



NTNU

Norwegian University of
Science and Technology

Computational FHE Circuit Privacy for Free

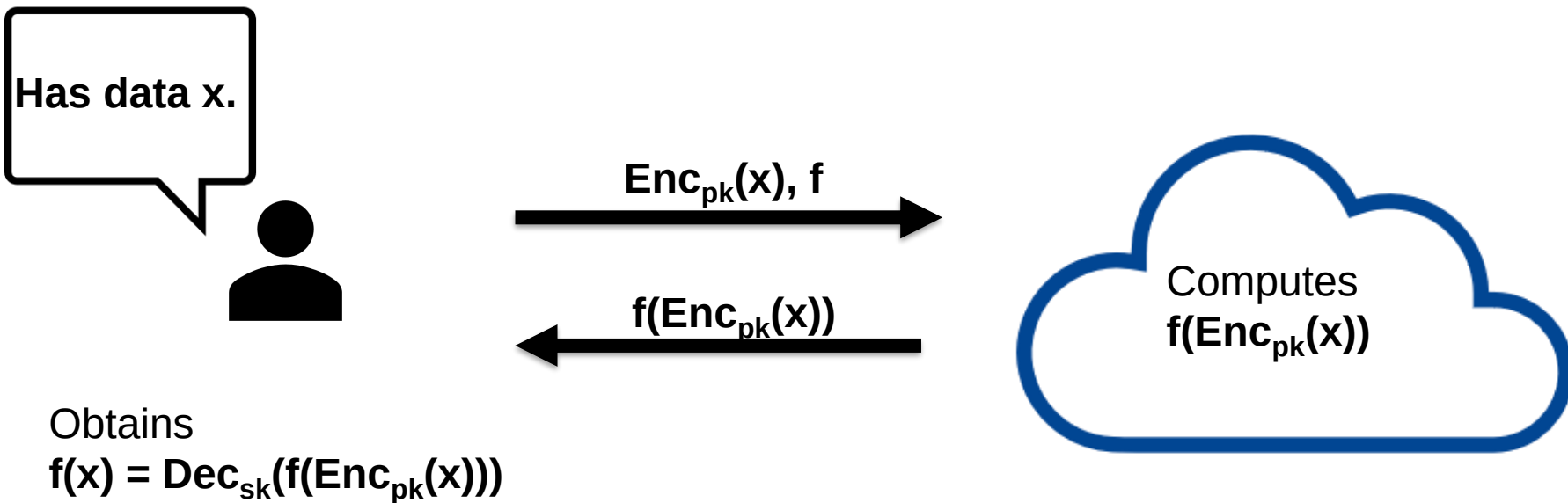
Anamaria Costache, **Lea Nürnberger**, Tjerand Silde

Norwegian Crypto Seminar 2023

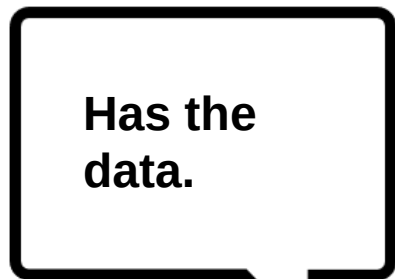
What is Fully Homomorphic Encryption?

- Data can be protected by cryptography while being stored or transmitted.
- It needs to be decrypted to be worked with.
- If data needs to be continuously accessed, it needs to be decrypted every time.

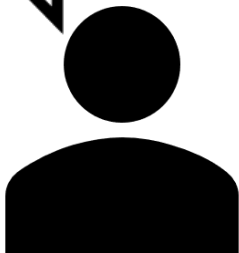
What is Fully Homomorphic Encryption?



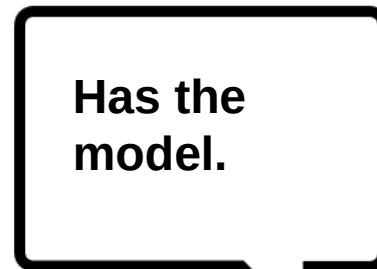
What is Circuit Privacy?



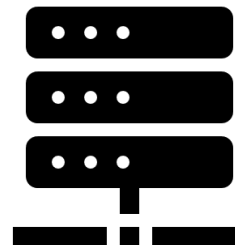
**Has the
data.**



- The client does not want the server to learn the data.
- The server does not want the client to learn the model.



**Has the
model.**



NTNU

Norwegian University of
Science and Technology

What is Circuit Privacy?

- FHE allows the client to keep the data private.
- Circuit Private FHE allows the server to additionally keep the model private.

How is information about the computations leaked?

- Computations on ciphertexts can influence the distribution of the ciphertexts.
- FHE ciphertexts are mostly inherently noisy. Computations affect the distribution of this noise.
 - The standard deviation of the noise after an addition is sum of the two previous standard deviations. Therefore, if an adversary knows the previous standard deviations, it knows the computation.

Current Approaches

- **Noise Flooding** (see for example [1],[2])
 - After having performed all computations, add enough noise to the ciphertext, that the statistical distance between the distributions of a “fresh” ciphertext and the result ciphertext resulting of the computation becomes negligible.

[1] Gentry, Craig: A Fully Homomorphic Encryption Scheme, PhD Thesis, Stanford, 2009.

[2] Kluczniak, Kamil; Santato, Giacomo: On Circuit Private, Multikey and Threshold Approximate Homomorphic Encryption, eprint, 2023.

Current Approaches

- **Ducas-Stehlé Washing Machine** (see [3])
 - Repeatedly apply an operation called bootstrapping to eventually strip the ciphertext of all information about the computation.
- **Use Programmable Bootstrapping for Re-randomization** (see [4])

[3] Ducas, Lucas; Stehlé, Damien: Sanitization of FHE Ciphertexts, EUROCRYPT 2016.

[4] Kluczniak, Kamil: Circuit Privacy for FHEW/TFHE-Style Fully Homomorphic Encryption Schemes in Practice, eprint, 2022.

Challenges with Current Approaches

- Noise flooding requires the usage of large parameter sets that make computations slower or impossible.
- The Ducas-Stehlé washing machine requires repeated applications of bootstrapping, which is a computationally costly operation for most schemes.
- Programmable bootstrapping is a solution for only very few schemes (TFHE and FHEW).

Our Idea

- The above techniques are necessary, if we want to achieve a statistical indistinguishability between ciphertext distributions of “fresh” ciphertexts and ciphertexts after an evaluation.
- Can we achieve circuit privacy with less expensive techniques if we downgrade to computational indistinguishability?

Our Contributions

- We consider 4 cases.
 - Semi-honest adversary without knowledge of the secret key.
 - Semi-honest adversary with knowledge of the secret key.
 - Malicious adversary without knowledge of the secret key.
 - Malicious adversary with knowledge of the secret key.
- We prove that BGV can be made circuit private in all those 4 cases without need for expensive machinery as noise flooding or bootstrapping.

Our Contributions

- An adversary with knowledge of the secret can use the ciphertext noise gain information about the computation.
- We use ideas from [5] to show how the noise can be stripped of all information about the computation.
- An adversary without knowledge of the secret key can use the ciphertext distribution to gain information.
- We show that in BGV the ciphertexts never leak any information.

[5] Costache, Anamaria; Nürnberger, Lea; Player, Rachel: Optimisations and Trade-Offs for HELib, CT-RSA 2023.

Some Technical Details

Gentry's statistical definition

Definition 1 ([Gen09]). We say that a homomorphic encryption scheme $\mathcal{E} = (\text{KeyGen}, \text{Enc}, \text{Eval}, \text{Dec})$ is (statistically) circuit private for circuits in $\mathcal{C}_{\mathcal{E}}$, if for any key set (sk, pk, evk) output by $\text{KeyGen}(1^\lambda)$, for any function f on plaintexts that has a corresponding circuit $C_f \in \mathcal{C}_{\mathcal{E}}$ and any fixed ciphertexts $ct_1, \dots, ct_k$ that are the encryption by $\text{Enc}_{\mathcal{E}}(\cdot, pk)$ of plaintexts $m_1, \dots, m_k$ respectively, the following holds for the distributions over the random coins in $\text{Enc}_{\mathcal{E}}(\cdot, pk)$ and $\text{Eval}_{\mathcal{E}}(\cdot, C_f, evk)$:

$$\Delta(\text{Enc}_{\mathcal{E}}(f(m_1, \dots, m_k), pk), \text{Eval}_{\mathcal{E}}(ct_1, \dots, ct_k, C_f, evk)) \leq \text{negl}(\lambda).$$

Some Technical Details

Our computational definition

Definition 14. Let $\mathcal{E} = (\text{KeyGen}, \text{Enc}, \text{Eval}, \text{Dec})$ be a homomorphic encryption scheme for circuits in $\mathcal{C}_{\mathcal{E}}$. Let λ be a security parameter. We say $\mathcal{E}$ is *computationally circuit private* if for any $(sk, pk, evk) \leftarrow \text{KeyGen}(1^\lambda)$, for any function f on plaintexts that has a corresponding circuit $C_f \in \mathcal{C}_{\mathcal{E}}$ and any fixed ciphertexts $ct_1, \dots, ct_k$ that are an encryption by $\text{Enc}(\cdot, pk)$ of $m_1, \dots, m_k$ the two distributions over the random coins in $\text{Enc}(\cdot, pk)$ and $\text{Eval}(\cdot, C_f, evk)$ of $\text{Enc}(f(m_1, \dots, m_k), pk)$ and $\text{Eval}(ct_1, \dots, ct_k, C_f, evk)$ are computationally indistinguishable. We say that $\mathcal{E}$ is maliciously computationally circuit private if the keys and the ciphertexts may be maliciously generated.

Some Technical Details

The RLWE Problem

Definition 5 (Definition 4 in [BGV12]). For the security parameter λ let $\mathcal{R}$ be a power of two cyclotomic ring. Let $q = q(\lambda) \geq 2$ be an integer and let $\mathcal{R}_q = \mathcal{R}/q\mathcal{R}$. Let χ be a distribution over $\mathcal{R}$. The $RLWE_{d,\lambda,\chi}$ problem is to distinguish the following two distributions: in the first distribution (a_i, b_i) is sampled uniformly randomly from $\mathcal{R}_q^2$. In the second distribution, $s \stackrel{\$}{\leftarrow} \mathcal{R}_q$ is drawn uniformly and (a_i, b_i) is sampled by drawing $a_i \stackrel{\$}{\leftarrow} \mathcal{R}_q$, $e_i \leftarrow \chi$ and setting $b_i = a_i s + e_i$. The RLWE assumption is, that the $RLWE_{d,\lambda,\chi}$ problem is infeasible.

Some Technical Details

- BGV is based on RLWE.
- Fresh BGV ciphertexts are RLWE samples, and if the RLWE problem is hard, they are computationally indistinguishable from uniform random.
- We can prove that even after computations have been performed on them, they remain RLWE samples.
- They therefore remain computationally indistinguishable from uniform random.
- They are therefore computationally indistinguishable from one another.

Thank you for your attention!

<https://eprint.iacr.org/2023/1597>