# A Secure Cloud Storage using Client-Server Mutual Authentication and Attribute based Encryption

**[1]RINA PATIL, [2]MR. PRITESH JAIN**

[1]Research Scholar, [2]Assistant Professor
Department of Computer Science & Engineering Patel College of Science and Technology, Indore, India

***Abstract*** **- Now in these days almost every person interacted with the different kinds of devices and applications that generate confidential and sensitive data. In addition of that to manage and preserve such kind of data a secure environment is also required. The cloud offers such kind of secure and authentic services to store the data. The cloud offers a cryptographic manner by which the data is encrypted first and then preserved on cloud. But there are not much effective techniques are available that provide authentication for secure data access and provide the security during the phishing and forge attacks. In this presented work a secure storage technique with client server mutual authentication technique is proposed for design and implementation that offers both the primary goal of secure data hosting on cloud. The proposed work first introduces the user attribute based encryption technique. This encryption technique provides the data access or recovery to only the user who encrypt the file using their attribute. In addition of that authentication technique is implemented using the image and tag. That helps to authenticate both client and server. Using this method client identifies the actual server and server identifies the user. The implementation of the proposed approach is conducted using the JAVA technology. After the implementation the performance of the system is measured which demonstrate the method is secure and efficient for achieving security and authenticity of user access of the data files on cloud servers.**

***Keywords*****: Cloud Computing, Access Control, Mutual Trust, Attribute based Encryption, Cryptography Authentication, AES*;***

## I. INTRODUCTION

Cloud technology is an efficient storage and computational platform for providing high scalable solution. But now in these days that is also used for preserving the confidential and sensitive data over cloud too. Therefore the security as well as strong authentication mechanism required for managing the data owner and their confidential data. The proposed work is focused on exploring the domain of cloud data storage techniques and their authentication techniques. During the investigation there a number of secure techniques for cloud data storage is observed but there are very fewer work are noticed that providing the solution for authentication.

In this presented work two techniques are designed for securing data and the data confidentiality. In first the attribute based cryptographic technique is designed that utilizes the user attributes as the security key for encryption and decryption of file which are required to store on server. in addition of that a client server mutual authentication technique is implemented for authenticating the user and server. The benefit of this technique is the user assured that this server is actual server or not. And server assured that the client is actual client or not. That technique also helps to secure the data and the data server during the forge and phishing attacks on the server. Therefore the proposed work is a promising approach for securing data over cloud and managing the user data according to their ownership. In addition of that the technique also protects the storage and data during the various kind of attack on server.

## II. PROPOSED WORK

This chapter includes understanding of the proposed methods implemented for providing security and authentication for the cloud data storage. Therefore first the overview of the system is provided and their methodology is described. Finally the models steps are explained using the algorithm steps.

### A. System Overview

The cloud computing is one of the popular technology that is involved common peoples too for offering different kinds of services. The cloud not only provides the efficient and scalable computing experiences to different large scale applications that also provide a trusted data hosting services too. Therefore a significant amount of cloud users are consumes the data hosting services. But the data hosting service providers are move their data on other storage servers to reduce the maintenance cost. That process of data movement is termed as the cloud data outsourcing. The outsourcing worries the clients and the service providers about the data confidentiality and the security. Thus the concept of cryptographic cloud is implemented for securing data. there are a number of cryptographic cloud concept are available but there are less technique for secure authentication is exist for access control and securing the data access.

The proposed work is intended to design a security technique for the cloud environment. That complete two primary goals first provide the secure and trustworthy hosting solution for user data. That data is only accessible or recoverable by the data owner.

Secondly a strong authentication mechanism which provides solution for phishing attacks and the forged site attacks. The proposed solution involves the development of an attribute based encryption technique. That encryption technique usage the user attributes as the encryption key. Therefore the actual cloud account holder can recover the uploaded data on server. In addition of that a client server mutual authentication technique implemented for authorizing the client and server. In this section the basic overview of the proposed security technique for cloud is presented and in next section the detailed system design and their components are explained.

**B. Methodology**

The proposed system architecture for providing the secure hosting and authentication technique is demonstrated in figure 2.1. Additionally the involved components are also explained with the description.

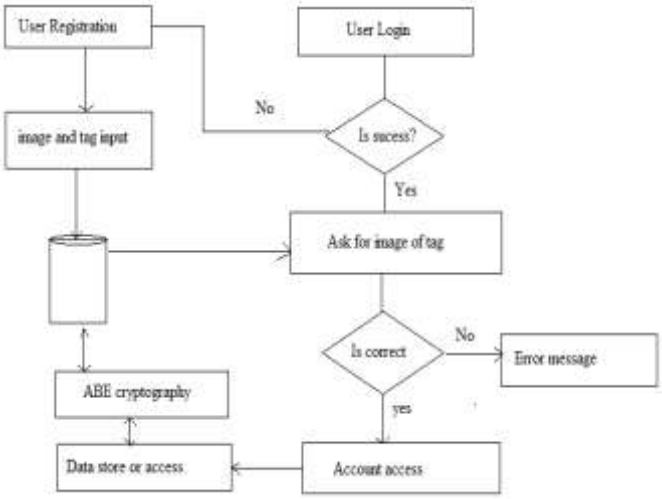


Figure 2.1 proposed system architecture

The proposed system includes an authentication module first to use the service of the secure cloud storage. Therefore a provision is made to verify the identity of the user, if the user not having an account with the server then it is required to create an account.

**User registration:** that is a simple GUI implemented for accepting the user selected credentials which involve the different user attributes such as user name, email id, mobile number and others. User put these credentials according to the information known by the user. After submitting the information on this screen information preserved into the server database and the user account is created.

**Image and tag:** during the user registration process a set of images are also available in the same registration screen. User selects an appropriate image from the list of images available in this screen. Additionally for the selected image user provide a tag or string. This information is also preserved on the server for authentication purpose. After creating account the user can use the cloud storage for hosting of data.

**User login:** in order to access the user account by a registered user it is required to pass the authentication. Therefore first user provides the user id to initialize the authentication process. According to the user id server provide an image which selected by user during the registration. If the displayed image by server is not correct it means either user provides wrong credential or invalid server then user can stop the authentication process. On the other hand for successful authentication it is required to provide a correct string for the image displayed by the server. If user can not recognize the image and the associated tag with image it means the client is invalid and server stop authentication process.

If all the credentials are verified then authentication is completed successfully and user can access their account. On the next process server provides the ability to upload the data to the cloud hosting. Therefore a cryptographic security is also implemented with the system.

**ABE cryptography:** ABE cryptography is known as the attribute based cryptography. Here the term attribute is used because for encryption of the data system usages the data attributes or the user attributes. In this presented design the user attribute is used for performing encryption of data. The figure 2.2 contains the functioning of the proposed attribute based encryption technique.

According to the given diagram when the user provides the file to upload on server, system first select a random attribute from the registration database. The selected attribute can be user's name, mobile number or anything other available attribute in database. That selected attribute is processed using the TIGERHASH algorithm. The TIGERHASH algorithm generates the 128 bit hash code. The TIGERHASH hash code and the input file are used with the AES algorithm to generate the encrypted file. That encrypted file is stored on server for further use.

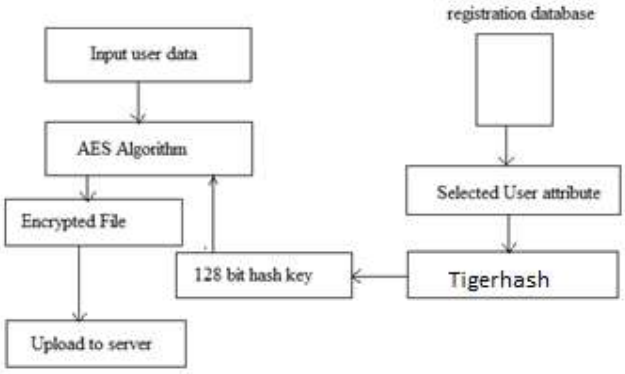


Figure 2.2 proposed cryptographic technique

**C. Proposed Algorithm**

This section provides the summary involved process in methodology. Therefore two different steps of algorithm is provided using table 2.1 and table 2.2 for providing the sequence of authentication and the encryption respectively.

| |
|---|
| Input: user id UID, input image tag T<br>Output: authentication decision (True/false) |
| Process:<br>1. $U = readUserID(UID)$<br>2. $if(U == true)$<br>a. Display image<br>b. $tag = readUserTag(T)$<br>c. $if(tag == true)$<br>i. $D = sucess$<br>d. Else<br>i. $D = retry$<br>e. End if<br>3. Else<br>4. Redirect to registration<br>5. End if<br>6. Return D |

Table 2.1 authentication process

| |
|---|
| Input: user input file F, registration database D<br>Output : encrypted file EF |
| Process:<br>1. $R = readUserData(F)$<br>2. $A = selectrandomAttribute(D)$<br>3. $key = Tigerhash.genrateHash(A)$<br>4. $EF = AES.encrypt(R, key)$<br>5. Return EF |

Table 2.2 encryption process

## III. RESULT ANALYSIS

The implementation of the proposed Attribute based Access Control technique is described in previous chapter. This chapter provides the detailed understanding about the experimental evaluation and performance computation. Therefore essential parameters which are used for evaluation are listed with their observations.

### A. Encryption time

The amount of time required to perform encryption using the selected algorithm is termed as the encryption time complexity of the system. This can be calculated using following formula:

$$\text{Time Consumption} = \text{Algorithm End Time} - \text{Start Time}$$

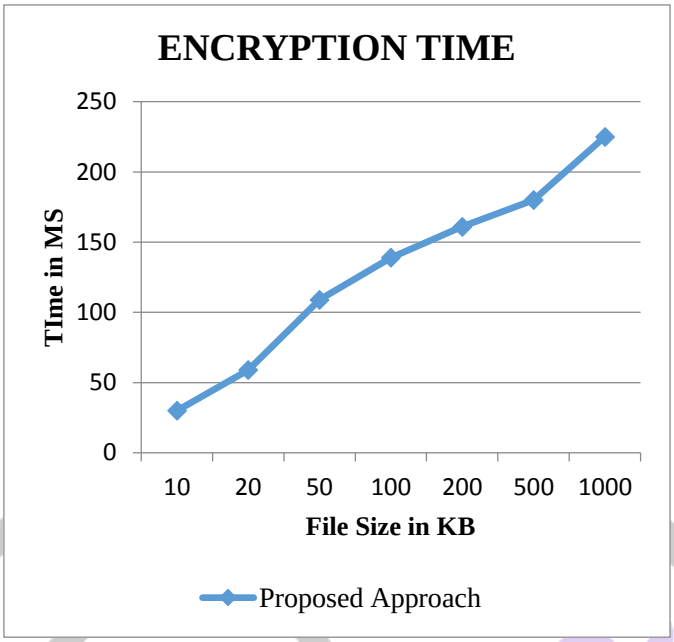


**Figure 3.1 Encryption Time**

The encryption time of the proposed Attribute based access control approach is demonstrated using figure 3.1 and the table 3.1. In this diagram the X axis shows the different file size on which the experimentation is performed, and Y axis contains the amount of time consumed for processing the input data file. The estimated time is given here in terms of milliseconds. Additionally the file size is reported in terms of KB (kilobytes) and performance pf the approach is given using blue line. According to the given results the proposed approach consumes less time to process input file. Additionally the results shows the amount of time consumed is depends on the amount of data provided for execution.

**Table 3.1 Tabular form of Encryption Time**

| S. No. | File size (in KB) | Proposed Approach |
|---|---|---|
| 1. | 10 | 30 |
| 2. | 20 | 59 |
| 3. | 50 | 109 |
| 4. | 100 | 139 |
| 5. | 200 | 161 |
| 6. | 500 | 180 |
| 7. | 1000 | 225 |

### B. Decryption time

The amount of time required to recover the original data from the cipher text is known as the decryption time complexity of the algorithms. The time consumption of the cryptographic algorithm is computed using the following formula.

$$\text{Time Consumption} = \text{Algorithm End Time} - \text{Start Time}$$

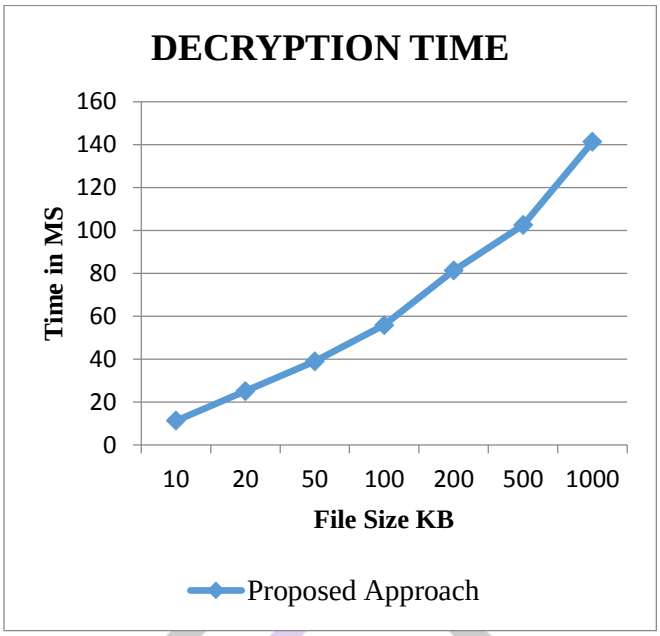


**Figure 3.2 Decryption Time**

The figure 3.2 and table 3.2 shows the obtained performance of the system in terms of decryption time. The time computed of decryption in terms of millisecond. To show the performance of both proposed developed approach is depict using blue line. In given figure 3.2, X axis shows the different file size on which the experiments are performed in terms of kilobytes (KB). Additionally Y axis contains the amount of time consumed in milliseconds (MS). According to the observations decryption time of the proposed algorithm is much adoptable for processing of the file.

**Table 3.2 Tabular form of Decryption Time**

| S. No. | File size (in KB) | Proposed Approach |
|---|---|---|
| 1. | 10 | 11.32 |
| 2. | 20 | 25.14 |
| 3. | 50 | 39.02 |
| 4. | 100 | 55.89 |
| 5. | 200 | 81.32 |
| 6. | 500 | 102.55 |
| 7. | 1000 | 141.36 |

**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.

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

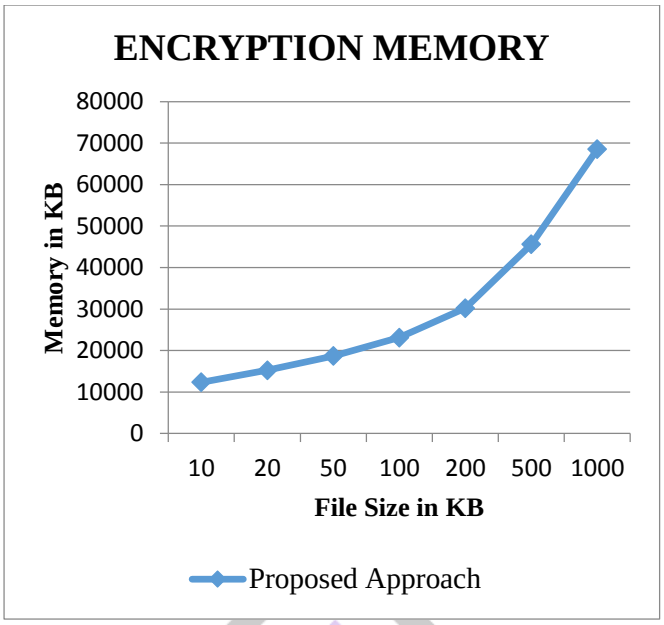


**Figure 3.3 Encryption Memory**

The figure 3.3 and the table 3.3 show the encryption memory or space complexity of encryption algorithm. In this diagram the amount of main memory consumed in terms of kilobytes (KB) is given in Y axis and the file size in terms of kilobytes (KB) which are used for experiments are reported at X axis. According to the obtained results the proposed algorithm consumes lesser resources for better efficiency of the system.

**Table 3.3 Tabular form of Encryption Memory**

| S. No. | File size (in KB) | Proposed Approach |
|---|---|---|
| 1. | 10 | 12365 |
| 2. | 20 | 15235 |
| 3. | 50 | 18663 |
| 4. | 100 | 23114 |
| 5. | 200 | 30215 |
| 6. | 500 | 45612 |
| 7. | 1000 | 68521 |

**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 or the space complexity of the decryption algorithm. The decryption time required is computed using the following formula as the encryption algorithm.

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

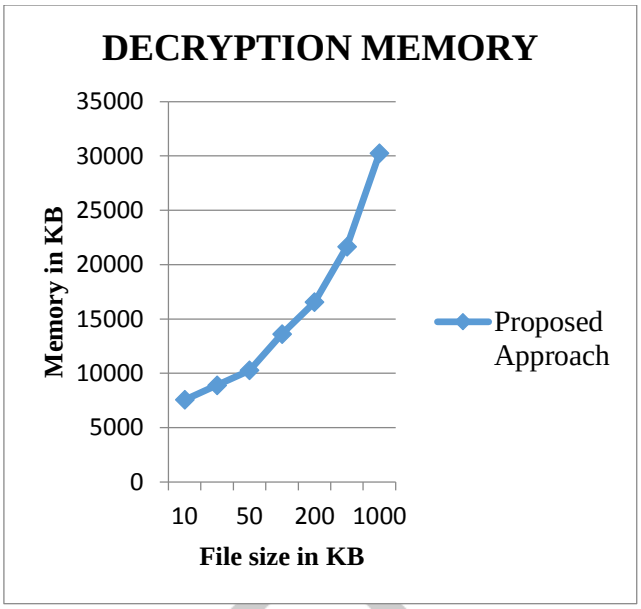


**Figure 3.4 Decryption Memory**

The figure 3.4 and table 3.4 shows the amount of main memory consumed during the data recovery. In this diagram the blue line shows the performance of proposed Attribute based access control approach. The X axis shows the different file size used for decryption in terms of kilobytes (KB) and the Y axis shows the amount of main memory consumed during the decryption in terms of kilobytes (KB). According to the results the proposed algorithm consumes less memory for data recovery.

**Table 3.4 Tabular form of Decryption Memory**

| S. No. | File size (in KB) | Proposed Approach |
|---|---|---|
| 1. | 10 | 7562 |
| 2. | 20 | 8896 |
| 3. | 50 | 10265 |
| 4. | 100 | 13621 |
| 5. | 200 | 16552 |
| 6. | 500 | 21654 |
| 7. | 1000 | 30251 |

## IV. CONCLUSION

This chapter includes the summary of the performed experiments and the observation as the conclusion of the conduced research work. In addition of that the future extension of the work is also included with the work.

### A. Conclusion

Cloud servers involve different services such as data hosting and computing services. The hosting services are aimed to provide the secure and trusted solution for the users to host their confidential and sensitive data without any worries. But due to large scale of data arrival on server it is required to move the data from other servers for preserving the cost of maintenance. Therefore the proposed work aimed to design a efficient and secure data hosting technique which provide the security and confidentiality of data. Basically the cloud servers are not only insecure from internal attack that is also possible an attacker from outside of the server can break the security of cloud. In this context an attacker can make a forge web page which is actually look alike the cloud server page and can used for still the user credentials. Such kind of attack is also studied under the phishing attack scenarios. In this context for validating both the user and server a new kind of authentication system is required.

The proposed technique provides the two different solutions first the implementation of the cryptographic cloud using user attributes. That technique is implemented using the random selection of the user attributes for developing the user key. For developing this key the TIGERHASH algorithm is used to generate the 128 bit encryption key which is again user with the data and the AES encryption algorithm. The secured data is hosted on the cloud server. In addition of that a client server mutual

authentication scheme is developed which includes the image and tag method to recognize by the user. Basically that is a kind of two factor authentication where a part of information is available one party and second part of information available with second party the combination of both the information authenticates both the parties are genuine or not.

The implementation of the proposed technique is performed using the JAVA technology and for implementing it for web the JSP technology is used. After the implementation of the proposed concept the performance of cryptographic technique is measured. The performance is summarized using table 4.1.

| S. No. | Parameters | Remark |
|---|---|---|
| 1 | Encryption time | The encryption time is depends upon the data which is need to be encrypt using the proposed technique |
| 2 | Decryption time | The decryption time is also depends on the data amount but it is less than the encryption time |
| 3 | Encryption memory | The encryption memory is directly depends on the amount of data and it is acceptable |
| 4 | Decryption memory | The less memory consumption is noticed that is less in amount as compared to encryption scenario |

Table 4.1 performance summary

According to the obtained performance the proposed technique found for efficient therefore that is acceptable for different cryptographic applications as well as the cloud data security.

**B. Future Work**

The main aim of the proposed work is to demonstrate the attribute based encryption technique with a trusted authentication system. The obtained performance of the system shows the method is effective securing the cloud data and provides authenticity for user data access. In near future the following extension of the work is possible for work.

1. The proposed work currently authenticate the user and server both but that is need to involve the biometric attributes too for providing more secure authentication system.
2. The current system only considers a single party data placement on server and access it in near future it is need to extend the technique for multiparty data hosting and access management for data.
3. The system is also extendable for the data sharing technique over the cloud for different access control technique

REFERENCES

[1] Guoyuan Lin, Danru Wang, Yuyu Bie, and Min Lei, "MTBAC: a mutual trust based access control model in cloud computing", China Communications 11, Number 4, (2014): pp. 154-162
[2] Mell and T. Grance, “Draft NIST Working Definition of Cloud Computing,” 2009; http://csrc.nist.gov/groups/SNS/cloud-computing/index.html
[3] Armbrust, Michael, Armando Fox and Rean Griffith, “Above the clouds: A berkeley view of cloud computing”, Volume 17, Technical Report UCB/EECS-2009-28, EECS Department, University of California, Berkeley, 2009.
[4] Cloud computing for e-governance, White paper, IIIT-Hyderabad, January 2010, Available online (13 pages).
[5] Vaishali Jain and Akshita Sharma, “A Taxonomy on Cloud Computing”, International Journal of Emerging Technology and Advanced Engineering, Certified Journal, Volume 4, Issue 3, March 2014
[6] M. Armbrust et al., “Above the Clouds: A Berkeley View of Cloud Computing,” Univ. California, Berkeley, Tech. Rep. UCBEECS-2009-28, Feb. 2009.
[7] “Cloud Computing Architecture”, available online at: http://communication.howstuffworks.com/cloudcomputingl.htm
[8] J.E. Smith and R. Nair, “An overview of virtual machine architectures”, pages 1– 20, October 2001, http://www.ece.wisc.edu/~jes/902/papers/intro.pdf
[9] V. Abricksen, “A Survey on Cloud Computing and Cloud Security Issues”, International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622International Conference on Humming Bird (01st March 2014).
[10] Mohammad Asadullah and R. K. Choudhary, “Data Outsourcing Security Issues and Introduction of DOSaaS in Cloud Computing”, International Journal of Computer Applications (IJCA), PP. 40-45, Volume 85 – No 18, January 2014.
[11] Protect Data Privacy, http://www- 01.ibm.com/software/data/optim/protect-data-privacy/.
[12] OpenCrowd. Opencrowd cloud taxonomy. http://www.opencrowd.com/views/cloud.php, 2009.Ro