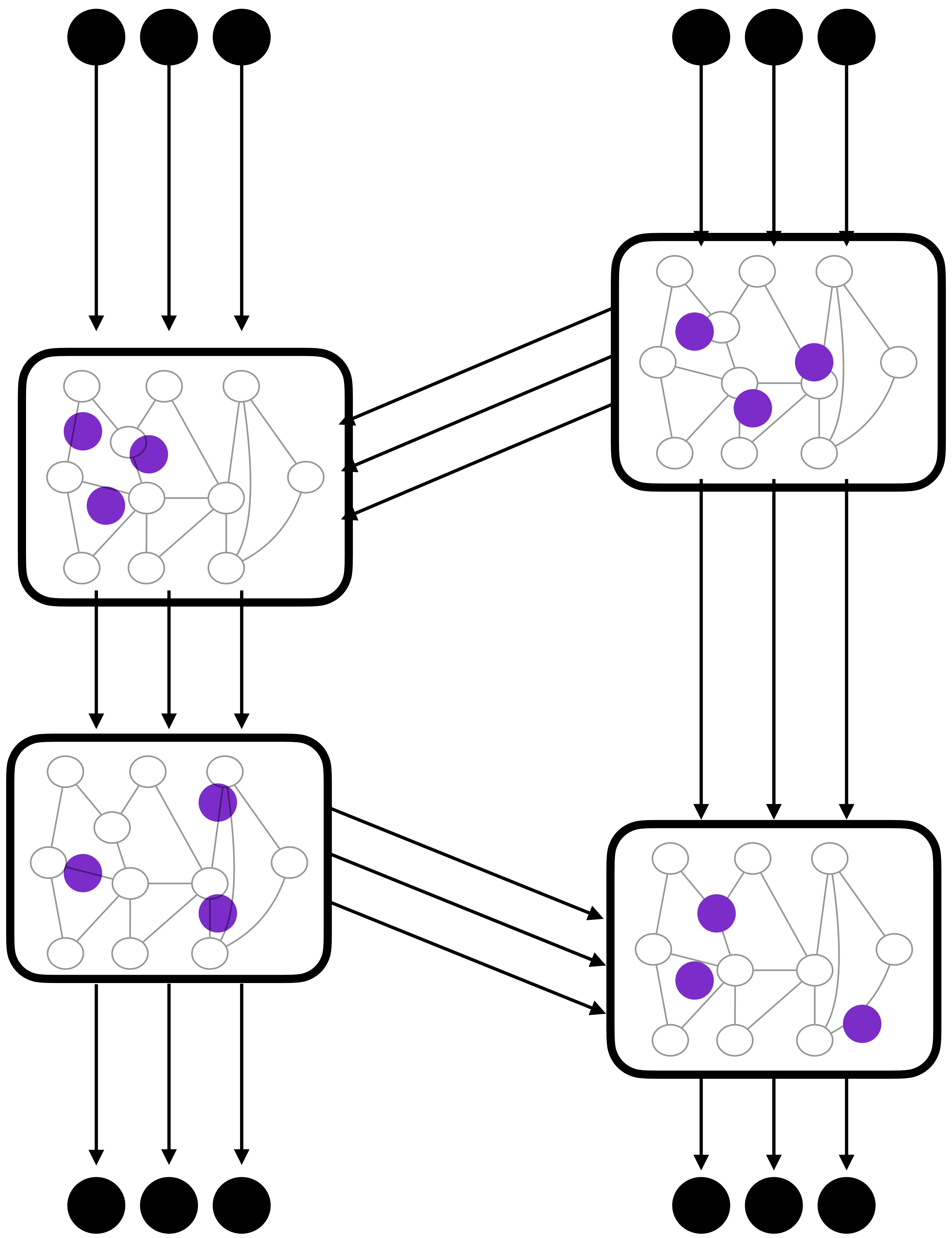
security against

t-order DPA



any t leakage points
independent of the secrets

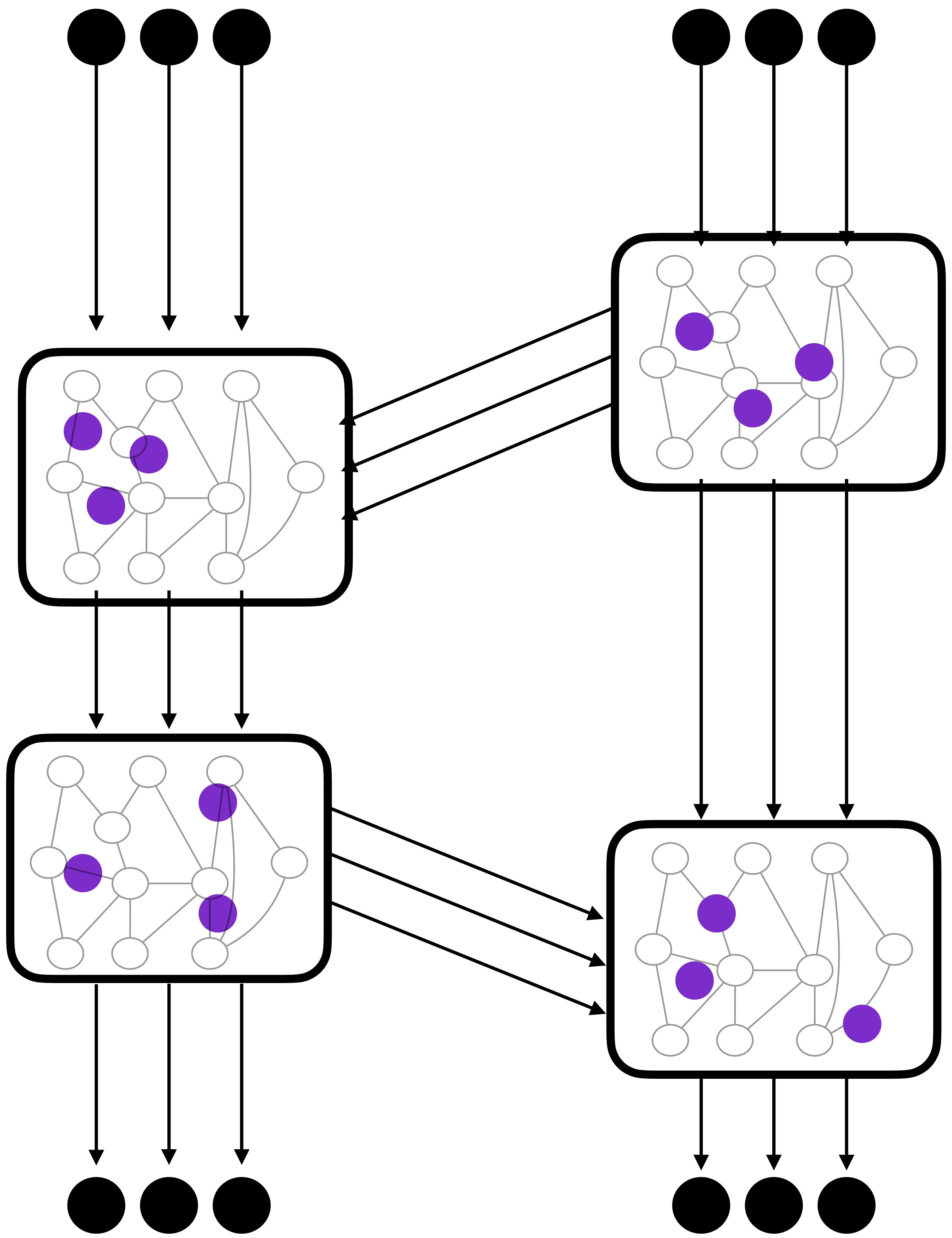
Region probing model



t probes per gadget (or region)

with $t = r \times |G|$

Region probing model



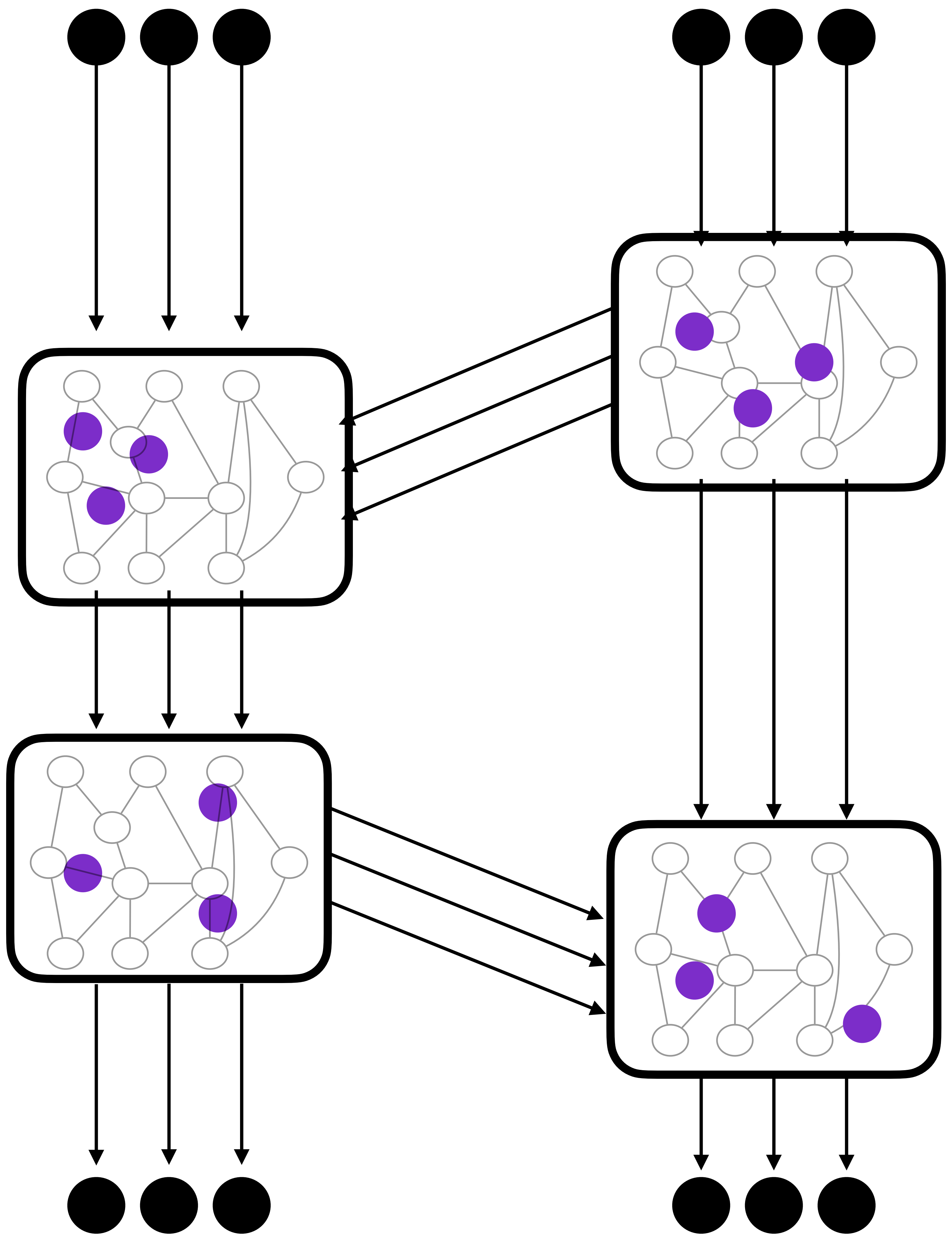
t probes per gadget (or region)

with $t = r \times |G|$

probing rate

number of wires in G

Region probing model



t probes per gadget (or region)

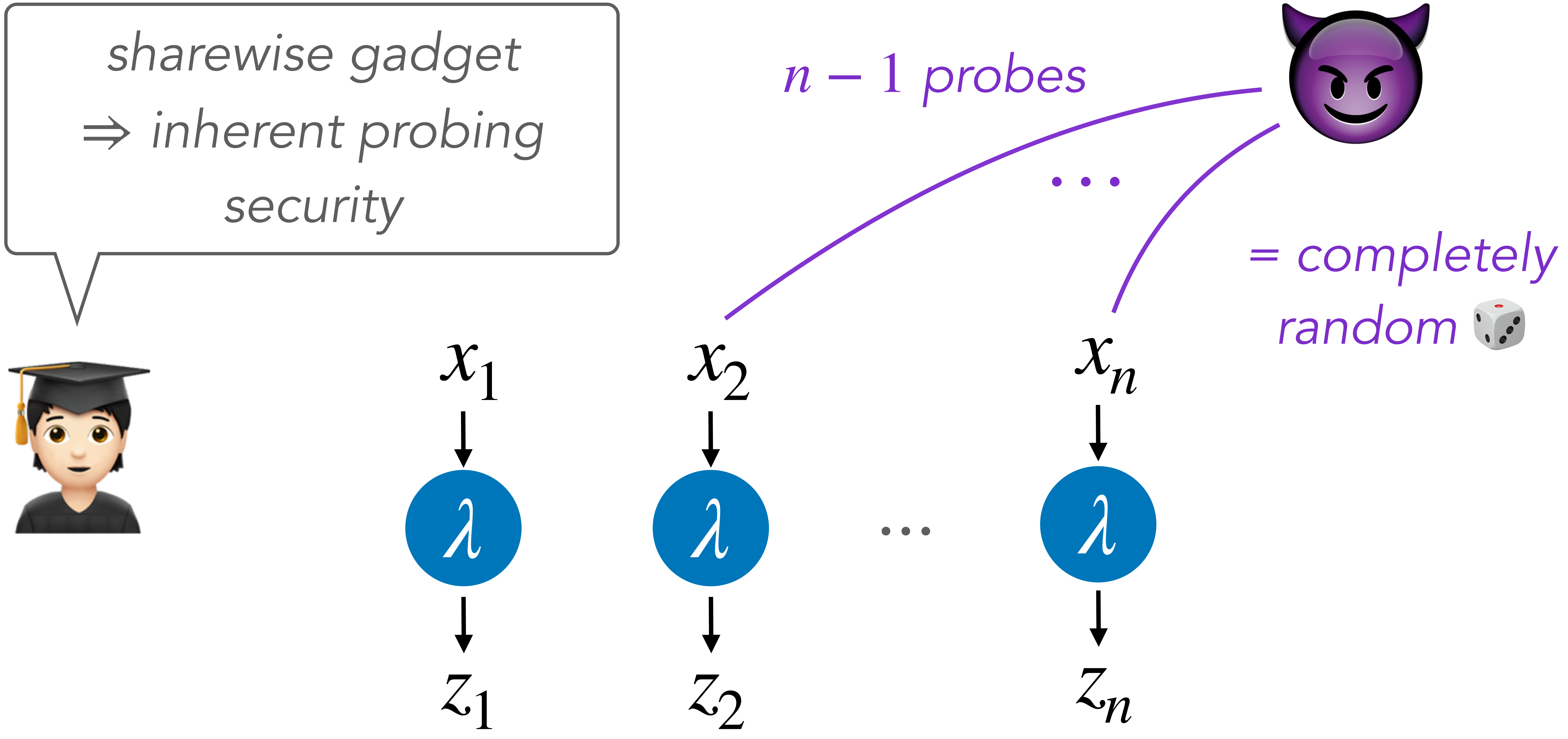
with $t = r \times |G|$

probing rate

number of wires in G

$\Rightarrow$ r-region probing security

Security of sharewise gadgets



ISW multiplication gadget

$$\begin{pmatrix} x_1y_1 & x_1y_2 & x_1y_3 \\ & x_2y_2 & x_2y_3 \\ & & x_3y_3 \end{pmatrix} + \begin{pmatrix} & & \\ x_2y_1 & & \\ x_3y_1 & x_3y_2 & \end{pmatrix}^{\mathsf{T}}$$

ISW multiplication gadget

cross-products $\sum_{i,j} x_i y_j$

$$\begin{pmatrix} x_1y_1 & x_1y_2 & x_1y_3 \\ & x_2y_2 & x_2y_3 \\ & & x_3y_3 \end{pmatrix} + \begin{pmatrix} & & \\ x_2y_1 & & \\ x_3y_1 & x_3y_2 & \end{pmatrix}^{\mathsf{T}}$$

ISW multiplication gadget

cross-products $\sum_{i,j} x_i y_j$

$$\left(\left(\begin{matrix} x_1y_1 & x_1y_2 & x_1y_3 \\ & x_2y_2 & x_2y_3 \\ & & x_3y_3 \end{matrix}\right) + \left(\begin{matrix} & & \\ x_2y_1 & & \\ x_3y_1 & x_3y_2 & \end{matrix}\right)^{\top}\right) + \left(\begin{matrix} & r_{1,2} & r_{1,3} \\ -r_{1,2} & & r_{2,3} \\ -r_{1,3} & -r_{2,3} & \end{matrix}\right)$$

ISW multiplication gadget

cross-products $\sum_{i,j} x_i y_j$

fresh randomness
(cancelling out)

$$\left(\begin{pmatrix} x_1y_1 & x_1y_2 & x_1y_3 \\ & x_2y_2 & x_2y_3 \\ & & x_3y_3 \end{pmatrix} + \begin{pmatrix} & & \\ x_2y_1 & & \\ x_3y_1 & x_3y_2 & \end{pmatrix}\right)^{\mathsf{T}} + \begin{pmatrix} & r_{1,2} & r_{1,3} \\ -r_{1,2} & & r_{2,3} \\ -r_{1,3} & -r_{2,3} & \end{pmatrix}$$

ISW multiplication gadget

cross-products $\sum_{i,j} x_i y_j$

fresh randomness (cancelling out)

$$\left(\begin{pmatrix} x_1y_1 & x_1y_2 & x_1y_3 \\ & x_2y_2 & x_2y_3 \\ & & x_3y_3 \end{pmatrix} + \begin{pmatrix} & & \\ x_2y_1 & & \\ x_3y_1 & x_3y_2 & \end{pmatrix}\right)^{\top} + \begin{pmatrix} & r_{1,2} & r_{1,3} \\ -r_{1,2} & & r_{2,3} \\ -r_{1,3} & -r_{2,3} & \end{pmatrix}$$

row sum →

$$\begin{pmatrix} z_1 \\ z_2 \\ z_3 \end{pmatrix}$$

ISW multiplication gadget

cross-products $\sum_{i,j} x_i y_j$

fresh randomness (cancelling out)

$$\begin{pmatrix} x_1y_1 & x_1y_2 & x_1y_3 \\ & x_2y_2 & x_2y_3 \\ & & x_3y_3 \end{pmatrix} + \begin{pmatrix} & & \\ x_2y_1 & & \\ x_3y_1 & x_3y_2 & \end{pmatrix}^{\top} + \begin{pmatrix} & r_{1,2} & r_{1,3} \\ -r_{1,2} & & r_{2,3} \\ -r_{1,3} & -r_{2,3} & \end{pmatrix}$$

row sum → $\begin{pmatrix} z_1 \\ z_2 \\ z_3 \end{pmatrix}$

t probes ⇒ info on at most t shares

Composition security

⚠️ probing security for gadgets $\not\Longrightarrow$ global (region) probing security

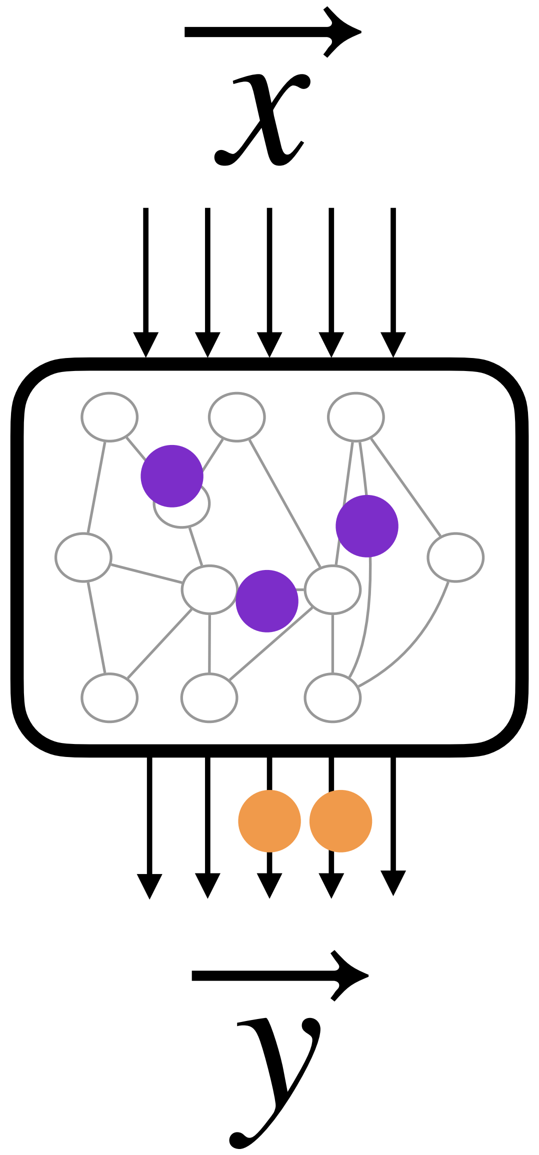
💡 composition security notions

Composition security

⚠️ probing security for gadgets $\not\Rightarrow$ global (region) probing security

💡 composition security notions

Example: strong non-interference (SNI)



t_1 *internal probes*

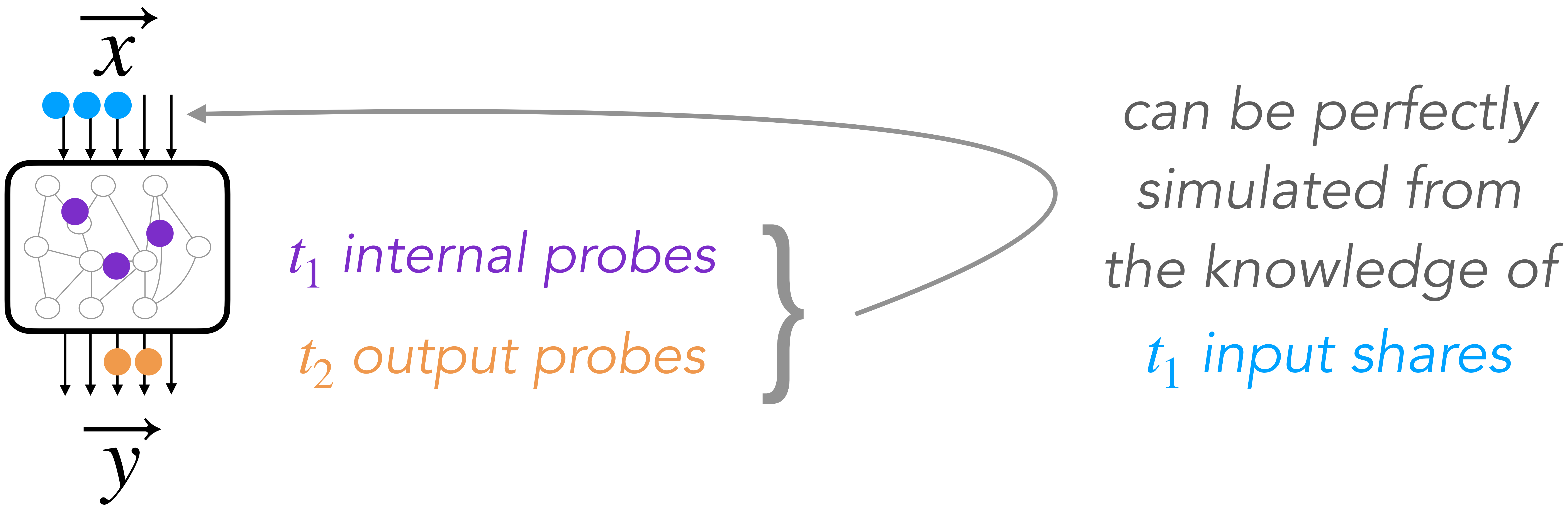
t_2 *output probes*

Composition security

⚠️ probing security for gadgets ⇏ global (region) probing security

💡 composition security notions

Example: strong non-interference (SNI)

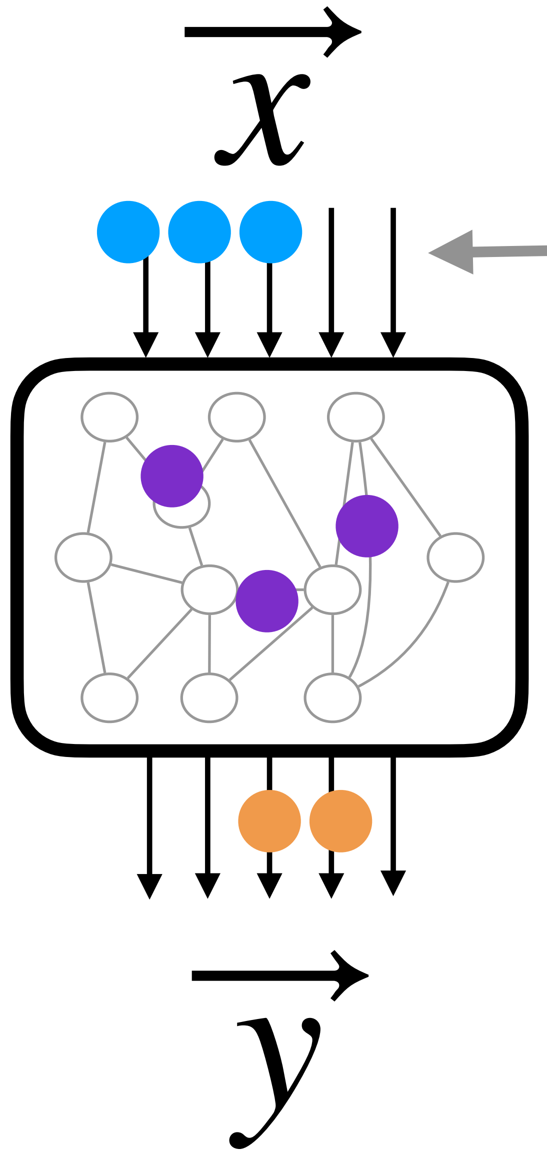


Composition security

⚠️ probing security for gadgets $\not\Rightarrow$ global (region) probing security

💡 composition security notions

Example: strong non-interference (SNI)

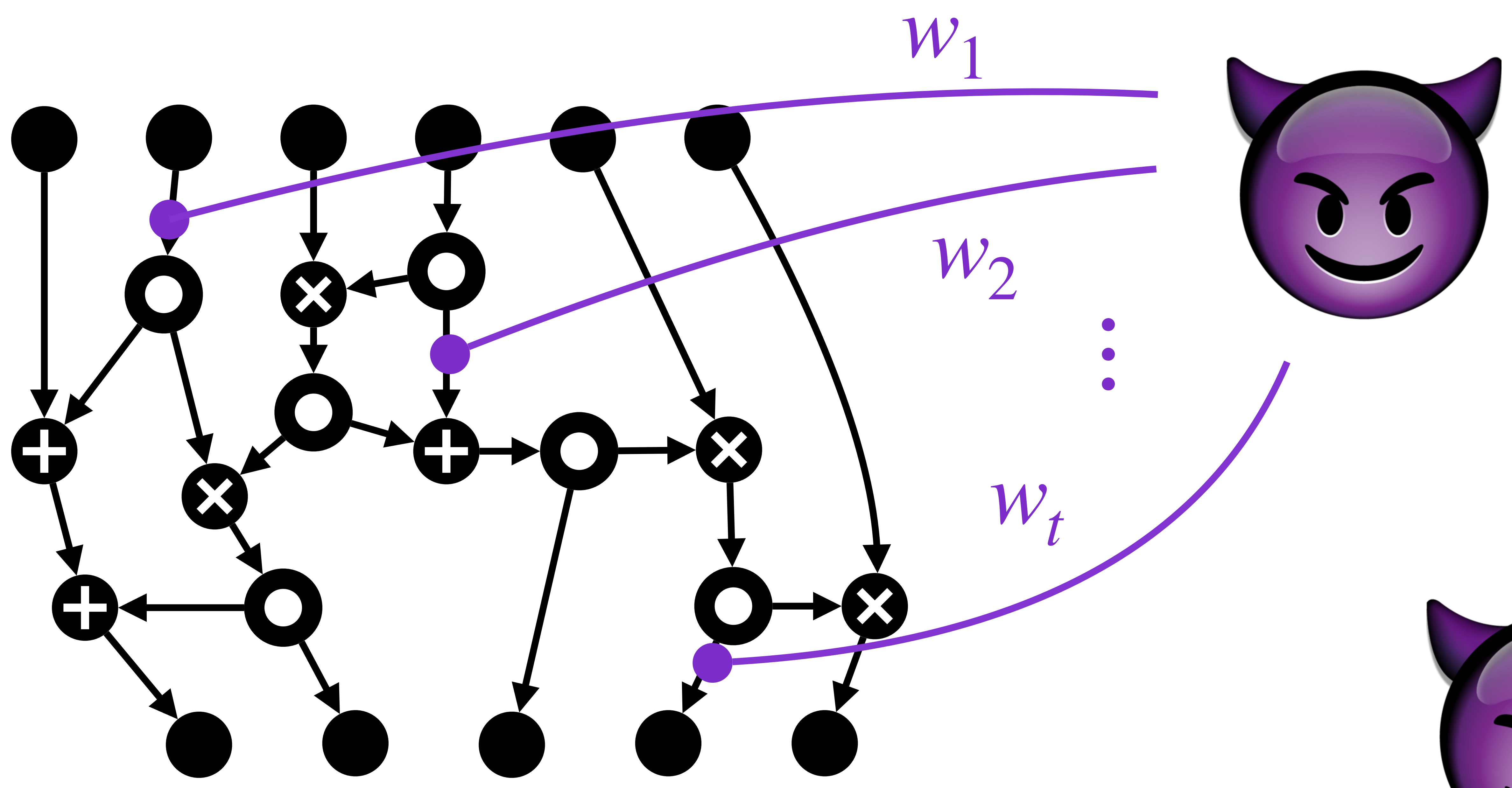


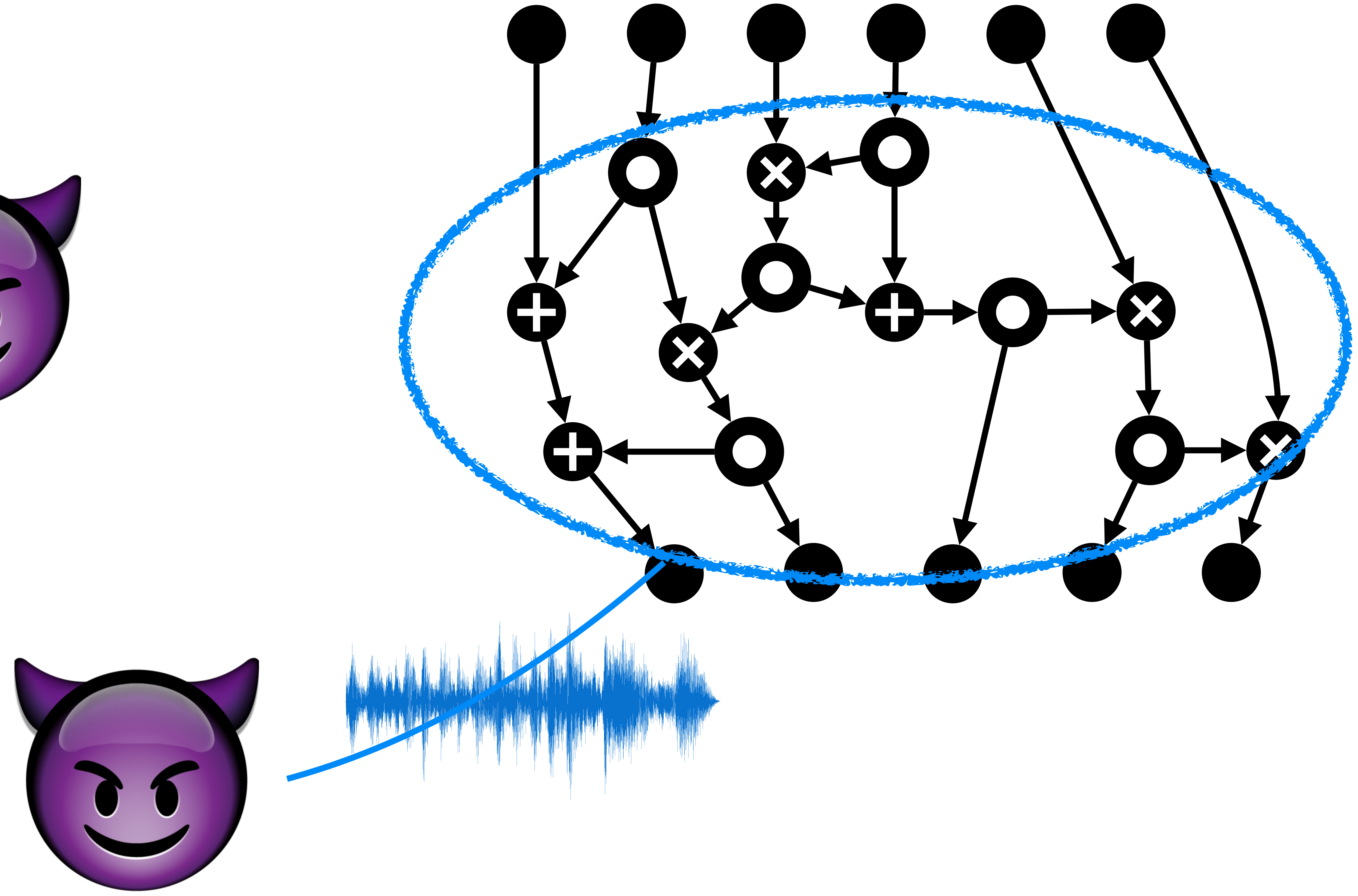
t_1 internal probes
t_2 output probes

can be perfectly simulated from the knowledge of t_1 input shares

SNI gadgets $\Rightarrow$ global region probing security

But… wait a minute!

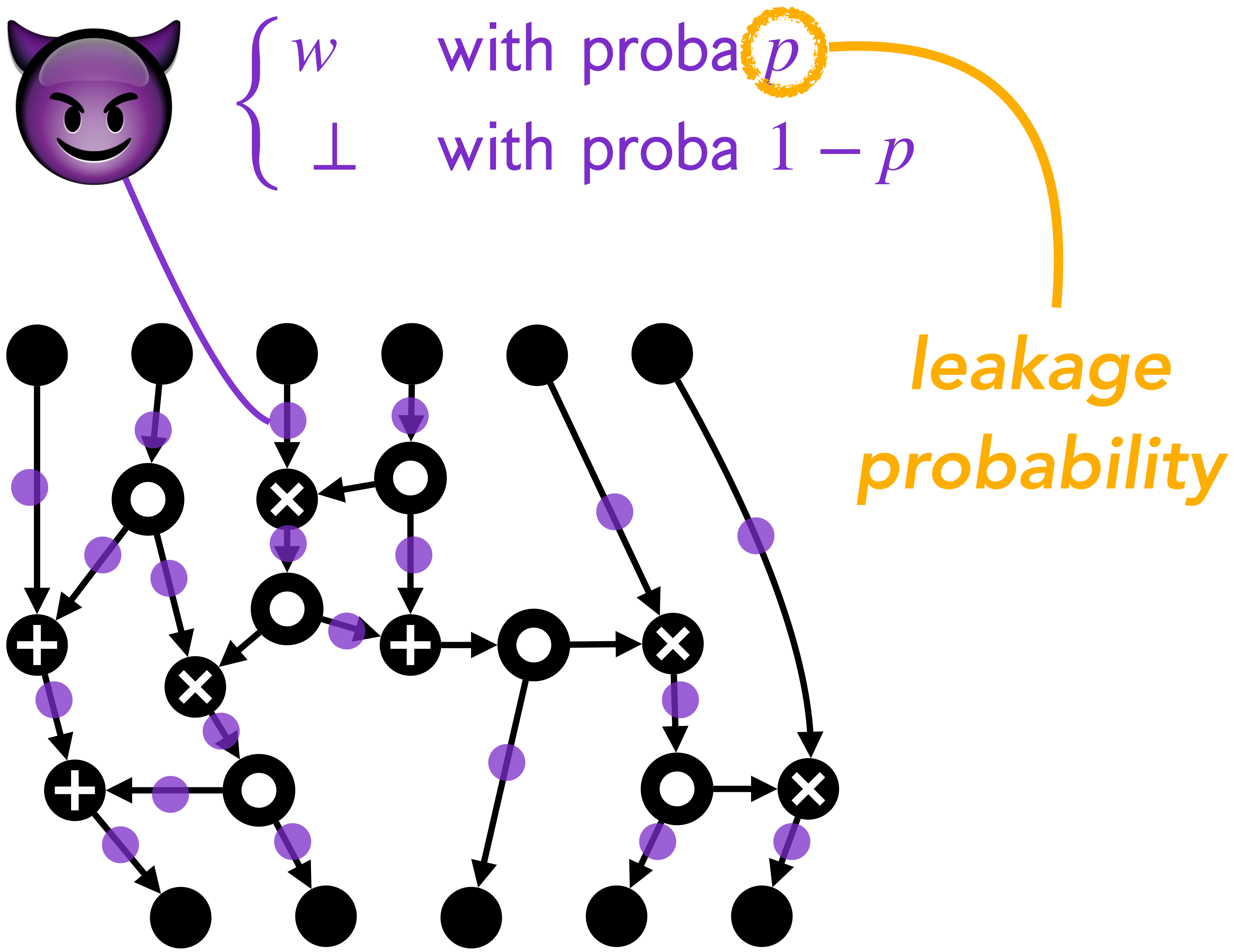




How do I get from probing security …

… to noisy leakage security?!

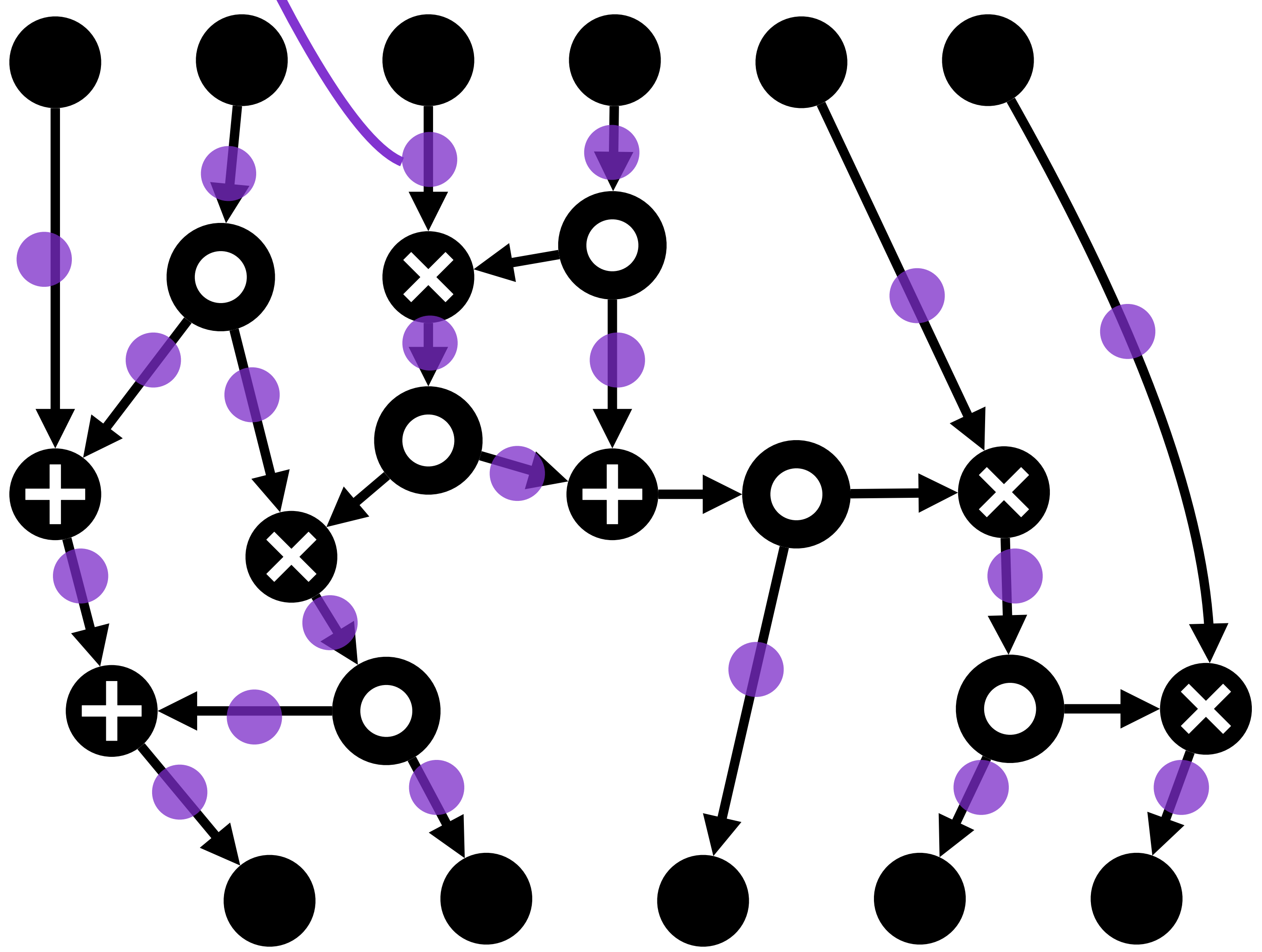
Random probing model



Random probing model

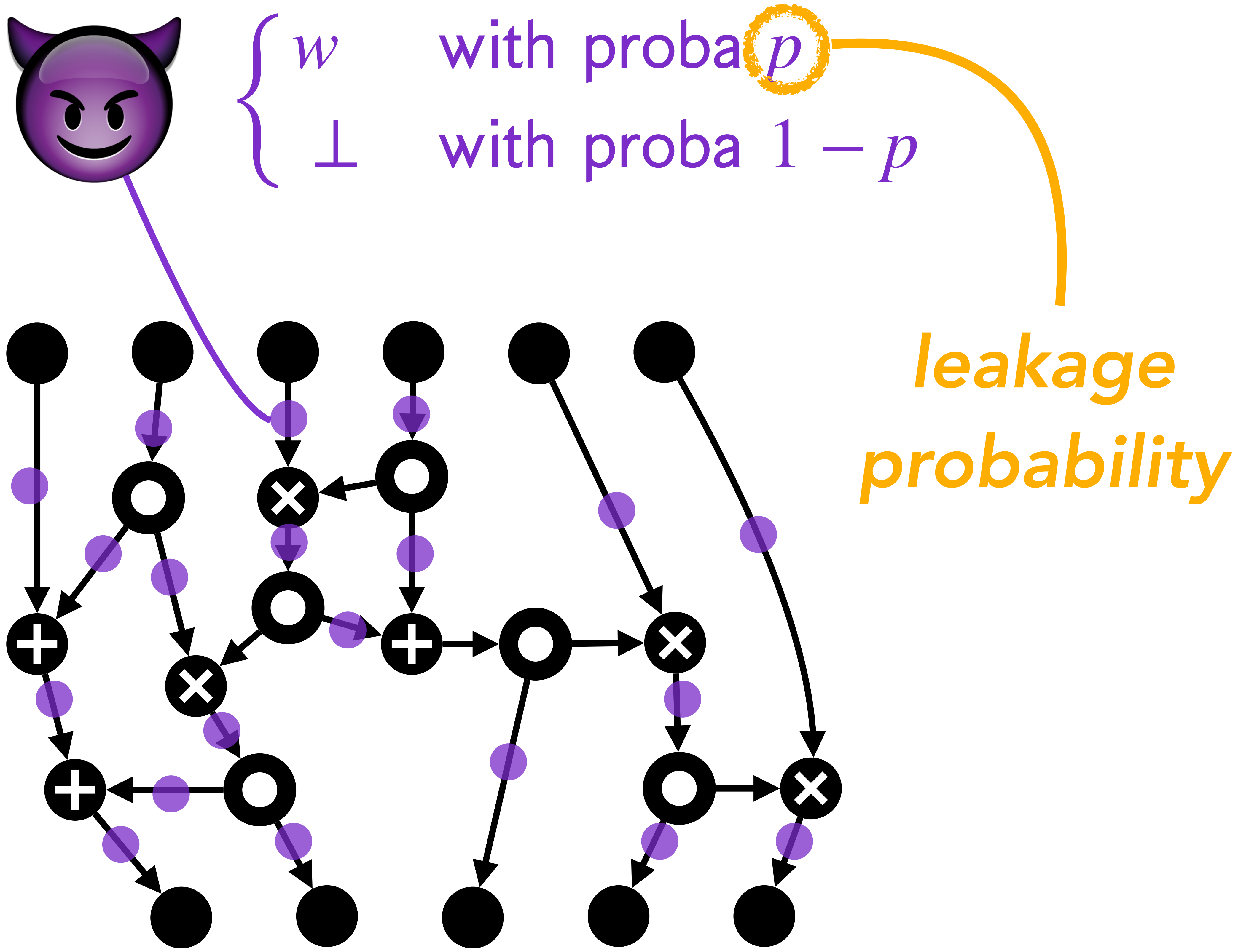
$$\begin{cases} w & \text{with proba } p \\ \perp & \text{with proba } 1-p \end{cases}$$

leakage probability



In average $p \cdot |G|$ leaking wires per gadget

Random probing model

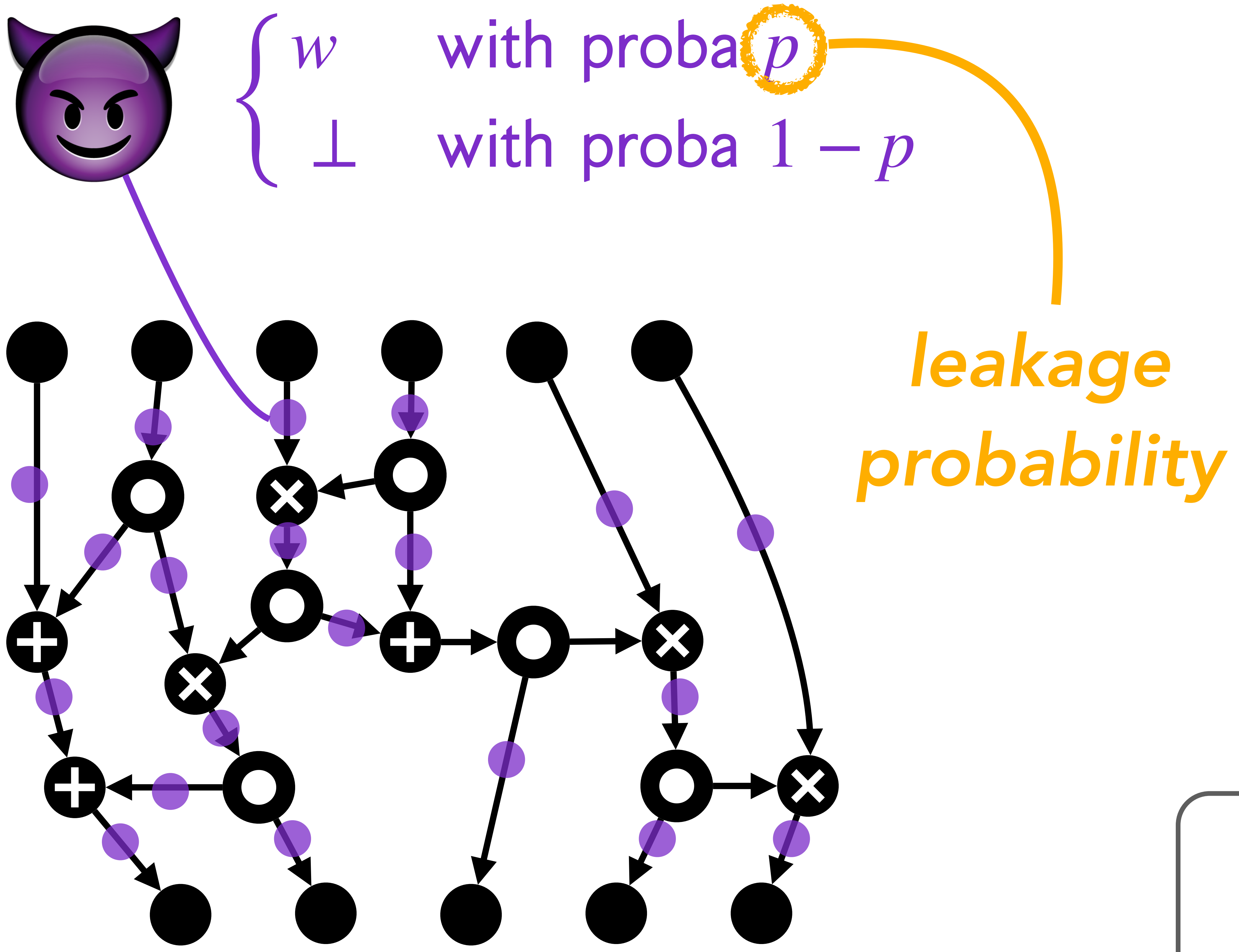


In average $p \cdot |G|$ leaking wires per gadget

Chernoff bound:

$< 2p\,|G|$ leaking wires with overwhelming probability

Random probing model



In average $p \cdot |G|$ leaking wires per gadget

Chernoff bound:

$< 2p|G|$ leaking wires with overwhelming probability

r-region probing security

$\Rightarrow$ p-random probing security

with $p = \Theta(r)$

Unifying probing and noisy models

Key lemma:

If f is δ-noisy, then $\exists f'$ s.t.

$$f(x) = f'(\phi(x))$$

where

$$\phi(x) := \begin{cases} x & \text{with proba } p \leq \delta \cdot |\mathcal{X}| \\ \perp & \text{with proba } 1 - p \end{cases}$$

Unifying probing and noisy models

<u>Key lemma:</u>

If f is δ-noisy, then $\exists\, f'$ s.t.

$$f(x) = f'(\phi(x))$$

where

$$\phi(x) := \begin{cases} x & \text{with proba } p \leq \delta \cdot |\mathcal{X}| \\ \bot & \text{with proba } 1-p \end{cases}$$

$\Rightarrow$ δ-noisy leakage can be simulated from p-random probing leakage

Unifying probing and noisy models

Key lemma:

If f is δ-noisy, then $\exists f'$ s.t.

$$f(x) = f'(\phi(x))$$

where

$$\phi(x) := \begin{cases} x & \text{with proba } p \leq \delta \cdot |\mathcal{X}| \\ \perp & \text{with proba } 1-p \end{cases}$$

⇒ δ-noisy leakage can be simulated from p-random probing leakage

Random probing leakage

$$\phi(w_1),\ \phi(w_2),\ \ldots\ ,\ \phi(w_N)$$

Unifying probing and noisy models

Key lemma:

If f is δ-noisy, then $\exists f'$ s.t.

$$f(x) = f'(\phi(x))$$

where

$$\phi(x) := \begin{cases} x & \text{with proba } p \leq \delta \cdot |\mathcal{X}| \\ \perp & \text{with proba } 1 - p \end{cases}$$

$\Rightarrow$ ***δ-noisy leakage can be simulated from p-random probing leakage***

Random probing leakage

$\phi(w_1), \phi(w_2), \ \dots \ , \phi(w_N)$

Apply $f_1', \dots, f_N'$

Noisy leakage

$f_1(w_1), f_2(w_2), \ \dots \ , f_N(w_N)$

Unifying probing and noisy models

Key lemma:

If f is δ-noisy, then $\exists f'$ s.t.

$$f(x) = f'(\phi(x))$$

where

$$\phi(x) := \begin{cases} x & \text{with proba } p \leq \delta \cdot |\mathcal{X}| \\ \perp & \text{with proba } 1 - p \end{cases}$$

$\Rightarrow$ δ-noisy leakage can be simulated from p-random probing leakage

Random probing leakage

$\phi(w_1), \phi(w_2), \ \dots \ , \phi(w_N)$

Apply $f_1', \dots, f_N'$

Noisy leakage

$f_1(w_1), f_2(w_2), \ \dots \ , f_N(w_N)$

p-random probing security

$\Rightarrow$ δ-noisy security with $\delta = \Theta(p)$

Unifying probing and noisy models

r-region probing security

$\Rightarrow$

p-random probing security

with $p = \Theta(r)$

$\Rightarrow$

δ-noisy leakage security

with $\delta = \Theta(p)$

Unifying probing and noisy models

r-region probing security

$\Rightarrow$

p-random probing security

with $p = \Theta(r)$

$\Rightarrow$

δ-noisy leakage security

with $\delta = \Theta(p)$

leakage rate

$$\begin{cases} 1 = \text{lot of leakage (low noise)} \\ 0 = \text{no leakage (infinite noise)} \end{cases}$$

Unifying probing and noisy models

r-region probing security

$\Rightarrow$

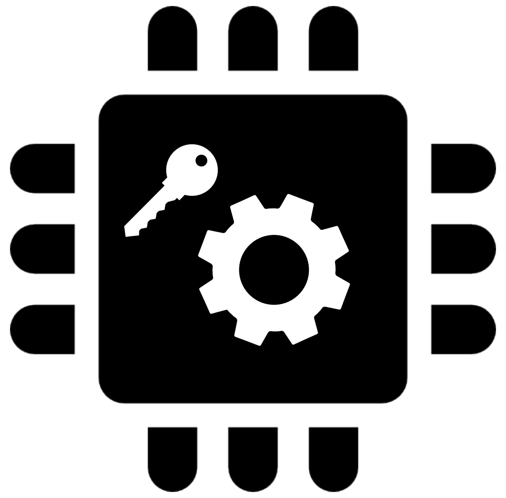
p-random probing security
with $p = \Theta(r)$

$\Rightarrow$

δ-noisy leakage security
with $\delta = \Theta(p)$

leakage rate

$\begin{cases} 1 & = \text{ lot of leakage (low noise)} \\ 0 & = \text{ no leakage (infinite noise)} \end{cases}$



the noise / leakage rate depends on the hardware

Unifying probing and noisy models

r-region probing security

$\Rightarrow$

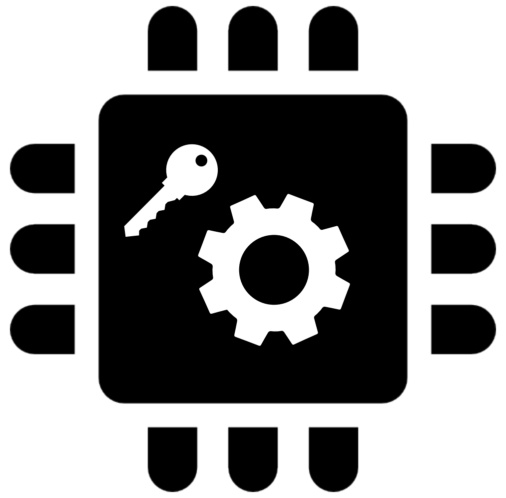
p-random probing security
with $p = \Theta(r)$

$\Rightarrow$

δ-noisy leakage security
with $\delta = \Theta(p)$

leakage rate

$\begin{cases} 1 & = \text{ lot of leakage (low noise)} \\ 0 & = \text{ no leakage (infinite noise)} \end{cases}$



the noise / leakage rate depends on the hardware

efficient masking schemes secure vs. constant (high) leakage rate

Secure schemes

State of the art

- State-of-the-art noisy-leakage-secure schemes
  - most schemes with **at least $\mathcal{O}(n^2)$ complexity**
  - a few schemes with $\mathcal{O}(1)$ leakage rate, but **constant not explicit**
- In what follows
  - region probing security in **quasilinear complexity**
  - random probing security with **explicit constant leakage rate**

Security in quasilinear complexity

Quasilinear masking

A $\overrightarrow{v}$-sharing of x

$$\overrightarrow{x} = (x_0, x_1, \ldots, x_{n-1}) \quad \text{s.t.} \quad \langle \overrightarrow{v}, \overrightarrow{x} \rangle = x$$

Quasilinear masking

A $\overrightarrow{v}$-sharing of x

$$\overrightarrow{x} = (x_0, x_1, \ldots, x_{n-1}) \quad \text{s.t.} \quad \langle \overrightarrow{v}, \overrightarrow{x} \rangle = x = \sum_{i=0}^{n-1} x_i \cdot \omega^i$$

$$\overrightarrow{v} = (1, \omega, \omega^2, \ldots, \omega^{n-1}) \quad \text{for} \quad \omega \xleftarrow{\$} \mathbb{F}$$

Quasilinear masking

Polynomial $P_{\vec{x}}(\omega)$

(shares = coefficients)

A $\vec{v}$-sharing of x

$$\vec{x} = (x_0, x_1, \ldots, x_{n-1}) \quad \text{s.t.} \quad \langle \vec{v}, \vec{x} \rangle = x = \sum_{i=0}^{n-1} x_i \cdot \omega^i$$

$$\vec{v} = (1, \omega, \omega^2, \ldots, \omega^{n-1}) \quad \text{for} \quad \omega \xleftarrow{\$} \mathbb{F}$$

Efficient multiplication

- Let $\vec{t}$ such that

$$P_{\vec{t}} = P_{\vec{x}} \cdot P_{\vec{y}}$$

- We get

$$P_{\vec{t}}(\omega) = \sum_{i=0}^{2n-1} t_i \omega^i = x \cdot y$$

Efficient multiplication

- Let $\vec{t}$ such that

$$P_{\vec{t}} = P_{\vec{x}} \cdot P_{\vec{y}}$$

- We get

$$P_{\vec{t}}(\omega) = \sum_{i=0}^{2n-1} t_i \omega^i = x \cdot y$$

Efficient multiplication

- Let $\vec{t}$ such that

$$P_{\vec{t}} = P_{\vec{x}} \cdot P_{\vec{y}}$$

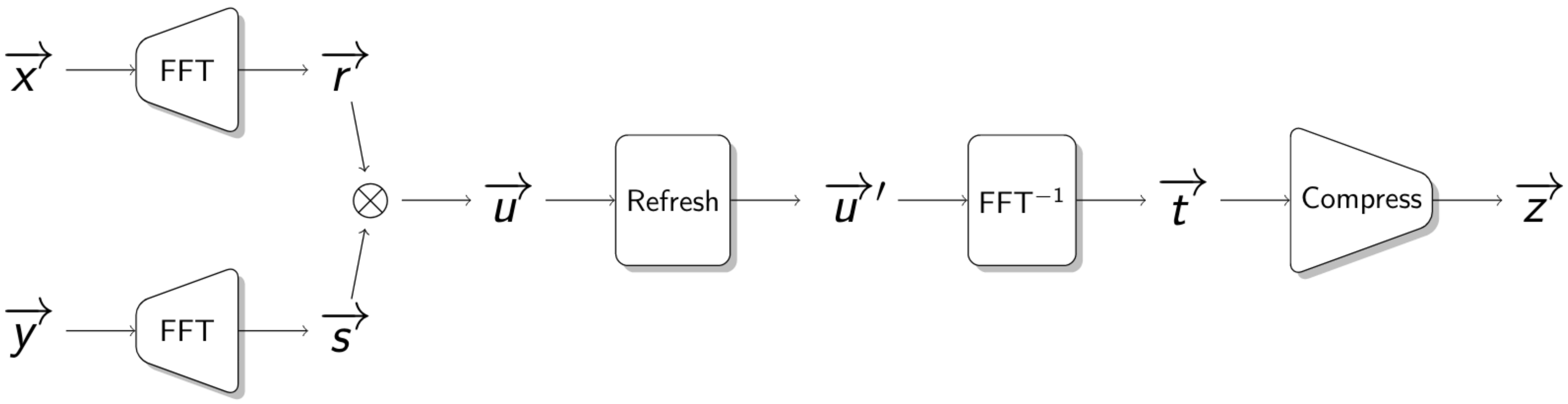
- We get

$$P_{\vec{t}}(\omega) = \sum_{i=0}^{2n-1} t_i \omega^i = x \cdot y$$

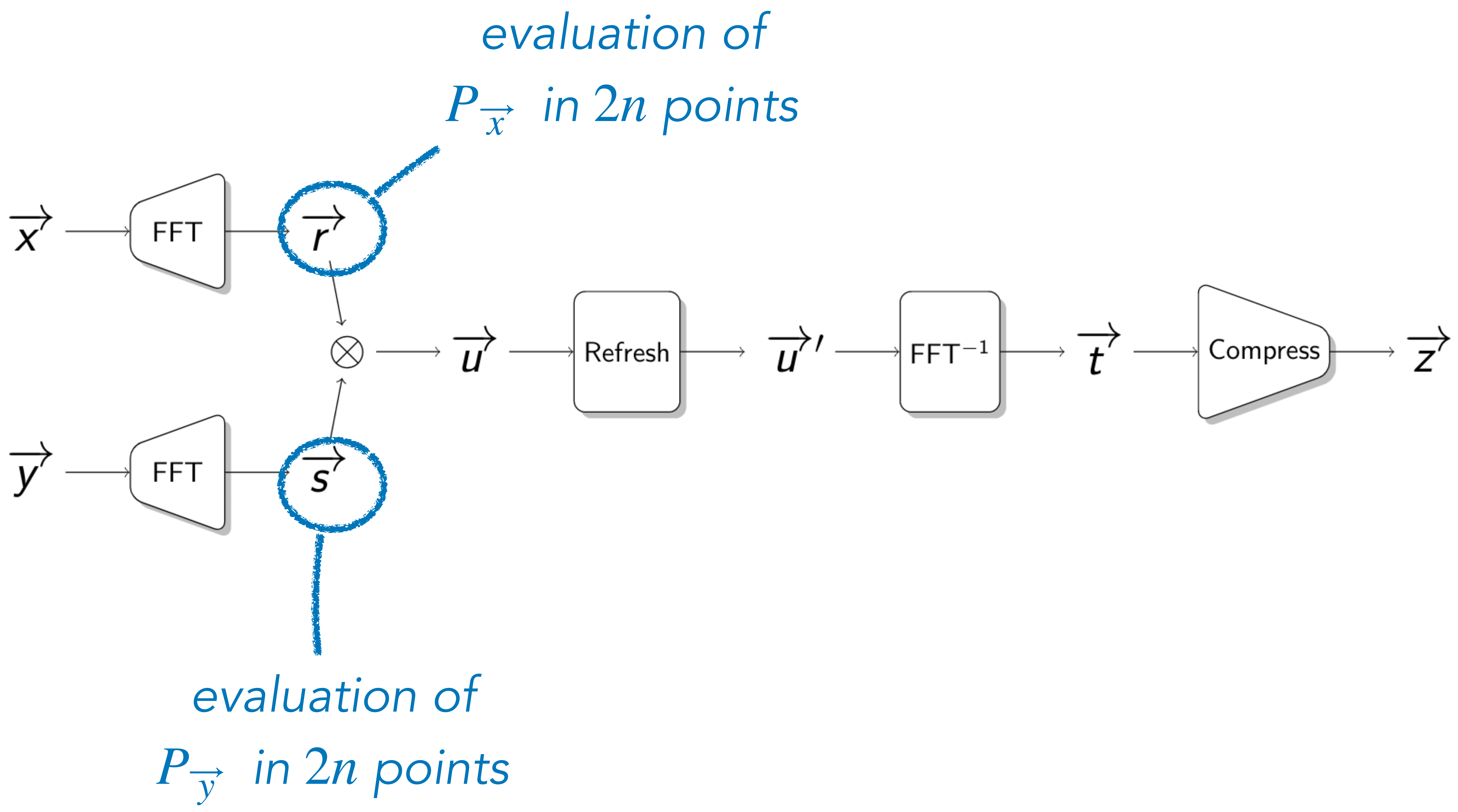
- Compression:

$$\vec{z} = (t_0, \ldots, t_{n-1}) + \omega^n \cdot (t_n, \ldots, t_{2n-1})$$

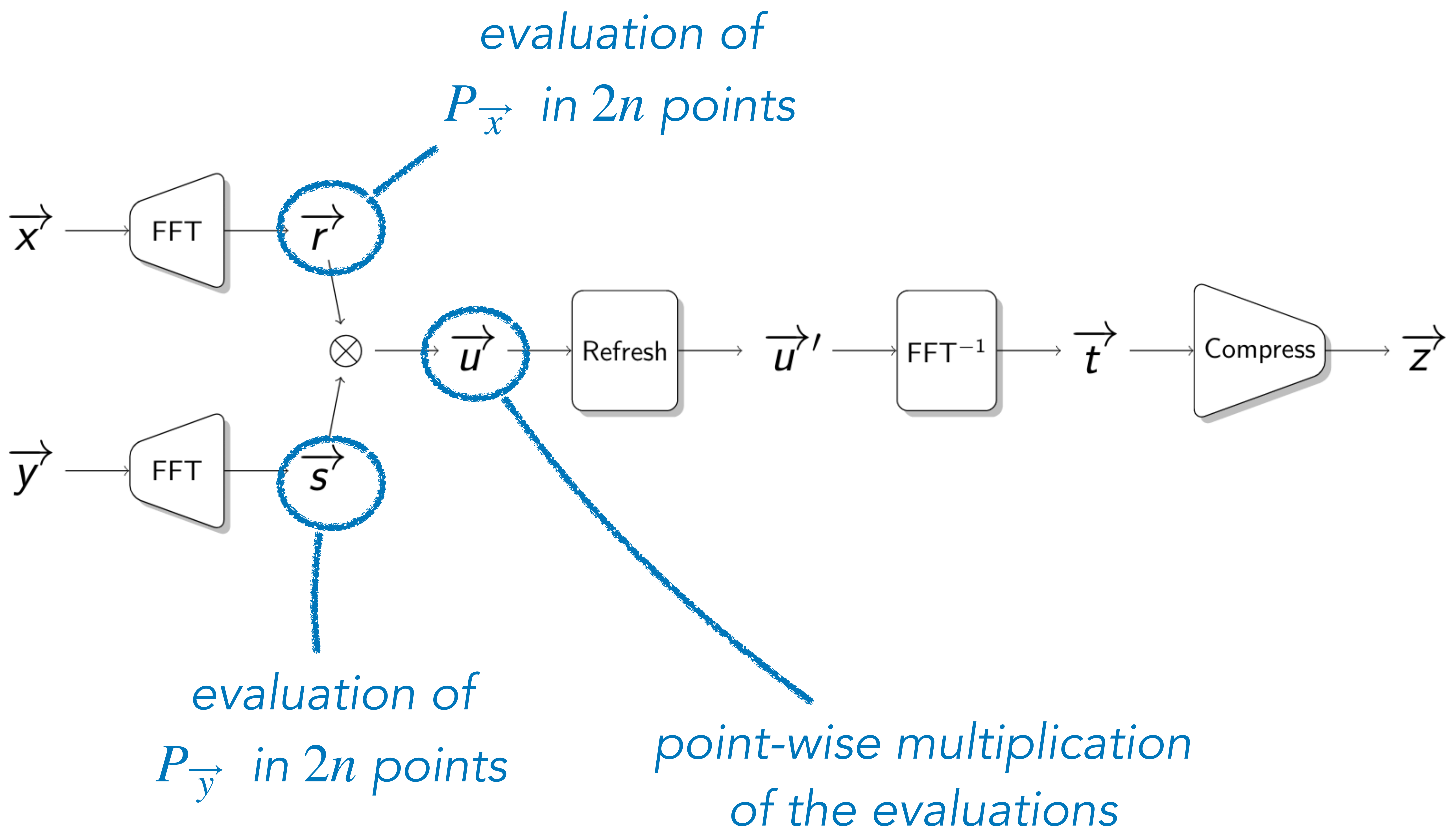
Multiplication gadget



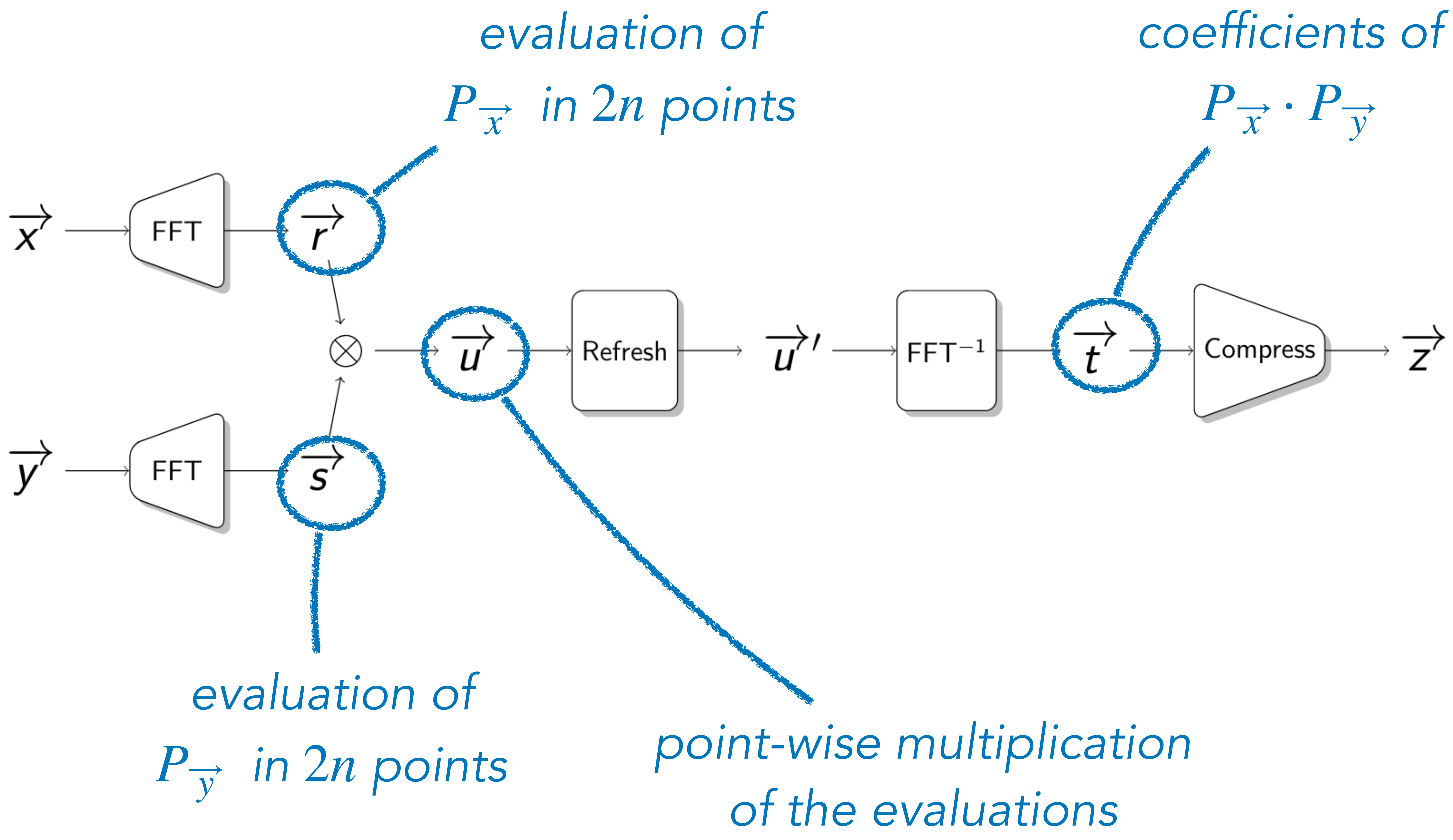
Multiplication gadget



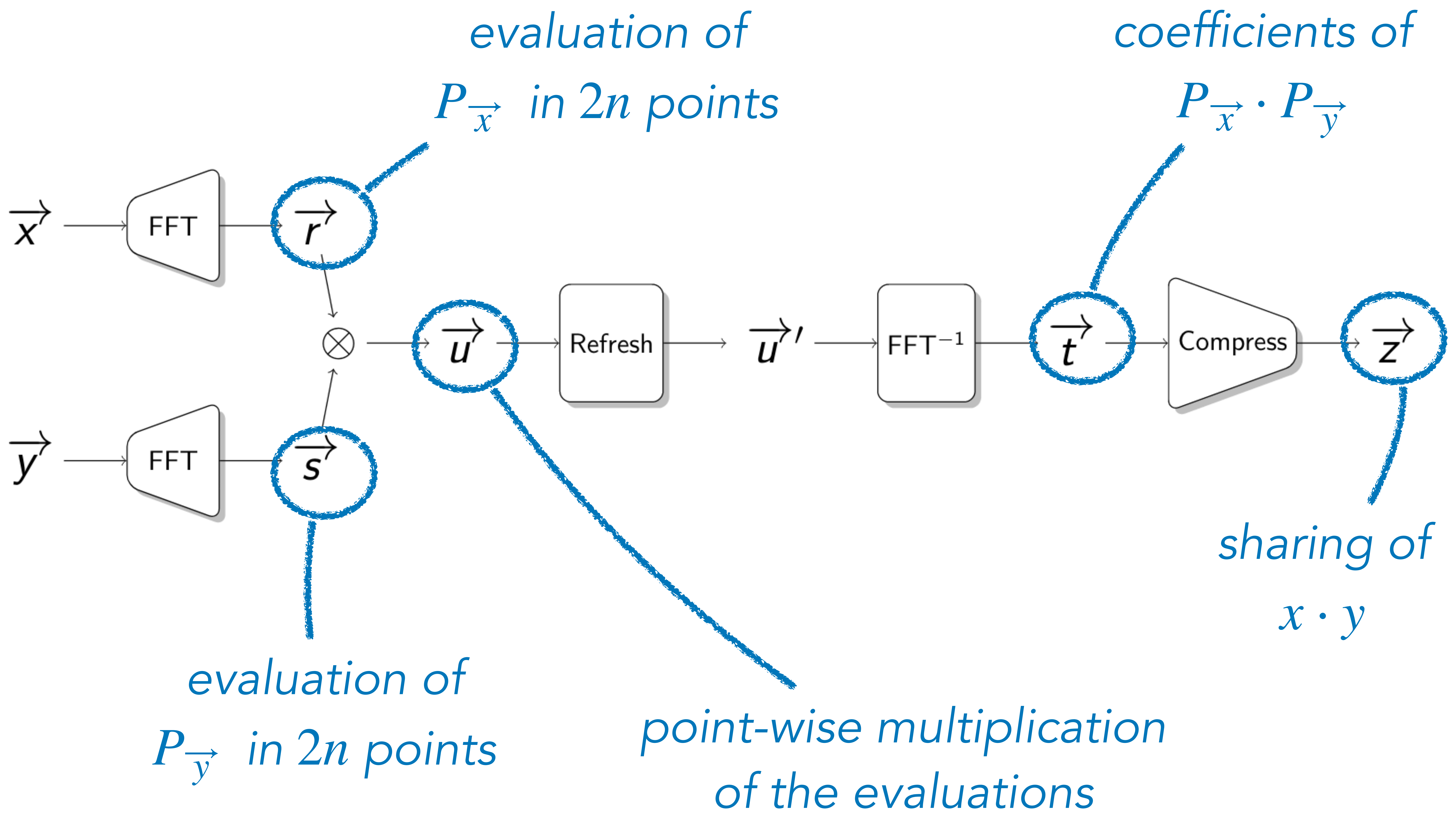
Multiplication gadget



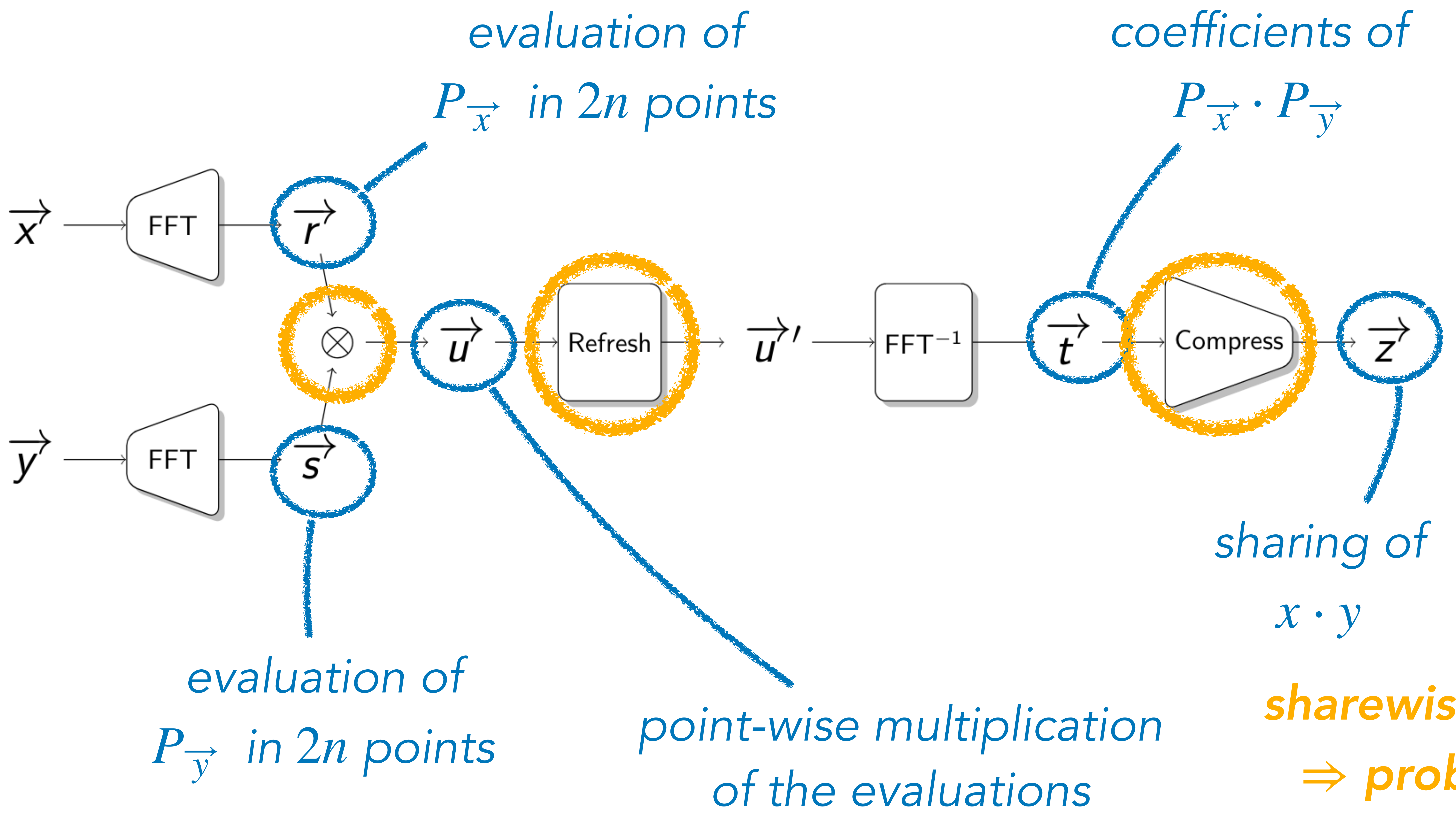
Multiplication gadget



Multiplication gadget



Multiplication gadget



sharewise operations
$\Rightarrow$ probing secure

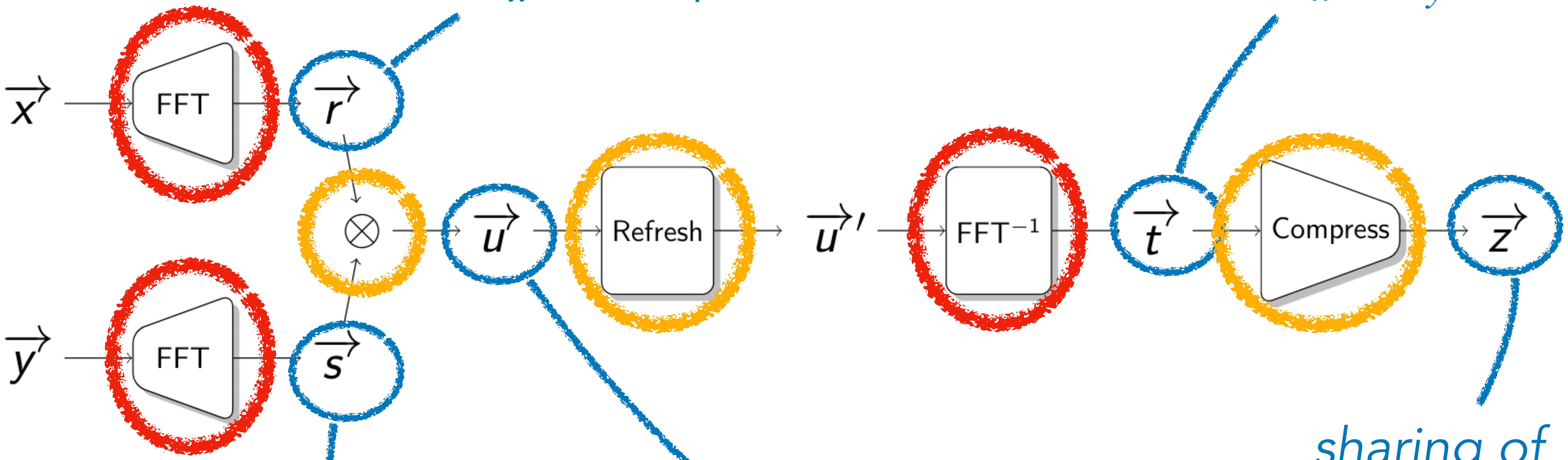
Multiplication gadget

evaluation of

$P_{\vec{x}}$ in $2n$ points

coefficients of

$P_{\vec{x}} \cdot P_{\vec{y}}$



sharing of

$x \cdot y$

what about the FFT?

evaluation of

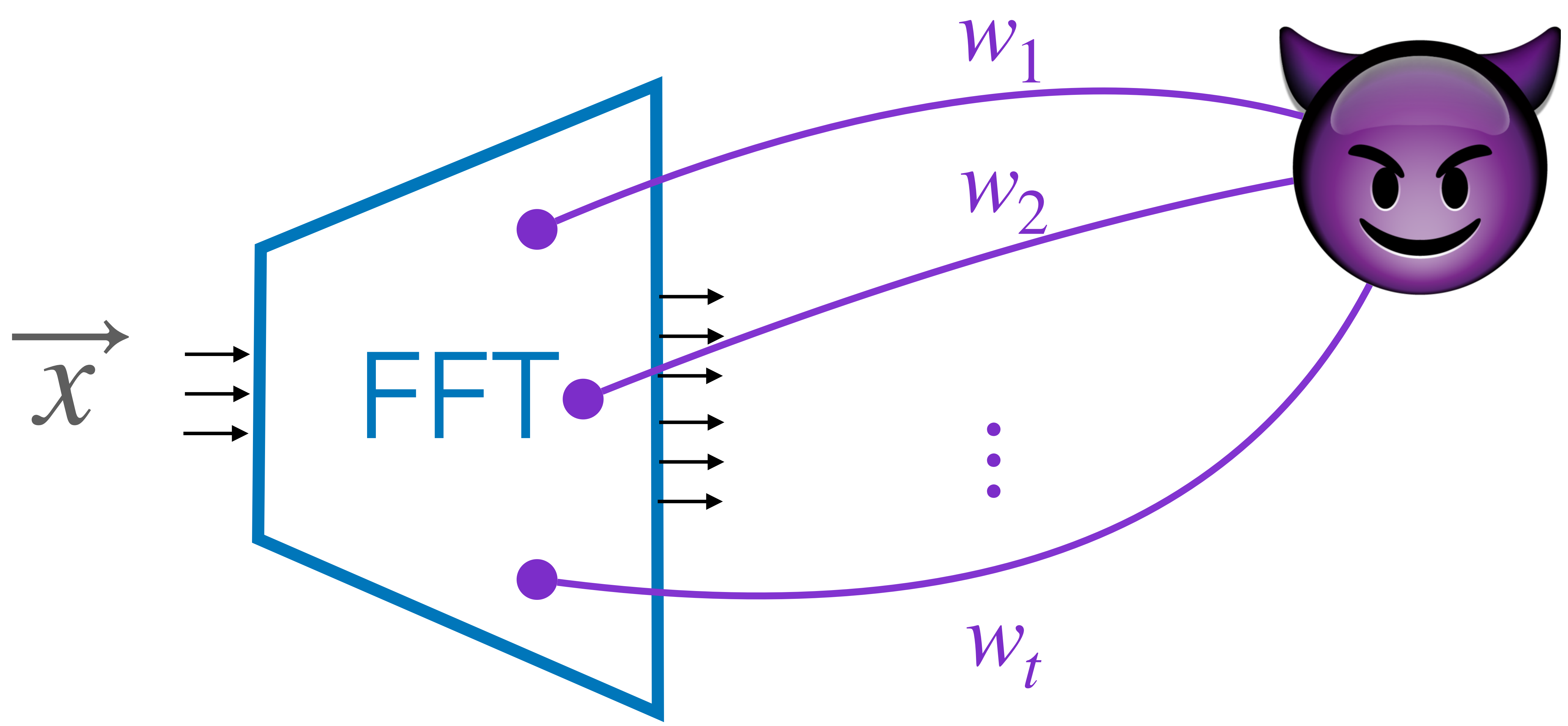
$P_{\vec{y}}$ in $2n$ points

point-wise multiplication of the evaluations

sharewise operations

$\Rightarrow$ probing secure

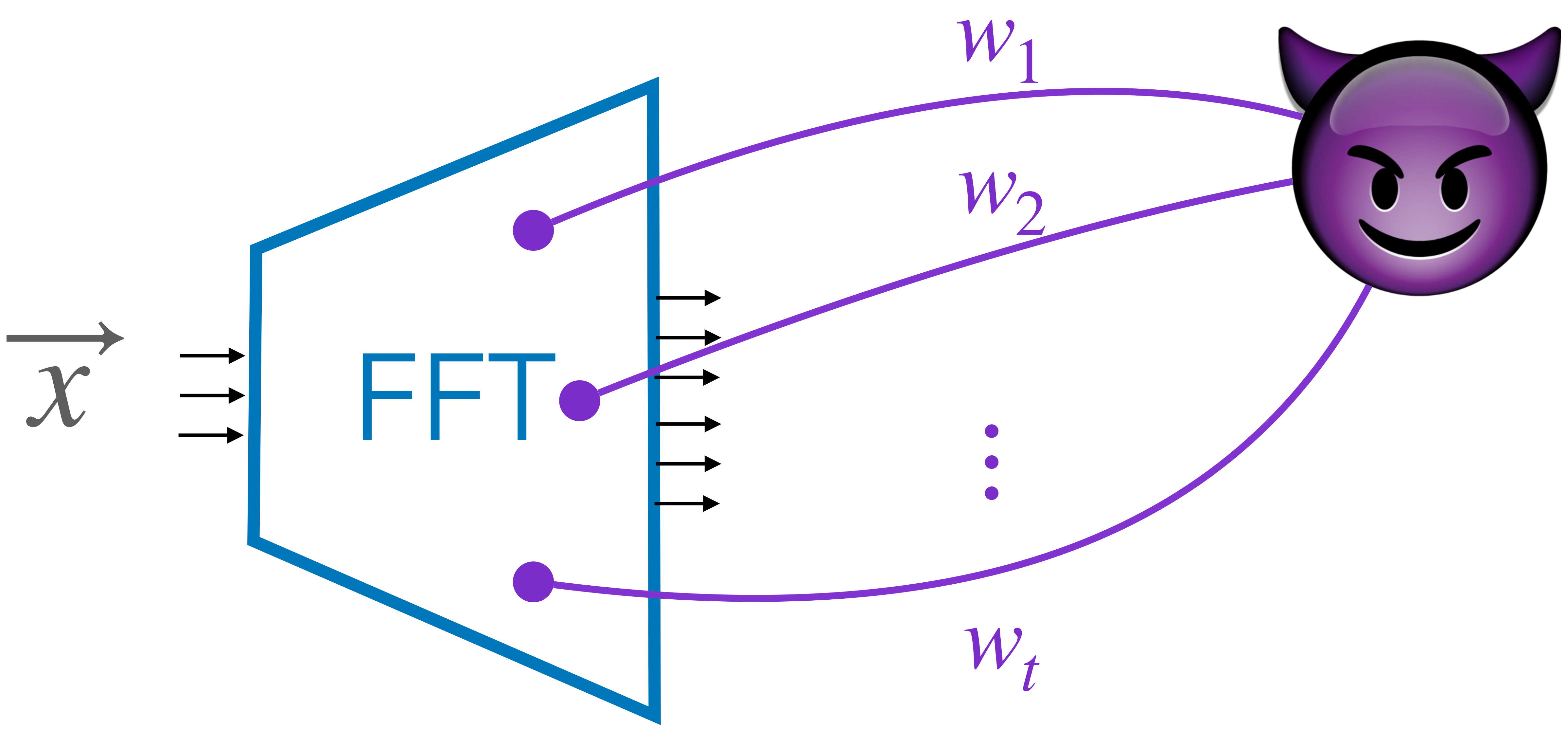
Probing security



💡 *FFT computes linear combinations of the x_i's*

$$\begin{pmatrix} w_1 \\ w_2 \\ \vdots \\ w_t \end{pmatrix} = \left[\ A \ \right] \cdot \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_t \end{pmatrix}$$

Probing security



FFT computes linear combinations of the x_i's

$$\begin{pmatrix} w_1 \\ w_2 \\ \vdots \\ w_t \end{pmatrix} = \begin{bmatrix} & A & \end{bmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_t \end{pmatrix}$$

Lemma 1

If $\vec{v} = \begin{pmatrix} \omega^0 \\ \omega^1 \\ \vdots \\ \omega^{n-1} \end{pmatrix} \notin \langle \begin{bmatrix} & A & \end{bmatrix} \rangle$ then $\begin{pmatrix} w_1 \\ w_2 \\ \vdots \\ w_t \end{pmatrix} \sim \mathcal{U}(\mathbb{F}^t)$ *(assuming A full rank wlog)*

Probing security

Lemma 2

$\exists$ at most t values of $\omega \in \mathbb{F}$ s.t. $\vec{v} = \begin{pmatrix} \omega^0 \\ \omega^1 \\ \vdots \\ \omega^{n-1} \end{pmatrix} \in \langle \left[\; A \; \right] \rangle$

Probing security

Lemma 2

$\exists$ at most t values of $\omega \in \mathbb{F}$ s.t. $\vec{v} = \begin{pmatrix} \omega^0 \\ \omega^1 \\ \vdots \\ \omega^{n-1} \end{pmatrix} \in \langle [\ A\ ] \rangle$

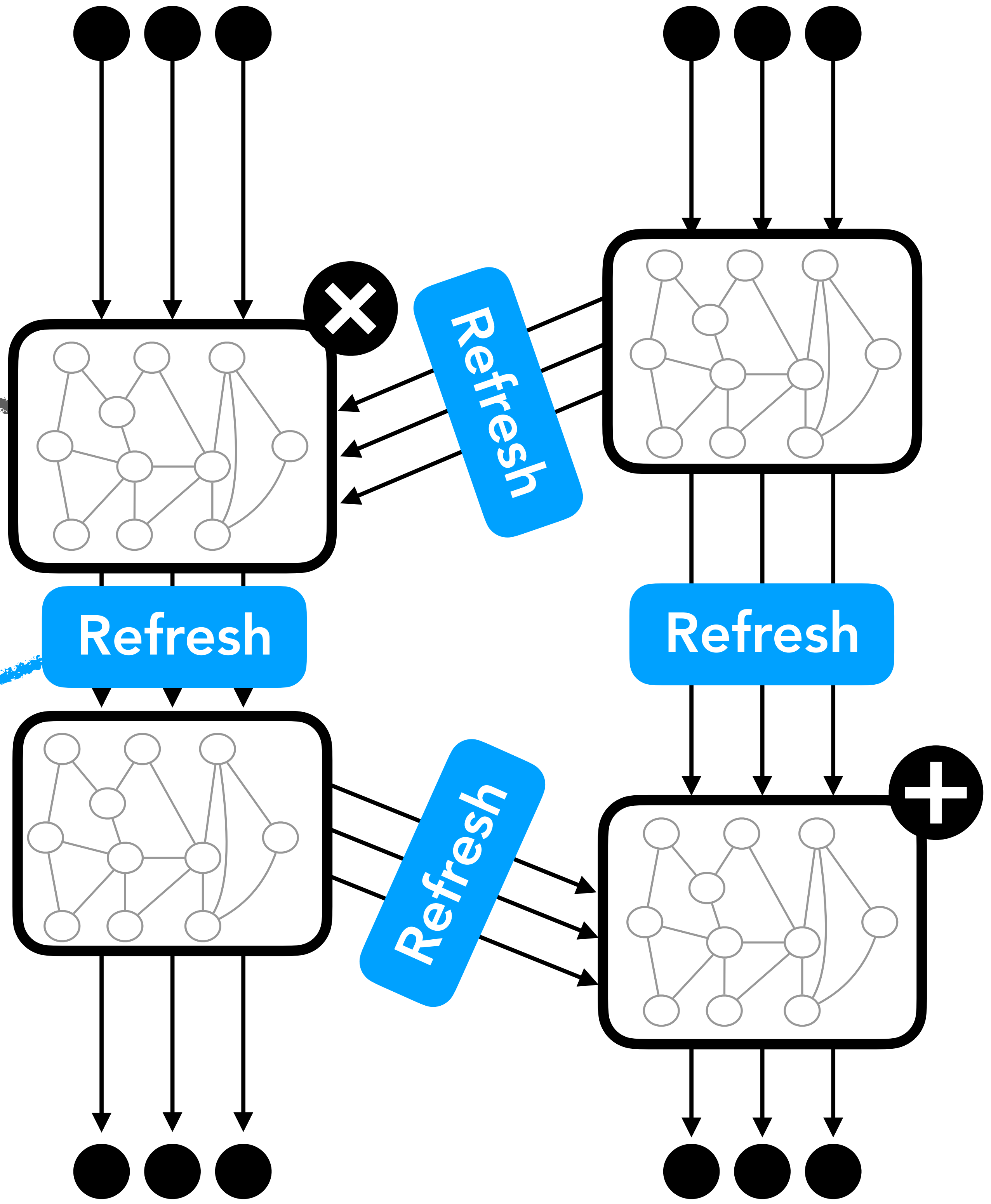
Lemma 1 + Lemma 2

$P\left[\,(w_1, \ldots, w_t) \text{ cannot be simulated}\,\right] \leq \dfrac{t}{|\mathbb{F}|} < \dfrac{n}{|\mathbb{F}|}$

Composition security

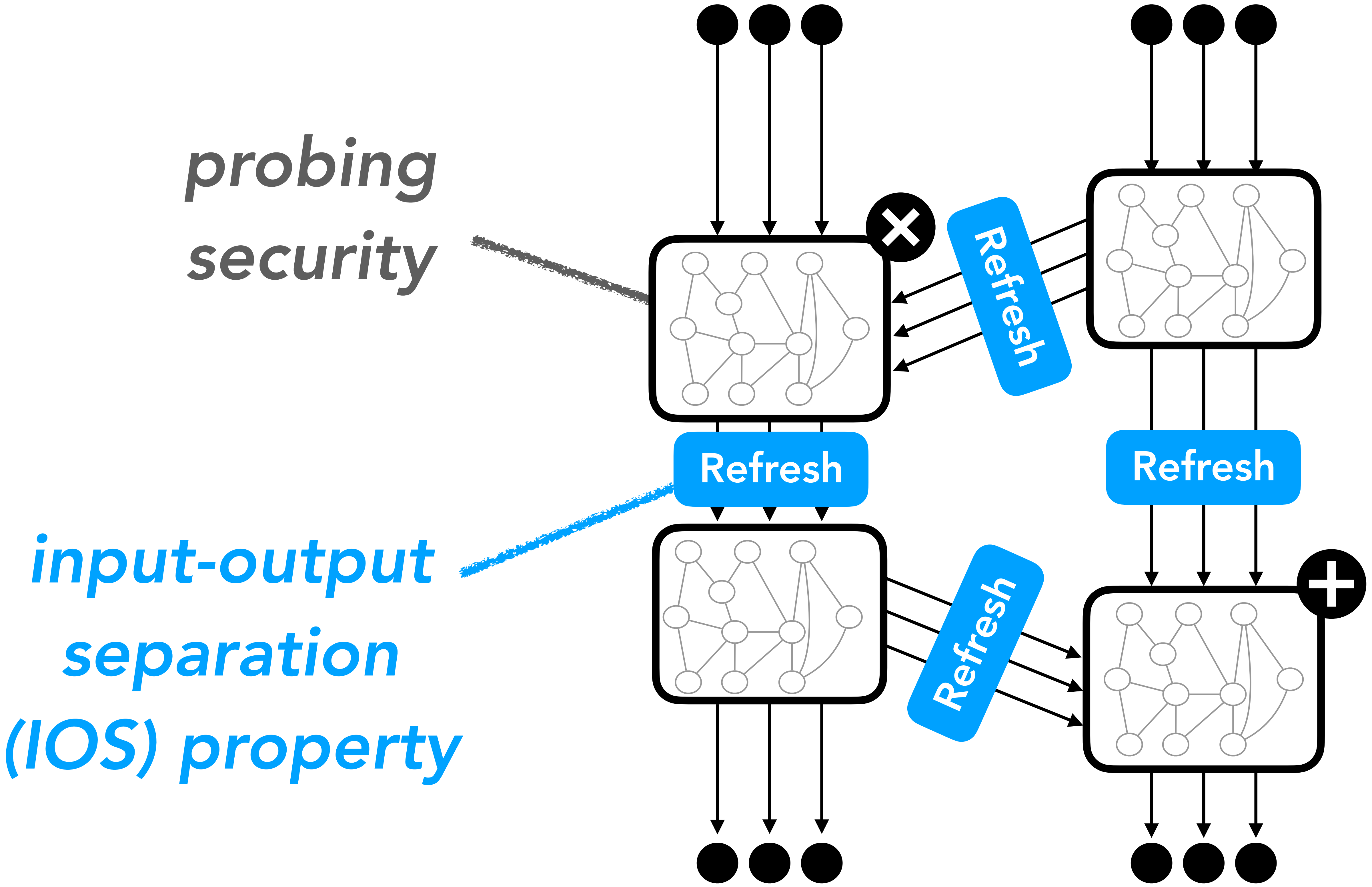
probing security

input-output separation (IOS) property



⇒ region probing security

Composition security



⇒ region probing security

Wrapping up:

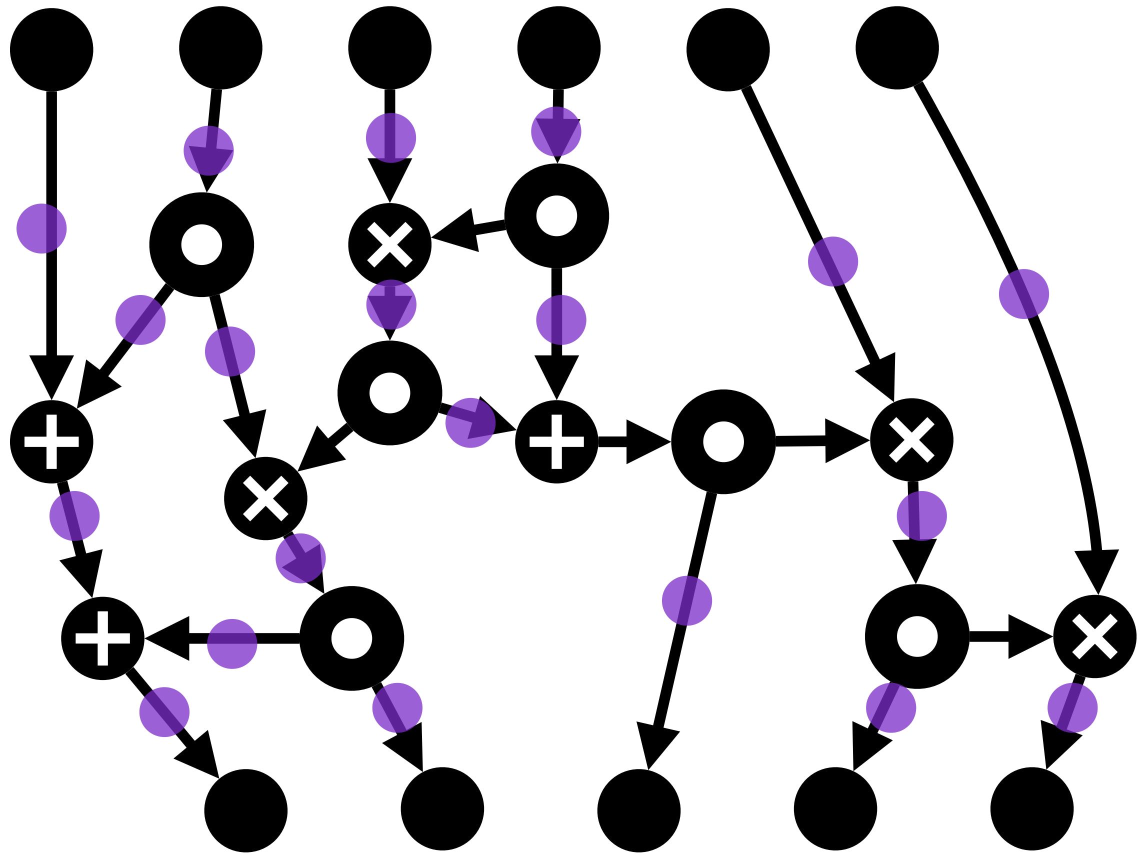
- Gadget complexity: $\Theta(n \log n)$
- Probes per gadget: $\Theta(n)$
- Leakage rate: $\Theta(1/\log n)$

Security with constant leakage rate

Simulation with abort

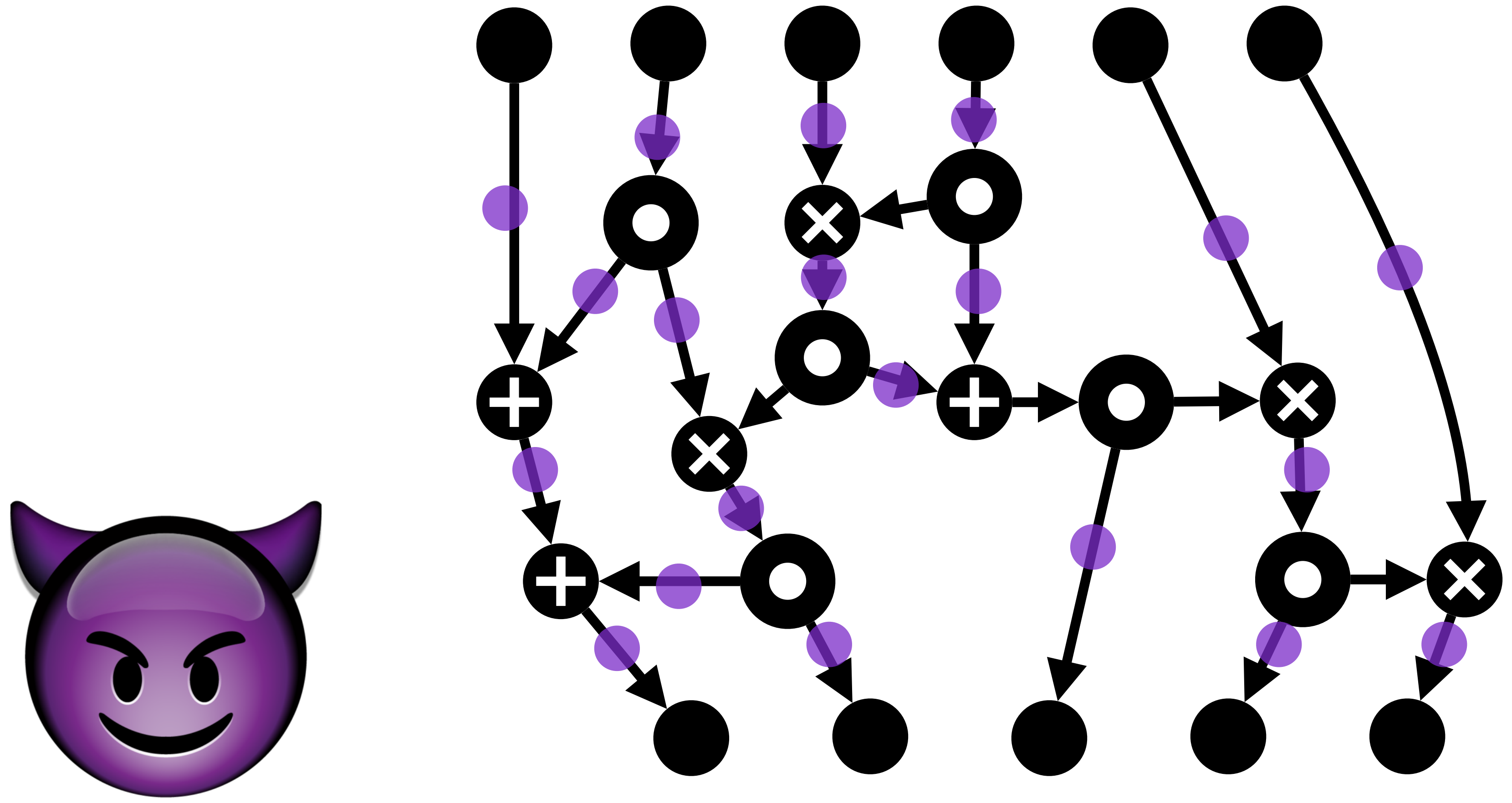
1. Sample a set of leaking wires

$$W \leftarrow \text{LeakingWires}(\hat{C}, p)$$



$$\begin{cases} w & \text{with proba } p \\ \perp & \text{with proba } 1-p \end{cases}$$

Simulation with abort



$$\begin{cases} w & \text{with proba } p \\ \perp & \text{with proba } 1-p \end{cases}$$

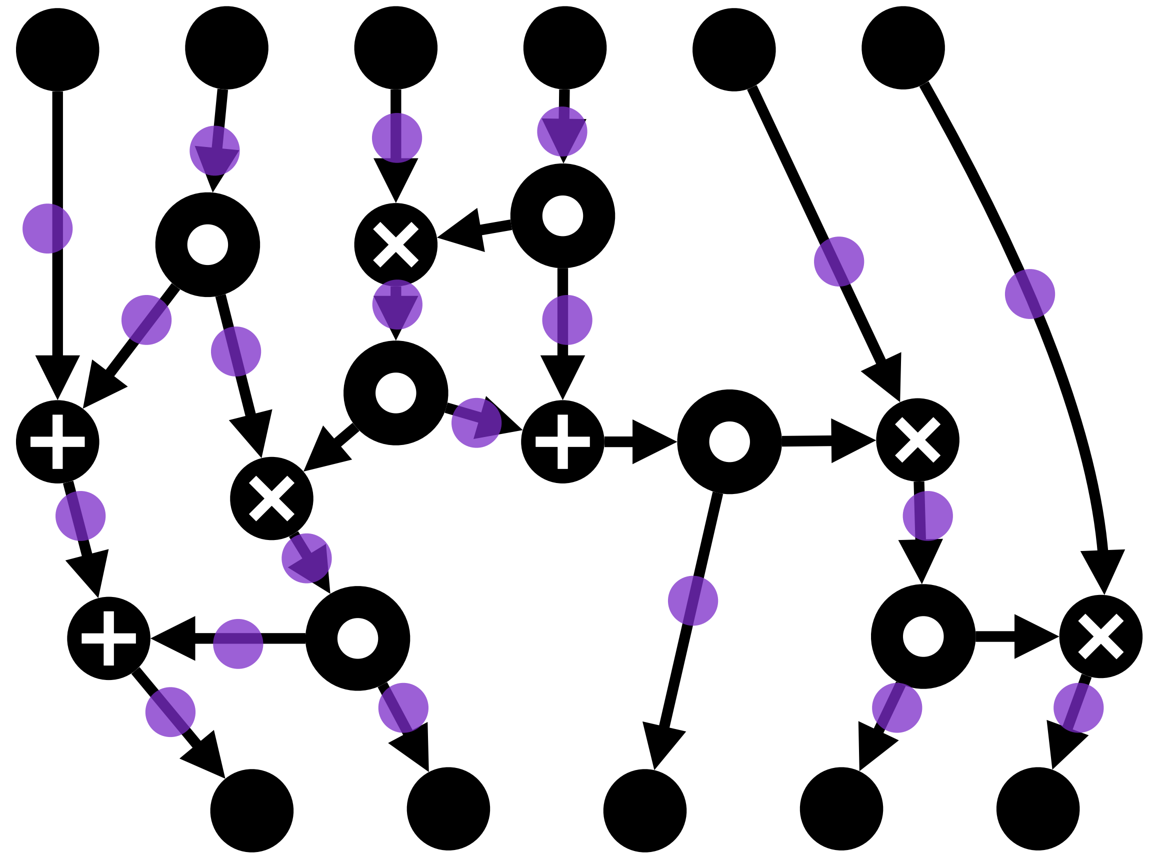
1. Sample a set of leaking wires

$$W \leftarrow \mathsf{LeakingWires}(\hat{C}, p)$$

2. Simulate the corresponding wire values

$$\mathsf{Sim} : W \mapsto \begin{cases} \text{perfect simulation} \\ \perp \text{ (abort)} \end{cases}$$

Simulation with abort



$\begin{cases} w & \text{with proba } p \\ \bot & \text{with proba } 1-p \end{cases}$

1. Sample a set of leaking wires

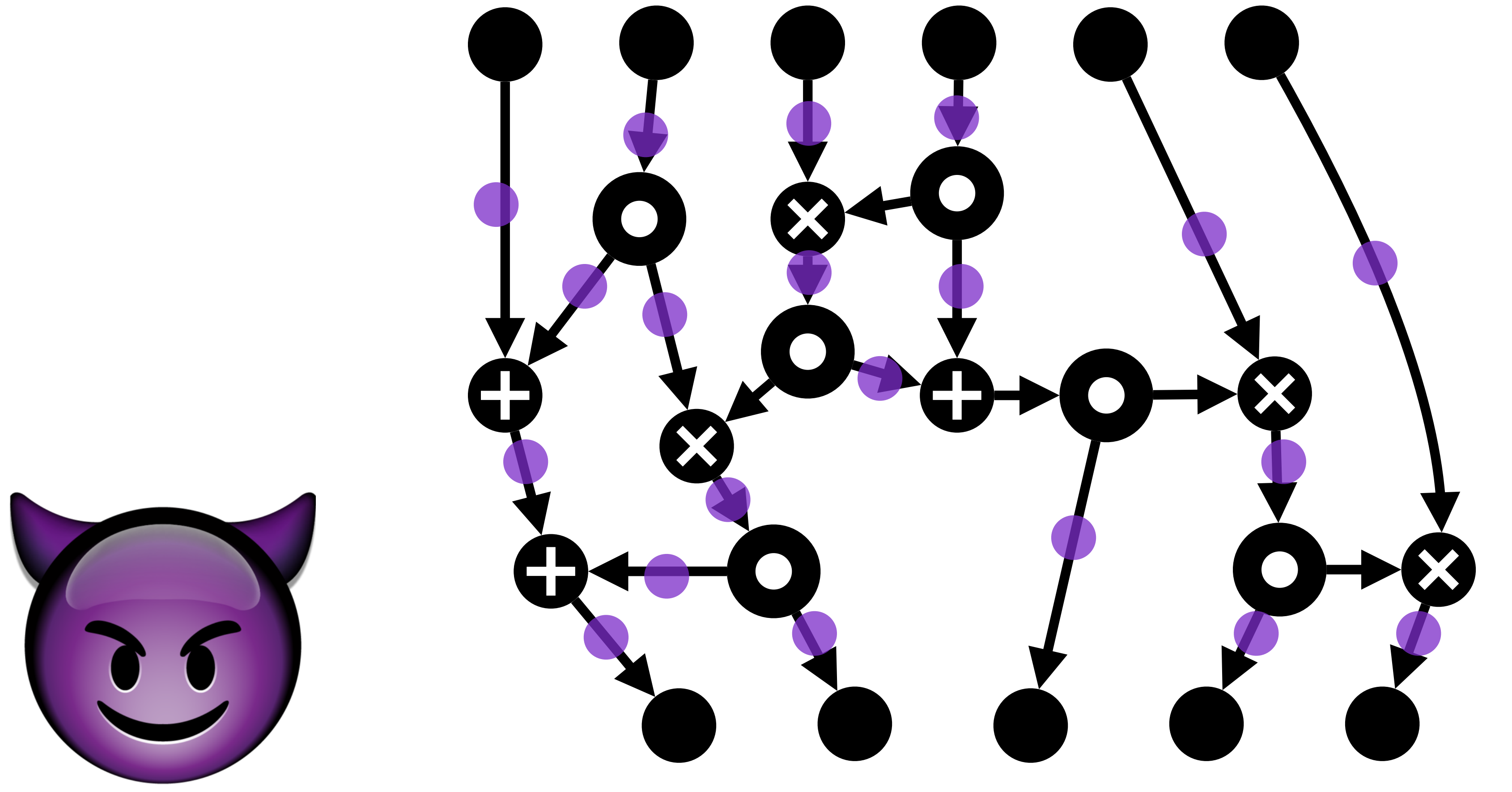
$$W \leftarrow \mathsf{LeakingWires}(\hat{C}, p)$$

2. Simulate the corresponding wire values

$$\mathsf{Sim} : W \mapsto \begin{cases} \text{perfect simulation} \\ \bot \text{ (abort)} \end{cases}$$

$$\delta_W = \begin{cases} 1 & \text{if } \mathsf{Sim}(W) = \bot \\ 0 & \text{otherwise} \end{cases}$$

Simulation with abort



1. Sample a set of leaking wires

$$W \leftarrow \mathsf{LeakingWires}(\hat{C}, p)$$

$$\begin{cases} w & \text{with proba } p \\ \perp & \text{with proba } 1-p \end{cases}$$

2. Simulate the corresponding wire values

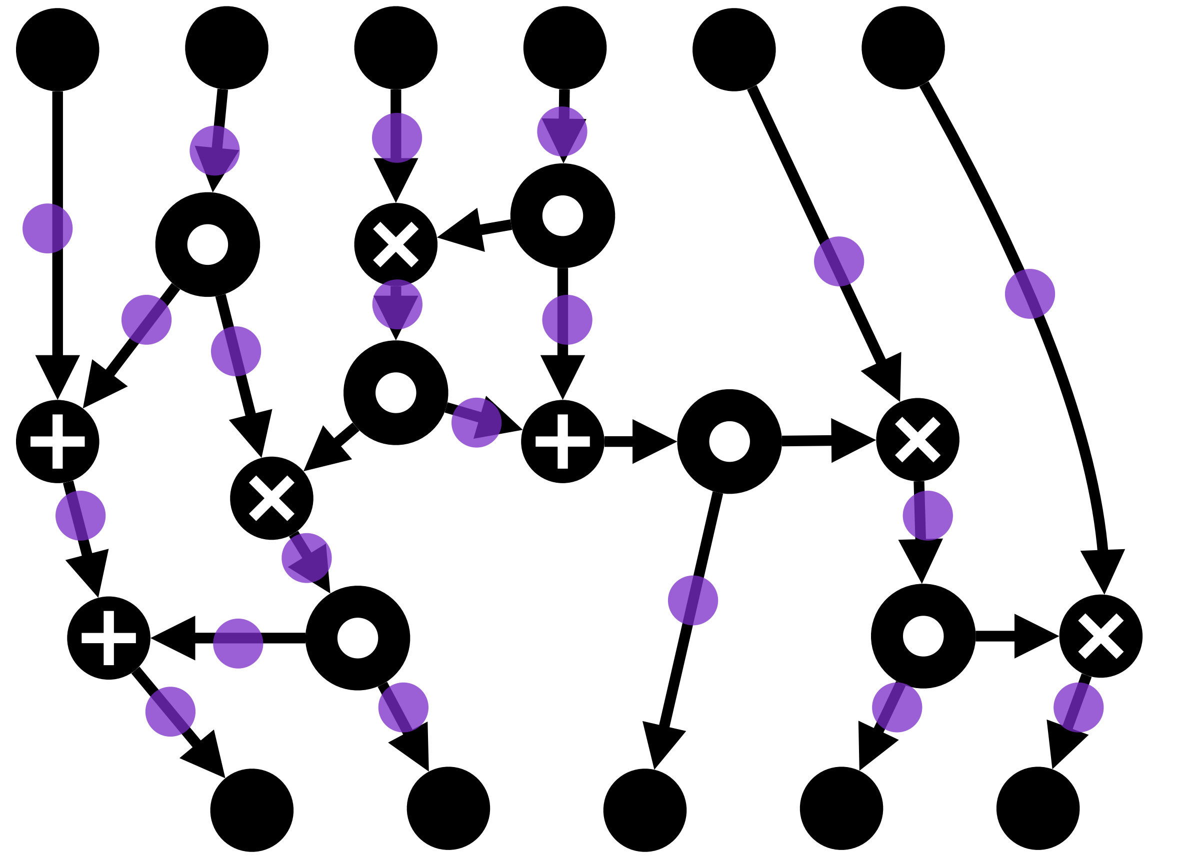
$$\mathsf{Sim} : W \mapsto \begin{cases} \text{perfect simulation} \\ \perp \text{ (abort)} \end{cases}$$

$$\delta_W = \begin{cases} 1 & \text{if } \mathsf{Sim}(W) = \perp \\ 0 & \text{otherwise} \end{cases}$$

- Failure probability

$$f(p) = \sum_W \delta_W p^{|W|}(1-p)^{s-|W|}$$

Simulation with abort



$$\begin{cases} w & \text{with proba } p \\ \bot & \text{with proba } 1-p \end{cases}$$

1. Sample a set of leaking wires

$$W \leftarrow \mathsf{LeakingWires}(\hat{C}, p)$$

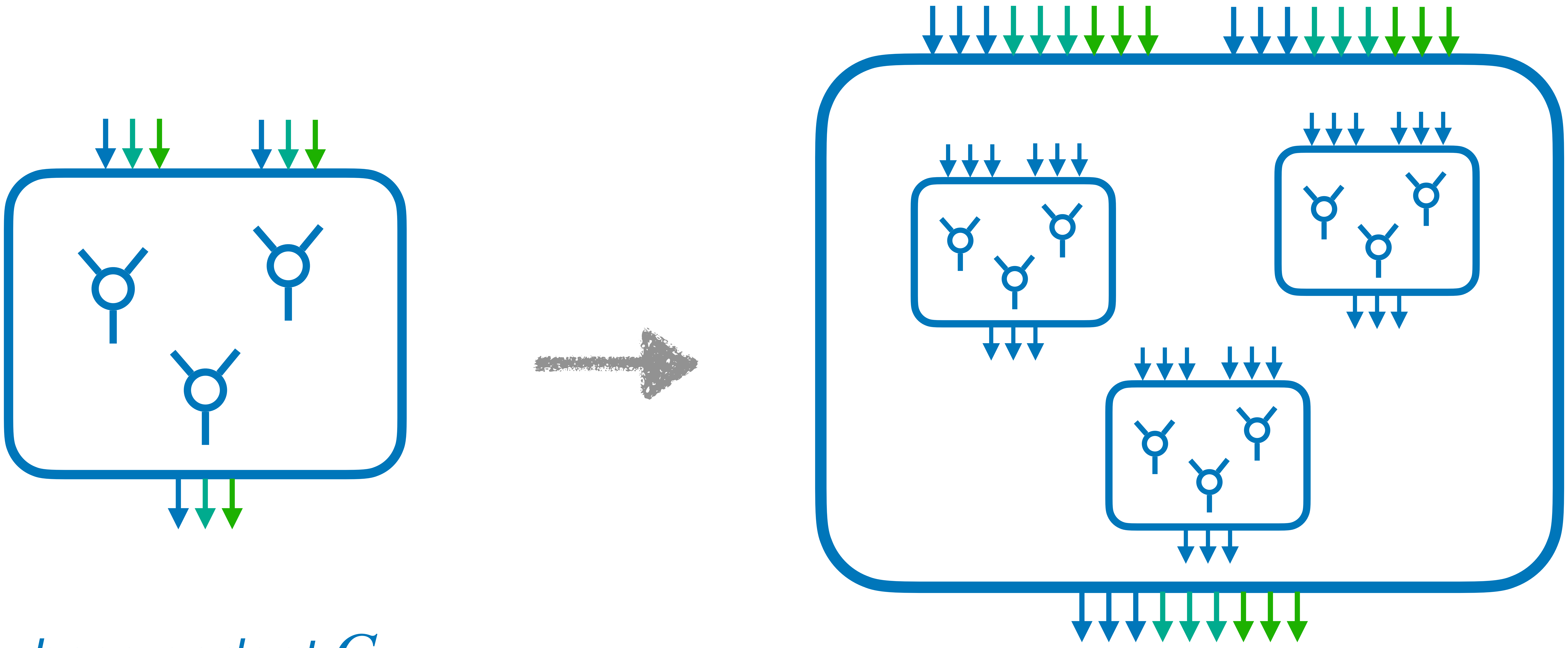
2. Simulate the corresponding wire values

$$\mathsf{Sim} : W \mapsto \begin{cases} \text{perfect simulation} \\ \bot \text{ (abort)} \end{cases} \qquad \delta_W = \begin{cases} 1 & \text{if } \mathsf{Sim}(W) = \bot \\ 0 & \text{otherwise} \end{cases}$$

- Failure probability

$$f(p) = \sum_W \delta_W p^{|W|}(1-p)^{s-|W|} \leq \sum_i c_i\, p^i$$

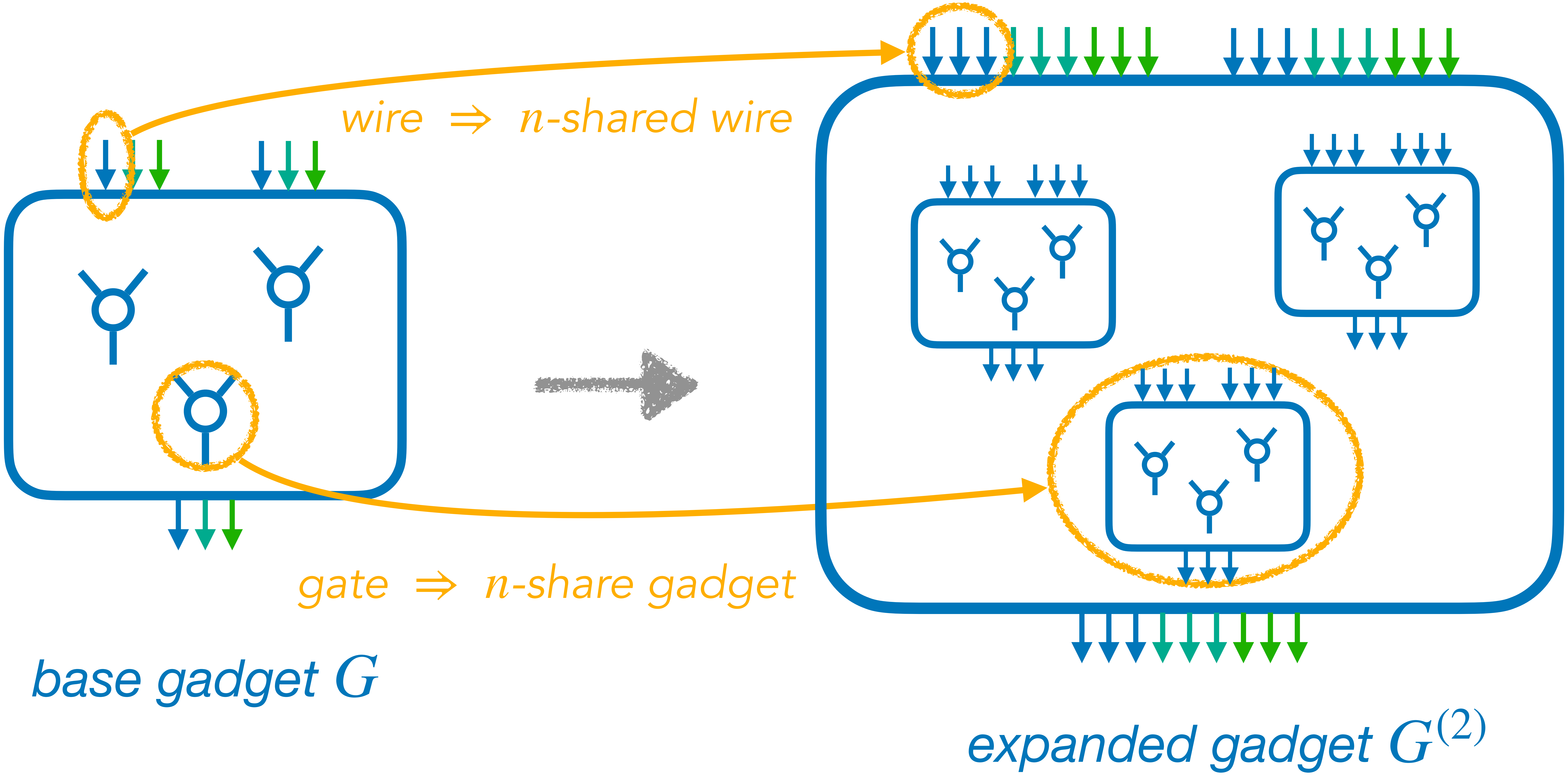
The expansion strategy



base gadget G

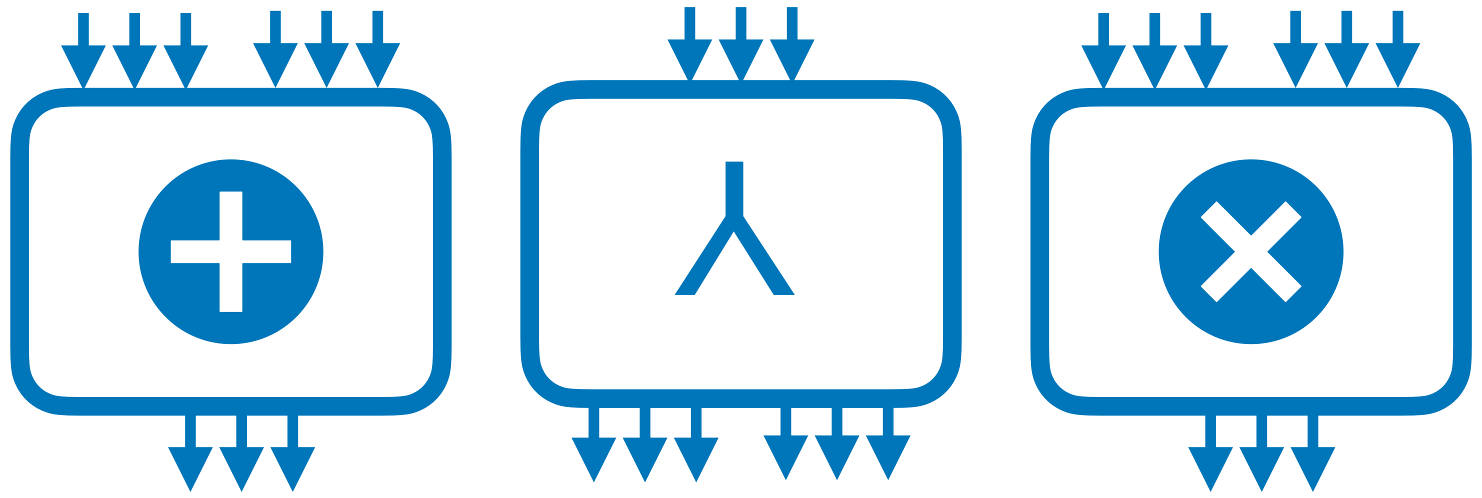
expanded gadget $G^{(2)}$

The expansion strategy



The expansion strategy

💡 Idea: bootstrap constant-size (small) gadgets

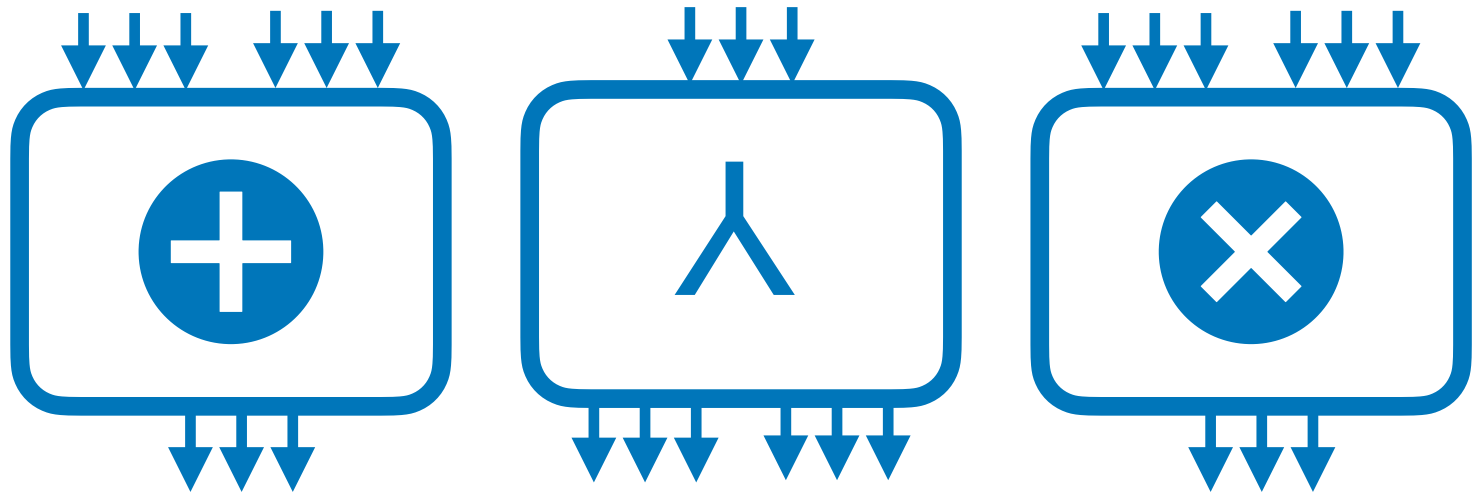


base gadgets

$$\{G\} \to \{G^{(2)}\} \to \cdots \to \{G^{(k)}\}$$

The expansion strategy

Idea: bootstrap constant-size (small) gadgets



base gadgets

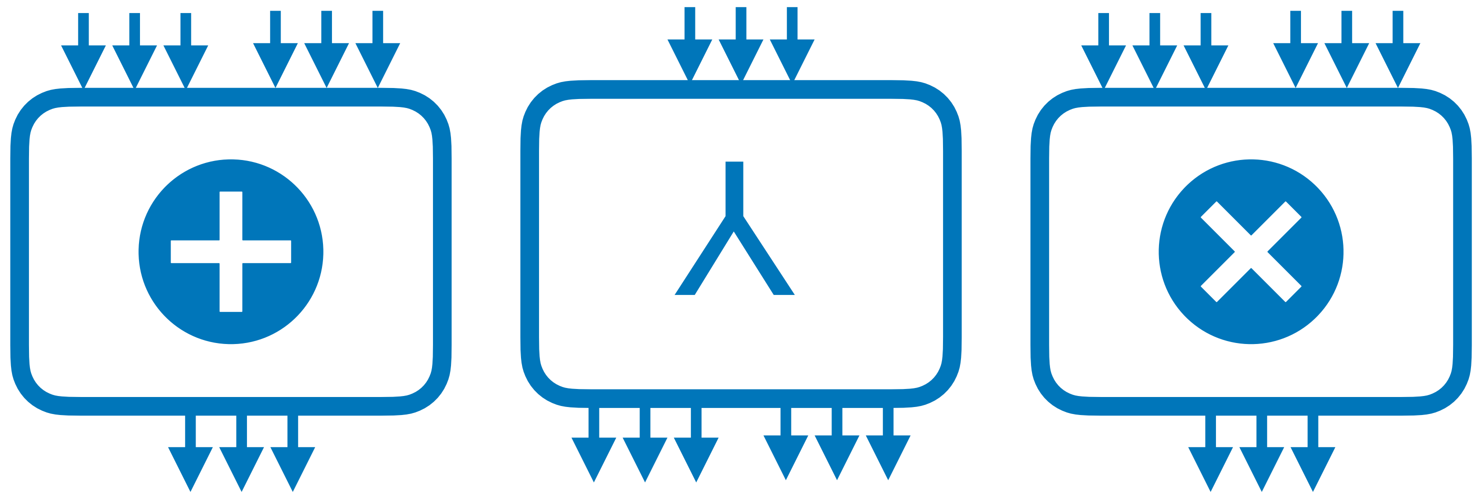
$$\{G\} \to \{G^{(2)}\} \to \cdots \to \{G^{(k)}\}$$

Goal: amplification of random probing security

$$p \longrightarrow f(p)$$

The expansion strategy

Idea: bootstrap constant-size (small) gadgets



base gadgets

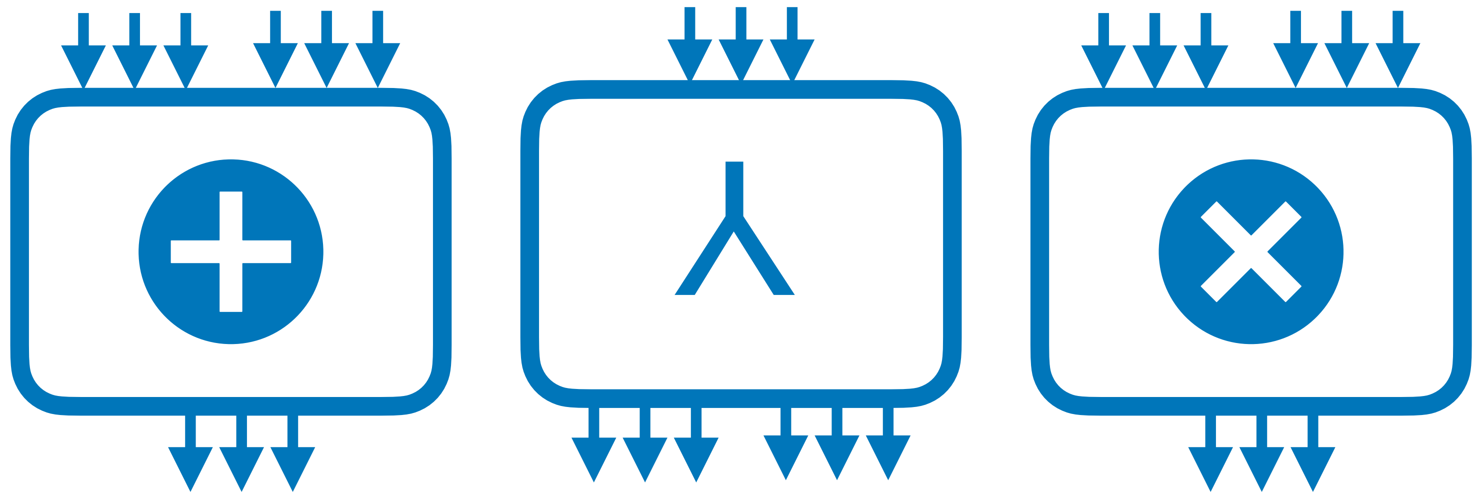
$$\{G\} \to \{G^{(2)}\} \to \cdots \to \{G^{(k)}\}$$

Goal: amplification of random probing security

$$p \longrightarrow f(p) \longrightarrow f(f(p))$$

The expansion strategy

Idea: bootstrap constant-size (small) gadgets



base gadgets

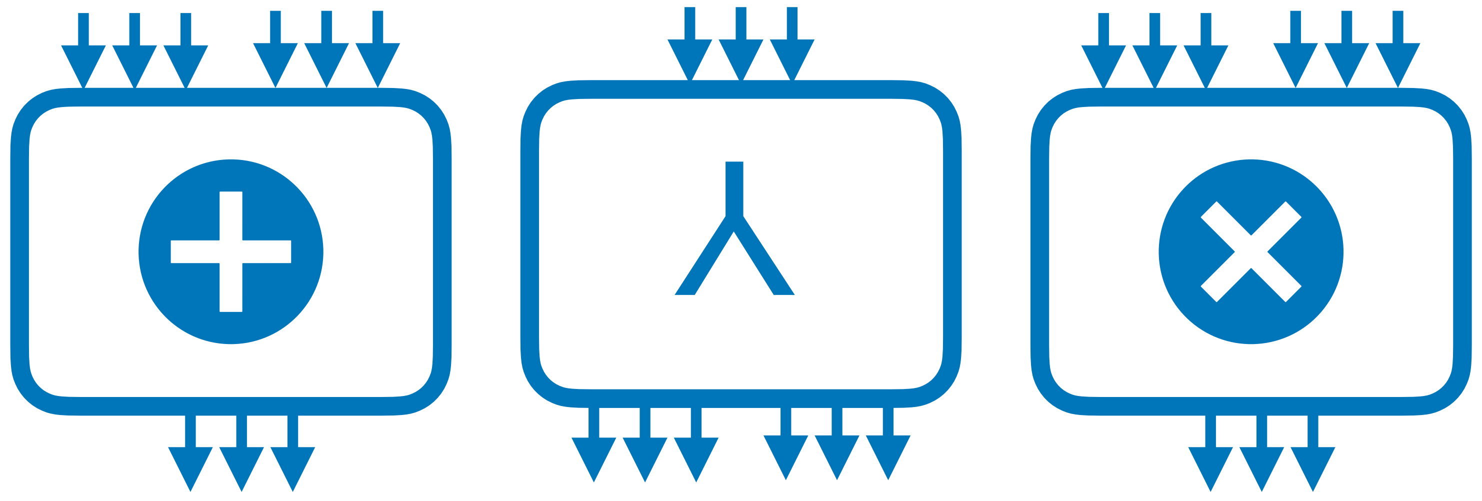
$$\{G\} \to \{G^{(2)}\} \to \cdots \to \{G^{(k)}\}$$

Goal: amplification of random probing security

$$p \longrightarrow f(p) \longrightarrow f(f(p)) \longrightarrow \cdots \longrightarrow f^{(k)}(p)$$

The expansion strategy

Idea: bootstrap constant-size (small) gadgets



base gadgets

$$\{G\} \to \{G^{(2)}\} \to \cdots \to \{G^{(k)}\}$$

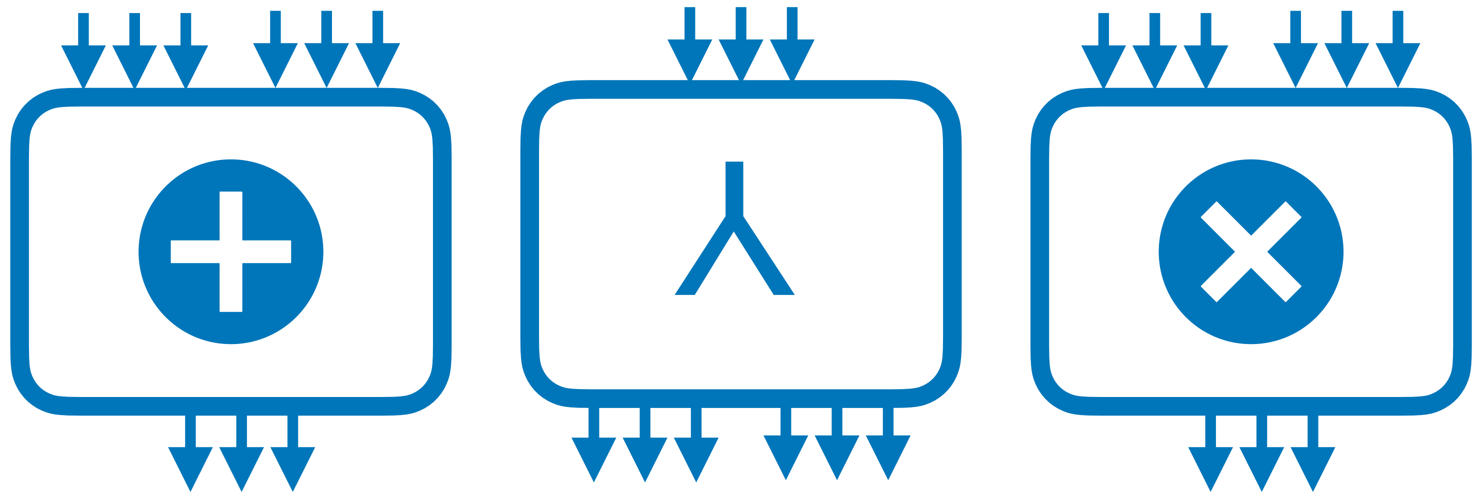
Goal: amplification of random probing security

$$p \longrightarrow f(p) \longrightarrow f(f(p)) \longrightarrow \cdots \longrightarrow f^{(k)}(p)$$

$$f(p) \leq \mathcal{O}(p^d)$$

The expansion strategy

Idea: bootstrap constant-size (small) gadgets



base gadgets

$$\{G\} \rightarrow \{G^{(2)}\} \rightarrow \cdots \rightarrow \{G^{(k)}\}$$

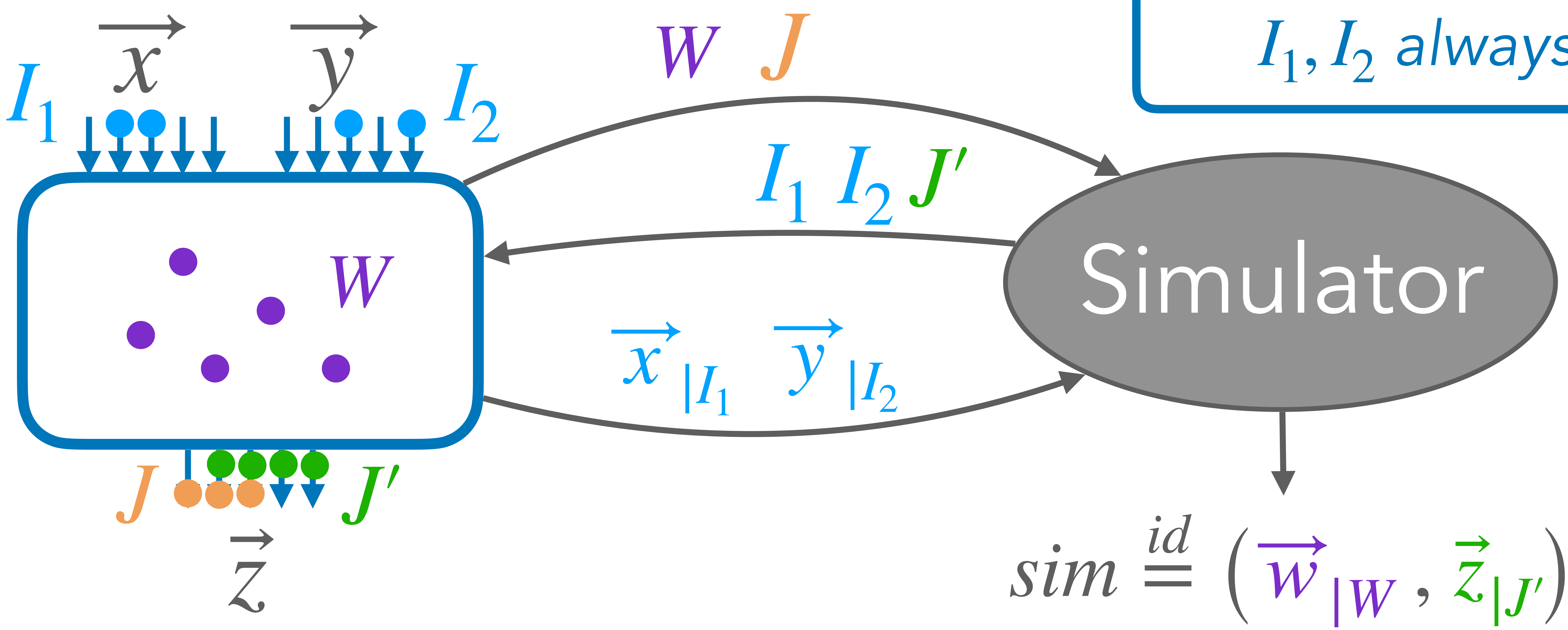
Goal: amplification of random probing security

$$p \longrightarrow f(p) \longrightarrow f(f(p)) \longrightarrow \cdots \longrightarrow f^{(k)}(p)$$

$$f(p) \leq \mathcal{O}(p^d)$$

$$\mathcal{O}\left(p^{d^k}\right)$$

Random probing expandability (RPE)



I_1, I_2 *always exist*

$sim \stackrel{id}{=} (\vec{w}_{|W}, \vec{z}_{|J'})$

RPE threshold t: $|J| \leq t$,
$(|I_1| > t$ *or* $|I_2| > t)$ = *simulation failure*

if $|J| > t$*, sim. can choose*
J' *s.t.* $|J'| = n - 1$

Random probing expandability (RPE)

Base gadgets $\{G\}$ f-RPE $\Rightarrow$ expanded gadgets $\{G^{(2)}\}$ $f^{(2)}$-RPE

$\Rightarrow$ expanded gadgets $\{G^{(k)}\}$ $f^{(k)}$-RPE

Random probing expandability (RPE)

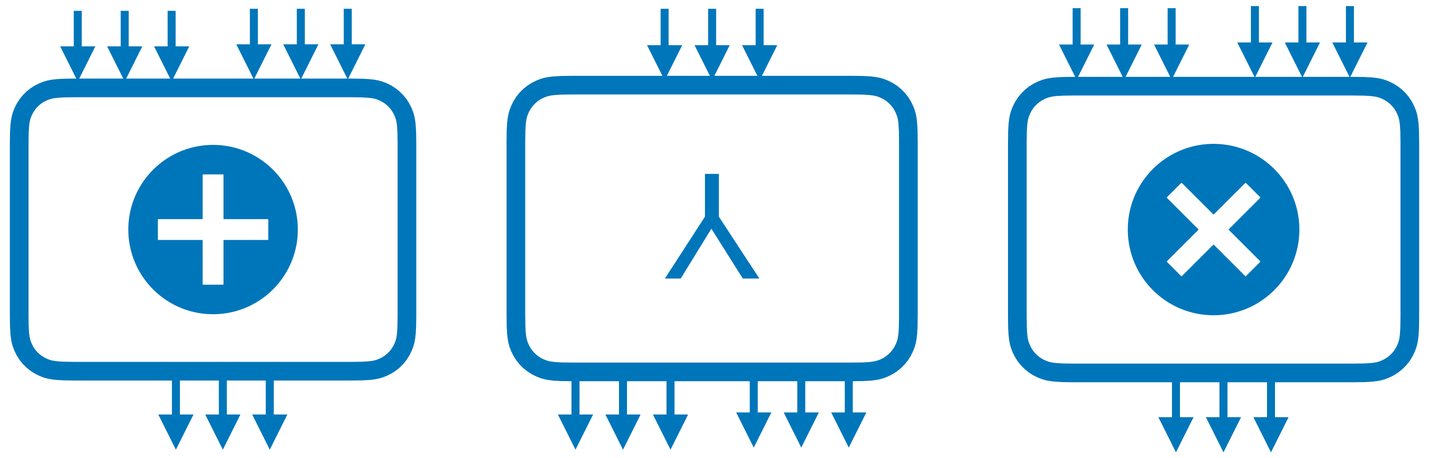
Base gadgets $\{G\}$ f-RPE $\Rightarrow$ expanded gadgets $\{G^{(2)}\}$ $f^{(2)}$-RPE

$\Rightarrow$ expanded gadgets $\{G^{(k)}\}$ $f^{(k)}$-RPE

$f^{(k)}(p)$ simulation security vs.

p-random probing leakage

Concrete instantiations

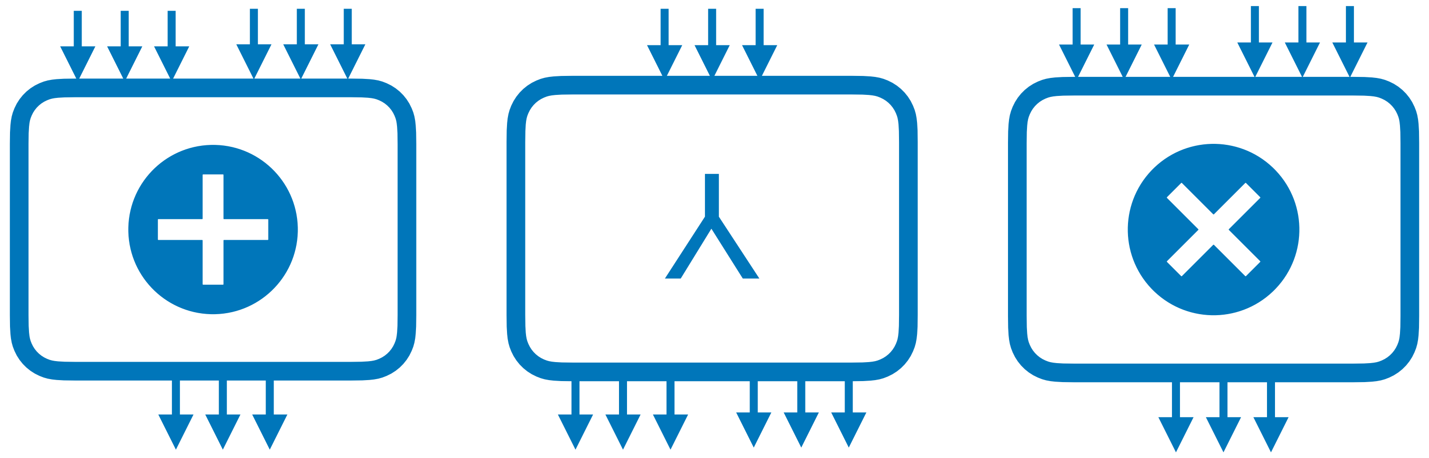


constant-size (small) gadgets

Your gadget is f-RPE

IronMask

Concrete instantiations



constant-size (small) gadgets

Your gadget is f-RPE

IronMask

Maximum tolerated leakage probability

$$p_{max} \in [0,1) \quad \text{such that} \quad f(p_{max}) < p_{max}$$

Concrete instantiations

3-share gadgets

$$G_{\mathsf{R}} : z_1 \leftarrow r_1 + x_1$$
$$z_2 \leftarrow r_2 + x_2$$
$$z_3 \leftarrow (r_1 + r_2) + x_3$$

$$\Rightarrow \quad \mathcal{O}(|C|\kappa^{3.9}) \ , \quad p_{max} = 2^{-7.5}$$

5-share gadgets

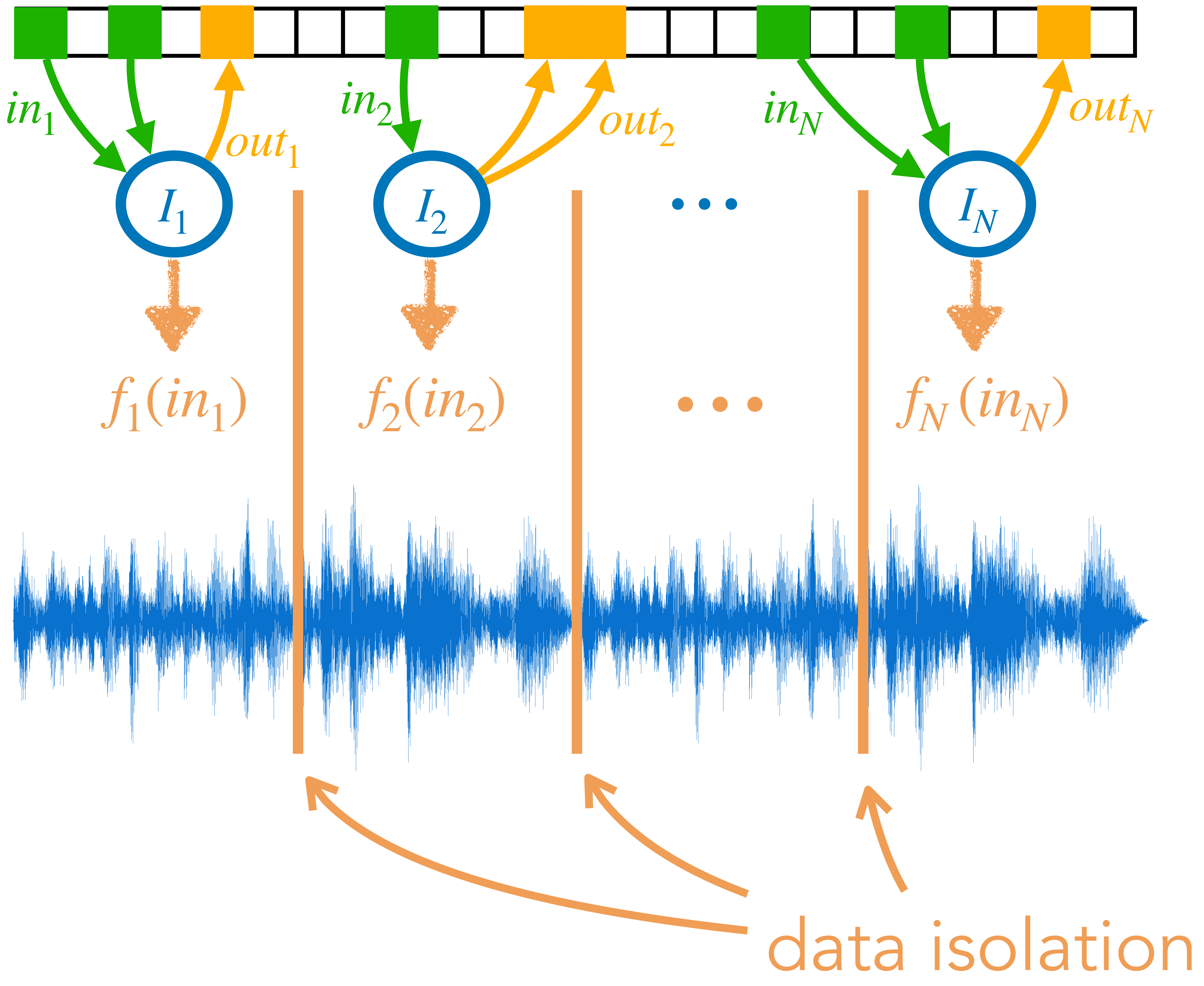
$$G_{\mathsf{R}} : z_1 \leftarrow (r_1 + r_2) + x_1$$
$$z_2 \leftarrow (r_2 + r_3) + x_2$$
$$z_3 \leftarrow (r_3 + r_4) + x_3$$
$$z_4 \leftarrow (r_4 + r_5) + x_4$$
$$z_5 \leftarrow (r_5 + r_1) + x_5$$

$$\Rightarrow \quad \mathcal{O}(|C|\kappa^{3.2}) \ , \quad p_{max} = 2^{-12}$$

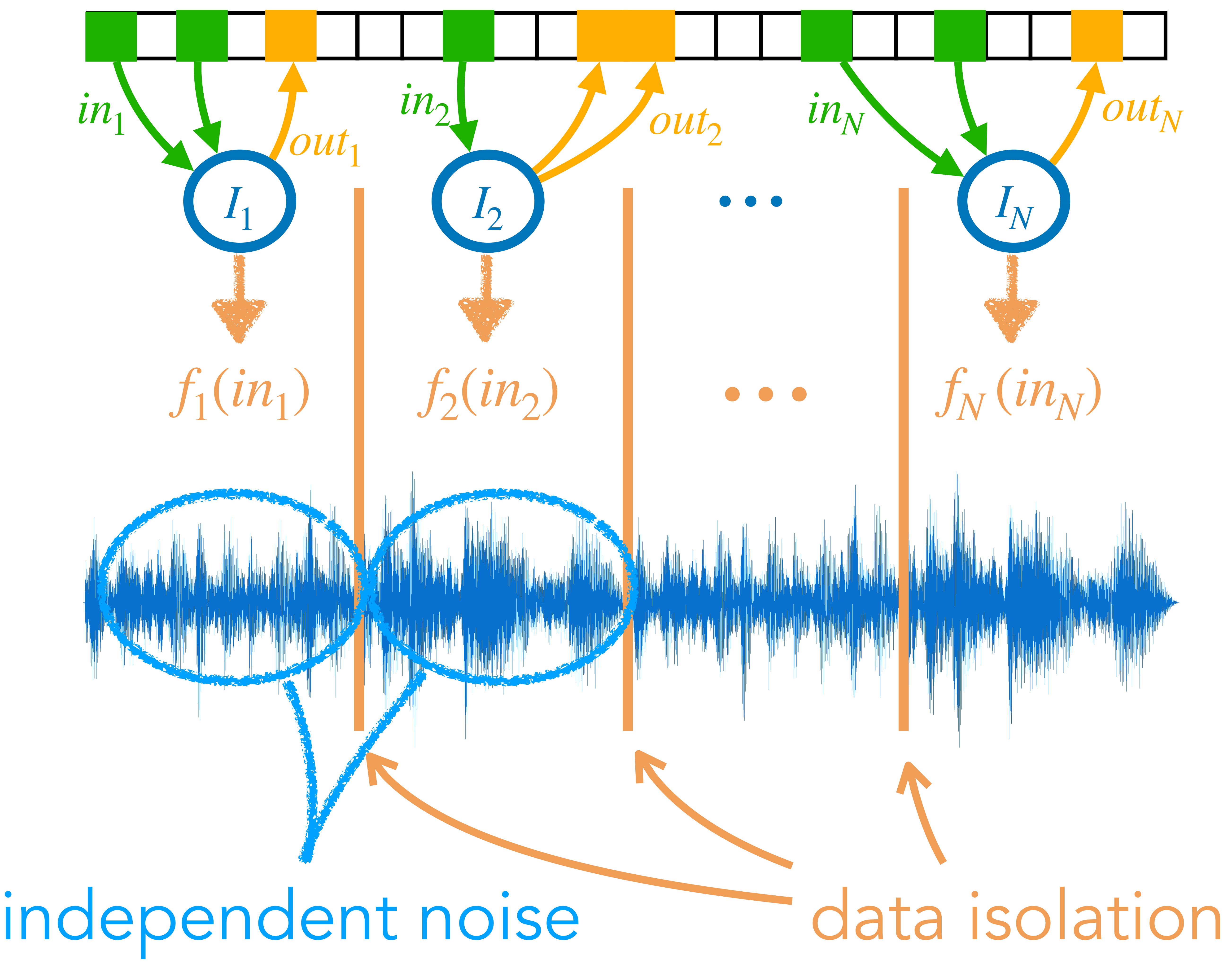
Conclusion

- [x] Provable security against side-channel attacks
- [ ] (Fully) bridging theory and practice

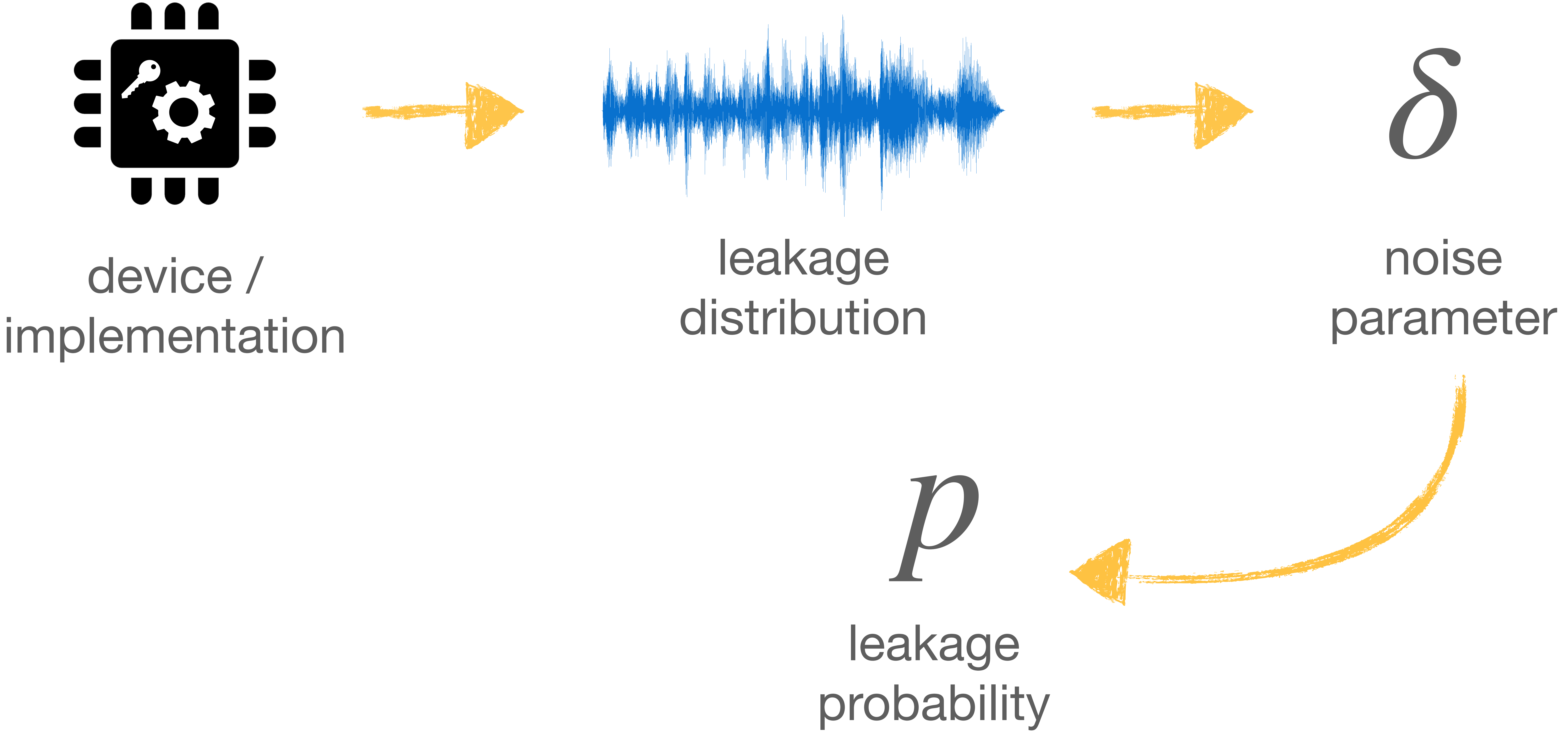
Physical assumptions



Physical assumptions



Noise parameters



Performances

3-share gadgets

$$G_R : z_1 \leftarrow r_1 + x_1$$
$$z_2 \leftarrow r_2 + x_2$$
$$z_3 \leftarrow (r_1 + r_2) + x_3$$

$\Rightarrow \mathcal{O}(|C|\kappa^{3.9})$, $p_{max} = 2^{-7.5}$

→ improved complexity

→ optimised implementations

5-share gadgets

$$G_R : z_1 \leftarrow (r_1 + r_2) + x_1$$
$$z_2 \leftarrow (r_2 + r_3) + x_2$$
$$z_3 \leftarrow (r_3 + r_4) + x_3$$
$$z_4 \leftarrow (r_4 + r_5) + x_4$$
$$z_5 \leftarrow (r_5 + r_1) + x_5$$

$\Rightarrow \mathcal{O}(|C|\kappa^{3.2})$, $p_{max} = 2^{-12}$

VeriSiCC Seminar 2022

Verification and Generation of Side-Channel Countermeasures

September 22, 2022, Paris

https://cryptoexperts.com/verisicc/seminaire-2022.html

Thank you!

🙏🤔⁉️🙋

References

- **Differential Power Analysis** — Paul C. Kocher, Joshua Jaffe, Benjamin Jun: Differential Power Analysis. CRYPTO 1999
- **Masking / soundness of masking with noise** — Suresh Chari, Charanjit S. Jutla, Josyula R. Rao, Pankaj Rohatgi: Towards Sound Approaches to Counteract Power-Analysis Attacks. CRYPTO 1999
- **Masking applied to DES** — Louis Goubin, Jacques Patarin: DES and Differential Power Analysis (The "Duplication" Method). CHES 1999
- **Probing model / ISW scheme** — Yuval Ishai, Amit Sahai, David A. Wagner: Private Circuits: Securing Hardware against Probing Attacks. CRYPTO 2003
- **"Only computation leaks" model** — Silvio Micali, Leonid Reyzin: Physically Observable Cryptography. TCC 2004

References

- **Noisy leakage model** — Emmanuel Prouff, Matthieu Rivain: Masking against Side-Channel Attacks: A Formal Security Proof. EUROCRYPT 2013
- **Unifying probing and noisy models** — Alexandre Duc, Stefan Dziembowski, Sebastian Faust: Unifying Leakage Models: From Probing Attacks to Noisy Leakage. EUROCRYPT 2014
- **Composition security for masking** — Gilles Barthe, Sonia Belaïd, François Dupressoir, Pierre-Alain Fouque, Benjamin Grégoire, Pierre-Yves Strub, Rébecca Zucchini: Strong Non-Interference and Type-Directed Higher-Order Masking. CCS 2016
- **Random probing expansion strategy** — Prabhanjan Ananth, Yuval Ishai, Amit Sahai: Private Circuits: A Modular Approach. CRYPTO 2018

References

- **Quasilinear masking**

  - Dahmun Goudarzi, Antoine Joux, Matthieu Rivain: How to Securely Compute with Noisy Leakage in Quasilinear Complexity. ASIACRYPT 2018
  - Dahmun Goudarzi, Thomas Prest, Matthieu Rivain, Damien Vergnaud: Probing Security through Input-Output Separation and Revisited Quasilinear Masking. IACR TCHES 2021

- **Random probing expandability**

  - Sonia Belaïd, Jean-Sébastien Coron, Emmanuel Prouff, Matthieu Rivain, Abdul Rahman Taleb: Random Probing Security: Verification, Composition, Expansion and New Constructions. CRYPTO 2020
  - Sonia Belaïd, Matthieu Rivain, Abdul Rahman Taleb: On the Power of Expansion: More Efficient Constructions in the Random Probing Model. EUROCRYPT 2021

References

- **IronMask tool** — Sonia Belaïd, Darius Mercadier, Matthieu Rivain, Abdul Rahman Taleb: IronMask: Versatile Verification of Masking Security. IEEE S&P 2022