Transistor

a TFHE-friendly stream cipher

Jules Baudrin, Sonia Belaïd, Nicolas Bon, Christina Boura, Anne Canteaut, Gaëtan Leurent, Pascal Paillier, Léo Perrin, Matthieu Rivain, Yann Rotella, and Samuel Tap

CRYPTO 2025 - 19th August 2025

https://eprint.iacr.org/2025/282

Table of Content

Part 1

FHE and transciphering

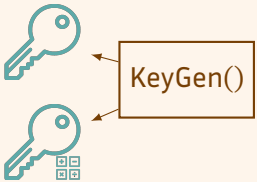
Fully Homomorphic Encryption

Client

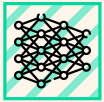
Server

Fully Homomorphic Encryption

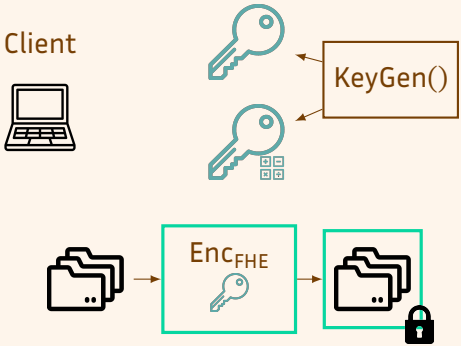
Client



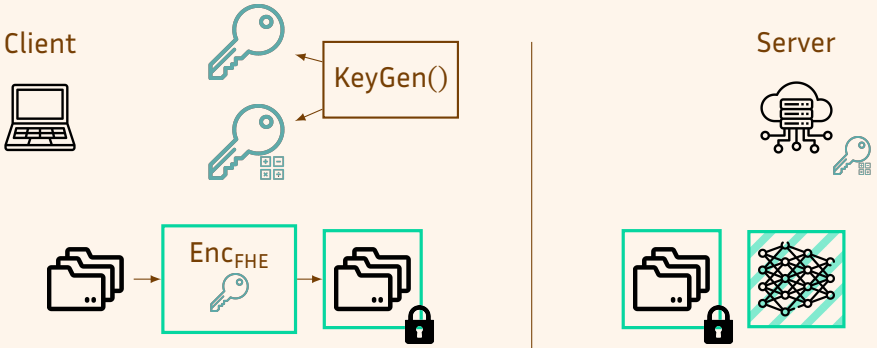
Server



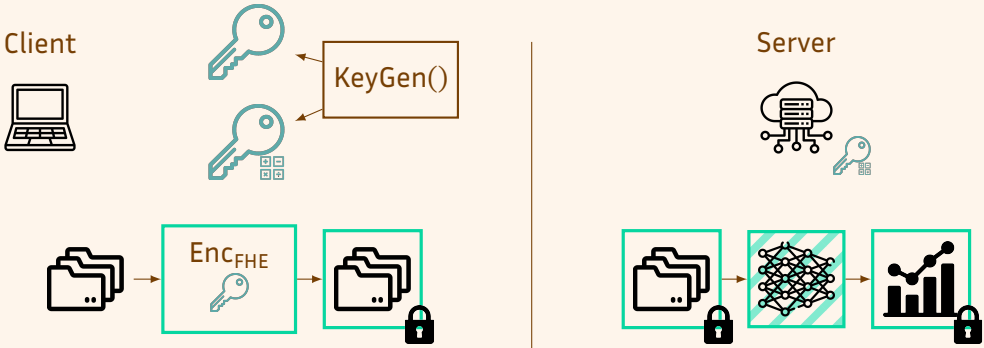
Fully Homomorphic Encryption



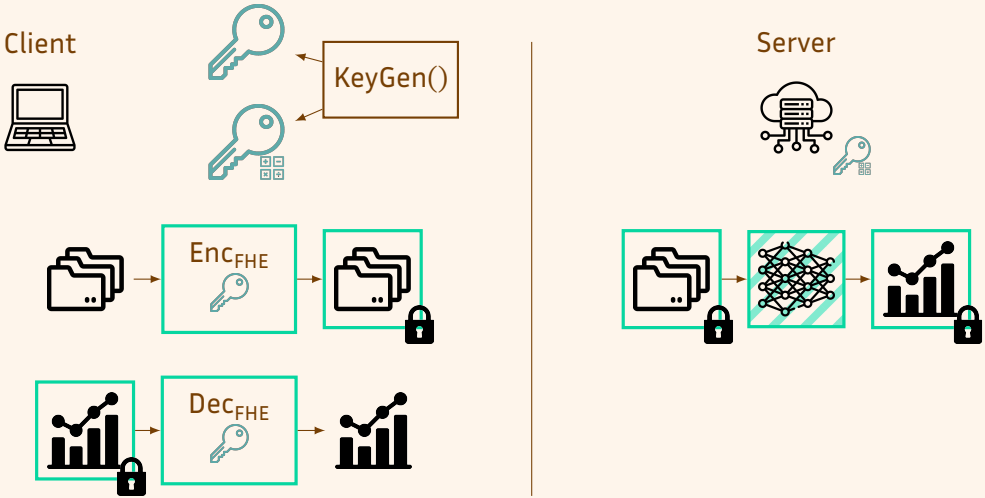
Fully Homomorphic Encryption



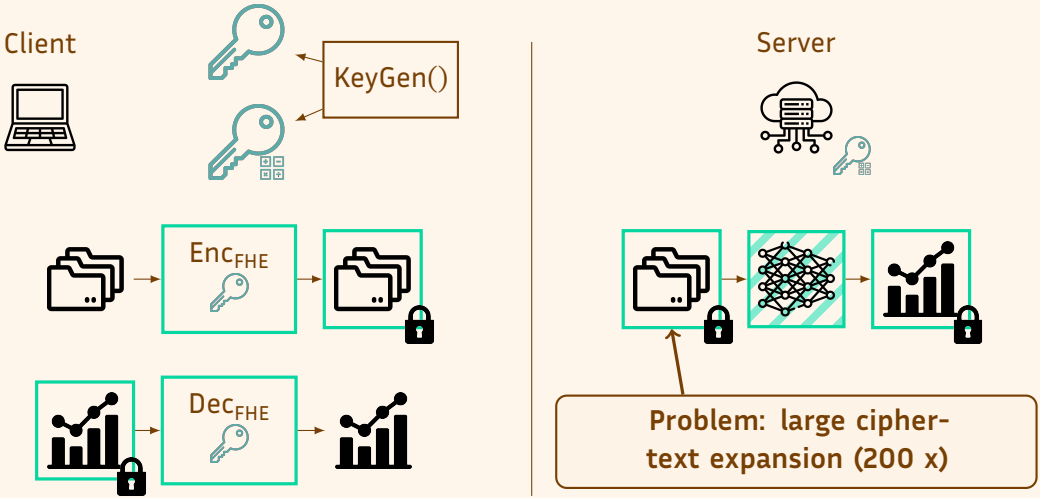
Fully Homomorphic Encryption



Fully Homomorphic Encryption



Fully Homomorphic Encryption



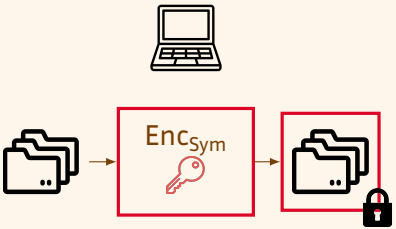
A solution: Transciphering

Client

Server

A solution: Transciphering

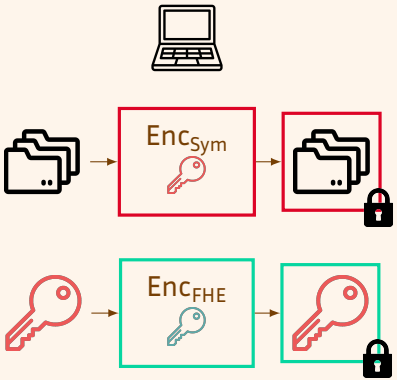
Client



Server

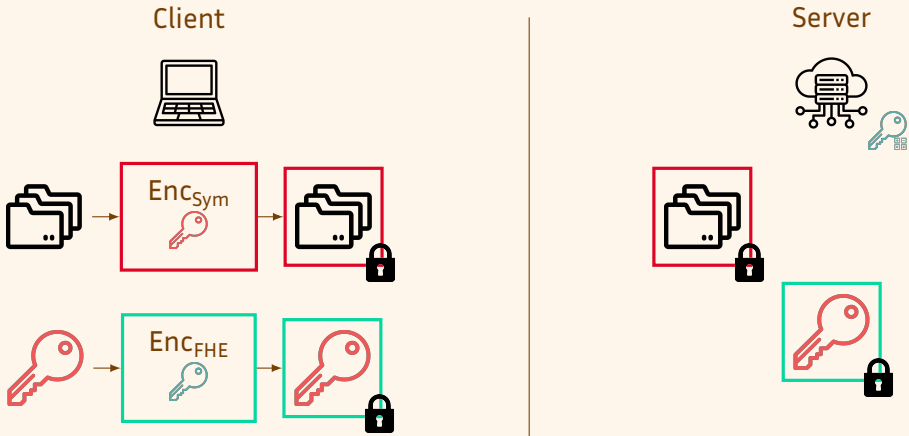
A solution: Transciphering

Client

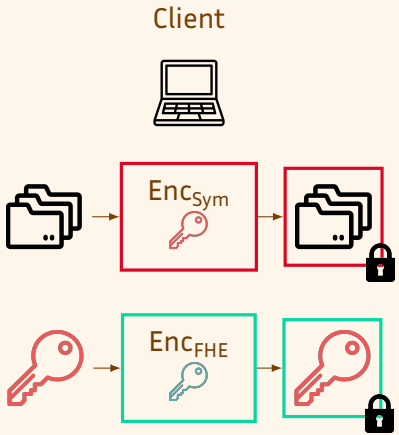


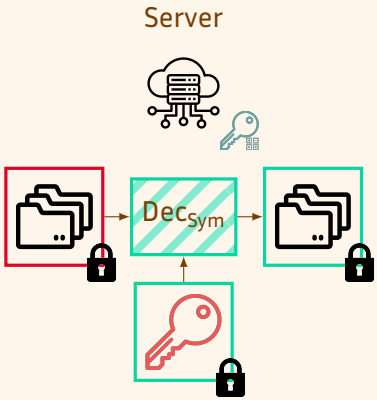
Server

A solution: Transciphering

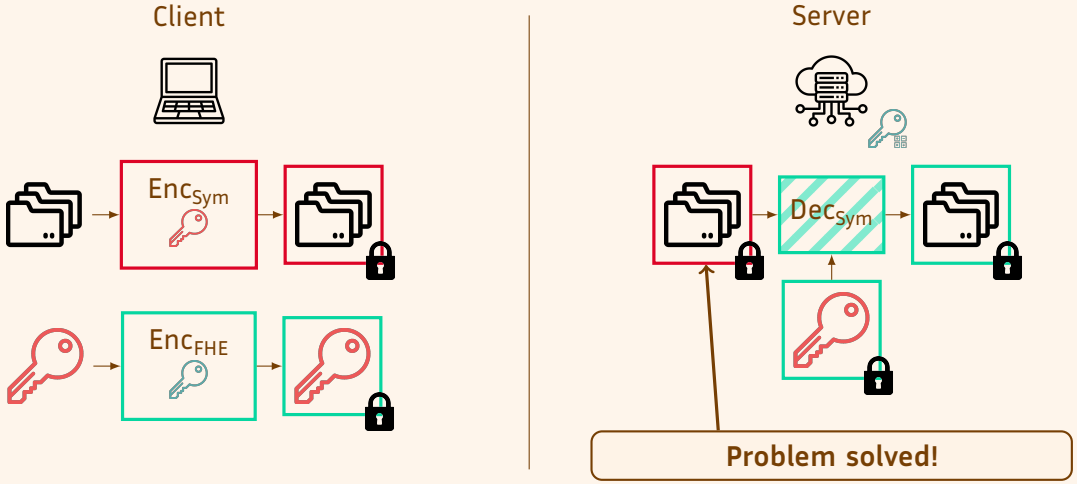


A solution: Transciphering





A solution: Transciphering



Which symmetric cipher ?

A standard one ?

- Evaluation of AES [GHS12] is too slow due to large Sbox size
- Lightweight ciphers such as ASCON, PRESENT, ... more promising, but still not fast enough

Which symmetric cipher ?

A standard one ?

- Evaluation of AES [GHS12] is too slow due to large Sbox size
- Lightweight ciphers such as ASCON, PRESENT, ... more promising, but still not fast enough

Or a FHE-tailored one ?

- Active line of research [BOS23, AGHM24, DJL+24, CCH+24]
- Nonstandard design choices: leads to weaker confidence in security

Which symmetric cipher ?

A standard one ?

- Evaluation of AES [GHS12] is too slow due to large Sbox size
- Lightweight ciphers such as ASCON, PRESENT, ... more promising, but still not fast enough

Or a FHE-tailored one ?

- Active line of research [BOS23, AGHM24, DJL+24, CCH+24]
- Nonstandard design choices: leads to weaker confidence in security

With `Transistor`, we look for the best of both worlds (fast in FHE and secure)

Part 2

TFHE and its specifies

Basics on TFHE

Basics on TFHE

- Plaintext space : $\mathbb{Z}_p$ with p of a few bits.

Basics on TFHE

- Plaintext space : $\mathbb{Z}_p$ with p of a few bits.
- Secret Key:

$$\vec{s} = (s_0, \dots, s_{n-1}) \in \{0,1\}^n.$$

Basics on TFHE

- Plaintext space : $\mathbb{Z}_p$ with p of a few bits.
- Secret Key:

$$\vec{s} = (s_0, \dots, s_{n-1}) \in \{0,1\}^n.$$

- Encryption:

$$\vec{c} = (a_0, \dots, a_{n-1}, b) \in \mathbb{Z}_q^{n+1}$$

with:

$$b = \langle \vec{a}, \vec{s} \rangle + \frac{q}{p} \cdot m + e.$$

$\vec{a}$ is random and e is a small Gaussian noise.

Homomorphisms in TFHE

Homomorphisms in TFHE

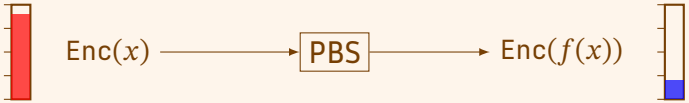
- Linear homomorphisms: Very fast, but **increase the noise**.
  - Sum of ciphertexts
  - Multiplication by a constant

Homomorphisms in TFHE

- Linear homomorphisms: Very fast, but **increase the noise**.
  - Sum of ciphertexts
  - Multiplication by a constant
- Programmable Bootstrapping (PBS): Very slow. Gets even slower as p increases.
  - **Resets** the noise to a nominal level
  - Evaluates a **Look-Up Table** from $\mathbb{Z}_p$ to $\mathbb{Z}_p$.

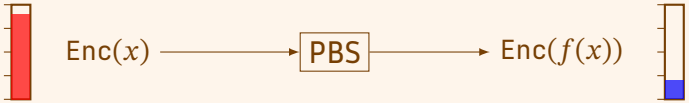
Homomorphisms in TFHE

- Linear homomorphisms: Very fast, but **increase the noise**.
  - Sum of ciphertexts
  - Multiplication by a constant
- Programmable Bootstrapping (PBS): Very slow. Gets even slower as p increases.



Homomorphisms in TFHE

- Linear homomorphisms: Very fast, but **increase the noise**.
  - Sum of ciphertexts
  - Multiplication by a constant
- Programmable Bootstrapping (PBS): Very slow. Gets even slower as p increases.



- **Negacyclicity problem:**
  - If p is **even**, restricts the functions that can be evaluated.
  - Disappears when p is **odd**

Our wishlist for a TFHE-friendly cipher

Our wishlist for a TFHE-friendly cipher

- A small prime field of odd characteristic...

Our wishlist for a TFHE-friendly cipher

- A small prime field of odd characteristic...
  to avoid the negacyclicity problem and ease the design.

Our wishlist for a TFHE-friendly cipher

- A small prime field of odd characteristic...
  to avoid the negacyclicity problem and ease the design.
- As little non-linear operations as possible...

Our wishlist for a TFHE-friendly cipher

- A small prime field of odd characteristic...
  to avoid the negacyclicity problem and ease the design.
- As little non-linear operations as possible...
  to be fast (low number of PBS).

Our wishlist for a TFHE-friendly cipher

- A small prime field of odd characteristic...
  to avoid the negacyclicity problem and ease the design.
- As little non-linear operations as possible...
  to be fast (low number of PBS).
- A controlled noise growth...

Our wishlist for a TFHE-friendly cipher

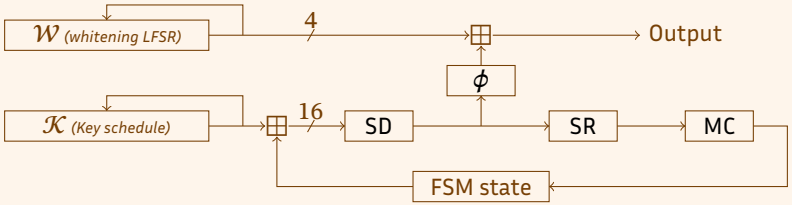
- A small prime field of odd characteristic...
  to avoid the negacyclicity problem and ease the design.
- As little non-linear operations as possible...
  to be fast (low number of PBS).
- A controlled noise growth...
  to guarantee the correctness of the computations.

Part 3

Description of Transistor

Design of Transistor

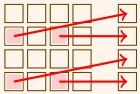
Prime field: $\mathbb{F}_{17}$



(a) SD.

(b) SR.

(c) MC.



(d) ϕ.

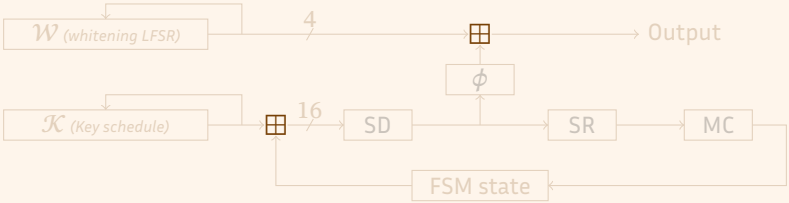
Design of Transistor

Our wishlist:

- A small prime field of odd characteristic
- As little non-linear operations as possible
- A controlled noise growth

Design of Transistor

Prime field: $\mathbb{F}_{17}$



(a) SD.

(b) SR.

(c) MC.

(d) ϕ.

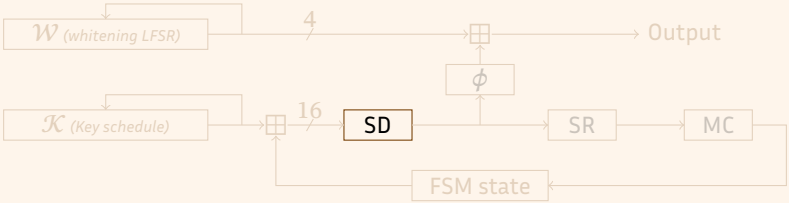
Design of Transistor

Our wishlist:

- A small prime field of odd characteristic
- As little non-linear operations as possible
- A controlled noise growth

Design of Transistor

Prime field: $\mathbb{F}_{17}$



(a) SD.

(b) SR.

(c) MC.

(d) ϕ.

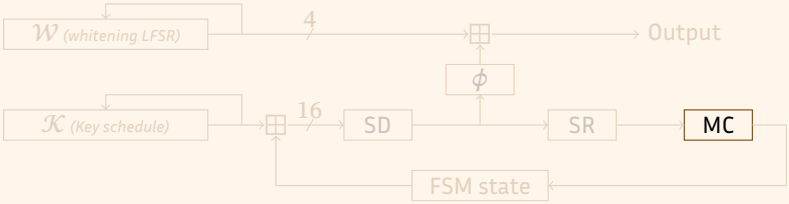
Design of Transistor

Our wishlist:

- A small prime field of odd characteristic
- As little non-linear operations as possible
- A controlled noise growth

Design of Transistor

Prime field: $\mathbb{F}_{17}$



(a) SD.

(b) SR.

(c) MC.

(d) ϕ.

MixColumns

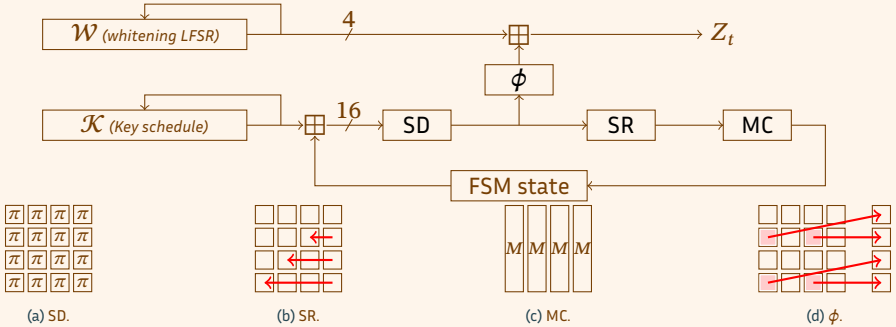
The matrix we chose for MixColumns is:

$$M = \begin{bmatrix} 2 & 1 & 1 & 1 \\ 1 & -1 & 1 & -2 \\ 1 & 1 & -2 & -1 \\ 1 & -2 & -1 & 1 \end{bmatrix}.$$

- Matrix MDS to ensure optimal diffusion,
- Symmetric,
- Minimal ℓ_2-norm of 7 $\rightarrow$ important for noise management.

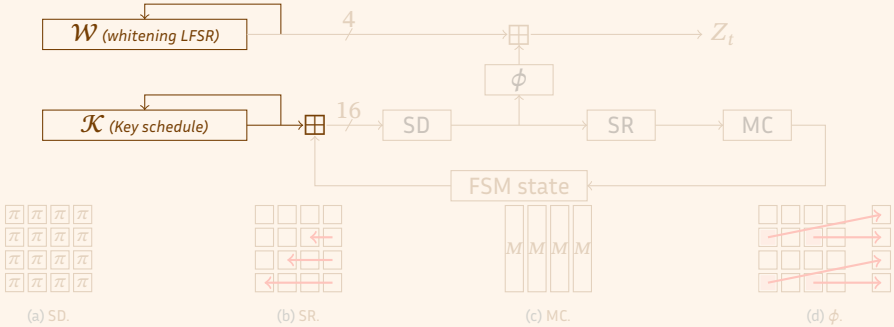
Design of Transistor

Prime field: $\mathbb{F}_{17}$

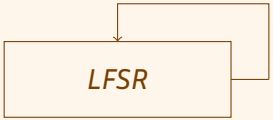


Design of Transistor

Prime field: $\mathbb{F}_{17}$

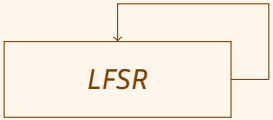


Silent LFSR



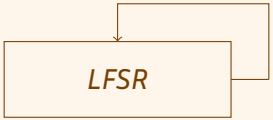
- Naive approach : linearly update the state at each clock.

Silent LFSR



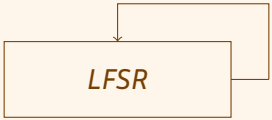
- Naive approach : linearly update the state at each clock.
- Problem: the noise accumulates over time.

Silent LFSR



- Naive approach : linearly update the state at each clock.
- Problem: the noise accumulates over time.
- Solution: Computing on the fly the coefficients of the linear combination in clear

Silent LFSR



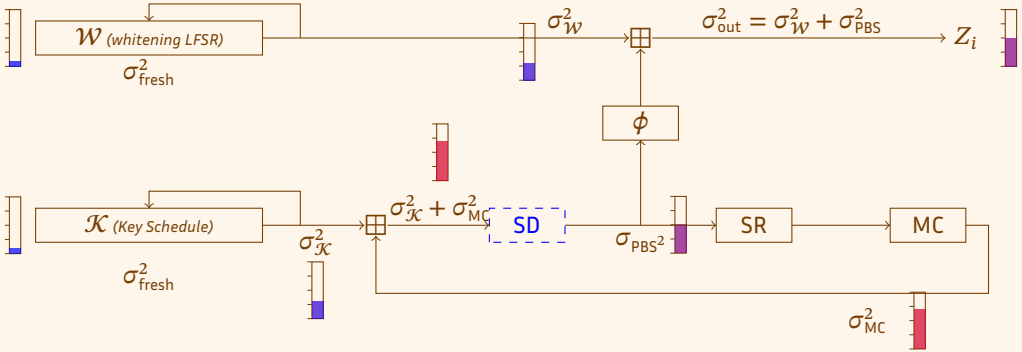
- Naive approach : linearly update the state at each clock.
- Problem: the noise accumulates over time.
- Solution: Computing on the fly the coefficients of the linear combination in clear

The noise variance in the output of the silent LFSR remains stable over time, without using any PBS.

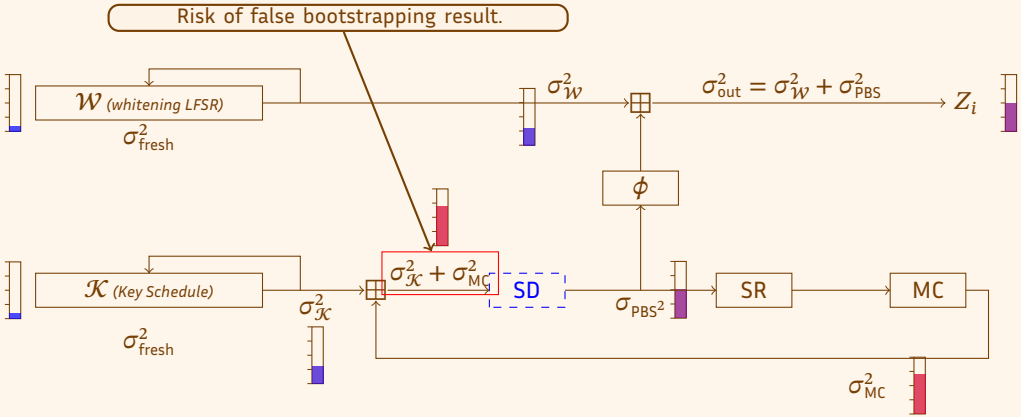
Part 4

Noise Management

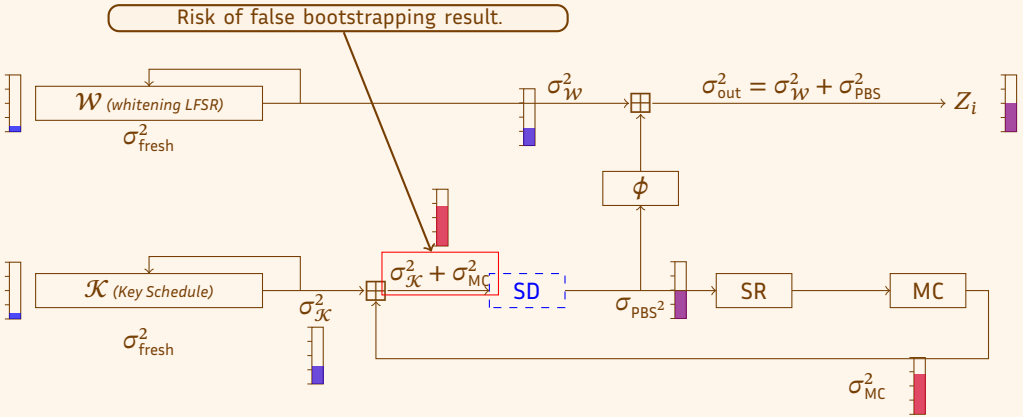
Noise Management



Noise Management

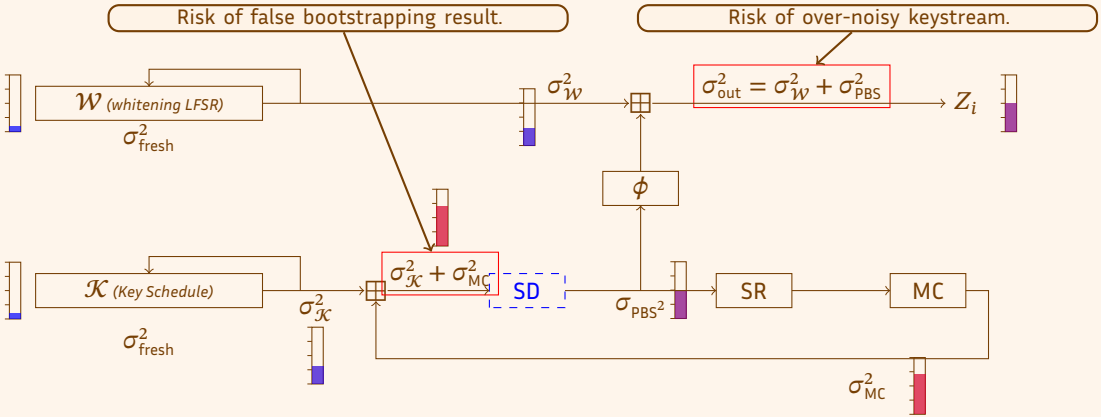


Noise Management



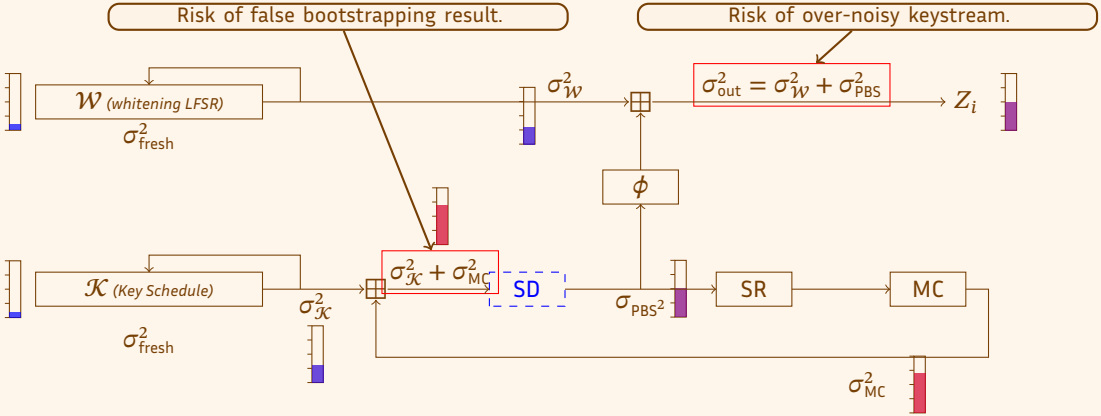
Takeaway 1: No Restriction on the size of the LFSRs (since $\sigma^2_{\text{MC}} \gg \sigma^2_{\mathcal{K}}$ and $\sigma^2_{PBS} \gg \sigma^2_{\mathcal{W}}$)

Noise Management



Takeaway 1: No Restriction on the size of the LFSRs (since $\sigma^2_{\text{MC}} \gg \sigma^2_{\mathcal{K}}$ and $\sigma^2_{PBS} \gg \sigma^2_{\mathcal{W}}$)

Noise Management



Takeaway 1: No Restriction on the size of the LFSRs (since $\sigma^2_{\text{MC}} \gg \sigma^2_{\mathcal{K}}$ and $\sigma^2_{PBS} \gg \sigma^2_{\mathcal{W}}$)

Takeaway 2 : Dimensioning the TFHE parameters can be reduced to select parameters for a simple PBS

Part 5

Cryptanalysis

TMDTO and Guess & Determine

TMDTO and Guess & Determine

Time-Memory Data Trade-Offs: Dimensions of the LFSRs: $|\mathcal{K}| = 64$ and $|\mathcal{W}| = 32$ elements of $\mathbb{F}_{17}$. Ensures a limit on the keystream of 2^{31} digits with a single key.

TMDTO and Guess & Determine

Time-Memory Data Trade-Offs: Dimensions of the LFSRs: $|\mathcal{K}| = 64$ and $|\mathcal{W}| = 32$ elements of $\mathbb{F}_{17}$. Ensures a limit on the keystream of 2^{31} digits with a single key.

Guess-and-Determine: The filtering procedure of `Transistor` shows that the attacker has to guess the content of the whithening LFSR and $\frac{12}{16}|\mathcal{W}|$ digits, leading to a complexity:

$$p^{\frac{12}{16}|\mathcal{K}|+|\mathcal{W}|} \approx 2^{294}.$$

Correlation Attacks against `Transistor`

Correlation Attacks against `Transistor`

We prove that:

Three consecutive outputs are statistically independent from the secret key.

Correlation Attacks against `Transistor`

We prove that:

Three consecutive outputs are statistically independent from the secret key.

Consequence: The correlation of linear relations between the content of key-LFSR and output 4-digits sequence is very low:

$$\left|C^{(4)}(\alpha,\beta)\right|^2 \leq 2^{-35.98}$$

Main arguments of the proof:

- Amount of active S-boxes over n rounds,
- Modulus of the Fourier coefficients of the S-box.

Correlation Attacks against `Transistor`

We prove that:

Three consecutive outputs are statistically independent from the secret key.

Consequence: The correlation of linear relations between the content of key-LFSR and output 4-digits sequence is very low:

$$\left|C^{(4)}(\alpha,\beta)\right|^2 \leq 2^{-35.98}$$

Main arguments of the proof:

- Amount of active S-boxes over n rounds,
- Modulus of the Fourier coefficients of the S-box.

Any correlation attack based on the span of a linear trail requires $2^{41.5}$ digits of the output sequence.

And more!

More analysis in the paper about:

- Linear Distinguishers on the keystream,
- Algebraic attacks.

Part 6

Performances

Performances

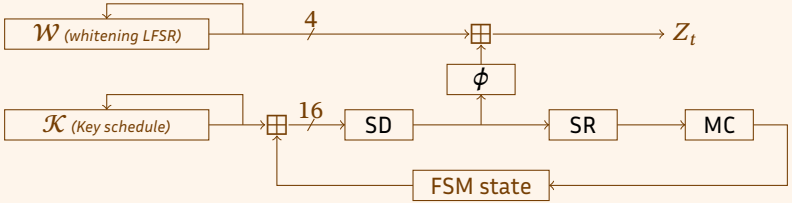
Cipher	Setup	Latency	Throughput	Communication Cost[a]	p_{err}
`Trivium` [BOS23] (128 thr.)	2259 ms	121 ms	529 bits/s	640 B + 35.6 MB †	2^{-40}
`Kreyvium` [BOS23] (128 thr.)	2883 ms	150 ms	427 bits/s	1024 B + 35.6 MB †	2^{-40}
`Margrethe` [AGHM24]	No	27.2 ms	147.06 bits/s	64 MB *	$< 2^{-1000}$
	No	54.2 ms	73.8 bits/s	128 MB *	$< 2^{-1000}$
PRF-based construction [DJL+24]	No	5.675 ms	881 bits/s	32.8 MB = 8.9 MB + 23.9 MB	2^{-64}
FRAST [CCH+24]	25 s (8 thr.)	6.2 s	20.66 bits/s	34.05 MB = 148 KB + 33.91 MB	2^{-80}
`Transistor`	No	251 ms	65.10 bits/s	13.54 MB = 780 B + 12.78 MB	2^{-128}

[a] Includes size of encrypted symmetric key + size of evaluation keys. † Values recomputed from the data of the papers. For consistency's sake, we applied the classical technique of ciphertexts compression to estimate the communication cost.

* In `Margrethe`, no keyswitching nor bootstrapping keys are required.

Thank You !

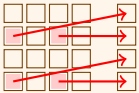
Prime field: $\mathbb{F}_{17}$



(a) SD.

(b) SR.

(c) MC.



(d) ϕ.

Bibliography I

Diego F. Aranha, Antonio Guimarães, Clément Hoffmann, and Pierrick Méaux.
Secure and efficient transciphering for FHE-based MPC.
Cryptology ePrint Archive, Paper 2024/1702, 2024.

Thibault Balenbois, Jean-Baptiste Orfila, and Nigel P. Smart.
Trivial transciphering with trivium and TFHE.
In Michael Brenner, Anamaria Costache, and Kurt Rohloff, editors, *Proceedings of the 11th Workshop on Encrypted Computing & Applied Homomorphic Cryptography, Copenhagen, Denmark, 26 November 2023*, pages 69–78. ACM, 2023.

Bibliography II

- Mingyu Cho, Woohyuk Chung, Jincheol Ha, Jooyoung Lee, Eun-Gyeol Oh, and Mincheol Son.
  FRAST: tfhe-friendly cipher based on random s-boxes.
  IACR Trans. Symmetric Cryptol., 2024(3):1–43, 2024.
- Amit Deo, Marc Joye, Benoit Libert, Benjamin R. Curtis, and Mayeul de Bellabre.
  Homomorphic evaluation of LWR-based PRFs and application to transciphering.
  Cryptology ePrint Archive, Paper 2024/665, 2024.

Bibliography III


Craig Gentry, Shai Halevi, and Nigel P. Smart.
Homomorphic evaluation of the AES circuit.
In Reihaneh Safavi-Naini and Ran Canetti, editors, *Advances in Cryptology - CRYPTO 2012 - 32nd Annual Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2012. Proceedings*, volume 7417 of *Lecture Notes in Computer Science*, pages 850–867. Springer, 2012.