



Implementation of Data Privacy Framework for Public Cloud Storage

Apeksha A. Dhone¹, Prof. Avinash P. Wadhe²

¹Department of Computer Science & Engineering, G.H. Rasoni College of Engineering & Management, Amravati, Maharashtra, India

²Department of Computer Science & Engineering, G.H. Rasoni College of Engineering & Management, Amravati, Maharashtra, India

Abstract —Distributed computing is an Internet based registering which empowers sharing of administrations. Utilizing Cloud Storage, clients can remotely store their information and appreciate the on-request great applications and administrations from a mutual pool of configurable registering assets. Security in cloud is accomplished by marking the information hinder before sending to the cloud. Also, clients ought to have the capacity to simply utilize the distributed storage as though it is nearby, without agonizing over the need to confirm its honesty. Along these lines, empowering open auditability for distributed storage is of basic significance so clients can turn to a Third Party Auditor to check the uprightness of outsourced information and that outsourced information to ensure in distributed storage against debasements, adding adaptation to internal failure to distributed storage together with information respectability checking and disappointment reparation ends up plainly basic. Numerous instruments managing the trustworthiness of outsourced information without a nearby duplicate have been proposed under various framework and security models up to now. The most critical work among these reviews are the provable information ownership model and confirmation of retrievability model, which were initially proposed for the single-server situation separately. Considering that documents are generally striped and repetitively put away crosswise over multi-servers or multi-mists, investigate uprightness check plans appropriate for such multi-servers or multi mists setting with various excess plans, for example, replication, eradication codes and all the more as of late, recovering codes. The overhead of utilizing distributed storage ought to be limited however much as could reasonably be expected with the end goal that a client does not have to perform excessively numerous operations to their outsourced information.

Keywords- Cloud Storage, Regenerating Codes, Public Audit, Privacy Preserving, Authenticator Regeneration, Proxy, Privileged, and Third Party Auditor

I. INTRODUCTION

All Cloud storage is now gaining popularity because it offers a flexible on-demand data outsourcing service with appealing benefits relief of the burden for storage management, universal data access with location independence, and avoidance of capital expenditure on hardware, software, and personal maintenances, etc. [1]. Nevertheless, this new paradigm of data hosting service also brings new security threats toward users data, thus making individuals or enterprisers still feel hesitant. It is noted that data owners lose ultimate control over the fate of their outsourced data; thus, the correctness, availability and integrity of the data are being put at risk. On the one hand, the cloud service is usually faced with a broad range of internal/external adversaries, who would maliciously delete or corrupt users' data; on the other hand, the cloud service providers may act dishonestly, attempting to hide data loss or corruption and claiming that the files are still correctly stored in the cloud for reputation or monetary reasons. Thus it makes great sense for users to implement an efficient protocol to perform periodical verifications of their outsourced data to ensure that the cloud indeed maintains their data correctly[5]. To protect Outsourced data in cloud storage against corruption adding fault tolerance to cloud storage together with data integrity checking and failure reparation becomes critical.

Cloud Storage, users can remotely store their data and enjoy the on-demand high quality applications and services from a shared pool of configurable computing resources, without the burden of local data storage and maintenance. Regenerating codes have gained popularity due to their lower repair bandwidth while providing fault tolerance. Public auditing scheme for the regenerating-code-based cloud storage[8]. Users should be able to just use the cloud storage as if it is local, without worrying about the need to verify its integrity. Cloud storage is of critical importance so that users can resort to a third party auditor (TPA) to check the integrity of outsourced data and be worry-free. Considering the large size of the outsourced data and the user's constrained resource capability, the tasks of auditing and reparation in the cloud can be formidable and expensive for the users. The overhead of using cloud storage should be minimized as much as possible such that a user does not need to perform too many operations to their outsourced data.

The most fundamental services is offered by cloud providers was data storage. Let's consider a limited data application the company allows its staffs in the same group or department to stored and shared files in the cloud. By utilizing the cloud that the staffs could be completely released from the troublesome local data storehouse and maintenance however, it is also poses a significant risk to the confidentiality of those stored files. Specifically the cloud servers are managed by

cloud providers is not fully trusted by users while the data files stored in the cloud might be confidential and sensitive such as business plans. To preserve data privacy is primary solution for encrypt data files and then uploaded the encrypted data into the cloud. The designing of the efficient and secure data sharing scheme for groups in the clouds is not an easy task due to the following challenging issues. First of all identity the privacy is being one of the most significant restriction for the wide deployment of cloud computing. Here not holding the guaranteed of identity privacy user may be unwilling to append in cloud computing systems because their real identities can be easily disclose to cloud providers and also attackers. On the other hand its unconditional identity privacy might incur the abuse of privacy for example the misconduct staff could deceive others on the company to sharing false files without being traceable. Therefore, traceability and which are enables the TPA to expose the real identity of a user's are also highly desirable. Second, it is highly recommended that any member in the groups should able to fully enjoy the data storing as well as sharing services provided by the cloud which are defined as the multiple owner manner. Compare with the single owner manner where only the group manager could store and modify data in the cloud, the multiple owner manners are more flexible in practical applications. The integrity of data in cloud storage, however, is subject to as data stored in an untrusted cloud can easily be lost or corrupted, due to hardware failures and human errors[2]. To protect the integrity of cloud data, it is best to perform public auditing by introducing a third party auditor (TPA), who offers its auditing service with more powerful computation and communication abilities than regular users.

1.1 Project objective

Proposed system address the problem of privacy preserving deduplication in cloud computing and propose a new deduplication system supporting for

1. Convergent encryption has been proposed to enforce data confidentiality while making deduplication feasible. The user is only allowed to perform the duplicate check for files marked with the corresponding privileges.
2. Cloud architecture is introduced to solve the problem of private cloud.
3. Each authorized user is able to get his/her individual token of his file to perform duplicate check based on his privileges. Under this assumption, any user cannot generate a token for duplicate check out of his privileges or without the aid from the private cloud server.
4. Authorized user is able to use his/her individual private keys to generate query for certain file and the privileges he or she owned with the help of private cloud, while the public cloud performs duplicate check directly and tells the user if there is any duplicate.
5. Unauthorized users without appropriate privileges or file should be prevented from getting or generating the file tokens for duplicate check of any file stored at the storage cloud server provider.
6. It requires that any user without querying the private cloud server for some file token, he cannot get any useful information from the token, which includes the file information or the privilege information.
7. Unauthorized users without appropriate privileges or files, including the storage cloud service provider and the private cloud server is prevented from access to the underlying plaintext stored at storage cloud service provider.

The main objectives of the proposed system the study are listed below:

- To calculate the time of communication of cloud data storage.
- To generate the security for the TPA.
- To provide the time of encryption and decryption for security analysis
- To provide the time of computation and communication.

II. LITERATURE REVIEW

2.1 Background:

A survey is done to propose a new method to offer a better Regarding-code-based cloud storage using Privacy-Preserving public auditing. This system is public auditing for regenerating code and it does not require data owner to always stay online Regarding codes have gained popularity due to their lower repair bandwidth while providing fault tolerance.[1] We motivate the public auditing system of data storage security in Cloud Computing and provide a privacy-preserving auditing protocol. Our scheme enables an external auditor to audit user's cloud data without learning the data content.[2] In this paper, we focus on the integrity verification problem in regenerating-code-based cloud storage, especially with the functional repair strategy. Considering the large size of the outsourced data and the user's constrained resource capability, the tasks of auditing and reparation in the cloud can be dangerous and expensive for the users.

2.2 Literature review

It is the technique to analyze storage architectures that combine any form of coding and replication, as well as presenting two new schemes for maintaining redundancy using erasure codes. It minimize the amount of bandwidth used to maintain that redundancy. Storing a file using an erasure code, in fragments spread across nodes, promises to require less redundancy and hence less maintenance bandwidth than simple replication to provide the same level of reliability.[1]

In this paper, focuses on combination the public key based homomorphic authenticator with random masking to achieve the privacy-preserving public cloud data auditing system. It support efficient handling of multiple auditing tasks. They further explore the technique of bilinear aggregate signature to extend result into a multi-user setting, where TPA can perform multiple auditing tasks simultaneously.[2]

In this paper a flexible distributed storage integrity auditing mechanism, utilizing the homomorphism token and distributed erasure-coded data. proposed design allows users to audit the cloud storage with very lightweight communication and computation cost. The proposed design further supports secure and efficient dynamic operations on outsourced data, including block modification, deletion, and append. It proposed scheme is highly efficient and resilient against malicious data modification attack, and even server colluding attacks.[3].

In this paper, we propose a new privacy preserving public auditing mechanism for shared data in an untrusted cloud. Here, we utilize ring signature so that the third party auditor is able to verify the integrity of shared data for a group of users without retrieving the entire data while the identity of the signer on each block in shared data is kept private from the TPA. This paper provides a privacy preserving public auditing scheme that supports public auditing and identity privacy on shared data stored in the cloud storage service for enhancing its security and efficiency. This paper has mainly concentrated on improving the security mechanism of own Cloud storage service[4].

III. METHODOLOGY

3.1 Implementation Plan

The implementation plan includes all the activities that must occur to implement the new system and put it into operation. It identifies the personnel responsible for the activities and prepares a time chart for implementing the proposed system. The implementation plan consists of the following steps.

- List all the data required for implementation.
- Identify all data required to generate report during implementation.
- List all product details that are entered into the new system.

3.2 Implementation of Login

As there were numerous clients of cloud every client will have a different login to enter in. At the point when the client enters in they need to get in with their client id and secret key. The accuracy of the client id and secret key will be checked, on the off chance that it is a right client it permits in else it shows a message wrong client. In the event that the client is new then the client needs to fill enrollment shape by clicking join they on the off chance that they are new to utilize this they need to join. By filling enrollment frame they will be redirected to page that contain a few points of interest like username, secret key, email id and organization name which the new client needs to top off. By clicking register the page again backpedal to Login page. The new client can login now with the new client id and secret word. On the off chance that the client fills enrollment shapes with a similar detail that client is not registries since it as of now get registries on the grounds that each time the client id and secret word get checked climate that client id and watchword substantial are or not.

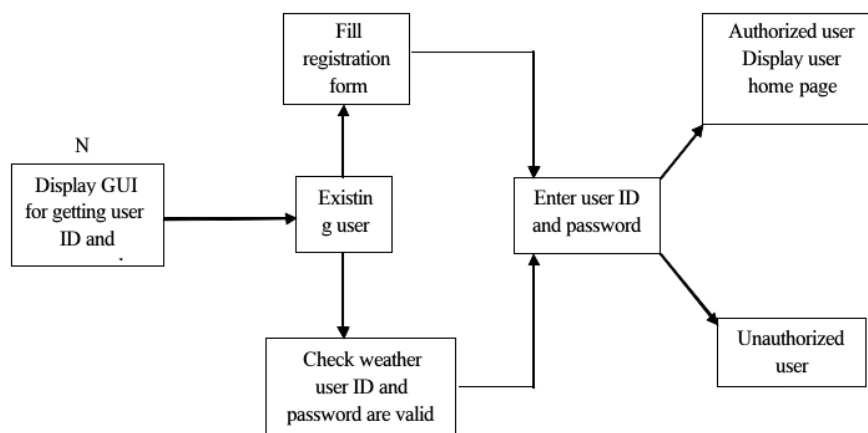


Figure 1:- Implementation of login

3.3 Implementation of Differential Authorization

Each approved client can get his/her individual token of his record to perform copy check in light of his benefits. The private keys for benefits won't be issued to clients straightforwardly, which will be kept and overseen by the private cloud server. Along these lines, the clients can't share these private keys of benefits in this proposed development, which implies that it can keep the benefit enter sharing among clients in the above clear development. To get a record token, the client needs to send a demand to the private cloud server. On the off chance that it is approved client private cloud gives get to appropriate to the client. It requires that any client without questioning the private cloud server for some document token, he can't get any helpful data from the token, which incorporates the record data or the benefit data.

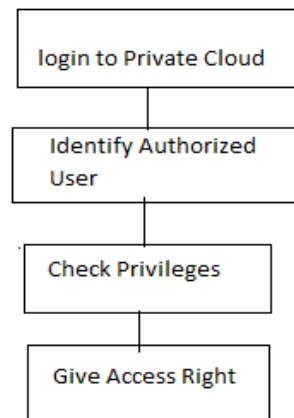


Figure 2:-Implementation of Differential Authorization

3.4 Implementation of Authorized Duplicate checker

To play out the copy check for some record, the client needs to get the document token from the private cloud server. The private cloud server will likewise check the client's personality before issuing the relating document token to the client. The approved copy check for this record can be performed by the client with general society cloud before transferring this document. In light of the aftereffects of copy check, the client either transfers this document. In the event that copy record happen client can nat ready to transfer document. On the off chance that copy record does not happen then client can ready to transfer document. This document will be put away in broad daylight cloud in a scrambled arrangement.

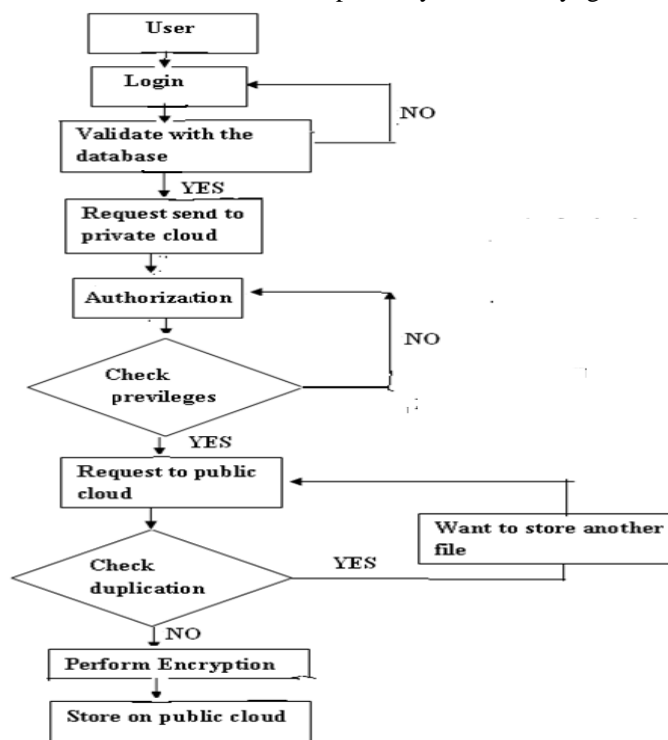


Figure 3:- Implementation of Authorized Duplicate check

3.5 System Implementation

1. This is the client module. This is Home page here we can login the new user for registration. If the user already register then login from the same page. In that below page, here to fill the details of the particular user.

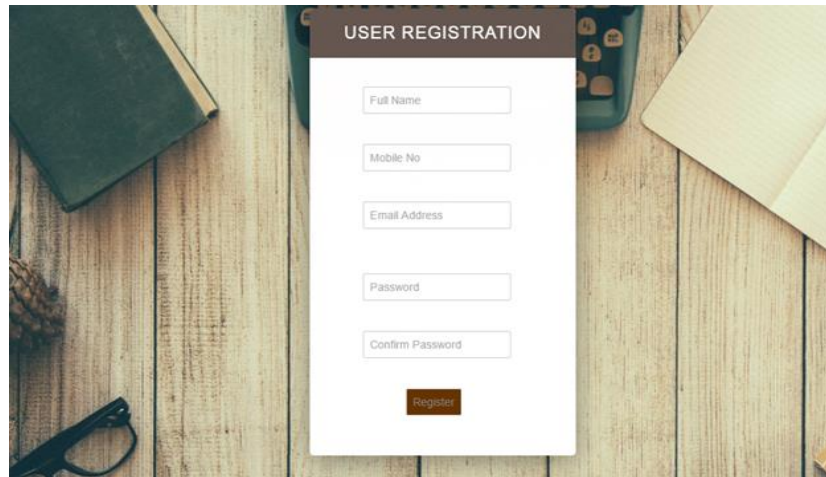
A screenshot of a 'USER REGISTRATION' form. The form is white with a dark brown header bar containing the title 'USER REGISTRATION'. It features five input fields: 'Full Name', 'Mobile No', 'Email Address', 'Password', and 'Confirm Password'. Below these fields is a dark brown 'register' button. The form is set against a background of a wooden desk with a green book, a calculator, and a pair of glasses.

Figure 4. Home page

2. In that below page, after user can be register on the registration page. This log in page can be open. This is the client module. This is Login page here we can login the new user for registration.

A screenshot of a 'LOG IN' form. The form is white with a dark brown header bar containing the title 'LOG IN'. It features two input fields: 'USERNAME' (containing 'monika@gmail.com') and 'PASSWORD'. Below these fields are two dark brown buttons: 'SignIn' and 'SignUp'. The form is set against a background of a wooden desk with a green book, a calculator, and a pair of glasses.

Figure 5. Login page

3. In this phase, firstly here the admin can be login and after that new user complete register and the and request to admin for update rights, upload data and download data.

A screenshot of an 'Admin login' form. The form is white with a dark red header bar containing the title 'Data Privacy'. It features two input fields: 'UserId' and 'Password'. Below these fields is a dark red 'SignIn' button and a 'Label' button. The form is set against a background of a wooden desk with a green book, a calculator, and a pair of glasses.

Figure 6 Admin login

4. In this phase the admin login and view the request of user. In this page to check the how many users can send request to admin for updates rights. And admin can be disable and enabled the particular user for the further process can be proceed.

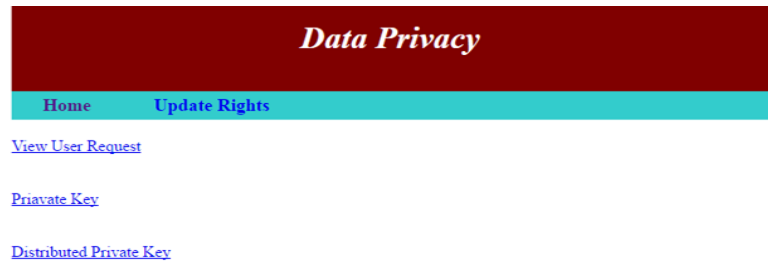


Figure 7 Admin View request

5. In this phase admin view the request and give right to update rights, upload data and download data.

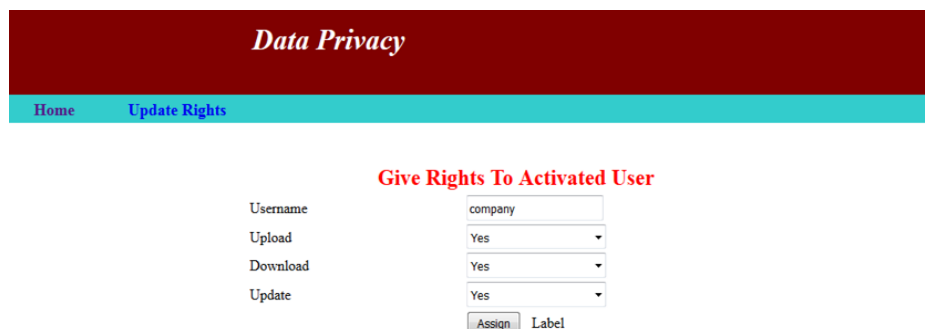


Figure 8 Admin Assign Rights

6. Here the admin assign the right to the user to upload, download and update .After that the user login again and view the rights. When admin can be edit that option after that user can be used that's rights for update, upload and download the data.

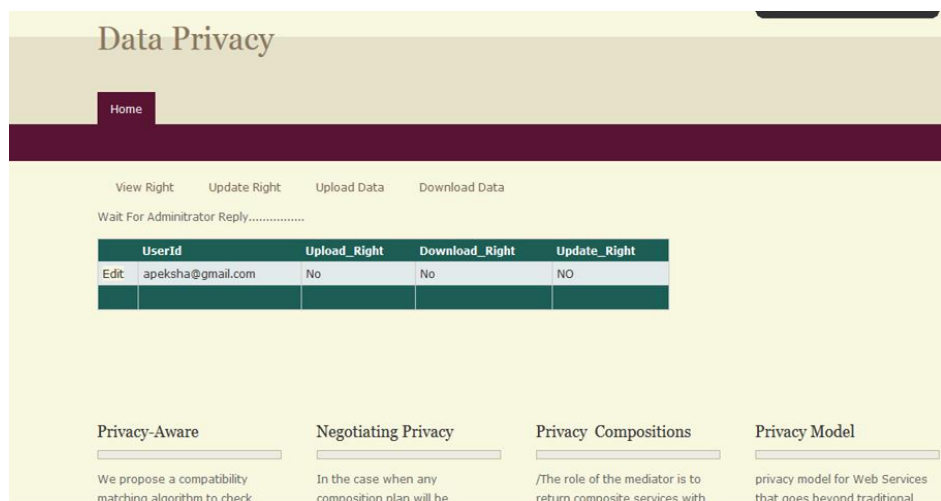


Figure 9 User View Rights

7. In this phase user view there right. And there are three rights also can be used i.e. update, upload and download respectively. For that first attached the file and after that generate token for particular file. And time of encryption also can be done just like in below screenshot.

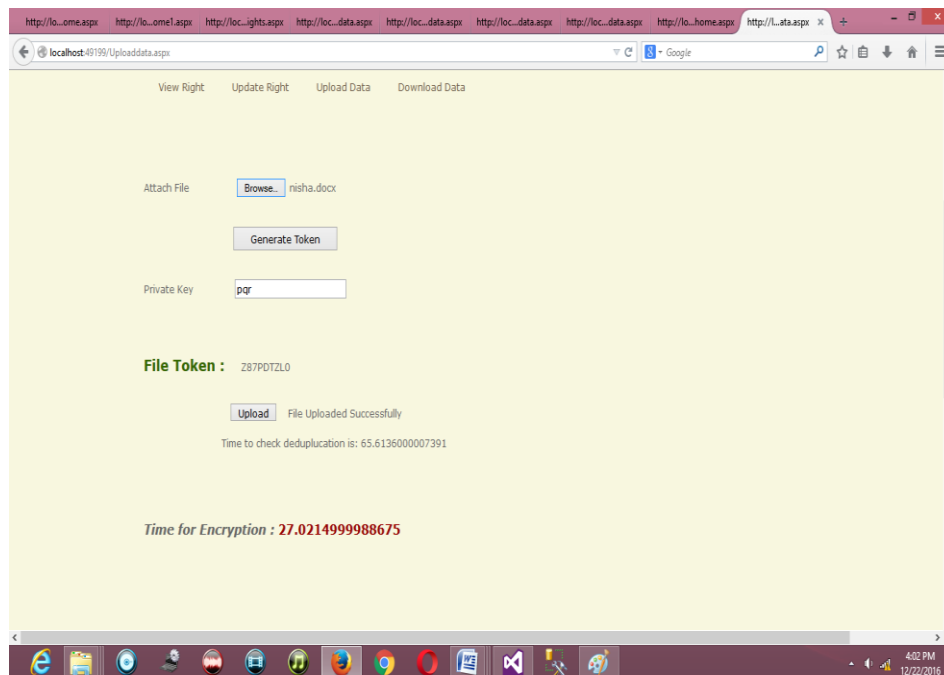


Figure 10 User Upload File

8. In this frame, the new user download the file then the file is downloaded the cloud. And show the time to check duplication and time for encryption. In case the contents are same in different two files then one message can be generate in red colour, "Dulpicate File Content".

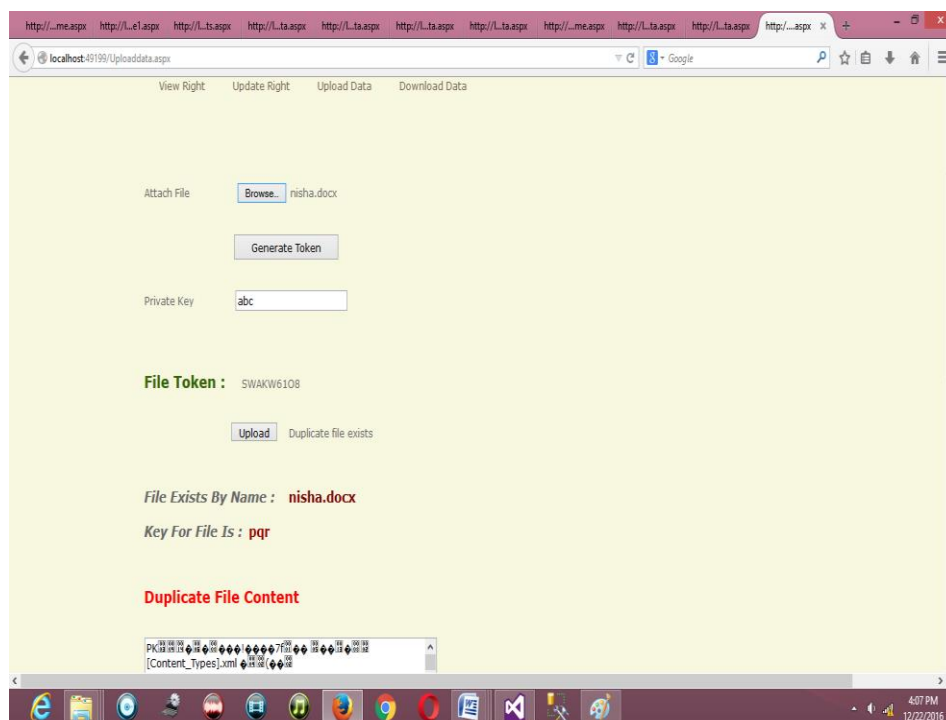


Figure 11 Check Duplication

IV. RESULT

We execute the proposed approved deduplication framework, in which we show three substances as partitioned projects. Microsoft Azure cloud is utilized to store information. A Client program is utilized to demonstrate the information clients to do the document transfer handle. A Private Server program is utilized to display the private cloud which deals with the keys and handles the record token calculation.

Assessment of the proposed framework concentrates on contrasting the overhead incited by approval steps, including document token era and the encryption against record transfer steps. The transfer procedure breakdown into three stages Token Generation, Duplicate Check and Encryption. For each progression, we record the begin and end time of it and in this manner get the breakdown of the aggregate time spent. We exhibit the normal time taken in every informational index in the figures. To assess the impact of record size to the time spent on various strides, we transfer one of a kind documents. The time spent on encryption straightly increment with the record estimate, interestingly, different strides, for example, token era and copy check just utilize the document metadata for calculation and in this manner the time spent stays steady. As appeared in table 1 with the document measure expanding from 50 KB to 200KB, the overhead of the proposed approval steps diminishes from 1.75% to 1.43%.

Table 1:- Time for different File size

File Size	Token Gen(ms)	Duplicate Check(ms)	Encryption(ms)
50 kb	59	113	143
100 kb	61	115	151
150 kb	66	118	159
200 kb	69	123	175

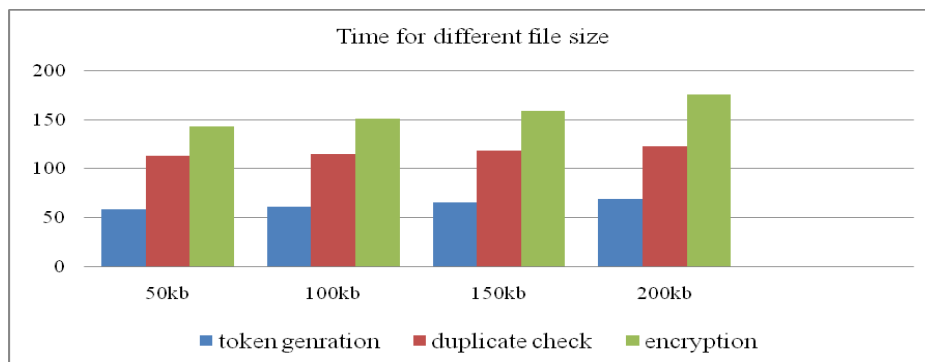


Fig 12:- Time for different file size

Result shown in table 3 is the file size after deduplication it correspondence to results shown in figure 3. Four files of different size of .txt files are chosen for deduplication. The set of sample file along with their copies are chosen for deduplication. Result shown in Table 2 is the file size before deduplication it correspond to result shown in figure 2. While Result shown in Table 3 is the file size after deduplication it corresponds to result shown in figure 3. By applying deduplication approach on these files we are able to save the cloud storage space up to 50%.

Table 2:- Different File size before Deduplication. In that table, Four files of different size of .txt files are chosen for deduplication. The set of sample file along with their copies are chosen for deduplication. Result shown in below Table is the file size before deduplication it correspond to result shown in figure 12.

File Name	File Size in byte
Deepu.txt	3736
Graph.txt	3291
Abc.txt	2301
File1.txt	3007
Total Size	12335

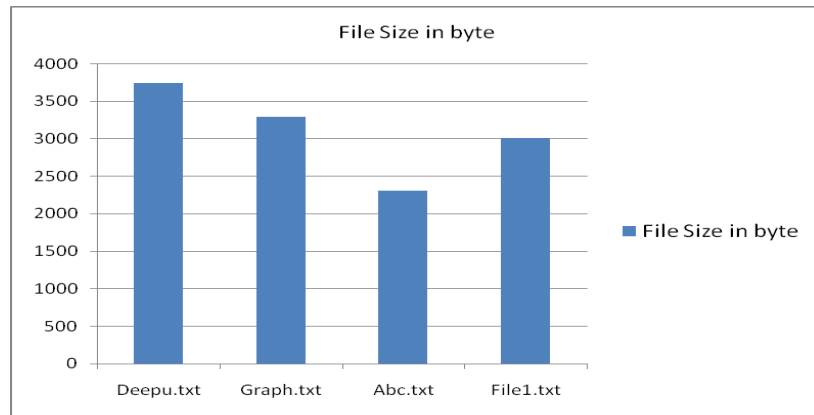


Fig 13:- Different File size before Deduplication

Table 3:- Different File size After Deduplication. In that below table, Result shown is in the file size after deduplication it corresponds to result shown in figure 13. By applying deduplication approach on these files we are able to save the cloud storage space up to 50%.

File Name	File Size in byte
Deepu.txt	536
Graph.txt	3001
Abc.txt	309
File1.txt	607
Total Size	4453

Total space saved= 12335-4453= 7882

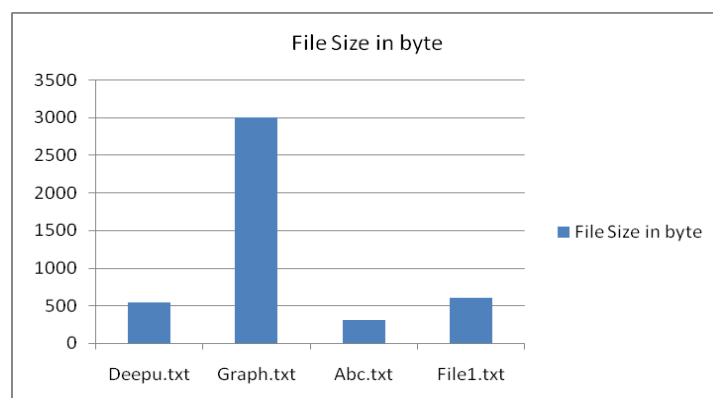


Figure 14:- Different File size After Deduplication

In above graph, four .txt file can be compare size wise and after that the graph can be create. Files can be used in that graph is in the byte. Just like as an above figure.

IV. CONCLUSION

In this venture, we propose a protection saving open evaluating framework for information stockpiling security in Cloud Computing. In this paper, utilize the Third Party Auditor(TPA)to keep up to pointless excess this issue here, we are giving open examining process for distributed storage that clients can make utilization of an outsider inspector (TPA) to check the uprightness of information. Not just confirmation of information honesty, the proposed framework likewise bolsters information. Considering TPA may simultaneously deal with numerous review sessions from various clients for their information records which are outsourced, we additionally amplify our protection safeguarding open inspecting assention into a multi-client setting, where the TPA can play out the different examining undertakings in a clump way for better proficiency. Broad investigation demonstrates that our plans are provably secure and exceptionally proficient.

REFERENCES

- [1] D. Boneh and M. Franklin, "Identity-based encryption from the Weil pairing," in *Advances in Cryptology*. Berlin, Germany: Springer-Verlag, vol. 32. Feb. 2001, pp. 213–229.
- [2] FarzadSabahi Student Member, IEEE, "Cloud Computing Security Threats and Responses", Vol. 11. No. 6, Mar 2006, pp. 670-684.
- [3] C. Wang, Q. Wang, K. Ren, N. Cao, and W. Lou, "Toward secure and dependable storage services in cloud computing," *IEEE Trans. Service Comput.*, vol. 5, no. 2, pp. 220–232, Apr./Jun. 2012.
- [4] A. Juels and B. S. Kaliski, Jr., "PORs: Proofs of retrievability for large files," in *Proc. 14th ACM Conf. Comput. Commun. Secur.*, 2007, pp. 584–597.
- [5] J. Stanek, A. Sorniotti, E. Androulaki, and L. Kencl, "A secure data regeneration scheme for cloud storage," in *Technical Report*, Vol. 4, 2013.
- [6] J. R. Douceur, A. Adya, W. J. Bolosky, D. Simon, and M. Theimer, "Reclaiming space from regenerate files in a serverless distributed file system." in *ICDCS*, Vol. 5, Issue 2, 2002, pp. 617–624.
- [7] H. C. H. Chen and P. P. C. Lee, "Enabling data integrity protection in regenerating coding-based cloud storage: Theory and implementation," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 2, pp. 407–416, Feb. 2014.
- [8] Henry C.H. Chen, Yuchong Hu, Patrick P.C. Lee, and Yang Tang, "NCCloud: A Network-Coding-Based Storage System in a Cloud-of-Clouds", vol. 25, no. 2, pp. 407–416, Feb. 2014.
- [9] Jian Liu, Kun Huang, Hong Rong, Huimei Wang, and Ming Xian: Privacy-Preserving Public Auditing for Regenerating-Code-Based Cloud Storage, VOL. 10, NO. 7 JULY 2015.
- [10] K. Yang and X. Jia, "An efficient and secure dynamic auditing protocol for data storage in cloud computing," *IEEE Trans. Parallel Distrib. Syst.*, vol. 24, no. 9, pp. 1717–1726, Sep. 2013.
- [11] S. G. Worku, C. Xu, J. Zhao, and X. He, "Secure and efficient privacy-preserving public auditing scheme for cloud storage," *Comput. Elect. Eng.*, vol. 40, no. 5, pp. 1703–1713, 2013.
- [12] Y. Zhu, H. Hu, G.-J. Ahn, and M. Yu, "Cooperative provable data possession for integrity verification in multicloud storage," *IEEE Trans. Parallel Distrib. Syst.*, vol. 23, no. 12, pp. 2231–2244, Dec. 2012.
- [13] J. He, Y. Zhang, G. Huang, Y. Shi, and J. Cao, "Distributed data possession checking for securing multiple replicas in geographically dispersed clouds," *J. Comput. Syst. Sci.*, vol. 78, no. 5, pp. 1345–1358, 2012.
- [14] C. Wang, Q. Wang, K. Ren, N. Cao, and W. Lou, "Toward secure and dependable storage services in cloud computing," Apr./Jun. 2012.
- [15] C. Wang, S. S. M. Chow, Q. Wang, K. Ren, and W. Lou, "Privacy-preserving public auditing for secure cloud storage," *IEEE Trans. Comput.*, vol. 62, no. 2, pp. 362–375, Feb. 2013.
- [16] A. G. Dimakis, K. Ramchandran, Y. Wu, and C. Suh, "A survey on network codes for distributed storage," *Proc. IEEE*, vol. 99, no. 3, pp. 476–489, Mar. 2011.
- [17] K. D. Bowers, A. Juels, and A. Oprea, "Proofs of retrievability: Theory and implementation," in *Proc. ACM Workshop Cloud Comput. Secur.*, 2009, pp. 43–54.
- [18] Y. Dodis, S. Vadhan, and D. Wichs, "Proofs of retrievability via hardness amplification," in *Theory of Cryptography*. Berlin, Germany: Springer-Verlag, 2009, pp. 109–127.
- [19] V. Vankireddy, N. Sudheer, R. Lakshmi Tulasi, "Enhancing Security and Privacy in Multi Cloud Computing Environment", Vol. 6 (2) , 2015, 1267-1270
- [20] M. Ben Swarup, Chukkala Varaha Sampath Pothabathula Srikanth, "design and implementation of a secure multi-cloud data storage using encryption", *Volume 3 Issue 5, May 2014*.