TPADB: is the TPA database that consist of hash information as well as user who has generated the file partitions.
C : is the set of challenges send by the TPA to the server.
R : is the result generated by the cloud storage server to generate the proof of the integrity.
L : is the log record maintained at the TPA.

Functionalities:
The funcionalities of the existing system working as follows.
1.U=Register(uid,Password,Uname,Address, ContactNo,EmailID)
U=UserLogin(UID,Password)
2. F= Selectfile (FileName)
3. P= Partitionfile (F, Psize)
4. Ep= EncryptPartitions (P, Key)
5. H = HashCalculation(Ep)
6. TPADB= StoreHashInfo (H)
7. SDB =StorePartitions (Ep ,uid)
8. C=RandomChallenges(TPADB)
9. R=ComputeResult(SDB, Ep)
10. L=AddLogEntry(Pname, PDate, State)

5.3Algorithms of proposed system

Algorithm 1: **Key Generation(KeyGen)**
Key Generation is a key generation algorithm running at the data owner side, which generates the secret key PKi upon getting input of some security parameter using AES algorithm.
Input: File (F).
Output: Private Key (PKi) [key is used for encryption and decryption mechanism..
1. Result = ValidateFormat(F);
validateFormat = { png ,.pdf .jpg , ...}

2. If the format is valid then client want to secure the data using AES Algorithm.
CipherData(Fi) = AES(Fi);

Algorithm 2: Sign Generation(SigGen)
Sign Generation algorithm used to generate digital signature of the encrypted file partitions.
Input: File partitions(P1,P2,P3...Pn)
Output:Hash values of encrypted partitions(H1,H2...Hn)

1. MessageDigest md = MessageDigest.getInstance("SHA");
MessageDigest is generic class used to provide the default functionality of the SHA-1 algorithm for the application.The getInstance() method generates the MessageDigest Object.

2. md.update(datArr, 0, datArr.length);
Updates method takes the input _le by appending a byte array at the end.

3. byte[] sha1hash = md.digest();
This method applied SHA-1 algorithm on current input message and returns the array of bytes.

4.Separate the first 4 bits of the particular byte i.e MSB(Most Significant Bits)and convert then to character.

5. Separate the last 4 bits of the particular byte i.e LSB(Least Significant Bits)and convert then to character.

Algorithm 3: VerifyProof
Verify proof algorithm run by the TPA for performing the auditing task.
Input: Random File blocks
Output:Verification Status
1. File Indices = getRandom();
This method returns the random file indices and store in the vector array.

2. Result=updateLog()
The updateLog method update the result of the verified partitions maintain the file status like file is safe or file is unsafe.

Algorithm 4:Generation proof(GenProof)
This algorithm run by the cloud storage server for the possession of the data files.
Input:File
Output:Hash of the files

1. Result=GenProof(File, Chal)

This method accept the challenge from TPA and send the proof of possession of the data files to the TPA.

6. Result

Verified Blocks (bytes)	TPA Individual Audit time(ms)	TPA Batch Audit Time(ms)
3072	1014	921
3891	1154	1014
4096	1155	999
4505	1532	1045
5734	1632	1357

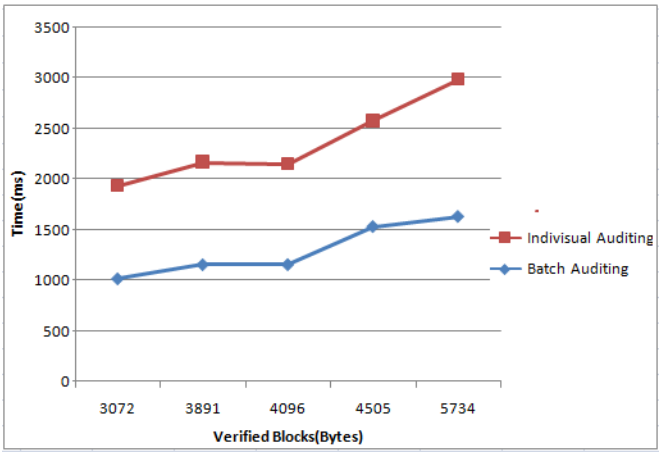

Figure 3: Comparison on auditing time between batch and individual auditing.

The Fig.3 shows the results for the auditing task. The number of auditing task & time required for auditing shown in the graph. The performance of the corresponding individual auditing is provided as a baseline for the measurement. Following the same settings for the data blocks of the file. The average per task auditing time, which is computed by dividing total auditing time by the number of tasks, is given

for both batch and individual auditing. It can be shown that compared to individual auditing, batch auditing indeed helps reducing the TPA's computation cost, as more than 15 percent of per task auditing time is saved.

7. Conclusion

Use of third-party auditing service provides a cost-effective method for users to gain trust in cloud. We assume the TPA, who is in the business of auditing, is reliable and independent entity. In this paper, public auditing system is proposed for data storage security in cloud computing along with preserving privacy mechanism. It can utilizes the homomorphism linear authenticator and random masking to guarantee that the TPA would not learn any knowledge about the data content stored on the cloud server during the efficient auditing process, which not only eliminates the burden of cloud user from the tedious and possibly expensive auditing task, but also alleviates the users fear of their outsourced data leakage. Public auditing is able to allow clients for delegating the tasks of integrity verification to TPA while they are independently not reliable or cannot commit required resources of computation performing verifications in a continuous manner. One more important concern is the procedure for construction of verification protocols which can be able to accommodate data files that are dynamic.

For achieving data dynamics that are effective, the existing proof of storage models is enhanced through manipulation of the construction of classic SHA-I for authentication of block tag. For supporting good handling of multiple numbers of auditing tasks, the method of bilinear aggregate signature is further explored for extending the main result into a multiuser setting, where TPA is able to perform multiple auditing tasks in a simultaneous manner. Huge security as well as performance analysis proves that the proposed scheme is efficient and secure to a greater extent. As the future work, efficiently audit the integrity of shared data with dynamic groups while still preserving the identity of the signer on each block from the third party auditor and also include the features to enable dynamic operations (e.g. inserting/deleting data block) in this system.

8. Acknowledgement

I sincerely thanks to prof. S. B. Sonkamble for her continuous and constructive support for the work. I would also like to thanks to my friends for their valuable comments.

References

[1] P. Mell and T. Grance, "The NIST Definition of Cloud Comput- ing," technical report, Nat'l Inst. of Standards and Technology, 2009. J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68-73.
[2] M. Armbrust, A. Fox, R. Griffith, A.D. Joseph, R.H. Katz, A. Konwinski, G. Lee, D.A. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "A View of Cloud Computing," Comm. ACM, vol. 53, no. 4, pp. 50-58, 2010K.
[3] Cloud Security Alliance, "Top Threats to Cloud Computing,"http://www.cloudsecurityalliance.org, 2010.
[4] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, "Provable Data Possession at Untrusted Stores," Proc. 14th ACM Conf. Computer and Comm. Security (CCS '07), pp. 598-609, 2007.
[5] A. Juels and J. Burton, S. Kaliski, "PORs: Proofs of Retrievability for Large Files," Proc. ACM Conf. Computer and Comm. Security (CCS '07), pp. 584-597, Oct. 2007.
[6] G. Ateniese, R.D. Pietro, L.V. Mancini, and G. Tsudik, "Scalable and Efficient Provable Data Possession," Proc. Int'l Conf. Security and Privacy in Comm. Networks (SecureComm '08), pp. 1-10, 2008.
[7] M.A. Shah, R. Swaminathan, and M. Baker, Privacy-Preserving Audit and Extraction of Digital Contents, Cryptology ePrint Archive, Report 2008/186, 2008.
[8] M.A. Shah, M. Baker, J.C. Mogul, and R. Swaminathan, Auditing to Keep Online Storage Services Honest, Proc. 11th USENIX Workshop Hot Topics in Operating Systems (HotOS 07), pp. 1-6, 2007
[9] H. Shacham and B. Waters, "Compact Proofs of Retrievability," Proc. Int'l Conf. Theory and Application of Cryptology and Information Security: Advances in Cryptology (Asiacrypt), vol. 5350, pp. 90-107, Dec. 2008.
[10] Q. Wang, C. Wang, K. Ren, W. Lou, and J. Li, Enabling Public Auditability and Data Dynamics for Storage Security in Cloud Computing, IEEE Trans. Parallel and Distributed Systems, vol. 22, no. 5, pp. 847859, May 2011.
[11] D. Boneh, B. Lynn, and H. Shacham, "Short Signatures from the Weil Pairing," Proc. Seventh Int'l Conf. Theory and Application of Cryptology and Information Security: Advances in Cryptology (ASIACRYPT'01), pp. 514-532, 2001
[12] C. Wang, Q. Wang, K. Ren, and W. Lou, "Privacy-Preserving Public Auditing for Storage Security in Cloud Computing," IEEE Transactions On Computers, vol. 62, no. 2, FEB 2013.
[13] Wang C, Wang Q et al. Privacy-Preserving Public Auditing for Storage Security in Cloud Computing, Proceedings IEEE INFOCOM'10.2010
[14] Wang B, Li B et al. Oruta: Privacy-Preserving Public Auditing for Shared Data in the Cloud, IEEE Fifth International Conference on Cloud Computing, 295–302.2012
[15] C. Wang, Q. Wang, K. Ren, and W. Lou, "Towards Secure and Dependable Storage Services in Cloud Computing," IEEE Trans. Service Computing, vol. 5, no. 2, 220-232, Apr.-June 2012.
[16] C. Erway, A. Kupcu, C. Papamanthou, and R. Tamassia, "Dynamic Provable Data Possession," Proc. ACM Conf. Computer and Comm. Security (CCS '09), pp. 213-222, 2009.

Author Profile

Madhumati Shinde received the B.E in computer Engineering from ADCET,Ashta, Maharashtra . She is now persuing M.E.in computer Engineering at RSSOER, Pune, Maharashtra. Her area of interest is Cloud Computing.

Mrs. Sulochana Sonkamble is HOD of Computer Science Department at Rajarshi Shahu College of Engineering and Research, JSPM NTC Pune-India. She obtained Bachelor of Engineering, Masters of Engg. And PhD in Computer Science Shri Guru Gobind Singhji Institute of Engineering and Technology, Swami Ramand Tirth Marathwada University, Vishnupuri, Nanded-India.