International Journal of Scientific Research in Computer Science, Engineering and Information Technology
© 2020 IJSRCSEIT | Volume 6 | Issue 2 | ISSN : 2456-3307
DOI : https://doi.org/10.32628/IJSRCSEIT

Ensuring Data Security in Cloud Computing using Homomorphic Encryption

Kedir Salih Siraj
Lecturer & Chair, PG Programme, School of Computing and Informatics, Dilla University, Ethiopia

ABSTRACT

Cloud computing is the most interesting and new technology which offers computing as service rather a product to its clients on demand through Internet. Since cloud computing provider stores the data and distributed resources in the open environment, however the major issues in cloud computing is the security of the data being stored in the cloud and privacy while the data is being transmitted to and from a cloud organization. We can use traditional encryption algorithms (like, AES, DES, TDES, and RSA etc.) to secure the storage of data in cloud provider. But they don't allow to perform operations on encrypted data without giving the private keys to cloud remote server. So; there is a need for new mechanism to perform on ciphered data which provide data security such as confidentiality and privacy for cloud service users. Homomorphic encryption is a form of security technique which allows to execute computations on ciphered user's data without having to decrypt and produce an encrypted result which, when decrypted, it is same as the result when operations performed on the plaintext. In this paper, paillier homomorphic algorithm is applied on encrypted data, MD5 algorithm for authentication and one time password for verification of identity is used. Thus, result of confidentiality and privacy of cloud client data are achieved through paillier homomorphic encryption and OTP and also proposed system allows cloud service provider to perform computations on encrypted data without knowing the secret key.

Keywords : Cloud computing, Paillier Homomorphic Encryption, MD5, OTP.

I. INTRODUCTION

Data security is a science and study of methods of protecting data in computer and communication systems [1]. Protecting data means such as a database from destructive forces and from the unwanted actions of unauthorized users where protecting a data is defined by a mixture of confidentiality, integrity and availability (CIA) factors, in accordance with business needs and any legal, regulatory requirements and constraints [2].

The nature and techniques of data security should consider the underlying cloud computing platform. In cloud computing, which is developing technology and internet-based computing, large groups of remote servers are networked to allow the centralized data storage and online access to computer services and resources. Cloud computing offers several benefits like fast deployment, pay for use, lower costs, ubiquitous network access, low-cost disaster recovery and data storage solutions, real time detection of system tampering and rapid re-constitution of services, greater flexibility and optimal resource utilization [3, 4].

Cloud computing has services like Software as a Service (SaaS), Platform as a Service (PaaS) and Infrastructure as a Service (IaaS) implemented through four cloud deployment models. These are private, public, hybrid and community [5]. However, there are also common challenges of cloud computing like security and privacy, interoperability and portability, reliability and availability, resource management and scheduling, energy consumption, virtualization, performance and bandwidth cost are some of the known lists [7].

II. Data Security in Cloud Computing

2.1 Importance of Security

Security is the main component in storing the sensitive data in cloud and also strong privacy protections in cloud computing environments. Cloud consumers and service providers both are willing to use cloud computing service only if they trust that, their data will be remain confidential and secure, because the data stored on the cloud is often seen as valuable to individuals with malicious intent [31]. Security refers to confidentiality, integrity and availability, which pose major issues for cloud service providers and cloud consumers [32].

2.2 Data Security Issues

Data security is the major issues when it comes to cloud computing. Since a third party stores the sensitive data and confidential business data, it never known what is going on with the data. Along with the benefits of business process outsourcing (BPO) comes an increased risk of data, unless the organization can protect its data [30, 31]. Data security has played an important role in hindering cloud computing acceptance. According to different research result, the importance of security in cloud compared to other parameters takes the higher part as shown in **Figure 2.3** [34].

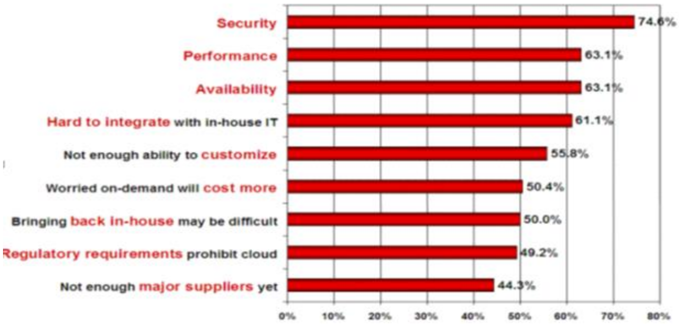


Figure 2.1: The issues which mainly affect the performance of cloud computing [6].

Even though, the virtualization and cloud computing delivers wide range of dynamic resources, the security concern is generally perceived as the huge issue in the cloud which makes the consumers to resist themselves in adopting the technology of cloud computing. Some of the main security issues in the cloud computing are discussed below [34].

a. Confidentiality is an act that ensures private or sensitive data from being disclosed to an unauthorized person. Confidentiality loss occurs when data can be viewed or read by any individual who is unauthorized to access it [34]. A good example of methods used to ensure confidentiality for a data like account number when banking online is encryption. Data encryption is a common method of ensuring confidentiality. User IDs and passwords constitute a standard procedure; two -factor authentication. Other options include biometric verification and security tokens [35].

b. Integrity: Involves maintaining the consistency, accuracy, and trustworthiness of data over its entire life cycle [35]. Integrity is the process of ensuring that user's data captured in a system is the original representation of the data and it has not been modified in transit by an unauthorized person [25]. In addition, some means must be in place to detect any changes in data that might occur as a result of non-human-caused events such as server crashes. Some data might include checksums, for verification of integrity. Backups or redundancies must be available to restore the affected data to its correct state [35].

c. Availability ensures that data processing resources are not made unavailable by malicious action. It is the simple idea that when a user tries to access, it is available to be accessed by authorized users only at any time [34]. Redundancy, failover, Fast and adaptive disaster recovery is essential for the worst case scenarios [35].

d. Authentication: is a procedure by which one entity checks the identity of another entity. This can be a person or program. The authentication procedure can be done in three ways; something that the user knows, for example, password and user name, something the user has, such as a PIN, something the user is, for example, finger print [25].

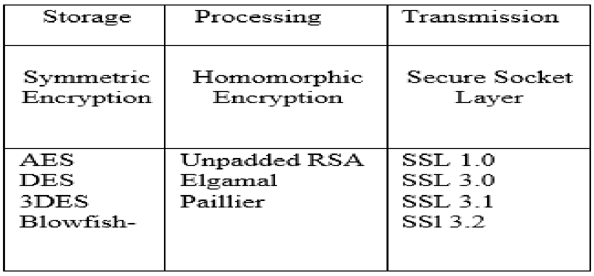
e. Authorization: is the process that ensures that a person has the privilege to access certain resources. Users can not be permitted to access any resources without knowing the attributes of such users. Users can have access rights to resources; but the authority to do something is not within their reach. For instance, a customer can use ATM card to withdraw money from the ATM machine. Having been authenticated, he cannot withdraw beyond a recommended maximum irrespective of any amount he has in his bank account. Cloud computing uses these access control and authorization to manage resources usage and limit fake practices [25].

f. Privacy: Only the authorized users can access their sensitive data at any time and can do any operations such as read, write, execute and also update etc. It involves retaining confidentiality [25].

2.3 Data Security Techniques

There are three types of data in cloud computing, the first data in transit (transmission data), the second data at rest (storage data), and finally data in processing (processing data). Every cloud service provider encrypts the data in three types according to **Table 2.1** [9].

Table 2.1: *Data security in cloud computing*



Storage	Processing	Transmission
Symmetric Encryption	Homomorphic Encryption	Secure Socket Layer
AES DES 3DES Blowfish-	Unpadded RSA Elgamal Paillier	SSL 1.0 SSL 3.0 SSL 3.1 SSl 3.2

2.4 Cryptography Methods

Cryptography is a techniques, which provide security by encrypting the message that can be unreadable and also it is very important in the field of information technology and used to secure data privacy using standard methods. Cloud computing, one of the dominant methods where computing resources are delivered on demand, make use of several data encryption techniques to secure the cloud servers. Basically, there are three types of cryptographic techniques [36].

2.4.1 Symmetric Key Cryptography

Symmetric key (secret key cryptography) uses a single key for both encryption and decryption of the message. The key used for encryption and decryption is called the secret key. Symmetric cryptography is the best known technique and perhaps the oldest. There are two types of symmetric-key encryption, stream ciphers and block ciphers. Stream ciphers encrypt a message as a stream of bits one at a time. Block ciphers take blocks of bits, encrypt them single unit. Some well know examples Block ciphers include Twofish, AES (Rijndael), Blowfish, RC4, TDES [36].

In a symmetric cryptography, the secret key is applied to the text of a message to change its content in specific manner way. The problem by using secret keys is, it is transmitted over the Internet, where intermediate unauthorized users can hack them. This problem can be resolved with asymmetric encryption, where two related keys, a public key which is freely available to anyone who wants to send a message and a private key which is kept secret is used for decryption [37].

2.4.2 Asymmetric Key Cryptography

In Asymmetric Cryptography, public key is used for encrypting the message and secret key is used for decrypting the message. The public key will be available to anyone to send the message and secret key is kept confidential, only the authorized person who have the secret key can view the message. Confidentiality and integrity can be achieved. Public key algorithms provide key distribution security Diffie Hellman, Digital Signatures, Elliptic curve cryptography and RSA [36]. Homomorphic encryption is asymmetric technique applied on the cloud computing security. With this technique, computation can be performed on encrypted data without decryption. Once operations are performed on client data by the cloud server. The client can decrypt the result without the cloud service provider knowing anything about the data it operated on [17].

2.4.3 Hash Function Cryptography

In hash functions, uses a mathematical computations that takes input value and it generates fixed size blocks of data. The inputs to a hash function are typically called messages, and the outputs the hash function is called message digest. Primarily used for message integrity. The most widely used hash function are MD5, SHA1.

Table 2.2 below shows the comparison of symmetric (secret key encryption) and asymmetric (public key) encryption techniques [38].

Table 2.2: *Comparison of Encryption Techniques.*

Property	Symmetric Encryption	Asymmetric Encryption
Types	DES, TDES, AES, Blowfish	RSA, Elgamal, Paillier Algorithm
Key Distribution	Difficult	Easy
Security	Moderate	Highest
Security secure Service	Confidentiality	Confidentiality, integrity and non-repudiation
Encryption and Decryption	Faster	Slow
Complexity	DES	RSA

III. Homomorphic Encryption

Homomorphic encryption is a form of encryption that allows specific types of computations to be carried out on encrypted data (on cipher texts), generating an encrypted result, when decrypted, matches the result of operations performed on the plaintext [17]. Homomorphic encryption can also be used to securely chain together different services without exposing sensitive data. For example, services from different companies can calculate the tax, currency exchange rate and shipping, on a transaction without exposing the unencrypted data to each of those services. Homomorphic encryption can be used to create secure systems such as secure voting systems, collision-resistant hash functions, and private information retrieval schemes and enable widespread use of cloud computing by ensure the confidentiality of processed data [45].

Figure 3.1 [46] gives a visual representation of how homomorphic encryption is implemented.

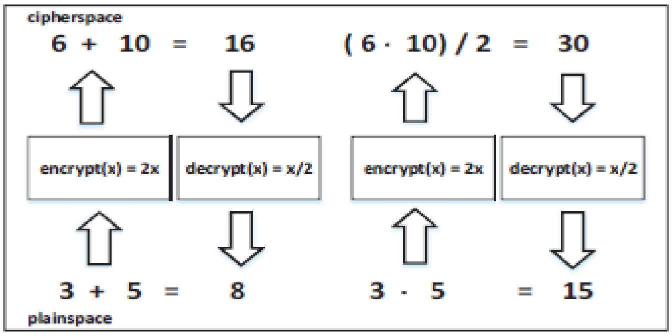


Figure 3.1: Example of an encryption of numerical values using homomorphic encryption.

Homomorphic encryption scheme consists of four function in [47].

- **KeyGen (λ):** Input-the security parameter λ and generates pair of keys (public key pk and secret key sk). **(pk, sk) ←KeyGen (λ).**
- **Encrypt:** an encryption algorithm that takes the public key and the plain text to encrypt M and gives the cipher text. **$C \leftarrow Enc_{pk}$ (M).**
- **Decrypt:** a decryption algorithm that takes the secret key and the cipher text c and recovers the plain text M. **$M \leftarrow Dec_{sk}$ (C).**
- **Evaluation:** Server has a function f for doing evaluation of cipher text and calculate the function using public key pk. Function **Eval:** applies a function f to a cipher text c using the public key. **$C^* \leftarrow Eval_{pk}$ (f, c),** where f can be: +, ×, ⊕ and without using the private key.

Homomorphic encryption cryptosystems can be classified into different types based on the operations that it allows on its raw data. Examples of homomorphic cryptosystems are [47].

- Partially Homomorphic Encryption
- Fully Homomorphic Encryption

3.1 Partially Homomorphic Encryption

Partially homomorphic encryption cryptosystem can support only a single operation on encrypted data. Those cryptosystem can divided based on the operation and also they support on additive and multiplicative homomorphic encryption [48]. Example of partially homomorphic encryption are as follows.

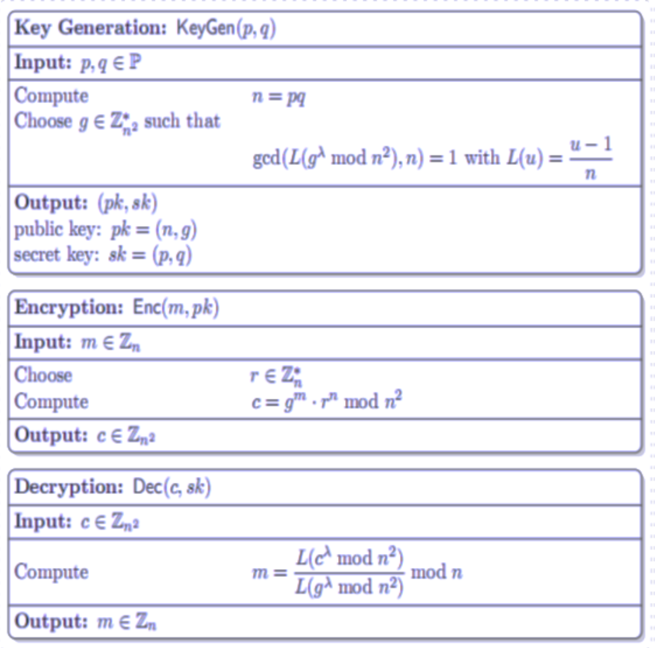
3.1.1. Additive Homomorphic Encryption

This kind of Encryption scheme enables homomorphic computation of only the addition operation. Some well-known example of additive homomorphic is Paillier Encryption cryptosystem. These cryptosystems satisfy the property that the product of two cipher texts will decrypt to the sum of their plain texts [48].

Enc (x + y) = Enc (x).Enc (y).

3.1.2. Paillier Encryption Cryptosystem

The Paillier cryptosystem, presented by Pascal Paillier in 1999, is a probabilistic asymmetric algorithm for public key cryptography and inherits additive homomorphic properties. Paillier scheme is an efficient and can be proven semantically secure [49]. The three stages (key generation, encryption and decryption) can be found in the following figure.



Key Generation: $\mathsf{KeyGen}(p, q)$

Input: $p, q \in \mathbb{P}$

Compute $n = pq$

Choose $g \in \mathbb{Z}^*_{n^2}$ such that $\gcd(L(g^\lambda \bmod n^2), n) = 1$ with $L(u) = \frac{u-1}{n}$

Output: (pk, sk)

public key: $pk = (n, g)$

secret key: $sk = (p, q)$

Encryption: $\mathsf{Enc}(m, pk)$

Input: $m \in \mathbb{Z}_n$

Choose $r \in \mathbb{Z}^*_n$

Compute $c = g^m \cdot r^n \bmod n^2$

Output: $c \in \mathbb{Z}_{n^2}$

Decryption: $\mathsf{Dec}(c, sk)$

Input: $c \in \mathbb{Z}_{n^2}$

Compute $m = \frac{L(c^\lambda \bmod n^2)}{L(g^\lambda \bmod n^2)} \bmod n$

Output: $m \in \mathbb{Z}_n$

Figure 3.2: *Paillier Algorithm*

3.1.3. Multiplicative Homomorphic Encryption

This kind of cryptosystem enables homomorphic computation of only the Multiplication operation. These cryptosystem satisfy the property that the product of the cipher texts equals the cipher of the product [48].

Enc (x. y) = Enc(x).Enc(y).

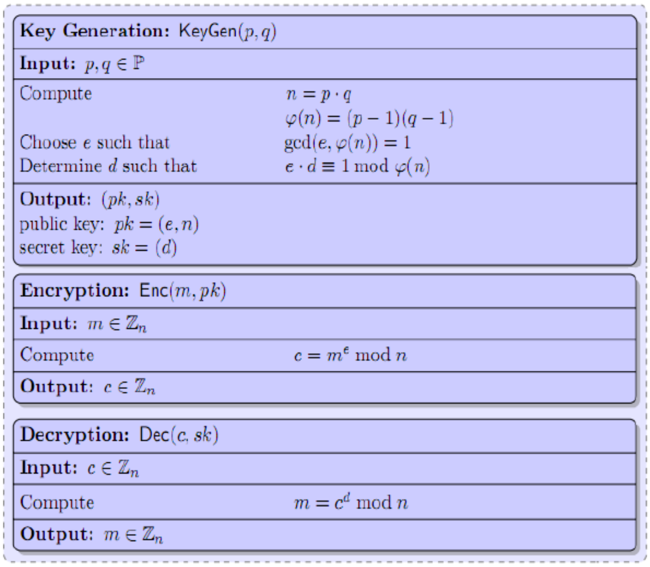
Some well-known example of multiplicative homomorphic are RSA and Elgamal Encryption cryptosystem. Some of the applications of homomorphic encryption are: Cloud computation, Electronic voting, financial transactions, Electronic cash, and Medical records [48].

3.2 RSA Encryption Cryptosystem

RSA Encryption was introduced in the year 1978 by Rivest, Shamir, and Adleman published their public-key cryptosystem [51]. RSA is an asymmetric key algorithm which is also used for encryption and decryption of the message. Due to asymmetric nature, there are two keys used in the algorithm. These keys are regarded as public key and private key [51].

The public key is available to everyone to send a message and the private key is kept secret by authorized user.

RSA scheme include the key generation, encryption and decryption procedure is described below in **Figure 3.3** [50].



Key Generation: KeyGen(p, q)

Input: $p, q \in \mathbb{P}$

Compute $n = p \cdot q$

$\varphi(n) = (p-1)(q-1)$

Choose e such that $\gcd(e, \varphi(n)) = 1$

Determine d such that $e \cdot d \equiv 1 \bmod \varphi(n)$

Output: (pk, sk)

public key: $pk = (e, n)$

secret key: $sk = (d)$

Encryption: Enc(m, pk)

Input: $m \in \mathbb{Z}_n$

Compute $c = m^e \bmod n$

Output: $c \in \mathbb{Z}_n$

Decryption: Dec(c, sk)

Input: $c \in \mathbb{Z}_n$

Compute $m = c^d \bmod n$

Output: $m \in \mathbb{Z}_n$

Figure 3.3: *RSA Algorithm.*

3.3 Elgamal Encryption Cryptosystem

Another cryptosystem created during the same period was the Elgamal Cryptosystem [53]. Elgamal algorithm was introduced in 1985 by Taher El Gamal [47]. Elgamal is an asymmetric key encryption algorithm that is based on the Diffie-Helman key exchange as an alternative to RSA for public key encryption. Elgamal is also used in digital signature generation algorithm called Elgamal signature scheme [54].

Key Generation: The key generator works as follows: Alice generates an efficient description of a cyclic group G, of order q, with generator *g* [44].

Alice chooses a random $x \in \{1, \ldots, q-1\}$.
Alice computes

$$y = g^x \tag{2.8}$$

Alice publishes y along with the description of *G; q; g*, as her public key. Alice retains x, as her private key which must be kept secret.

Encryption: The encryption algorithm works as follows: To encrypt a message *m*, to Alice under her public key (G, q, g, y), Bob chooses

a random $r \in \{1, \ldots, q-1\}$, then computes

$$c_1 = g^r \tag{2.9}$$

Bob computes the shared secret

$$s = y^r \tag{2.10}$$

Bob converts his secret message m, into an element $m' \in G$.
Bob computes

$$c_2 = m' \cdot s \tag{2.11}$$

Bob sends the ciphertext $(c_1, c_2) = (g^r, m' \cdot y^r)$ to Alice.

Decryption: The decryption algorithm works as follows:

To decrypt a cipher text (c1, c2), with her private key x, Alice computes the shared secret

$$t = c_1^x \tag{2.12}$$

and then computes

$$m' = c_2 \cdot t^{-1} \tag{2.13}$$

which she then converts back into the plaintext message m, where t^{-1} is the inverse of t in the group G (e.g., modular multiplicative inverse if G is a subgroup of a multiplicative group of integers modulo n).

The decryption algorithm produces the intended message, since

$$\begin{aligned} c_2 \cdot t^{-1} &= (m' \cdot s) \cdot c_1^{-x} \\ &= m' \cdot y^r \cdot g^{-xr} \\ &= m' \cdot g^{xr} \cdot g^{-xr} \\ &= m' \end{aligned}$$

Comparison of partial homomorphic encryption and fully homomorphic encryption cryptosystems can be described below in **Table 3.1** [47].

Table 3.3: *Partial HE vs. Fully HE.*

Partial HE	Fully HE
Allows either additive or multiplicative operations.	Allows both additive and multiplicative operations
Key used by the client (different keys are used	Key used by the client (different keys

for encryption and decryption)	are used for encryption and decryption)
Security applied to cloud provider server	Security applied to cloud provider server
Requires less computational efforts	Requires more computational efforts
Privacy of data is ensured in both communication and storage process.	Privacy of data is ensured in both communication and storage process.
Faster in performance	Slower performance
Small cipher text size	Large cipher text

IV. Related Works

This part presents a review on most recent related works, all of this researches discuss data security issues in cloud computing. We analyzed and identified gaps that exist in previous works. Finally, we summarize the works reviewed.

In [15] described that cloud computing emerges as a new computing paradigm which is used to provide reliable, customized and QoS guaranteed dynamic computing environments for users. However, adopting a cloud computing have positive and negative effects on the data security of cloud service consumers. Since the data and application is controlled by the cloud service provider. This leads to concern about data safety and its protection from internal and external threats. So this paper discusses about security of data being stored on cloud service provider and privacy while data being transmitted are the main challenging issues in cloud computing. Rijndael Encryption Algorithm have proposed to provide security of the data and EAP-CHAP for authentication purpose.

In [39] discuses that cloud computing is new and growth technology. The cloud data and services reside in massively scalable data centers and can be accessed everywhere. The growth of the cloud consumers has unfortunately been attended with a growth in malicious activity in the cloud. More and more vulnerabilities are discovered. In this paper discusses about data security and privacy issues in cloud computing. Some of the security issues like multi tenancy, data loss and leakage, easy accessibility of cloud, identity management, unsafe API's, service level agreement inconsistencies, patch management, internal threats etc. They has proposed a simple data protection model where data is encrypted using Advanced Encryption Standard (AES) before it is moved in the cloud.

In [40] have described that cloud computing is a conceptual service based technology which is used by many organizations. As different types of resources and private data are stored in the cloud, the consumer expects to protect their data by providing security and maintaining privacy. So this paper explains about data privacy protection and data retrieval control are the main challenging issues to be address in cloud computing. The proposed security solutions, is authentication and encryption for secure data transmission from one cloud service to other cloud that requires to secure and authenticated date with Elliptic curve cryptography. Most of the works have been done on data security and privacy issues in cloud computing, such as paper In [15], In [39], In [40] and have been used traditional methods of encryption (like Rijndael, AES, and ECC) to ensure storage of the data in cloud, but they cannot do any processing such us (to performed arithmetic operation, and searching…etc.) on encrypted data without giving private key to the server. This means cloud server can‘t operate on the data until it decrypts. The last one needs to decrypt data at every operation. The client will need to provide the private key to the cloud server to decrypt data before execute the calculations required and then subsequent operations can be done on the data, which might affect the confidentiality and privacy of data stored in cloud.

In [41] discuss that cloud computing is most interesting and enticing technology which is offering the services to users on demand over the internet and also many companies are using the cloud architectures. Cloud data security depends more on the procedures and count measures. So this paper discusses about the data security issues in cloud computing. Some of the issues like privacy and confidentiality, data integrity, data allocation and reallocation, data availability, storage and backup recovery etc. to overcome all these type of issues, RSA algorithm has been used. The data will be encrypted and sent to the user, when the user wants the data it will be sent in the decrypted format. In this approach stated above we analyze that the security is provided data at rest (security of the data being stored) i.e., encryption is done by cloud service provider at cloud server side and decryption is done by cloud consumer. But it leaves the data insecure while consumers outsources it to the cloud as the data travel in the plaint text form.

In [43] had described what cloud computing is and how can be benefitted from it. With the emergence of cloud computing organizations can store data in the cloud provider's data center, but the security of the data is a major concern in cloud computing as data is stored in a cloud in an encrypted format and it becomes very difficult to perform operations on the encrypted data, hence homomorphic encryption can be used to secure data and also perform operations on it. The RSA and El Gamal cryptosystem is discussed and also described how they can be used to perform calculations. RSA and El Gamal has a multiplicative homomorphic property i.e., it is possible to find the product of the plain text by multiplying the cipher texts. El Gamal is randomized, which is one of its superior positions over RSA and also, El Gamal uses a smaller key length when compared to RSA. The limitation of this paper, unpadded RSA has the homomorphic property. Unfortunately, the Unpadded RSA is semantically insecure (i.e., does not padding a random bit into a message before encryption). Even though, Elgamal encryption scheme is probabilistic, has public key exchange problem [44].

V. Proposed Solution

5.1 Proposed Architecture

In this paper, a prototype of web service for managing banking database is taken for implementing proposed system. This web service is a cloud based application which is developed using pillar homomorphic algorithm in java and Amazon RDS MySQL database. This system allow cloud service users to manage and upload the data into a cloud computing environment. Even though, the cloud based banking system database is quite huge, paillier homomorphic encryption is applied only for doing basic bank operations like credit, debit and transfer money. The application is deploy on local machine and only Amazon RDS MySQL database engine is outsourced from amazon web service public cloud (i.e., only outsource platform as a service from cloud) where database is developed and running on cloud.

The **figure 5.1** shown below contains different modules that are part of the proposed scheme. These modules are the, encryption module, OTP module and decryption module.

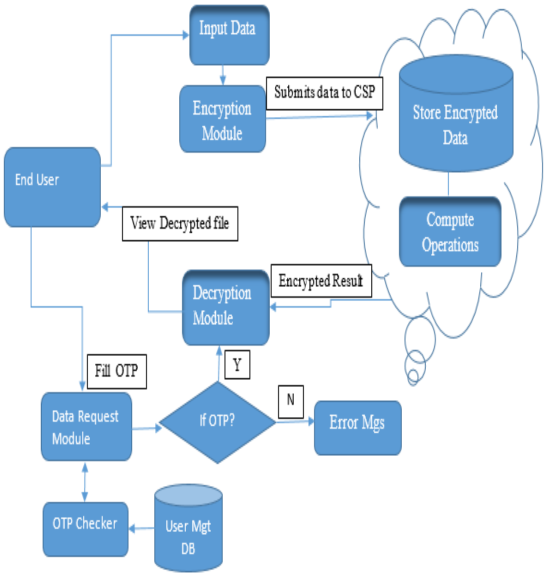


Figure 5.1. *Workflow of the proposed architecture.*

In this proposed system, the inputs are bank account number and amount of money to be transferred / credited / debited passed from client side. When the input data reaches the cloud service for doing the required operation, first the data will be fetched from the database in the encrypted format, the specified operations will be done on encrypted data with the help of homomorphic encryption algorithm. The result of this operation also will be in encrypted form that will be finally gets stored in banking database. When client need to check their balance of a particular account, the encrypted data will be send to client with OTP. The end user need to supply correct OTP in addition to secret key for decrypting the result data. After decrypting the result data, the plain data will be sent to the client. In this system Homomorphic encryption algorithm is used to do computation on the encrypted data securely, quickly and reliably. Operations such as addition can executed on the ciphered data stored in the cloud provider based on the request of the users and the encoded result also moves over the internet from back end to the front end. If the OTP is incorrect an error message is shown for the user.

5.1.1 Stepwise Procedure to Development

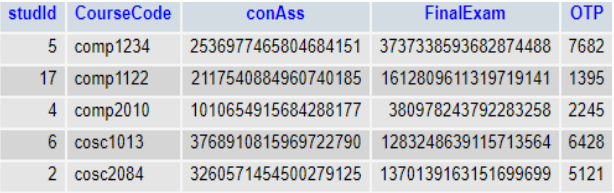
The steps involved in the development of this cloud based application to ensure the cloud data security using homomorphic encryption as follows.

1. Study and analyze different encryption techniques that can be used to ensure cloud data.
2. Design a cloud web based application that can be deployed on the cloud and can be provided functionality to secure the user data.
3. Develop cloud web based application using Java 2 enterprise edition. This can be used to collect user data which is then processed through Java servlets and also Java servlet packages.
4. Amazon RDS MySQL database engine outsourced from Amazon web service public cloud. In order to build the data tables and store information such as authentication details, user data uploaded on to the cloud, in addition to this, to operate and scale a relational database in the cloud.
5. Paillier homomorphic algorithm was implemented using the java security packages.

VI. Experimental Results

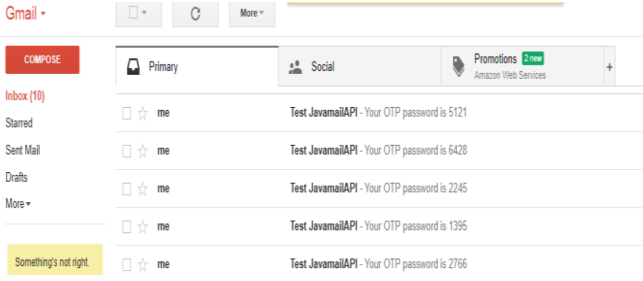
In this paper work, sample cloud based application which provide functionality for cloud service users is implemented to encrypt data on the cloud remote server. Both encryption and decryption are done in the front end using pillar homomorphic algorithm, with this technique operations such us addition can be perform on encrypted user data without decryption. The cloud remote server can operates on encrypted users data without disclose the client data privacy. Furthermore, if the cloud provider servers are hacked by malicious attackers, the user's data is secured and cannot be misused as it is homomorphic encrypted.

Sample of results:



studId	CourseCode	conAss	FinalExam	OTP
5	comp1234	2536977465804684151	3737338593682874488	7682
17	comp1122	2117540884960740185	1612809611319719141	1395
4	comp2010	1010654915684288177	380978243792283258	2245
6	cosc1013	3768910815969722790	1283248639115713564	6428
2	cosc2084	3260571454500279125	1370139163151699699	5121

Figure 6.2. *Encrypted Input Data in big Integer format.*



□ ☆ me Test JavamailAPI - Your OTP password is 5121

□ ☆ me Test JavamailAPI - Your OTP password is 6428

□ ☆ me Test JavamailAPI - Your OTP password is 2245

□ ☆ me Test JavamailAPI - Your OTP password is 1395

□ ☆ me Test JavamailAPI - Your OTP password is 2766

Figure 6.2 *Send OTP Using JavaMailAPI.*

- The major problems that any cloud service organization must overcome are to ensure that if its servers are attacked by unauthorized user, the client data cannot be stolen and changed and this was accomplished by encode the client data before transfer it to cloud organization.
- The confidential client data must stay invisible even to the cloud service organizations. This was achieved by the user private key store locally and both encryption and decryption done on the client side.
- The cloud service provider can perform operations on ciphered user's data stored in cloud without having to decrypt. This was achieved by using Homomorphic encryption.
- OTP authentication method was used for verification of identity make more secure user sensitive data. Even though the unauthorized users is able to get one credential (username, and password), still they can't ready to get access the user data in cloud provider because of the multi-authentication method used in the thesis. OTP automatically generate during service request and send to user through his Gmail.

VII. CONCLUSION

In cloud computing, data exchanged among different components like web services, database, and client system. Cloud service providers have responsibility to make secure consumers data is keep confidential in all times. Even though, cloud computing provides many benefits, there are still many security concerns that need to be resolved. Implementing Homomorphic encryption algorithm with OTP meachanism in cloud services could improve the level of security on cloud data and also reduces the computation speed since computation on data are performed without decrypting process. This leads less communication overhead with security in cloud environment. Thus, the data security and privacy can be achieved through homomorphic encryption.

VIII. REFERENCES

[1]. D. E. Robling Denning, "Cryptography and Data Security," Addison-Wesley Longman Publishing Co., Inc. 1982.

[2]. Open Data Center Alliance Inc., "Open Data Center Alliance Usage : Data Security Framework Rev 1 . 0," pp. 1–21, 2013.

[3]. R. Kaur and P. K. Pateriya, "A Study on Security Requirements in Different Cloud Frameworks," Int. J. Soft Comput. Eng., vol. 3, no. 1, pp. 133–136, 2013.

[4]. R. Maheshwari and S. Pathak, "A Proposed Secure Framework for Safe Data Transmission in Private Cloud," Int. J. Recent Technol. Eng., vol. 1, no. 1, pp. 78–82, 2012.

[5]. P. Mell and T. Grance, "The NIST Definition of Cloud Computing Recommendations of the National Institute of Standards and Technology," Nist Spec. Publ., vol. 145, p. 7, 2011.

[6]. A. Bucur, "Banking 2.0: Developing a Reference Architecture for Financial Services in The Cloud," 2011.

[7]. T. Dillon, C. Wu, and E. Chang, “Cloud Computing: Issues and Challenges,” 2010 24th IEEE Int. Conf. Adv. Inf. Netw. Appl., pp. 27–33, 2010.

[8]. “Security Guidance for Critical Areas of Focus in Cloud,” pp. 0–176, 2011.

[9]. N. Jose and C. K. A, “Data Security Model Enhancement in Cloud Environment,” vol. 10, no. 2, pp. 1–6, 2013.

[10]. D. Chen and H. Zhao, “Data Security and Privacy Protection Issues in Cloud Computing,” 2012 Int. Conf. Comput. Sci. Electron. Eng., vol. 1, no. 973, pp. 647–651, 2012.

[11]. R. V. Rao and K. Selvamani, “Data Security Challenges and Its Solutions in Cloud Computing,” Procedia Comput. Sci., vol. 48, pp. 204–209, 2015.

[12]. H. Patel and J. Jha, “Securing Data in Cloud Using Homomorphic Encryption,” Int. J. Sci. Res., vol. 4, no. 6, pp. 1892–1895, 2015.

[13]. D. Patil, R. Bhavsar, and A. Thorve, “Data Security over Cloud,” Int. J. Comput. Appl., pp. 11–14, 2012.

[14]. S. Singla and J. Singh, “Implementing Cloud Data Security by Encryption using Rijndael Algorithm,” Glob. J. Comput. Sci. Technol., vol. 13, no. 4, pp. 0–4, 2013.

[15]. I. Mouhib and E. L. O. Driss, “Enhanced Data Security Approach for Cloud Environment Based on Various Encryption Techniques,” vol. 80, no. 3, 2015.

[16]. M. Tebaa and S. E. L. Hajii, “Secure Cloud Computing through Homomorphic Encryption,” pp. 29–38, 2013.

[17]. S. S. Gaikwad and A. R. Buchade, “Homomorphic Encryption Approach For Cloud Data Security,” pp. 105–111, 2016.

[18]. V. Sravan and K. Maddineni, “Security Techniques for Protecting Data in Cloud Computing,” 2011.

[19]. L. M. Vaquero, L. Rodero-Merino, J. Caceres, and M. Lindner, “A break in the clouds,” ACM SIGCOMM Comput. Commun. Rev., vol. 39, no. 1, p. 50, 2009.

[20]. M. B. Mollah, K. R. Islam, and S. S. Islam, “Next generation of computing through cloud computing technology,” 2012 25th IEEE Can. Conf. Electr. Comput. Eng. Vis. a Greener Futur. CCECE 2012, pp. 1–6, 2012.

[21]. M. Amziani, T. Melliti, and S. Tata, “A generic framework for service-based business process elasticity in the cloud,” 2013.

[22]. European Network and Information Security Agency (ENISA), “Cloud Computing: Benefits, Risks and Recommendations for Information Security,” Computing, vol. 72, no. 1, pp. 1–50, 2012.

[23]. M. Y. Pandith, “Data security and privacy concerns in cloud computing,” Internet Things Cloud Comput., vol. 2, no. 2, pp. 6–11, 2014.

[24]. S. Eludiora, O. Abiona, A. Oluwatope, and A. Oluwaranti, “A User Identity Management Protocol for Cloud Computing Paradigm,” Int. J. Commun. Netw. Syst. Sci., vol. 4, no. 3, pp. 152–163, 2011.

[25]. B. G. Garrison, S. Kim, and R. L. Wakefield, “Success Factors for Deploying Cloud,” Commun. ACM, vol. 55, no. 9, 2012.

[26]. J. Varia and S. Mathew, “Overview of Amazon Web Services,” Amaz. Web Serv., p. 22, 2014.

[27]. “Microsoft Azure: Cloud Computing Platform &amp; Services.” [Online]. Available: https://azure.microsoft.com/en-us/. [Accessed: 05-May-2017].

[28]. S. Goyal, “A comparative study of cloud computing service providers,” Int. J. Adv. Res. Comput. Sci. Softw. Eng., vol. 2, no. 2, 2012.

[29]. D. Chappell, “Introducing the Windows Azure Platform,” vol. 26, no. 4, pp. 56–70, 2011.

[30]. R. Ashalatha and M. Vaidehi, “The Significance of Data Security in Cloud: A Survey on Challenges and Solutions on Data Security,” pp. 15–18, 2012.

[31]. T. Andrei, “Cloud Computing Challenges and Related Security Issues,” A Surv. Pap., pp. 1–10, 2009.

[32]. A. O. Kuyoro, S.O, Ibikunle F., “Cloud Computing Security Issues and Challenges,” Int. J. Comput. Networks, no. 3, pp. 247–255, 2011.

[33]. S. M. Hashim, “Security and Authentication and Access on Data Transfer under the Cloud computing by using key,” Int. J. Adv. Comput. Sci. Technol., vol. 4, no. 6, pp. 4–7, 2015.

[34]. “What is confidentiality, integrity, and availability (CIA triad)?” [Online]. Available: http://whatis.techtarget.com/definition/Confidentiality-integrity-and-availability-CIA. [Accessed: 05-May-2017].

[35]. V. Hemamalini, G. Zayaraz, V. Susmitha, M. Gayathri, and M. Dhanam, “A Survey on Elementary , Symmetric and Asymmetric Key Cryptographic Techniques,” Int. J. Adv. Comput. Sci. Appl., vol. 5, no. 1, pp. 11–26, 2016.

[36]. J. Katz and Y. Lindell, Introduction to Modern Cryptography. 2008.

[37]. D. Okunbor and C. Sarami, “Homomorphic Encryption: A Survey Abstract,” vol. 14, no. 1, pp. 64–69, 2017.

[38]. A. Sachdev, “Enhancing Cloud Computing Security using AES Algorithm,” vol. 67, no. 9, pp. 19–23, 2013.

[39]. C. Engineering, “Data Security in Cloud Using Elliptic Curv[1] C. Engineering, ‘Data Security in Cloud Using Elliptic Curve,’ no. March 2009, pp. 4187–4192, 2014.e,” no. March 2009, pp. 4187–4192, 2014.

[40]. P. Kalpana, “Data Security in Cloud Computing using RSA Algorithm,” Int. J. Res. Comput. Commun. Technol., vol. 1, no. 4, pp. 143–146, 2012.

[41]. R. Alattas, “Cloud Computing Algebraic Homomorphic Encryption Scheme,” Int. J.Innov. Sci. Res., vol. 8, no. 2, pp. 191–195, 2014.

[42]. S. Ravindran and P. Kalpana, “Data storage security using partially homomorphic encryption in a cloud,” Int. J. Adv. Res. Comput. Sci. Softw. Eng., vol. 3, no. 4, pp. 603–606, 2013.

[43]. E. Yi, Xun and Paulet, Russell and Bertino, Homomorphic Encryption and Applications, vol. 3. 2014.

[44]. S. Ramachandram, R. Sridevi, and P. Srivani, “A Survey Report On Partially Homomorphic Encryption Techniques In Cloud Computing,” vol. 2, no. 12, pp. 3278–3287, 2013.

[45]. M. Ogburn, C. Turner, and P. Dahal, “Homomorphic encryption,” Procedia Comput. Sci., vol. 20, pp. 502–509, 2013.

[46]. Y. Bensitel and R. Romadi, “Secure Data in Cloud Computing Using Homomorphic Encryption,” vol. 82, no. 2, pp. 206–211, 2015.

[47]. D. Hrestak and S. Picek, “Homomorphic Encryption in the Cloud,” Inf. Commun. Technol. Electron. Microelectron., no. 2, pp. 1–5, 2014

[48]. N. Jain, S. K. Pal, and D. K. Upadhyay, “Implementation and Analysis of Homomorphic Encryption Schemes,” Int. J. Cryptogr. Inf. Secur., vol. 2, no. 2, 2012.

[49]. K. Benzekki, A. El Fergougui, A. El, and B. El, “A Secure Cloud Computing Architecture Using Homomorphic Encryption,” vol. 7, no. 2, pp. 293–298, 2016.

[50]. L. A. Rivest, Ronald L., Adi Shamir, “A method for obtaining digital signatures and public-key cryptosystems,” Commun. ACM, vol. 12, no. 2, 1978.

[51]. M. K. Mohanty, “Secure Data Storage on the Cloud using Homomorphic Encryption.,” PhD diss., 2013.

[52]. T. Elgamal, “A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms,” IEEE Trans. Inf. theory, vol. 31, no. 4, pp. 469–472, 1985.

[53]. M. M. Ali, “Cryptography : A Comparative Analysis for Modern Techniques,” vol. 8, no. 6, pp. 442–448, 2017.

Author Profile

Kedir Salih received the BS Degree in Computer Science from Dilla University in 2013 GC and M.Sc. Degree in Compuer Science and Networking in Dilla University. His research area includes Wireless Communicaiton, Network Security & Cloud Computing.

Cite this article as :

Kedir Salih Siraj, "Ensuring Data Security in Cloud Computing using Homomorphic Encryption", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 6 Issue 2, pp. 379-391, March-April 2020. Journal URL : http://ijsrcseit.com/CSEIT206296