Turkish Online Journal of Qualitative Inquiry (TOJQI)
Volume 12, Issue 7, July 2021: 8194 - 8201

Research Article

A Three Layer Privacy Preserving Cloud Storage Scheme with FileDeduplication

Asma Sousen[a], M. Swapna[b]

[a] PG- Scholar Department of Computer Science and Engineering. Stanley College of Engineering And Technology for Women., Abids, Hyderabad, India - 500001
[b] Assistant-Professor Department of Computer Science and Engineering., Stanley College of Engineering And Technology for Women., Abids, Hyderabad, India - 500001

***Corresponding author:** [a] asmasousen4@gmail.com, [b] mswapna@stanley.edu.in

Abstract

Cloud computing has always been an inquisitive topic for discussion in terms of the recent computer science field. As the number of people using computer is increasing exceedingly at an exponential rate there is also a need to manage and store the data that is generated.These data has to be processed at a quicker rate .Cloud and fog are interconnected fog is closer to the end devices therefore more faster in terms of the response .As the number of users are growing exponentially the security for cloud also is of great importance. Data integrity of users using the cloud is of utmost importance. This work enlightens the importance as well as the proposed method for a secure data storage to safe guard and protect the users who store their data on cloud from unauthorized exploitation.

Keywords: Cloud Computing,Fog Computing ,Hashing Alogorithms,SHA-512,Data Security ,Data Deduplication

1. Introduction

Our data is our property and we understand the need for securing it ,with increase in the number of people who are using a computer there is also a need to secure and preserve the data along with storing it .We are well acquainted with the term cloud storage with thousands of channels that are connected globally it's quite obvious to understand the vulnerability of the system to the cyber attacks and the data losses . There has been many events of cloud data leakage and also the loss that organizations and individuals faced due to it . Cloud is a wonderful avenue and platform for people who want to expand their enterprises and businesses but it also comes up with a lot of threats.This is because the owner does not have power over the data management. Though the user owns it but they do not have the control of the physical possession of the data that is stored on cloud.

Traditional methods of cloud security has been access restriction and encryption though they help preventing from the external attacks but does not work in the case where the CSP itself is not trustworthy . CSP can access the data and make illicit and felonious use of the user's data.

Instead of storing all the data in the cloud, we break the information into proportions based on user preferences. The other proportion stored in the other servers . This comprises of three tiers the cloud servers , the fog servers and the native computer and among various servers each consists some part of data .This division is based on the user's allocation strategy .The data is divided into k parts .Only if anybody has all parts only then one can reinstate the entire information.Since this scheme requires the next higher servers to hold no more than k-1 pieces and the remainder to be stored in the lower servers.By this, the unauthorised user would not be able to steal the data even though one of the server's data is compromised.

Firstly, user's data file is checked for deduplication at single user level before outsourcing it to the cloud.and then the file is further divided into 3 portions of 15%,30%,and 55% of data based on character based division. Then, for instance, allow 15 percent of the data to be processed on the local computer. Now transmit residual 55% information to cloud and 30% data to the fog server. While uploading it generates hashes of the 3 file proportions contents so that these hashes can be used as filenames to be stored in these 3 different servers .While retrieval user uses these hash value as file names to retrieve the files. As the user has the information of hashed filenames user owns the responsibility of its own data. Hence making the user data secure. The hashing algorithm that is being used is SHA512.

The aim is to give some power of management to the user . Avoiding reliance on the CSP for security of user data.Storing data in 3 different servers makes it difficult for the attacker to compromise all the 3 servers at once.

2. Literature survey

With the increase in the network bandwidth the number of people who have researched in the field of cloud security is increasing . There are various researchers who have worked towards the enhancement of cloud privacy and worked to make cloud not only a resourceful but a reliant platform. There has also been researchers who have worked in the field of deduplication .

Mr Dama Tirumala Babu1, Prof Yaddala Srinivasulu,"A Survey on Secure Authorized Deduplication Systems"Here the user devices a key that is convergent from each original copy of data this copy is encrypted using the convergent key ,So as to use the tag to detect the duplicates and if two copies of data are the same than their tags also are same.Inorder to understand whether there are duplicates the sender sends the tags to the server location to see if the same copy is already stored[15].

Bellare M, S Keelveedhi, and T Ristenpart "Message-locked encryption and secure deduplication" ,this uses message consistency and also the tag generation algorithm inorder to return a tag,but the the tag is as long as the cipher text for the contents[17] .

Wei et al found out that more of the old cloud security works focus on storage security than on computing security [12]Thereby proposing a discouragement for privacy cheating and auditing protocol of secure computation also named as SecCloud which tends to the the very first protocol for bringing the storage security and computation security audit in cloud thereby discouraging the cheating in privacy by verifier signature that is designated and batch verification[12] .

Kalpana Batra ,Ch. Sunitha , Sushil Kumar [5] proposed scheme in ensuring accuracy of owner's data in cloud data storage, They also proposed distributed scheme that is flexible with support of dynamic data , including block update, insert,delete, and append[5]. "Depending on erasure-correcting code in the distributing file preparation to address the redundancy data vectors and warranty the dependability of data"[5]. Using the token that are homomorphic with verification that is distributed of erasure coded data, the work achieves the integration of storage correctness insurance, i.e., whenever there is a detection of data corruption during the verification of storage correctness across the distributed servers, it can prove the identification of the faulty servers[5].

In paper [7] "Wang et al Describes out that users does not have the physical control over their storage as the data is outsourced and it makes the integrity of data a difficult task when it comes to storing it in cloud Therefore allowing ability of public audit for storing in cloud is of prime importance so that the owner of the data relies on the third party auditor that is TPA to conform the integrity of the data which is stored in cloud"[7].[7].

Swapnali More, Sangita Chaudhari, "Third Party Public Auditing scheme for Cloud Storage" Third Party Public Auditing scheme for Cloud Storage Prevents the integrity and tampering of data in cloud but it requires an extra module i.e, TPA that has to compute on the data blocks that has already been worked upon by the data owner[18].

Santosh P. Jadhav, R. Nandwalkar,"Privacy preserving and batch auditing in secure cloud data storage using AES",Data is stored in the cloud by using the most prominent algorithm AES but AES uses quite simple encrypt and key schedule operations therefore can be broken[19].

All these researches do protect external attacks but not internal one due in the work proposed in our paper we try to address this .

3. Methodology

It consists of 5 steps i.e,

1. User Uploads File
2. Deduplication Check
3. File Division and File Naming
4. Data uploaded in various servers
5. Download

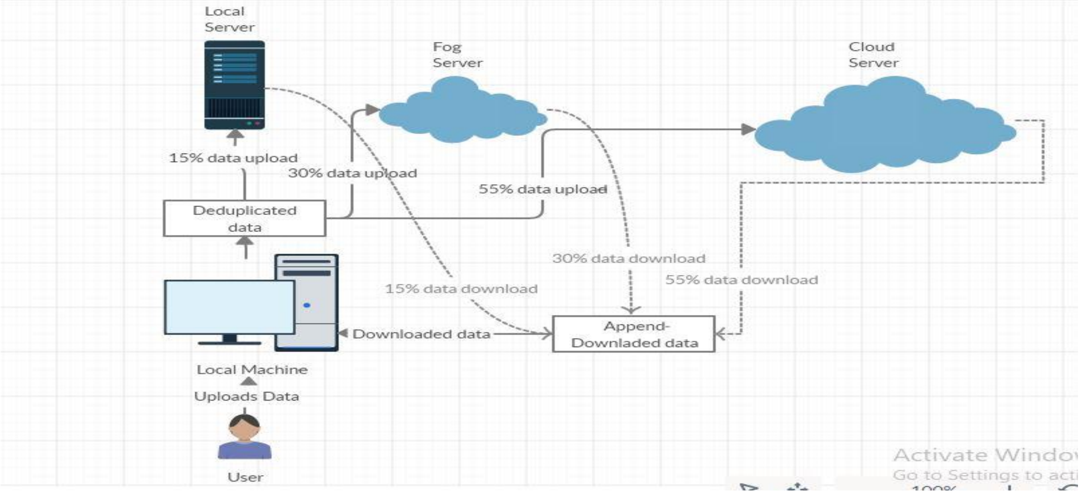


Fig 1. System Architecture

3.1 Data Upload

It all starts with user choosing to upload a file .Firstly,entire user’s file data are going to be checked for deduplication before outsourcing it to the servers ,the entire file data are going to be hashed on user’s local machine to get a message digest this message digest is sort of a tag for checking deduplication where the system checks whether there have been already existing tags of previously uploaded file that match the new one’s tag (message digest).If therefore the new file won't be uploaded and therefore the user will get a message of “file already exists” .The hash generated for entire file is for deduplication alone these tags or message digest don't hold any use for the division or storage steps. Then, the whole file data (plain text) is split into portions to be stored in several servers these individual portions are then hashed to generated message digests that are to be used as file names for the various file proportions that has got to be stored in several servers. Since the user maintains these hashed file names the safety is strong .And when needed the user can retrieve the files from different servers using these hashed file names .The reason of them being stored in 3 different servers is that it uses the cloud storage potential as well as makes it difficult for the attacker to compromise 3 systems i.e, cloud,fog and native machine.

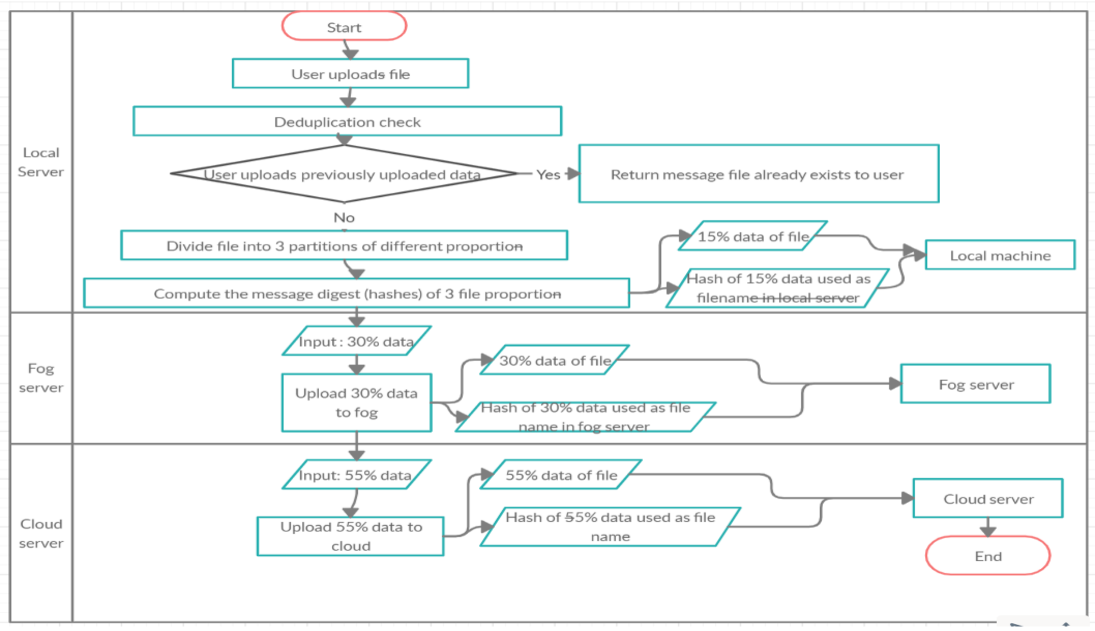

Local Server
Start
User uploads file
Deduplication check
User uploads previously uploaded data
Yes
Return message file already exists to user
No
Divide file into 3 partitions of different proportion
15% data of file
Local machine
Hash of 15% data used as filename in local server
Compute the message digest (hashes) of 3 file proportion
Fog server
Input : 30% data
30% data of file
Fog server
Upload 30% data to fog
Hash of 30% data used as file name in fog server
Cloud server
Input: 55% data
55% data of file
Cloud server
Upload 55% data to cloud
Hash of 55% data used as file name
End

Fig 2. File Upload Procedure

3.2 Deduplication Approaches :

One of the foremost important techniques for data compression is deduplication which helps eliminate the duplicate copies of an equivalent data . It's been used widely within the field of cloud computing to eliminate excessive use of bandwidth and to scale back the quantity of space used for storage.

Data Deduplication : often also know as intelligent compression/single-instance storage it's a way that permits or eliminates copies of information that are redundant and to scale back storage overhead. Data deduplication techniques make sure that only one single unique instance is stored on cloud storage.

Block-level deduplication: sees in file and stores unique chunks of every data block. All the blocks that are chunks of a specific length. Each data block is computed using a hash algorithm such as MD5 or SHA-1. In our case we are using SHA-512.

When a data block chunk receives a hash number, the number is then compared to the opposite of the actual hash numbers of existing hashes of other data blocks. If this hash value is already inside the scheme, the most recent piece of data to be submitted will be taken into account as a replica and will not be stored once again. Otherwise, the newest hash value will be applied to the system data and the most recent data will be stored. In some cases, the hash algorithm can produce identical hashes for two different blocks of information.It is referred to as a hash collision if this happens, and when this occurs, the machine will not store the new data block because it knows that the hash value already exists inside the index.This approach is referred to as a false positive, and data loss will result. SHA512 proves to be the simplest choice within the case of knowledge deduplication because it is collision resistant. Deduplication is done on the local server before outsourcing the information to cloud and fog servers. This deduplication is restrained to one user.

3.3 Where And How Are We Using Sha512

We are using this algorithm so as to compute the hash of the whole file and now if user uploads an equivalent content again the system searches if the hash generated for this new file is that the same as the hash generated by previous files that were uploaded. If the hash is that the same this new file with an equivalent content as the already existing one is prevented from being uploaded. Coming to the second way where SHA is employed it's when the file names of the 30% file F1 and 55% file data F2 is being stored within the system by generating a hash for them and using these hashes as file names in-order to access them from the local server .The algorithm takes a message with a maximum length of 2^{128} bits as input and generates a 512-bit message digest as output.

The input is processed in 1024-bit blocks. Thanks to avalanche effect, even little change within the message would end in a mostly distinct hash. SHA512 proves to be the simplest choice within the case of deduplication because it is collision resistant.

For Example:

[1]SHA512("Me and my husband will go to a movie today")No full stop MD: EB235A7EC36536EB78A5C51586A78FDE70B7423C9E7EFCD4D491E34B9FB4E6F2961815D2BBB5B02DBEFE58D0DD5C1B72178835C8F2D87713550F1F62F9549EE6

[2]SHA512("Me and my husband will go to a movie today.")Extra full stop wrt [1]. MD: 8B006E94829717EF448A3641D13CBD0C942B4AA277395644983BF4BEB71F7EBDE8FBBFFBD4257AF71DC31DD9D46E862F052CBBD4E1896D05B645D622A08629D5

3.4 File Processing

File Division

The division is completed so as to stop from giving the whole management of knowledge to the CSP, and this scheme prevents from retrieving the whole data from the partial data proportions . Moreover its difficult for an attacker to compromise the safety of three different servers directly . The division here is character based i.e, counting the amount of characters within the file and dividing them supported the user's allocation strategy ,in this aspect we have taken 15%,30% and 55% proportion from the whole file.

1. Count characters in file including whitespaces ,special symbols etc..

2. Take 15% ,30% and 55% of the entire count value

3. Allocate as many characters from the particular file to three file proportions basing on the count of characters

   their percentages contain.

4. Each proportions start character and end character are defined by the indexes assigned within the master file .

File Naming

Each of the file proportions' content are further hashed to get message digest each of their message digests are used as their filenames in 3 different servers i.e, local,cloud and fog servers. The three file data proportions are stored with their files named with the hashes of their contents.

3.5 Data Download

When the user requests for data download the local system prepares to send requests to the various servers where the information is stored .This request consists of the hashed filenames of the file proportions. On receive of request the servers cloud,fog and native server send back the data they held of the user i.e, 55% from cloud ,30% from fog and 15% from local server as per the filenames provided from the user's local machine. Appending all this data together the system returns back 100% data to the user . Hence the data is downloaded.

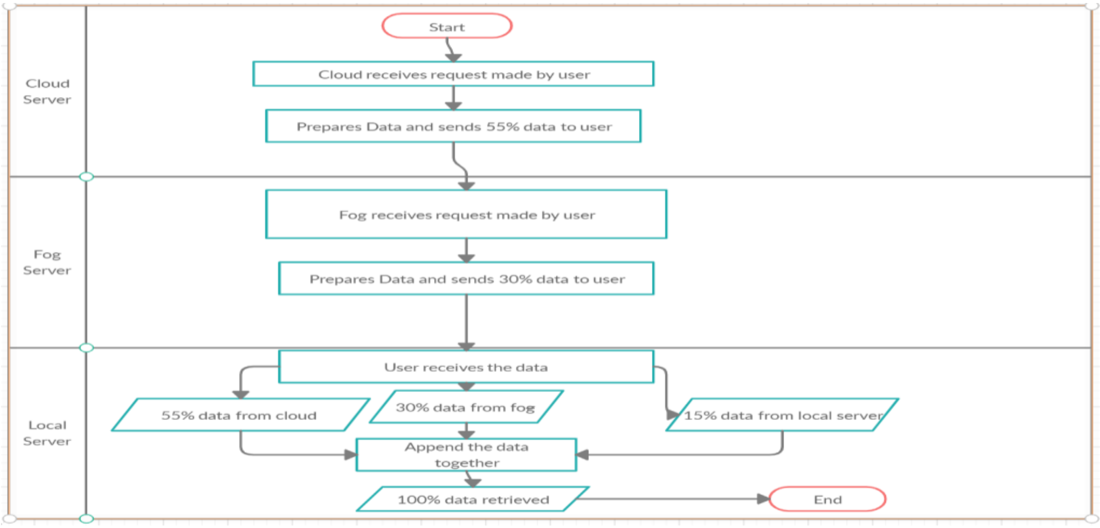

Cloud Server
Start
Cloud receives request made by user
Prepares Data and sends 55% data to user
Fog Server
Fog receives request made by user
Prepares Data and sends 30% data to user
Local Server
User receives the data
55% data from cloud
30% data from fog
15% data from local server
Append the data together
100% data retrieved
End

Fig 3. File Download Procedure

4. Result

Tested with files of various character lengths below is the result for the same and the splitting ratios of each of these files taken based on number of characters based split.

Table 1 : Data Division Experimental Results

FileName	**Size in characters**	**Local**		**Fog**		**Cloud**	
		15% of total characters	**Observed output**	**30% of total characters**	**Observed output**	**55% of total characters**	**Observed output**
Cisco.txt	600	90	90	180	180	330	330
Php.txt	800	120	120	240	240	440	440
india.txt	100	15	15	30	30	55	55
internet.txt	1800	270	270	540	540	990	990

Table 2 : Experimental Results and Validation of interface

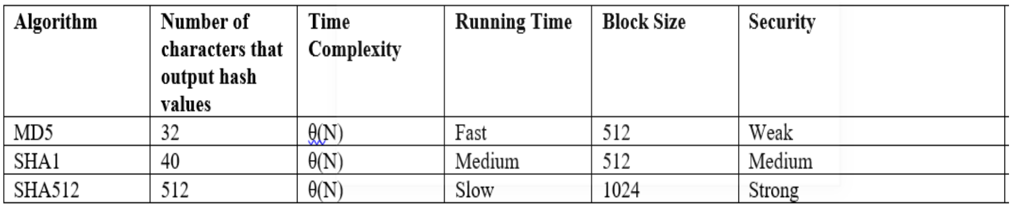
Name	**Description**	**Expected Output**	**Actual Output**	**Result**
File Upload	User uploads file into system	Option box to choose and upload file from user's end to servers	Option box to choose and upload file from user's end to severs	Success
File Deduplication	Prevent from uploading duplicate data at single user level	File already exists message	File Already exists message	Success
File division	Dividing the user's file into 3 parts	3 file proportions created	3 file proportions created	Success
File Naming	Naming of each file proportions	Naming of each file proportions with hashed values of their file contents	Naming of each file proportions with hashed values of their file contents	Success
File Download	Download of the user's file from different servers	Download link for the user to retrieve data	Download link for the user to retrieve data	Success
Same File Name different content	While user tries to upload and makes use of 2 same filename with different contents	File Uploaded message	File Uploaded message	Success

Different File Names same content	While user tries to upload and makes use of 2 different filename with same contents	File Already exists message	File Already Exists message	Success

5. Conclusion and Future Work

Why SHA512 is a better choice?

Table 3: Comparison in algorithms



Algorithm	Number of characters that output hash values	Time Complexity	Running Time	Block Size	Security
MD5	32	θ(N)	Fast	512	Weak
SHA1	40	θ(N)	Medium	512	Medium
SHA512	512	θ(N)	Slow	1024	Strong

Though the time complexity of all the three algorithms are the same SHA512's security has been considered more robust in comparison to the others. Here N is the block numbers necessary for input. Security in cloud isn't a goal nevertheless it's a process that can be never ending always having scope of development.Nonetheless, execution of cloud security is a shared liability of the data owner and resource provider. By assigning data block ratios into various servers rationally, ensuring privateness of information in individual server. This approach uses fog model and fog having limited storing and procession capabilities in contrast with cloud. The proposal and unification of data deduplication into this improves and upgrades the process of securing data meanwhile also providing reliant platform for users in storing their data onto cloud. Recent studies point to 17% up-flow of cloud market from 2019 to 2020.With this magnanimous usage of cloud there is a need of strong and robust mechanism for security. Security isn't end product but a process. Research certain to this avenue is ongoing activity . Thereby coming with better mechanism for cloud securities

References

[1] M.R.Tribhuwan,V.A.Bhuyar, Shabana Pirzade (2010) Ensuring Data Storage Security in Cloud Computing through Two way Handshake based on Token Management, International Conference on Advances in Recent Technologies in Communication and Computing edn.,

[2] A.Venkatesh,Marrynal S Eastaff (n.d.) A Study of Data Storage Security Issues in Cloud Computing, International Journal of Scientific Research in Computer Science, Engineering and Information Technology© 2018 IJSRCSEIT | Volume 3 edn., : .

[3] G.Ateniese,R.Burns,R.Curtmola, J. Herring, L. Kissner,Z. Peterson,and D. Song (n.d.) Provable Data Possession at Untrusted Stores, Proc. Of CCS '07, pp. 598–609, 2007 edn., :

[4] G.Ateniese, R. D. Pietro, L. V. Mancini, and G. Tsudik (n.d.) Scalable and Efficient Provable Data Possession, Proc. of SecureComm '08, pp. 1–10, 2008 edn., : .

[5] Kalpana Batra ,Ch.Sunitha , Sushil Kumar (n.d.) An Effective Data Storage Security Scheme for Cloud Computing, International Journal of Innovative Research in Computer and Communication Engineering edn., : .

[6] G.Kulkarni,R.Waghmare,R.Palwe,V.Waykule,H.Bankar,andK.Koli (n.d.) Cloud storage architecture, in Proc. 7th Int. Conf. Telecommun. Syst., Serv., Appl., 2012, pp. 76–81 edn., : .

[7] C.Wang,S.S.Chow,Q.Wang, K.Ren, and W.Lou (n.d.) Privacy-preserving public auditing for secure cloud storage, IEEE Trans. Comput., vol. 62, no. 2, pp. 362–375, Feb. 2013. edn., : .

[8] Po-Wen Chi,Chin-Laung Lie (n.d.) Audit-Free cloud storage via Deniable Attribute based encryption, Member IEEE,IEEE Transactions on Cloud Computing edn., : .

[9] Z.Fu, K.Ren, J.Shu, X.Sun, and F.Huang (n.d.) Enabling personalized search over encrypted outsourced data with efficiency improvement, IEEE Trans. Parallel Distrib. Syst., vol. 27, no. 9, pp. 2546–2559, Sep. 2016. edn., : .

[10] Z.Xia, X.Wang, X.Sun, and Q.Wang (n.d.) A secure and dynamic multikeyword ranked search scheme over encrypted cloud data, IEEE Trans. Parallel Distrib. Syst., vol. 27, no. 2, pp. 340–352, Feb. 2016. edn., : .

[11] Z.Fu, F.Huang, X.Sun, A.Vasilakos, and C.-N.Yang (n.d.) Enabling semantic search based on conceptual graphs over encrypted outsourced data, IEEE Trans. Serv. Comput.. [Online]. edn., http://doi.ieeecomputersociety.org/10.1109/TSC.2016.2622697: .

[12] L.Wei et al. (n.d.) Security and privacy for storage and computation in cloud computing, Inf. Sci., vol. 258, pp. 371–386, 2014. edn., : .

[13] R.Atan,A.M.Talib, and M.A.A.Murad (n.d.) Formulating a security layer of cloud data storage framework based on multi agent system architecture, GSTF J. Comput., vol. 1, no. 1, pp. 121–124, 2014 edn., : .

[14] J. Zeng,T.Wang,Y. Lai, J. Liang,and H. Chen (n.d.) Data delivery from WSNs to cloud based on a fog structure, in Proc. Int. Conf. Adv. Cloud Big Data, 2016, pp. 104–109 edn., : .

[15] Mr. Dama Tirumala Babu1, Prof.Yaddala Srinivasulu (n.d.) A Survey on Secure Authorized Deduplication Systems, : .

[16] PasqualoPuzio,RefikMolva,MelekOnen (n.d.) CloudDedup: SecureDeduplication with Encrypted Data for CloudStorage, SecludIT and EURECOM, France edn., : .

[17] M. Bellare,S. Keelveedhi, and T. Ristenpart (n.d.) Message-locked encryption and secure deduplication, : .

[18] Swapnali More,Sangita Chaudhari (n.d.) Third Party Public Auditing scheme for Cloud Storage, : .

[19] Santosh P.Jadhav,R.Nandwalkar (n.d.) Privacy preserving and batch auditing in secure cloud data storage using AES, : .