A picture containing font, text, logo, graphics

Description automatically generated

**Module Code: 7COM1039**

**Module Title: Advanced Computer Science Masters Project**

**Course Name: Computer Networking and Systems Security**

**Interim Project Report**

**on**

“Securing sensitive information on cloud computing with Homomorphic Encryption”

**Student Name: Praveen Kumar Killo**

**Student ID: 21020252**

Declaration

I, **Praveen Kumar Killo**, hereby state that this research project, titled “Securing Sensitive Information on Cloud Computing with Homomorphic Encryption” is my own work. All of the sources used to collect ideas and information in this research are duly referenced and acknowledged. This research does not contain any plagiarism.

All of the required steps have been followed throughout this research to ensure that it is conducted ethically. All the desired approvals to conduct this research were taken from relevant authorities. I know that if any of my above statements are wrong, then my research might be inhibited and I might be endangered to disciplinary actions.

# Abstract

In the current hyperconnected world, both cloud computing and the Internet of Things complement each other in many areas. Cloud computing represents a technology, which delivers resources as services. The users could access them anytime and anywhere through the Internet without the knowledge of authority, infrastructure, or experience. It becomes a vital part of the organizations in developing their networks. With the increased demand for cloud computing, the security of sensitive data become very important for both organizations and individuals. Cloud computing carries added risks of data security and privacy due to the deployment of the provided services to third-party platforms, which poses difficulties in facilitating data integrity, security, privacy, confidentiality, and authentication. Most users store their sensitive data on the cloud platforms in an encrypted format to reduce security concerns. However, the cloud-first needs to decrypt this data for performing any operation on the cloud server. It might cause the most critical issues related to the privacy and confidentiality of the stored data on the cloud. Thus, this research mainly aims to develop and propose a homomorphic encryption-based system to resolve the issues of data security and privacy by increasing the security of sensitive data stored in cloud computing. Homomorphic encryption provides the ability to the users to prosecute the computations on the encrypted data. We used fully homomorphic encryption, which is more efficient and secure to perform computations on third-party platforms and provides advantages of the properties of both multiplicate and additive homomorphism. Further, we used SEAL and PALISADE to realize a more flexible & robust framework to perform homomorphic encryption operations and realize significant performance enhancements.

Acknowledgment

I would like to express my sincere gratitude to all those who contributed and supported me in completing this research dissertation.

First, I am extremely obliged to my supervisor [Superviro’s name] for his invaluable expertise, insights, unwavering guidance & support throughout this research project. His motivation, expertise, and patience have largely shaped the quality of this research dissertation. Also, I want to extend my appreciation to members of the University for the valuable feedback, intellectual contribution, and constructive criticism during this research’s different stages. Their different point of view and experiences significantly enriched this research’s content. I would also like to acknowledge the provided assistance and support by the librarians and staff for their expertise in literature search, research methodology, and access to related resources that are precious in completing this work.

Finally, I would like to express my heartfelt gratitude to my family and friends for their unwavering understanding, encouragement, and support during this research. Their constant encouragement and belief in my abilities have largely helped me to overcome the arising challenges and successfully complete this research.

Table of Contents

[Abstract 3](#_Toc151283656)

[1 Introduction 6](#_Toc151283657)

[1.1 Research background 6](#_Toc151283658)

[1.2 Problem statement 6](#_Toc151283659)

[1.3 Aim and objectives 7](#_Toc151283660)

[1.3.1 Aim 7](#_Toc151283661)

[1.3.2 Objectives 7](#_Toc151283662)

[1.4 Research questions 7](#_Toc151283663)

[1.5 Project description 7](#_Toc151283664)

[1.6 Project Methodology 8](#_Toc151283665)

[1.7 Project Plan 8](#_Toc151283666)

[1.8 Risk assessment 8](#_Toc151283667)

[1.9 Project Overview 9](#_Toc151283668)

[Conclusion 10](#_Toc151283669)

[Introduction 11](#_Toc151283670)

[2 Literature review 11](#_Toc151283671)

[2.1 Cloud computing 11](#_Toc151283672)

[2.2 Security of cloud computing and homomorphic encryption 11](#_Toc151283673)

[2.3 Homomorphic encryption to secure Cloud database 12](#_Toc151283674)

[2.4 Survey on homomorphic encryption to secure data in cloud computing 13](#_Toc151283675)

[2.5 Challenges in using homomorphic encryption for securing cloud computing 14](#_Toc151283676)

[2.6 A secured cloud computing architecture using homomorphic encryption 14](#_Toc151283677)

[2.7 HE-based secured scheme for mobile multi-cloud computing 15](#_Toc151283678)

[2.8 Homomorphic encryption for preserving privacy in data mining on the cloud computing environment 16](#_Toc151283679)

[2.9 Homomorphic encryption to ensure confidentiality of the cloud data 16](#_Toc151283680)

[2.10 Fast Cloud-Paillier homomorphic mechanism to protect data confidentiality in cloud computing 17](#_Toc151283681)

[2.11 The homomorphic encryption algorithm to secure computations on the cloud 18](#_Toc151283682)

[2.12 Homomorphic encryption-based secure outsourced matrix multiplication scheme 19](#_Toc151283683)

[Conclusion 19](#_Toc151283684)

[Introduction 20](#_Toc151283685)

[3 Methodology 20](#_Toc151283686)

[3.1 Algorithm 20](#_Toc151283687)

[3.1.1 Homomorphic encryption 20](#_Toc151283688)

[3.2 Data collection 21](#_Toc151283689)

[3.3 System requirements 21](#_Toc151283690)

[3.3.1 Hardware requirements 21](#_Toc151283691)

[3.3.2 Software requirements 21](#_Toc151283692)

[Conclusion 23](#_Toc151283693)

[Introduction 24](#_Toc151283694)

[4 Project Development 24](#_Toc151283695)

[4.1 Project design 24](#_Toc151283696)

[4.2 Implementation 25](#_Toc151283697)

[4.3 Testing 26](#_Toc151283698)

[4.3.1 Black box testing 26](#_Toc151283699)

[Conclusion 26](#_Toc151283700)

[Introduction 27](#_Toc151283701)

[5 Project Evaluation 27](#_Toc151283702)

[5.1 Feasibility of the proposed system in the commercial and economic contexts 31](#_Toc151283703)

[Conclusion 32](#_Toc151283704)

[6 Conclusion 33](#_Toc151283705)

[6.1 Conclusion 33](#_Toc151283706)

[6.2 Encountered problems 33](#_Toc151283707)

[6.3 Recommendations 33](#_Toc151283708)

[References 35](#_Toc151283709)

Chapter 1: Introduction

# Introduction

This chapter is based on the introduction of this research dissertation, which is considered the most critical part of this research and provides a detailed overview of this research topic. In this chapter, we will discuss the research background, aim & objectives, problem statement, research questions, description of the project, research methodology, project plan, risk assessment, and project overview. By incorporating these necessary elements, we can successfully engage the readers in the introduction chapter while establishing the research's importance. Here we will also highlight this research contribution by demonstrating the added values of this research in the current body of research knowledge.

## Research background

The increased evolution of digital technologies makes public networks very popular among people, particularly the Internet. The unprecedented increment the Internet users raises various security concerns, especially for user privacy and confidentiality. The risk of user privacy violation intensifies with the increased use of smart devices, cloud services emergence, and mobile solutions. Cloud computing provides the benefits of centralized large space, computational power, and efficiency to enable clients/customers to outsource their complicated problems (Mishra, 2023). However, it suffers from a few challenges related to customer checkability, privacy, and confidentiality. Cloud computing has huge potential to realize computation power for aggregated management of adaptable resources. Thus, customers should encrypt their private data before outsourcing to protect it from unauthorized access. Clou servers face difficulties in performing computations of the user’s encrypted data. There is an extreme need to provide user security at various stages, including data, host, network, application, etc.

The homomorphic encryption (HE) methods enable in performing of various computational functionalities on the derivative data without the knowledge of the private key, the customers are the only owners of the private secret key. Whenever any computational function’s result is decrypted, it represents that the calculation process is accused of the calculation process on the unstructured and unanalyzed data. Gentry has launched an FHE (fully homomorphic encryption) scheme for allowing users to perform multiplicative and additive homomorphic operations on the encrypted data. It addresses an evolutionary innovation in the cryptography field. Still, it does not have real-time practical scenarios. Either additive or multiplicative homomorphic operation is enabled in the moderately homomorphic encryption scheme (Yakoubov et al., 2014). An algorithm can only be considered as wholly homomorphic encryption if the multiplicative and additive homomorphism properties are simultaneously satisfied. The fully homomorphic encryption can be utilized in storing data in AWS (Amazon Web Service)’s DynamoDB. The operational procedures of users have been carried out on the encrypted/unoriginal data within the public cloud. They could be just downloaded on the client’s machine whenever there is a requirement for results. In this particular case, the confidential data of users will not be stored in plain text form over the public cloud.

## Problem statement

The increased growth and use of cloud computing platforms revolutionized how organizations store and process data. However, the privacy and security of the stored confidential information on cloud storage remain a significant issue. The conventional encryption methods can protect the data but they hinder the ability to perform computations on the user’s encrypted data without decryption limits cloud computing’s practicality. The use of homomorphic encryption provides the most promising solution to enable computations on the encrypted data without any decryption while protecting user privacy and facilitating secure data analysis over cloud platforms. However, there are several obstacles in the practical deployment of the homomorphic mechanisms (Yang et al., 2023). This research addressed the problem present in the current state of securing users’ sensitive data on cloud computing using the homomorphic encryption method as it lacks an inclusive understanding of the underlying performance limitations, technical challenges, and practical considerations. There is a need to recognize and address these issues while ensuring efficient deployment of homomorphic encryption to secure confidential information stored on the cloud. Thus, this research intends to examine these performance limitations, technical challenges, and practical considerations to secure user’s confidential information on cloud computing using homomorphic encryption. Overall this research will offer vital research insights to address the effectiveness, practicality, and feasibility of using homomorphic encryption methods to secure sensitive data over the cloud to enhance the privacy and security of the cloud computing atmospheres.

## Aim and objectives

### Aim

The main aim of this research project is to examine and validate the efficacy of homomorphic encryption mechanisms to secure user’s confidential information on cloud computing platforms.

### Objectives

The main objectives of this research project are as follows:

* For evaluating the currently faced security issues in the cloud computing platforms with their impacts on the user’s sensitive data or information.
* For examining the methods and principles of homomorphic encryption and understanding its applicability in cloud computing scenarios (Medileh et al., 2023).
* For designing and implementing prototype systems that utilize homomorphic encryption to secure the cloud platform’s sensitive information.
* For assessing the security and performance implications of the proposed system and comparing it with the previous encryption methods.
* For providing potential guidelines and recommendations related to the adoption of homomorphic encryption methods in the cloud computing atmospheres.

## Research questions

1. How effective are the homomorphic encryption methods in securing user’s sensitive information on cloud computing platforms and compare it with the traditional encryption methods in terms of usability, security, and performance?
2. How can be homomorphic encryption methods efficiently integrated with cloud computing for the security user’s confidential data while maintaining scalability and performance?
3. What are the main issues with executing and adopting homomorphic encryption methods to secure confidential information in the cloud computing atmosphere and how these challenges can be addressed?
4. What are the potential trade-offs among performance, security, and computational complexity during the utilization of homomorphic encryption in cloud computing atmospheres?
5. How could be homomorphic encryption-based security solution optimized for diverse kinds of cloud services?

## Project description

This research project addressed the comprehensive adoption of cloud computing and transformed how companies store, access, and process user data. However, this tremendous shift also raised potential concerns related to user privacy and security. The traditional methods can protect data privacy but they deter computations on the encrypted data that restrict cloud services utility (R et al., 2023). This research focuses on utilizing homomorphic encryption as the most promising solution to address and resolve this issue. It will help in enabling the computations on the encrypted data with decryption and protecting data privacy to facilitate a secure data analysis over the cloud. This innovative technology has immense potential to revolutionize cloud security and enable novel stages of data security and privacy in cloud-centric applications.

## Project Methodology

This research uses a mixed research methodology by combining the vital research insights of both qualitative and quantitative research methods to provide a nuanced and inclusive understanding of this research topic to address both practical and technical aspects of homomorphic encryption-based security solutions (Munjal & Bhatia, 2022). Further, we used homomorphic encryption to perform computation on the encrypted data with decryption which makes it a suitable and efficient solution to protect sensitive data on cloud computing. Moreover, we collected secondary research data from various online sources using relevant key works. Then we used Python programming language with SEAL and PALISADE libraries. Also, we used Amazon Web Service and Heroku cloud computing services.

## Project Plan

This project section illustrates the details of the inclusive project plan by discussing the various project phases.

* **Requirement gathering**: In this phase of the project, we gather the project requirements, such as topic selection, research methodology, methods, techniques, and other research data.
* **Project planning**: In this phase, we create a project plan by addressing the project scope, breaking down this project into small tasks, estimating required resources, creating a project schedule, identifying & assessing risks, and developing a risk management plan.
* **Design**: In this project phase, we collect and analyze the design requirements and convert these requirements into design elements to create an efficient system design for this project.
* **Development**: This project phase is based on the project development using the chosen research methodology and other selected tools & techniques (Zhao E & Geng, 2019).
* **Testing**: In this phase, the entire project and developed system are tested to identify any error or bug and validate the system’s effectiveness against created test cases to ensure whether the system functions as expected or not.
* **Deployment**: In this phase, the developed system is deployed in real-time scenarios to check its effectiveness while gathering user feedback to identify the issues faced by users and collect their feedback to make any desired changes.

## Risk assessment

|  |  |  |  |
| --- | --- | --- | --- |
| Risks | Frequency | Severity | Mitigation steps |
| Technical challenges | High | High | To mitigate this project risk, we need to conduct extensive research and experiments to recognize and overcome the technical challenges. Also, we need to collaborate with the experts in cloud security and homomorphic encryption. |
| Performance overhead | Medium | Medium | To mitigate this risk, we should optimize the homomorphic algorithms and protocols along with using hardware acceleration methods (Medileh et al., 2023). |
| Security vulnerabilities | Medium | High | To mitigate this risk, we need to conduct a critical security analysis of cryptographic algorithms and homomorphic schemes and utilize the most efficient and comprehensively used homomorphic schemes. |
| Usability & practicality challenges | Medium | Medium | To mitigate this risk, We should conduct an inclusive user study and usability testing for developing user-friendly tools and interfaces. |
| Integration challenges | Medium | Medium | To mitigate this risk, design HE-based security solutions that have to be compatible with the existing APIs and cloud platforms. We should also collaborate with the cloud providers to integrate HE-based security solutions. |
| Privacy concerns | High | High | To mitigate this risk, we need to implement robust data security and privacy measures for protecting stored data over cloud platforms (Kangavalli & Vagdevi S, 2015). |
| Lack of standardization | Low | Medium | To mitigate this risk, we should participate in standardization efforts to promote interoperability and adopt HE-enabled security solutions. |
| Regulatory & compliance challenges | Low | Medium | To mitigate this risk, we need to stay informed of the applicable compliance requirements and regulations to develop HEcentrci security solutions that must comply with all the applicable standards and laws. |

## Project Overview

This research project mainly focuses on examining and validating the efficacy of homomorphic encryption mechanisms to secure user’s confidential information on cloud computing platforms. We will divide this research project into six main chapters to provide a better and more systematic understanding to the reader about this research.

**Chapter 1: Introduction**

This is the first chapter of this research in which we introduce this research project in detail. This chapter introduces this research based on investigating the effectiveness of the homomorphic encryption methods for the security of the sensitive information of users stored on overall cloud platforms (Biksham & Vasumathi, 2016). In this chapter, we briefly discuss the research background, problem statement, aim & objectives, research questions, research description, research methodology, project plan, risk assessment, and project overview to provide a better understanding of this project.

**Chapter 2: Literature Review**

This second chapter of this research project. In this chapter, we will search and select the most relevant research papers and journal articles using relevant keywords to conduct a detailed literature review to gain an improved understanding of this research. This chapter will also help to analyze other author’s works in the same research field and domain and what research theories and concepts they used to select the most efficient research method and techniques for this research.

**Chapter 3: Project Methodology**

This second chapter of this research project. In this chapter, we will discuss the research methodology along with other algorithms and techniques that are used in this research. Here we will also discuss the data collection methods and system requirements, including used programming language, libraries, and frameworks.

**Chapter 4: Project Development**

This second chapter of this research project. In this chapter, we will design and test the proposed system to attain the best research results (Das, 2018). Here will also discuss the implementation flow of the proposed system along with the used testing methods to test the effectiveness of this system.

**Chapter 5: Project Evaluation**

This second chapter of this research project. In this chapter, we will provide a detailed evaluation of this research by evaluating the efficacy of the proposed HE-based security solution along with providing evidence of the performed practical work to develop the proposed solution. Here the economic and commercial feasibility of the proposed solution will also be deliberated.

**Chapter 6: Conclusion**

This second chapter of this research project. In this chapter, we concluded this research project by summarizing all the done work and main research findings. Here will will also discuss the encountered problems during this research and provide recommendations to further proceed this research in the future.

# Conclusion

In this chapter, we critically introduced this research project, including the research background, problem statement, aim & objectives, research questions, research description, research methodology, project plan, risk assessment, and project overview to provide a better understanding of this project. We also provided an overview of this project structure by briefly discussing the six chapters, namely the introduction, literature review, project methodology, project development, project evaluation, and conclusion.

Chapter 2: Literature Review

# Introduction

This chapter is based on the discussion of existing and relevant research papers. Here we will conduct a detailed review of the previously published research works that are relevant to this research topic and research field. We will review all the selected research papers based on the research context, addressed research problem, selected methodology, proposed solution, and limitations to provide a better understanding of this research topic. This chapter also helps in analyzing the used research theories and concepts by other authors in their respective research to choose the most efficient research methodology and techniques to conduct this research.

# Literature review

## Cloud computing

Cloud computing is a novel paradigm that revolutionized how organizations store, access, and process data. It focuses on delivering various cloud computing services, such as storage, servers, networking, databases, analytics, software, and intelligence over the internet. To use cloud computing services, one only needs to connect their physical devices or computers to cloud computing services, such as AWS (AmazonWeb Service). One can access technological services, like storage, databases, and computing power from the cloud provider instead of buying, possessing, and maintaining physical data servers and centers (Bhowmik, 2017). Cloud computing can be used in three different forms, namely IaaS (Infrastructure as a Service), SaaS (Software as a Service), and PaaS (Platform as a Service). The main benefits of cloud computing include scalability, accessibility, cost-savings, agility, and reliability. Thus, cloud computing refers to a powerful and robust tool that can be helpful for organizations of all sizes in saving money, becoming more agile, and improving their IT performance.

## Security of cloud computing and homomorphic encryption

Kangavalli & Vagdevi S, (2015) Cloud computing represents resource mass that could be accessed by paying desired money and it decreases the burden on software, infrastructure, hardware, and other organizational resources. More specifically, cloud computing refers to a technology that has gained huge popularity. Data security is one of the greatest issues in the implementation of cloud computing. For ensuring an efficient level of data security, there a several conventional encryption algorithms like DES and play fair cipher, but these methods or algorithms can only be utilized in encrypting plain texts into cipher texts during communication. To process it over cloud platforms, it needs to convert cipher texts into plain texts which makes it an easy target for hackers. One of the popular and robust algorithms, namely homomorphic encryption can be used for overcoming this issue. This algorithm can efficiently help ensure secure data processing and transmission over cloud platforms without any compromise to security and privacy. This research will discuss the concept and consequence of homomorphic encryption using relevant examples. Further, an RSA algorithm is used with partial homomorphic encryption to demonstrate its practical utilization. Moreover, the performance measures of the RSA partial homomorphic encryption algorithm are compared with the Paillier algorithm in terms of decryption and encryption time.

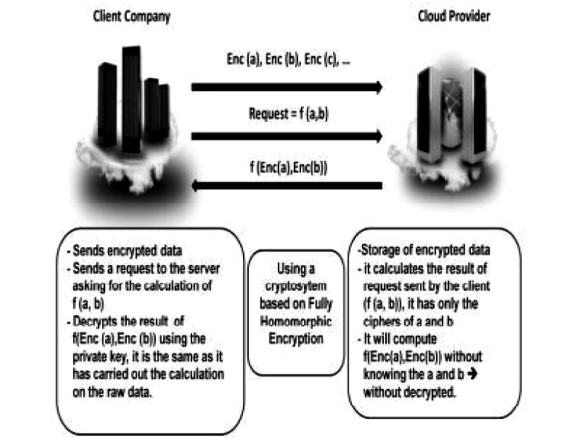


Figure : Homomorphic encryption on cloud

## Homomorphic encryption to secure Cloud database

According to Marwan et al., (2016), over the past decade, there a noteworthy increment in the demand for ICT (information & communication technology). Generally, it is helpful for organizations to improve the quality of the provided services. Also, it seeks to automate general administration tasks. Moreover, it facilitates efficient collaboration among diverse organizations. That is why most organizations need to purchase the required software and hardware for running their businesses. Cloud computing represents a model that provides on-demand computational resources to the customers as services. This concept charges users following the pay-per-use business model. Thus, it significantly decreases the associated operating costs with the local data center’s maintenance. Currently, data is offered as a service for fulfilling the demands of clients. The organization depends on the remote database for better managing its confidential data. Organizations showing huge interest in adopting this novel technology due to its potential benefits. Despite its challenges, it also poses several challenges, such as legal, technical, and managerial challenges. However, several types of solutions were presented for securing client data but only a few of those solutions are interested in exploiting databases. Thus, this research proposes a novel method based on a homomorphic mechanism for securing this novel paradigm (Marwan et al., 2016). This approach guarantees to protection of data confidentiality while enabling users to perform arithmetic operations on the encrypted data.

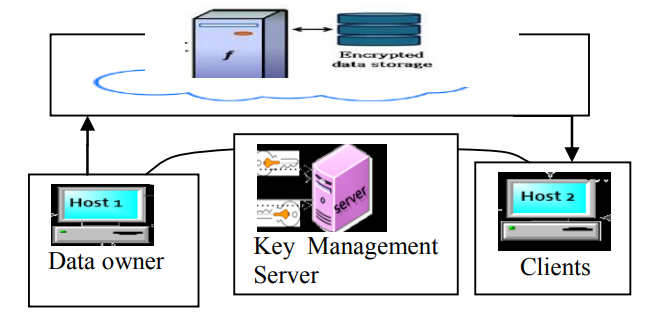


Figure : Fundamentals of the proposed method

## Survey on homomorphic encryption to secure data in cloud computing

Biksham & Vasumathi, (2017) state that cloud computing represents a growing computational model, where data and other services are associated with the ascendable data centers over the cloud and could be attained through the Internet. Cloud computing also poses several risks due to the deployment of the provided services to third-party platforms, while facing difficulty in enabling data privacy, security, factor, integrity, confidentiality, and authentication. So most users choose to store their private data over cloud platforms in an encrypted format to reduce data security and privacy concerns. However, for performing any type of operation on the data, cloud computing has to first decrypt user data. It may pose the most challenging issues, such as data privacy and confidentiality in the cloud. Thus, this research paper introduces a state-of-the-art HE domain for resolving the issues of data privacy and confidentiality in the cloud. Homomorphic encryption represents a type of encryption method that provides the ability to the user to prosecute the computations on the cipher texts. It produces encrypted results when decrypted and it demonstrates similarity in the prosecuted operation’s results. Homomorphic encryption has two types, namely FHE (Fully Homomorphic Encryption) and PHE (Partial Homomorphic Encryption). For third-party computations, FHE is more efficient and secure because it has advantages of both properties, multiplicative and additive homomorphism (Biksham & Vasumathi, 2017).

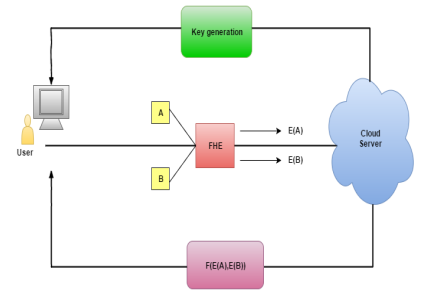


Figure : Homomorphism encryption in cloud computing

## Challenges in using homomorphic encryption for securing cloud computing

According to the National Insititute of Standards & Technology (NIST), cloud computing represents a model to enable convenient, ubiquitous, and on-demanding network access to configured computing resources that are frequently provisioned with the least management efforts. With the increased emergence of cloud computing, information security has become a major concern. The security of these systems become a critical issue for cloud service providers, computer scientists, and companies that want to utilize cloud computing services (El Makkaoui et al., 2015). Cloud computing facilitates efficient concepts to ensure an efficient availability of hardware, network security, control & access strategies, and data storage for these services. All of these elements play a vital role in preserving the security of data while ensuring the availability of these associated services with cloud computing for better satisfying the customers and building trust. The misuse of sensitive data is a major obstacle in the adoption of cloud computing services. To overcome this obstacle, we need to use methods that could perform operations on the encrypted data with the knowledge of secret ley, which appears to be an efficient way to strengthen information confidentiality. In this research, we investigate the associated challenges with the use of homomorphic encryption for enabling cloud suppliers to perform operations on the encrypted data while providing similar-level results.

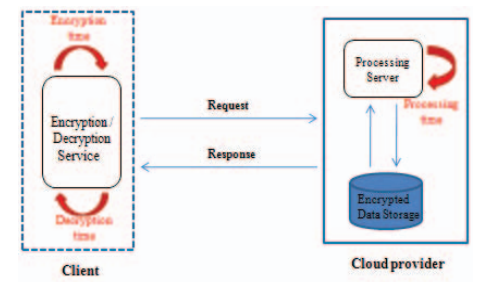


Figure : Cloud provider system using Homomorphic Encryption

## A secured cloud computing architecture using homomorphic encryption

According to Benzekki et al., (2016), the cryptosystems supply mechanisms for ensuring an effective level of data integrity and confidentiality. A strong and permanent encryption of data helps to ensure complete control over data while removing data security and privacy concerns. When the encryption algorithms do not allow random computations on the encrypted data, then the encrypted data should be decrypted before computation and this decrypted data should no longer remain under control., cloud computing is a novel paradigm but it is infeasible for several businesses that need to download confidential data from the cloud to a trusted computer to perform desired operations and send the obtained encrypted results back to the cloud platform. Thus, the homomorphic encryption algorithm is used to ensure an adequate level of data privacy and security during entire data communication, storage, and processing with the traditional cryptography methods with advanced cloud computing capabilities over the encrypted data. Homomorphism represents a specific property using which one algebraic system’s problem is transformed into another algebraic function’s problem to resolve it and later effectively translate this solution. Thus, homomorphism helps in security computation delegation to third-party platforms. However, many traditional encryption methods are used in respective applications that pose either additive or multiplicative homomorphic properties. In 2009, Gentry launched an FHE (Fully Homomorphic Encryption) mechanism for performing random computations on the encrypted data (Benzekki et al., 2016). This research paper proposed a multi-cloud architecture based on N dispersed for data repartition and realizing an FHE.

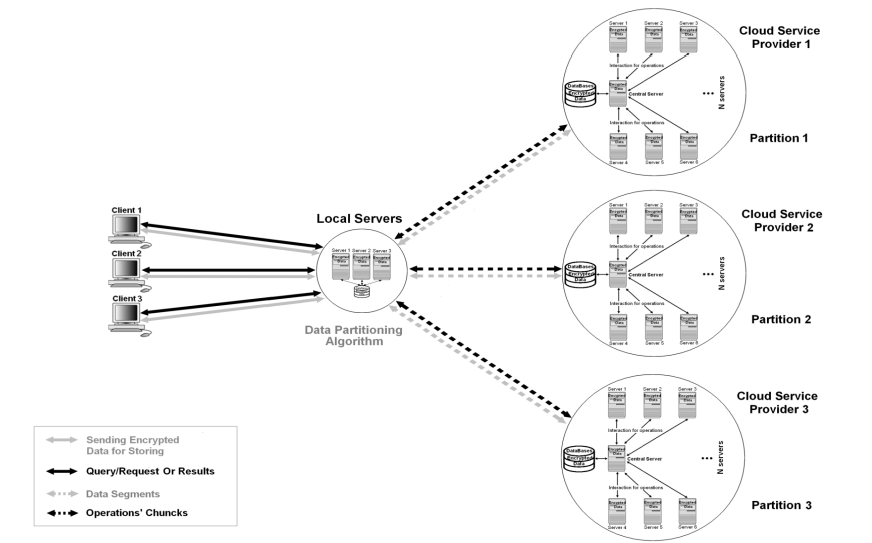


Figure : The proposed architecture for securing data using a homomorphic encryption mechanism

## HE-based secured scheme for mobile multi-cloud computing

Zkik et al., (2016) state that in the past two decades, the utilization of mobile technologies significantly grew phenomenally. According to a survey by Ericsson, mobile phone traffic increased by 55% from 2013 to 2014 and it is expected that about 90% of the population across the world will have smartphones by 2020. The emergence of cloud computing increases the use of mobile-based services due to its ability to enable users to easily access and utilize diverse applications and platforms while storing sensitive data on the remote server so that they can be accessed anywhere and anytime. Mobile technologies become gradually important in human life because they provide huge possibilities for users. Mobile cloud computing is an innovative paradigm that enables users to wholly utilize mobile technologies beyond the calculation limit of resources. This exponential utilization of cloud services via mobile devices enabled users to download, share, and retrieve their private data anytime and anywhere. This increased mobile traffic and circulation of users’ private data via the Internet caused several security issues and flashed the attacker’s interest. Thus, this research study proposes a strong and lightweight authentication scheme for mobile users to remotely access multi-cloud servers (Zkik et al., 2016). It also offers a deployment and simulation of the selected mechanism for demonstrating its reliability and robustness along with providing proof of concept of integrity, authentication, and confidentiality.

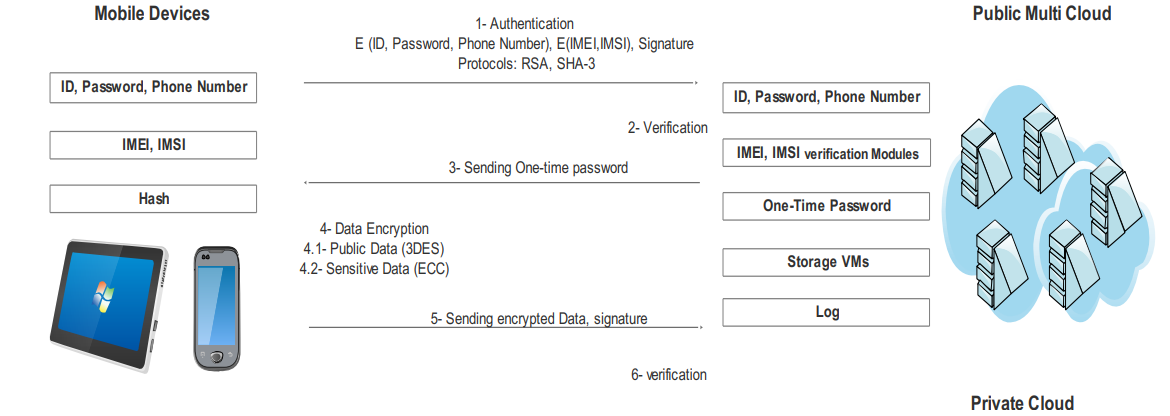


Figure : Secure communication between multi-cloud servers and access plan

## Homomorphic encryption for preserving privacy in data mining on the cloud computing environment

According to Hammami et al., (2017), the development of broadband networks and standardization of the Internet during the last decade, the computer world promoted a novel paradigm, namely cloud computing. Cloud computing represents an IT (information technology) infrastructure where software and data are stored and administered in a remote data center and can be accessed as a specific service via the Internet. The data centers present in these fields are noisy, large, and complex. However, privacy preservation is a major challenge in the data mining. It is necessary to protect the data confidentiality through the extraction of recurrent closed patterns. In actuality, no sites are capable of learning the transaction’s contents at other sites. This research work mainly focused to address and resolve this problem. Thus, this research recommends an efficient approach for combining the extraction of the frequent and closed data patterns in a dispersed environment like a cloud. This research also aims to maintain these sites’ privacy throughout the data mining task within the homomorphic encryption-base cloud atmosphere. The conducted performance analysis and obtained simulation results demonstrate that the proposed scheme needs minimal computation and communication overheads (Hammami et al., 2017). Also, this approach can efficiently check data integrity, preserve data privacy, and ensure higher efficiency in data transmission.

## Homomorphic encryption to ensure confidentiality of the cloud data

According to Suveetha & Manju, (2016), there is a growing need for resources to harshly process and uphold massive amounts of data in different fields. However one faces difficulties in owing and maintaining these resources. Cloud computing is an effective and increasingly emerging technology that offers these resources to user’s demands with the least effort. However, there are some issues with cloud computing related to performance, privacy, security, availability, reliability, portability, scalability, scheduling, elasticity, virtualization, interoperability, bandwidth cost, resource management, and energy consumption. Thus, the main aim of this research study is to perform homomorphic encryption on the encrypted data to provide data privacy and confidentiality to cloud computing users. A banking application is chosen in this research, which includes bank customer’s sensitive data. This research used the Paillier method’s multiplicative properties of the homomorphic encryption for calculating the overall interest in the encrypted banking dataset. This research attained an adequate level of user data confidentiality using the Pailier homomorphic encryption algorithm. The cloud services providers can implement computations on stored ciphered user data on the cloud data center without the knowledge of a secret key. The Paillier and RSA algorithms can be together utilized for evaluating performance considering both decryption and encryption time (Suveetha & Manju, 2016). Moreover, this research addressed that further enhancements could be made to optimize this proposed algorithm, which replicates changes in the decryption and encryption time based on selected file size.

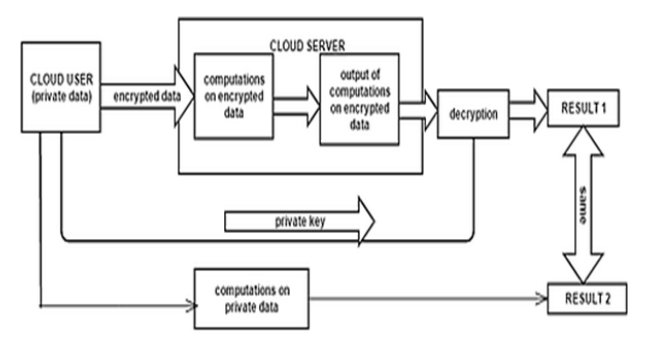


Figure : Homomorphic encryption to ensure cloud data confidentiality

## Fast Cloud-Paillier homomorphic mechanism to protect data confidentiality in cloud computing

El Makkaoui et al., (2019) state that currently cloud computing is increasingly used for obtaining on-demand computing and storage services by customers along with gaining access to their stored data anywhere and anytime with higher elasticity, availability, and lower cost through the Internet without any complex management of data storage. Any type of data could be processed using homomorphic encryption in its encrypted format in cloud computing. This property of homomorphic encryption is considered the most efficient and useful solution for overcoming the concerns that limit the comprehensive adoption of cloud computing services. Cloud computing environments are largely threatened by insider and outsider security attacks and cloud users often access cloud computing services using resource-restricted devices, the homomorphic mechanisms should be endorsed in terms of the security levels. This research boosted the performance of the main Paillier scheme by presenting a novel variant of this scheme, namely Cloud-Paillier. This proposed method addresses a major exception of the original Paillier scheme by supporting a preservative homomorphism over integers. Also, to realize faster decryption, we present two faster variants of the proposed Cloud-Paillier method. These proposed two variants utilize moduli shaped of k>=2 different primes. The first proposed variant uses the Chinese remainder theory for decryption, whereas the second variant somewhat adapts from the encryption algorithm of Cloud Paillier and decrypts similarly to Cloud-Paillier (El Makkaoui et al., 2019). The simulation and theoretical results demonstrate that the recommended variants provide higher speed during decryption than Cloud-Paillier along with preserving an effective level of security.

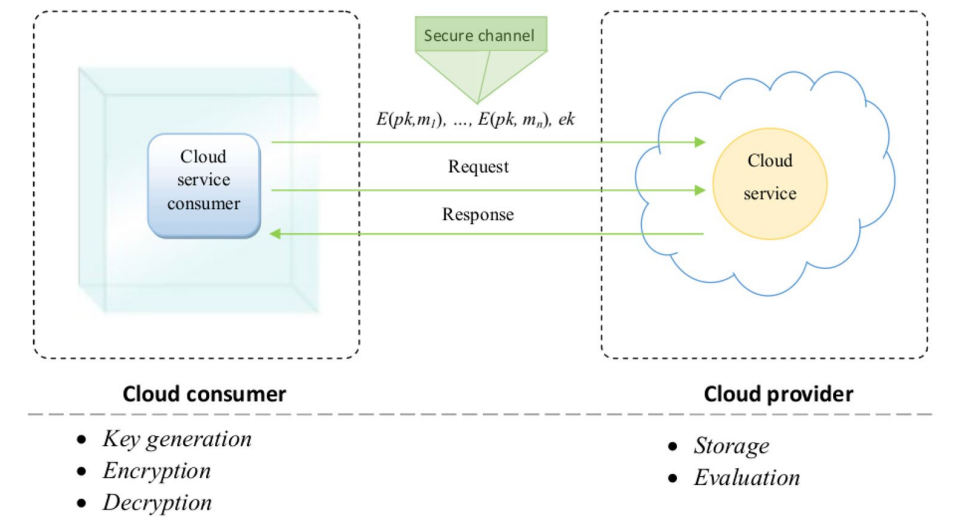


Figure : The proposed Cloud-Paillier homomorphic algorithm

## The homomorphic encryption algorithm to secure computations on the cloud

Alkharji & Liu, (2016) address cloud computing is emerging at a great pace but this shift also poses several security challenges. Organizations increasingly adopt cloud technologies for faster and more efficient computing but data security remains their major concern. The sensitive data is highly prone to data leakage due to the modern trends of outsourcing computations to third-party platforms. Also, these data breaches remove the benefits of adopting cloud computing technology. However, many cryptographic algorithms are used for providing efficient protection of sensitive data throughout the data storage and communication process, but these algorithms cannot be applicable in practical scenarios. They need to make data visible to the cloud providers for doing thing and the private key also need to be transmitted to the server for performing the desired operations. In the past few years, privacy homomorphism has been utilized to solve this problem. The homomorphic encryption algorithm enables us to execute the arithmetical operations directly on the ciphertext while keeping a secret key for decrypting the results. Moreover, it offers exactly the same results for preserving data privacy while performing computations on plain text. There are several fully homomorphic encryption methods are implemented for evaluating a random n. of multiplications and additions but the researchers could not design a robust and secure mechanism. Thus, this research study provides a detailed survey of the homomorphic encryption algorithm using public key algorithms, like Paillier, RSAn, and EI-Gamal algorithms (Alkharji & Liu, 2016). Then the FHE algorithms can be used to secure cloud computations. This research can help in guiding the best properties and principles of FHE to realize potential advancements in the area of FHE.

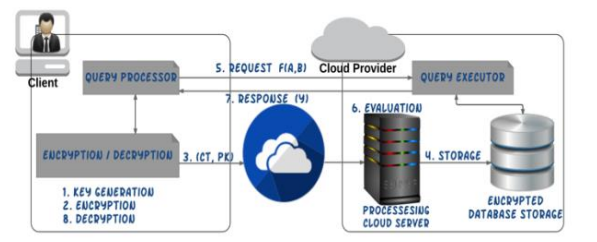


Figure : Homomorphic encryption algorithm on the cloud platforms

## Homomorphic encryption-based secure outsourced matrix multiplication scheme

Babenko et al., (2023) state that third-party platform-based services provide the most efficient and convenient way of building, complementing, or extending their cloud infrastructures. They offer unlimited data storage, convenient data access, and processing capabilities. However, securely storing and processing confidential data needs the selection of an efficient third-party service provider with a higher level of data protection. The homomorphic encryption algorithms represent the most efficient and promising solution to handle users’ sensitive data in semi-trusted third-party atmospheres because they allow secured processing of the user’s encrypted data. However, the use of sophisticated methods like image processing, statistics, and machine learning remains a major challenge. Some encrypted operation’s computational complexity may significantly raise the processing time. In this research, we mainly focus on analyzing the two different state-of-art homomorphic encryption matrix multiplication methods with the best space and time complexities. We demonstrate how the performance of these methods depends on the execution context and used libraries, considering the CKKS (Cheon-Kim-Kim-Song) HE mechanism with the fixed numbers based on the Microsoft PALISADE and SEAL libraries. This research will also demonstrate that the Linux operation system is the best option for the PALISADE library and the Windows operation system is the best option for the SEAL library. Generally, PALISADE-Linux outperforms the SEAL-Linux, SEAL-Windows, and PALISADE-Windows by 1.59, 1.67, and 1.28 times, respectively for diverse matrix sizes (Babenko et al., 2023). Moreover, this research services higher-precision extrapolation formulas for estimating the required processing time for larger matrix’s HE multiplication.

# Conclusion

In this chapter, we conducted a detailed literature review of the existing and relevant research papers to obtain a better understanding of this research. It sheds light on different aspects of cloud computing, such as definition, significance, benefits, and challenges. It also addressed the significance of cloud computing on individuals and organizations to revolutionize how computing resources are used, accessed, and managed. This review highlighted the main benefits of cloud computing, such as flexibility, scalability, accessibility, and reliability. Moreover, the collaborative nature of cloud computing facilitated enhanced collaboration among the dispersed teams.

Chapter 3: Methodology

# Introduction

The methodology chapter is another critical part of this research project, as it outlines the used methods and techniques for collecting and analyzing the research data. It also provides a detailed discussion and explanation of the used research design, approach, and statistical methods for addressing the research questions. This chapter mainly focuses on providing a clear discussion of how this research will be conducted while establishing the validity and credibility of the research findings. Here we will also discuss the data collection and data analysis methods. A detailed discussion of the system requirements, such as programming language, libraries, and frameworks will also be provided. Overall this chapter will serve as the roadmap to conducting this research while offering an inclusive overall of the used methods in this research for improving the trustworthiness of the obtained research findings.

# Methodology

In this research, we used mixed research methodology to combine vital research insights from both qualitative and quantitative research methodologies. This research methodology is well-suited for conducting this research to secure users’ sensitive data using homomorphic encryption on cloud computing. This research methodology provides a nuanced and comprehensive understanding of this research study while addressing both practical and technical aspects of the homomorphic encryption-based security solution (Tebaa et al., 2018). The **qualitative** **research** methodology is used for exploring the attitudes, perceptions, or experiences of individuals and organizations to secure sensitive data using homomorphic encryption on cloud computing. It includes conducting focus groups or interviews with the cloud service providers or users. The collected qualitative data provides vital insights related to the benefits, challenges, and limitations of homomorphic encryption to secure sensitive data on cloud computing. On the other hand, **quantitative research** methodology is used for gathering numerical data that could be statistically analyzed. For instance, a survey could be conducted to gather quantitative data to identify the organizations that currently use homomorphic encryption to secure sensitive data on cloud computing. The quantitative data can help in identifying the patterns, trends, and relationships between different variables to provide a generalized understanding of this research topic.

## Algorithm

### Homomorphic encryption

Homomorphic encryption is a robust cryptographic algorithm that allows computations to be performed on the encrypted data with decryption. This exclusive property makes homomorphic encryption one of the most promising solutions to secure the user’s sensitive data on cloud computing platforms. In the existing scenario of cloud computing, the sensitive data is stored and managed on the cloud servers, which poses concerns related to data security and privacy. The standard encryption methods need to be decrypted before performing computations (Thabit et al., 2022). The homomorphic encryption facilitates computations to be directly performed on the encrypted data. It means that sensitive data remains encrypted while processing to protect it from any kind of unauthorized access and ensure that one can only perform authorized operations on this data. Some of the main benefits of homomorphic encryption include secure computations, data privacy, cloud security, and privacy-protecting applications. It can be used to encrypt the stored data in the cloud, perform secure data analytics on the encrypted data, and securely outsource the computations to the cloud service providers.

## Data collection

In this research, we have gathered and used secondary research data to support the facilitation of this research paper. The required research data is collected from published sources (i.e. industry reports, academic journals, research papers, books, government reports, newspapers, etc.) unpublished sources (organizational documents, theses, dissertations, company reports, conference proceedings, etc.), online sources (social media, database, websites, etc.), and other sources (case studies, experts, historical records, etc.). Here I choose a few keywords to select the most relevant and efficient information. Also, I follow some considerations, such as the credibility of the sources, data currency, relevancy of the information, and ethical considerations throughout this entire data collection process.

## System requirements

The system requirements for this research project include both hardware and software requirements.

### Hardware requirements

* **Computer with a powerful processor**: The used homomorphic mechanism in this research project is computationally concentrated, so we need a computer with a robust processor, such as AMD Ryzen 7 or Intel Core i7.
* **Larger RAM**: The homomorphic scheme also needs a noteworthy amount of RAM, so we preferred to use 16 GM RAM or recommend using at least 8 GB RAM for performing such complicated project tasks.
* **Larger storage capacity**: The homomorphic mechanisms could produce a larger amount of the encrypted data, so we recommend using a large storage capacity, like 1 TB or more.

### Software requirements

#### Programming language

##### Python

Python represents an efficient and versatile programming language that could be largely used for developing homomorphic encryption algorithms to secure sensitive data on cloud computing. More specifically, it is a general-purpose programming language mainly used to create different types of programs and applications that are specialized for a specific problem. It provides several types of libraries and frameworks for realizing different types of homomorphic encryption functionalities and mechanisms to enable the researchers to select the most suitable and effective solution as per the requirements. The versatility and simplicity of this programming language make it a more accessible language to implement and experiment with various homomorphic encryption algorithms. Homomorphic encryption represents a complicated and intensive research field, so it needs a consideration of the efficiency and performance aspects of the selected frameworks or libraries, especially when handling real-time applications or large-scale computations on cloud computing platforms. Thus, Python is a great choice of programming language for this research project due to its higher flexibility, ease of use, and widespread library support.

#### Encryption libraries

This research described the most effective schemes for developing homomorphic encryption algorithms using the necessary encryption libraries. The encryption algorithms represent mathematic methods that are utilized for transforming the data into an unreadable format to protect it from any type of disclosure or unauthorized access. These encryption algorithms could be applied to any complicated mathematical operation to render unintelligible without knowing the decryption key. In this research, we used SEAL and PALISADE encryption libraries. These are open-source powerful homomorphic libraries that could be used for developing secure and efficient solutions to safeguard user’s sensitive data on cloud computing platforms. Both of these libraries provide a varied range of functionalities and features for enabling developers and researchers to better implement HE-base security solutions for the different cloud-centric applications. SEAL

SEAL stands for Selective Encryption for Arithmetic Libraries. It is a C++ library, that offers a powerful and flexible framework to better perform homomorphic encryption operations. It provides several features, including flexible parameter selection, effective homomorphic operations, compatibility with the cloud platforms, and support for several HE mechanisms. SEAL library can support different HE mechanisms, such as BGV, CKKS, and BFV to enable the developers to select the most effective scheme to meet their specific requirements (Gnana Sophia et al., 2022). It offers an optimized execution of the homomorphic multiplication, addition, and other arithmetic operations to facilitate an efficient computation of the encrypted data. It also enables the developers to better customize the HE parameters, such as noise budget and security level for balancing the performance and security requirements. The SEAL library could be easily integrated with the cloud computing platforms to efficiently facilitate HE-enabled security solutions for being deployed in cloud computing.

##### PALISADE

PALISADE stands for Privacy-Preserning advanced Lattice-based Cryptography for Arithmetic Libraries. It is another C++ library that offers higher-performance frameworks for different HE operations. It provides several advantages, such as supporting large-scale computations, higher-performing homomorphic operations, machine learning compatibility, and integration with cloud platforms (Wanjale et al., 2021). This library provides extremely optimized deployment of the homomorphic operations to attain significant enhancements in the performance overall other homomorphic encryption libraries. It is designed in a manner to efficiently support large-scale HE computations to enable it to better handle the complicated computations on the encrypted data. It efficiently supports the privacy-protecting machine learning algorithms to enable secure inference and training on the encrypted data. Moreover, it could be integrated with various cloud computing platforms to enable HE-enabled security solutions to be deployed in cloud computing.

#### Cloud computing services

Cloud computing represents the delivery of computing services, such as storage servers, networking, software, databases, intelligence, and analytics over the Internet platform. The users just need to simply connect their computers or physical devices with the cloud computing services. In this research, we used AWS and Heroku cloud computing services.

##### Amazon Web Services

AWS (Amazon Web Services) represents the world’s most comprehensively and largely used cloud computing platform. It offers many full features and services globally from the data centers. Millions of consumers, such as larger enterprises, growing startups, and leading government firms are using AWS to reduce their business costs, become more agile, and innovate faster (EL-YAHYAOUI & Dafir ECH-CHERIF EL KETTANI, 2019). Some of the key features of AWS include cost-efficiency, scalability, security, reliability, agility, and innovation. This cloud service enables the users to scale their IT resources up/down as per requirement without making larger upfront investments. It provides several pricing models to reduce the associated costs. It provides a higher level of reliability with a globalized data center to ensure sufficient availability of data. Also, it offers a range of potential security measures, including access control, encryption, and intrusion detection for better protecting the user’s sensitive data. Moreover, it facilitates an easy and quick deployment of the new services and applications. It constantly focuses on innovating and adding novel services to meet diverse business requirements.

##### Heroku

Heroku represents PaaS (Platform as a Service) that allows developers to better deploy, manage, and scale the applications. It provides an inclusive range of features that make it one of the most suitable platforms to develop and deploy various applications, specifically the proposed system to secure users’ sensitive data using homomorphic encryption. It offers several benefits, such as higher security, scalability, ease of deployment, and cost-efficacy. To use this cloud service in this research, we need to first create a Heroku account and create a new application. Then we need to deploy the proposed HE-based solution to Keroku. Then we have to configure its security features to protect the proposed solution (Munjal & Bhatia, 2022). Further, we need to continually monitor the performance of this system and make necessary adjustments. Some of the key considerations are performance, compliance, and key management that need to be followed while using HE-based solutions. Overall it is a powerful platform that could be utilized in developing and deploying the proposed HE-based solution to preserve sensitive data on cloud computing.

# Conclusion

In this chapter, we have discussed the selected research methodology, which is mixed research methodology. We used it to combine the vital research insights of both qualitative and quantitative research methods. Then we discussed the use homomorphic encryption algorithm to develop a security system to secure sensitive data present on cloud computing platforms. We discussed the following data collection process to collect secondary research data. Then we discussed the system requirements, including hardware and software requirements along with programming language, libraries, and cloud computing services.

Chapter 4: Project Development

# Introduction

This chapter is another crucial part of this research project, which provides a detailed discussion of the design, implementation, and testing of this project. It can serve as a detailed guide to understanding the methodology, contribution, and results of this project. The proposed homomorphic encryption-based system is developed using Python programming language with the SEAL and PALISADE libraries (Hallman et al., 2018). These libraries are important encryption libraries and among these libraries, SEAL is used to provide an efficient framework to perform homomorphic encryption operations and PALISADE provides highly optimized deployment of the homomorphic encryption to attain the most significant enhancements. Then we perform the developed system using black box testing to demonstrate its effectiveness in securing users’ sensitive data stored on the cloud computing. For this purpose, we will use different black box testing methods.

# Project Development

Here we will provide a comprehensive discussion of the project design and implementation along with the project testing. Here the project methodology and main findings will be communicated while demonstrating the credibility and rigor of this research to enable the most informed judgments (Mahmood & Ibrahem, 2018). Further, it contributes to advancing this research field by sharing vital research insights and knowledge from this research.

## Project design

Project design is an important stage of the whole project lifecycle, placing a solid foundation for the successful execution of the project. It includes defining the scope, objectives, and deliverables of the project while establishing a clear roadmap for this project and recognizing the processes and resources for attaining the project goals. This project design decides how to carry out the project in the most efficient manner. In this research, we used Python programming language to develop the back-end of the proposed homomorphic encryption-based system (Cholkar & Patel, 2023). This proposed method to secure users’ sensitive data on the cloud computing platform. This system is based on SEAL and PALISADE libraries that provide a flexible & robust framework for better performing homomorphic encryption operations and optimized disposition of the homomorphic operations for realizing significant performance enhancements.

Figure 10 depicts the architecture of the proposed system to secure user’s sensitive data stored on cloud computing. According to this diagram, a sender sends any message or data in plain text format. Then the encryption module of this proposed system encrypted this plain text into cipher text. Here a recipient’s public key is used to convert plain text into cipher text. Then after performing computations on this encrypted data without encrypting it, the decryption module of this system converts it into plain text using the recipient’s private key. Then this decrypted data is delivered to the recipient in its original form (Olzak, 2022). The data of users remained encrypted throughout this entire process to preserve their privacy and confidentiality

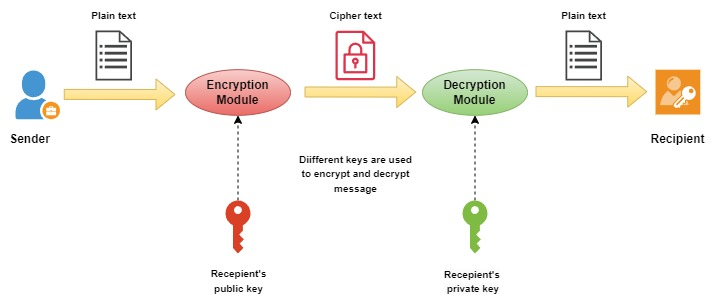


Figure : Architecture of the proposed system

## Implementation

In this section, we will discuss a detailed discussion of the implementation of the proposed homomorphic encryption-based security system to secure sensitive data on cloud computing. The below-illustrated figure depicts the workflow of the proposed system in this research to secure user’s sensitive data on cloud computing by performing computations on the encrypted data without decrypting it. This proposed system used a Fully Homomorphic Encryption algorithm, which is more efficient and secure to perform computations on third-party platforms and provides advantages of the properties of both multiplicate and additive homomorphism.

This diagram provides a detailed workflow of the proposed fully homomorphic encryption-based security system to secure user’s sensitive data on cloud computing. The owner creates data in the form of classified plain text. Then the data owner designs and uses a fully homomorphic encrypted-based security method to perform encryption for converting this plain data into encrypted data. Then this encrypted data is sent to the cloud computing platform, whether the cloud service providers provide the desired cloud services by the data owner (Albakri et al., 2023). This data is processed by cloud service providers without seeing it. Then homomorphic computations are performed on this encrypted data through any third-party platform while keeping the user’s data encrypted. After performing computations, the results are collected from the cloud platform in encrypted formats. Then decryption is performed to convert this encrypted data into its original format using a private key. Finally, the decrypted results are delivered to the data owner. The owner’s data remained encrypted throughout this entire process to ensure an effective level of privacy and confidentiality of the user’s sensitive data.

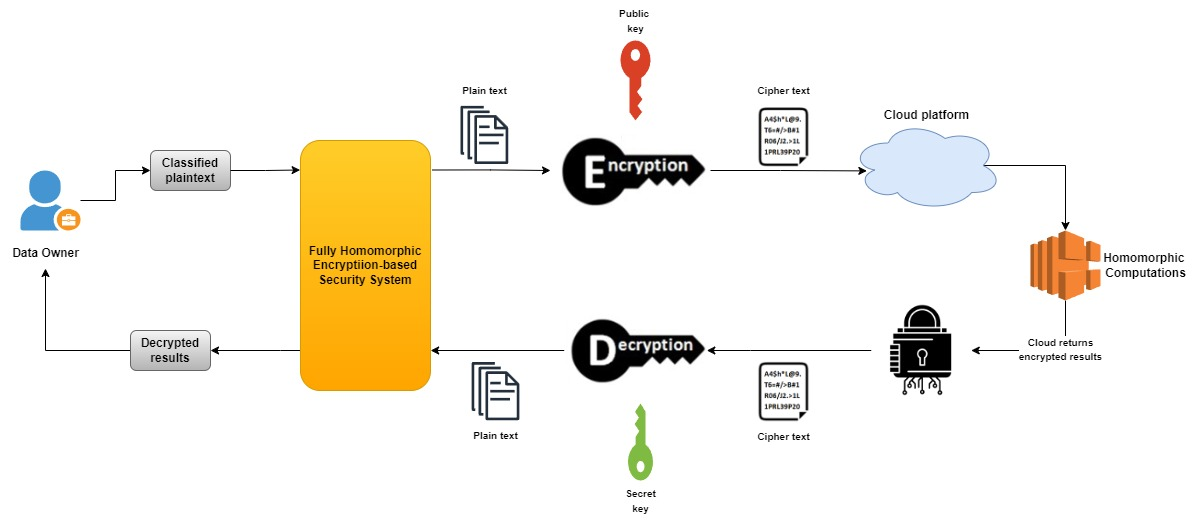


Figure : Workflow of the proposed homomorphic encryption-based system

## Testing

Testing of the proposed homomorphic-based system to secure the user’s sensitive data on cloud computing is performed to ensure the efficacy and reliability of protecting confidential information in the cloud computing atmospheres. A testing method comprises several system aspects, such as performance, security, functionality, and usability.

### Black box testing

This testing also known as behavioral testing, represents a testing approach that mainly focuses on dealing with the proposed software system’s external behavior without considering the details related to its internal implementation (Jabbar, 2016). In this research, we can use black box testing to validate the system’s functionalities as envisioned and ensure that it meets of not the stated requirements without investigating underlying code structures or cryptographic algorithms.

Unit testing, integration testing, and system testing play a critical role in ensuring the efficacy and reliability of the proposed system in this research work.

**Unit testing**: This testing method mainly focuses on testing the individual modules or components of the homomorphic encryption-based security system. Using this testing method, we can test that each component of the proposed system functions correctly and meet all stated requirements. It includes testing of individual classes, functions, or modules in isolation to ensure that they generate anticipated for different input scenarios.

**Integration testing**: The integration testing approach mainly focuses on communication and interaction between diverse modules or components of the proposed homomorphic encryption-bases system. It validates that the individual components of the proposed system work seamlessly together and generate the anticipated behavior (Awadallah & Samsudin, 2020). It includes the testing of how these diverse system components interact, handle errors, and exchange data.

**System testing**: The system testing test and evaluate the proposed system as a whole to ensure that it correctly meets all the intended functions and requirements in the considered cloud computing atmosphere. It includes the system testing against the state scenarios, use cases, and non-functional requirements, such as security, usability, and performance.

# Conclusion

In this chapter, we discussed the development process of the proposed FHE-based security solution. Here we presented the design, implementation, and testing of the proposed system. Here we first provided an architecture of the proposed FHE-based solution, which comprises a sender, encryption module, decryption module, and recipient. Here two different keys, namely the public key and the private key are used to perform encryption and decryption, respectively. Then a detailed workflow of this proposed system is provided to discuss the entire process of performing computations on the encrypted data by the third-party platforms while keeping the user’s data encrypted throughout the entire process. Then we used black box testing to test the effectiveness of the proposed system while ensuring that it functions as expected or not.

Chapter 5: Project Evaluation

# Introduction

This chapter is based on the discussion of inclusive project evaluation of the entire research project by addressing the running and completed project activities. This chapter serves as the most valuable section to evaluate the performance and effectiveness of this project. Here we will present the evidence of performance practical work along with interpreting the obtained research results. Then the commercial and economic feasibility of the proposed solution will be justified by addressing its usage in practical scenarios for checking whether it performs as expected or not. Here specific criteria will be addressed and followed to evaluate the proposed system in this research. The presented evidence addresses that the proposed solution can efficiently preserve the privacy of users’ sensitive data stored on the cloud platforms.

# Project Evaluation

The below-illustrated figure depicts the details of the installation of TenSeal and Pillow libraries to perform the homomorphic encryptions on tensors (Hallman et al., 2018). It provides an easier use via Python API while preserving the efficiency of the performed operations.

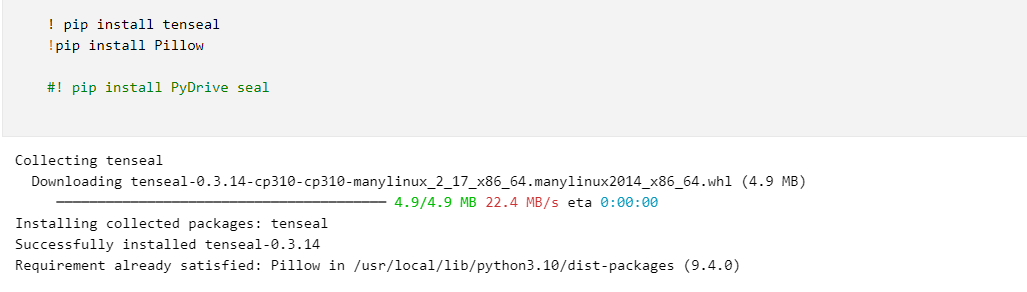


Figure : Installing TenSeal and Pillo libraries

This diagram depicts the details of the defined functions to create the folders that don’t exist. We define create\_folders for creating encrypted\_folder and decrypted\_folder.

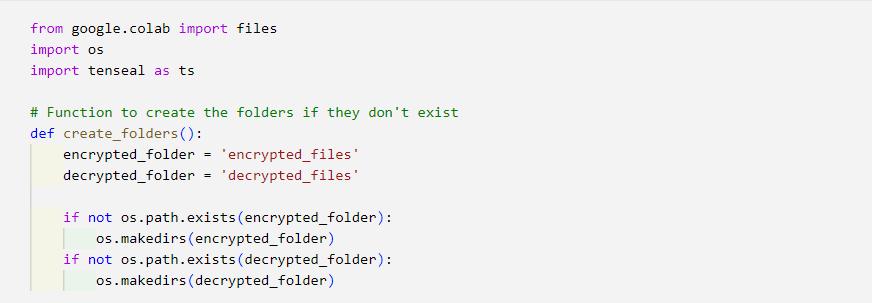


Figure : Declaring functions to created folders that don’t exist

Figure 14 depicts the details of the defined function, namely the encrypt\_and\_save function for encrypting and saving the stored file. Here we read the content of the file as the binary data, set a suitable scale value based on the given range of data, and then encrypte the content of the file into chunks.

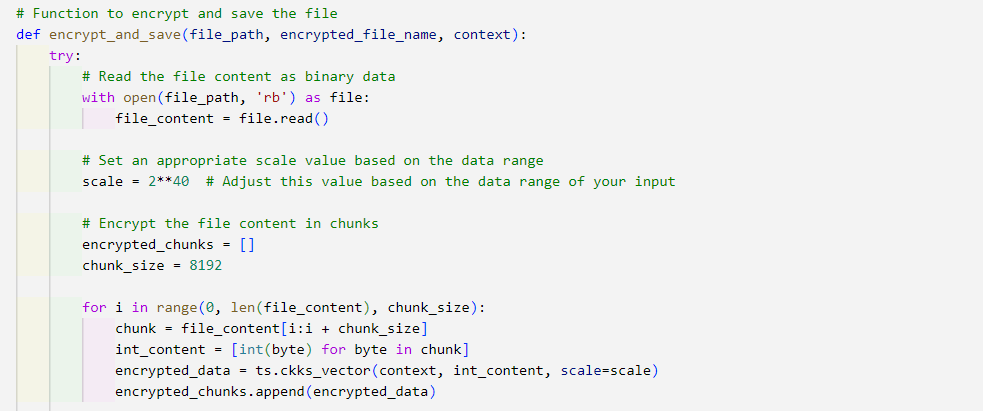


Figure : Defining function for encrypting and saving the file

In the below illustrated diagram, the encrypted chunks are serialized to strings. It helps to encrypt data from being stolen, changed, and stolen (Awadallah & Samsudin, 2020). Further, these serialized encrypted chunks are saved following an encrypted\_file\_path.



Figure : Serializing the encrypted chunks into strings

The below-illustrated diagram depicts the details of the functions for downloading the saved encrypted file. Here we define the download\_encrypted\_file function for downloading the saved encrypted file. Then we create folders and widgets to upload the files.



Figure : Defining function to download the encrypted file

The below-illustrated diagram presents the process for uploading the file. Here first we create a TenSEAL context with the increased polymodulus degree. Then we encrypt and save all the files present in the context. Then all the encrypted files are downloaded and moved inside the loop.

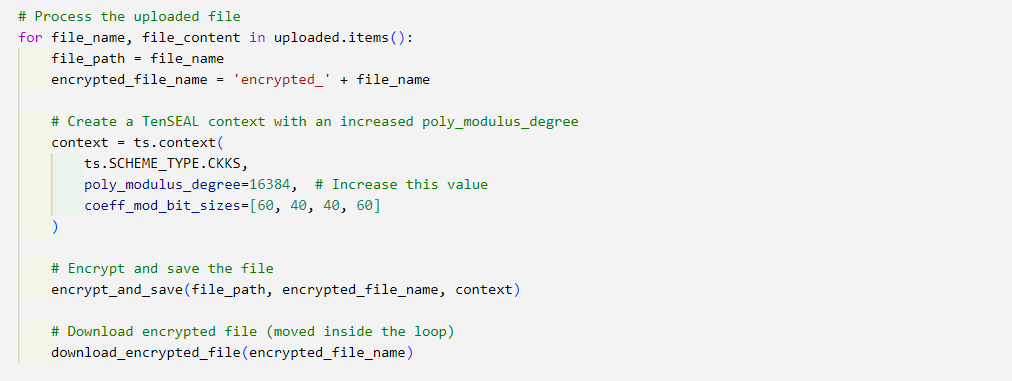


Figure : Process for uploading the file

The below diagram addresses the defined list\_encrypted\_files function, which is used for listing all the files present in the considered encrypted folder (Joseph & Mohan, 2022). Here we check for encrypted files, if no encrypted files are found then the function returns to none and if encrypted files are found, then the function prints all the found encrypted files.



Figure : Getting lost in the present files in the encrypted folder

In the below-illustrated diagram, we define a decrypt\_and\_download function with try and catch block. Here first the content of the encrypted file is read and then this encrypted content is deserialized into encrypted chunks. Further, these encrypted chunks are concatenated to join into a string.

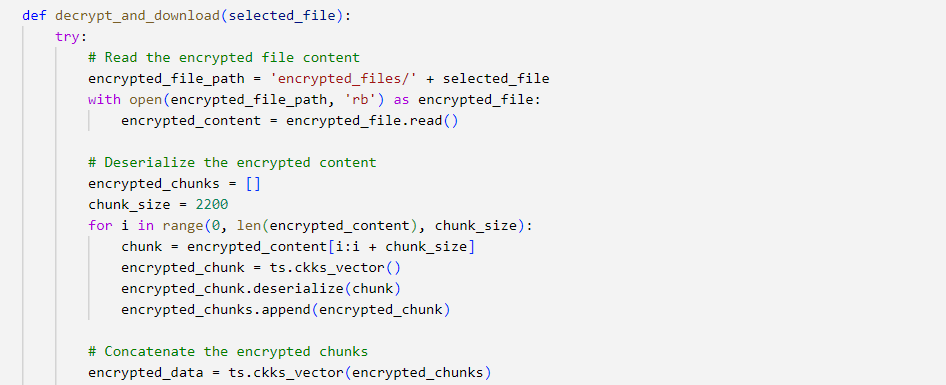


Figure : Reading and deserializing the encrypted file's content

In the below-illustrated diagram, we first decrypt the encrypted data and then save the decrypted data into a separate file. Further, this decrypted file is downloaded (Rupa et al., 2023). Here exception handling is also performed to handle the error if any decrypted file is not found in the file.



Figure : Decrypting the encrypted data and saving it to a separate file

Here we define the download\_original\_image function on the selected files. Here, we also try and catch blocks to handle the arising exceptions. Here first the name of the original files is extracted and then the original image is downloaded. An error message will be printed if any error is found.

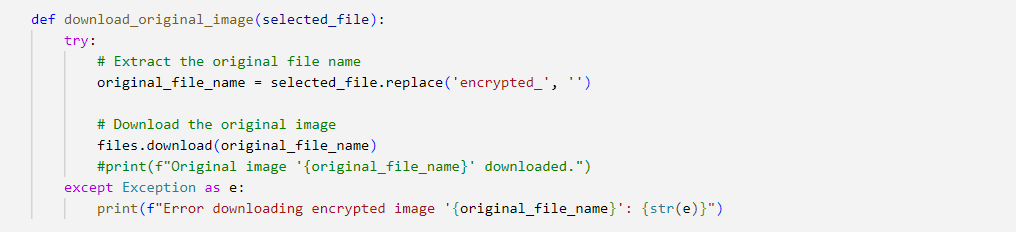


Figure : Extracting the original file name and downloading the original image

In the below-illustrated diagram, the encrypted files are listed, and using try and catch block the exceptions are handled. Here the selected files are decrypted and downloaded (Mei et al., 2022).

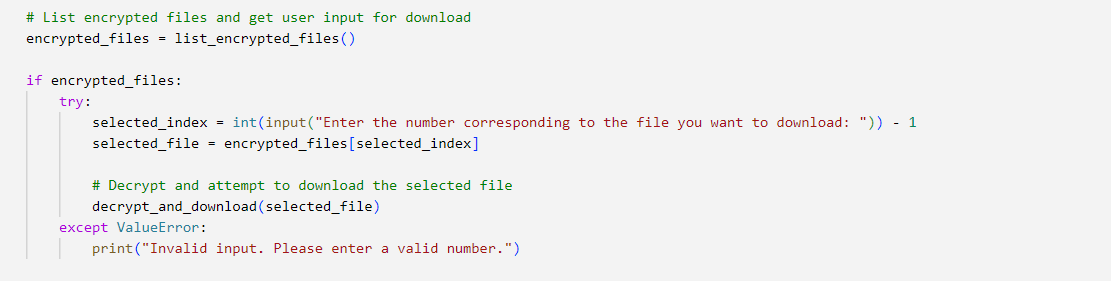


Figure :Listing encrypted files and downloading selected files

In the below-illustrated figure, a delete\_files\_in\_folder function is defined and then a try block is used to list all the files in the folder and delete each of these files. Then except is used to handle any exception that arises. Here we specify the path of the folder from which we want to delete files. Then we delete files from the encrypted and decrypted folders.



Figure : Listing and deleting the files in the folder

## Feasibility of the proposed system in the commercial and economic contexts

The proposed homomorphic encryption-enabled security system has immense potential to secure sensitive data and revolutionize cloud computing by facilitating secured processing and analysis of data without data privacy compromise. Homomorphic encryption is a novel technology and its latent benefits make it a promising solution to secure sensitive data stored in the cloud environments.

Its ability to provide several advantages, such as improved compliance, enhanced data privacy, novel business opportunities, increased efficiency, and decreased costs makes it commercially feasible. The proposed HE-based solution facilitates secure processing and analysis of cloud data by disclosing underlying data and addressing the concerns related to data security and privacy in cloud computing (Joseph & Mohan, 2022). It can also help companies in complying with the regulations of data privacy, such as CCPA and GDPR by ensuring that the user’s confidential data always remains protected. It also opens novel business opportunities for cloud providers to enable them to offer secure data processing and analysis services. It significantly reduces the associated costs to eliminate the need to invest in on-premises infrastructure to protect sensitive data during processing and analysis. It also enables the companies to process and analyze data without decrypting it to ultimately save the spent time and resources.

The proposed HE-based solution enables the organization to not continually invest and maintain the on-premises infrastructure to securely process and analyze the user’s data along with decreasing the software & hardware acquisition, operational costs, and maintenance. It also reduces the requirement of expensive on-site infrastructure to optimize the costs of cloud computing usage. It fosters confidence and trust among the customers, stakeholders, and businesses by ensuring that the user’s sensitive data always remains secure and protected in the cloud atmosphere (Rupa et al., 2023). It facilitates secured data-centric decision-making by improving risk management, operational efficiency, and competitive advantages. Moreover, it contributes to economic innovation and growth by facilitating novel business services, models, and products that can influence the secure processing and analysis of cloud data. However, the initial deployment cost of the HE-based security system might be higher than the conventional encryption techniques but their long-lasting cost savings, improved security, and novel business opportunities make the highly economically feasible and attractive solution for organizations in diverse industries.

# Conclusion

In this chapter, we evaluate the proposed system in this project. The proposed solution has been evaluated based on specific evaluation criteria along with presenting the evidence of performed practical work. The presented evidence demonstrates that the proposed solution is capable of defending the user’s sensitive data throughout the storage, processing, and analysis. Also, the proposed solution is highly feasible in both commercial and economic contexts. The ability of this proposed solution to secure data and eliminate the need to invest huge money makes it more feasible.

Chapter 6: Conclusion

# Conclusion

In this chapter, we conclude this research by summarizing the work done in this research and key findings. Here we will also discuss the encountered problems during this research and recommendations to proceed with this research in the future.

## Conclusion

This research is mainly focused on securing the sensitive data of users, stored and processed on the cloud computing platforms. The traditional methods provide an effective level of security but they need to decrypt the user’s data for performing computations, which pose security vulnerabilities. Thus, this research mainly focused on developing and presenting a novel security solution based on homomorphic encryption to resolve this issue and potentially protect the user’s sensitive data. This whole dissertation is divided into 6 main chapters, including introduction, literature review, project methodology, project development, project evaluation, and conclusion. In the first chapter, this research project is introduced, including the research background, problem statement, aims & objectives, research questions, project description, project methodology, project plan, risk assessment, and project overview. In the second chapter, we conducted a detailed literature review of the existing and previously published relevant research papers to gain a better understanding of this research. In the third chapter, we discussed the use of mixed research methodology and homomorphic encryption algorithm to protect user’s sensitive data. Then we discussed the hardware and software requirements, including programming language, libraries, and cloud services. Then in the fourth chapter, we discussed the development of the proposed system, including the design, implementation, and testing of the system. The fifth chapter is based on the evaluation of this project by providing evidence of the performed practical work and the feasibility of the proposed solution in commercial and economic contexts. Then in the sixth chapter, we concluded this research by discussing the encountered problems and recommendations to further proceed with this research.

## Encountered problems

However, the proposed homomorphic encryption-based security system efficiently protects the user’s sensitive data on the cloud computing platforms but it poses a few challenges. The performed computations on the homomorphic encryption might be computationally expensive for complex and larger datasets (Mei et al., 2022). It may adversely impact the cloud application’s performance in security data. Also, the HE method lacks standardization, which may pose compatibility problems and deter its comprehensive adoption. Organizations need to invest in robust practices for secret key management to protect the integrity of the secret keys and prevent unauthorized access. It preserves data privacy by facilitating computations on the encrypted data, which can prevent the encrypted data’s malicious modification. Moreover, organizations need to educate the users about the usage and benefits of HE for securely handling data in cloud computing atmospheres.

## Recommendations

Based on the current state of this research and encountered problems, here we provide some recommendations to further explore HE-based security systems to secure sensitive data in cloud computing.

* Future research needs to focus on optimizing the HE scheme’s computational efficiency by decreasing performance overhead and facilitating real-time processing of the encrypted data.
* Future research should recommend collaborating with standard organizations and industry stakeholders to develop consistent HE APIs and protocols to promote interoperability and comprehensive adoption across diverse cloud providers and platforms.
* They should develop advanced security protocols and methods for key management to secure the secret keys from any unauthorized access and ensure the confidentiality and integrity of the encrypted data.
* They should examine the mechanisms for ensuring the integrity of user data and preventing encrypted data tampering.
* They need to explore the application-centric optimizations of HE-enabled security systems to tailor this technology to fulfill the specific requirements of the diverse cloud computing applications (Zhang et al., 2019).
* They should focus on developing effective methods to integrate the HE-enabled security system with the existing cloud computing architectures and frameworks to enable seamless integration and widespread protection of sensitive data.
* Effective open-source tools and implementations should be fostered to promote a more secure, efficient, and comprehensively applicable security system in the landscape of cloud computing.

# References

Albakri, A. *et al.* (2023) ‘Fully homomorphic encryption with Optimal Key Generation Secure Group Communication in internet of things environment’, *Applied Sciences*, 13(10), p. 6055. doi:10.3390/app13106055.

Alkharji, M. and Liu, H. (2016) ‘Homomorphic Encryption Algorithms and Schemes for Secure Computations in the Cloud’, *International Conference on Secure Computation and Technology* [Preprint].

Awadallah, R. and Samsudin, A. (2020) ‘Homomorphic encryption for cloud computing and its challenges’, *2020 IEEE 7th International Conference on Engineering Technologies and Applied Sciences (ICETAS)* [Preprint]. doi:10.1109/icetas51660.2020.9484283.

Babenko, M. *et al.* (2023) ‘A comparative study of secure outsourced matrix multiplication based on homomorphic encryption’, *Big Data and Cognitive Computing*, 7(2), p. 84. doi:10.3390/bdcc7020084.

Benzekki, K., El, Abdeslam and El, Abdelbaki (2016) ‘A secure cloud computing architecture using homomorphic encryption’, *International Journal of Advanced Computer Science and Applications*, 7(2). doi:10.14569/ijacsa.2016.070241.

Bhowmik, S. (2017) ‘Cloud computing services’, *Cloud Computing*, pp. 76–94. doi:10.1017/9781316941386.006.

Biksham, V. and Vasumathi, D. (2017) ‘Homomorphic encryption techniques for securing data in Cloud computing: A survey’, *International Journal of Computer Applications*, 160(6), pp. 1–5. doi:10.5120/ijca2017913063.

Biksham, V. and Vasumathi, D. (2016) ‘Query based computations on encrypted data through homomorphic encryption in Cloud computing security’, *2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)* [Preprint]. doi:10.1109/iceeot.2016.7755429.

Cholkar, P. and Patel, M. (2023) ‘Introduction: Cloud storage security and homomorphic encryption in cloud computing’, *International Journal of Science and Research (IJSR)*, 12(10), pp. 1816–1822. doi:10.21275/sr231017115439.

Das, D. (2018) ‘Secure cloud computing algorithm using homomorphic encryption and multi-party computation’, *2018 International Conference on Information Networking (ICOIN)* [Preprint]. doi:10.1109/icoin.2018.8343147.

EL-YAHYAOUI, A. and Dafir ECH-CHERIF EL KETTANI, M. (2019) ‘A verifiable fully homomorphic encryption scheme for cloud computing security’, *Technologies*, 7(1), p. 21. doi:10.3390/technologies7010021.

El Makkaoui, K. *et al.* (2019) ‘Fast cloud–paillier homomorphic schemes for protecting confidentiality of sensitive data in cloud computing’, *Journal of Ambient Intelligence and Humanized Computing*, 11(6), pp. 2205–2214. doi:10.1007/s12652-019-01366-3.

El Makkaoui, K., Ezzati, A. and Hssane, A.B. (2015) ‘Challenges of using homomorphic encryption to secure cloud computing’, *2015 International Conference on Cloud Technologies and Applications (CloudTech)* [Preprint]. doi:10.1109/cloudtech.2015.7337011.

Gnana Sophia, S., Thanammal, K.K. and Sujatha, S.S. (2022) ‘Secure storage and accessing the data in cloud using optimized homomorphic encryption’, *Journal of Control and Decision*, 10(1), pp. 90–98. doi:10.1080/23307706.2022.2078436.

Hallman, R.A. *et al.* (2018) ‘Building applications with homomorphic encryption’, *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security* [Preprint]. doi:10.1145/3243734.3264420.

Hammami, H. *et al.* (2017) ‘Using homomorphic encryption to compute privacy preserving data mining in a cloud computing environment’, *Information Systems*, pp. 397–413. doi:10.1007/978-3-319-65930-5\_32.

Jabbar, I. (2016) ‘Using fully homomorphic encryption to secure cloud computing’, *Internet of Things and Cloud Computing*, 4(2), p. 13. doi:10.11648/j.iotcc.20160402.12.

Joseph, M. and Mohan, G. (2022) ‘Design a hybrid optimization and homomorphic encryption for securing data in a cloud environment’, *International Journal of Computer Networks and Applications*, 9(4), p. 385. doi:10.22247/ijcna/2022/214502.

Kangavalli, R. and Vagdevi S (2015) ‘A mixed homomorphic encryption scheme for secure data storage in cloud’, *2015 IEEE International Advance Computing Conference (IACC)* [Preprint]. doi:10.1109/iadcc.2015.7154867.

Mahmood, Z.H. and Ibrahem, M.K. (2018) ‘New fully homomorphic encryption scheme based on multistage partial homomorphic encryption applied in cloud computing’, *2018 1st Annual International Conference on Information and Sciences (AiCIS)* [Preprint]. doi:10.1109/aicis.2018.00043.

Marwan, M., Kartit, A. and Ouahmane, H. (2016) ‘Applying homomorphic encryption for Securing Cloud Database’, *2016 4th IEEE International Colloquium on Information Science and Technology (CiSt)* [Preprint]. doi:10.1109/cist.2016.7804968.

Medileh, S. *et al.* (2023) ‘A multi-key with partially homomorphic encryption scheme for low-end devices ensuring data integrity’, *Information*, 14(5), p. 263. doi:10.3390/info14050263.

Mei, R. *et al.* (2022) ‘Considerations on evaluation of practical cloud data protection’, *Communications in Computer and Information Science*, pp. 51–69. doi:10.1007/978-981-19-8285-9\_4.

Mishra, A., 2023. Homomorphic Encryption: Securing Sensitive Data in the Age of Cloud Computing.

Munjal, K. and Bhatia, R. (2022) ‘A systematic review of homomorphic encryption and its contributions in healthcare industry’, *Complex &amp; Intelligent Systems*, 9(4), pp. 3759–3786. doi:10.1007/s40747-022-00756-z.

Olzak, T. (2022) *Homomorphic encryption: How it changes The way we protect data*, *Spiceworks*. Available at: https://www.spiceworks.com/it-security/data-security/articles/how-homomorphic-encryption-protects-data/ (Accessed: 19 November 2023).

R, R.C. *et al.* (2023) ‘A multi-stage partial homomorphic encryption scheme for secure data processing in cloud computing’, *2023 2nd International Conference on Edge Computing and Applications (ICECAA)* [Preprint]. doi:10.1109/icecaa58104.2023.10212153.

Rupa, Ch., Greeshmanth and Shah, M.A. (2023) ‘Novel secure data protection scheme using Martino homomorphic encryption’, *Journal of Cloud Computing*, 12(1). doi:10.1186/s13677-023-00425-7.

Suveetha, K. and Manju, T. (2016) ‘Ensuring confidentiality of cloud data using homomorphic encryption’, *Indian Journal of Science and Technology*, 9(8). doi:10.17485/ijst/2016/v9i8/87964.

Tebaa, M., Hajji, S.E. and Ghazi, A.E. (2018) ‘Homomorphic encryption method applied to cloud computing’, *2012 National Days of Network Security and Systems* [Preprint]. doi:10.1109/jns2.2012.6249248.

Thabit, F. *et al.* (2022) ‘A novel effective lightweight homomorphic cryptographic algorithm for Data Security in cloud computing’, *International Journal of Intelligent Networks*, 3, pp. 16–30. doi:10.1016/j.ijin.2022.04.001.

Wanjale, K., Mangla, M. and Marathe, P. (2021) ‘Security of sensitive data in cloud computing’, *Machine Learning Approach for Cloud Data Analytics in IoT*, pp. 99–118. doi:10.1002/9781119785873.ch5.

Yakoubov, S. *et al.* (2014) ‘A survey of cryptographic approaches to securing big-data analytics in the cloud’, *2014 IEEE High Performance Extreme Computing Conference (HPEC)* [Preprint]. doi:10.1109/hpec.2014.7040943.

Yang, W. *et al.* (2023) ‘A review of homomorphic encryption for privacy-preserving biometrics’, *Sensors*, 23(7), p. 3566. doi:10.3390/s23073566.

Zhang, Y. *et al.* (2019) ‘Foresee: Fully outsourced secure genome study based on homomorphic encryption’, *BMC Medical Informatics and Decision Making*, 15(S5). doi:10.1186/1472-6947-15-s5-s5.

Zhao E, M. and Geng, Y. (2019) ‘Homomorphic Encryption Technology for Cloud Computing’, *Procedia Computer Science*, 154, pp. 73–83. doi:10.1016/j.procs.2019.06.012.

Zkik, K., Orhanou, G. and El Hajji, S. (2016) ‘Secure scheme on mobile multi-cloud computing based on Homomorphic encryption’, *2016 International Conference on Engineering &amp; MIS (ICEMIS)* [Preprint]. doi:10.1109/icemis.2016.7745297.