MJRET
Open Access
Multidisciplinary Journal of Research in Engineering and Technology
Journal homepage: www.mjret.in
ISSN:2348 - 6953

SECURING SHARED DATA OVER PUBLIC CLOUD USING REAL-TIME ENCRYPTION AND OAUTH TOKEN

Mr. Suryakant Kadam[1], Prof. P. M. Agarkar[2]
Department of Computer Engineering, Savitribai Phule Pune University
Dr. D. Y. Patil School of Engineering
Pune, India

Abstract: *In today's Computing world Cloud enrolling is one of the best advancement which utilizes advanced computational power and it improves data sharing and data securing limits. Key inconvenience in dispersed processing was issues of data reliability, data security and data access by unapproved customers. TTA (Trusted Third Party) is used to store and share data in dispersed registering. Change and sharing of data is totally essential as a social event. To check uprightness of the shared data, people in the social affair needs to figure blemishes on all common data squares. Various pieces in shared data are all things considered set apart by differing customers in view of data changes performed by particular customers. Customer repudiation is one of the best security perils in data sharing in get-togethers. In the midst of customer foreswearing shared data piece set apart by repudiated customer needs to download and re-sign by existing customer. This errand is particularly inefficacious due to the broad size of shared data pieces on cloud. PANDA Plus is the new open assessing instrument for the keeping up respectability of conferred data to successful customer revocation in the cloud. This segment relies on upon middle person re-marks thought which allows the cloud to re-sign squares on advantage of existing customers in the midst of customer repudiation so that downloading of shared data squares is not required. PANDA other than is the overall public controller which surveys the reliability of shared data without recouping the entire data from the cloud. It also screen bunch to affirm different surveying endeavors at the same time. For user revocation or data revocation system does not relies on group manager it does automatically over public cloud.*

Keywords: *PANDA, TPA, Cloud, DB, CSP, TTP.*

1. INTRODUCTION

Today information stockpiling and sharing administrations, (for example, Dropbox what's more, Google Drive) gave by the cloud, individuals can without much of a stretch cooperate as a sharing so as to gather information with one another. All the more particularly, once a client makes shared information in the cloud, each client in the gathering cannot just get to and alter shared information, additionally share the most recent adaptation of the mutual information with whatever is left of the gathering[1].Despite the fact that cloud suppliers guarantee a more secure and dependable environment to the clients, the uprightness of information in the cloud might still be traded off, because of the presence of equipment/programming disappointments and human blunders [2], [3]. To secure the uprightness of information in the cloud, various components have been proposed. In these components, a mark is connected to every piece in information, and the respectability of information depends on the rightness of the considerable number of marks. One of the most huge and normal elements of these systems is to permit an open verifier to productively check information respectability in the cloud without downloading the whole information, alluded to as open examining (or meant as Provable Data Ownership [3]). This open verifier could be a customer who might want to use cloud information for specific purposes (e.g., seek, calculation, information mining, and so forth.) or an outsider inspector (TPA) why should capable give check administrations on information trustworthiness to clients. The greater part of the past works concentrate on evaluating the uprightness of individual information. Not the same as these works, a few late works concentrate on the best way to safeguard personality protection from open verifiers while evaluating the uprightness of shared information. Sadly, nothing from what was just mentioned components, considers the effectiveness of client disavowal while reviewing the accuracy of shared information in the cloud. With shared information, once a client changes a square, she likewise needs to figure another mark for the changed square. Because of the adjustments from various clients, distinctive squares are marked by various clients. For security reasons, when a client leaves the gathering or acts mischievously, this client must be renounced from the gathering. Accordingly, this renounced client ought to never again have the capacity to get to and adjust shared information, and the marks produced by this renounced client are no more legitimate to the gathering [15]. Consequently, despite the fact that the substance of shared information is not changed amid client renouncement, the pieces, which were beforehand marked by the renounced client, still should be re-marked by a current client in the gathering. Subsequently, the honesty of the whole information can at present be confirmed with people in general keys of existing clients just. Unmistakably, if the cloud could have every client's private key, it can without much of a stretch complete the re-marking undertaking for existing clients without requesting that they download and re-sign pieces. Be that as it may, subsequent to the cloud is not in the same trusted space with every client in the gathering, outsourcing each client's private key to the cloud would present critical security issues. Another vital issue we have to consider is that the re-calculation of any mark amid client renouncement should not influence the most appealing property of open inspecting examining information trustworthiness openly without recovering the whole information. Accordingly, how to proficiently diminish the huge weight to existing clients presented by client repudiation, what's more, still permit an open verifier to check the trustworthiness of shared information without downloading the whole information from the cloud, is a testing reran.

2. RELATED WORK

There are some assorted frameworks which used as a piece of different inspecting frameworks. This range introduce some the frameworks like MAC, HLA et cetera which are used for various purposes like data approval, data respectability in assessing arrangements on cloud[1]. The framework model appeared in Figure 1 incorporates three elements: the cloud, the general population verifier, and clients (who offer information as a gathering). The cloud offers information stockpiling also, sharing administrations to the gathering. The general population verifier, for example, a customer who might want to use cloud information for specific purposes (e.g., seek, calculation, information mining, and so on.) or an outsider reviewer who can give check administrations on information uprightness, plans to check the respectability of shared information through a test and-reaction convention with the cloud[2]. In the gathering, there is one unique client and a number of gathering clients. The first client is the first proprietor of information. This unique client makes and imparts information to other clients in the gathering through the cloud. Both the first client also, aggregate clients can get to, download and adjust shared information. Shared information is separated into various squares [1].

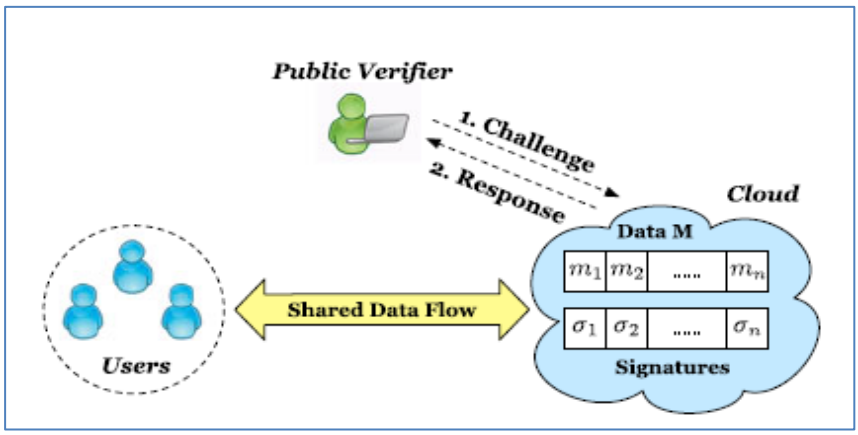

Figure.1. The system model includes the cloud, the public verifier, and users [1].

A client in the gathering can alter a piece in shared information by performing an addition, erase or overhaul operation on the square. We accept the cloud itself is semi trusted, which implies it takes after conventions and does not contaminate information uprightness effectively as a vindictive enemy, be that as it may, it might mislead verifiers about the error of shared information with a specific end goal to spare the notoriety of its information benefits and abstain from losing cash on its information administrations. In expansion, we additionally expect there is no intrigue between the cloud and any client amid the configuration of our instrument [1]. By and large, the mistake of offer information under the above semi-trusted model can be presented by equipment/programming disappointments or human mistakes happened in the cloud. Considering these elements, clients don't completely believe the cloud with the respectability of shared information. To secure the uprightness of shared information, every piece in imparted information is connected to a mark, which is registered by one of the clients in the gathering. In particular, when shared information is at first made by the first client in the cloud, all the marks on shared information are registered by the first client. After that, once a client alters a piece, this client too requirements to sign the altered piece with his/her own particular private key. By sharing information among a gathering of clients, distinctive squares might be marked by various

clients because of adjustments from various clients. For the most part, as the maker of shared information, the first client goes about as the gathering director what's more, can repudiate clients in the interest of the gathering. Once a client is repudiated, the marks figured by this denied client get to be invalid to the gathering, and the hinders that were beforehand marked by this repudiated client ought to be re-marked by a current client's private key, so that the accuracy of the whole information can in any case be checked with people in general keys of existing clients just. Elective methodology permitting each client in the gathering to offer a typical gathering private key and sign every square with it, is likewise a conceivable approach to secure the respectability of shared information [11], [12]. In any case, when a client is disavowed, a new gathering private key should be safely circulated to each current client and every one of the squares in the common information must be re-marked with the new private key, which builds the intricacy of key administration and diminishes the proficiency of client renouncement.

3. PROPOSED SYSTEM

The proposed model of this project is as shown in the figure 2 which consists of four main phase as follows,

1. OAuth Login and Token Extraction
2. Key-Generation
3. Real-Time Encryption
4. User Revocation

With shared data, once a user modifies a block, she also needs to compute a new signature for the modified block. Due to the modifications from different users, different blocks are signed by different users. For security reasons, when a user leaves the group or misbehaves, this user must be revoked from the group. As a result, this revoked user should no longer be able to access and modify shared data, and the signatures generated by this revoked user are no longer valid to the group.

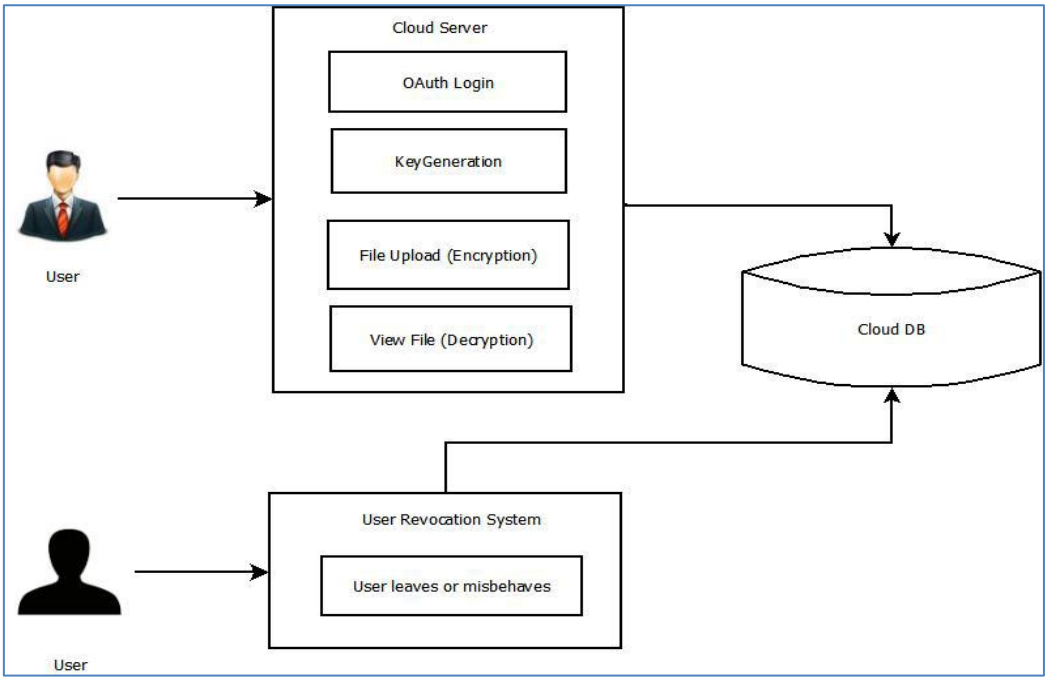

Figure2. Overview of Proposed System

A. Problem Definition

Data integrity is one of the critical issues; as there is lack of identity privacy because of relinquish trends in cloud, where the users are unacquainted with the auditor of the data, over geographically scattered datacenters. This features of cloud computing evolved various concerns related to users identity, data integrity and users availability. Ultimately this influences to propose an enhanced model in order to audit the data integrity and keeping the identity privacy with efficient user revocation while sharing.

B. Mathematical Model

1. Initialize Tokens
   (a) At={}
   (b) Ot={}
2. Initialize path/files upload to Cloud
   F = {}
3. Process encryption module
   En=fp,uid_otn
   Where fp Ɛ F
   uid_otn Ɛ OT
4. Decryption module D=Fc,uid otn
   Where Fc Ɛ En
5. Encrypted files obtained by equation

   $$S(En)= \sum_{n+1}^{fn} fp\text{^}uid\_OT$$

   Where n is total number of files in a file set F={}, fp is the plain text file and uid_OT is a user Authorization token

6. Original files obtained by equation

   $$S(Dn)= \sum_{n+1}^{fn} fp\text{^}uid\_OT$$

   Where n is total number of files in a file set F= {}, fc is the cipher text file and uid_OT is a user Authorization token

C. Algorithmic Strategies

Algorithms for OAuth Authorization Server

Input: User Credentials

Output: Authentication token & authorization token

The following steps explain the server-side flow:

1. Start
2. Obtain an access token.
3. User decides whether to grant access to your application
4. OAuth Server redirects user to your application
5. Exchange authorization code for refresh and access tokens.
6. Process response and store tokens

7. Stop

The following steps explain the client-side flow:

1. Start
2. Obtain an access token.
3. Server decides whether to grant access to your application
4. OAuth Server redirects user to your application
5. Validate the user's token
6. Process the token validation response.
7. Stop

D. Real Time Encryption Algorithm

Encryption Steps:

1. Start
2. Retrieve OAuth token at successful user login
3. Generate random key using key generator
4. Read data from file and XoRing with the key
5. Add key in the XoRed data, which generated by key generator
6. Write encrypted data in file and load file to Cloud Storage
7. Stop

Decryption Steps:

1. Start
2. Retrieve data to decrypt
3. Extract key from data using key generator
4. Read data from file and XoRing with the key
5. Pass decrypted data to User
6. Write output to result file and load file to Cloud Storage.
7. Stop

4. EXPERIMENTAL SETUP AND RESULTS

1. Dataset

As over public cloud user upload its personal data in the form of files and these file having different formats like txt, doc, pdf or it may contains images, audio or video. Before data load to cloud storage it encrypted using real time encryption technique. Data processes by system is a user input file and user authentication token retrieved at authentication process by OAuth server.

2. *Results*

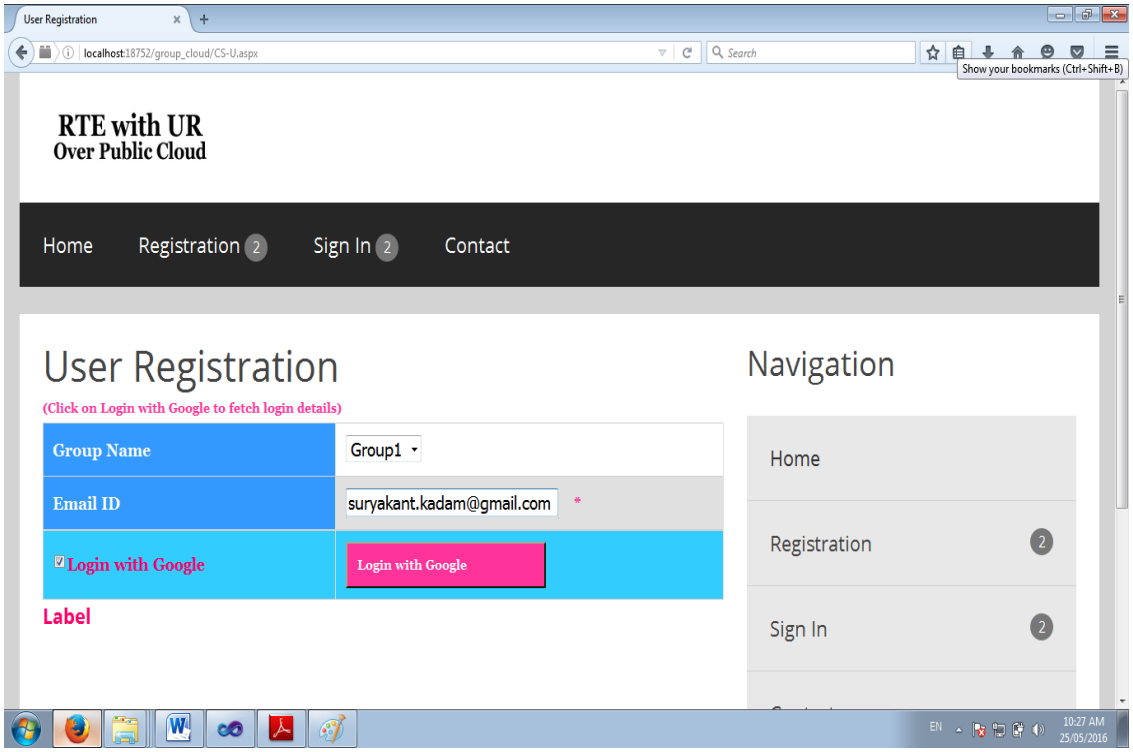

Figure3. System Login using Google OAuth Server

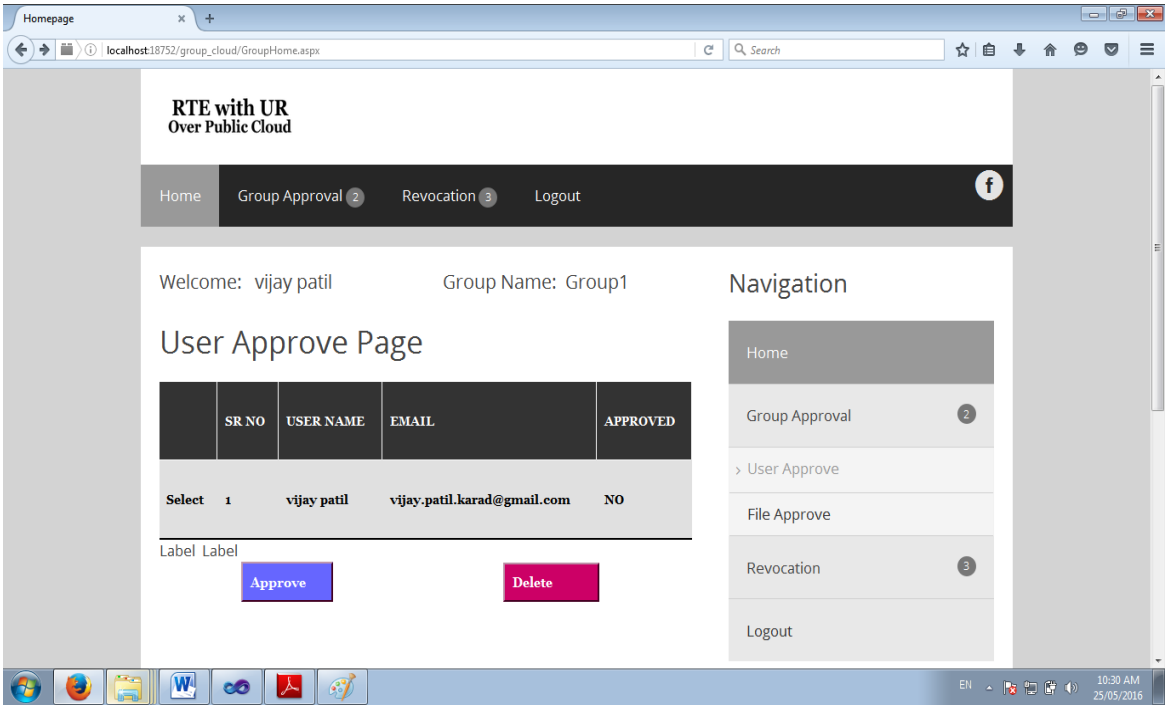

Figure4 Group Administrator Interface to Approve User

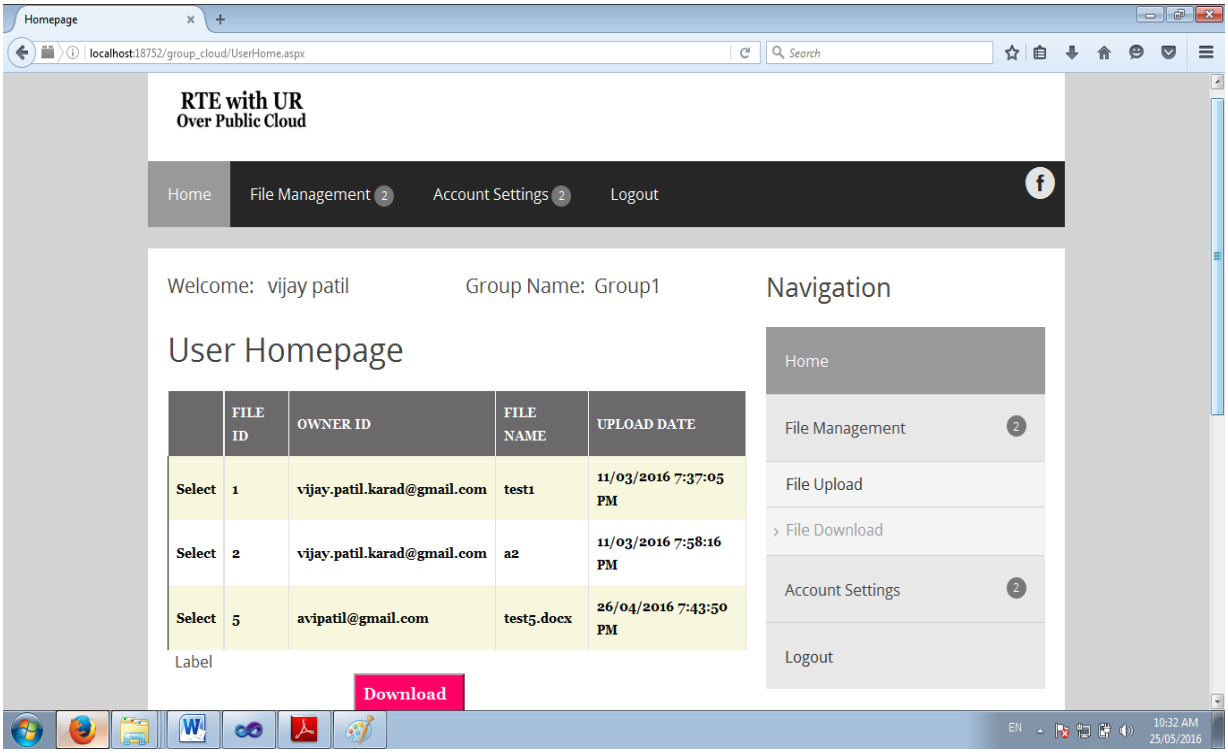

Figure5 Group Administrator Interface to Approve File

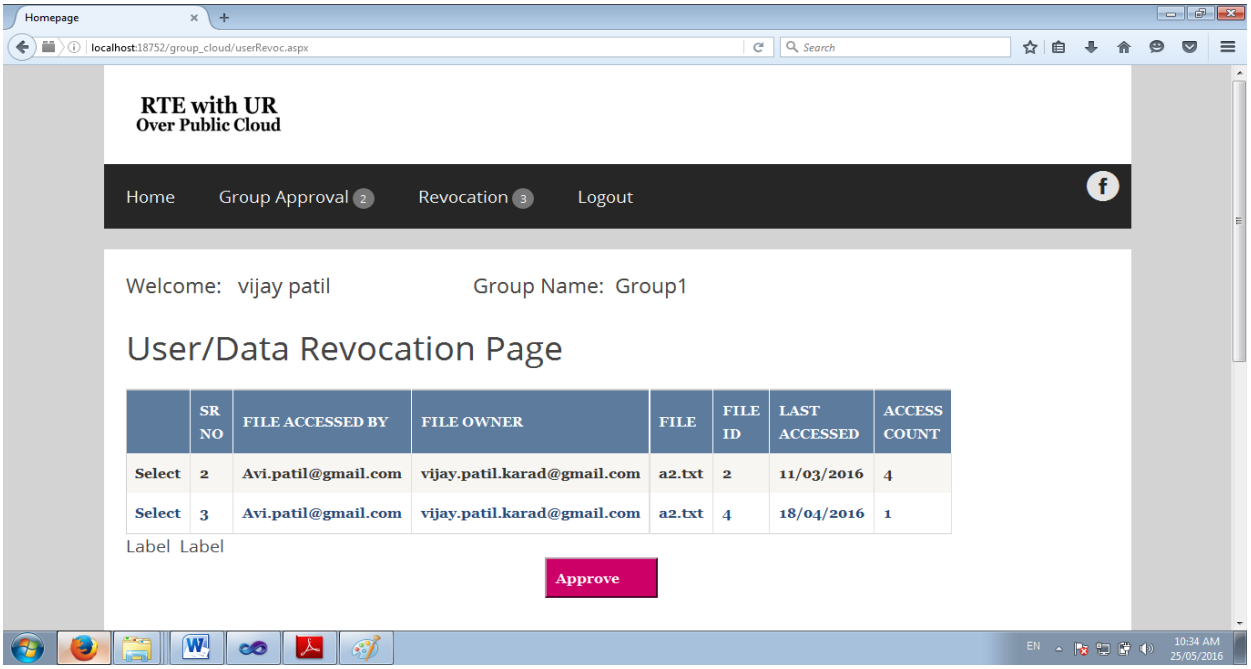

Figure6 Group Administrator Interface to Revocation of User/File

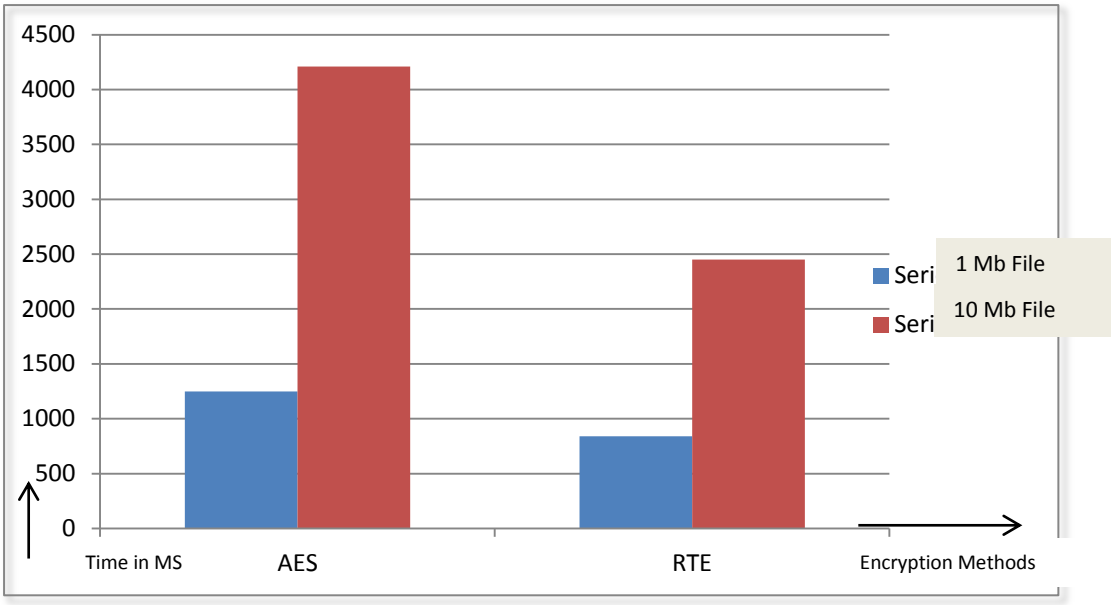

Figure7.Impact of real time encryption on file uploads.

The comparison of RTE and AES shown in above figure; where time required to uploading 1 MB file with RTE is 840 MS and 1250 MS using AES. When file of 10 MB uploaded by user then 2450 MS and 4210 MS required by RTE and AES respectively.

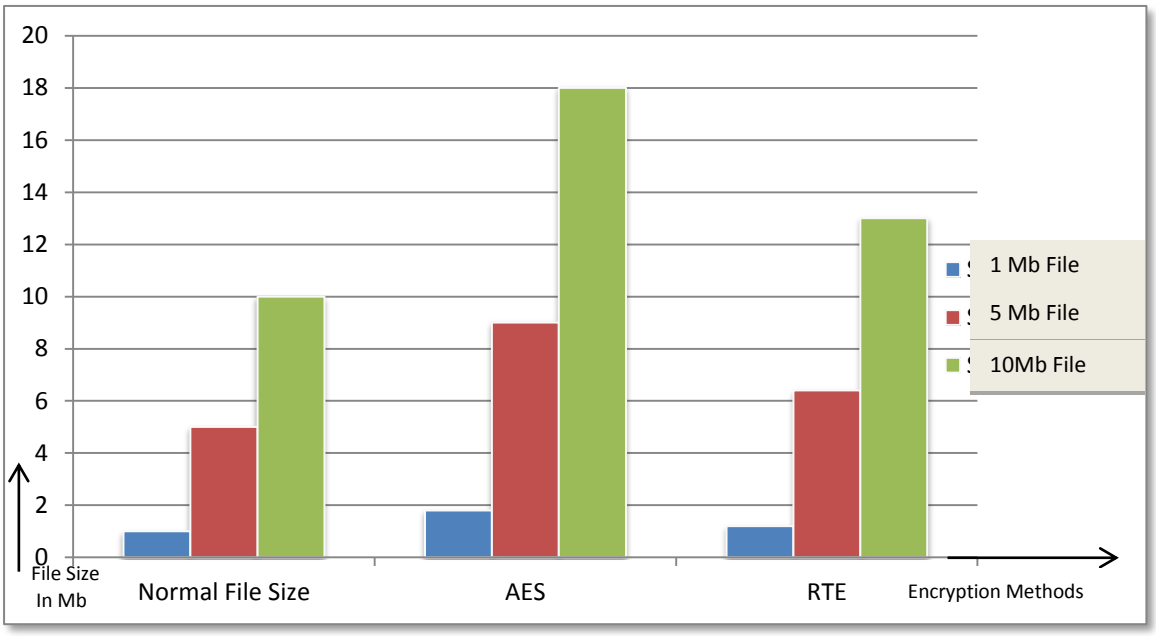

Figure8. Impact on cloud storage after file encryption

The comparison of File size of RTE and AES shown in above figure; where 1 MB file increased up to 1.8 MB 1.2 MB by RTE and AES respectively similarly 5 MB and 10 MB increased up to 9 MB and 18 MB by using AES and 6.4 MB and 13 MB by using RTE. Cloud used by different user on pay per use basis so storage saving reduce the cost of cloud use of uses.

5. CONCLUSION

Cloud computing is world's biggest innovation which uses advanced computational power an improve data sharing and data storing capabilities. It increases the ease of usage by giving access through any kind of internet connection. OAuth authentication is good choice to improve security of cloud system. OAuth server provide authorization token which used in encryption technique. The RTE technique improves speed of data encryption as well as overcome the problem of collusion. When a user in the group is revoked, we allow the semi-trusted cloud to re-sign blocks that were signed by the revoked user with proxy re-signatures. Experimental results show that the cloud can improve the encryption speed and data upload time and also prove the space required to load data at cloud storage.

REFERENCES

[1] *Boyang Wang, IEEE, Baochun Li and Hui Li, "Panda: Public Auditing for Shared Data with Efficient User Revocation in the Cloud", IEEE transactions on services computing, vol. 8, no. 1, january/february 2015*

[2] *C. Wang, Q. Wang, K. Ren, and W. Lou, "Towards Secure and Dependable Storage Services in Cloud Computing," IEEE Transactions on Services Computing, vol. 5, no. 2, pp. 220–232, 2011.*

[3] *Y. Zhu, G.-J. Ahn, H. Hu, S. S. Yau, H. G. An, and S. Chen, "Dynamic Audit Services for Outsourced Storage in Clouds," IEEE Transactions on Services Computing*

[4] *S. Marium, Q. Nazir, A. Ahmed, S. Ahthasham and Aamir M. Mirza, "Implementation of EAP with RSA for Enhancing The Security of Cloud Computig", International Journal of Basic and Applied Science, vol 1, no. 3, pp. 177-183, 2012*

[5] *Balkrishnan. S, Saranya. G, Shobana. S and Karthikeyan.S, "Introducing Effective Third Party Auditing (TPA) for Data Storage Security in Cloud", International Journal of computer science and Technology, vol. 2, no. 2, ISSN 2229-4333 (Print) | ISSN: 0976- 8491(Online), June 2012*

[6] *K. Kiran Kumar, K. Padmaja, P. Radha Krishna, "Automatic Protocol Blocker for Privacy-Preserving Public Auditing in Cloud Computing", International Journal of Computer science and Technology, vol. 3 pp, ISSN. 0976-8491(Online), pp. 936-940, ISSN: 2229-4333 (Print), March 2012*

[7] *Jachak K. B., Korde S. K., Ghorpade P. P. and Gagare G. J.,"Homomorphic Authentication with Random Masking Technique Ensuring Privacy & Security in Cloud Computing", Bioinfo Security Informatics, vol. 2, no. 2, pp. 49-52, ISSN. 2249-9423, 12 April 2012*

[8] *https://developers.google.com/identity/protocols/OAuth2*

[9] *H. Wang, "Proxy Provable Data Possession in Public Clouds," IEEE Trans. Services Computing, vol. 6, no. 4, pp. 551-559, Oct.- Dec. 2013.*

[10] *B. Wang, B. Li, and H. Li, "Oruta: Privacy-Preserving PublicAuditing for Shared Data in the Cloud," Proc. IEEE CLOUD,pp. 295-302, 2012.*

[11] *S.R. Tate, R. Vishwanathan, and L. Everhart, "Multi-User Dynamic Proofs of Data Possession Using Trusted Hardware," Proc. Third ACM Conf. Data and Application Security and Privacy (CODASPY'13), pp. 353-364, 2013.*

[12] *B. Wang, B. Li, and H. Li, "Knox: Privacy-Preserving Auditing for Shared Data with Large Groups in the Cloud," Proc. 10th Int'l Conf. Applied Cryptography and Network Security (ACNS'12), pp. 507-525, June 2012.*

[13] *M. Blaze, G. Bleumer, and M. Strauss, "Divertible Protocols and Atomic Proxy Cryptography," Proc. Int'l Conf. the Theory and Application of Cryptographic Techniques (EUROCRYPT'98), pp. 127-144, 1998.*

[14] *A. Shamir, "How to Share a Secret," Comm. ACM, vol. 22, no. 11,pp. 612-613, Nov. 1979.*

[15] *B. Wang, H. Li, and M. Li, "Privacy-Preserving Public Auditingfor Shared Cloud Data Supporting Group Dynamics," IEEE Int'l Conf. Comm. (ICC'13), pp. 1946-1950, June 2013.*