IJCSE International Journal of Computer Sciences and Engineering Open Access

Research Paper Vol.-6, Issue-5, May 2018 E-ISSN: 2347-2693

SDSA: An Implementation of Secure Data Sharing Approach Using Homomorphic Encryption

T. Aaliya[1*], R. Sharma[2]

[1*]Computer Science, RKDF School Of Engineering, RGPV, Indore, India
[2] Computer Science, RKDF School Of Engineering, RGPV, Indore, India

**Corresponding Author: tayyebaaaliya@gmail.com, Tel.: +91-81090-51955*

Available online at: www.ijcseonline.org

Accepted: 19/May/2018, Published: 31/May/2018

***Abstract*—** Cloud computing is fast growing technology that enables the users to store and access their data remotely. While accessing the data from cloud, different users may have relationship among them depending on some attributes, and thus sharing of data along with user privacy and data security becomes important to get effective result. In this paper, we design and implement Secure Data Sharing Approach i.e. SDSA, for dynamic groups in public cloud environment. In this technique, user uploaded their data on cryptographic server in encrypted format using Homomorphic encryption algorithm tiger hash algorithm is used for key generation which is input in the encryption algorithm. In SDSA a user is able to share data with others in the group without revealing characteristics privacy to the other user. Moreover, SDSA supports efficient user revocation and fresh user joining. More especially, efficient user revocation can be achieved through a public revocation list without harming security of the other remaining users in user portal. In addition, the storage overhead and the encryption decryption computation cost are constant. Extensive analyses show that this proposed scheme satisfies the desired security requirements along with the secure sharing with other and preserve privacy policy when group sharing is processed that guarantees efficiency as well.

***Keywords*—** Cloud Computing, Homomorphic Encryption, Security, Secure Sharing, Cryptography, Cryptographic Server, Decryption

I. INTRODUCTION

In recent years a significant change in technology is observed. The new technology and applications are frequently consuming the network and their services. These applications not only carrying data on private networks but the applications are also utilizing the services of public network. But the use of public network is not much trustworthy and secure for private and confidential data exchange. Because a number of times applications requires the private and sensitive data such as banking information, private images and others. In this context the security in network communication and their data is a primary concern in network and data security [1, 2, 3].

There are a number of different approaches that exist for securing the data on network & among them the cryptography is a popular and classical approach to secured data. Additionally the key reason behind use of cryptography for security is their low cost implementation and freedom and flexibility to change the security according to needs. Therefore, in this paper key area of work is investigated and design of a secure data sharing approach for cloud storage.

Rest of the paper is organized as follows, Section I contains the introduction of secure data sharing in cloud computing, Section II contain the background overview of security of the network data and key concept of secure data sharing in cloud computing. Section III contains the core methodology of our SDSA approach and their description, Section IV contains the result and discussion of the obtain output, Section V contain concludes research work with future directions).

II. BACKGROUND

The background of a study is an important part of our research paper. It provides the context and purpose of the study. Hence there is need for background study that contributes to prepare proposed system.

A. What is network security?

The technology used in daily life is changing. Information technologies are transforming the ways we create, gather, process, and share information. Computer networking is driving many of these changes; electronic transactions and records are becoming central to everything from commerce to health care. The explosive growth of the Internet exemplifies this transition to a networked society [4].

In a broadly way, Network security is any activity designed to protect the usability and integrity of your network and data. It includes both hardware and software technologies. Effective network security manages access to the network. It targets a variety of threats and stops them from entering or spreading on your network.

Network security combines multiple layers of defences at the edge and in the network. Each network security layer implements policies and controls. Authorized users gain access to network resources, but malicious actors is blocked from carrying out exploits and threats [5].

B. *Secure Data Sharing*

Define Cloud systems can be used to enable data sharing capabilities and this can provide an abundant of benefits to the user.

To enable data sharing in the Cloud, it is essential that only authorized users are able to get access to data stored in the Cloud. Figure 1 demonstrates the Secure Group Sharing in Cloud. When the data owner wants to share their own data to a group, he/she sends the key used for data encryption to each member of the group. Any of the group members can then get the encrypted data from the Cloud and decrypt the data using the key and hence group member does not require the interference of the data owner [6, 7].

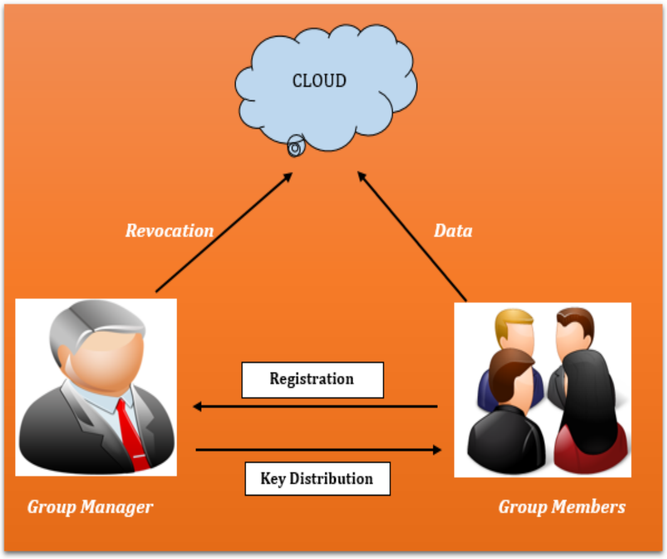

Figure 1 Secure Group Sharing in Cloud

III. PROPOSED WORK

This section introduces the functional aspects of the proposed system. In addition of that the core system design concept and the algorithm steps are explained in detail.

A. *Methodologyy*

The proposed methodology is described in this section which includes the different component of the model which is used to process the data one by one. The proposed technique is described in three major entities: Admin, Cryptographic Server and User. The given diagram 2 shows the list of entities of the system.

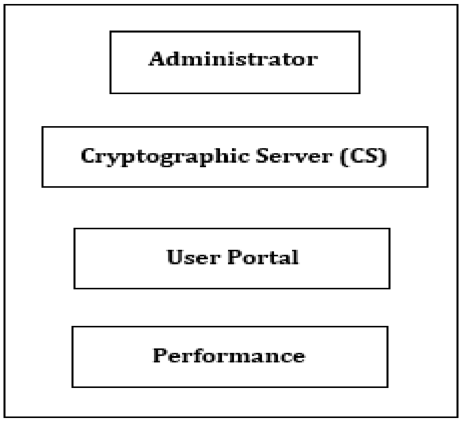

Figure 2 Layers of Entity

The above figure depicts the different entity of the proposed SDSA system. This is the combination of the entire sharing system which ensured about secure file sharing mechanism. In this diagram, there are four layers of the system, which constitute process of the working model. The admin entity is independent to the CS portal and user portal, but these two CS portal and user portal dependent to each other for accessing the sharing mechanism and file encryption and decryption. In last overall system performance is measured in terms of encryption time and memory, decryption time and memory and server response time. This performance parameter show the system accessibility and availability to the end user application where necessity for secure sharing technique.

B. *Description of the Working Model*

This section deals with the concern technique and describes step by step working and entire process in such a way that protecting and securing sensitive information from the malicious activity. Figure 3 show the admin entity.

1) Admin Portal Entity:

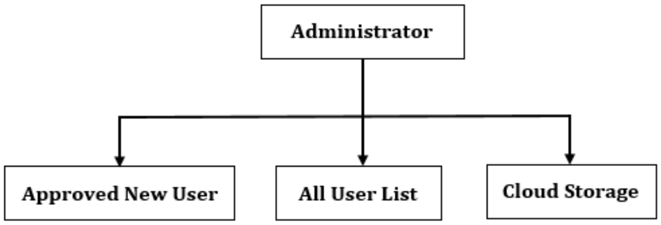

Figure 3 Admin View

The admin entity is responsible for handle activity of all users that means different users are sharing their file and have the access permission of system privilege. In this panel, admin have to right to approved or decline newly registered user request. If admin granted the permission to new user, the user can access log in to their panel and access privilege i.e. upload and sharing file. Admin can also view all files as storage of all users and activity of sharing mechanism. There are different number of user listed in user portal also view by the admin portal.

2) User Portal Entity:

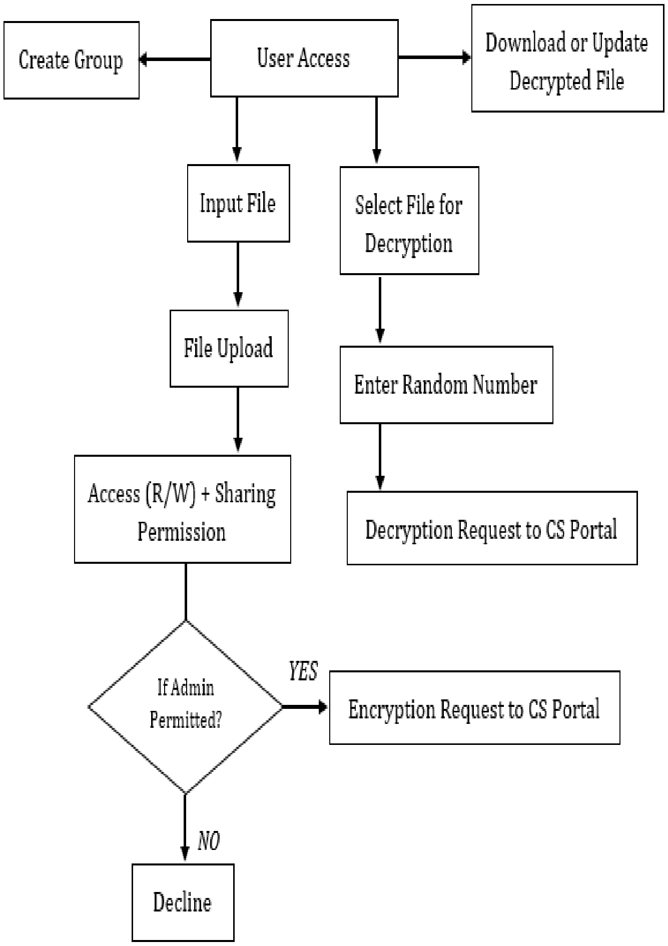
Create Group
User Access
Download or Update Decrypted File
Input File
Select File for Decryption
File Upload
Enter Random Number
Access (R/W) + Sharing Permission
Decryption Request to CS Portal
If Admin Permitted?
YES
Encryption Request to CS Portal
NO
Decline

Figure 4 User Entity Working

The In this entity, to access the functionality on user portal, firstly user need to login via their credentials e.g. (User name, login Id, Password). Once the user is successfully login then it will redirect to user portal. Hence, if there are number of user login on portal simultaneously, then they will create group for secure sharing file among the user. Finally, users have been created group then they will be given an access and sharing permission to other users of the group.

In the second scenario, user input the file for access and sharing purpose. This input file users uploaded on the cryptographic server, for this user gives the access permission to the group members that who will be access and share file according to read, write permission. Once the user have permitted then CS portal receives the request of file encryption. How file will be encrypted by CS? We will explain in next phase.

Therefore, if the is encrypted and ready to decrypt and download, it will be processed by user. First the user need to select the file which he want to decrypt, then for ensure security of legitimacy of user system generate the random number and mail to the users email id. After entering this random number, decryption request is send to the CS portal.

In third scenario, if user want to download then he can also download file and can also edit/view file on this portal. If users want to update file then similar process is repeated of encryption request to CS portal. In the next phase we explain encryption, decryption and key management process by means of third party cryptographic server (CS) portal.

3) Cryptographic Server (CS) Portaly:

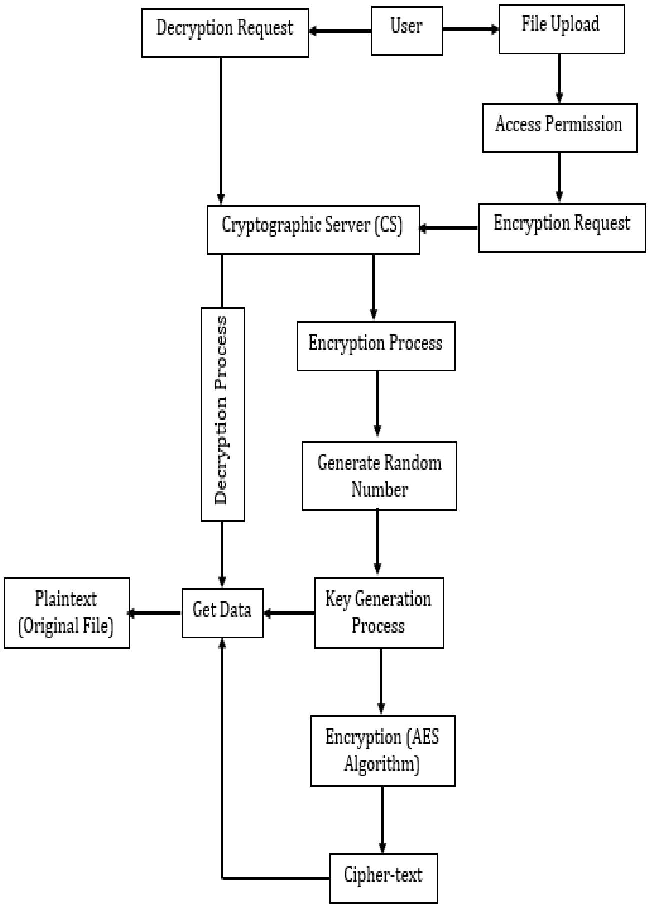

Figure 5 Cryptographic Server (CS) Working

In the above section we have seen how encryption request arrives at the CS portal for encryption and decryption. Once a file has uploaded by user and similarly access and share permission is done CS perform main task for providing security of the user data by encrypting and decrypting.

In first sight of view, for encryption, the CS starts the encryption process and it will generate random number which is share to the user through email id of user itself. After this, symmetric key is generated which is used by encryption algorithm. The overall process of key generation is depicted using figure 6.

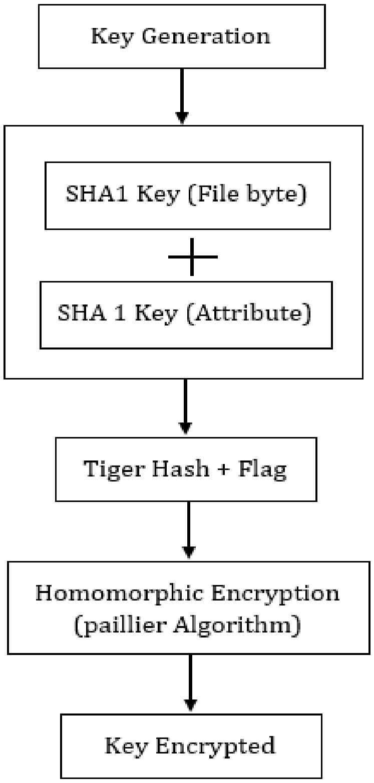

Figure 6 Key Generation Process

In key generation phase it is important to emphasize the process of key encryption. For the security assurance, here we also encrypt the key which is simultaneously used for file encryption. Firstly, we have generated SHA1 key of the file byte data and SHA1 key of the file attribute. The following table shows the list of attribute:

Table 1. Attribute List

S. No.	*Attribute Name*
1	<attr1, MCA>
2	<attr2, MTECH>
3	<attr3, ME>
4	<attr4, BE>
5	<attr5, BCA>

In the above table we lost the number of attribute which is access through the cryptographic server for key generation. On this generated SHA1 key, pass the Tigerhash algorithm and flag. The flag should be 0 or 1 depend on the Key. The produced output of this process, we apply encryption algorithm i.e. paillier algorithm which is commonly known as homomorphic encryption algorithm. This algorithm is successfully encrypting the key.

Finally, we are encryption out data file using AES encryption algorithm and encrypted key (used in AES algorithm). Consequently, it will generate the cipher-text (encrypted text). The overall process is the mixture of key generation and file encryption.

Similarly, for decryption of the data file, user send the decryption request to the CS and CS will process this request. The CS will decrypt the data by getting the cipher-text and encrypt key which is pass into AES algorithm. After, this process, decrypted data i.e. original text is produced. This data is downloaded from the user portal.

The encryption, decryption and key management is the entire process of the cryptographic server (CS) which is fully maintained in a systematic way. This is ensure secure sharing of data file among the user and fully secure from outside malicious activity.

In order to demonstrate the cryptographic scenario of the proposed Secure Data Sharing Approach for encryption and decryption process is presented in this section. The proposed cryptographic technique in terms of algorithm is given in figure 7, 8 and 9 of key generation, encryption and decryption respectively.

```
Input: File Byte ($F_B$), Flag ($F$), Attribute ($Attr$)
Output: $Enc_{Key}$

Process:
    1. $F_B(SHA1) = generate.SHA1\ (F_B)$
    2. $AttrByte\ (SHA1) = generate.SHA1\ (Attr)$
    3. $sum = [F_B(SHA1) + F_B(SHA1)]$
    4. $TigerHash_{Key} = generate.TigerHashAlgo(sum) + F$
    5. $Enc_{Key} = paillierAlgo.encrypt\ (TigerHash_{Key})$
    6. Return $Enc_{Key}$
```

Table 7 Key Generation Process

```
Input: Input Text $I_t$, $Enc_{Key}$, Text $t$
Output: Ciphertext ($C$)

Process:
    1. $R = ReadInputData\ (I_t)$
    2. $Enc_t = AES.encrypt(R, Enc_{Key})$
    3. Return C
```

Table 8 Encryption Process

```
Input: Ciphertext ($C$), $Enc_{Key}$
Output: Original text $O_t$

Process:
    1. $D = Ciphertext\ (C)$
    2. $Dec_t = AES.decrypt\ (D, Enc_{Key})$
    3. Return $O_t$
```

Table 9 Decryption Process

IV. Results and Discussion

The proposed work is intended to provide a dynamically secure group data sharing and access services in a decentralized manner. This chapter provides the summary of the performed for cloud oriented in security concerns and the future extension of the work is also suggested.

A. *What is network security?*

The amount of time required to perform encryption using the selected algorithm is termed as the encryption time of the system. The encryption time of the proposed system is demonstrated using figure 5.1 and the table 5.1.

$$\text{Time consumption} = \text{Algo End Time} - \text{Algo Start Time}$$

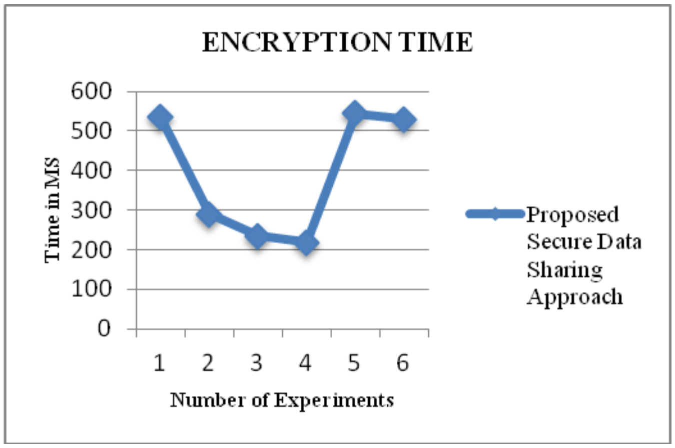

Figure 10 Encryption Time

In order to show the performance of implemented data sharing scheme, encryption execution time is reported in figure 10 and table 2. In this diagram the X axis shows the different experiments on which we run different files as an input and the Y axis shows the amount of time consumed for encrypting the input text file. Additionally the performance of proposed system is given using blue line. According to the given results the proposed system consumes less time for file uploading. Additionally the results shows the amount of time consumed is depends on the amount of data provided for execution. Moreover, while using proposed data security, enhance the security respect to the sharing of file among different parties.

Table 2. Encryption Time

Number of Experiments	*Proposed Secure Data Sharing Approach*
1	534
2	290
3	234
4	218
5	544
6	530

B. *Decryption Time*

The amount of time required to recover (Decrypt) the original data from the cipher text is known as the decryption time of the algorithms. The figure 11 and table 3 shows the obtained performance of the system in terms of millisecond. To show the performance of secure sharing scheme the blue line shows the performance of proposed algorithm.

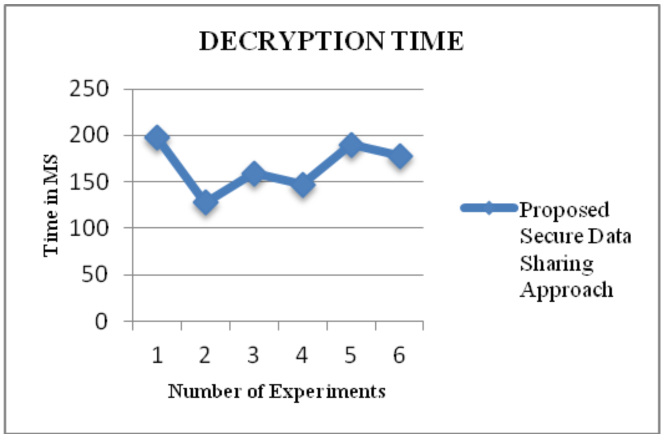

Figure 11 Decryption Time

In given figure 11, X-axis shows the different numbers of experiments are performed and the Y-axis shows the amount of time consumed for decryption process. According to the generated results the encryption time is higher than the decryption time in the system, but the decryption time of the proposed algorithm is much adaptable and after secure sharing user can be downloaded in their system.

Table 3. Decryption Time

Number of Experiments	*Proposed Secure Data Sharing Approach*
1	198
2	128
3	159
4	147
5	190
6	178

C. *Encryption Memory*

The amount of main memory required to execute the algorithm with the input amount of data is known as the encryption memory. The total memory consumption of the algorithm is computed using the following formula.

$$\text{Consumed Memory} = \text{Total Memory} - \text{Free Memory}$$

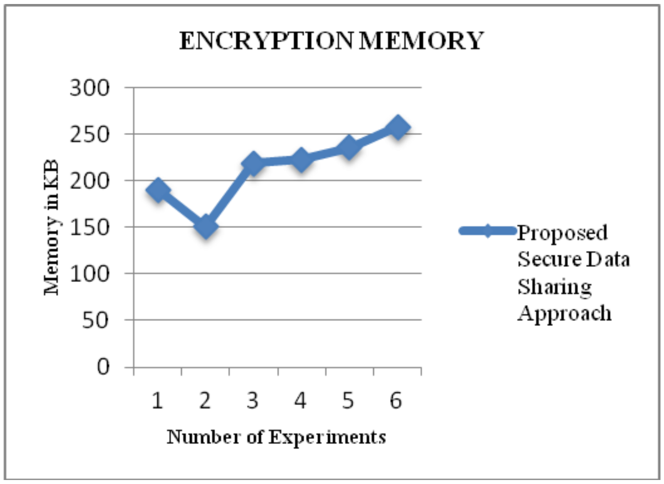

Figure 12 Encryption Memory

The figure 12 and the table 4 show the encryption memory consumption of the proposed approach. In this diagram the amount of main memory consumed is given in Y axis and the number of experiments are reported in X axis. According to the obtained performance the proposed algorithm consumes fewer resources as we seen during the execution of algorithm.

Table 4. Memory Consumption

Number of Experiments	*Proposed Secure Data Sharing Approach*
1	191
2	151
3	219
4	223
5	236
6	258

D. Decryption Memory

The amount of main memory required to recover the original file from the cipher text is known as the decryption memory consumption. The figure 13 and table 5 shows the amount of main memory consumed during the data recovery process. In this diagram the X-axis depicts the different experiments of different file size used for decryption and the Y axis shows the amount of main memory consumed during the decrypting data file. According to the obtained results the amount of main memory used is less than of encryption memory and consume less space of proposed algorithm.

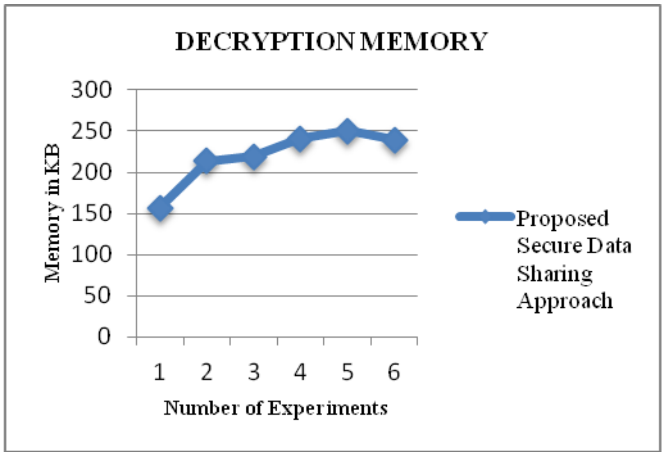

Figure 13 Decryption Memory

Table 5. Decryption Memory

Number of Experiments	*Proposed Secure Data Sharing Approach*
1	157
2	214
3	219
4	241
5	250
6	239

E. Server Response Time

The amount of time required to produce the outcome after making the request from the server is termed as the server response time. The response time not included the encryption or decryption activity during these measurements. The computed response time for proposed cryptographic technique is shown in figure 14 and table 6.

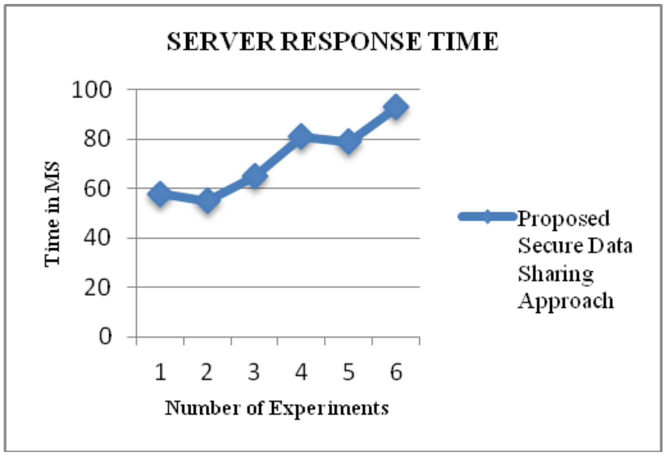

Figure 11 Response Time

X axis of this diagram contains the amount of experiments performed and the Y axis shows the amount of time required for generating the response through the server. This can also term as the communication overhead for the system. According to the computed results the response time is not depends on the amount of file size or other parameters. That is directly depends on the amount of work load on the target server where the data is stored or the application is hosted.

Table 6. Response Time

Number of Experiments	*Proposed Secure Data Sharing Approach*
1	58
2	55
3	65
4	81
5	79
6	93

V. Conclusion and Future Scope

A. Conclusion

The Public clouds are popular nowadays, where they are generally used in the storage and retrieval of the user’s information. It is given as a secure way of data sharing with multiple members. It has very impact in the user’s way of data storage. The study of secure data sharing is an increasingly research problem. The main aim of the proposed work is to provide a secure and efficient data group sharing and storage services using the public cloud. Reducing data storage and processing cost is a mandatory requirement of any organization, while analysis of data and information is always the most important tasks in all the organizations for decision making. In this proposed secure sharing approach, file sharing is initiated in the cloud dispersed environment. The file is shared between one user to multiple user or/and single to single user.

In addition of that for preventing the unauthorized access to the system we make the admin module which is responsible to handle uncommon activities in on user portal. In this work, whenever, file is uploaded by means of user portal then it will be go for encryption on the cryptographic server (CS) portal and this CS gives the access and sharing (read or write or both) permission to particular user that want to access privileged. CS generates the random number and key before encrypting the file. Once, file is encrypted successfully then it will be available for sharing to user which has access permission already. In this process, data are encrypted using AES algorithm and key is generated and encrypted using homomorphic (paillier) algorithm.

B. Future Work

The The proposed work for group sharing data and their secure access from the dispersed manner is implemented successfully. Additionally the system performance with the cryptographic implementation of the system is also obtained which is adoptable.

- The phishers and attackers are also aware about the technology changes therefore the security based on cryptography need to be change with the time.
- Currently the decentralized technique of data distribution or access is efficient according to the newly appeared techniques of the performance improvement of server need to adopt the newer techniques for security and efficiency.
- The proposed work need to modify more for text based cryptography because the Paillier algorithm is suitable for numerical data encryption.

References

[1] Stallings, William. Cryptography and network security: principles and practice. Pearson Education India, 2003

[2] Nadeem, Aamer, and M. Younus Javed. "A performance comparison of data encryption algorithms", In Information and communication technologies, 2005. ICICT 2005, First international conference on, pp. 84-89. IEEE, 2005.

[3] Denning, Dorothy E., and Peter J. Denning. "Data security." ACM Computing Surveys (CSUR) 11, no. 3 (1979): 227-249.

[4] Herdman, R. "Information security and privacy in network environments." The Office of Technology Assessment (OTA) (1994).

[5] Sattarova Feruza Y. and Tao-hoon Kim, “IT Security Review: Privacy, Protection, Access Control, Assurance and System Security”, International Journal of Multimedia and Ubiquitous Engineering Vol. 2, No. 2, April, 2007

[6] Wei, Jianghong, Wenfen Liu, and Xuexian Hu. "Secure data sharing in cloud computing using revocable-storage identity-based encryption." IEEE Transactions on Cloud Computing (2016).

[7] B. V. Varshini, M. Vigilson Prem and J. Geethapriya, “A Review on Secure Data Sharing in Cloud Computing Environment”, International Journal of Advanced Research in Computer Engineering & Technology (IJARCET), Volume 6, Issue 3, March 2017.

Authors Profile

Miss.T. Aaliya pursed Bachelor of Engineering from RGPV University, India in 2014. She is currently pursuing M.tech (Cyber security) from RGPV, University, India since 2014. She has published Survey paper “Cloud Data Security Using Cryptography: A Review” in reputed international journal it’s also available online. Her main research work focuses on Cryptography Algorithms, Network Security, Cloud Security and Privacy.

Mr R. Sharma pursed Bachelor of Engineering from RGPV University, india in year 2015 and M.tech from DAVV University, India in year 2017. He is currently working as Assistant Professor in Department of Computer Science, RKDF School Of Engineering, India since 2017. He has published research paper “Experimental Analysis Of Complex Network Using Routing Protocols” in reputed international journals it’s also available online. Under his guidence more than 10 paper has published. His main research work focuses on Network Management and Information Security. He has 1 years of teaching experience.