How to compute on encrypted Data

Fully Homomorphic Encryption

Alexandru Eugen Bulboaca, Lilika Markatou

Introduction

Why is Homomorphic Encryption relevant?

- Encryption generally protects data at **rest** and in **transit** but **not during processing**
- Relevant for privacy-critical use cases, such as **Medical Analysis, Confidential Machine Learning, Recommender Systems**

FHE enables **any computation to be performed on encrypted data**

In 2009 Craig Gentry[1,2], introduced the first **Fully Homomorphic Encryption (FHE)**

Methodology

How was the research performed?

The research was based on the **Forward Referencing methodology,** based on Pikert FHE taxonomy[3], guided by the **Research Questions:**

1. What are HE schemes? How is FHE **constructed**?
2. How does FHE **compare** in terms of **functionality, performance,** and **security?**
3. How does FHE compare to **other** privacy-enhancing **techniques** (MPC, ORAM, TEE, StE)?
4. How does or could FHE **integrate** in the industry?

Background

What are the FHE schemes?

Pre-2009: Limited homomorphic capabilities - PHE, SwHE

Gentry's via **bootstrapping, squashing** over **Lattices**

DGHV[4] over **integers,** introduced **batching (SIMD)**

BGV[5] introduced **LFHE** via **mod & key switching**

B/FV[6,7] **scale invariant,** improved **error growth**

GSW[8] with **no switching,** better **performance**

FHEW[9] via **new NAND** approach, fast **bootstrapping**

TFHE[10] reduced **key sizes,** optimised **bootstrapping**

CKKS[11] uses real numbers - **approximate arithmetic**

CHIMERA[12] allows **switching** between schemes

1 AGCD

2 (R)LWE

3 (R)LWE

4 (R)LWE

Different **constructions** — Different **characteristics**

Characteristics

What attributes do Homomorphic Encryption schemes present?

Why FHE schemes?

Feature	PHE	LFHE	FHE
Supported operations	Add **OR** Multiply	Add **AND** Multiply	Add **AND** Multiply
Number of operations	Unlimited	Limited (fixed depth)	Unlimited
Performance	Very High	High	Low
Versatility	Low	Medium	High
Use Cases	Voting Protocols	Bioinformatics Data	BDP, ML

Which FHE scheme?

Scheme	BGV, B/FV	FHEW, TFHE	CKKS
Features	Batching Fast linear functions	Fast Boostrapping Fast comparisons	Efficient batching Real Numbers
Drawbacks	Slow Bootstrapping, Slow complex operations	No batching, Frequent Bootstrapping	Slow Bootstrapping, Accuracy Loss
Application	Big data	Data streaming	Machine Learning

Security

Schemes can achieve **Semantic Security,** extended to **IND-CCA1** security level

IND-CCA2 infeasible given concept malleability

Performance

Key Generation procedure is very slow

Polynomial overhead compared to plaintext

Displays **large ciphertexts and key sizes**

Comparison

How does FHE compare to other privacy-enhancing techniques?

Other technologies

MPC FHE allows parties to perform **computations on secret shares locally**

Reduces communication rounds, facilitates delegation, **without reavealing intermediate data**

ORAM FHE secures **data content - ORAM data patterns**

StE StE offers fast encrypted index search, **leaking access pattern** data

FHE ensures **confidentiality during updates and queries**, at the cost of efficiency

TEE TEE offers **trusted spaces** for **secured** computation

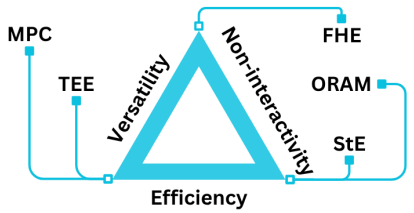
FHE offers **computation security** in **untrusted** spaces

Key Points

FHE does **not require or permit interactions** or shared states

FHE facilitates **confidential data content**, but not **data access patterns**

FHE is a general-purpose non-interactive tool, with **reduced efficiency**



Single-user, low-interaction scenarios

Discussion and Future Work

How can FHE integrate in the industry?

Impractical for mass adoption due to:

Reduced performance
Lack of standardisation
Insufficient interoperability

FHE computations are **relatively slow** for adoption, despite extensions for SIMD or fast bootstrapping

hardware accelerators advancements are needed

Common standard schemes, practices, and use accessibility to developer are needed for integration

government-companies-researchers consensus to establish standard schemes, SDKs and APIs

Developing interactive protocols, such as MKFHE [13], can achieve the benefits of multiple technologies, becoming more practical

research FHE interaction with existing infrastructure

Conclusion

Takeways

FHE helped lay the groundwork for **zero-knowledge computing, supported ethical innovation and increased privacy,** but it does **not replace human responsibility**, policy, or oversight.

FHE makes possible a future where data can be used without being seen.

References

[1] Craig Gentry. Fully homomorphic encryption using ideal lattices. volume 9, pages 169–178, 05 2009.
[2] Craig Gentry. Computing arbitrary functions of encrypted data. Communications of the ACM, 53(3):97–105, 3 2010.
[3] Chris Peikert. A decade of Lattice Cryptography. Foundations and Trends® in Theoretical Computer Science, 10(4):283–424, 1 2016.
[4] Marten Van Dijk, Craig Gentry, Shai Halevi, and Vinod Vaikuntanathan. Fully Homomorphic Encryption over the Integers. 1 2010.
[5] Zvika Brakerski, Craig Gentry, and Vinod Vaikuntanathan. (Leveled) Fully Homomorphic Encryption without Bootstrapping. ACM Transactions on Computation Theory, 6(3):1–36, 7 2014.
[6] Zvika Brakerski. Fully Homomorphic Encryption without Modulus Switching from Classical GapSVP. 1 2012.
[7] Junfeng Fan and Frederik Vercauteren. Somewhat practical fully homomorphic encryption. Cryptology ePrint Archive, Paper 2012/144, 2012.
[8] Craig Gentry, Amit Sahai, and Brent Waters. Homomorphic Encryption from Learning with Errors: Conceptually-Simpler, Asymptotically-Faster, Attribute-Based. 1 2013.
[9] L'eo Ducas and Daniele Micciancio. FHEW: Bootstrapping homomorphic encryption in less than a second. 1 2015.
[10] Ilaria Chillotti, Nicolas Gama, Mariya Georgieva, and Malika Izabachene. TFHE: Fast fully homomorphic encryption over the torus. Journal of Cryptology, 33(1):34–91, 4 2019.
[11] Jung Hee Cheon, Andrey Kim, Miran Kim, and Yongsoo Song. Homomorphic encryption for arithmetic of approximate numbers. 1 2017.
[12] Christina Boura, Nicolas Gama, Mariya Georgieva, and Dimitar Jetchev. CHIMERA: Combining Ring-LWE-based fully homomorphic encryption schemes. Journal of Mathematical Cryptology, 14(1):316–338, 8 2020.
[13] Pratyay Mukherjee and Daniel Wichs. Two round multiparty computation via multi-key FHE. 1 2016.