Homomorphic Encryption (HE)

What is Homomorphic Encryption

Homomorphic encryption encrypts the data in a way that computation/learning can happen on the data w/o decrypting the data. Once data is encrypted in this fashion it can be moved to a central location for training/analytics or compute without sacrificing privacy.  HE, is expensive to implement requires lot of compute, as of now the GPU (Graphic Processing Unit) is best suited to run the HE algorithm.

Why was it developed and how it works.

In today's data-driven landscape, safeguarding sensitive information is paramount. The shift to cloud computing and storage has transformed data processing, offering unparalleled resource availability and accessibility while reducing workload significantly. This revolution has spurred immense demand for outsourced applications, allowing clients to upload data to the cloud for processing and accessing results conveniently. While this offers substantial benefits, it also raises concerns about exposing sensitive data to third-party service providers.

Traditional encryption methods typically require data to be decrypted for processing, exposing it in its original form. However, homomorphic encryption breaks this norm by enabling computations on encrypted data, delivering encrypted results to users. This groundbreaking approach not only allows processing of encrypted data but also maintains utmost privacy throughout the entire process, ensuring confidentiality while enabling seamless computation. The egress cost associated with data leaving the cloud can we very significant, HE can play a vital role to perform analytic on the data without the egress cost.

## Homomorphic encryption enables mathematical operations directly on ciphertexts, generating encrypted results that, upon decryption, align with the outcomes of operations conducted on plaintext

- Enable computing on encrypted data.
- Enables multiple parties to collaborate without sharing data.
- Apply Artificial Intelligence/Machine learning/Data Analytic

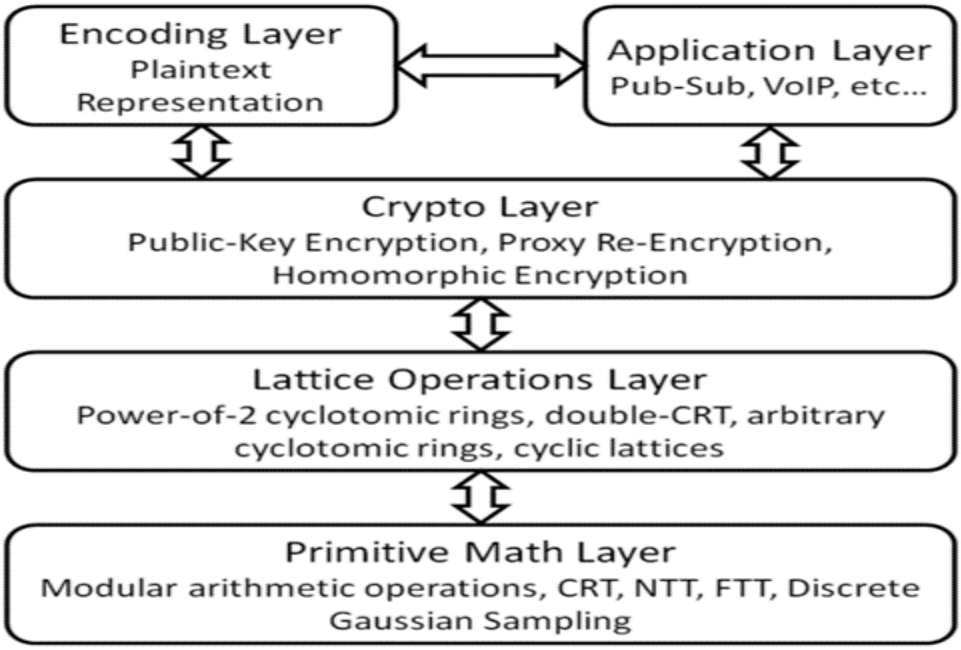
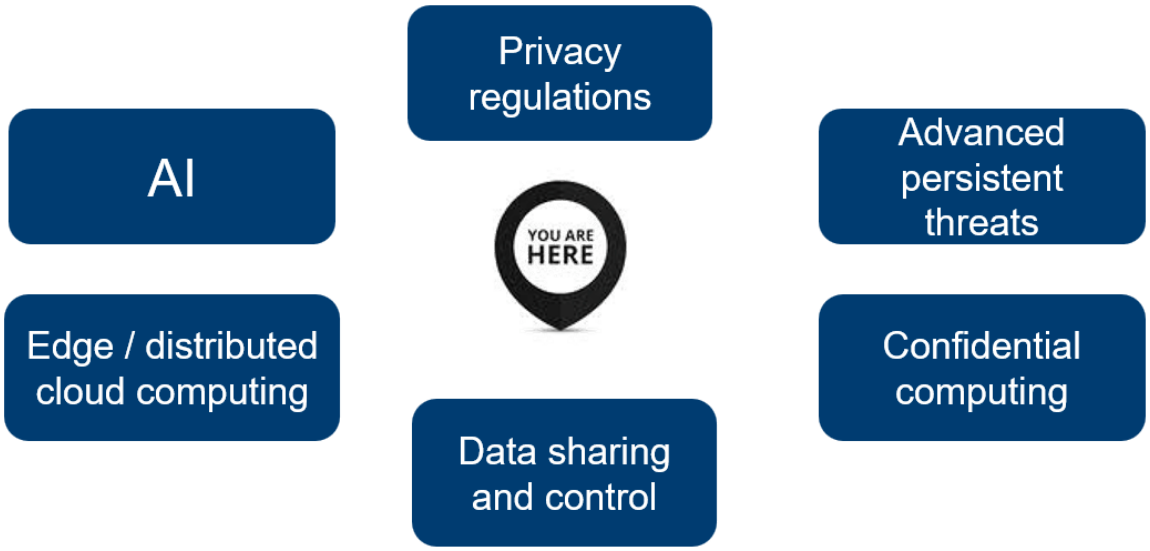
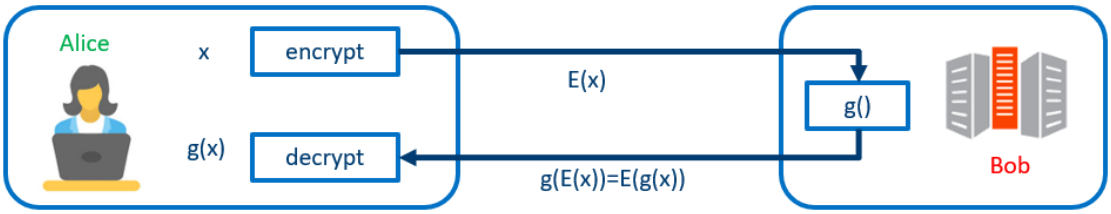
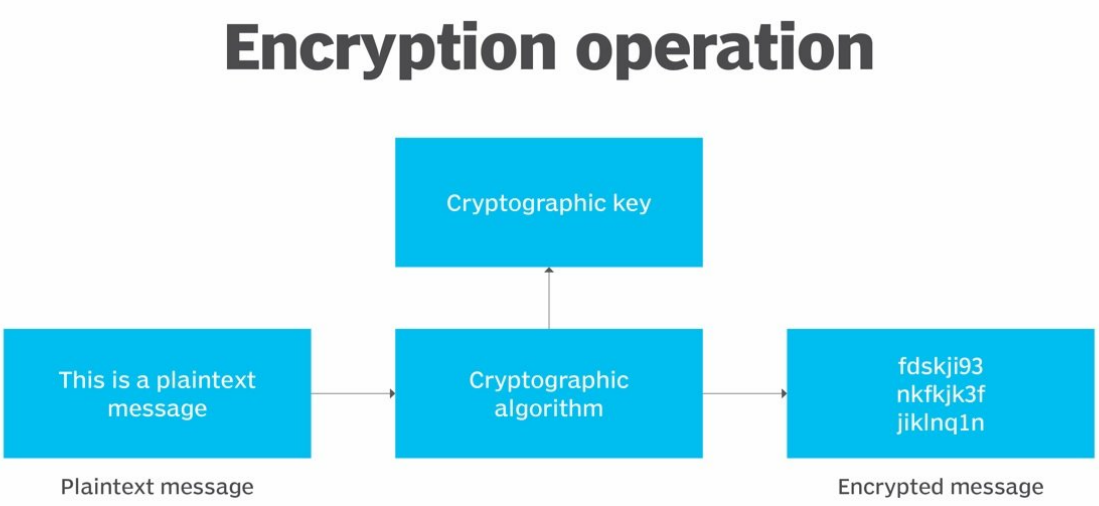



Figure 1: High-level PALISADE architecture






Secure collaboration between Alice and Bob




Types of Homomorphic Algorithms:

Partially Homomorphic Encryption (PHE): Allows either addition or multiplication operations on encrypted data, but not both. Examples include RSA and ElGamal encryption schemes.

Somewhat Homomorphic Encryption (SHE): Enables a limited number of both addition and multiplication operations on encrypted data. The operations are constrained, making it suitable for

specific computations. Examples include the Gentry's first level of Fully Homomorphic Encryption (FHE) and the TFHE (Tripartite Fully Homomorphic Encryption).

Fully Homomorphic Encryption (FHE): Allows unlimited iterations of both addition and multiplication operations on encrypted data. This advanced encryption form allows arbitrary computations on encrypted data without decryption.

- Current advantages and disadvantages of HE?

Advantages:

- The egress cost associated with data leaving the cloud can we very significant, HE can play a vital role to perform analytic on the data without the egress cost. This is significant saving to any business. Furthermore, remember the Ingress data coming into the cloud is free. The Cloud service provider will charge for all the egress cost, considering generative AI application such as Falcon 40B or Llama2 which uses substantial corpus data can leverage HE to compute the token latency in AI deep learning without the egress cost.
- Homomorphic encryption helps minimize the cost associated with data breaches, once the data is stored encrypted using HE it will be required to decrypt it using decryption method which is revers of encryption method.

- Homomorphic encryption stands as a pivotal solution in cloud computing, allowing organizations to securely store encrypted data within a public cloud infrastructure. This setup enables leveraging the cloud provider's analytic services while keeping sensitive information entirely encrypted. It ensures data privacy and security, allowing computations and analyses to be performed on the encrypted data without the need for decryption. This capability enables organizations to harness the benefits of cloud-based analytics without compromising on data confidentiality, maintaining control and privacy over their valuable information
- Homomorphic encryption shields supply chains: Encrypting data for third parties ensures continuous protection. Even if breaches occur, data remains incomprehensible, securing the supply chain's integrity
- Homomorphic encryption empowers companies like Meta to conduct analytics on user data without accessing the original information. This innovation enables more private targeted advertising, allowing analysis while preserving user data confidentiality

Disadvantages:

- Computational Overhead:  Performing computations on encrypted data is significantly more computationally intensive than on plaintext data. This overhead can slow down operations and increase processing times. "Homomorphic encryption's significant drawback lies in its computational intensity, causing slower operations compared to plaintext. Encrypting, decrypting, and processing ciphertexts demands notably more resources and time. This limitation makes it unsuitable for real-time or high-throughput scenarios, potentially incurring substantial costs and latency.
- Key Management Security: Managing keys securely in homomorphic encryption systems is critical. Any compromise in key management could lead to a breach of the encrypted data.
- Computational Complexity: Implementing and working with homomorphic encryption requires specialized knowledge and expertise. Developing efficient algorithms and applications that use this encryption can be complex and challenging
- Performance Trade-offs: There's often a trade-off between security and performance. Increasing security measures in homomorphic encryption can lead to decreased performance or increased computational requirements.
- Resource Intensiveness: Deploying and maintaining systems that utilize homomorphic encryption can require significant computational resources, impacting scalability and cost-effectiveness.
- Usability and Interoperability: The usability and interoperability of homomorphic encryption face challenges due to the absence of common frameworks, libraries, and standards. This hinders ease of understanding, implementation, and compatibility among different schemes and cryptographic tools. Additionally, legal, ethical, and social concerns—like data ownership, consent, accountability, and trust—may arise, posing further obstacles to widespread adoption and implementation

- Use cases in financial services?
- How can it help increase collaboration to combat financial crime?

Ref:

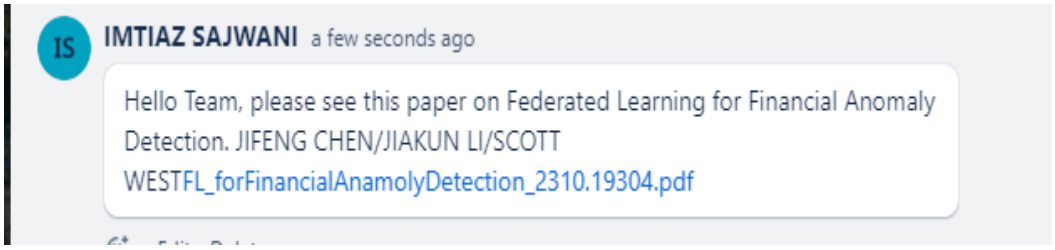

https://homomorphicencryption.org/introduction/

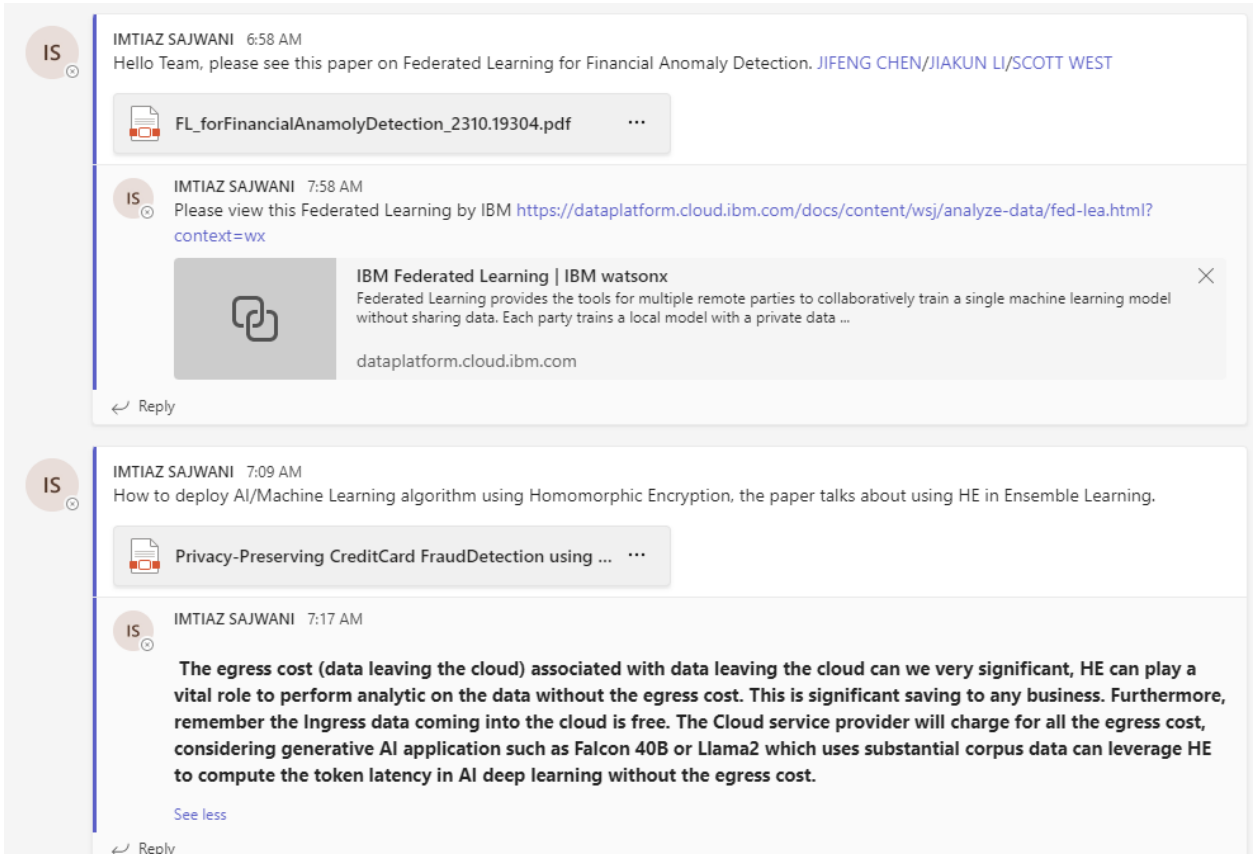
https://arxiv.org/pdf/2301.07041.pdf

https://www.future-of-computing.com/zama-shaping-the-future-of-homomorphic-encryption-for-machine-learning/ (AI/ML)

https://eprint.iacr.org/2022/915.pdf (opne source FHE library)

Bank of America:
https://www.retailbankerinternational.com/data-insights/bank-of-america-files-patent-for-homomorphic-encryption-based-testing-system-for-application-testing/?cf-view


IS
IMTIAZ SAJWANI a few seconds ago
Hello Team, please see this paper on Federated Learning for Financial Anomaly Detection. JIFENG CHEN/JIAKUN LI/SCOTT WESTFL_forFinancialAnamolyDetection_2310.19304.pdf


IS
IMTIAZ SAJWANI 6:58 AM
Hello Team, please see this paper on Federated Learning for Financial Anomaly Detection. JIFENG CHEN/JIAKUN LI/SCOTT WEST
FL_forFinancialAnamolyDetection_2310.19304.pdf
IS
IMTIAZ SAJWANI 7:58 AM
Please view this Federated Learning by IBM https://dataplatform.cloud.ibm.com/docs/content/wsj/analyze-data/fed-lea.html?context=wx
IBM Federated Learning | IBM watsonx
Federated Learning provides the tools for multiple remote parties to collaboratively train a single machine learning model without sharing data. Each party trains a local model with a private data ...
dataplatform.cloud.ibm.com
Reply
IS
IMTIAZ SAJWANI 7:09 AM
How to deploy AI/Machine Learning algorithm using Homomorphic Encryption, the paper talks about using HE in Ensemble Learning.
Privacy-Preserving CreditCard FraudDetection using ...
IS
IMTIAZ SAJWANI 7:17 AM
The egress cost (data leaving the cloud) associated with data leaving the cloud can we very significant, HE can play a vital role to perform analytic on the data without the egress cost. This is significant saving to any business. Furthermore, remember the Ingress data coming into the cloud is free. The Cloud service provider will charge for all the egress cost, considering generative AI application such as Falcon 40B or Llama2 which uses substantial corpus data can leverage HE to compute the token latency in AI deep learning without the egress cost.
See less
Reply