



Homomorphic Encryption: Computing on Secrets Without Breaking Them

Homomorphic Encryption, or FHE, lets computers do math on secret data without unlocking it first. This keeps information safe when it's being used in cloud and AI systems. There are some challenges with FHE, like it being really complicated, creating lots of extra data, and being hard to manage the keys. This is a major step forward for keeping big data and cloud computing safe.

By **Muhammad Asif** | Department of Computer Science | Published: October 2026

✉ muhammadasifdogar9@gmail.com | Position: Software Engineer

Introduction

The importance of encryption has really gone up with the rise of data and cloud computing. There is something called homomorphic encryption, which only lets you do some things with the data. Then there is homomorphic encryption, or FHE for short, which is really cool because it lets you do any kind of computation on encrypted data. Homomorphic encryption is a big deal because it lets you do almost any computation on the data without having to decrypt it first. But homomorphic encryption still has some problems. It is too complex, and it is really slow. Homomorphic encryption and FHE are really important for keeping data safe. The use of encryption will grow as more data is stored in the cloud.

“Fully Homomorphic Encryption is transforming cybersecurity by proving that data can stay encrypted even while being used--turning the biggest weakness of cloud computing into its strongest protection.”

Technology Overview

Homomorphic encryption is really a way to keep our data safe. It allows us to do lots of things with the encrypted data without actually decrypting it. This means our homomorphic encryption data stays totally private and secure even when we are using the encryption data. The old ways of encrypting data were not good enough because they only kept the data safe when it was not being used or when it was being sent somewhere. When we actually used the data, it was not safe. Homomorphic encryption fixes this problem. This is really useful when we need to keep our data safe, like when we're looking at the data or sending the data to someone we do not trust.

Homomorphic encryption makes it possible to use the encryption on our data without seeing the data, which is really important, in many different areas where we use the data, and homomorphic encryption. We can use the data with encryption in a way that is really useful for us, and that is what makes



homomorphic encryption so important for people who work with the data. Homomorphic encryption is a deal because it helps us to keep the data safe while we are using the data. Homomorphic encryption is very good for keeping data private and secure, especially when we are working with encryption, in situations where we need to keep the data private.

Key Findings & Impact

One big thing we found out is that Fully Homomorphic Encryption enables people who own data to send calculations to other parties they do not trust or to cloud environments without worrying about keeping their data secret. For example, Fully Homomorphic Encryption has been used with Extreme Gradient Boosting, which is also known as XGBoost, for classifying classes.

In this case, Fully Homomorphic Encryption makes it possible to train engineering features. We can make guesses with data, and that means the data stays secret the whole time we are using the machine to learn from it. Fully Homomorphic Encryption makes the machine take longer to do its calculations. I have a time making sure the answers are correct, and the machine is working properly. The thing is, we can actually use Homomorphic Encryption to keep our information private, and that is a big deal. This means we can take a step towards looking at secret information in a safe way. Fully Homomorphic Encryption is really good for keeping our information private. We can use Homomorphic Encryption to keep our private information safe, and that is what matters.

What's Next?

Homomorphic encryption is a change in the way we do cryptography. It lets us do things with encrypted data without having to decrypt it. This means we can keep our data secret all the time. We can do things with the data. It will still be encrypted. Homomorphic encryption is really good because it helps keep our data safe. We can use encryption to do lots of things with our data, and it will always be secret.

Current State and Applications

- **Database Privacy Protection**

Database Privacy Protection is really important these days. You see, with much data being collected, people want to make sure their information is safe. Database Privacy Protection is about keeping this data private. We have something called Homomorphic Encryption that helps with Database Privacy Protection. It lets us work with encrypted databases without showing any information. There are types of Homomorphic Encryption. Each type of Homomorphic Encryption has its good and bad points when it comes to how well it works and how fast it is.

- **Cloud Computing**

He makes sure that cloud computing is safe by letting people work with encrypted data, which reduces the risks that come with sending data for processing. Even though Homomorphic Encryption protects data when it is stored, moved, and used, Homomorphic Encryption methods have some limitations, such as the size of encrypted text, high costs of computing, and complicated ways of managing keys. Homomorphic Encryption schemes are still not perfect.



Challenges and Limitations

- **Encryption Volume and Complexity**

Homomorphic ciphertexts are usually a lot bigger than the texts, which makes them hard to store and send. The thing is, it is really tough to come up with plans that work well with sets of data or complicated math problems. Homomorphic ciphertexts are still a problem when we try to use them with datasets or complex computations because they are typically much larger than the original plaintexts.

- **Security and Key Management**

While HE gives privacy, we need to manage cryptographic keys safely. Also, it must resist side-channel and other new attacks. This is crucial for using it in applications. Managing keys and resisting attacks like side-channel ones are key.

About the Author

Muhammad Asif is a BS IT student and is interested in cryptography and advanced encryption to secure data. Learning Fully Homomorphic Encryption and privacy-preserving machine learning. They can be reached at muhammadasifdogar9@gmail.com

Keywords: Artificial Intelligence; Cybersecurity; Homomorphic encryption; Machine learning; Pakistan Tech

References

- [1] Chen, Q., Li, H., Ariffin, S., & Abdullah - <https://doi.org/10.18488/76.v11i3.3955>
- [2] Deviani - <https://doi.org/10.26740/jieet.v7n1.p49-58>
- [3] Chen, Q., Li, H., Ariffin, S., & Abdullah - <https://doi.org/10.18488/76.v11i3.3955>
- [4] Ma, Y., Zhao, J., Li, K., Cao, Y., Chen, H., & Zhang - <https://doi.org/10.4018/ijcini.287600>
- [5] Agarwal, P., & Shrivastava - <https://doi.org/10.17492/computology.v1i1.2104>
- [6] Chen, Q., Li, H., Ariffin, S., & Abdullah - <https://doi.org/10.18488/76.v11i3.3955>
- [7] Gilbert, C., & Gilbert - <https://doi.org/10.55248/gengpi.5.1124.3253>