IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Volume-4, Issue-9 E-ISSN: 2347-2693

Providing Arithmetic Operations on RSA using Homomorphic Technique

Shiwali[1*], Gaganpreet Kaur Bhalla[2]

[1,2]*Dept. of CSE , Rayat & Bahra Institute of Engineering & Biotechnology, Mohali, Punjab, India*

Available online at: www.ijcseonline.org

Received: 22/Aug/2016 Revised: 02/Sept/2016 Accepted: 20/Sept/2016 Published: 30/Sep/2016

Abstract--- Cloud computing is a sound area for the field of research. In the current scenario of advanced technology the client and server architecture is been shifting from distributed or cluster to cloud architecture. The main part of this research relies on robust architecture which deals with Cloud Storage as a Service (SAAS) and comparative security measures of improvement and modifications [1]. Cloud computing is the internet based technology which providing the computing resources in the form of services over the internet. But security is the main issue occurred in the cloud computing because of that growth of cloud computing is less.By using cryptography algorithms, we improve the data security in cloud computing. The cryptography algorithms are used for purpose of secure transmission of private or secret message. There are several number of cryptography algorithms but here we integrate the new technique with RSA algorithm using Homomorphic Encryption and performing the arithmetic process on RSA which improving the security level without compromising the security of existing technique [2]. In this paper our main work to ensure the security of data, so we proposed a method by implementing RSA algorithm using Homomorphic Encryption.

Keywords--- Cloud Computing; Encryption; Decryption; RSA; Homomorphic Technique

1. INTRODUCTION

A. *Cloud Computing*

Cloud Computing is a technology that uses the internet and central remote servers to maintain data and applications. Cloud computing allows consumers and businesses to use applications without installation and access their personal files at any computer with internet access. It provides on-demand access through internet to computing different services [1,2, s3,7,8,13,14,15]. A simple example of cloud computing is Yahoo email, Gmail, or Hotmail etc. All you need is just an internet connection and you can start sending emails. The server and email management software is all on the cloud (internet) and is totally managed by the cloud service provider Yahoo, Google etc.

B. *Layers of Cloud Computing*

1) *Hardware layer:* Hardware layer is responsible for managing the physical resources of the cloud, including physical servers, routers, switches, power and cooling.

2) *Infrastructure layer:* The Infrastructure layer creates a pool of storage and computing resources by partitioning the physical resources using virtualization technologies.

3) P*lat-form layer:* The platform layer built on two of the infrastructure layer, the platform layer consists of operating systems and application framework.

4) A*pplication layer:* The application layer at the highest level of the hierarchy, the application layer consists of the actual cloud applications.

C. *Cryptography*

The cryptography is used for the secure transmission of secret message. Cryptography is the study and art of protecting information by encoding the data and transformation techniques [4]. In cryptography two widely used terms are Encryption and Decryption.

1) *Encryption:* Encryption is the method of encoding messages or information in such a way that only authorized parties can read it. It doesn't prevent hacking but it reduces the chances to read the data by the hacker because the data is in encrypted form. In an encryption scheme, the message or information that needs to be encrypted is known as plain text. The encryption of plain text using an encryption algorithm result into unreadable text known as cipher text [5]. This can be done with the use of an encryption key. These keys specify how the message is to be encoded.

2) *Decryption* is the method of decoding the cipher text into the plain text by using the decryption key. By using the cryptography it become very difficult for an adversary to recognize the original message from the cipher text because they do not have the access to secret key. But authorized person can easily get the original message from the encrypted message by using

the decryption algorithm that contains a secret decryption key [6, 10,11]. That is why the cryptography scheme needs an algorithm for key generation. On the basis of key generation algorithm, two types of cryptography schemes are available.

II. RELATED WORK

Related work introduces various investigates taking into account appearance-based and display based methodologies for Cloud Computing. A brief depiction of those late critical looks into is exhibited underneath:

A. *Yamuna[1]* Cloud computing is the way of providing computing resources in the form of services over internet. The cloud computing allows storing the user's data and to measure the applications and services provided by cloud server. There is an ample data stored at cloud storage server. Security is one of the major issues which reduce the growth of cloud computing So Cloud computing entails encyclopaedic security solutions .This paper presented secure file exchanging on Cloud using Blowfish, RSA algorithms which is capable of solving data security, authentication, and integrity problems of files on the cloud. Data security is improved by cryptography algorithms. The rightness of data is verified by introducing techniques. Enhanced system (RSA value) compares with simple RSA and Blowfish on basis of some performance parameters like: - throughput, encryption time, and cipher text and delay time. They integrate symmetric, asymmetric and algorithms which provide better results for performance parameters .TPA (Third party Auditor) which has to match the code for the integrity of user data in cloud on behalf of Data Owners. Data Owner can get notification from TPA when the data integrity is lost. If any unauthorized people access the data in cloud, they will be blocked.

B. *P.Saveetha[2]* studies on improvement in RSA algorithm and its implementation. The network security means to protect data during their transmission over channel of networks similarly internet security also to protect data during their transmission over a collection of interconnected networks in all over the world. Cryptography is the way of hiding the information during transmission over the channel. There are lots of cryptographic algorithms available to protect our data from intruders. RSA is also one of effective public key algorithm which needs the time and memory. The performance of RSA algorithm will be improved by reducing the modulus and private exponents.

C. *Yang[4]* explains ICDM: An Encryption that Supports Unlimited Times Homomorphic Arithmetic Operations on Encrypted Data. With the development of cloud computing, privacy has become the key problem of cloud security. The Homomorphic Encryption is an ideal way to protect users' privacy in cloud computing. But some of the existing Homomorphic Encryption schemes have less usability, and some are inefficient. There lacks of practical Homomorphic Encryption schemes in cloud computing at present. CESVMC is a scheme supposed to solve the problem. CESVMC ensures that after calculating the user's encrypted data and returning the cipher result to user by the service provider, the user can decrypt the cipher result and get the right service result. But CESVMC only supports multiplication or division operation once. Meanwhile, to decrypt the cipher text, user needs to tell which type of operation has been done to the cipher text. All these constrain the usability of CESVMC in cloud computing. To solve these problems, an improved CESVMC (ICDM) is proposed. In encryption algorithm, the information of plaintext and the operation type are hidden in a diagonal matrix. Then the diagonal matrix is encrypted by using an invertible matrix as secret key. In decryption algorithm, ICDM chooses the right encryption method by reading the sign of the operation type without any manual interventions. Besides, the arithmetic operations on cipher text correspond to the arithmetic operations on matrix. Security analysis indicates ICDM is IND-CPA.

III. TECHNIQUES USED

The RSA Algorithm using Homomorphic Technique as follows:

1) Choose two very large random prime integers p and q.
2) Compute n and φ(n): n = pq and φ(n) = (p-1)(q-1).
3) Choose an integer e, $1<e<\varphi(n)$ such that:
$$\gcd(e,\varphi(n)) = 1.$$
(Where, gcd means greatest common denominator).
4) Compute d, $1<d<\varphi$ such that:
$$ed = 1 (\mathrm{mod}\, \varphi(n)).$$
(Where, the public key is (n, e) and private key is (n, d), the values of p, q and φ (n) are private, e is the public and encryption exponent, d is the private and decryption exponent).
5) After that program reads the encrypted RSA file and then reads each character one by one and first converts that each character in ASCII standard value.
6) Then convert each ASCII value in binary value using java byte class and after that we apply first XOR gate operation and get the length of bits, then divide it by number of index.
7) So that we can encrypt the block size and copy new array list.
8) After that applying the shift left operation on all bits and storing it further in a new array list.
9) Then converts bits in ASCII standard value first and then convert in characters and then write in cipher text file.

10) The cipher text file is then received by the server and decryption process starts (reverse encryption) and final plain text is then sent to the user.

IV. PARAMETERS USED

Following are the parameters on the basis of which working of these two techniques is measured.

1) *Time:* In this method, time depicts the time required to encode, decode, encrypt and decrypt the plain text. The time is measured in milliseconds.

2) *Size:* Size depicts the size of file that is being fed for encryption and decryption. The time used in encryption and decryption depends on the size of the file.

3) *Throughput:* Throughput refers to the performance of tasks by a computing service or device over a specified period. It measure the amount of completed work against time consumed and may be used to measure the performance of processor, memory and/or network communications.

Throughput = size of file (in KB) / time taken (in seconds)

It is basically defined as number of data bits transferred over network in seconds.

V. PROPOSED WORK

There are some main phases of proposed work of this thesis. These phases are discussed in below points:

Phase 1:- In this phase, we will design the user interface using java swings.

Phase 2:- This phase includes the design of admin interface for the admin panel which can manage user list.

Phase 3:- This phase includes implement RSA algorithm for encrypt the content of file.

Phase 4:- In this phase we apply the Homomorphic encryption using gates.

Phase 5:- Final results will be validated and will be compared with other algorithm.

VI. RESULTS AND DISCUSSION

The following tables and figures are highlighted the results of proposed work:

The Table 1.1 represents the six different sizes of files and corresponding encryption execution time taken by simple RSA algorithm and RSA using Homomorphic encryption in seconds. By analyzing the Table 1.1, we conclude that the encryption time taken by RSA using Homomorphic encryption is very small as compare to RSA. The encryption time taken by simple RSA and RSA using Homomorphic encryption.

Table 1.1: Encryption Execution Time

Input File Size (KB)	Encryption Execution Time	
	RSA(ms)	RSA using HE(ms)
118	1000	950
153	7300	1358
196	8500	1138
312	7800	2050
868	8200	3950

The Table 1.2 represents the five different sizes of files and corresponding decryption execution time taken by simple RSA algorithm and RSA using Homomorphic encryption in seconds. By analyzing the Table 1.2, we conclude that the decryption time taken by RSA using Homomorphic encryption is very small as compare to RSA. The decryption time taken by simple RSA and RSA using Homomorphic encryption.

Table 1.2: Decryption Execution Time

Input File Size (KB)	Decryption Execution Time	
	RSA(s)	RSA using HE(ms)
118	5000	1823
153	4900	2057
196	5900	2115
312	5100	3679
868	5100	3814

The Table 1.3 represents the five different sizes of files and corresponding throughput execution time taken by simple RSA algorithms and RSA using Homomorphic encryption in KB/Seconds. By analyzing the Table 1.3, we conclude that the throughput time taken by RSA using Homomorphic encryption is large as compare to RSA. The throughput time taken by simple RSA and RSA using Homomorphic encryption.

Table 1.3: Throughput Execution Time

Input File Size (KB)	Throughput Execution Time(KB/Seconds)	
	RSA(s)	RSA using HE(ms)
118	94.400	99.368
153	167.671	901.325

196	184.71	1377.856
312	320.00	1217.561
868	846.829	1761.096

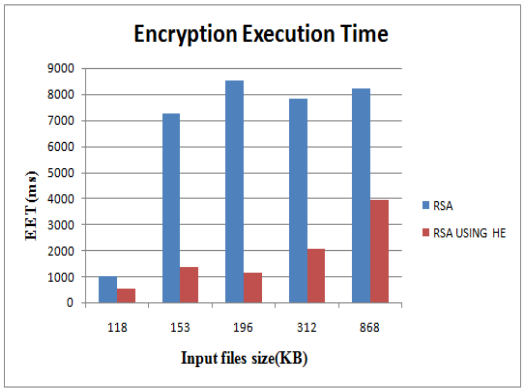

Figure 1.1: EET among RSA and modified RSA using Homomorphic Encryption

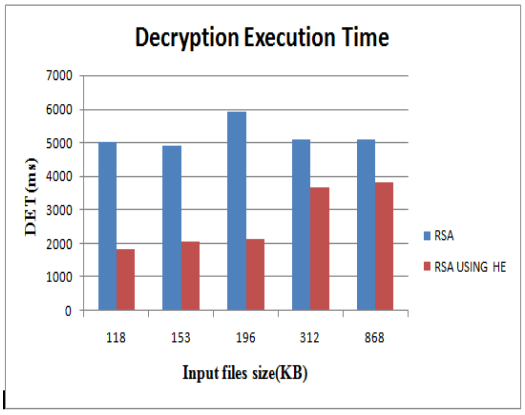

Figure 1.2: DET among RSA and modified RSA using Homomorphic Encryption

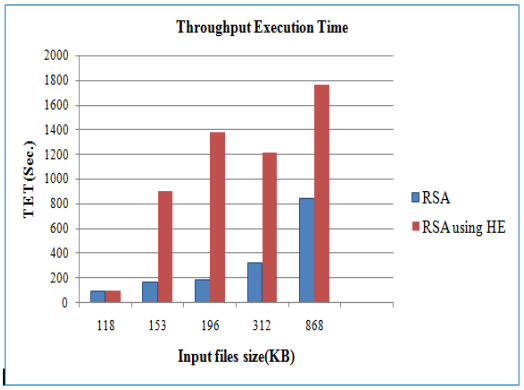

Figure 1.3: TET among RSA and modified RSA using Homomorphic Encryption

VII. CONCLUSION

During research on issues on cloud computing and after its successful implementation. It's been concluded that RSA was successful and provides a strong point of security to existing cloud architecture. The Purpose of adding Homomorphic in client and server side was successful and gave a marginal better outcomes than previous research as this method will help in authorization preservation the content value is manipulated with arithmetic XOR operation as a result the data stored in storage has changed its value. In architecture while adding Homomorphic we chose to keep data in form of text file. In the end the throughput of RSA is computed, the execution time of encryption and decryption both are computed in order to know if it has any effect on performance of actual RSA. These results outcome have opened a better area towards security. At last the comparison of simple RSA Encryption algorithm and RSA using Homomorphic technique and schemes is done which may help to extend current research techniques.

ACKNOWLEDGEMENT

I am extremely grateful to my Guide and my relatives who dependably backing and help and guide me amid my paper or research work. Furthermore uncommon because of my dad who dependably bolster my inventive thoughts.

REFERENCES

[1] Yamuna, V. and Priya, A. (2015), *"efficient and secure storage in cloud computing RSA and DES function"* International Journal of Innovation Research in Computer and Communication Engineering, ISSN 2320-9801, Vol. 3, issue 7, July 2015.

[2] Saveetha, P and Arumugam, S. (2015), *"Study on Improvement in RSA Algorithm and its Implementation"*, IJCCT, Vol. 3, Issue 6, 7 august 2012, pp 61-65.

[3] Jirwan, N.; Singh, A. and Vijay, S. (2013), *"Review and Analysis of Cryptography Techniques"*, International Journal of Scientific & Engineering Research, Vol. 4, March 2013.

[4] Yang, P.; Gui, X.; Yao, J.; Lin, J and Tian, F. (2013), *"ICDM: An Encryption that Supports Unlimited Times Homomorphic Arithmetic Operations on Encrypted Data"*, IEEE 16th International Conference on Computational Science and Engineering, 2013, pp 1220-1225

[5] Shivlal Mewada, Sharma Pradeep, Gautam S.S. (2016), "*Classification of Efficient Symmetric Key Cryptography Algorithms"*, International Journal of Computer Science and Information Security (IJCSIS) USA, Vol. 14, No. 2, pp (105-110), Feb 2016

[6] Kahte, A. (2013), "*Cryptography and Network Security*", 2nd Edition, Biswarup Nil Kundu, 09 August, 2013.

[7] Gupta, P. and Gupta, S. (2012), *"Mobile Cloud Computing: The Future of Cloud"*, IJAREEIE, Vol. 1, September 2012, pp 134-145.

[8] Prasad, M.R.; Gyani, J and Murti, P.R.K. (2012), *"Mobile Cloud Computing: Implications and Challenges"*, Journal of Information Engineering and Applications, Vol. 2, 7, 2012, pp 7-15.

[9] Shivlal Mewada, Sharma Pradeep, Gautam S.S. (2016), "*Exploration of Efficient Symmetric Algorithms*", 3rd IEEE International Conference on "Computing for Sustainable

Global Development", 16th -18th March, 2016, ISBN 978-93-80544-20-5

[10] Thambiraja, E.; Ramesh, G and Umarani, R. (2012), *"A Survey on Various Most Common Encryption Techniques",* IJARCSSE, Vol. 2, July 2012, pp 226-233.

[11] Mewada, Shivlal, Pradeep Sharma, and S. S. Gautam(2016). "*Exploration of efficient symmetric AES algorithm.*" In Colossal Data Analysis and Networking (CDAN), Symposium on, pp. 1-5. IEEE, 2016.

[12] Tebaa, M and Hajji, S.E. (2012), *"Homomorphic Encryption Method Applied to Cloud Computing",* IEEE, June 2012, pp 86-89.

[13] Ranjit Ranjan, Dr. A.S Baghel, Sushil Kumar "improvement of NTRU cryptosystem" international journal of advanced research in computer science volume 2, issue September 2012.

[14] Parsi, K. (2012), *"data security in cloud computing using RSA algorithm",* International Journal of Research in Computer and Communication technology, ISSN 2278-5841, Vol. 1, Issue 4, September 2012.

[15] Brenner, M.; Wiebelitz, J.; Vonvoigt, G and Smith, M. (2011), *"Secret Program Execution in the Cloud Applying Homomorphic Encryption",* IEEE 5th International Conference on Digital Ecosystems and technologies (IEEE DEST), June 2011, pp 114-119.

