

HOMOMORPHIC ENCRYPTION IMPLEMENTATION TO ENSURE DATA SECURITY IN CLOUD COMPUTING

¹ MOHAMAD T. SULTAN, ² KHALED N. YASEN

^{1,2} Department of Computer Science, Cihan University-Erbil, Erbil 44001, Iraq

E-mail: ¹ taha@engineer.com, ² khalid_aldabbagh@cihanuniversity.edu.iq

ABSTRACT

Cloud computing is an evolving technology paradigm with incredible momentum. While the benefits of cloud computing are clear, it introduces new security challenges. Security issues in cloud computing is shown to be the biggest obstacle that could affect its wide benefits. These security issues become a critical concern when outsourcing sensitive data and business application to a third party. Sensitive information such as medical records, financial records or high impact business data has its own strict and sensitive security requirements for confidentiality, availability to authorized users, and traceability of access. This paper focuses on storing data on the cloud in the encrypted format to avoid data contact from an unauthorized access. The paper introduces the principle of homomorphic encryption and suggests the use of fully homomorphic encryption (FHE) scheme to provide a secure environment for sensitive user data in the cloud. The proposed scheme is secure and practical. And the performance assessment and analysis demonstrate the practice and validity of the proposed technique.

Keywords: *Cloud Computing, Homomorphic Encryption, Cloud Security, Security Threats.*

1. INTRODUCTION

Over the past few years, cloud computing has rapidly emerged as a widely accepted computing paradigm as it's expected to play a major role in the future Internet of Services, enabling on-demand provisioning of applications, platforms, and computing infrastructures. Cloud computing provides clients with a virtual computing infrastructure on top of which they can store data and run applications [1][2]. The main core concepts that the cloud computing paradigm has been built around include on demand computing resources, elastic scaling, lower operational expenses, and promoting a pay-per-use business model for information technology and computing services. Furthermore, cloud computing could significantly enhance collaboration, proficiency, and scale, thus enabling a truly global computing model over the internet infrastructure [2].

Cloud computing helps users to access internet resources online from anywhere at any time [1]. This can be achieved without worrying about technical or physical maintenance and management problems of the original resources. However, the concept of cloud computing technology remains for many as a vague idea. Therefore, if cloud computing is to achieve its potential then there is need to have a clear understanding of the factors

that can influence its adoption in different organizations. Since many organizations realize that cloud computing will quickly become a key component of long-term IT strategies with less costs and reduced time-consuming efforts [3]. They become more interested in shifting their systems to the cloud and identify more ways to leverage those systems in real time to ensure the vast benefits of the cloud computing paradigm. This helps those organizations to access cloud-based systems from any location and it facilitates better collaboration among different organizations. However due to the lack of effective privacy and security techniques many organizations have been slow to jump on the cloud approach [4]. Cloud operators are expected to manipulate client data without necessarily being fully trusted. Several surveys of potential cloud adopters indicate that security and privacy is the primary concern hindering its adoption [4]. Customer sensitive data requires high levels of security measures due to the sensitivity of such data where most patients for example in the healthcare sector usually prefer their data to be private and secure. If there is a lack of security measures in protecting customers sensitive data it may affect their willingness to disclose necessary health information in the future. Without appropriate security and privacy solutions designed for clouds, this potentially revolutionizing computing paradigm

could become a huge failure. A critical challenge for the cloud computing industry is to ensure privacy and security of customer sensitive data, including information in electronic health records (HER), financial records in banks or high impact business data. This requires adequate privacy and security measures to be undertaken. The technique that makes this magic solution possible is called fully homomorphic encryption, or FHE. Fully homomorphic encryption scheme has been suggested in this work. And if the encryption scheme is homomorphic, the cloud can still perform meaningful computations on the data, even though it is encrypted. Therefore, a practical simple fully homomorphic encryption scheme, derived from Gentry cryptosystem is proposed to ensure the privacy preserving in cloud servers, in which encrypted data can be operated on instantly without touching the confidentiality of the stored data. The rest of this paper firstly introduces the concept of cloud computing and its characteristics, then it introduces the security issues of cloud computing and finally it suggests a corresponding security mechanism to overcome and mitigate these security issues.

2. WHAT IS CLOUD COMPUTING

Cloud computing plays an increasingly important role in IT infrastructure, and IT professionals need to be aware of fundamental cloud principles and techniques. Cloud computing, or “the cloud”, has become a leading trend in IT [1][3]. However, its definition is ambiguous and some of the terminology related to it is confusing. Cloud computing is basically an umbrella expression refers to Internet based development and services. It supplies a shared pool of configurable IT resources such as network, software and database. It is best to think of it as being an abstract concept that encapsulates techniques used to provide computing services from a pool of shared resources. Cloud computing is a new generation of technology which is designed to offer the possible necessities, resolve the IT management issues, and run the suitable applications. Several properties define cloud data, infrastructure and applications services which includes remotely hosted services, Ubiquitous which means services are available from anyplace and finally Commodified that means “You pay for what you would like to use”[1]. Cloud solutions abstract the physical hardware and present them as virtualized resources to be used for processing, memory, storage, and networking. Many cloud solutions add further layers of abstraction to define specific services that can be

provisioned and used. The main aim of cloud computing is to make a better use of distributed resources and achieve higher performance in the network [2].

Regardless of the specific technologies that organizations use to implement cloud computing solutions, the National Institute of Science and Technology (NIST) [1], the official US based standards and technology definitions body, has identified five essential characteristics that are part of a cloud computing solution.

- On-demand self-service: The ability to allocate required resources by the end user themselves as needed without involvement from the cloud service provider.
- Broad network access: Being accessible via standard network access mechanisms, without the need for any specialized infrastructure.
- Resource pooling: The pooling of the various resources to be allocated from and returned to as needed.
- Rapid elasticity: The ability to scale up and scale down as required, whether automatically or manually, without lead times being required.
- Measured service: The ability to measure exactly what resources are being used, to monitor and control those services and to be able to present that data to the service provider or end user.

A. Cloud Computing Architecture

The main architecture of the cloud consists of three basic layers which are the application, platform and infrastructure [5]. These three layers are basic service elements in any communication between the client and server. Figure 1 presents the main types of cloud computing services.

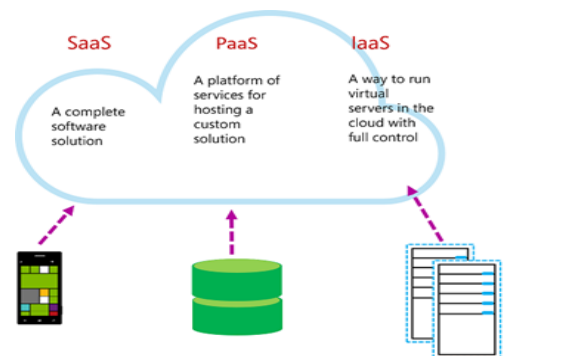


Figure 1: Cloud computing service types

Cloud computing offers three service models. A cloud application offers "Software as a Service (SaaS)" over the internet, therefore eliminating the need to install and execute the application on the user's platform [6]. SaaS offerings consist of fully-formed software applications that are delivered as cloud-based services. In SaaS, the cloud providers enable and provide application software as on-demand services. An example of this model is the Google Apps which is the most widely used SaaS services. Users can subscribe to the service and use the application, normally through a web browser or by installing a client-side app. The primary advantage of SaaS services is that they enable users to easily access applications without the need to install and maintain them [5][6]. Figure 2 presents the cloud computing service and deployment models.

The second model is the Platform services "Platform as a Service (PaaS)". This model enables programming environments to access and utilize additional application building blocks [6][7]. Typically, PaaS encapsulates fundamental operating system (OS) capabilities, including storage and compute, in addition to functional services for custom applications. In other words the clients are provided platforms access, which enables them to put their own customized software's and other applications on the clouds. It has all the application typically required by the client deployed on it. Thus the client need not go through the hassles of buying and installing the software and hardware required for it. Windows Azure Cloud Services, OrangeScape, Heroku and Ielastic are examples of PaaS.

In Infrastructure as a Service (IaaS), the cloud provider supplies a set of virtualized infrastructural components such as virtual machines which means it provides the required infrastructure as a service [1][7]. Rent processing, storage, network capacity, and other basic computing resources are granted, enables consumers to manage the operating systems, applications, storage, and network connectivity. Typically, IaaS facilities are managed in a similar way to on-premises infrastructure, and provide an easy migration path for moving existing applications to the cloud. Amazon, HP Cloud, Linode, Rackspace, Google Compute Engine and Ready Space Cloud Services are main examples of IaaS.

B. Deployment of Cloud Computing Service

There are mainly four basic deployment models, which explain the structure of where cloud services and applications can be deployed for consumers to use. For a successful deploying of cloud computing solution, the major factor is to make a decision on the type of cloud to be installed. These are; public cloud, private cloud, hybrid cloud and community cloud [1][2][6].

Public cloud is an open model allows the user to access the cloud using interfaces via web browsers. A public cloud is owned by the cloud service provider also known as a hosting provider. The cloud service provider provides cloud resources for an organization, which the end user connects to via a secure network connection, typically over the internet.

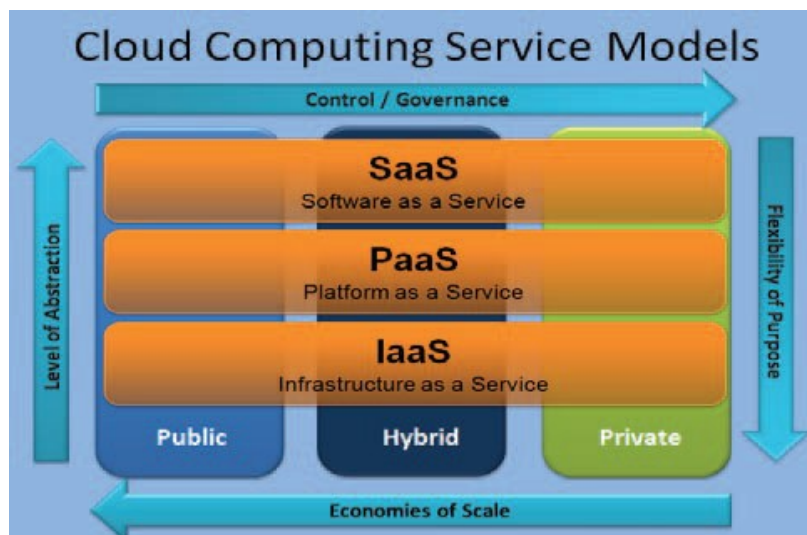


Figure 2: Cloud computing service and deployment models

This includes applications and storage which are rendered over a network. Users usually have to pay only for the time duration they use the service on pay per use basis. The physical infrastructure owned and managed by the service provider. The advantage of the public cloud is that it's cost effective which helps in reducing the operation costs on IT payments. However, it suffers several security issues compared to other cloud models since all the applications and data on the public cloud are more prone to malicious attacks [4]. However, these issues can be mitigated by taking several security checks to be implemented. A private cloud on other hand is owned and managed by the organization. A private cloud operates only within one organization on a private network and is highly secure [8]. It offers many of the benefits of a public cloud computing environment, but it is appropriate for a single organization and should be managed within an organization. It provides cloud functionality to external customers or specific internal departments, such as Accounting or Human Resources. Security is enhanced here as only the organizations users have access to the private cloud. It's also easier to manage maintenance and upgrades and moreover provides more control over the deployment and use [6][8]. The private cloud can actually be handled by a third-party provider. This is a dedicated environment that is internally designed but externally hosted and externally managed. It combines the benefits of controlling the service and architectural design with the benefits of outsourcing.

A hybrid cloud is a combination of a public and private cloud or composition of two or more clouds which can be public or private [8]. In this model a private cloud is linked to one or more external cloud services typically, non-critical information is outsourced to the public cloud, while business critical services and data are kept within the control of the organization. It is more secure way to control data and applications and allows the party to access information over the internet. In such scenario a hybrid cloud would suits as it can store and manage sensitive customer data in its own private cloud while offering services to customers through a public cloud and gaining the benefits of doing so for the business.

Finally, the other type which is the community cloud, where in this deployment model the cloud infrastructure is used in shared concerns manner. Which mean this model is provisioned for exclusive use by a specific community of organization jointly construct and share a cloud

infrastructure, security requirements and policies. The cloud maybe owned, managed, and operated by one or more of the organizations in the community, a third party, or some combination of these, and it may also exist on or off premises [3][8].

3. SECURITY THREATS IN CLOUD COMPUTING

Cloud Computing represents one of the most considerable shifts in information technology and many users are likely to see it in daily lifetime. For both cloud service providers and cloud service customers the security threats in cloud computing are important aspects. The issues of privacy, security and trust of cloud remain areas of concern, interest and uncertainty, despite the tremendous potential and rapid growth of this technology.

Table 1. The security domains in cloud computing

Threats usually are related information security because of data and applications. Cloud computing service has wide variety of threats because of being combination of several technologies. The resulted risks are needed to be better understood. A risk management process must be used to balance the advantages of cloud computing with the associated security risks. According to reports of CSA (Cloud Security Alliance), the 13 security domains [9], [11] and the 7 top threats [10],[11] on cloud computing were defined as shown in Table 1 and Table 2.

Domain No.	Domain description
1	Cloud Computing Architectural Framework
2	Governance and Enterprise Risk Management
3	Legal and Electronic Discovery
4	Compliance and Audit
5	Information Lifecycle Management
6	Portability and Interoperability
7	Traditional Security, Business Continuity, and Disaster Recovery
8	Data Center Operations
9	Incident Response, Notification, and Remediation
10	Application Security
11	Encryption and Key Management
12	Identity and Access Management
13	Virtualization

Threat No.	Threat description
1	Abuse and Nefarious Use of Cloud Computing
2	Insecure Application Programming Interfaces
3	Malicious Insiders
4	Shared Technology Issues
5	Data Loss or Leakage
6	Account, Service & Traffic Hijacking
7	Unknown Risk Profile

The figure below shows the associated 7 security threats and other related security domains in relation to the service models of cloud computing.

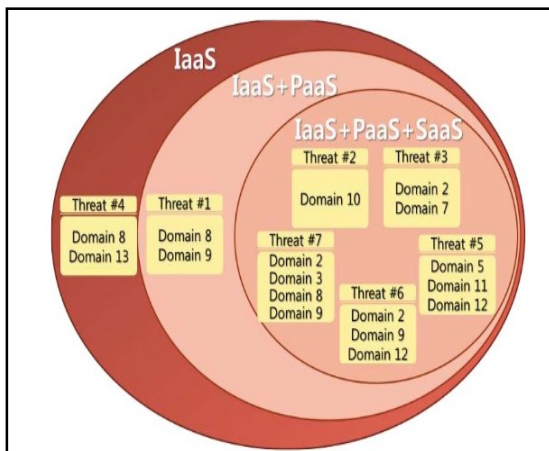


Figure 3: Security Threats, Domains and Related Service Models

4. SENSITIVE DATA SECURITY IN CLOUD COMPUTING

Providing a third party with data and business sensitive information requires a lot of trust. Typically, businesses take time to build up a relationship with cloud providers and evaluate their trustworthiness and ability to deliver what they promise. At the moment the current cloud computing techniques for different types of cloud (Private, Public and Hybrid) have numerous features to guarantee the general security of any data and system uses the cloud. The mechanisms of the transmission of data between the local devices and cloud computing data centers must ensure secure and safe communications [4].

As part of the security risk assessment, a privacy on other hand needs to be considered to

ascertain potential risks to the data and operations in the cloud. When the data transmitted to the cloud, the standard encryption techniques are used to protect the operations and the storage of the data. However, there are some problem arises when there is a need to perform some computations on that data stored in the cloud which definitely requires decrypting the data first before being able to operate on it. The current data mining and other data analysis techniques for the encrypted databases is a far distant thing to achieve using different available encrypting standards. The proposal here is to encrypt data before the data is being sent to the cloud providers. Thus, to allow a cloud computing provider to carry out different computations on customers data at their request, such as processing sales patterns, without revealing the raw data or being able to see its contents. This section and the subsequent sections elaborates the application of Fully-Homomorphic encryption technique to the security of sensitive data in the cloud computing environment.

A. Research Questions

There are two major questions that present a challenge to cloud computing providers for designing a trusted cloud solution which are:

1. How to design a cloud computing scheme that can fulfil the existing security and privacy concerns?
2. How to keep the client private information confidential, to assist in increasing the trust on cloud computing to be adopted in critical information industries?

B. History of the Homomorphic encryption

In the year of 1978 Ronald Rivest, Leonard Adleman and Michael Dertouzos proposed for the first time the initial idea of Homomorphic encryption shortly after the invention of the RSA cryptosystem [13]. However, a little advance has been made for almost 30 years. Despite the optimism of Rivest, Adleman, and Dertouzos fully homomorphic encryption remained out of reach. The encryption concept which is been proposed by Shafi Goldwasser and Silvio Micali in 1982 was a attestable security encryption system which touched a remarkable level of safety, it was an additive Homomorphic encryption, however that main problem was it can only encrypt a single bit [13]. In the year of 1999 Pascal Paillier has also developed another encryption concept that was working on the additive Homomorphic encryption technique. Few

years later, in the year of 2005, Dan Boneh, Eu-Jin Goh and Kobi Nissim [12] invented a system of provable security encryption, with which we can perform an unlimited number of additions but only one multiplication.

C. Implementing Fully-Homomorphic Encryption (FHE)

The implementation of FHE has been viewed for years as a fantasy that would never come true. However, in a breakthrough work by Craig Gentry of IBM has been emerged [13][15]. He has developed an innovation discovery by constructing a fully-homomorphic encryption scheme (FHE). Craig was the first person who got it right and figured out how to make the math work. Craig Gentry’s scheme consist of 4 phases. This include the key generation algorithm, encryption algorithm, decryption algorithm and additional Evaluation algorithm. The fully homomorphic encryption (FHE) has two essential homomorphism categories which are the multiply homomorphic encryption algorithm and additively homomorphic encryption algorithm. This has solved the issue because before the year of 2009 the Homomorphic algorithm used to only support addition homomorphism and multiplication homomorphism. The FHE can evaluate a random number of additions and multiplications on encrypted data and therefore calculate any type of function on this encrypted data. It’s possible to encrypt x and y, and apply sequence of processes to the ciphertext values, then decrypt the result to come up with the final desired answer. For plaintexts P1 and P2 and corresponding ciphertext C1 and C2, a homomorphic encryption scheme permits meaningful computation of P1 ⊕ P2 from C1 and C2 without revealing P1 or P2. That’s why Gentry’s system allows encrypted data to be analyzed and processed in the cloud.

In this example, we wish to add 1 and 2. The data is encrypted so that 1 becomes 33 and 2 becomes 54. The encrypted data is sent to the cloud and processed; the result (87) can be downloaded from the cloud and decrypted to generate the final answer (3).



The symmetric homomorphic encrypt scheme:

Select encrypt parameter: r, p and q, $r \sim 2n$, $p \sim 2n^2$, $q \sim 2n^5$ and p is prime
P is the secret key
Encrypt: for plain text m
Compute $c = pq + 2r + m$ where c is the cipher text
Decrypt: $m = (c \text{ mod } p) \text{ mod } 2$

Correctness: because pq is larger than $2r + m$ so $(c \text{ mod } p) = 2r + m$

Finally $(c \text{ mod } p) \text{ mod } 2 = (2r + m) \text{ mod } 2 = m$

Homomorphic: for two cipher text

$$C1 = q1p + 2r1 + m1$$

$$C2 = q2p + 2r2 + m2$$

Compute:

$$C1 + C2 = (q1 + q2)p + 2(r1 + r2) + m1 + m2$$

So if $2(r1 + r2) + m1 + m2 \ll p$

$$\text{Then } (c1 + c2) \text{ mod } p = 2(r1 + r2) + m1 + m2$$

So its additive homomorphic.

$$\text{And } c1 * c2 = [q1 * q2p + (2r1 + m1) + (2r2 + m2)]p + 2(r1r2 + r1m1 + r2m1) + m1m2$$

So if $2(r1r2 + r1m1 + r2m1) + m1m2 \ll p$

Then

$$(c1 * c2) \text{ mod } p = 2(r1r2 + r1m1 + r2m1) + m1m2$$

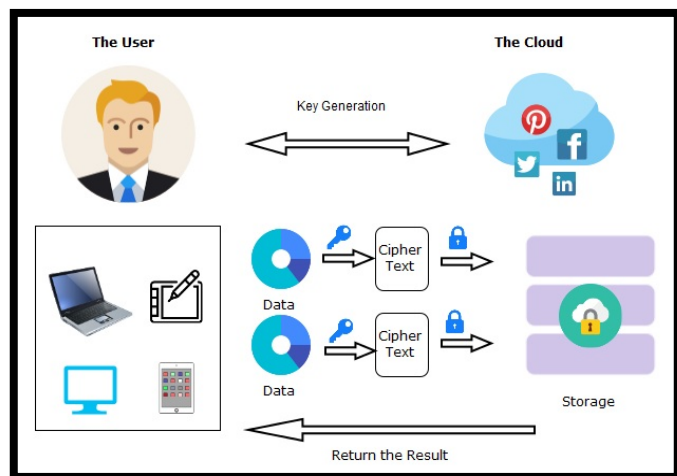


Figure 4. The Proposed Fully-Homomorphic Encryption Scheme for Cloud Computing

A homomorphic encryption (HE) scheme encrypts data in such a way that computations can be performed on the encrypted data without knowing the secret key. So, given two encryptions $c1=Ep_k(m1)$ and $c2=Ep_k(m2)$ of messages $m1$ and $m2$ under public key $pk1$, a HE scheme allows anyone to compute an encryption $Ep_k(m1 \otimes m2)$ without needing to decrypt either $c1$ or $c2$. Here \otimes denotes some arbitrary operation.

D. Related Work

Recent traditional encryption algorithms like RC4, RC6, MARS, AES, DES and 3DES still play the significant role in data security and privacy. The evaluation and analysis has been conducted for those encryption algorithms according to randomness testing by using NIST statistical testing in cloud computing environment (Amazon EC2) [22]. From the conducted experiments and acquired simulation results, the authors presented that those traditional encryption algorithms are more suitable for traditional PC environment but not very effective in achieving a robust security when applied in cloud computing environments. The authors in [23] have proposed Elliptic Curve Cryptography to explore data confidentiality and authentication in the cloud environment. In Elliptic curve cryptography [ECC] every user has a public and a private key in which public key is used for encryption and signature verification while private key is used for decryption and signature generation. Elliptic curves are used as a modification and improvement to other current cryptosystems, such as Elliptic Curve Diffie-Hellman Key Exchange and Elliptic Curve Digital Signature Algorithm.

Many researchers proposed the variants of Gentry's model with sound improvement. Homomorphic encryption on smaller size cipher text is proposed by [18] Smart and Vercauteren. The proposed cryptosystem operational problems with larger size of cipher text. Another encryption scheme, based on Residue Number System (RNS), was proposed to enhance the security [24]. In HORNS scheme, a secret is split into multiple shares on which computations can be conducted separately. Efficiency is attained via the use of smaller shares. HORNS scheme depends on the RNS property that creates multiple shares of a data and the operations on these shares are homomorphic. The proposed scheme helps in preventing the independent clouds to collude. However the system suffered complexity and more

enhancements are required on the secret split shares. The authors in [19] have proposed arithmetic operations over integers. Divya et al [27] have proposed a design to allow end users to audit the cloud storage devices having incredibly light in weight verbal exchanges along with calculation price tag. Suganya and Damodharan [21] proposed a system to check the cloud storage using the really light-weight connection. It supplies durable cloud storage reliability and also provides for more fault position data. The Proposed system simplifies extended secure and also useful energetic actions.

E. The Proposed FHE Scheme

The designed scheme in figure 4 could explain how the implementation of Fully-Homomorphic Encryption on the cloud works. The proposed scheme overcomes the data confidentiality and integrity concerns for using cloud storage services as well as for ensuring the delivery of trusted services to the clients by considering their data security policies.

The initial stage happens when a key is generated between the user of the cloud and the cloud computing provider. Both parties firstly negotiate a key and this key must be secret and generated based on a seed value. The user is the only party who knows the key and can use it to encrypt the data in prior of sending it to the cloud. This technique helps to ensure a safe passage of user's sensitive data to the cloud and in the same time it provides safe storage of data on the cloud. Even though the cloud providers handle user's data they still can't access its contents because it's encrypted. While transmitting data, it's also possible to support it with other cryptographic techniques such as digital signature to assure the integrity and nonrepudiation. When the user of the cloud sends a message to the server requesting a specific kind of operation on their data, the encrypted data can be operated on directly without affecting its confidentiality. The server or any authorized third party can process ciphertext data directly, instead of the original data. Then the cloud server returns the encrypted results to the client. The client at the final stage will use the secret key to decrypt the data and get meaningful results. Using cryptographic techniques like the proposed homomorphic encryption, it's possible to provide security and privacy without sacrificing efficiency and utility and there are no limitations on what manipulations can be performed. This provides an

improved as well as enhanced solution for designing as well as developing confidentiality and integrity preserved secure model to use cloud computing services.

F. Simulation Results

The simulation is being done using AWS DynamoDB [17], whereby the user through Eclipse IDE for Java EE Developers can connect to AWS DynamoDB.

The user will be granted a login credentials which can be used to perform operations on their data based on requirements. Once user complete the required tasks, he can choose to exit the system. The steps performed for this implementation is presented as follows:

- Step 1: Create a DynamoDB request on AWS
- Step 2: Create Database Tables
- Step 3: Get the privileges from AWS to perform access controls
- Step 4: Install Eclipse Kepler along with Java SDK
- Step 5: The user now has the required tools after the compiling AWS SDK on Eclipse framework
- Step 6: Implement the guidelines of AWS SDK according to AWS SDK standards [16].
- Step 7: Run the Java code that is built to interact with the DynamoDB
- Step 8: quit the Java code after implementation

The built Java code runs on this eclipse platform. All operations Addition, Subtraction, or check balance in the database is performed using this platform. The authorized user logs in to the system using the interface and credentials provided and then performs all required tasks. Figure 5 shows installation of Java SDK on Eclipse Kepler.

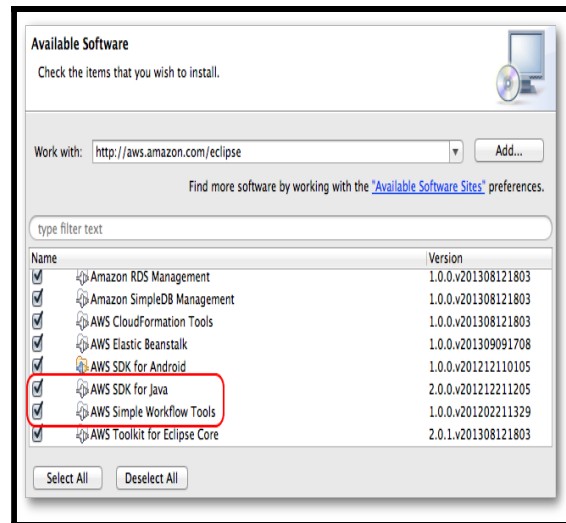


Figure 5. Screenshot of installing Java SDK on Eclipse Kepler

After the installation is complete, then we create database tables on the DynamoDB. The data is presented as balance which is stored using Homomorphic encryption structure. The authorized user is allowed to perform addition and subtraction on this encrypted balance. Then the User is capable of checking balance in the plaintext. Figure 6 shows a DynamoDB.

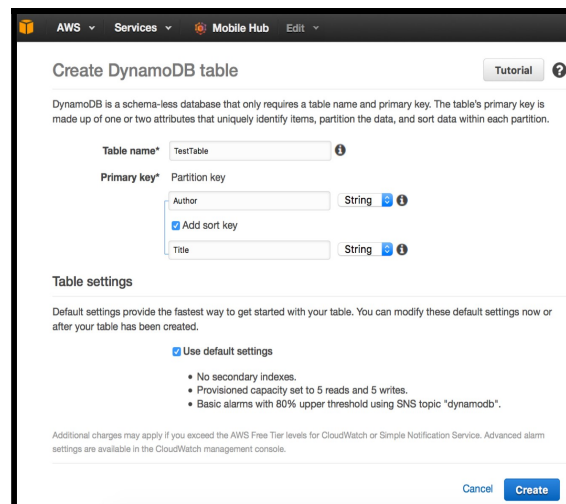
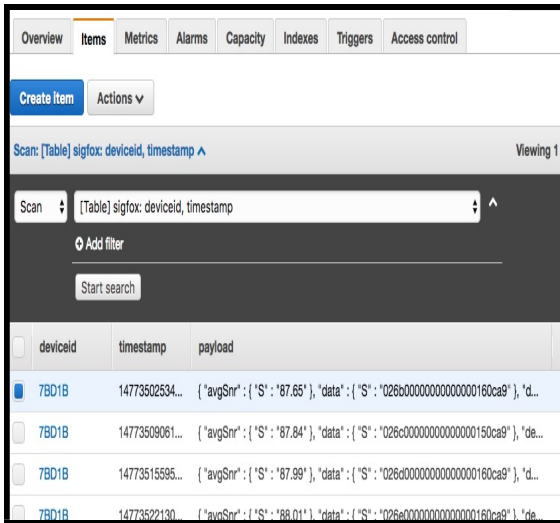


Figure 6: Database on DynamoDB

Then the built Java code is executed at client side machine and performed several operations. Finally, the data is updated back to AWS DynamoDB table. Figure 7 shows a sample Data in DynamoDB after update.



deviceid	timestamp	payload
7BD1B	14773502634...	{'avgSnr': {'S': '87.65'}, 'data': {'S': '026c000000000000000000160ca9'}, 'de...
7BD1B	14773509061...	{'avgSnr': {'S': '87.84'}, 'data': {'S': '026c000000000000000000150ca9'}, 'de...
7BD1B	14773515595...	{'avgSnr': {'S': '87.99'}, 'data': {'S': '026c000000000000000000160ca9'}, 'd...
7BD1B	14773522130...	{'avgSnr': {'S': '88.01'}, 'data': {'S': '026c000000000000000000160ca9'}, 'de...

Figure 7: Sample of updated Data in DynamoDB table

G. Performance Assessment

The assessment is being decided on the criteria of time taken by a specific algorithm to perform encryption and decryption for an input file. Then the computation time will be calculated for both encryption and decryption process.

First we took into consideration the encryption process. Thus, the time consumed by the algorithm to generate the cipher text from the plain text is considered the encryption computation period. Figure 8 shows the encryption time taken by different algorithms (ex: AES, RSA, RC6 and the Proposed homomorphic design) to calculate the encrypted data of an input file.

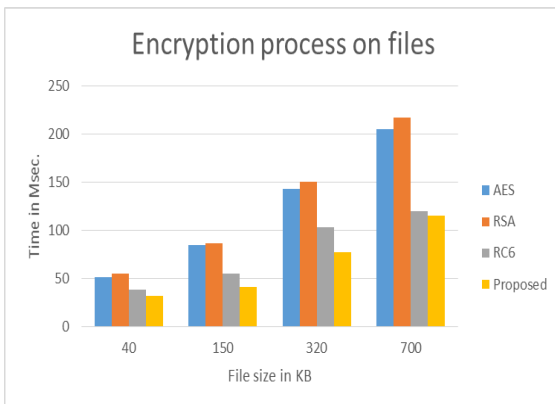


Figure 8: Encryption process on input files

It can be noticed from the chart presented in figure 8, that for a file size of 40 KB the encryption calculation time for all the compared algorithms are 51, 55, 38 and 32 M.sec respectively. I can also be noticed that the proposed algorithm outperforms other counterparts by consuming less time for all types of file sizes.

On the other hand the time consumed during decryption process is the time used by the algorithm to convert plain text into cipher text. Figure 9 shows the decryption time taken by different algorithms to calculate the decryption of data for different file sizes.

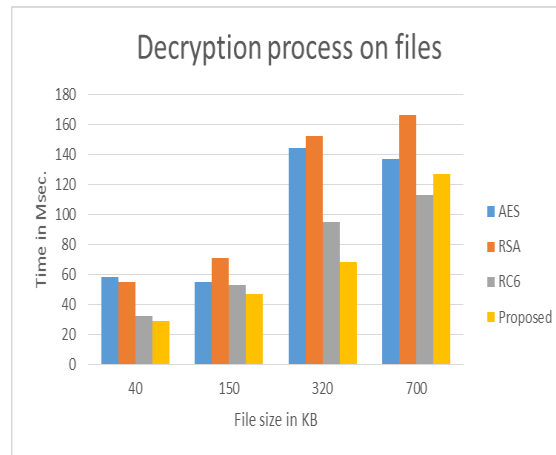


Figure 9: Decryption process on input files

It can be noticed from the chart presented in figure 9, that for a file size of 40 KB the decryption calculation time for all the compared algorithms are 58, 56, 32 and 29 Msec respectively. Moreover it shows that the proposed algorithm presents enhanced performance results in comparison to other algorithms by consuming an average less decryption time.

5. DIFFERENCE TO PRIOR WORK

Fully homomorphic encryption for cloud computing is a new concept of security which is to allow to provide the outcomes of calculations on encrypted data without knowing the raw entries on which the calculation was carried out respecting the confidentiality of data. As depicted in the section of Related Works, some of the proposed solutions supported addition or multiplication and very sound improvements to only smaller size cipher text in homomorphic encryption. However those works suffered from complexity and neither of them has

referred to the ciphertext retrieval decryption process. Our work is based on the application of fully Homomorphic encryption to the security of Cloud Computing with a light weight secure encryption/decryption scheme. The performance assessment and analysis of our work demonstrate the practice and validity of the proposed technique.

6. CONCLUSION AND FUTURE WORK

The adoption of cloud computing technology is growing rapidly among many IT and business firms. This is because of the immense advantages and features that has cloud computing is expected to offer. However, the security of the cloud is still at big risk and the threat of security and privacy is continuously increasing. Cloud operators are expected to manipulate client data without necessarily being fully trusted. Moreover, the users of cloud services are in fear of data loss and privacy since the data is hosted and stored on third party machines. The existence of highly sensitive data, and the high risks associated with its leakages is one of the most extensive factors that barricades many organizations and people from shifting their data to the cloud. For this reason, there is an emerging necessity to deploy mechanisms to prevent online piracy, privacy breach and violations of digital data. The aim of this research was to investigate the use of new encryption technology scheme based on fully homomorphic encryption (FHE) to mitigate the risk of cloud computing adoption and to help in providing confidentiality, integrity and verifiability of client data against untrusted cloud providers. The proposed scheme ensures the transmission of data between the cloud and the user safety. The encryption scheme also helps to overcome and mitigate the data confidentiality and integrity concerns, to gain the trust of organizations dealing with sensitive data to adopt cloud storage services. According to simulation results, the proposed scheme is more efficient and has better results in comparison to traditional encryption techniques. With homomorphic encryption, a company could encrypt its entire database of e-mails and upload it to a cloud. Then it could use the cloud-stored data as desired for example, to search the database to understand how its workers collaborate. The results would be downloaded and decrypted without ever exposing the details of a single e-mail.

While the proposed scheme is still in its development process we believe there should be an

improvement to extend this work and reduce the complexity of the proposed technique. For future works our proposed system will also be validated, and enhancing the encryption and decryption computation time will be considered.

REFERENCES:

- [1] Mell, P., & Grance, T. (2011). The NIST definition of cloud computing.
- [2] Ostermann, S., Iosup, A., Yigitbasi, N., Prodan, R., Fahringer, T., & Epema, D. (2009). A performance analysis of EC2 cloud computing services for scientific computing. In International Conference on Cloud Computing (pp. 115-131). Springer, Berlin, Heidelberg.
- [3] Armbrust, M., Fox, A., Griffith, R., Joseph, A. D., Katz, R., Konwinski, A., & Zaharia, M. (2010). A view of cloud computing. *Communications of the ACM*, 53(4), 50-58.
- [4] Li, H., Dai, Y., Tian, L., & Yang, H. (2009). Identity-based authentication for cloud computing. *Cloud computing*, 157-166.
- [5] Qian, L., Luo, Z., Du, Y., & Guo, L. (2009). Cloud computing: An overview. *Cloud computing*, 626-631.
- [6] Rittinghouse, J. W., & Ransome, J. F. (2016). *Cloud computing: implementation, management, and security*. CRC press.
- [7] Dinh, H. T., Lee, C., Niyato, D., & Wang, P. (2013). A survey of mobile cloud computing: architecture, applications, and approaches. *Wireless communications and mobile computing*, 13(18), 1587-1611.
- [8] Goyal, S. (2014). Public vs private vs hybrid vs community-cloud computing: a critical review. *International Journal of Computer Network and Information Security*, 6(3), 20.
- [9] Cloud Security Alliance, Security Guidance for Critical Areas of Focus in Cloud Computing V2.1, <http://www.cloudsecurityalliance.org/csaguide.pdf>
- [10] Cloud Security Alliance, Top Threats to Cloud Computing V1.0, <http://www.cloudsecurityalliance.org/topthreats/csathreats.v1.0.pdf>.
- [11] Hashizume, K., Rosado, D. G., Fernández-Medina, E., & Fernandez, E. B. (2013). An analysis of security issues for cloud computing. *Journal of Internet Services and Applications*, 4(1), 5.

- [12] Dan Boneh, Eu-Jin Goh, and Kobbi Nissim (2005). Evaluating 2-DNF formulas on ciphertexts. In Theory of Cryptography Conference, TCC'2005, volume 3378 of Lecture Notes in Computer Science, pages 325-341. Springer.
- [13] Rivest, R. L., Adleman, L., & Dertouzos, M. L. (1978). On data banks and privacy homomorphisms. Foundations of secure computation, 4(11), 169-180.
- [14] Gentry, C. (2009, May). Fully homomorphic encryption using ideal lattices. In STOC (Vol. 9, No. 2009, pp. 169-178).
- [15] Gentry, C., Groth, J., Ishai, Y., Peikert, C., Sahai, A., & Smith, A. (2015). Using fully homomorphic hybrid encryption to minimize non-interactive zero-knowledge proofs. Journal of Cryptology, 28(4), 820-843.
- [16] AWS Toolkit for eclipse, available online: http://docs.amazonaws.cn/en_us/AWSToolkitEclipse/latest/GettingStartedGuide/aws-tke-gsg.pdf?
- [17] Brunozzi, S. (2012, September). Big data and nosql with amazon dynamodb. In Proceedings of the 2012 workshop on Management of big data systems (pp. 41-42). ACM.
- [18] Smart, N. P., & Vercauteren, F. (2010, May). Fully Homomorphic Encryption with Relatively Small Key and Ciphertext Sizes. In Public Key Cryptography (Vol. 6056, pp. 420-443).
- [19] Van Dijk, M., Gentry, C., Halevi, S., & Vaikuntanathan, V. (2010, May). Fully homomorphic encryption over the integers. In Annual International Conference on the Theory and Applications of Cryptographic Techniques (pp. 24-43). Springer Berlin Heidelberg.
- [20] Divya, K., Sri, P. N., & Singh, T. C. (2014). Data Storage Transparent Security in Cloud Computing.
- [21] Suganya, S., & Damodharan, P. (2013, July). Enhancing security for storage services in cloud computing. In Current Trends in Engineering and Technology (ICCTET), 2013 International Conference on (pp. 396-398). IEEE.
- [22] Eletriby, S., Mohamed, E. M., & Abdul-kader, H. S. (2012). Modern encryption techniques for cloud computing. In ICCIT (pp. 800-805).
- [23] Gampala, V., Inuganti, S., & Muppidi, S. (2012). Data security in cloud computing with elliptic curve cryptography. International Journal of Soft Computing and Engineering (IJSCE), 2(3), 138-141.
- [24] Gomathisankarn, M., Tyagi, A., & Namuduri, K. (2011). HORNS: A homomorphic encryption scheme for Cloud Computing using Residue Number System. In Information Sciences and Systems (CISS), 2011 45th Annual Conference on (pp. 1-5). IEEE.