Selected Areas in Cryptography 2025

# Downlink (T)FHE ciphertexts compression

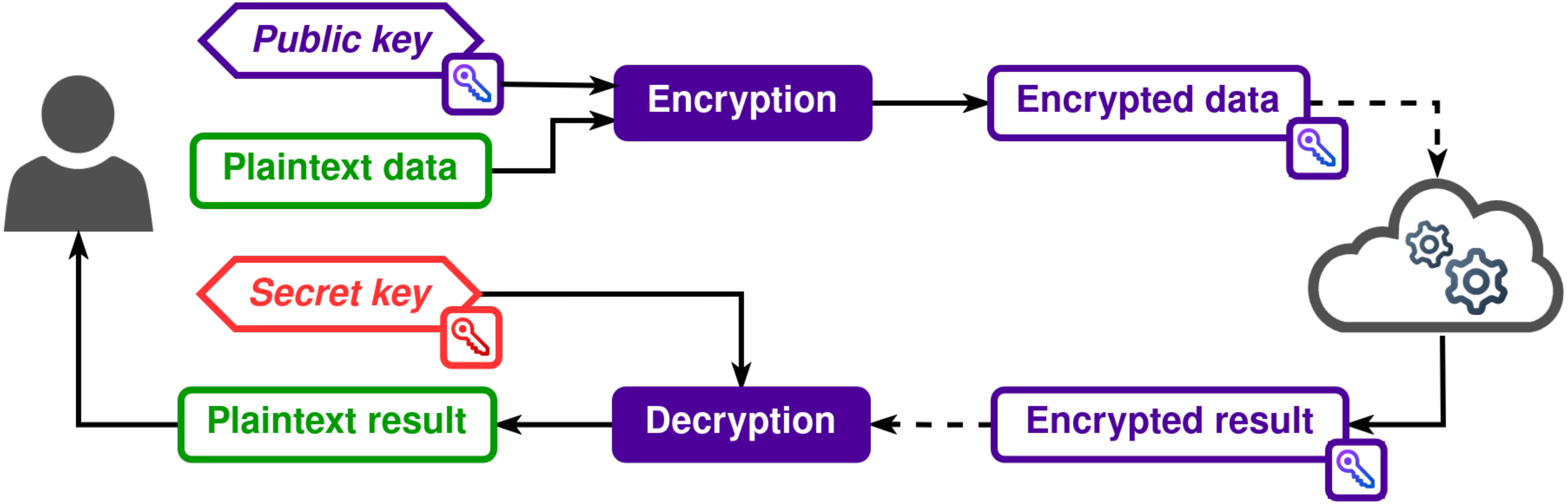
Presents : Antonina BONDARCHUK*

Olive CHAKRABORTY*, Geoffroy COUTEAU**, Renaud SIRDEY*

*name.surname@cea.fr **surname@irif.fr

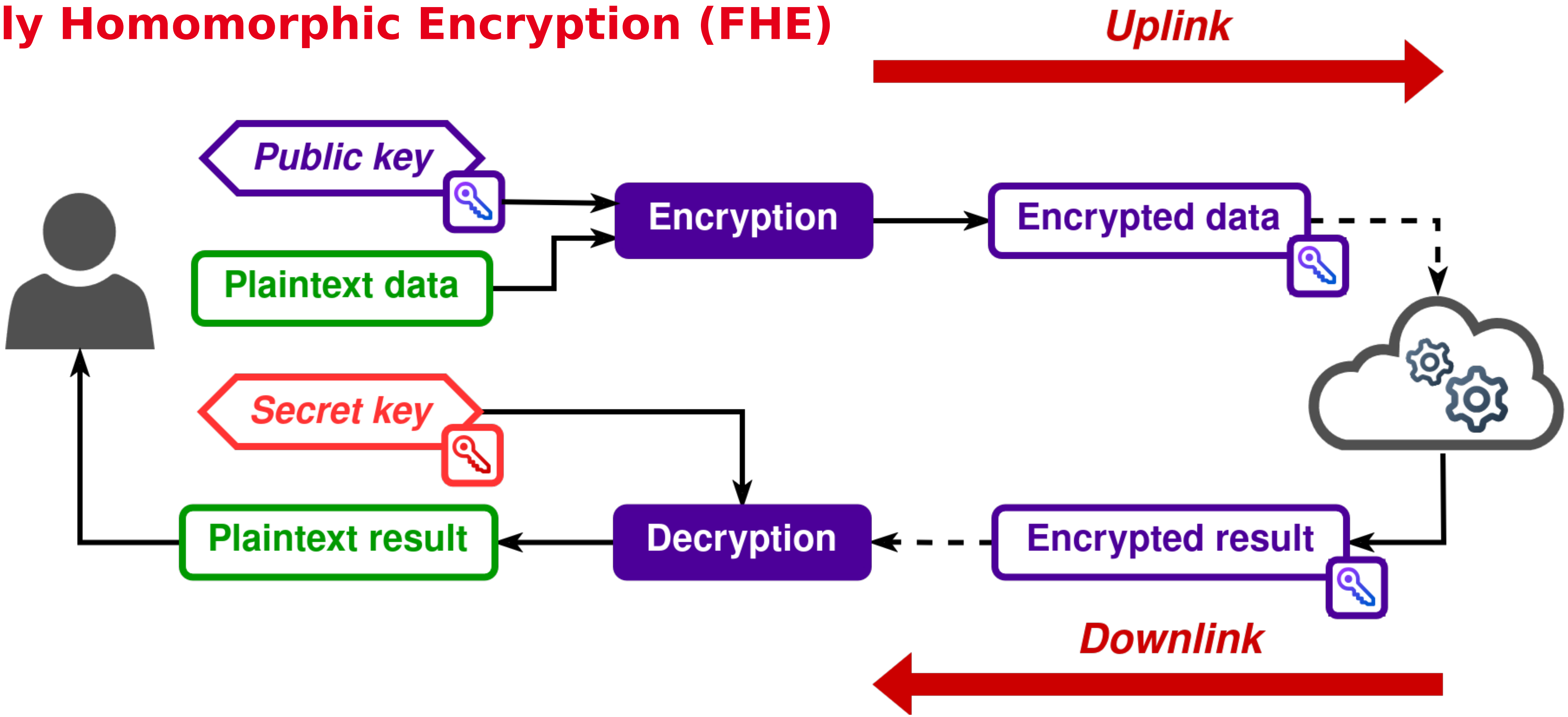
# Context

## Fully Homomorphic Encryption (FHE)




# Context

## Fully Homomorphic Encryption (FHE)




# Problem statement

## TFHE Overview

### LWE

$$\boxed{m} = (\overbrace{\underbrace{a_0, \ldots, a_{n-1}}_{a_i \in \mathbb{Z}_q}, b}^{n+1}), \text{ where } b = \sum a_i s_i + \Delta m + e \in \mathbb{Z}_q$$

$a_i \in \mathbb{Z}_q$
uniform random

$e \in \mathbb{Z}_q$
Gaussian coefficient

$s \in \{0,1\}^n$

### RLWE

$$\Delta m(x) = \Delta m_0 + \Delta m_1 x + \ldots + \Delta m_{N-1} x^{N-1}$$

$$\boxed{m} = (\overbrace{\underbrace{a(x)}_{a(x) \in R_q}, b(x)}^{2N}), \text{ where } b = a(x)s(x) + \Delta m(x) + e(x) \in \mathbb{Z}_q$$

$a(x) \in R_q$
uniform random coefficients

$e(x) \in R_q$
Gaussian coefficients

$s(x) \in R_q$

$m = m_0, \ldots, m_{N-1}$

# Problem statement

## TFHE (over the torus)

$\mathbb{T}$ is the real $[0,1)$ torus, $\mathbb{T}_N[X]$ denotes $\mathbb{R}[X]/(X^N+1)$ mod 1 and $\mathbb{B}_N[X]$ denotes polynomials in $\mathbb{Z}[X]/(X^N+1)$ with binary coefficients

### TLWE

$$m = (\underbrace{a_0, \ldots, a_{n-1}}_{a_i \in \mathbb{T}}, b) \ (n+1), \text{ where } b = \sum a_i s_i + \frac{m}{t} + e \in \mathbb{T}$$

$a_i \in \mathbb{T}$ uniform random

$e \xleftarrow{\mathcal{N}(0,\sigma^2)} \mathbb{T}$ Gaussian coefficient

$m \in \mathbb{Z}_t \quad s \in \{0,1\}^n$

### TRLWE

$$m = m_0 + m_1 x + \ldots + m_{N-1} x^{N-1} \in \mathbb{Z}_t[X]/(X^N+1)$$

$$m = (\underbrace{a}_{a \in \mathbb{T}_N[X]}, b) \ (2N), \text{ where } b = a \cdot s + \frac{m}{t} + e \in \mathbb{T}_N[X]$$

$a \in \mathbb{T}_N[X]$ uniform random coefficients

$e \xleftarrow{\mathcal{N}(0,\sigma^2)} \mathbb{T}_N[X]$ Gaussian coefficients

$s \in \mathbb{B}_N[X]$

$m = m_0, \ldots, m_{N-1}$

# Problem statement

## Expansion factor

**LWE**

$$m = (\overbrace{\underbrace{a_0, \dots, a_{n-1}}_{}, b}^{n+1}), \text{ where } b = \sum a_i s_i + \Delta m + e \in \mathbb{Z}_q$$

$a_i \in \mathbb{Z}_q$
uniform random

$e \in \mathbb{Z}_q$
Gaussian coefficient

$s \in \{0,1\}^n$

**LWE expansion factor:** $\varepsilon = \frac{(n+1)\log_2 q}{\log_2 t}$

**RLWE**

$$\Delta m(x) = \Delta m_0 + \Delta m_1 x + \dots + \Delta m_{N-1} x^{N-1}$$

$$m = (\overbrace{\underbrace{a(x)}_{}, b(x)}^{2N}), \text{ where } b = a(x)s(x) + \Delta m(x) + e(x) \in \mathbb{Z}_q$$

$a(x) \in R_q$
uniform random coefficients

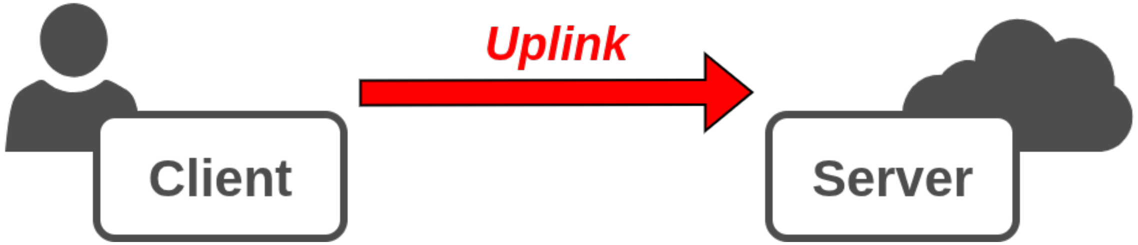
$e(x) \in R_q$
Gaussian coefficients

$s(x) \in R_q$

$m = m_0, \dots, m_{N-1}$

# Problem statement

## How to compress TFHE ciphertexts?




$a_0, \ldots, a_{n-1} \leftarrow \mathrm{PRF}$ $a_0, \ldots, a_{n-1} \leftarrow \mathrm{PRF}$

(~~$a_0, \ldots, a_{n-1}$~~, $b$), sends just $b$

$\xrightarrow{b} (a_0, \ldots, a_{n-1}, b)$

**Plaintext**

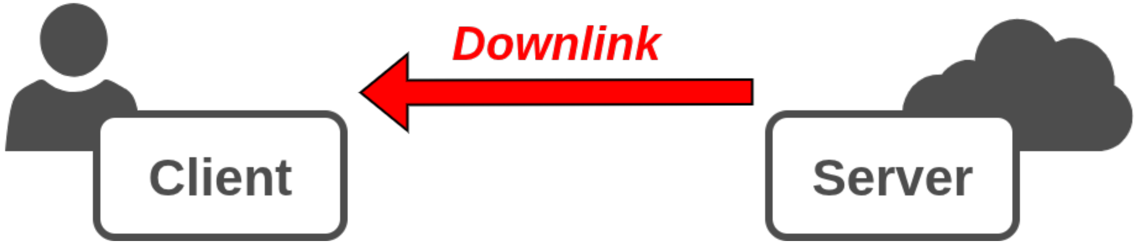
$t = 16$

4 bits

**TLWE**

$q = 2^{32}, n = 750$

32 bits




$\xleftarrow{?} (a_0, \ldots, a_{n-1}, b)$

**Plaintext**

$t = 16$

4 bits

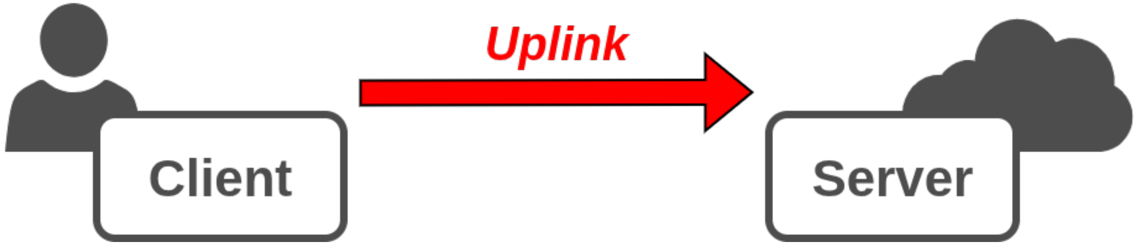
**TLWE**

$q = 2^{32}, n = 750$

24032 bits

# Problem statement

## How to compress TFHE ciphertexts?




$a_0, \dots, a_{n-1} \leftarrow \mathrm{PRF}$ $\quad$ $a_0, \dots, a_{n-1} \leftarrow \mathrm{PRF}$

$(\cancel{a_0, \dots, a_{n-1}}, b)$, sends just $b$

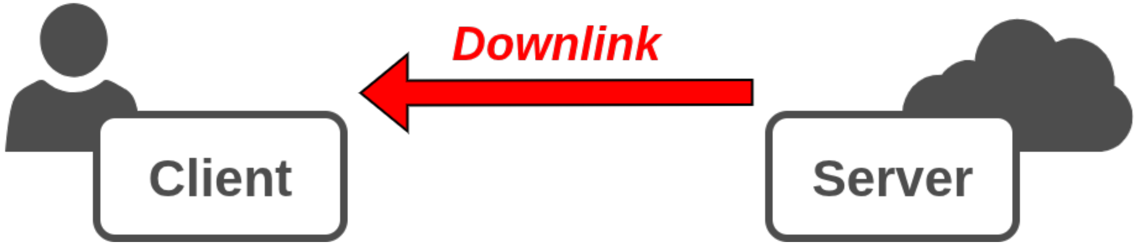
$\xrightarrow{\ b\ } (a_0, \dots, a_{n-1}, b)$

**Plaintext** $\quad$ **TLWE**

$t = 16$ $\quad$ $q = 2^{32}, n = 750$

4 bits $\quad$ 32 bits

$\varepsilon = 8$




$\xleftarrow{\ ?\ } (a_0, \dots, a_{n-1}, b)$

**Plaintext** $\quad$ **TLWE**

$t = 16$ $\quad$ $q = 2^{32}, n = 750$

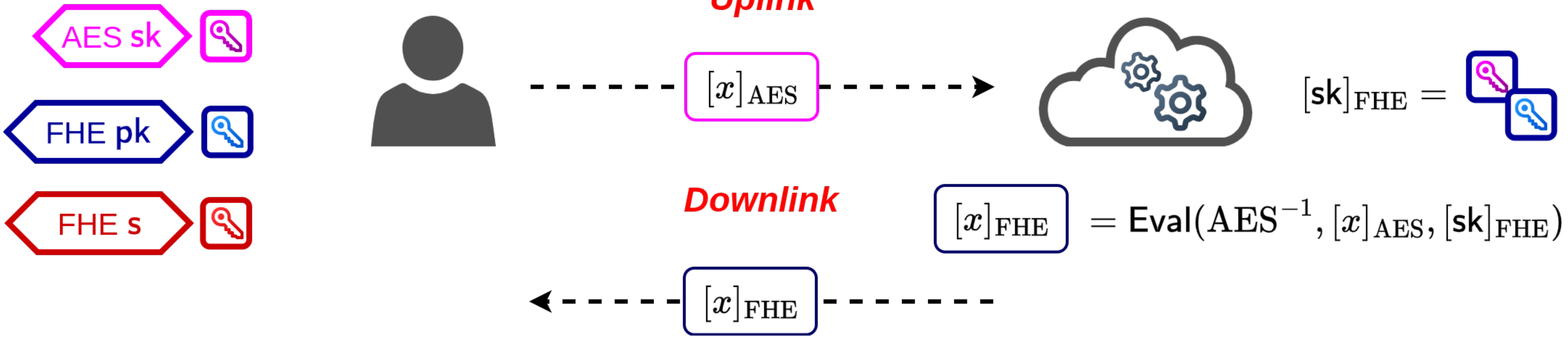
4 bits $\quad$ 24032 bits

$\varepsilon = 6008$

# Problem statement

## Transciphering

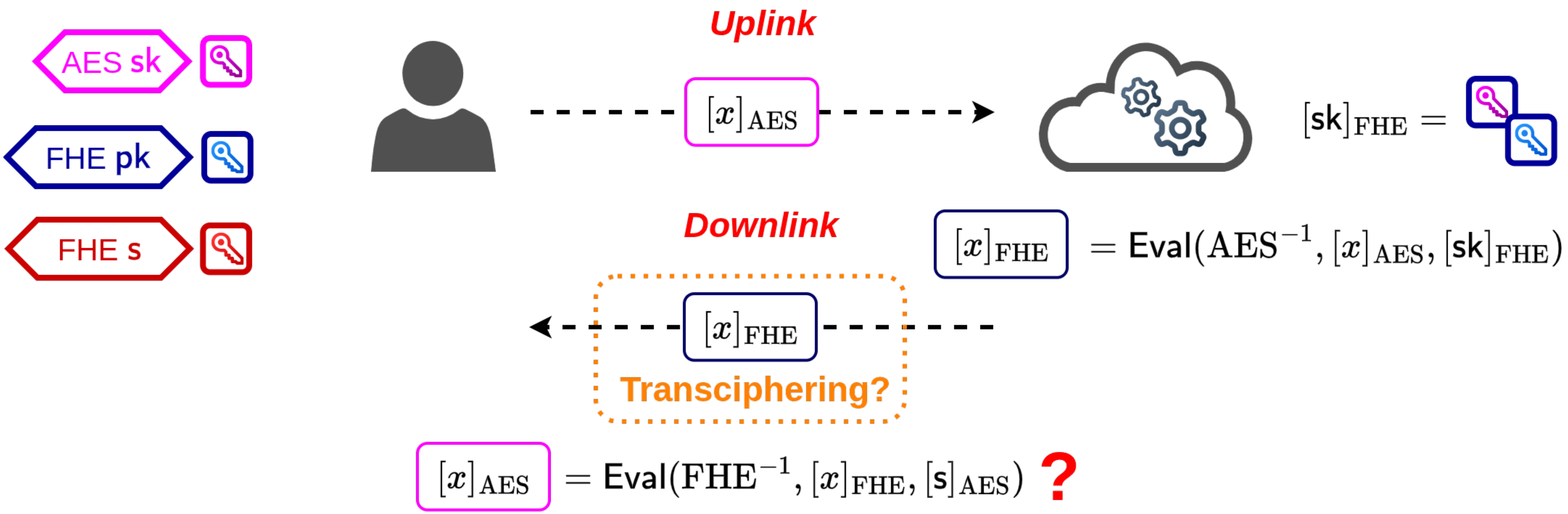
**1-rate expansion!**




**Expansion is large...**

# Problem statement

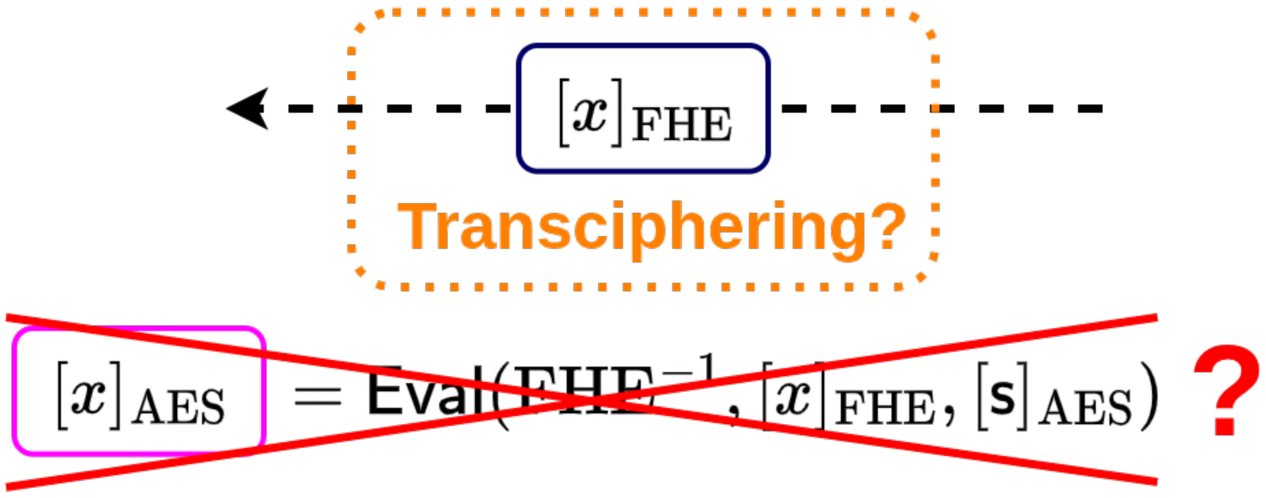
## Transciphering on the downlink?



# Problem statement

## Transciphering on the downlink?

*Downlink*

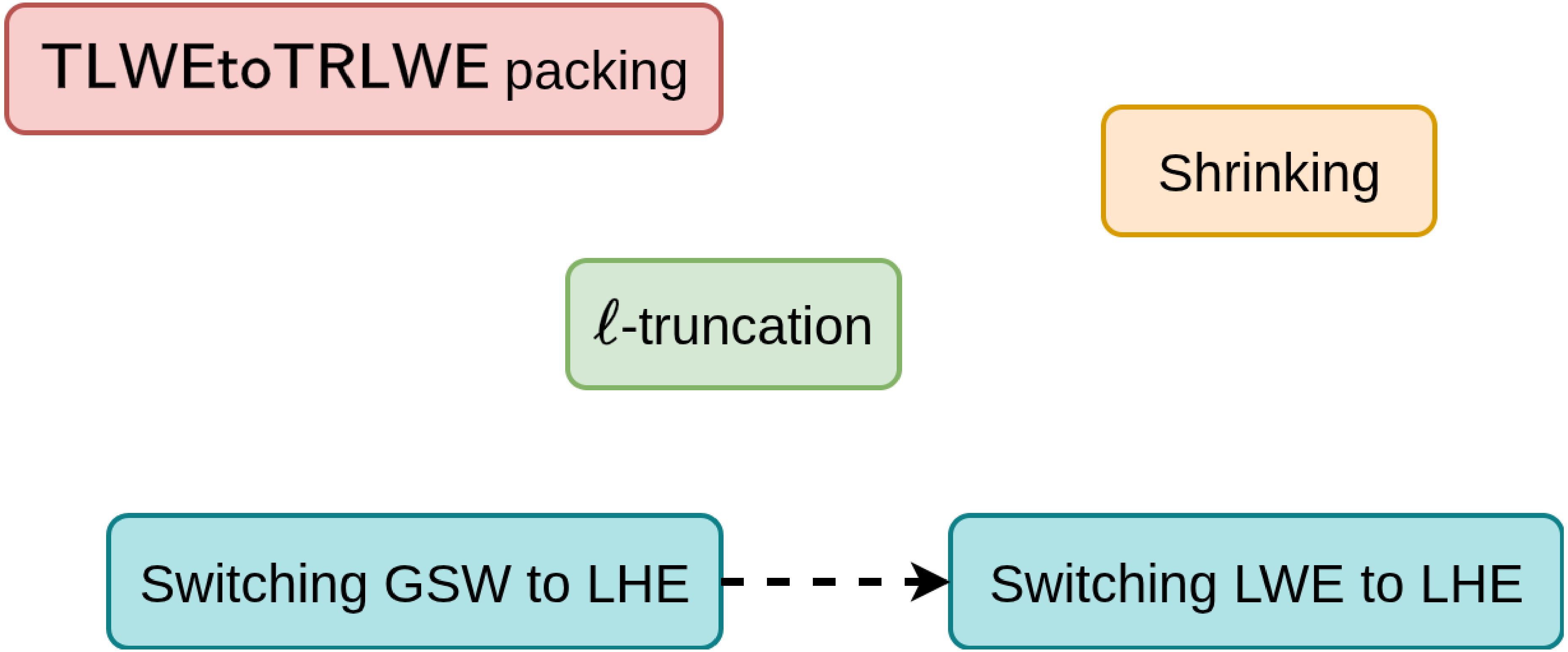



Reminder: a part of decryption function is linear

$$b - \sum a_i s_i = \Delta m + e$$

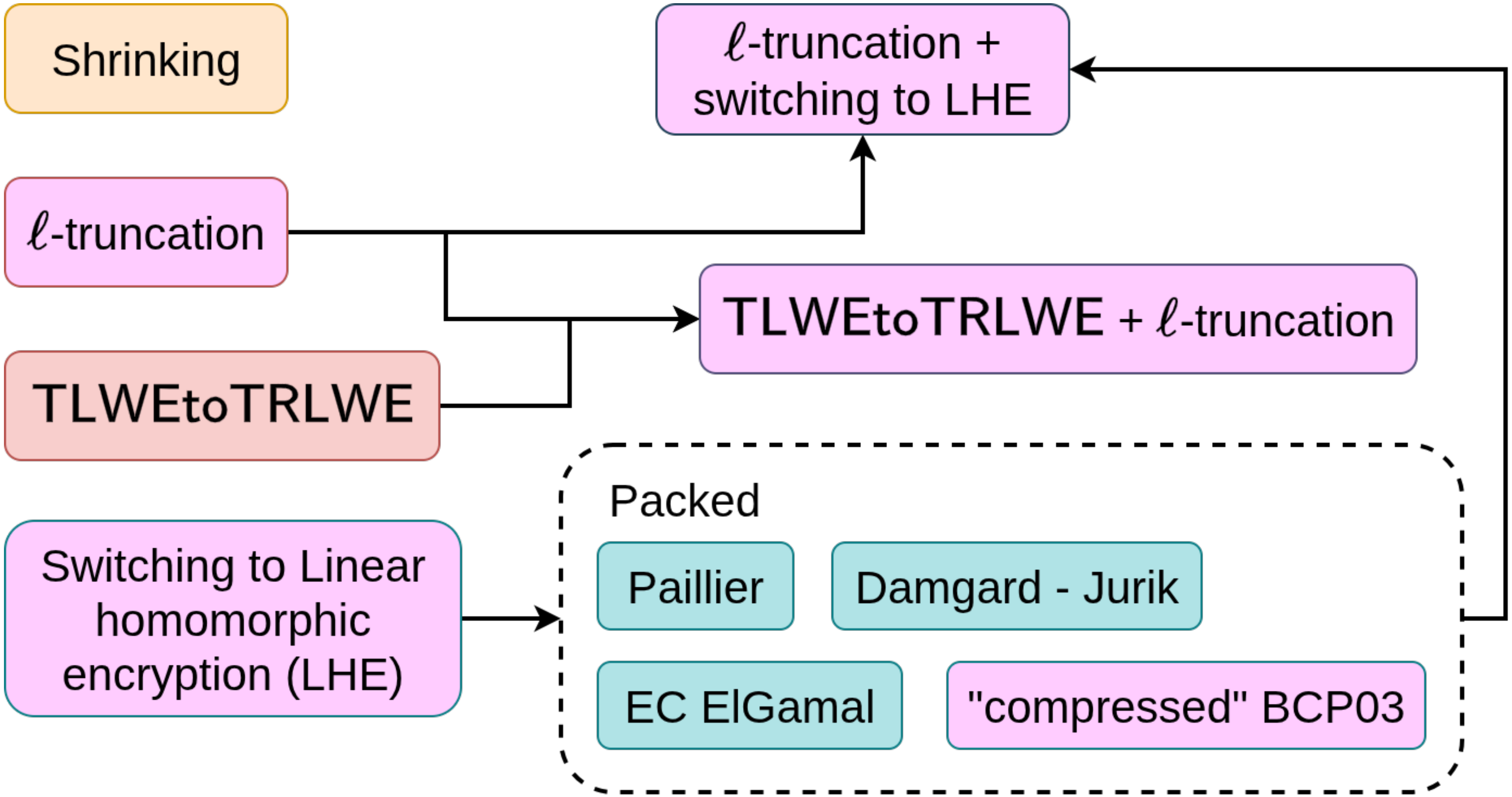
Try to perform transciphering to Linear Homomorphic Encryption (LHE)

# Existing approaches



# Main idea

## Study different (T)LWE compression techniques



# LSB truncation

## Definition

Let $\mathbf{c} = (a_0, \ldots, a_{n-1}, b = a_n)$ denotes a TLWE encryption of $m$. Given $\ell < \lceil \log_2(q) \rceil$, we define the following three operations:

- $\mathsf{Dec}(\mathbf{c}, \mathbf{s})$: return $\lceil (a_n - \langle \mathbf{a}, \mathbf{s} \rangle)/\Delta \rfloor = m$, with $\Delta = \frac{q}{t}$.
- $\mathsf{PartialDec}(\mathbf{c}, \mathbf{s})$: return $a_n - \langle \mathbf{a}, \mathbf{s} \rangle = \Delta m + e$.
- $\mathsf{Trunc}(\mathbf{c}, \ell)$: set $a'_i = \left\lfloor \frac{a_i}{2^\ell} \right\rfloor$ for $i \in \{0, ..., n\}$ and return $\mathbf{c}' = (a'_0, \ldots, a'_n)$.
- $\mathsf{Rescale}(\mathbf{c}', \ell)$: set $a''_i = 2^\ell a'_i$ for $i \in \{0, ..., n\}$ and return $\mathbf{c}'' = (a''_0, \ldots, a''_n)$.

It follows that when $\mathbf{c}$ is a TLWE encryption of $m$ with noise $e$, then $\mathbf{c}''$ is an encryption of $m$ with noise

$$e'' = e - \sum_{i=0}^{n-1} e''_i s_i + e''_n$$

where $e''_i = -(a_i \mod 2^\ell)$.

# LSB truncation

## Relationship between truncation and probability of errorless decryption

$$a''_n - \sum_{i=0}^{n-1} a''_i s_i = b + e''_n - \sum_{i=0}^{n-1} a_i s_i - \sum_{i=0}^{n-1} e''_i \cdot s_i = \Delta m + \underbrace{e - \sum_{i=0}^{n-1} e''_i \cdot s_i + e''_n}_{e''}$$

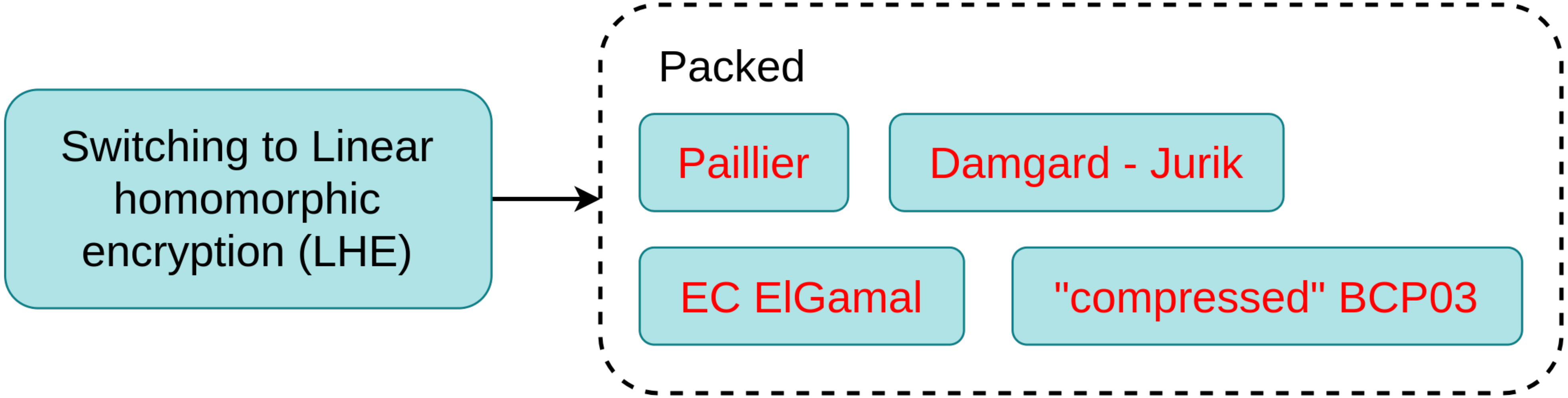
**Proposition 1.** *Let* $\mathbf{c}$ *denote a TLWE encryption of* $m$ *subject to a centered Gaussian noise* $e$ *with variance* $\sigma^2$*, and let* $\mathbf{c}' = \mathsf{Trunc}(\mathbf{c}, \ell_0)$ *with*

$$\ell_0 \leq \left\lfloor \log_2 \left( \frac{1}{n+1} \left( \frac{\Delta}{2} - \sigma\sqrt{2(k+1)\ln 2} \right) + 1 \right) \right\rfloor ,$$

*and* $\Delta = \frac{q}{t}$*. Then,* $\left\lceil \frac{1}{\Delta} \mathsf{PartialDec}(\mathsf{Rescale}(\mathbf{c}', \ell_0), \mathbf{s})) \right\rfloor = m$ *with probability at least* $1 - 2^{-k}$.

**Intuition:** bound the probability that $\mathbf{c}'' = \mathsf{Rescale}(\mathbf{c}', \ell)$ incorrectly decrypts, i.e. $\Pr\left(|e''| \geq \frac{\Delta}{2}\right)$, using a Chernoff bound.

# Scheme switching




**TLWE** $(a_0, \ldots, a_{n-1}, b)$, where $b = \sum a_i s_i + \Delta m + e \in \mathbb{Z}_q$

$\mathsf{PartialDec}(\mathbf{a}, b) : b - \sum a_i s_i = \Delta m + e$

# Linear Homomorphic Encryption

## Why switching is needed?

- Just one LHE ciphertext is transferred rather than $n + 1$ elements in $\mathbb{Z}_q$, achieving compression as soon as the size of an LHE ciphertext is smaller than $(n + 1) \log_2 q$.
- Depending on the LHE, several dot products may be packed in a single LHE ciphertext in order to further enhance compression.

## Summary of main characteristics of the listed LHE schemes

| Cryptosystem | Plaintext domain | Ciphertext domain | Plaintext size (bits) | Ciphertext size (bits) | Expansion factor |
|---|---|---|---|---|---|
| Paillier | $\mathbb{Z}_\mu$ | $\mathbb{Z}_{\mu^2}$ | $\log_2 \mu$ | $2 \log_2 \mu$ | $2$ |
| Dåmgard-Jurik | $\mathbb{Z}_{\mu^y}$ | $\mathbb{Z}_{\mu^{y+1}}$ | $y \log_2 \mu$ | $(y+1) \log_2 \mu$ | $1 + \frac{1}{y}$ |
| EC ElGamal | $\mathbb{F}_\omega$ | $\mathbb{F}_\omega^2$ | $\mathfrak{p}$ | $2 \log_2 \omega$ | $\frac{2 \log_2 \omega}{\mathfrak{p}}$ |
| BCP03 | $\mathbb{Z}_\mu$ | $\mathbb{Z}_{\mu^2}^2$ | $\log_2 \mu$ | $4 \log_2 \mu$ | $4$ |

# Contribution

## Compressed Paillier-ElGamal

### A variant of BCP03 with shorter ciphertexts

KeyGen: $\mu$ be an RSA modulus. For some $\alpha \leftarrow \mathbb{Z}^*_{\mu^2}$ and $d \leftarrow [1, \mathrm{ord}(\mathbb{G})]$, set $g = \alpha^2 \bmod \mu$ and $h = g^{\mu \cdot d} \bmod \mu^2$. Return **pk** $= (\mu, g, h)$ and **sk** $= d$.

Enc: For message $m \in \mathbb{Z}_\mu$, return a ciphertext $\mathbf{c} = (c_0, c_1)$, where $c_0 = g^r \bmod \mu$ and $c_1 = h^r(1+\mu)^m \bmod \mu^2$ for some random pad $r \leftarrow \mathbb{Z}_{\mu^2}$.

Dec: Compute $c = c_1(c_0)^{-\mu \cdot d} \bmod \mu^2$ and return $m = \frac{c-1}{\mu}$.

**Remark:** compared to BCP03, $h$ is computed as a $\mu$-th power and $c_0$ is now given modulo $\mu$, reducing the ciphertext size by $25\%$.

# Contribution

## Compressed Paillier-ElGamal

**Compress:** $c_0 = g^r \bmod \mu,\, c_1 = h^r(1+\mu)^m \bmod \mu^2$

$\mathsf{DDLog}_\mu$: given divisive shares of $(1+\mu)^m \bmod \mu^2$ over $\mathbb{Z}^*_{\mu^2}$ allows to non-interactively derive substractive shares of $m$ over $\mathbb{Z}_\mu$.

Down from $3\log_2 \mu$ to $2\log_2 \mu$

### Compressing ciphertexts via $\mathsf{DDLog}_\mu$

**Idea:** given $c_0$, the holder of $\mathsf{sk} = d$ can locally compute $u = c_0^{\mu \cdot d} = h^r \bmod \mu^2$. Then, $u$ and $c_1$ form divisive shares of $(1+\mu)^m \bmod \mu^2 \Rightarrow$ apply $\mathsf{DDLog}_\mu$ to derive $v', v$ substractive shares of $m$ over $\mathbb{Z}_\mu$: $m = v' - v \bmod \mathbb{Z}_\mu$.

Down from $2\log_2 \mu$ to $\log_2 \mu + \log_2 U$

### Subtractive shares over the integers.

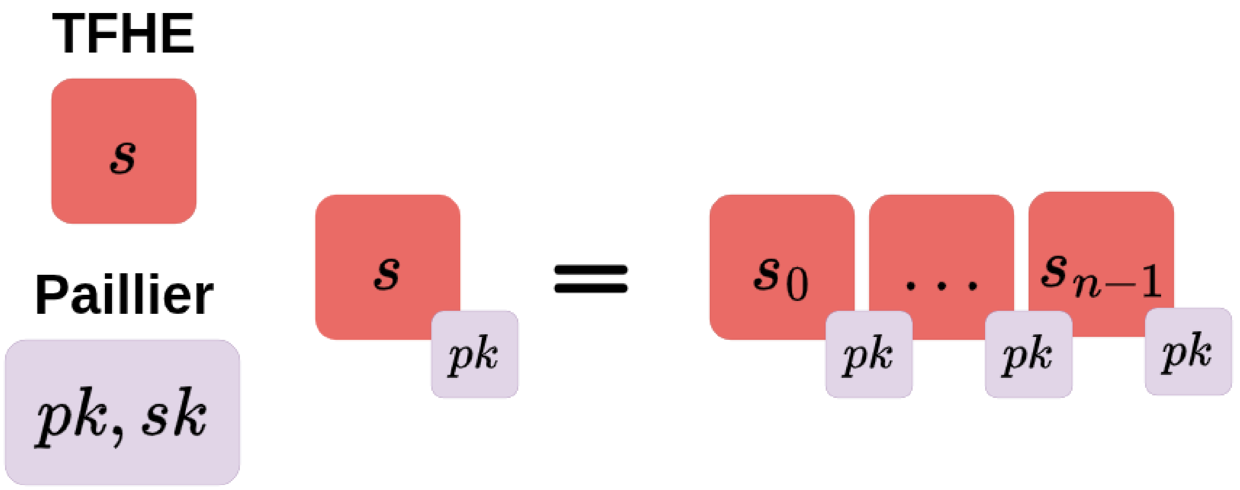
**Idea:** if $m$ is known to be smaller than a bound $U < \mu/2^\lambda$, then $v', v$ form subtractive shares of $m$ over the integers: $m = [v' \bmod U] - v \bmod U$.

**The compression procedure is incompatible with the homomorphic features of the scheme**

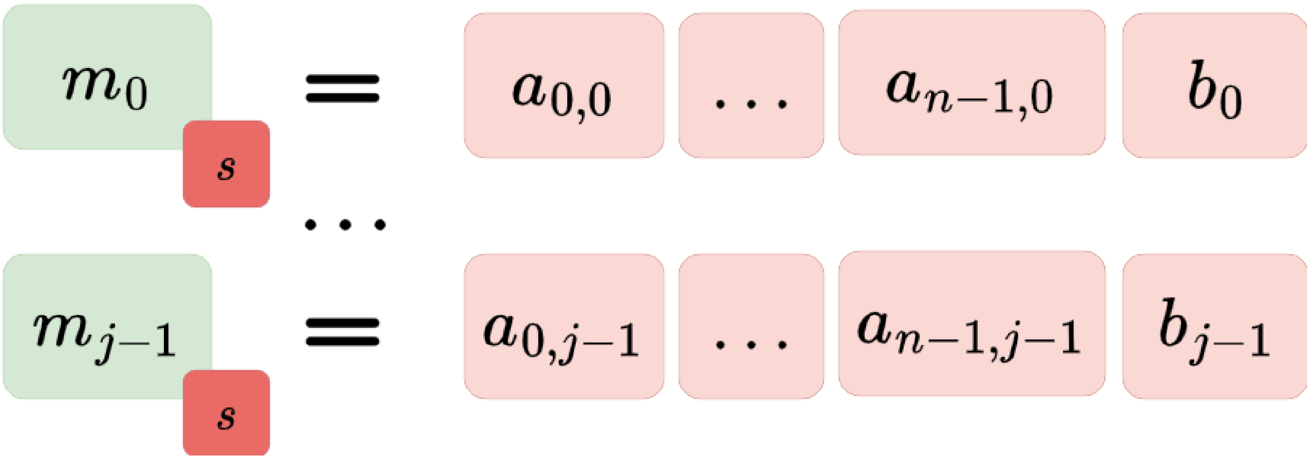
# Switching explained

## “Decrypt-then-pack”

1 Generate parameters:




2 $j$ TLWE ciphertexts:



# Switching explained

## “Decrypt-then-pack”

3 TLWE decryption:

Ciphertext multiplication by constant; Ciphertext-ciphertext addition

$$s_0^{pk} \bullet a_{0,0} + s_1^{pk} \bullet a_{1,0} + \ldots + s_{n-1}^{pk} \bullet a_{n-1,0}$$

$$+\ b_0 = m_0^{pk}$$

Add constant to ciphertext

*Parallelize*

$$\cdots = m_{j-1}^{pk}$$

$q$ is a $32$ bit TFHE ciphertext modulus
$\mu$ is a $2048$ bit Paillier plaintext modulus
$\mu^2$ is a $4096$ bit Paillier ciphertext modulus

4 Pack:

$\log_2 \mu$ bits

$$\left[ m_0 \mid m_1 \mid \ldots \mid m_{j-1} \right]^{pk}$$

$y$ bits

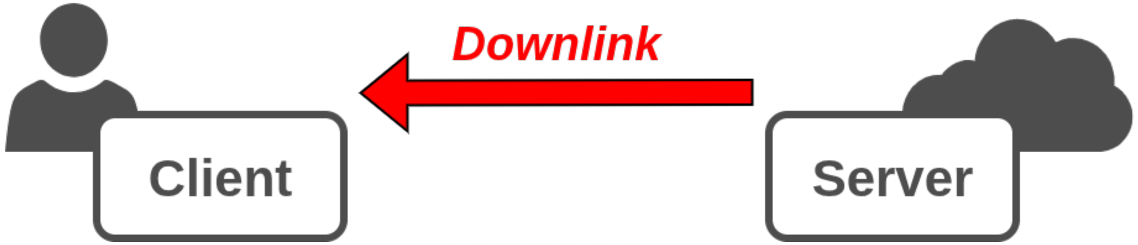
$y = \lceil \log_2(n+1) + \log_2 q \rceil$ bits: slot size
$j = \left\lfloor \frac{\lfloor \log_2 \mu \rfloor}{y} \right\rfloor$: pack $j$ TLWEs

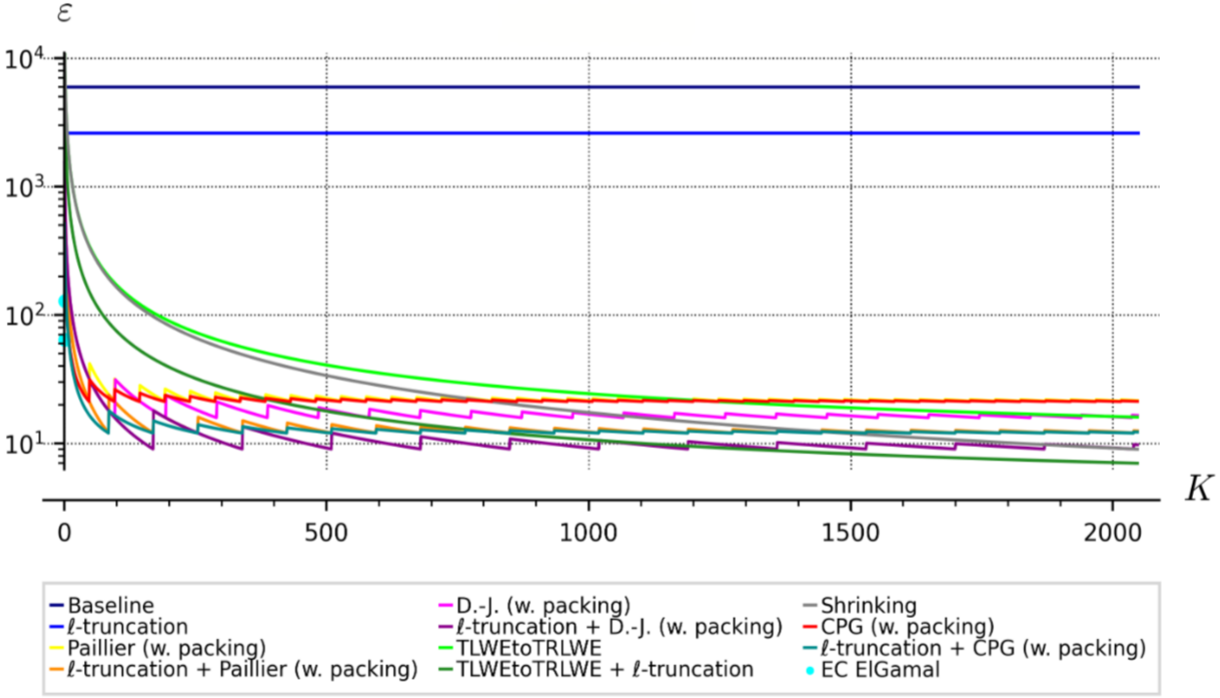
5 Decrypt and unpack:

$$m_0 \quad m_1 \quad \ldots \quad m_{j-1}$$

# Experimental study

## Which compression technique to choose?



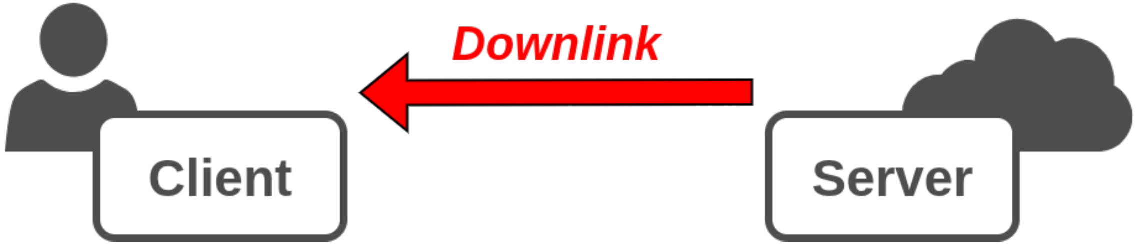

**How many TLWEs do we want to transmit?**

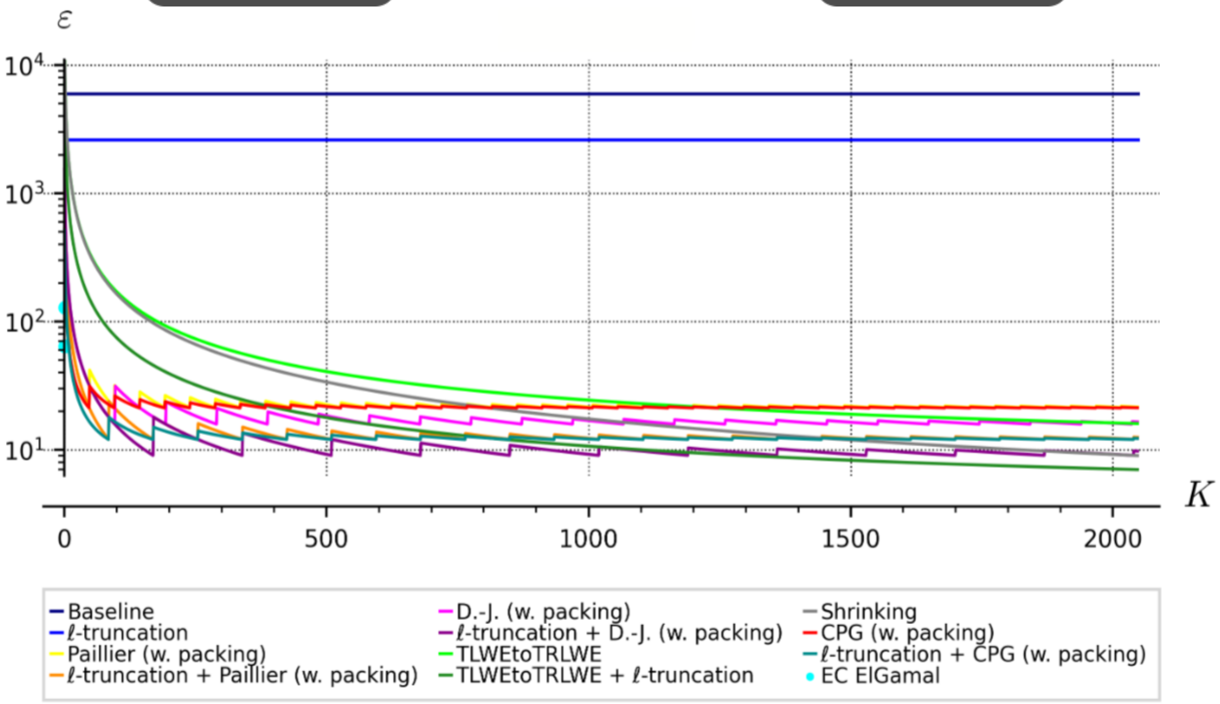
$t = 16$:

| $K$ | 1 | 50 | 150 | 250 | 500 | $\infty$ |
|---|---|---|---|---|---|---|
| TLWE | 6008 | 6008 | 6008 | 6008 | 6008 | 6008 |
| TLWE $\ell$-truncation | 2628.5 | 2628.5 | 2628.5 | 2628.5 | 2628.5 | 2628.5 |
| Shrinking | 16393 | 328.8 | 110.2 | 66.5 | 33.7 | 9 |
| TLWEtoTRLWE | 16392 | 335.6 | 117.2 | 73.5 | 40.7 | 16 |
| TLWEtoTRLWE + $\ell$-truncation | 7171.5 | 146.8 | 51.2 | 32.1 | 17.8 | **7** |
| Paillier (w. packing) | 1024 | 40.9 | 27.3 | 24.5 | 22.5 | 21.3 |
| $\ell$-truncation + Paillier (w. packing) | 1024 | 20.4 | 13.6 | **12.2** | 12.2 | 12 |
| Dåmgard-Jurik (w. packing) | 1536 | 30.7 | 20.4 | 18.4 | 18.4 | 15.8 |
| $\ell$-truncation + D.-J. (w. packing) | 1536 | 30.7 | **10.2** | **12.2** | **9.2** | 9 |
| CPG (w. packing) | 522.5 | 30.9 | 24.1 | 22.7 | 21.7 | 21.1 |
| $\ell$-truncation + CPG (w. packing) | 518.0 | **16.2** | 12.8 | 12.1 | 12.1 | 12 |
| EC ElGamal | **128** | – | – | – | – | – |

# Experimental study

## Which compression technique to choose?




**How many TLWEs do we want to transmit?**

$t = 16$:

| $K$ | 1 | 50 | 150 | 250 | 500 | $\infty$ |
|---|---|---|---|---|---|---|
| TLWE | 6008 | 6008 | 6008 | 6008 | 6008 | 6008 |
| TLWE $\ell$-truncation | 2628.5 | 2628.5 | 2628.5 | 2628.5 | 2628.5 | 2628.5 |
| Shrinking | 16393 | 328.8 | 110.2 | 66.5 | 33.7 | 9 |
| TLWEtoTRLWE | 16392 | 335.6 | 117.2 | 73.5 | 40.7 | 16 |
| TLWEtoTRLWE + $\ell$-truncation | 7171.5 | 146.8 | 51.2 | 32.1 | 17.8 | **7** |
| Paillier (w. packing) | 1024 | 40.9 | 27.3 | 24.5 | 22.5 | 21.3 |
| $\ell$-truncation + Paillier (w. packing) | 1024 | 20.4 | 13.6 | **12.2** | 12.2 | 12 |
| Dåmgard-Jurik (w. packing) | 1536 | 30.7 | 20.4 | 18.4 | 18.4 | 15.8 |
| $\ell$-truncation + D.-J. (w. packing) | 1536 | 30.7 | **10.2** | **12.2** | **9.2** | 9 |
| CPG (w. packing) | 522.5 | 30.9 | 24.1 | 22.7 | 21.7 | 21.1 |
| $\ell$-truncation + CPG (w. packing) | 518.0 | **16.2** | 12.8 | 12.1 | 12.1 | 12 |
| EC ElGamal | **128** | – | – | – | – | – |

**Remind: the uplink PRF synchronisation $\varepsilon = 8$**
**For the downlink we decrease $\varepsilon$ from $6008$ to a value between $16$ and $7$**

# Conclusion

**Compression techniques for TFHE ciphertexts** → **Significantly reduce the expansion factor**

## Most appropriate compression methods in function of *K*

| $K$ | Most compressive method |
|---|---|
| $1 \leq K \leq 2$ | Switch. to EC ElGamal |
| $2 < K \leq 81$ | $\ell$-truncation + switch. to CPG (w.pack.) |
| $81 < K \leq 163$ | $\ell$-truncation + switch. to D.-J. (w.pack.) |
| $163 < K \leq 243$ | $\ell$-truncation + switch. to CPG (w.pack.) |
| $243 < K \leq 1141$ | $\ell$-truncation + switch. to D.-J. (w.pack.) |
| $1141 < K \leq 1228$ | TLWEtoTRLWE + $\ell$-truncation |
| $1228 < K \leq 1304$ | $\ell$-truncation + switch. to D.-J. (w.pack.) |
| $K > 1304$ | TLWEtoTRLWE + $\ell$-truncation |

# Key takeaways

- First complete study on TFHE downlink ciphertext compression.

- Provide concrete guidelines on how to choose the best compression technique depending on a ciphertext number to transmit.

- Demonstrate that downlink expansion factors **below 10** are practically achievable and comparable with the expansion factor for the simple uplink ciphertext compression technique (have the same order of magnitude).

- Propose a new CPG LHE. Switching to CPG makes a transition from the FHE to the not-at-all HE scheme and is the most communication-efficient option for transmitting up to around 100 evaluated TFHE ciphertexts.

- The techniques developed in this paper are beneficial only to LWE-based schemes, as the LHEs have a plaintext domain that is too small to absorb the large N typically used for RLWE schemes.

- The LSB truncation technique is not universally applicable, as it significantly increases the ciphertext noise. It can be applied only to schemes with an efficient bootstrapping procedure (like TFHE).

**Bondarchuk A., Chakraborty O., Couteau G., Sirdey R.: Downlink (T)FHE ciphertexts compression (https://eprint.iacr.org/2024/1921)**

# Thank you for your attention!

**If you liked the presentation and want to know more, contact me!**

**antonina.bondarchuk@cea.fr**

# References


[1] Brakerski, Z., Döttling, N., Garg, S., Malavolta, G.: Leveraging linear decryption: Rate-1 fully-homomorphic encryption and time-lock puzzles. In: TCC. pp. 407–437 (2019)

[2] Bresson, E., Catalano, D., Pointcheval, D.: A simple public-key cryptosystem with a double trapdoor decryption mechanism and its applications. In: ASIACRYPT. pp. 37–54 (2003)

[3] Checri, M., Sirdey, R., Boudguiga, A., Bultel, J.P.: On the practical cpad security of "exact" and threshold FHE schemes. In: CRYPTO (2024)

[4] Cheon, J.H., Choe, H., Passelègue, A., Stehlé, D., Suvanto, E.: Attacks against the IND-CPAD security of exact FHE schemes. In: Tech. Rep. 127, IACR ePrint (2024)

[5] Chillotti, I., Gama, N., Georgieva, M., Izabachène, M.: Faster Fully Homomorphic Encryption: Bootstrapping in Less Than 0.1 Seconds. In: Advances in Cryptology - ASIACRYPT 2016. Ed. by Jung Hee Cheon and Tsuyoshi Takagi. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 3–33 (2016), isbn: 978-3-662-53887-6

[6] Damgård, I., Jurik, M., Nielsen, J.B.: A generalization of Paillier's public-key system with applications to electronic voting. In: Int. J. Inf. Secur. 9, 371–385 (2010)

[7] Lyubashevsky, V., Peikert, C., Regev, O.: On ideal lattices and learning with errors over rings. In: EUROCRYPT. pp. 1–23 (2010)

[8] Menezes, A.: Elliptic Curve Public Key Cryptosystems. In: Springer Science and Business Media, vol. 234 (1993)

[9] Orlandi, C., Scholl, P., Yakoubov, S.: The rise of paillier: Homomorphic secret sharing and public-key silent OT. In: EUROCRYPT. pp. 678–708 (2021)

[10] Paillier, P.: Public-Key Cryptosystems Based on Composite Degree Residuosity Classes. In: EUROCRYPT. pp. 223–238 (1999)

[11] Python Paillier lib.: `https://python-paillier.readthedocs.io/en/develop/`

[12] Regev, O.: On lattices, learning with errors, random linear codes, and cryptography. In: Proceedings of the thirty-seventh annual ACM symposium on Theory of computing (Baltimore, MD, USA: ACM. pp. 84–93 (2005)

[13] Sagemath lib.: `https://www.sagemath.org/`

[14] TFHE lib.: `https://tfhe.github.io/tfhe/`

# Existing approaches

- **Chillotti, I., Gama, N., Georgieva, M., Izabachène, M.: Faster fully homomorphic encryption: Bootstrapping in less than 0.1 seconds (2016)**

**Assembling** **TLWEs to TRLWE**

Up to $N$ TLWE ciphertexts can be assembled into $1$ TRLWE ciphertext, whereby $N$ TLWE messages $m_0, \ldots, m_{N-1}$ maps to $m(x) = \sum_{i=0}^{N-1} m_i x^i$

---

- **Brakerski, Z., Döttling, N., Garg, S., Malavolta, G.: Leveraging linear decryption: Rate-1 fully homomorphic encryption and time-lock puzzles (2019)**

**Switch GSW to LHE**

**Shrinking** **TRLWE**

Compute a helper $r \in \mathbb{Z}_q$ and a value $w \in \mathbb{Z}_t$. The decryption of the original TRLWE can be corevered exactly from $r, w$ and a secret key $s$

# Existing approaches

- **Chen, H., Chillotti, I., Ren, L.: Onion ring ORAM: Efficient constant bandwidth oblivious RAM from (leveled) TFHE (2019)**

LSB truncation **TLWE and TRLWE**

Remove $\ell$ less significant bits in $a$'s and $b$'s coefficients of TLWE or TRLWE sample by dividing the coefficients by $2^\ell$

- **Mahdavi R. A., Diaa A., Kerschbaum F.: HE is all you need: Smaller FHE Responses via Additive HE (2024)**

Switch LWE to LHE [1]: switch LWEs to Paillier, Damgard-Jurik

LSB truncation: modswitch LWEs to the lowest modulus in the BGV parameter set

# Positioning

## ‘1-rate FHE’ and ‘HE is all you need’

- Revisit ideas from both ‘1-rate FHE’ and ‘HE is all you need’, but adapt them to the specificities of the TFHE scheme.
- Focuse mainly on the non-asymptotic regime.
- Provide a rigorous analysis of the induced decryption error probability, eventually leading to better compression ratios (4 to 5 times better than in ‘HE is all you need’).
- Consider a more exhaustive set of LHE depending on the number $K$ of TFHE ciphertexts to transmit (including a new variant of the BPC03 scheme that allows us to achieve best-in-class compression in the regime where $K$ is a few tens).

# Compressed Paillier-ElGamal

## Distributed discrete logarithm

The scheme above enjoys shorter ciphertexts than BCP, but still larger than Paillier ($3\log\mu$ versus $2\log\mu$). At a high level, this procedure allows two parties, given divisive shares of $(1+\mu)^m \bmod \mu^2$ over $\mathbb{Z}^*_{\mu^2}$, to non-interactively derive *substractive shares* of $m$ over $\mathbb{Z}_\mu$.

$\mathsf{DDLog}_\mu$:

**Input.** An element $u \in \mathbb{Z}^*_{\mu^2}$.
**Output.** A value $v \in \mathbb{Z}_\mu$.
**Procedure.** Write $u = u_0 + \mu \cdot u_1$, where $u_0, u_1 \in \mathbb{Z}_\mu$ denote the base-$\mu$ decomposition of $u$. Return $v = u_1/u_0 \bmod \mu$.

We now explain why this procedure has the intented behavior. Let $u, u'$ denote two divisive shares over $\mathbb{Z}^*_{\mu^2}$ of $(1+\mu)^m \bmod \mu^2$; that is, $u'/u = (1+\mu)^m = 1 + \mu m \bmod \mu^2$. Writing $u = u_0 + \mu \cdot u_1$ and $u' = u'_0 + \mu \cdot u'_1$, we obtain

$$u'_0 + \mu \cdot u'_1 = (u_0 + \mu \cdot u_1) \cdot (1 + \mu \cdot m) \bmod \mu^2.$$

The above equation yields $u_0 = u'_0 \bmod \mu$ and $u'_1 = u_1 + u_0 m \bmod \mu$. Therefore, $m = u'_1/u'_0 - u_1/u_0 \bmod \mu$: $u'_1/u'_0$ and $u_1/u_0$ form substractive shares of $m$ over $\mathbb{Z}_\mu$, as intended.

# Compressed Paillier-ElGamal

**Compressing ciphertexts via $\mathsf{DDLog}_\mu$.** The distributed discrete logarithm procedure implies a simple and efficient compression mechanisms for Paillier-ElGamal. The key observation is that given $c_0 = g^r \bmod \mu$, the holder of the secret key $d$ can locally compute $u = c_0^{\mu \cdot d} = h^r \bmod \mu^2$. Then, $u$ and $c_1$ form divisive shares of $c_1/u = (1 + \mu m) \bmod \mu^2$. This immediatly yields the following compression mechanism:

- $\mathsf{Compress}(c_0, c_1)$: run $v' \leftarrow \mathsf{DDLog}_\mu(c_1)$. Output $(c_0, v')$.
- $\mathsf{Dec}'(c_0, v')$: compute $u \leftarrow c_0^{\mu \cdot d} \bmod \mu^2$ and $v \leftarrow \mathsf{DDLog}_\mu(u)$. Output $m = v' - v \bmod \mu$.

The resulting compressed ciphertext size is $2 \log \mu$, down from $3 \log \mu$, matching the size of a standard Paillier ciphertext.

# Compressed Paillier-ElGamal

However, if $m$ is known to be smaller than a bound $B < \mu/2^\lambda$ (where $\lambda$ denotes a security parameter), we can do better. The main observation is that with overwhelming probability, $v', v$ form substractive shares of $m$ *over the integers*. This observation allows to further reduce the compressed ciphertext size by reducing $v'$ modulo $B$:

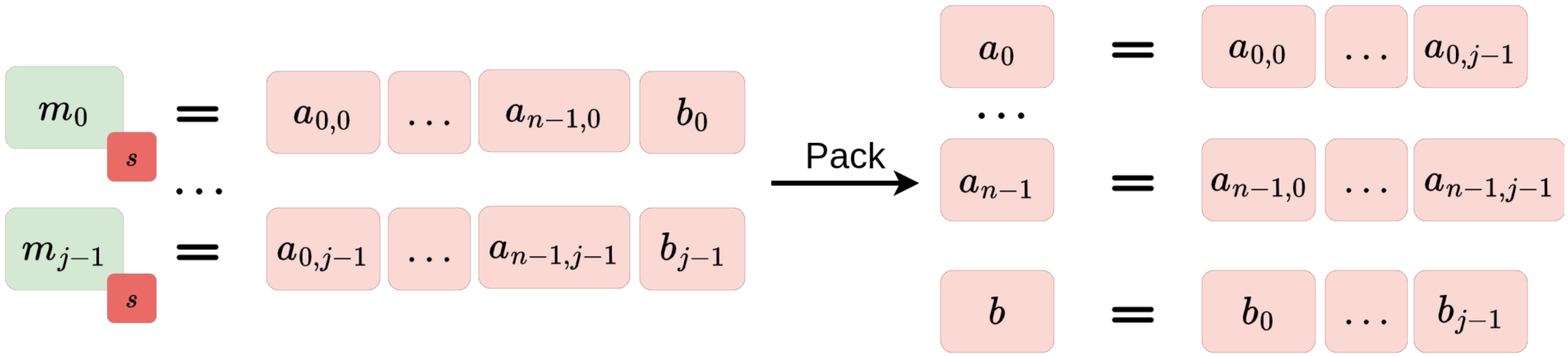
– $\mathsf{Compress}(c_0, c_1)$: run $v' \leftarrow \mathsf{DDLog}_\mu(c_1)$ and set $v'' \leftarrow [v' \bmod B]$. Output $(c_0, v'')$.
– $\mathsf{Dec}'(c_0, v')$: compute $u \leftarrow c_0^{\mu \cdot d} \bmod \mu^2$ and $v \leftarrow \mathsf{DDLog}_\mu(u)$. Output $m = v'' - v \bmod B$.

With this last optimization, the ciphertext size went down to $\log \mu + \log B$ bits. When $B$ is small (e.g. $B \approx 2^{40}$ as in our application), this yields an almost twofold size improvement over a standard Paillier encryption.

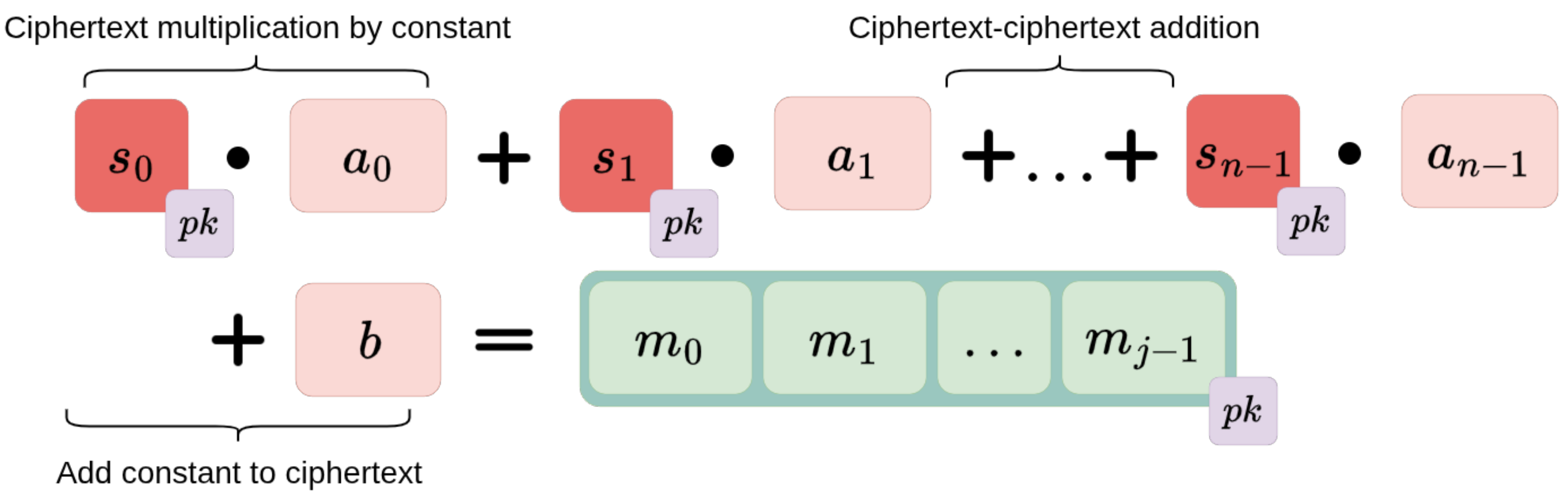
# Switching explained

## “Pack-then-decrypt”

2




3 TLWE decryption:



# Conclusion

Compression techniques for TFHE ciphertexts → Significantly reduce the expansion factor

## Most appropriate compression methods in function of *K*

| $K$ | Most compressive method | Timing | | |
|---|---|---|---|---|
| | | (1) | (2) | (3) |
| $1 \leq K \leq 2$ | Switch. to EC ElGamal | 0.02 | 0.01 | 0.001 |
| $2 < K \leq 81$ | $\ell$-truncation + switch. to CPG (w.pack.) | 6.93 | 5.66 | 0.86 |
| $81 < K \leq 163$ | $\ell$-truncation + switch. to D.-J. (w.pack.) | 13.87 | 11.32 | 1.73 |
| $163 < K \leq 243$ | $\ell$-truncation + switch. to CPG (w.pack.) | 20.79 | 16.89 | 2.58 |
| $243 < K \leq 1141$ | $\ell$-truncation + switch. to D.-J. (w.pack.) | 97.09 | 79.24 | 12.11 |
| $1141 < K \leq 1228$ | TLWEtoTRLWE + $\ell$-truncation | 0.4 | | |
| $1228 < K \leq 1304$ | $\ell$-truncation + switch. to D.-J. (w.pack.) | 110.96 | 90.56 | 13.84 |
| $K > 1304$ | TLWEtoTRLWE + $\ell$-truncation | 0.4 | | |

The timings are given in seconds for the maximum value of $K$ on the intervals:

(1): "Pack-then-decrypt" switching

(2): "Decrypt-then-pack" switching

(3): Parallelized "decrypt-then-pack" switching