IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 5, Issue 5, May 2016

Secure Large File Deduplication Technique Over Distributed Cloud Environment To Store Anonymous User Data

Mr. Mahesh Bhaskar Gunjal[1], Prof. Rahul L. Paikrao [2]

M.E, Computer Department, AVCOE, Sangamner, Maharastra, India [1]

H.O.D, Computer Department, AVCOE, Sangamner, Maharastra , India [2]

Abstract: De-duplication is an abbreviated term to remove repeated copies of same data. It is necessary to keep most confidential data that is most susceptible while supporting de-duplication. We proposed data de-duplication system with an improved reliability as well as data confidentiality. We proposed data de-duplication process to reduced or replaced repeated data with same available data on cloud. This mainly results into saving of memory space. Data de-duplication is widely used to backup of data as well as it minimizes network overheads. It keeps only one single physical copy of data. To protect private data we used some secret sharing techniques such as Ramp secret sharing technique. Our model firstly, generates file block and then distribute them on different server. We used SHA-1 algorithm to generate block of data and hash values. Our proposed technique is totally different from existing de-duplication system as our system provides a smart solution for encrypted file duplication along with this; it must able to data secrecy as far as file is concern. To address challenges in unauthorized access, we described some privilege access. Therefore, privilege as well as non-privilege data can manage. Our system can work efficiently to reduced repetition in file level and block level data in distributed system. In this system, as a part of contribution our system will hide user's identity and trustee server is introduced to authorize the user. Also relative address concept for data blocks is used that keeps block data integrity and encrypted data more secure from cloud or server.

Keywords: De-duplication, distributed storage system, reliability, secret sharing, relative address.

I. INTRODUCTION

Cloud computing provides unlimited virtualized resources through internet. By providing this service it hides all the implementation details as well as the entire platform. Cloud services have continuously management of these services. It gives more attention towards the utilization of storage as well as to store space on cloud. In this paper proposed data de-duplication technique, with more reliability. In data de-duplication process are removing unnecessary copies of data and save memory space. Previously, many de-duplication systems are implemented based on the policies such as, file level, block level de-duplications and client-server side de-duplication technique[1][2]. In our proposed system we provide a smart solution to manage data redudancy on cloud that arises due to massive data transaction by user of cloud. Our technique helps to save the space of cloud as well as it save bandwidth and make it more responsive.

Many times there was same data stored by different users on the cloud and also this data have different encryption keys with respect to their owner hence it result into data redundancy[6][13]. Different level of data protection is provided[14]for cloud data,which required more bandwidth. It is a bottleneck problem in current situation. Our proposed Ramp secret sharing algorithm helps to solve this bottleneck problem. It preserve data secrecy during data encryption. Previously, PoW i.e. proofs of Ownerships are used to overcome the problem of hash signature of file [8]. Our system will neglects uploading of same files. In our system, metadata file is generated for each file to check whether user wants to upload the file is already present or not by comparing metadata file that is generated. Our system has two servers that help for check data sharing and privacy. These servers known as P –CSP and S-CSP, they allow user to deal with data by verifying from trustee.

We introduced P-CSP for locating relative file block adress and SCSP maintains logical mapping. Each time trustee will verify user's identity while uploading data as well as data de-duplication is also checked. While checking de-duplication our system avoids multiple transaction of file tags over network. Our system work better for hiding user's identity as well as neglecting data-duplication. We are implementing this system for file and block level de-duplication. Our system perfectly hides users identity and avoid data de-duplication. Our system, provides better data security as data is in encrypted format. Also data is stored on various servers hence,data retrival is impossible.

In this paper, further we representing, related work in section II, then architecture and its flow in section III. Moving towards algorithmic and mathematical representation of system is discussed in section IV,V&VI. Experimental results and used data set is given in section VII. Finally we are concluding our system in section VIII.

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 5, Issue 5, May 2016

II. RELATED WORK

Jin Li, Xiaofeng Chen, Xinyi Huang, Shaohua Tang and Yang Xiang [1], described distributed de-duplication system. This system aims to gain reliability & confidentiality of user's outsourced data. Authors were using ramp secret sharing for preserving data confidentiality as well as data reliability. Ramp secret sharing work better but it incurs some little bit overheads during data encryption and decryption.

J. Gantz and D. Reinsel[2] and J. R. Douceur, A. Adya, et al[3],gives analysis report of IDC researched in 2020 data volume. This data volume will be reaching 40trillion GB.

D M. Bellare, et al [4], represents Dup-Less works. It helps to obtain PRF protocol for encrypting client message that is based on message-based keys.In this system they want to show performance of encryption technique through de-duplicated storage.

G. R. Blakley and C. Meadows [5], A. D. Santis and B. Masucci[6] ,introducing multiple ramp schemes. Multiple ramp schemes are required for sharing secret among multiple participants. They were using entropy approach. Shamir represents management of key robustly in cryptographic system for secret sharing. They proved that their technique is efficient for secret sharing and managing as well as distributing encryption keys.

A. Shamir [7] , introducing multiple Sharing schemes. Multiple secret sharing schemes are required for sharing secret among multiple participants. They were using entropy approach. Shamir represents management of key robustly in cryptographic system for secret sharing. They proved that their technique is efficient for secret sharing and managing as well as distributing encryption keys.

S. Halevi, D. Harnik, et al[8],gives solution to overcome the problem of hash signature of files with the help of Proofs-of-ownership's i.e. PoW's.

J. S. Plank, S. Simmerman, and C. D. Schuman [9], describes code referred as a quasi-tutorial and a programmer's guide as an interface as well as techniques and algorithms for. Bit matrix. It is used for encoding and decoding.

M. Li, C. Qin, P. P. C. Lee [10], discussed about convergent dispersal. It is used to provide efficient security for cloud storage system. In this system they used original data for deriving deterministic cryptographic hash information. CRSSS and CAONT-RS are convergent dispersal algorithms proposed.

P. Anderson and L. Zhang [11], evaluates a local server to neglect the problem in sharing the data such as, time and cost of typical transfers and multi-user authentication.

A. Rahumed, H. C. H. Chen, Y. Tang, [12], avoids weaker areas of older FADE for protecting data stored in the cloud by extending previous FADE system (File Assured Detection). They also overcome the overhead in FADE.

M. W. Storer, K. Greenan, D. D. E. Long, [13], gives solution for management of space and data security in single-server system as well as distributed file system. Encryption keys are generated from chunk data in consistence manner and whole file utilizes the hash value as its identifier.

A.J. Stanek, A. Sorniotti, E. Androulaki, [14], determined the popularity of the data; in different level of protection that is provided for the data in cloud. This system provides the guarantee of semantic security for unpolar data as well as less security and better security with appropriate bandwidth benefits for popular data.

D. Harnik, B. Pinkas, and A. Shulman-Peleg [15], extracting de-duplication that is used as side channel. In this paper, authors studied about cross-user deduplication that provides guarantees of higher privacy with slightly reducing bandwidth savings in cloud storage.

J. Xu, E.-C. Chang, and J. Zhou [16], represents the deduplication for cross different users in which identical duplicated files from multiple users are detected and removed safely.

W. K. Ng, Y. Wen, and H. Zhu [17], described private data deduplication protocols & also formalized the context of two-party computations. Private data deduplication protocol is secure in simulation-based framework.

J. S. Plank and L. Xu[18] and C. Liu, Y. Gu, L. Sun, B. Yan, and D. Wang[19], were studied about Cauchy Reed-Solomon coding for the construction of distribution matrix. Distribution matrix is constructed for encryption and decryption of data. High reliability provision mechanism i.e. R-ADMAD is proposed by D. Wang. R-ADMAD is dynamic and distributed recovery process in the cloud storage.

From the above discussion of previous de-duplication we analyzed that they have certain overheads such as extra unneccessary key management is required which makes system more costly. There is need of such system which works on efficient management of encryption keys, efficient data sharing and data encryption over cloud. hence we proposed our system that provide all these beneficial features that will discussed in following sections.

III.PROPOSED SYSTEM

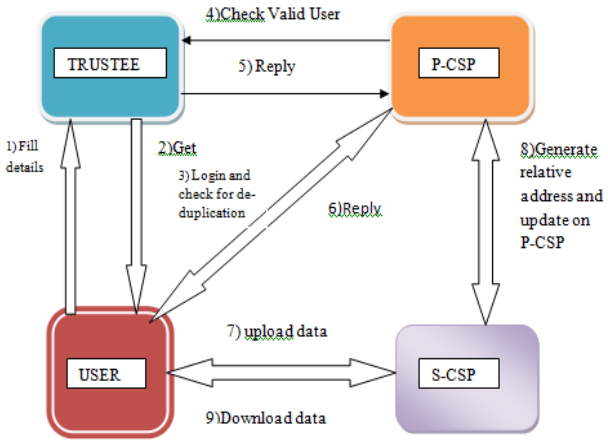

Fig1. Block diagram of secure deduplication system.

Fig.1. represents our proposed system. It contains user, trustee, P-CSP and multipal S-CSP. Systematic procedure for uploading and downloading file is described as follow:

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319 5940
International Journal of Advanced Research in Computer and Communication Engineering
Vol. 5, Issue 5, May 2016

A. Methodology for File Upload:

Step 1: Register user data on Trustee and get token.
Step 2: Using unique token User Login on P-CSP.
Step 3: P-CSP validates the token of user.
Step 4: User selects the file to upload and add the group members with whom the file will get shared.
Step 5: File tag generation at user end using SHA-1.
Step 6: send tags to P-CSP using HTTP connection.
Step 7: P-CSP checks privileges of user.
Step 8: If access Privilege Check passes then system will allow de-duplication check else gives error message.
Step 9: In de-duplication check, it matches the file tag with existing file tags.

Case 1: If tag matches run proof of ownership and share link with other users.
Case 2: If no file tag matches then it will check de-duplication at block level.
Case I: If Partial Duplication found:
It runs proof of ownership for partial no of blocks for new blocks generate convergent key and return token +key + block matching file information to the user. User encrypts unmatched block data using Sha-1. Upload token +encrypted data and file info to S-CSP. S-CSP save the file block and generate relative address mapping of file block of a file.
Returns the token and relative address tag information to P-CSP.

- P-CSP saves the token.
- Run proof of ownership.
- Share link with other users.

Case II: If no duplication found:
return token +no duplication response.

- Return token +no duplication response.
- User encrypt file block data using SHA-1and Ramp secret sahring.
- Upload token +encrypted data to S-CSP.
- S-CSP save the file block, generate relative address mapping of file block of a file.
- Returns the token and relative address tag information to P-CSP.
- P-CSP saves the token
- Run proof of ownership
- Share link with other users

B. Methodology for File Download:

Step 1: Register user data on Trustee and get token
Step 2: Using unique token User Login on P-CSP
Step 3: P-CSP authorize the user
Step 4: User Ask for file to download to P-CSP
Step 5: P-CSP checks the privileges of user.
Step 6: If user has privileges it returns file info to the user.
Step 7: User send file info and token to multipal S-CSP
Step8: S-CSP verifies the token and return file blocks to the user.
Step 9: User regenrate file block using Ramp technique and generate the original file.

IV. ALGORITHMS

A. SHA-1 Algorithm [7]

- Used for tag generation.
- A cryptographic hash function.
- SHA-1 produces a 160-bit (20-byte) hash value known as a message digest.
- A SHA-1 hash value is typically rendered as a hexadecimal number, 40 digits long.

B. Ramp secret sharing: [1]

- In cryptography, secret sharing refers to a process for distributing a secure / secret information amongst a group of users, each of which is allocated a share of the secret.
- There are two algorithm in secret sharing scheme,i)Share ii) Recover.
- Goal is to divide X file into m shares X1, ... , Xm in such a way that.

1) Share shared by using Share Algorithm.
2) And Recover By using Recover Algorithm.

C. Relative Addresses Generation Algorithm:
Input: blocks [], filename
Output: Block_relative_address [ ] BR, File_relative_address FR
Processing:
1. Define offset O1 and O2 as constant value
2. For each block i in blocks [ ]
3. Save file block and get physical address PAB
4. Calculate relative_address as
BR[i] = PAB + O1
5. END FOR
6. Save files information and BR and get Physical_address PAF of data
7. Calculate relative file address as:
FR = PAF + offset O2
8. Return

V. EXTENSION FOR PROOF OF STORAGE

In cloud storage environment, the server may be untrusted third-party in case of security and reliability. In cloud storage hide data loss/damage due to accidents or attacks can be happen.Therefore, it is very important for the users to checking their data availability . In order to allow the end user and the data storage server to perform secure and secure data storage checking, a new cryptographic primitive called proof of storage using tag generation algorithm has been proposed to achieve the goal.
In a proof of storage system, the data owner generate own tag id and tag id for each data block, and uploaded to the server. In our system file tag and block tags for a file generated by user. Therefore, user tags can be used as the authenticators for individual data blocks and files.

VI. MATHEMATICAL MODEL

S = {U, P-CSP, S-CSP}
U = {IU, OU, FU}

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319 5940
International Journal of Advanced Research in Computer and Communication Engineering
Vol. 5, Issue 5, May 2016

IU= {I1, I2, I3, I4}
I1 = user registration details. I2 = User login details.
I3 = File to upload. I4 = File name to download.
FU= {F1, F2, F3, F4, F5, F6, F7, F8}
F1 = User registration Request.
F2 = User Login Request.
F3 =File selection and block generation.
F4 = tag generation for file level and block level using sha-1 algorithm.
F5 = File encryption using AES.
F6 = File Decryption using AES.
F7 = generate tag for user access privileges using SHA-1.
F8 = Ramp Secrete sharing.
OU= {O1, O2, O3, O4}
O1 = File level Tag. O2= Bock level tag.
O3 = Access Privilege Tag.
O4 = Cipher text .
P-CSP= {IP, OP, FP}
IP= {I1, I2, I3, I4, I5}
I1 = User Register Data. I2 = User Login Data.
I3=File Tags for matching.
I4 = Access Privileges.
I5 = File location relative address detail array.
FP= {F1, F2, F3, F4, F5}
F1= User Registration.
F2 = User Identity Check.
F3 = De-duplication check using tag matching.
F4 = Proof Of ownership.
F5 = File information storage.
OP= {O1, O2, O3, O4}
O1 = User authentication response.
O2 = de-duplication response.
O3 = data sharing link. O4 = Access Privilege token.
S-CSP= {IS, OS, FS}
IS= {I1, I2, I3}
I1 = Encrypted file block.
I2 = File location relative address detail array.
I3 = File Access Privilege Token.
FS = {F1, F2, F3}
F1 = File block Storage.
F2 = Maintain file storage data structure.
F3 = File download.
OS = {O1, O2}
O2 = File location relative address detail array.
O1 = File to Download.

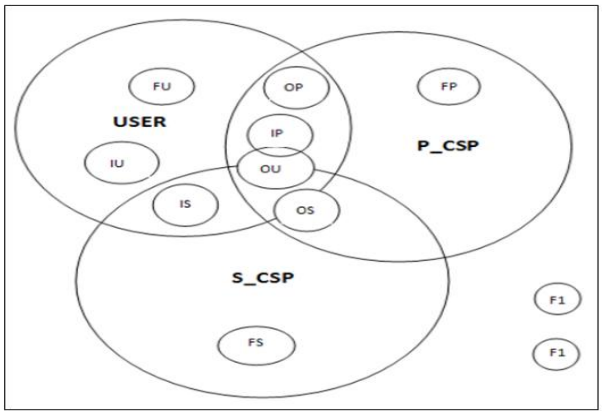

Fig. 2 Mathematical Model

VII. EXPERIMENTAL RESULTS

We have created system in java. Data is stored in mysql database. We have created a desktop application that communicates with P-CSP, S-CSP and Trustee Server using REST API. We have uploaded text document on cloud.
We have evaluated time required for tag generation and file deduplication checking for different file sizes. Following table I and Fig.3, and Fig.4, shows file level as well as block level deduplication check time and tag generation time.

TABLE I COMPUTATION TIME TO TAG GENERATION AND DEDUPLICATION CHECK

File size(in MB)	File Tag Generation Time	File Deduplication Check Time	Block Tag Generation Time	Block Deduplication Check Time
0.5	40	162	167	10729
1	43	179	250	20195
1.5	48	214	360	29930
2	56	273	510	45230

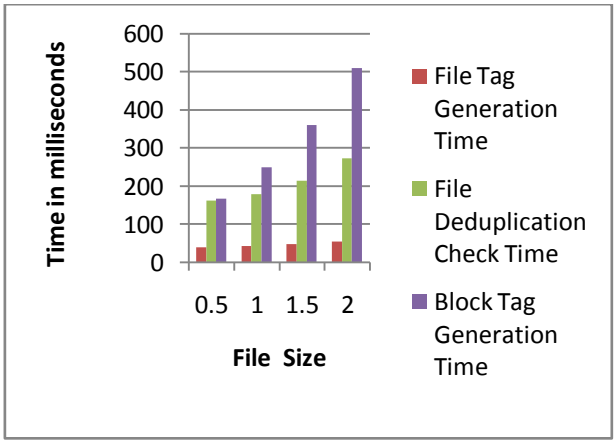

Fig. 3. Impact on file level deduplication check time and tag generation time.

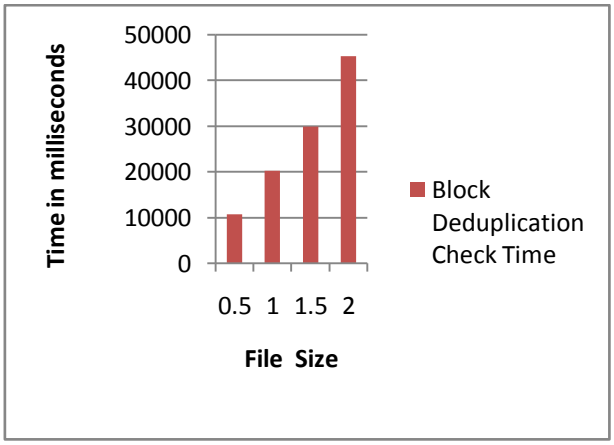

Fig. 4. Impact on block level deduplication check time.

Rather than uploading complete file on cloud we have uploaded secret k share on cloud. Following Table II and Fig.5, shows the share creation and block recreation time in milliseconds for different file sizes.

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 5, Issue 5, May 2016

TABLE II. SHARE CREATION AND BLOCK RECREATION TIME

Block size	share Creation	Block Re-creation
1kb	163	8
2kb	320	14
3kb	434	23
4kb	635	31

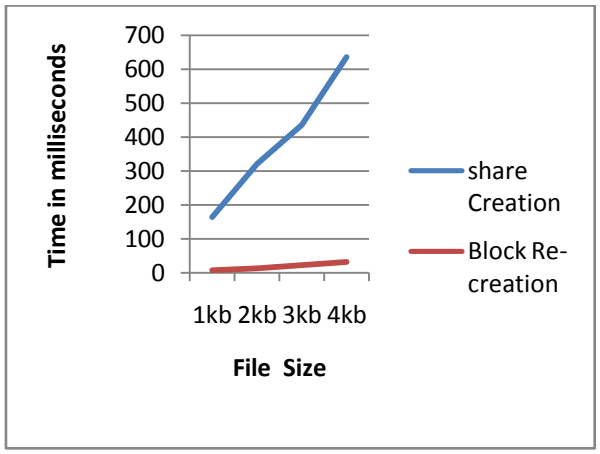

Fig. 5. Impact on share creation and block recreation time.

We have evaluated duplication ratio with upload time. As duplication ratio increases time required for uploading decreases. Table III and Fig. 6, shows duplication ratio increases time required for uploading decrease.

TABLE III. DUPLICATION RATIO INCREASES TIME REQUIRED FOR UPLOADING DECREASE

Duplication ratio	0.5MB	1MB	1.5MB	2MB
25%	8324	15128	22759	30649
50%	5638	10234	15134	21209
75%	2705	5032	7598	10469
100%	80	95	98	103

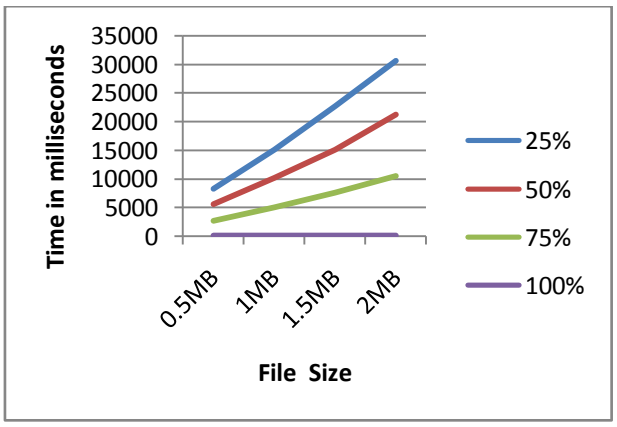

Fig. 6. Impact On ratio increases time required for uploading decrease

VIII. CONCLUSION

Our propose technique provides data security using data encryption in cloud environment. For effective usage of storage space we provide de-duplication check at file level as well as block level. We also provide new de-duplication constructions supporting authorized duplicate check in hybrid cloud architecture, in which the duplicate-check is done at private cloud server. This avoids multiple transaction of file tag over network while checking de-duplication. This technique support efficient usage of bandwidth. We introduce a relative addressing method in which P-CSP is having relative file block address and its proper mapping logic is maintained at S-CSP. This process properly blocks the hacking and data predictions. As a part of contribution our system hides end user identity.

REFERENCES

[1] Jin Li, Xiaofeng Chen, Xinyi Huang, Shaohua Tang and Yang Xiang Senior Member, "Secure Distributed Deduplication Systems with Improved Reliability"2015.

[2] J. Gantz and D. Reinsel, "The digital universe in 2020: Big data, bigger digi tal shadows, and biggest growth in the far east," http://www.emc.com/collateral/analyst-reports/idcthe- digital-universe-in-2020.pdf, Dec 2012.

[3] J. R. Douceur, A. Adya, W. J. Bolosky, D. Simon, and M. Theimer, "Reclaiming space from duplicate files in a serverless distributed file system." in ICDCS, 2002, pp. 617–624.

[4] M. Bellare, S. Keelveedhi, and T. Ristenpart, "Dupless: Serveraided encryption for deduplicated storage," in USENIX Security Symposium, 2013.

[5] G. R. Blakley and C. Meadows, "Security of ramp schemes," in Advances in Cryptology: Proceedings of CRYPTO '84, ser. Lecture Notes in Computer Science, G. R. Blakley and D. Chaum, Eds. Springer-Verlag Berlin/Heidelberg, 1985, vol. 196, pp. 242–268.

[6] A.D. Santis and B. Masucci, "Multiple ramp schemes," IEEE Transactions on Information Theory, vol. 45, no. 5, pp. 1720–1728, Jul. 1999.

[7] Shamir, "How to share a secret," Commun. ACM, vol. 22, no. 11, pp. 612–613, 1979.S. Halevi, D. Harnik, B. Pinkas, and A. Shulman-Peleg, "Proofs of ownership in remote storage systems." in ACM Conference on Computer and Communications Security, Y. Chen, G. Danezis, and V. Shmatikov, Eds. ACM, 2011, pp. 491–500.

[8] S. Halevi, D. Harnik, B. Pinkas, and A. Shulman-Peleg, "Proofs of ownership in remote storage systems." in ACM Conference on Computer and Communications Security, Y. Chen, G. Danezis, and V. Shmatikov, Eds. ACM, 2011, pp. 491–500.

[9] J. S. Plank, S. Simmerman, and C. D. Schuman, "Jerasure: A library in C/C++ facilitating erasure coding for storage applications - Version 1.2," University of Tennessee, Tech. Rep. CS-08-627, August 2008.

[10] M. Li, C. Qin, P. P. C. Lee, and J. Li, "Convergent dispersal: Toward storage-efficient security in a cloud-of-clouds," in The 6th USENIX Workshop on Hot Topics in Storage and File Systems, 2014.

[11] P. Anderson and L. Zhang, "Fast and secure laptop backups with encrypted de-duplication," in Proc. of USENIX LISA, 2010.

[12] A.Rahumed, H. C. H. Chen, Y. Tang, P. P. C. Lee, and J. C. S. Lui, "A secure cloud backup system with assured deletion and version control," in 3rd International Workshop on Security in Cloud Computing, 2011.

[13] M. W. Storer, K. Greenan, D. D. E. Long, and E. L. Miller, "Secure data deduplication," in Proc. of StorageSS, 2008.

[14] J. Stanek, A. Sorniotti, E. Androulaki, and L. Kencl, "A secure data deduplication scheme for cloud storage," in Technical Report, 2013.

[15] D. Harnik, B. Pinkas, and A. Shulman-Peleg, "Side channels in cloud services: Deduplication in cloud storage." IEEE Security & Privacy, vol. 8, no. 6, pp. 40–47, 2010.

[16] J. Xu, E.-C. Chang, and J. Zhou, "Weak leakage-resilient client-side deduplication of encrypted data in cloud storage," in ASIACCS, 2013, pp. 195–206.
[17] W. K. Ng, Y. Wen, and H. Zhu, "Private data deduplication protocols in cloud storage." in Proceedings of the 27th Annual ACM Symposium on Applied Computing, S. Ossowski and P. Lecca, Eds. ACM, 2012, pp. 441–446.
[18] J. S. Plank and L. Xu, "Optimizing Cauchy Reed-solomon Codes for fault-tolerant network storage applications,"
[19] Liu, Y. Gu, L. Sun, B. Yan, and D. Wang, "R-admad: High reliability provision for large-scale de-duplication archival storage systems".

BIOGRAPHY

Mr. Mahesh Bhaskar Gunjal Perusing Masters in Computer Engineering at the collage of Amrutvahini College of engineering. He has received his Bachelors in Computer Engineering in the collage of Amrutvahini College of engineering, Savitribai Phule Pune University, Sangamner, Maharashtra, India He is a member of Association of Computer Machinery (ACM).

Prof. R.L Paikrao is an Assistant Professor of Computer Engineering Department at the collage of Amrutvahini Collage of engineering (AVCOE) Savitribai Phule Pune University, Sangamner, Maharashtra, India. He is a member of Association of Computer Machinery (ACM).

DOI 10.17148/IJARCCE.2016.55154