CISPA
HELMHOLTZ CENTER FOR INFORMATION SECURITY

On Circuit Private, Multikey and Threshold Approximate HE

Dr. Kamil Kluczniak and Giacomo Santato

3rd FHE.org conference | 24th March 2024

Table of contents

Approximate HE

HE and Approximate HE

- Exact HE schemes (BGV,BFV,TFHE,FHEW)

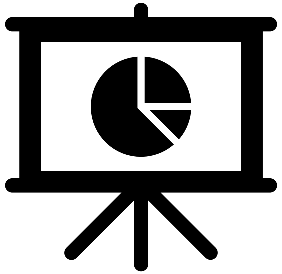
$$m \xrightarrow{Encrypt} ct \xrightarrow{Eval(f)} \widetilde{ct} \xrightarrow{Decrypt} f(m)$$

- Approximate HE schemes (CKKS)

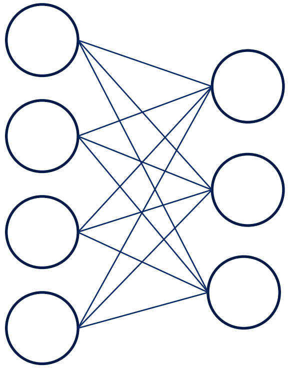
$$m \xrightarrow{Encrypt} ct \xrightarrow{Eval(f)} \widetilde{ct} \xrightarrow{Dec} \boldsymbol{f(m)+e}$$

Why Approximate HE?

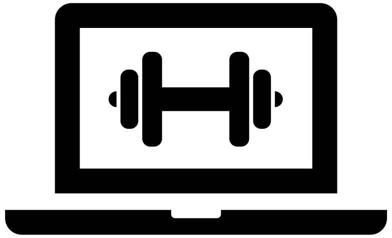
- Suitable for many applications that work with **floating point numbers** and already tolerate approximations.



Data Analysis



ML inference



ML training

- Most of them operate on **real/complex** numbers, allowing to homomorphically evaluate difficult functions by considering some **polynomial approximations** of them.

The CKKS scheme (2016)

- Has an encoding/decoding operation that allows to operate on discretized **real and complex** numbers by mapping them on/from Gaussian integers.

- A fresh encryption of m is a pair

$$(b, a) \in \mathcal{R}_Q^2 \text{ s.t. } b = s \cdot a + m + e \ (mod\ Q)$$

where e is sampled from a Gaussian distribution.

- Decryption returns

$$b - s \cdot a \ (mod\ Q)$$
$$\boldsymbol{m} + \boldsymbol{e} \approx \boldsymbol{m}\,.$$

IND-CPA security of CKKS

- [CKKS’17] Proved the scheme to be **IND-CPA** secure under the RLWE assumption.

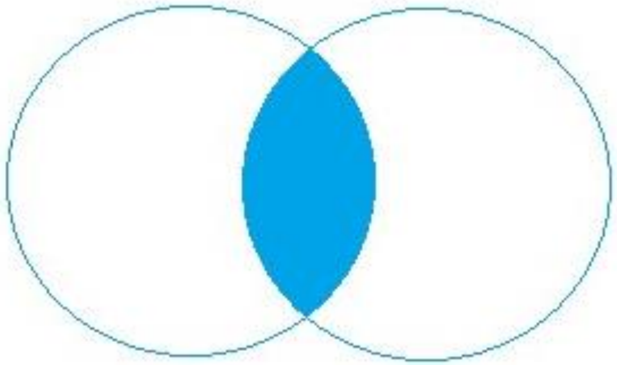
- [LM’21] Showed a **key recovery attack** when obtaining the error of enough decryptions. In fact, knowing $f(m)$ allows to retrieve exactly* the approximation error after decryption.

- [LMSS’22] Suggested a fix by adding a post-processing phase with the addition of **extra noise during decryption** and with the introduction of **IND-CPA-D** security definition.

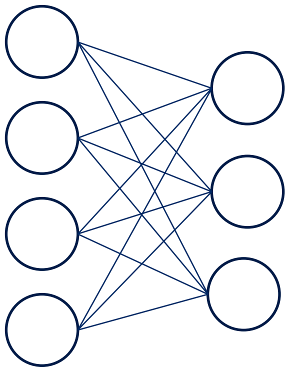
Circuit Privacy

Keeping f secret

- In many applications, it is important that the evaluated function f **remains secret.**



Private set Intersection



ML inference

Analysis of genomic data

- Example: Performing a privacy-preserving **machine learning inference**, using HE schemes without any additional properties, risks to **reveal information on the model** (i.e. the function f) to an adversary.

What is Circuit Privacy?

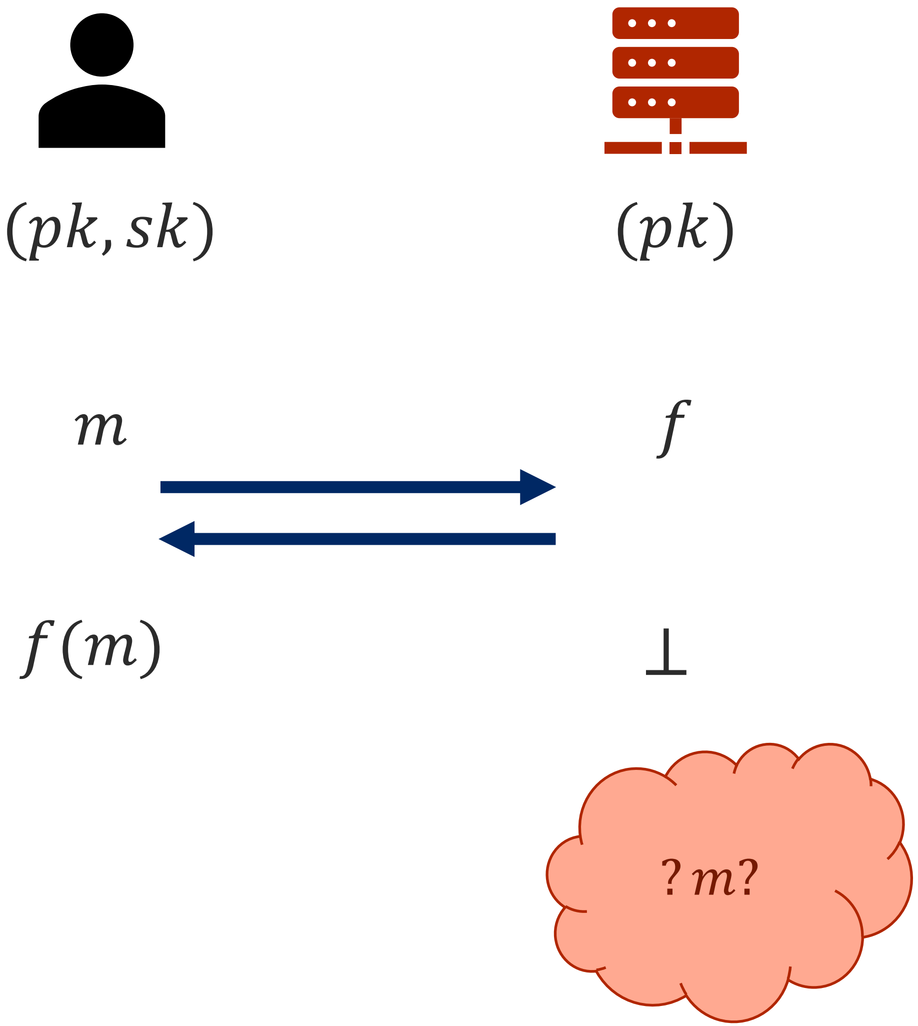
- Circuit privacy is a **security definition** for homomorphic encryption.

- While, in general, the main focus is on the security of the encrypted message, circuit privacy studies the secrecy of the evaluated function.

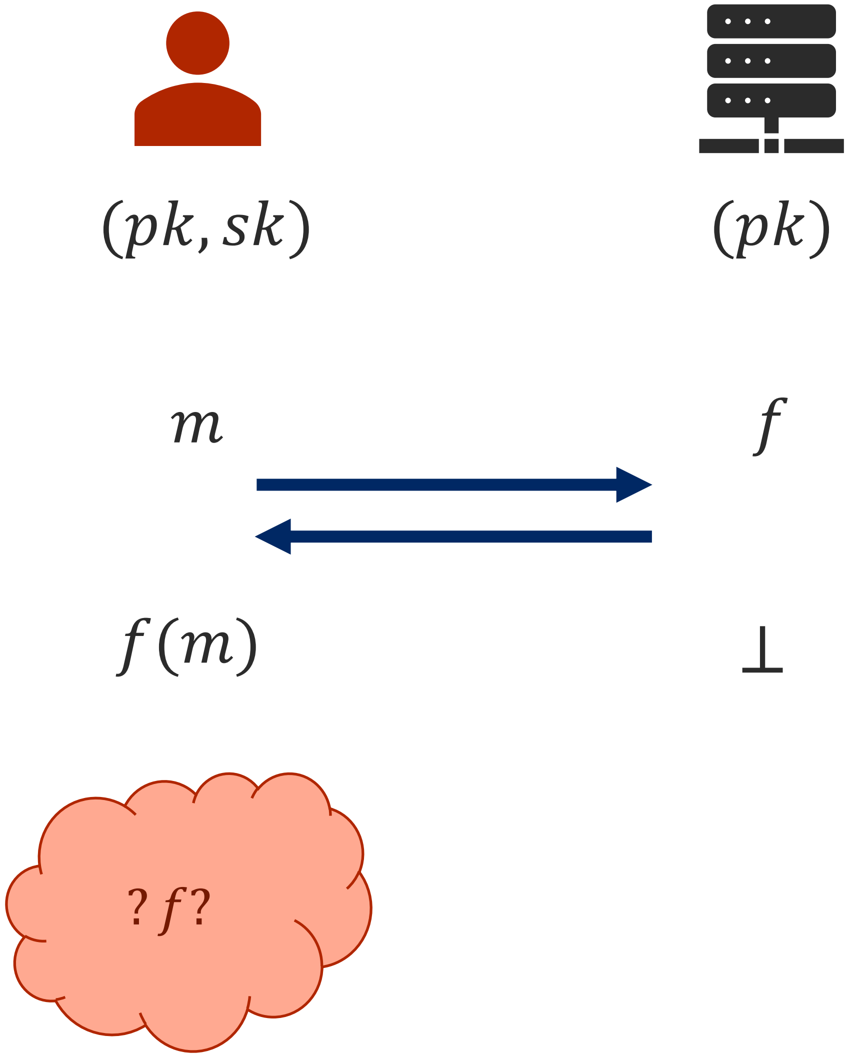
- The objective is to not reveal meaningful information except for the final result of the computation $\boldsymbol{f(m)}$.

- This is **not automatically guaranteed** by HE, and we need additional requirements.

IND-CPA



Circuit Privacy



Circuit Privacy in Exact HE

- Simulation definition:

$$Sim\big(pk, f(m)\big) \approx Eval(f, ct)$$

for every function f and for every $ct \leftarrow Enc(m)$.

- **Problems** when adapting to the approximate setting:

1. Implicitly **requires the correctness** of the scheme. The ciphertext on the right it is not an encryption of $f(m)$, but rather of $f(m) + e$.

2. The adversary **obtains the error** associated to the ciphertext. The magnitude of this error already reveals information about the function, like the size of the circuit or its topology.

Noise growth in CKKS Evaluation

- **Addition**:

$$Enc(m_1) \oplus Enc(m_2) \longrightarrow Enc(m_1 + m_2)$$

$$\boldsymbol{e_1} \qquad \boldsymbol{e_2} \qquad \boldsymbol{e_1 + e_2}$$

- **Multiplication**:

$$Enc(m_1) \otimes Enc(m_2) \longrightarrow Enc(m_1 * m_2)$$

$$\boldsymbol{e_1} \qquad \boldsymbol{e_2} \qquad \boldsymbol{m_1\, e_2 + e_1\, m_2 + e_1\, e_2}$$

We recall that every computable function can be represented as a circuit as composition of addition and multiplication gates.

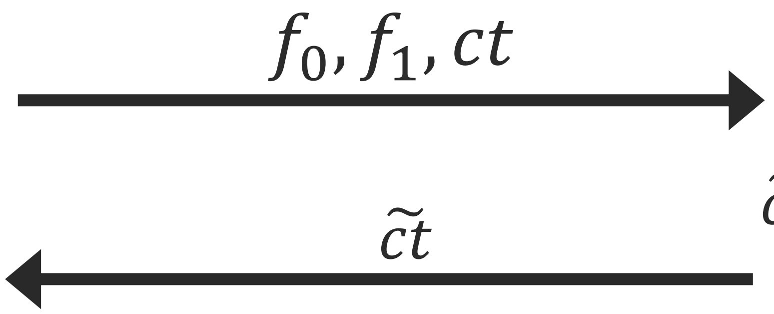
Circuit Privacy in Approximate HE

We introduce a relaxed circuit privacy definition: **IND-CP**.

(pk, sk) (pk)

The adversary must choose f_0, f_1, m s.t. $f_0(m) = f_1(m)$:

$ct \leftarrow Enc(m)$



$\widetilde{ct} \leftarrow Eval(f_b, ct)$

$Guess\ b$

Result I: Achieving IND-CP in CKKS

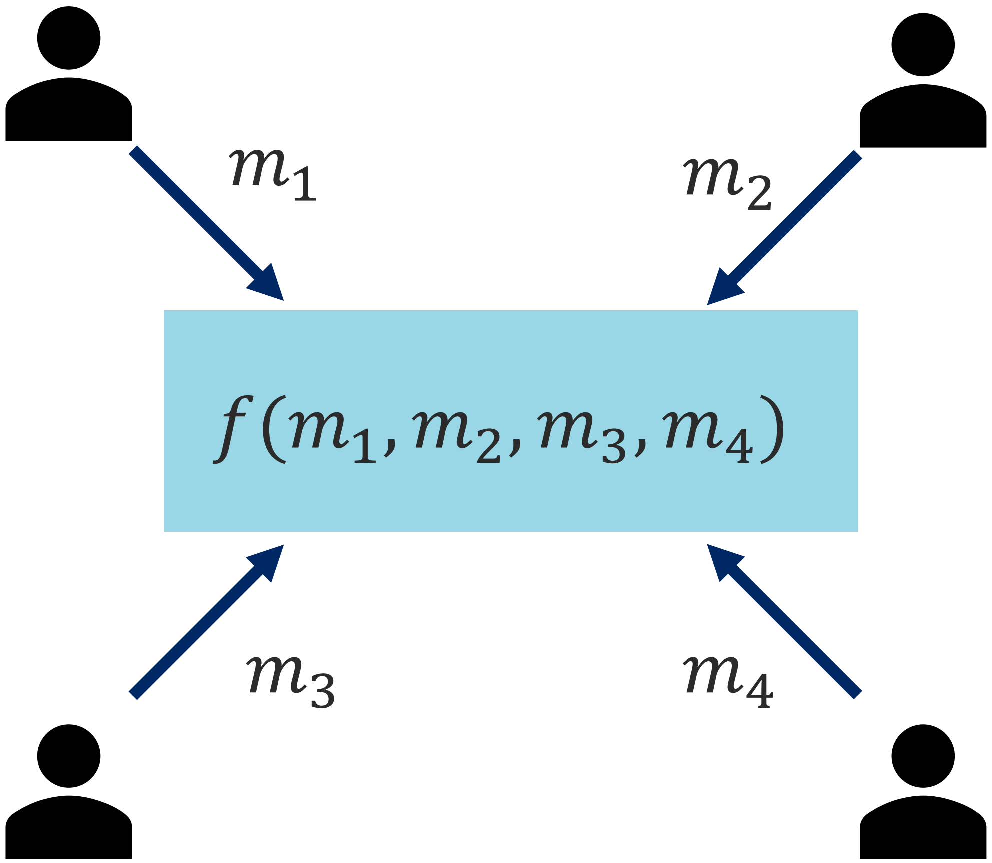
- Simple **post-processing** phase in CKSS **evaluation**:
  1. Add a fresh encryption of zero.
  2. Add Gaussian noise to the b term.
- **Extensive analysis** of the optimal Gaussian noise:
  – KL divergence-driven analysis.
  – **Tightness** of the parameters.
  – Generalize for different function spaces (like functions of fixed depth).
- **Parameters** for application to privacy-preserving **machine learning inference**.

(On the right, bits of additional Gaussian noise to add for 128-bits of IND-CP sec.)

depth \ width	$w=1$	$w=2^3$	$w=2^5$	$w=2^8$
$d=1$	85.50	87.67	89.54	92.50
$d=2$	97.08	100.99	104.63	110.51
$d=3$	108.08	113.45	118.76	127.53

Multiparty HE

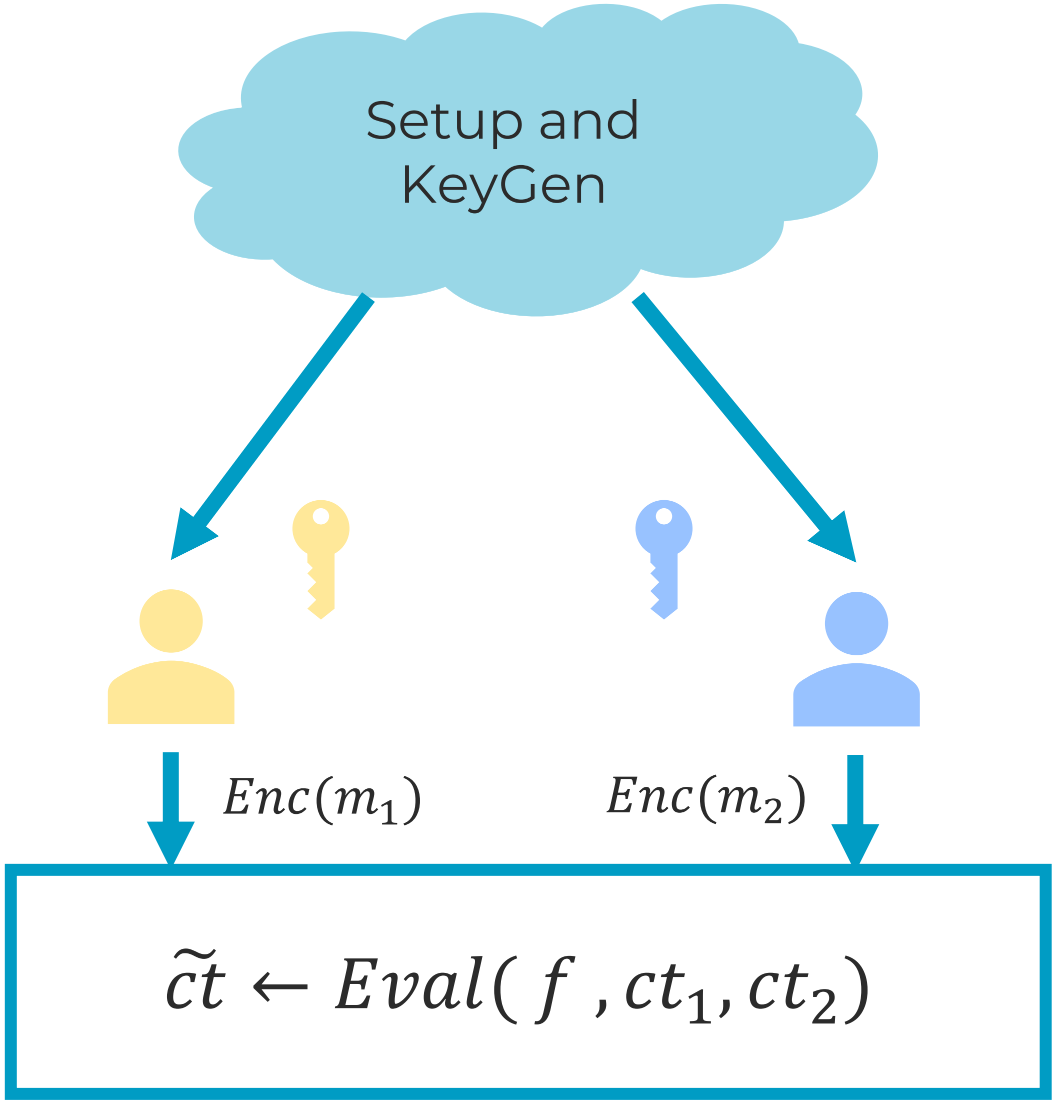
Multiparty computation



- Multiparty computation allows multiple parties to cooperate together to **evaluate a function**.
- Each party learns the final result **without learning** anything about the inputs of the others.
- One of the many ways to construct MPC protocols is by using HE.

Multiparty HE (I)

- We consider the case with a fixed set of two parties in the approximate setting.
- After the setup phase, the parties encrypt their messages and then homomorphically evaluate a chosen function on them.
- In this way, they obtain a **common ciphertext** containing the final result.

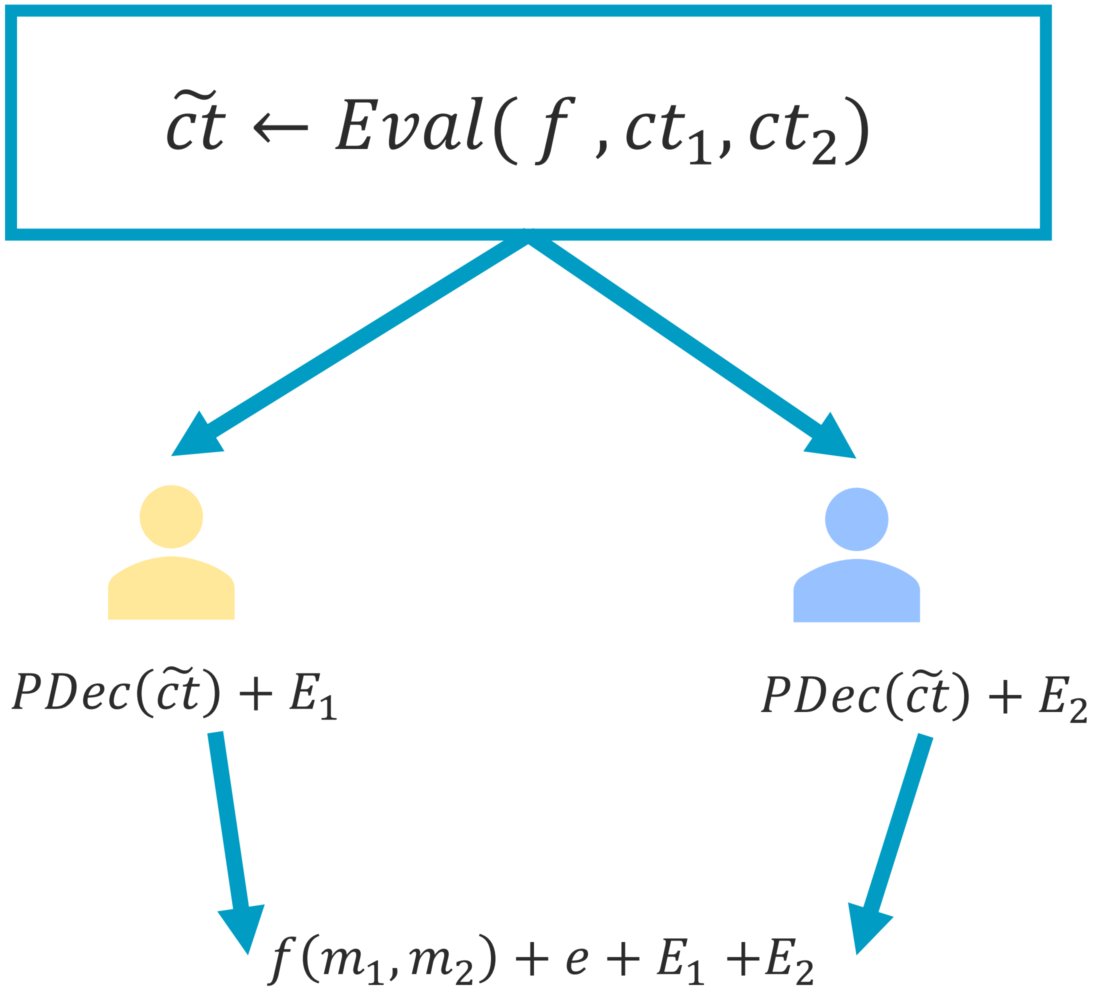


Multiparty HE (II)

- After the evaluation phase, multiparty HE schemes sum together the **partial decryptions** of the parties to obtain the final result.

$$\mathrm{Dec}(\mathrm{b},\mathrm{a}) = b - sk * a$$
$$\mathrm{PDec}(\mathrm{b},\mathrm{a}) = sk_i * a$$

- **Additional noise** must be added to each decryption share to avoid the leakage of the secret keys.



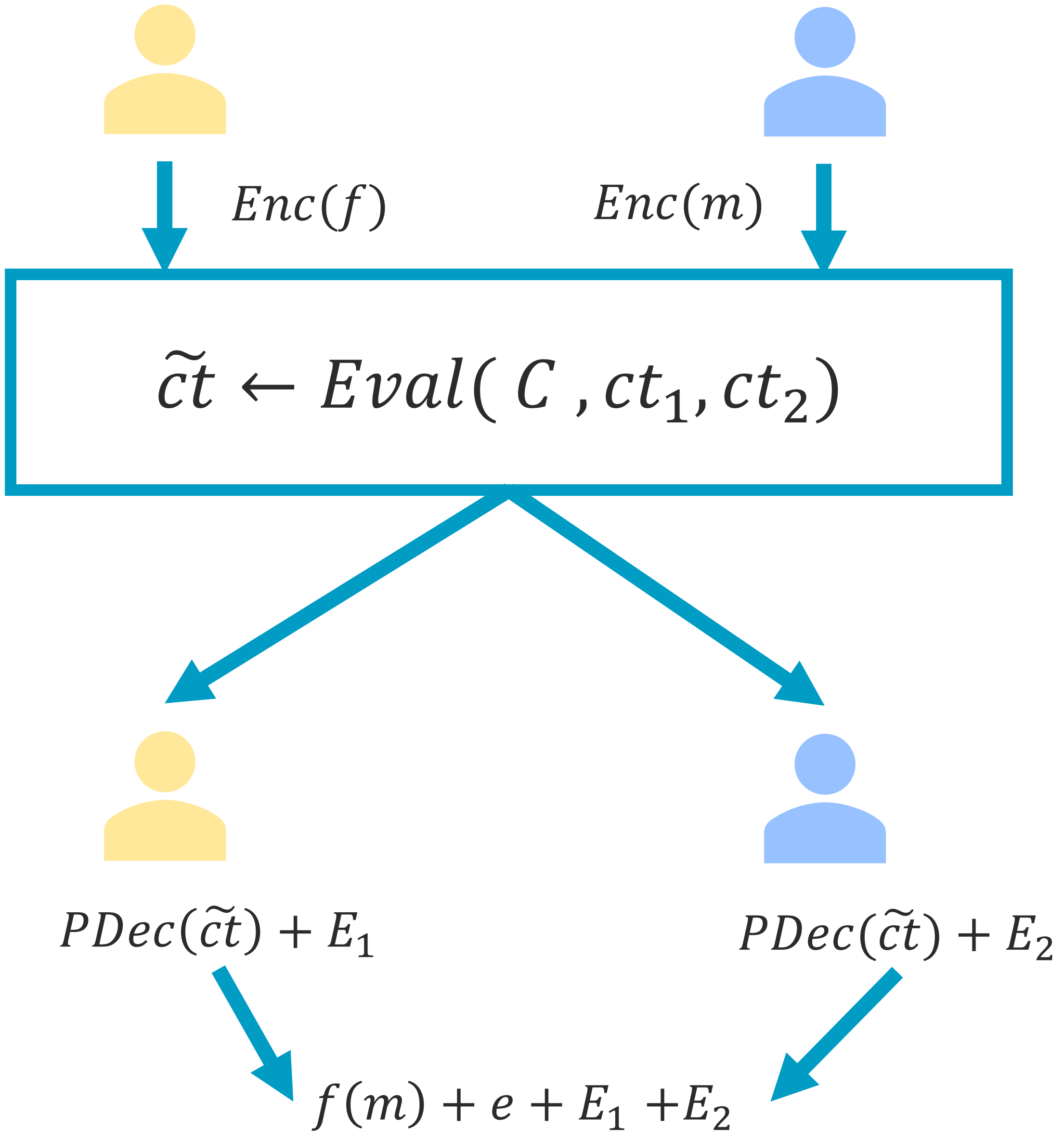
Multiparty HE and CP

- It is possible to achieve **circuit privacy** using 2-party HE, by homomorphically evaluating a universal circuit.

- A Universal Circuit (C) is a function:

$$C : \mathcal{F} \times \mathcal{M} \longrightarrow \mathcal{M}$$
$$(f, m) \mapsto f(m)$$

- It is 'folklore' belief that this construction reduces the amount of additional noise required.



Result II: CP from Approx Multiparty HE

- Security analysis of the multikey CKKS scheme [CDKS'19]:
  - Post-processing in the **evaluation phase** and quantified the noise flooding in the **partial decryption** phase.
  - New security definition that allows to **achieve CP**, considers the impact of **partial decryptions** and of the **IND-CPA-D** related attacks.
- Extensive analysis of the optimal Gaussian noise:
  - KL divergence-driven analysis.
  - **Tightness** of the parameters.
  - Generalized with (s,c)-bit security definition to better allow **trade-offs** between security and efficiency, depending on the application.

Work in progress: Impact on Exact Multiparty FHE

- Many recent schemes ([DWF22],[CSS+22],[BS23]) are using **Rényi divergence** to achieve (exact) Threshold FHE.
  - Using divergencies prevents from achieving security definitions that involve statistical simulation.
  - Each one of these papers introduces at least one new game-based security definition.
  - Analysis of the new security definitions: survey with comparison among them and attacks against two definitions that were unsuitable for use. **Already in the Appendix!**
  - Uninstantiability of a Random Oracle transform from OW-CPA security to IND-CPA security. **Already in the Appendix!**

Contacts

Full paper:

https://eprint.iacr.org/2023/301

Giacomo Santato

giacomo.santato [at] cispa.de

Dr. Kamil Kluczniak

kamil.kluczniak [at] gmail.com