A (gentle & short) overview of FHE and some of its applications

Renaud Sirdey
Research Director
CEA LIST
Université Paris-Saclay
(renaud.sirdey@cea.fr)

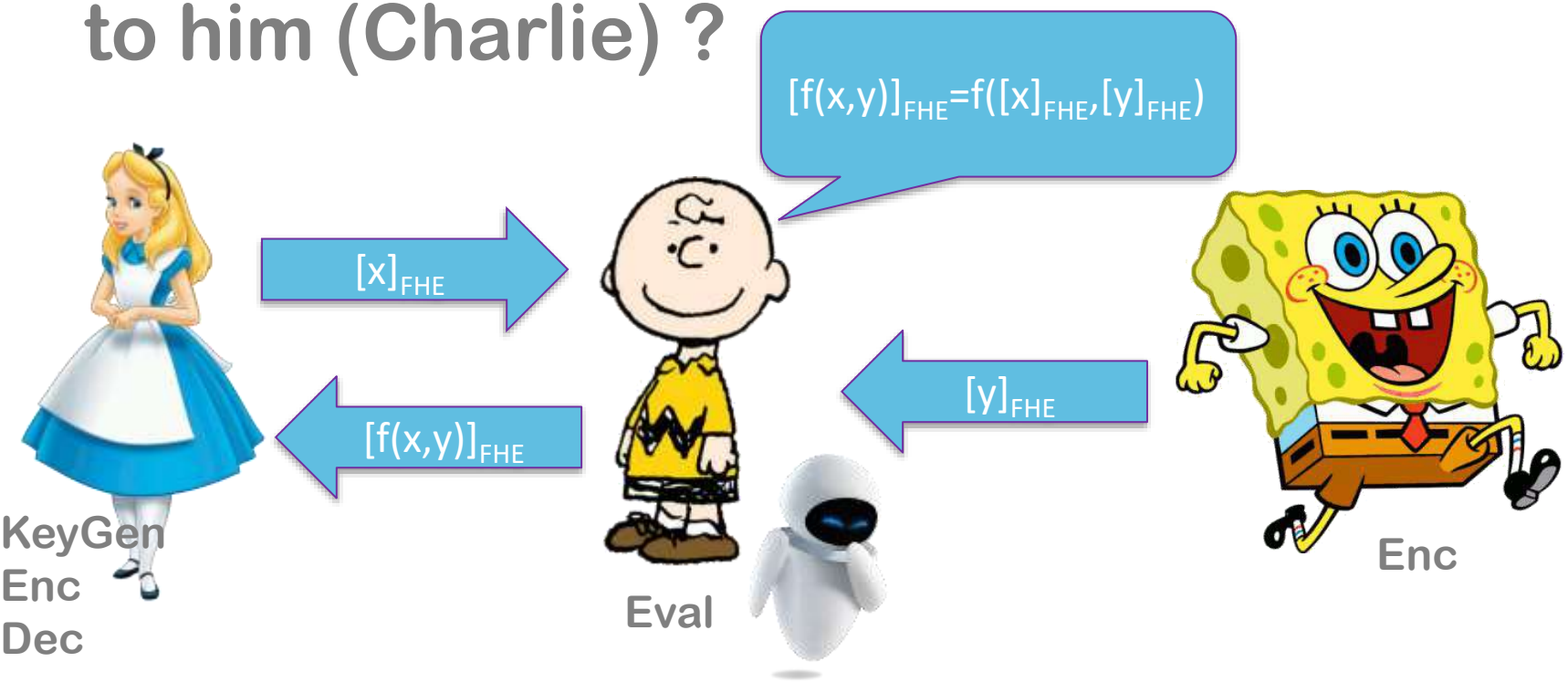
Séminaire ARISTOTE
Ecole Polytechnique, May 2024

Fully Homomorphic Encryption

- KeyGen: given λ generates ek and sk.
- Enc: given plaintext m (and ek) generates ciphertext c.
- Dec: given ciphertext c (and sk) generates plaintext m.
- Eval: given f and $c_1,\ldots,c_k$ generates a new ciphertext c_e.
  - Often materialized by homomorphic addition and multiplication operators.
- Main properties:
  - Dec(Enc(m))=m.
  - Dec(Eval(f;Enc(m_1),…,Enc(m_k))=f(m_1,…,m_k).

The FHE dream

- Can Charlie do something useful for Alice using both Alice and Bob data but without revealing them (the data) to him (Charlie) ?



- KeyGen
- Enc
- Dec

« FHEllo world! » (old school)

- Consider the Paillier cryptosystem where $c=Enc(m)=g^m r^n \bmod n^2$.
- Then $cc' \bmod n^2 = g^{m+m'}(rr')^n \bmod n^2 = Enc(m+m')$.
  - So the multiplication operator in the encrypted domain is an addition operator w. r. t. the clear domain.

- We even know how to extend it to support (1 level of) multiplications.

Learning With Errors (LWE)

$35s_0+69s_1+7s_2+81s_3+27s_4+36s_5+90s_6+113s_7+91s_8+25s_9=5 \bmod 127$

$97s_0+114s_1+121s_2+125s_3+17s_4+48s_5+27s_6+74s_7+90s_8+123s_9=2 \bmod 127$

$55s_0+61s_1+52s_2+17s_3+32s_4+114s_5+7s_6+14s_7+114s_8+113s_9=88 \bmod 127$

$31s_0+3s_1+63s_2+37s_3+56s_4+39s_5+60s_6+16s_7+32s_8+38s_9=98 \bmod 127$

$16s_0+64s_1+21s_2+88s_3+18s_4+90s_5+18s_6+40s_7+24s_8+17s_9=42 \bmod 127$

$40s_0+57s_1+1s_2+92s_3+53s_4+59s_5+20s_6+53s_7+66s_8+53s_9=59 \bmod 127$

$57s_0+101s_1+114s_2+29s_3+4s_4+76s_5+21s_6+7s_7+49s_8+20s_9=63 \bmod 127$

$106s_0+61s_1+44s_2+55s_3+108s_4+111s_5+82s_6+112s_7+41s_8+126s_9=3 \bmod 127$

$98s_0+8s_1+48s_2+43s_3+118s_4+105s_5+118s_6+57s_7+97s_8+112s_9=85 \bmod 127$

$66s_0+40s_1+94s_2+52s_3+80s_4+76s_5+18s_6+104s_7+97s_8+19s_9=97 \bmod 127$

$36s_0+93s_1+98s_2+70s_3+126s_4+69s_5+88s_6+102s_7+99s_8+88s_9=9 \bmod 127$

Learning With Errors (LWE)

$$35s_0+69s_1+7s_2+81s_3+27s_4+36s_5+90s_6+113s_7+91s_8+25s_9=3 \bmod 127$$

$$97s_0+114s_1+121s_2+125s_3+17s_4+48s_5+27s_6+74s_7+90s_8+123s_9=9 \bmod 127$$

$$55s_0+61s_1+52s_2+17s_3+32s_4+114s_5+7s_6+14s_7+114s_8+113s_9=103 \bmod 127$$

$$31s_0+3s_1+63s_2+37s_3+56s_4+39s_5+60s_6+16s_7+32s_8+38s_9=80 \bmod 127$$

$$16s_0+64s_1+21s_2+88s_3+18s_4+90s_5+18s_6+40s_7+24s_8+17s_9=46 \bmod 127$$

$$40s_0+57s_1+1s_2+92s_3+53s_4+59s_5+20s_6+53s_7+66s_8+53s_9=45 \bmod 127$$

$$57s_0+101s_1+114s_2+29s_3+4s_4+76s_5+21s_6+7s_7+49s_8+20s_9=59 \bmod 127$$

$$106s_0+61s_1+44s_2+55s_3+108s_4+111s_5+82s_6+112s_7+41s_8+126s_9=5 \bmod 127$$

$$98s_0+8s_1+48s_2+43s_3+118s_4+105s_5+118s_6+57s_7+97s_8+112s_9=85 \bmod 127$$

$$66s_0+40s_1+94s_2+52s_3+80s_4+76s_5+18s_6+104s_7+97s_8+19s_9=102 \bmod 127$$

$$36s_0+93s_1+98s_2+70s_3+126s_4+69s_5+88s_6+102s_7+99s_8+88s_9=11 \bmod 127$$

LWE « Hello world! » encryption

- Private key:
  - $s \in Z_q^n$.
- Encryption of $m \in \{0,1\}$:
  - Pick $a \in Z_q^n$ uniformly at random and e following χ.
  - c=(a,b) with b=<a,s>+$\lfloor q/2 \rfloor$m+e.
- Decryption:
  - If b-<a,s> closer to 0 than to $\lfloor q/2 \rfloor$ (modulo q) then 0, otherwise 1.

Homomorphic operations

- Let (a;b) and (a';b') and consider ciphertext (a+a';b+b').
  - b=<a,s>+mq/2+e mod q.
  - b'=<a',s>+m'q/2+e' mod q.
  - b+b'=<a+a',s>+q(m+m')/2+e+e' mod q.

- Multiplications are more complicated.
  - Requires tensor product and relinearization.
  - Larger noise amplification than additions.

FHE construction blueprints

- RLWE: let's work over $Z_q[X]/(X^n+1)$ i.e.,
  - We group n LWE pairs in a single ciphertext by (nega)cyclically shifting a.
    - Ciphertext expansion is O(1) (O(n) with LWE).

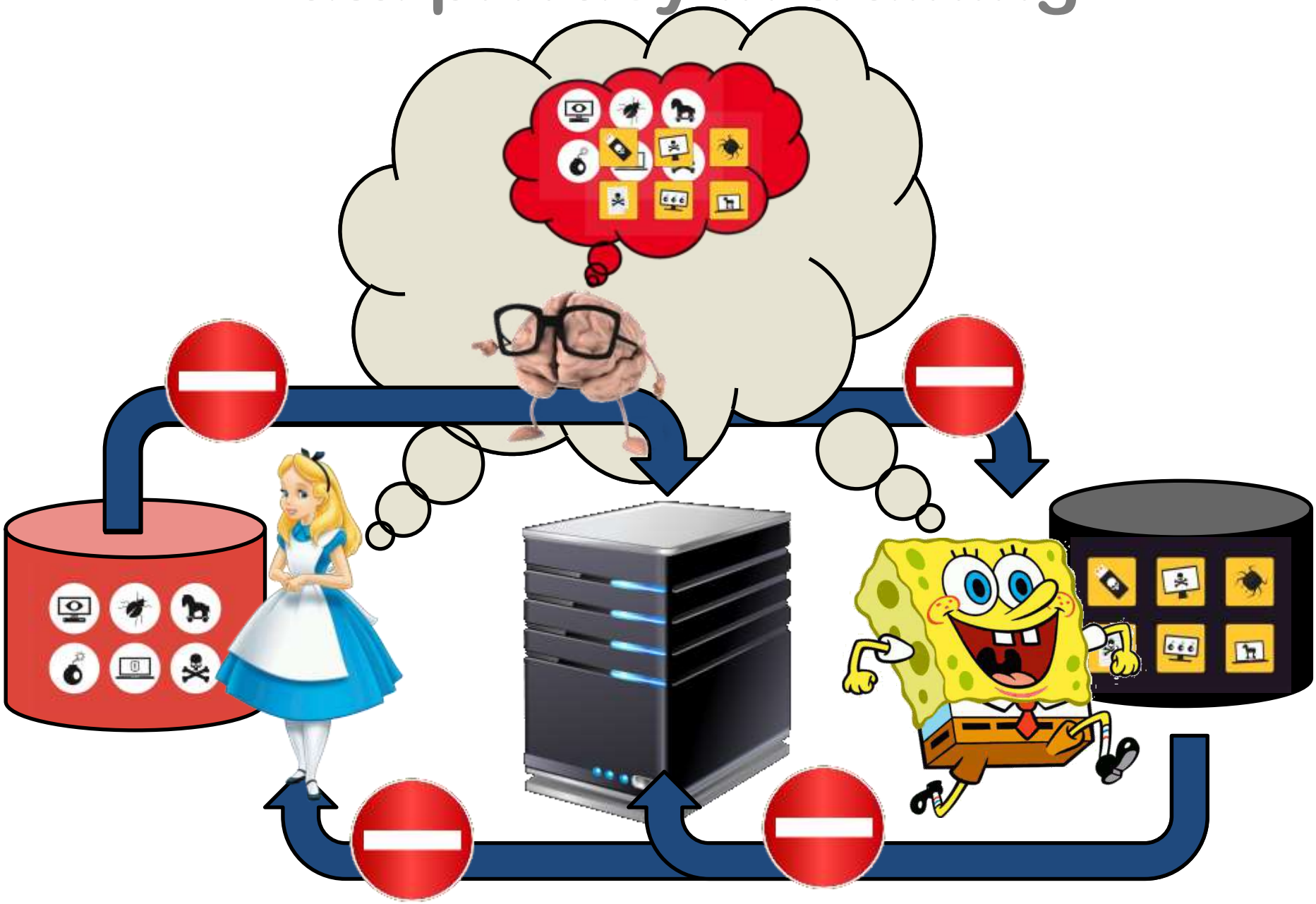
- Dealing with noise amplification:
  - Somewhat FHE: choose the parameters so as to absorb the noise induced by an a priori given (class of) algorithms.
  - Bootstrap: i.e., homomorphically execute a recrypt operation.
    - In some cases, bootstrapping may even compute arbitrary univariate functions at no additional cost (programmable bootstrapping).

The FHE zoo

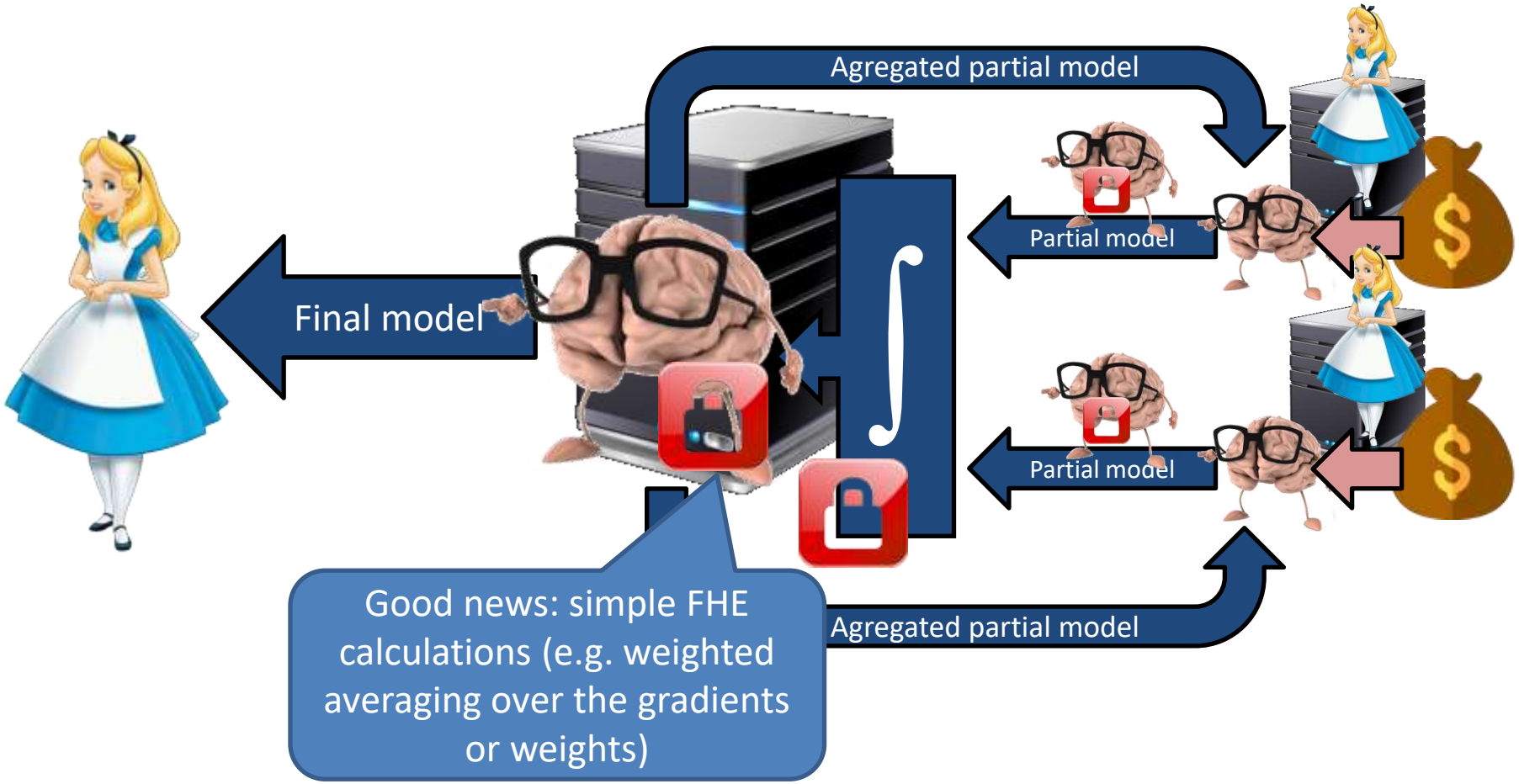
- BFV, BGV:
  - Large plaintext domain.
  - Heavy SIMD //-ism => competitive amortized performances.
  - Some support for non linear ops (beyond polynomial approx.).
  - No efficient bootstrapping.
  - Multiplicative depth dependency.
  - Multikey and threshold variants.
- CKKS:
  - Approximate computations (no message scaling).
  - Large plaintext domain.
  - Heavy SIMD //-ism => competitive amortized performances.
  - No support for non linear ops (beyond poly. approx).
  - No efficient bootstrapping.
  - Multiplicative depth dependency.
  - Weaker than BFV or BGV with respect to passive attackers***.
  - Multikey and threshold variants.
- TFHE (aka CGGI):
  - Efficient bootstrapping.
  - Functionnal bootstrapping => easy non linear ops.
  - Multiplicative-depth independance.
  - Small plaintext domain (32 values max).
  - No batching.
  - Multikey and threshold variants are WIP.

*** Recent updates on this (e.g., Checri et al., CRYPTO'24).

Data privacy in training



Example : FederatedLearning



FHE overhead in FL

- FEMNIST dataset.
- BFV with intensive batching (~8000 slots per ciphertexts).
- ~ 500000 model parameters.
- 1000 clients.
- Full FL cycle (without comm) on a GPU-based HPC cluster (FactoryIA) takes ~20 hours (12 mins per rounds).
- Between 1.5 and 8 secs of FHE calculations per round => only a **0.2 to 1.1% overhead** on the overall procedure duration imputable to FHE.
  - Encryption/decryption timings are negligible.

A. Grivet Sébert, et al., PST'23.

Takeaways

- FHE is probabilistic encryption.
  - So ciphertexts are larger than plaintexts.
- FHE is provable security.
  - FHE is even postquantum (by accident).
- Anything can be computed (in theory) over FHE encrypted data.
- FHE provides confidentiality garantees only against threats coming from where the FHE calculations are made.
  - FHE alone provides no integrity.
- Computing in the encrypted domain is not exactly what most people think.
  - Algorithms always realize (at least) their worst-case complexity!
  - No ifs, no data dependant loop termination, …
  - Strange cost model of FHE operators.

Takeaways (cont'd)

- We* have Somewhat and Bootstrapped FHE with non prohibitive efficiency (and many nice libs).
  - Probably nearing some performance optimum.
  - FHE is and will remain costly.
- We have « FHE-friendly » symmetric crypto.
- We have operational compiler toolchains to program the « FHE computer ».
  - E.g. https://github.com/CEA-LIST/Cingulata.
- We have cool new tools e.g. functional bootstrapping that we're still investigating.
- We are building more versatile multi-key/user schemes and protocols.
- We'll soon(ish) have standards for parameters setting (also thanks to postquantumness).
- Beyond CPA security of FHE is a mess (the community is also working hard on this).

* « We » = the FHE research community.

Some recent papers

- S. Canard, C. Fontaine, D. H. Phan, D. Pointcheval, M. Renard, R. Sirdey: Relations among new CCA security notions for approximate FHE, ePrint 2024/812
- M. Checri, R. Sirdey, A. Boudguiga, J.-P. Bultel: On the practical CPAD security of "exact" and threshold FHE schemes and libraries. CRYPTO 2024
- D. Trama, P.-E. Clet, A. Boudguiga, R. Sirdey: A Homomorphic AES Evaluation in Less than 30 Seconds by Means of TFHE. WAHC@CCS 2023: 79-90
- A. Grivet Sébert, M. Zuber, O. Stan, R. Sirdey, C. Gouy-Pailler: A Probabilistic Design for Practical Homomorphic Majority Voting with Intrinsic Differential Privacy. WAHC@CCS 2023: 47-58
- P.-E. Clet, A. Boudguiga, R. Sirdey, M. Zuber: ComBo: A Novel Functional Bootstrapping Method for Efficient Evaluation of Nonlinear Functions in the Encrypted Domain. AFRICACRYPT 2023: 317-343
- D. Trama, P.-E. Clet, A. Boudguiga, R. Sirdey: Building Blocks for LSTM Homomorphic Evaluation with TFHE. CSCML 2023: 117-134
- A. Choffrut, R. Guerraoui, R. Pinot, R. Sirdey, J. Stephan, M. Zuber: Practical Homomorphic Aggregation for Byzantine ML. CoRR abs/2309.05395 (2023)
- A.-A. Bendoukha, P.-E. Clet, A. Boudguiga, R. Sirdey: Optimized Stream-Cipher-Based Transciphering by Means of Functional-Bootstrapping. DBSec 2023: 91-109
- A. Grivet Sébert, R. Sirdey, O. Stan, C. Gouy-Pailler, « Combining homomorphic encryption and differential privacy in federated learning », PST'23.
- A. Madi, O. Stan, A. Mayoue, A. Grivet-Sébert, C. Gouy-Pailler, R. Sirdey, "A secure Federated Learning framework using FHE and Verifiable Computing", IEEE RDAPS'21.
- A. Grivet Sébert, R. Pinot, M. Zuber, C. Gouy-Pailler and R. Sirdey, "SPEED: Secure, PrivatE, and Efficient Deep learning", ECML'21.
- M. Zuber and R. Sirdey, "Efficient homomorphic evaluation of k-NN classifiers", PETS'21.
- M. Zuber, S. Carpov and R. Sirdey, "Towards real-time hidden speaker recognition by means of fully homomorphic encryption", ICICS'20.