Secure Cloud Data Processing with Fully Homomorphic Encryption

Amal Joy
Department of Computer Applications
Amal Jyothi College of Engineering Kanjirappally, India
amaljoy@mca.ajce.in

Sr. Mercy Joseph
Asst. professor Department of Computer Application
Amal Jyothi College of Engineering, Kottayam, India
elsinchakkalackal@amaljyothi.ac.in

***Abstract*: Fully homomorphic encryption (FHE) has been named the sacred goal of cryptography, a subtle objective that could tackle the IT world's concerns of safety and trust. Research in the space detonated after 2009 when Craig Gentry showed that FHE can be acknowledged on a fundamental level. Since that time impressive advancement has been made in seeing as more viable and more effective arrangements. While the research was immediately created, wording and ideas became assorted and befuddling with the goal that today, it might be difficult to determine the successes of diverse works. The ultimate goal of this paper is to answer three critical questions. What is FHE? How FHE ensures security? How FHE can be implemented on cloud? Just as reviewing the field, we explain distinctive phrasing being used and demonstrate associations between various FHE ideas.**

Keywords: *FHE, Homomorphic Encryption, Security, Cloud*

I. INTRODUCTION

Homomorphic encryption has been utilized for supporting basic conglomerations, numeric estimations on encrypted information just as for private data recovery. As of late, hypothetical forward leaps on homomorphic encryption came about in completely homomorphic encryption, which can register subjective capacities on encrypted information. As a result, homomorphic encryption is by and large accepted to be the Holy Grail for addressing information base questions on encrypted information. The interest in the security of computerized information and of calculations for dealing with more perplexing designs have expanded dramatically in the last decade. This goes in corresponds with the development in correspondence organizations and their gadgets and their expanding capacities. At a similar time, these gadgets and organizations are dependent upon an extraordinary assortment of assaults including control and annihilation of information and burglary of touchy data. For putting away and getting to information safely, current innovation gives a few strategies for ensuring protection, for example, information encryption and the use of alternative safe equipment. Be that as it may, the basic issue emerges when there is a prerequisite for processing (freely) with private information or to adjust the calculations so that they are executable while their security is guaranteed. Homomorphic cryptosystems importance comes into existence here, since these frameworks allow computations to be performed on encrypted data.

II. CLOUD COMPUTING

Cloud computing provides services over Internet such as servers, software, databases, storage, networking, and analytics in order to provide rapid development and more scalable resources. There are various issues identified with conveyed figuring traffic security and resource on the board. We can give security in the cloud in numerous ways like on information, organization and capacity. A homomorphic encryption strategy gives greater security on information since the supplier isn't included in key administration. I have used an intermediary re-encryption strategy that forestalls ciphertext from picked figure text assault. This framework is safer than the existing framework.

III. SECURITY ISSUES IN CLOUD COMPUTING

Cloud computing gives various benefits. But there are some security issues in cloud computing. Main security issues of cloud computing are as follows:

a) *Information Loss* –
Information Loss is one of the issues looked at in cloud computing. This is otherwise called data leakage. As we realize that our private information is in the possession of somebody else, and we don't have full command over our data. So, assuming the security of cloud administration is too broken by programmers then it could be conceivable that programmers will gain admittance to our confidential information or private data.

b) *Unauthorized access of Hackers & Insecure API's* –
As we probably are aware assuming we are discussing the cloud and its administrations it implies we are discussing the Internet. Likewise, we realize that the least demanding method for speaking with Cloud is utilizing API. So, it is vital to ensure the Interface's and APIs are utilized by an outside client. Yet in addition to cloud computing, scarcely any administrations are accessible in the public area. And it is the weak piece of Cloud Computing since it very well might be conceivable that these administrations are gotten to by some outsiders. So, it very well might be conceivable that with the assistance of these administrations' programmers can undoubtedly hack or destroy our data.

c) *Client Account Hijacking* –
If someway the Account of User or an Organization is commandeered by Hacker. Then, at that point, the programmer has full power to perform unauthorized activities.

d) *Changing Service Provider* –

DOI: 10.5281/zenodo.6365059
ISBN: 978-93-5607-317-3 @2022 MCA, Amal Jyothi College of Engineering Kanjirappally, Kottayam

Vendor lock-in is likewise a significant Security issue in Cloud Computing. Numerous associations will deal with various issues while moving to start with one seller then onto the next. For instance, An Organization needs to move from AWS Cloud to Google Cloud Services then they expert different issue resembles moving of all information, additionally both cloud administrations have various strategies and capacities, so they likewise deal with issues regards to that. Likewise, it very well might be conceivable that the charges of AWS are unique than Google Cloud, and so forth

e) Absence of Skill –

While working, moving o one more specialist co-op, need an additional a component, how to utilize an element, and so on are the principal issues caused in IT Company who doesn't have gifted Employee. So, it requires a talented individual to work with distributed computing.

f) Denial of Service (DoS) attack –

This is an attack that happens when the system gets requests from a huge traffic and exceeds the capacity of the system to respond to these requests. Most DoS attacks happen in enormous associations like the financial area, government area, and so on at the point when a DoS assault happens information is lost. So, to secure our confidential data requires a lot of cash just as an ideal opportunity to deal with it.

IV. HOMOMORPHIC ENCRYPTION

Homomorphic encryption is a cryptographic method where plain texts and ciphertexts are treated with a complex logarithmic function. Homomorphic Encryption permits the server to do the process on encrypted information without knowing the original data or plaintext.

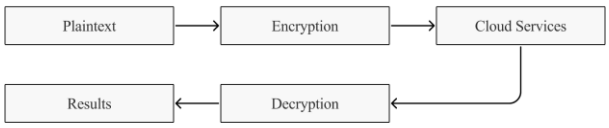

Fig 1. *Working of Homomorphic Encryption.*

Homomorphic encryption allows complex mathematical functions to be processed on encrypted information without utilizing the original information. For plaintexts P1 and P2 and comparing ciphertext C1 and C2, a Homomorphic encryption plot allows the calculation of P1 Θ P2 from C1 and C2 without utilizing X1 Θ X2. The cryptosystem is a multiplicative or additive substance Homomorphic relies on the activity Θ which can be multiplication or addition.

There are three types of Homomorphic Encryption. They are:

1) Fully Homomorphic Encryption (FHE), where both addition and multiplication can be performed on encrypted data.

2) Partially Homomorphic Encryption (PHE), only one operation can be performed on encrypted data by either addition or multiplication. (Pillars cryptosystem is used for addition operation only and RSA cryptosystem is used for performing multiplication operation on data).

3) Somewhat Homomorphic Encryption (SWHE), where the operation is performed on the limited number of multiplication or addition.

V. FULLY HOMOMORPHIC ENCRYPTION

FHE performs subjective calculations on encrypted data. Figuring on encrypted information implies that to acquire $f(m_1,\ldots,m_n)$ for certain data sources $m_1,\ldots,m_n$, it is feasible to rather register on encryptions of these data sources, $c_1,\ldots, c_n$, getting a result which is decrypted to $f(m_1,\ldots,m_n)$.

In some cryptosystems, the info messages (plaintexts) exist in some logarithmic construction, frequently a gathering or a ring. In such cases, the ciphertexts will regularly additionally exist in some connected design, which could be as old as the plaintexts. The capacity f in more established homomorphic encryption plans is ordinarily limited to be an arithmetical activity related to the design of the plaintexts. For example, think about ElGamal. On the off chance that the plaintext space is a gathering G, the ciphertext space is the item GxG, and f is limited to the gathering procedure on G. For sure most plans preceding 2009 fit such a construction. We can communicate the point of completely homomorphic encryption to be to stretch out the capacity f to be any capacity. This point can be accomplished in case the plan is homomorphic regarding a practically complete arrangement of activities and it is feasible to repeat tasks from that set.

While it is reliably a need that encryption plans are successful from a speculative perspective, specifically running in polynomial time in the security limit, common sense effectiveness was not the main goal in acquiring the principal FHE plans. One justification for the absence of effectiveness of these plans is that they utilize a plaintext space comprising of a solitary piece and are homomorphic concerning expansion and increase modulo 2. While any capacity of any intricacy can be developed from such essential tasks, that might require countless such activities.

VI. IMPLEMENTATION OF FHE

Craig Gentry of IBM has proposed the primary encryption framework "Fully Homomorphic" that assesses a self-assertive number of increments and duplications and consequently work out a capacity on encrypted information. Internal working of this adds one more layer of encryption each couple of steps and uses an encoded key to open the inward layer of scrambling. This decoding "revives" the information without uncovering it, permitting a limitless number of calculations on the equivalent.

DOI: 10.5281/zenodo.6365059
ISBN: 978-93-5607-317-3 @2022 MCA, Amal Jyothi College of Engineering Kanjirappally, Kottayam

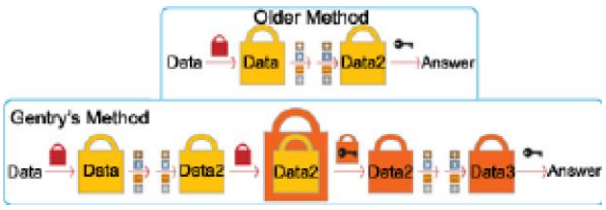

Fig 2. *Craig Gentry implementation of FHE.*

VII. FHE ON CLOUD

Fully Homomorphic encryption assumes a critical part in Security; all the more, for the most part, re-appropriating of the estimations on private information to the Cloud server is conceivable, maintaining the mystery key that can decrypt the aftereffect of computation. In our execution, we examine the presentation of existing homomorphic encryption cryptosystems, and are chipping away at a virtual stage as a Cloud server, a VPN network that connects the Cloud with the client, and afterwards reproducing various situations. For instance, a Database-Server speaking with a Client utilizing FHE Cryptosystem is as displayed in the figure underneath.

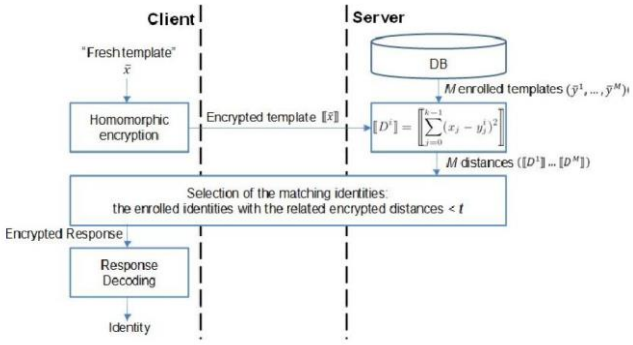

Fig 3. *A Database-Sever & Client implementing Homomorphic Encryption.*

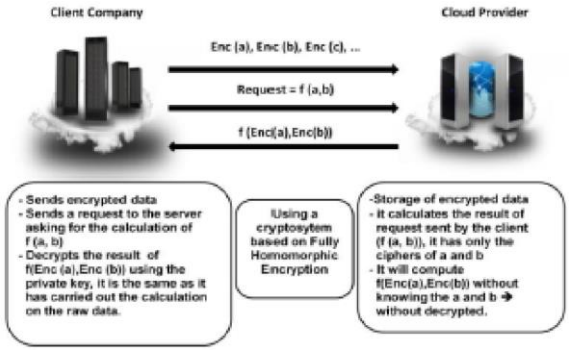

Fig 4. *FHE Implementation on Cloud Computing.*

VIII. CHALLENGES OF FHE ON CLOUD

The double layer of encryption causes the framework to run too leisurely for useful use. Researchers are working on further developing something basically the same for express applications, for example, scanning informational indexes for records reduce the time complexity. Additionally, to believe an exceptionally new encryption plot for privacy isn't possible and requires significant (~10 yrs.) of use openness. A group of researchers tried to join various plans to tackle these difficulties. The framework begins with homomorphic encryption, with an encrypted calculationss added in a distorted circuit which is itself secured by characteristic based encryption this guarantees the interaction stays encoded.

IX. CONCLUSIONS AND FUTURE DIRECTIONS

Even though there has been a great deal of late examination in the space of homomorphic cryptography, there are many excess open issues. As far as the speculative examination, set boundaries of choice for homomorphic encryption plans is right now a perplexing interaction, as each plan has explicitly chosen boundaries, which are all interlinked and these boundaries are generally chosen dependent on current conceivable grid-based assaults and their current cutoff points. An illustration of a shortcoming in this way to deal with boundary determination was taken advantage of in an assault by Lee, where the boundary choice in Gentry and Halevi's plan was not moderate enough to forestall a cross-section-based assault taking advantage of the meagre subset aggregate issue. More examination into boundary choice is expected to guarantee the most appropriate boundaries are picked to ensure both productivity and security.

As far as down to earth FHE executions, a further examination into appropriate equipment plans and improvements of existing plans could give an enormous accelerate, as demonstrated in. Advancements at an algorithmic level are needed; for instance, boundaries should be upgraded to augment the proficiency of executions. In addition, clumping procedures proposed for FHE plans could extraordinarily further develop the execution of any execution and ought to likewise be researched further. Enhancements at a building level are additionally required. One significant bottleneck in the execution of these plans is memory stockpiling. Enormous boundary sizes and exceptionally huge ciphertext sizes devour a lot of memory, which requires memory from the executives. Finally, improvements to target explicit gadgets, like utilizing the inserted multipliers on an FPGA, are required. For conceivable FPGA executions, the utilization of off-chip DDR3 memory is very likely required and accordingly information move could turn into an exhibition issue. In this manner, an examination into any enhancements lessening the memory prerequisites would be valuable for future executions. From all of this; we can conclude that the space of homomorphic cryptography stays forever with a lot of degrees for future exploration and improvement. Albeit further work on FHE plans is expected to improve and enhance execution, the new road of focusing on equipment or GPU development for smoothed out FHE models moreover looks outstandingly reassuring and brings the shot at nonstop executions of FHE a piece closer.

X. REFERENCES

1. Kristian Gjosteen, Christopher Carr, Frederik Armknecht, Angela Jaschke, Martin Strand, Christian A. Reuter, and Colin Boyd. “A Guide to Fully Homomorphic Encryption.” University of Mannheim https://eprint.iacr.org/2015/1192.pdf

DOI: 10.5281/zenodo.6365059
ISBN: 978-93-5607-317-3 @2022 MCA, Amal Jyothi College of Engineering Kanjirappally, Kottayam

2. Shashank Bajpai and Padmija Srivastava “A Fully Homomorphic Encryption Implementation on Cloud Computing.” International Journal of Information & Computation Technology. http://www.ripublication.com/irph/ijict_spl/ijictv4n8spl_05.pdf

3. Qussay Al-Jubouri, Hani Al-Zoubi, Waleed T. Al-Sit "Cloud Security based on the Homomorphic Encryption" (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 10, No. 8, 2019 https://thesai.org/Downloads/Volume10No8/Paper_38-Cloud_Security_based_on_the_Homomorphic_Encryption.pdf

4. Shai Halevi (IBM Research), “Homomorphic Encryption” April 2017 https://shaih.github.io/pubs/he-chapter.pdf

5. Leo de Castro "Practical Homomorphic Encryption Implementations & Applications" S.B., C.S M.I.T. 2018 https://dspace.mit.edu/bitstream/handle/1721.1/129883/1237358113-MIT.pdf?sequence=1&isAllowed=y

6. Geeks for Geeks “Security Issues in Cloud Computing” https://www.geeksforgeeks.org/security-issues-in-cloud-computing/

DOI: 10.5281/zenodo.6365059
ISBN: 978-93-5607-317-3 @2022 MCA, Amal Jyothi College of Engineering Kanjirappally, Kottayam