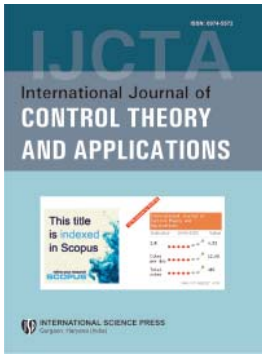

International Journal of Control Theory and Applications

ISSN : 0974–5572

© International Science Press

Volume 9 • Number 40 • 2016

A Reliable Auditing Model with Secure Cloud Storage

Pritam Debnath[a] and S. Jagadeesan[b]

[a]*PG student, School of Computing, CSE Department, SRM University, Kattankulathur Chennai, India*

E-mail: pritamvtu@gmail.com

[b]*Assistant Professor, School of Computing, CSE Department, SRM University, Kattankulathur Chennai, India*

E-mail: Jagadeesan81@gmail.com

Abstract : The major concern in cloud is its security. Due to the rapid development and digitalization of this modern world many users have focused on storing their data in cloud. So it has become highly necessary to focus on these security issues in cloud. So here in my system we have proposed a own-auditing system in cloud, Using which the file owner can easily find out whether his shared file has been accessed or edited by the shared user or not. The file uploaded by the file owner is securely stored in the cloud using the secure erasure code. The file uploaded breaks into many fragments and gets stored in many server location. The encryption has been done to the fragmented parts to provide a safe storage of the data. Permission of the file owner is required for downloading the file for which a private key is generated and sent to the file owner which has to be applied while downloading the file.

Keywords: *Security, cloud, own-auditing system, encryption, private key.*

1. INTRODUCTION

Cloud in few years has become a major area of interest and also the Interest among the users from various parts of the world has increased in the cloud storage [7]. All want their data to be safely stored in cloud for their future use. When we look deeply we can easily find out that the accuracy or the consistency of the data in cloud is missing.

So we proposed a secure erasure code, which splits the file in many fragments and store the fragment in different locations. Advanced encryption standard has been applied over the fragmented data for safe storage which blocks the access to the data by the fraud users [10].

The file owner share their data with the different users but he does not know whether his data has been really accessed or edited by the shared users for which we have introduced own-auditing system. Using the own-auditing system the file owner can easily come to know about the status of their data which is stored in cloud. Once the file owner uploads his personal file on the cloud, token A gets created and after the shared users edit the shared files token B gets created. Now these both token A and B is compared with each other, if there is a mismatch in the token notification message is sent to the file owner asking for his permission for the approval of the file. Once after the final approval by the file owner the file stored permanently in the cloud else the edited file gets deleted. The token is generated using the random ASCII values of the text data in the file.

2. PROBLEM STATEMENT

The existing technique of storing the data in cloud does not completely block the unauthorized person from the access to the private data. For auditing, the file owner's private information was given to the third party. There is a chance that the private information of file owner gets misused in different ways.

3. RELATED WORK

The authors have proposed a powerful mechanism which provides the privacy of the data. To design the coherent keyword search method, they designed variable searchable encryption process [1].

The authors have proposed a powerful basic Evident of provable data possession, which Utilize the Diffie-Hellman shared key to manufacture the homomorphism authenticator [2].

The authors have proposed a protected distributed storage framework supporting security saving open inspecting [3].

The authors have proposed a methodical reliable PPDP protocol which uses the bilinear pairing [4].

The author have proposed an exceedingly particularly deal to provably make safe PDP framework to develop through the symmetric key cryptography by not needing any huge encryption process [5].

The author proposed a paper which provided information on the encryption technique and about the auditing mechanism [6].

The authors also have provided the information of the TPA (Third Party Auditing) [11, 12].

4. OWN-AUDITING SYSTEM

Two token is produced, one after the uploading process of the original file by the file owner and other when the shared user edit the file which is uploaded and shared by the file owner to the shared users. These two token is compared with each other, if a mismatch found on the token than a notification message is directly send to the file owner asking for his approval. Once approved the edited file gets stored permanently.

5. TOKEN GENERATION

For example:

Original file = HOW ARE YOU

Taking ASCII values of text

H = 72,
O = 79,
W = 87,
A = 65,
R = 82,
E = 69,
Y = 89,
O = 79,
U = 85

Adding we get:

72 + 79 + 87 + 65 + 82 + 69 + 89 + 79 + 85 = 707 (Token A is generated).

Edited file = WHAT ARE YOU

Taking ASCII values of text

W = 87,
H = 72,
A = 65,
T = 84,
A = 65,
R = 82,
E = 69,
Y = 89,
O = 79,
U = 85

Adding we get:

87 + 72 + 65 + 84 + 65 + 82 + 69 + 89 + 79 + 85 = 777 (Token B is generated).

707 is not equal to 777, so there is a mismatch in the token which proves that the data has been changed. Hence therefore the notification message is sent to the file owner.

6. PROPOSED MODEL

The proposed model is shown here in different phases

6.1. System Architecture

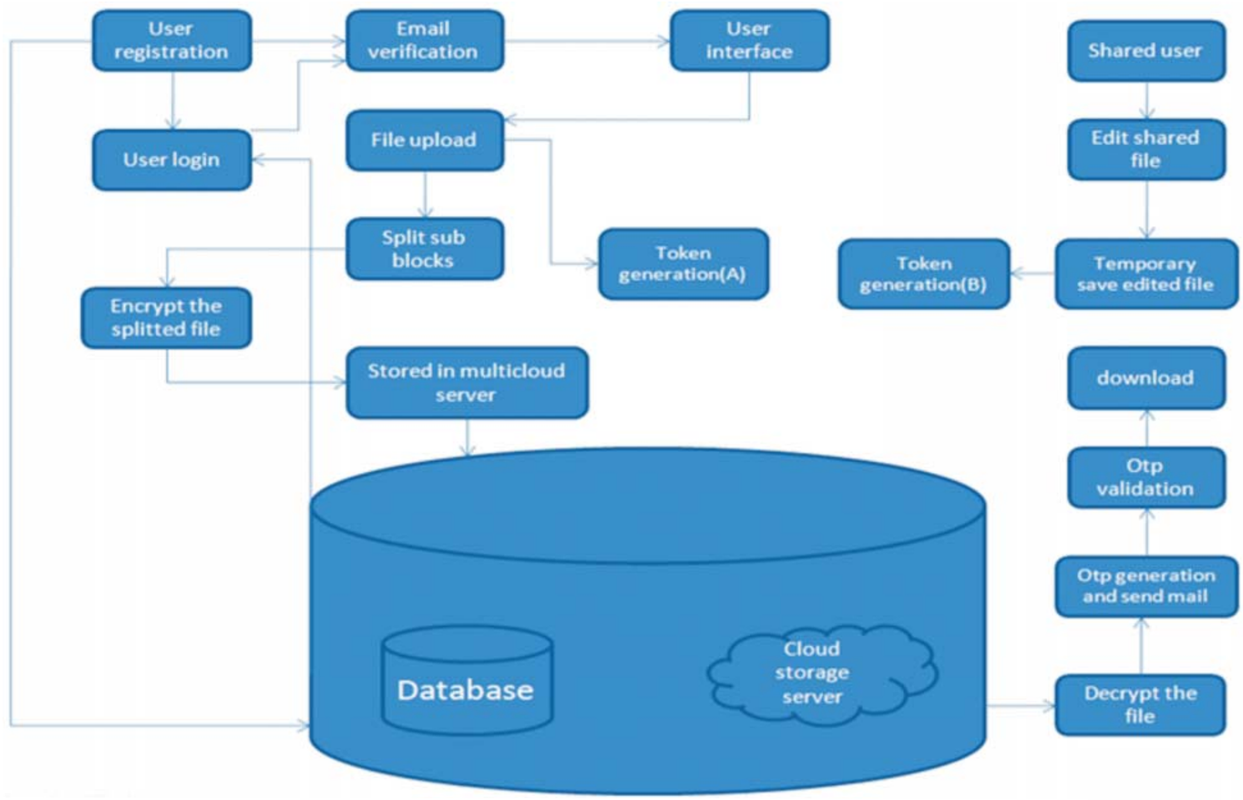

Figure 1: System Architecture

6.2. User interface

A simple user interface is proposed in order to have a user friendly communication. Another feature added in the user interface is the easy searching of the file technique. At the time of uploading process a keyword will be asked these keywords can be used instead of giving the full name of the file which is to be searched.

6.3. Private Key Generation

A private key is produced and send to the file owner, this private key will be required while downloading the file. So without the file owner permission no other person can download the file.

6.4. File Uploading

The file uploaded by the file owner to the cloud breaks into many fragments and gets stored in different locations. The fragments are encrypted using the Advanced Encryption Standard (AES) for safe storage.

This is done to protect the data from being accessed by the unauthorized users.

6.5. File sharing

The file owner can share his file with different users, he can also provide the permission to edit the file. The shared users can view, edit and download the file. Here it uses the Share Key Gen (SKA, *t*, *m*).

6.6. File auditing:

Checking the status of the file shared with the shared users is auditing. Here we have introduced a own-auditing system, using which the file owner himself can audit the file, there is no need of the third party to audit the file.

6.7. Mail alert process

While downloading the file a private key will be generated and send to the file owner by a mail alert process. This private key is needed to be applied while downloading the file. And also if the shared users edit the file notification message will be send to the file owner through the mail alert process.

6.8. File downloading process

The different encrypted fragment stored in different location will get decrypted and combine together to form the original message. This original message can be downloaded applying the private key which has been send to the file owner.

7. ALGORITHMS

SECURE ERASURE CODE

```
    Begin ;
→   slf and pass;
    Based: = It is based upon the simple scheme in the field of cloud computing.
    selfname = slf && password = pass
    Then
→   If( pkey == dfile )
    Data upload k;
```

```
k    → part 1, part 2, part 3;
     → Encryption & decryption with Advanced Encryption Standard
en   → ecyp1, ecyp2, ecyp3
de   → dcyp1, dcyp2, dcyp3
     data downloading  yu;
     serverfile from database & server
     if(yu==serverfile)
     pkey → mail send to file owner (key).
     Add oriignal data → (part 1 + part2 + part 3)
     Download the data.
→    Else
     Cancel the file;
     End;
```

8. MATHEMATICAL MODEL

1. Initialize Tokens
   a) Bt = {}
   b) Qt = {}
2. Files uploaded in the Cloud
   a) Fi = {}
3. method encryption module

   Encryp = *tp*, *uid_otn*

   Where *tp* ε Fi

   uid_otn ε QT
4. Method Decryption module
5. Decryp = Tc, *uid_otn*

   Where Tc ε Encryp
6. Files Encrypted which will be produced from the equation

   $S(\text{Encryp}) = \sum^{fn}_{n+1} tp\wedge uid\_\text{QT}$

   Number of files in the set Fi={} is *n*, *fp* is the plain text file and *uid*_QT is the Authorization token of the user.
7. The Original files which will be produced from the equation is

   $S(\text{Dn}) = \sum^{fn}_{n+1} tp\wedge uid\_\text{QT}$

   Number of files in a set Fi = {} is *n*, *fc* is the cipher text file and *uid*_OT is the Authorization token.

9. ADVANCED ENCRYPTION STANDARD:

9.1 Description

Advanced encryption standard (AES) is used for the encryption of the text. It provides protection to the data from being accessed by the fraud users [8].

The key length here is 128, 192, and 256 bits. AES has three blocks of ciphers, AES-128, AES-192 and AES-256[9].

9.2 The Features of AES

Cipher block is symmetric.

Block length of 128 bit.

The length of the key is 128, 192, 256 bits.

It is quicker than Triple-DES.

10. CONCLUSION

The files are securely stored in the cloud using the secure erasure code, which breaks the file into many fragments. The encryption is also applied to the fragments for the safe storage. Own-auditing system is introduced which helps to audit the file by self instead of giving it to the third party. Hence a reliable auditing model is formed.

REFERENCE

[1] Z. Fu, X. Sun, Q. Liu, L.Zhou, J. Shu. Achieving efficient cloud search services: multi-keyword ranked search over encrypted cloud data supporting parallel computing. IEICE Transactions on Communications. 2015; vol. E98.B, no. 1, pp.190-200.

[2] Y. Ren, J. Shen, J. Wang, J. Han, S. Lee. Mutual verifiable provable data auditing in public cloud storage. Journal of Internet Technology. 2015; vol. 16, no. 2, pp. 317-323.

[3] C. Wang, Q. Wang, K. Ren, W. Lou. Privacy-Preserving Public Auditing for Data storage security in cloud computing. Proceeding IEEE INFOCOM. 2010; pp. 525-533.

[4] H. Wang. Proxy provable data possession in public clouds. IEEE Transactions on Services Computing. 2013; vol. 6, no. 4, pp. 551-559.

[5] G.Ateniese, R.DiPietro, L. V. Mancini, G.Tsudik. Scalable and efficient provable data possession. SecureComm 2008.

[6] M. B. Jayalekshmi, S. H. Krishnaveni. A Study of data storage security issues in cloud computing. Indian Journal of Science & Technology. 2015.

[7] M. Armbrust, A. Fox, R. Griffith, A.D. Joseph, R.H. Katz, A. Konwinski, G. Lee, D.A. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, “A View of Cloud Computing,” Comm. ACM, vol. 53, no. 4, pp. 50-58, Apr. 2010.

[8] Poonam.M.Pardeshi,Deepali Borade, “Enhancing Data Dynamics and Storage Security for Cloud Computing using Merkle Hash Tree and AES Algorithms”,International Journal of Computer Applications; Vol. 98, p1, Jul 2014.

[9] P. Anitha, V.palanisamy. Data protection algorithm using AES. international journal of current research. 2011; vol. 3, issue, 6, pp.291-294.

[10] P. Debnath, S. Jagadeesan, “A Survey on a Reliable Method to Achieve Cohesion and Possession of the Data in Cloud”, International Journal of Pharmacy & Technology, 2016: vol. 8, no. 4, pp. 5251-5256.

[11] B. Wang, B. Li, and H. Li, “Oruta: Privacy-Preserving Public Auditing for Shared Data in the Cloud,” Proc. IEEE Fifth Int’l Conf.Cloud Computing, pp. 295-302, 2012.

[12] M.S.Shashi Dhara, “Privacy Preserving Third Party Auditing In Multi Cloud Storage Environment.”, IEEE International Conference on Cloud Computing in Emerging Markets (CCEM),pp.1-6,2014.