SECURE AND DEPENDABLE DATA CORRECTNESS APPROACH OVER CLOUD ARCHITECTURE

[1]BODDU NANDA KISHORE, [2]Vegi Srinivas

[1] **M.tech II Year(Pursuing) , Dadi Institute of Engineering and Technology**, Anakapalli, Visakhapatnam

[2] **Assoc.Prof, Dadi Institute of Engineering and Technology**, Anakapalli, Visakhapatnam

Abstract - Cloud computing has been envisioned as the next generation architecture of IT enterprise. It moves the application software and databases to the centralized large data centers where management of data and services may not be fully trustworthy. This unique paradigm brings out many new security challenges like, maintaining correctness and integrity of data in cloud. Integrity of cloud data may be lost due to unauthorized access, modification or deletion of data. Lacking of availability of data may be due to the cloud service providers (CSP), in order to increase their margin of profit by reducing the cost, CSP may discard rarely accessed data without detecting in timely fashion. To overcome above issues, flexible distributed storage, token utilizing, signature creations used to ensure integrity of data, auditing mechanism used assists in maintaining the correctness of data and also locating, identifying of server where exactly the data has been corrupted and also dependability and availability of data achieved through distributed storage of data in cloud. Further in order to ensure authorized access to cloud data a admin module has been proposed in our previous conference paper, which prevents unauthorized users from accessing data and also selective storage scheme based on different parameters of cloud servers proposed in previous paper, in order to provide efficient storage of data in the cloud. In order to provide more efficiency in this paper dynamic data operations are supported such as updating, deletion and addition of data.

Keywords*-Cloud computing, data integrity, distributed storage, error localization, authorized access, selective storage, dynamic operations.*

I. INTRODUCTION

Cloud computing is a term used to describe a new class of network based computing that takes place over internet a collection or group of integrated and networked hardware, software and internet infrastructure. In addition cloud computing provides on demand services that are always on, anywhere, anytime and anyplace. A number of characteristics define cloud data application services and infrastructures are remotely hosted, ubiquitous and commodified.

Moving of data into clouds by users is more advantageous because there is no overhead of maintaining hardware. Data stored on clouds are maintained by CSP with various incentives for different level of services. In order to ensure integrity and availability of cloud data efficient methods have to be designed for providing data dependability and correctness verification of cloud data on behalf of users. Local copy of data in not maintained at the local machine of users.

Hence it's of critical importance to ensure users that their data are being correctly stored and maintained. Integrity and availability threats cab be reduced by storing individual users data redundantly across multiple physical servers.

Distributed storage and verification scheme are used to ensure the correctness and availability of user data. File distribution which is dividing of file into smaller blocks and storing them across multiple cloud servers. Hence correctness and integrity check for each file block is separately carried out, and also maintaining redundant copy of each file blocks across cloud servers guarantees the data dependability against byzantine failures. Schemes like token computation, generation of digital signature for cloud data to be stored, token signature ensures integrity and security of cloud data. These schemes restricts against malicious data modification attack and server colluding attacks.

Third party auditor (TPA), which is trusted for accessing of the cloud data and finding out any corruption of cloud data that occurs at cloud servers on behalf of users upon request. Users interact with the cloud servers via CSP to store and retrieve data. The users can delegate data auditing tasks which includes verification of correctness of data stored at the cloud servers to a trusted TPA. Also TPA performs data error localization that is identification of misbehaving servers and which particular block of file has been corrupted at which server. Focus is more in support of file oriented cloud applications.

Number of related works [1] [2] [3] was proposed for ensuring the remote data integrity under different systems and security modes. But these techniques useful in ensuring storage correctness. Since users do not posses local copy of data, related works focuses only on single server scenario. But does not guarantee data availability in case of server failures.

In [16] paper the proposed scheme achieves authorized access of data stored at the cloud servers through admin module. And proposed scheme also includes storage of data across the clouds not randomly but according to the cost and quality of cloud servers there by achieving efficient storage of cloud data. For more efficiency in this paper schemes has been proposed to support dynamic operations at block level.

II. RELATED WORKS:

In this section some of the related works are discussed along with their schemes and disadvantages also solution for those disadvantages in the proposed scheme. Some of the related schemes proposed are [1] [2] [3]. Here TPA concept has been proposed. But redundant copy maintenance of file is not included hence does not assure availability of data in case of server failure or corruption of stored cloud data. Since distributed file storage not included, focuses only on single server scenario.

Related works [4] [5] proposed scheme which includes distributed storage that is dividing of files into multiple blocks and storing them randomly across multiple cloud servers, maintaining redundant copy of data to ensure cloud data availability, TPA which checks for cloud data integrity and also error localization performed by TPA to localize at which server the file block has been corrupted. But there is a overhead for users to generate keys. Also does not provide, efficient storage of cloud data, scheme for access of cloud data only for authorized users.

Hence to overcome above specified disadvantages proposed schemes in [16] paper includes efficient storage of cloud data which achieved by storing of data across cloud servers based on their cost and quality and also admin module proposed in [16] paper to achieve authorized access of data stored across cloud servers and reduces the overhead of users of key generation.

But in previous [16] paper in order to perform dynamic operations on data stored across servers entire file has to downloaded and token and signatures has to be recomputed for entire file, even though the dynamic operations is to be performed at block level, only for particular block which is inefficient one, in order to overcome this schemes proposed in this paper, so that only a particular block can be downloaded on which operations is to be performed. Token and signatures computed only for that particular block not for entire file as in previous scheme there by providing more efficiency.

In the next section proposed scheme and architecture explained in detail.

III. PROPOSED SCHEME

A. Proposed system architecture:

Fig 1 is architecture representing proposed system which provides secure dependable and selective storage services in cloud computing. Admin module proposed in this architecture reduces overhead of users of generating keys.

In the proposed system architecture, data which is uploaded by users divided into multiple blocks and stored across selected cloud servers but not randomly across servers as specified in other related schemes, in this paper as per proposed scheme cloud servers are selected based on constraints like cost and quality. There by ensuring efficient cloud data storage.

Admin module is responsible for ensuring authorized access of data stored across cloud servers thereby restricting unauthorized access of cloud data, also responsible for generation of master key used by users for creation of digital signature and also generates public key used by auditors during their auditing scheme and thereby reduces overhead of user module of generating keys.

In Fig 1 three different network entities can be identified as follows:

User: an entity, who has data to be stored in the cloud and relies on the cloud for data storage and computation, can be either enterprise or individual customers.

Cloud Server(CS): an entity, which is managed by CSP to provide data storage service and has significant storage space and computation resources.

Third-Party Auditor(TPA): who has expertise and capabilities that users may not have, is trusted to assess and expose risk of cloud storage services on behalf of the users upon request.

Admin: responsible for ensuring authorized access of data stored across cloud servers and also responsible for generation of keys.

B. Construction of proposed system architecture

The main methods of the proposed scheme include selecting of cloud servers based on constraints like cost and quality, generation of master key and public key, maintaining list of authorized users who can

access data stored across cloud servers, creation of digital and token signature. The modules considered in the architecture to perform above specified operations are TPA, users, admin.

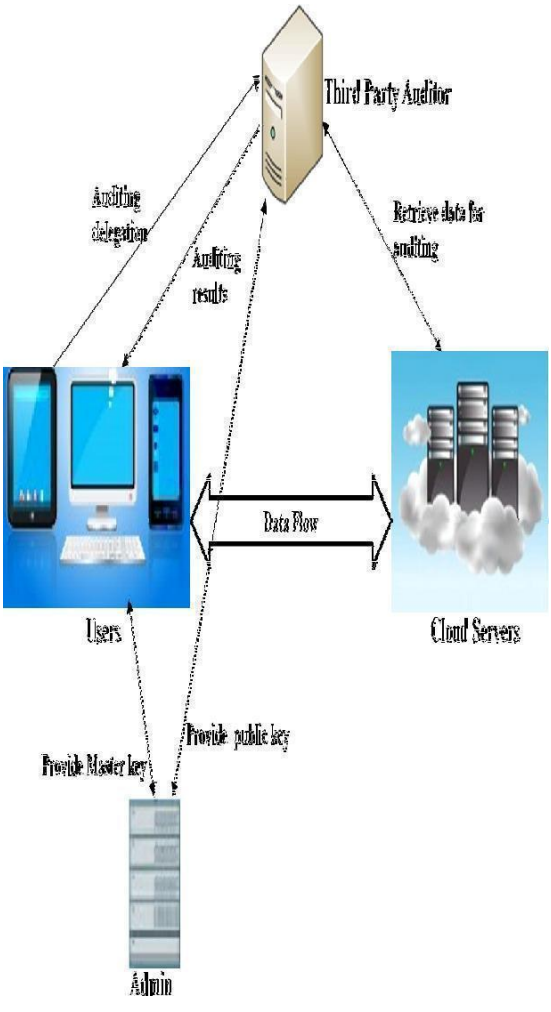


Fig 1: System Architecture

Notations:

F- File to be uploaded
U- User id
T(i)- ith Token
F(i)- ith file block
MD(i)- ith Message Digest
DS(i)- ith Digital Signature
TS(i)- ith Token Signature
DB- Database
P(K)- Public Key

a) Users

File Distribution Preparation

It is well known that erasure-correcting code may be used to tolerate multiple failures in distributed storage systems. In cloud data storage, this technique is used in this paper to disperse the data file F redundantly across a set of n distributed servers, where n = m + k. k redundancy parity vectors created from m data vectors in such a way that the original m data vectors can be reconstructed from any m out of the m + k data and parity vectors. And each of the m + k vectors are stored on a different server, such that the original data file can survive the failure of any k of the m + k servers without any data loss, with a space overhead of k/m. The unmodified m data file vectors together with k parity vectors are distributed across m + k different servers.

Users are one who does file uploading and file downloading operations. Whenever users uploads file the users are provided with options to specify in how many blocks file has to be divided, once user selects number of blocks then corresponding number of tokens are generated, where number of tokens generated is directly proportional to number of blocks specified by users.

Tokens are generated as follows,

Filename and length of the file and randomly generated secret matrix are the parameters used in generation of tokens. Secret matrix is different for different file blocks.

Choose parameters and then follow below steps:

1. Choose number of blocks to be taken

2. Token = Filename + length of the file + randomly generated secret matrix

3. If there are i number of file blocks, then token i number of tokens has to be generated.

4. Step 2 has to be repeated for i times in order to generate i number of tokens, with varying secret matrix which is different for each ith token.

Using token message digest is generated for each of file blocks separately and using master key provided by the admin message digest is encrypted to produce digital signature which is later during auditing scheme used by auditor for integrity checking of data stored at cloud servers. Then using same master key token is encrypted to produce token signature which is also later used by auditor during integrity checking of cloud data. These digital and token signatures are stored in the database along with their file id for later retrieval from auditors. Others are restricted from accessing that database.

Then store all the file blocks into respective clouds while uploading each file block redundant copy of file blocks are also stored across servers in order to ensure availability and dependability of data, as per proposed scheme file blocks are not randomly stored instead first cloud servers are selected based on their cost and quality, if users prefer cloud server with

lowest cost rather than quality then data stored across servers selected with lowest cost, if quality is preferred not the cost then data stored across selected cloud servers with high quality. Hence efficient storage of data across clod servers is achieved.

When user request for file download, only authorized users are allowed to download the file, authorized users detail are maintained by admin module thereby ensuring authorized access of cloud data. The TPA performs auditing scheme while downloading file to check for integrity of data thereby ensuring security of data. Users also maintain the list of transactions such as file uploaded and downloaded by them along with file name, date, day and time. Fig 2(a), (b) illustrates flow chart for sequence of operations during file upload.

Different parameters considered for cloud server selection:

A good service provider is the key to good service So, it is important to select the right service provider. Payment flexibility, API, Scalability, and Reliability (may also include support).

Cost benefits - the cloud promises to deliver computing power and services at a lower cost. Anywhere/anytime access - It promises universal access to high-powered computing and storage resources for anyone with a network access device. Quality- which includes cloud status active or inactive also cloud quality which includes speed provided by servers with which file can be uploaded and downloaded.

Capacity- amount of data that can be stored across cloud servers.

Load- amount of data already stored across cloud servers and its effect on the speed and quality of severs.

Encryption of data, to be stored at cloud servers for more security of data:

Before uploading of file across cloud servers by users, file is to be encrypted and then stored in order to ensure more security of data. In this paper since file is divided into smaller file blocks, each file block is separately encrypted using users private key and then stored, while storing redundant copy of file block first it is encrypted, compressed and then stored, to reduce storage overhead file blocks are compressed.

While downloading of file blocks from cloud servers, TPA performs auditing scheme to check for integrity of data stored, in order to check for correctness of data, first the data has to decrypted using master key and then as usual operations of message digest comparisons has to be performed. To check integrity of redundant copy of data, first file block has to be decompressed and decrypted and correctness checking is performed by TPA.

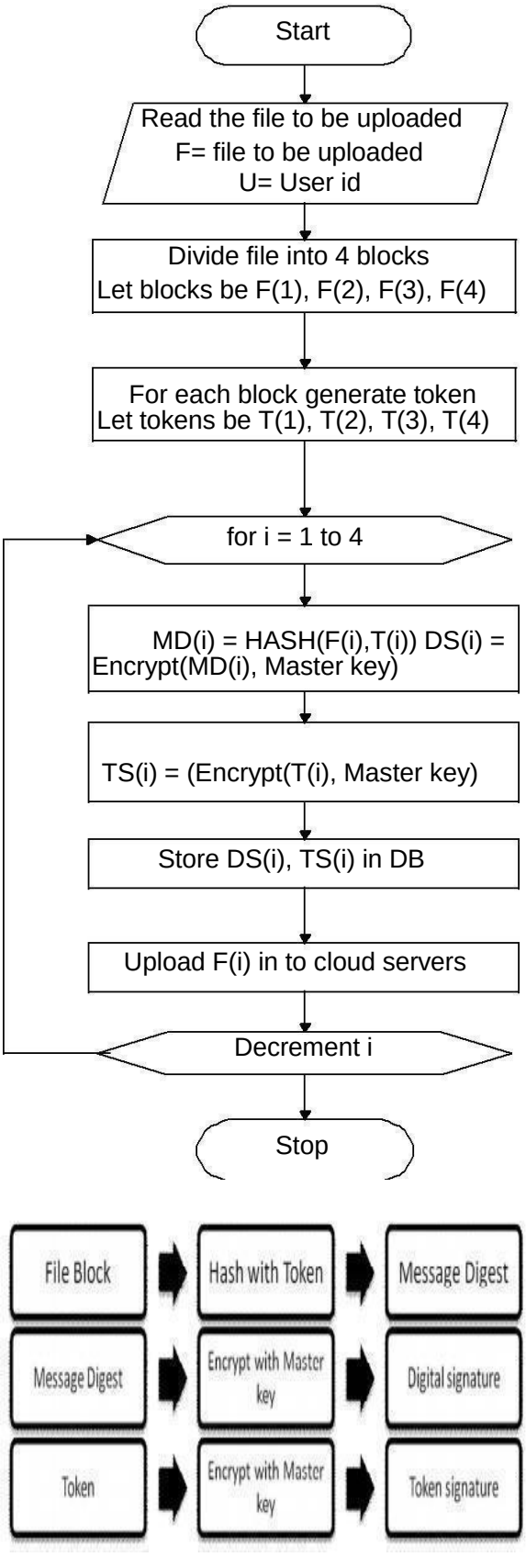


Fig 2(a) and 2(b): Operations during File upload process

b)TPA

TPA is one which performs auditing scheme upon request from users in order to check and ensure integrity of cloud data as shown in Fig 3, which represents flow chart of sequence of operation performed during auditing scheme. Auditor has to specify file id of corresponding file for which integrity has to be checked as per request from users. Based on file id respective set of file blocks of a particular file along with their redundant copies has to be retrieved from cloud servers. Based on user id corresponding public key is retrieved. Based on file id, for each of file blocks corresponding digital signature and token signature is retrieved.

Using public key digital signature is decrypted and message digest of each file block is obtained, which is first message digest of each block. Using same public key token signature is decrypted and token is obtained for each of file blocks. Using respective block token second message digest is generated for each file block. That is first message digest obtained from digital signature and second message digest generated from stored file block at servers.

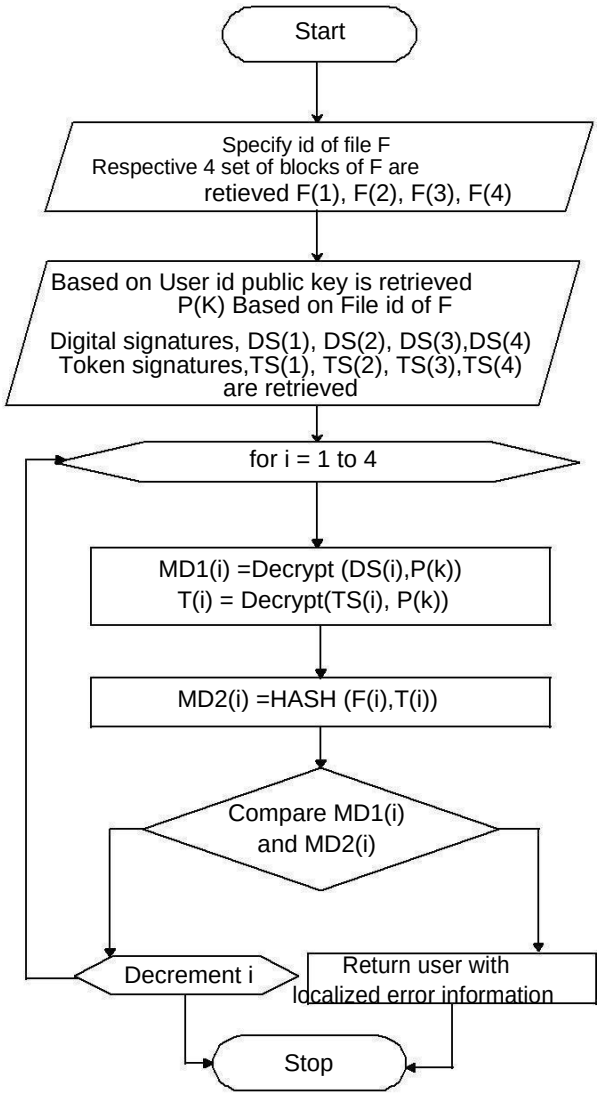
Compare first and second message digest for each block, if they differ for any one block then that block is said to be damaged and replaced with the redundant copy. TPA identifies which block is corrupted and localizes at which server the file block has been corrupted and returns to user with identity of server where block is corrupted.

The TPA maintains list of cloud servers along with their name and IP address which helps in locating of cloud server where exactly data corruption has occurred. TPA also maintains the list users along with their corresponding filenames, number of blocks according which file was divided.

c)Admin

Admin module responsible for generation of master key for users used by them during digital signature and token signature creation also generates public key for auditors used by auditor during auditing scheme for integrity check. There by reducing the overhead of key generation of user module. This admin module also ensures authorized access, maintains list of users who have access to cloud data, and thereby preventing unauthorized users from accessing data stored at cloud servers.

Admin module maintains the list of users along with their secret key and list cloud servers along with their name and IP address, list of cloud servers along with their corresponding size, cost quality and status. Also maintains list of user transactions along with their id, file name and which type of transaction has been carried out like uploading or downloading of file along with corresponding date, day and time of particular transaction.



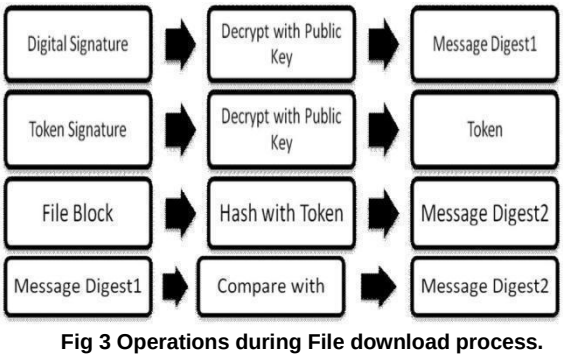


Fig 3 Operations during File download process.

IV. SUPPORT FOR DYNAMIC DATA OPEARATION

If we consider F as a file to be stored across cloud servers, some scenarios may arise where users may wish to perform various operations such as updating, deletion and addition at block level. In existing schemes user has to download entire file and then it

has perform operations and again it has to recompute token and digital signatures for entire file which results in overhead for users and is enfficient.

But scheme used in this paper divides file F is into blocks, hence user can dowload only blocks on which it wishes to perfoem dynamis opeartion and recompute token and signatures only for those particular blocks not entire file there by providing efficient method for dynamic operations and reducing overhead for users.

Updating:

If user wish to modify any particular block then , it has to specify index of that particular block, then only that particular block is downloaded rather than downloading entire file, only that block is modified and token and signature is recomputed only for that particular modified block and then again uploaded.

Deletion:

Whenever user wish to delete a particular block then it has to specify the index of that block, download that block and replaces that block with zero or any apecial character and then recomputes signature for that and uploads back to cloud server.

Addition:

Some scenarios may arise where user wish to increase size of data stored across servers by increasing the number of data blocks. This addition of block will be at the end of the file also known as appending of file blocks, token and signature has to be computed for the newly added block and then uploaded.

V. CONCLUSION

Scheme used in this paper includes, storing of file blocks across cloud servers which are selected according to their cost and quality and other parameters, thereby providing the efficient storage. Admin module used ensures authorized access of data stored across cloud servers and also reduces overhead of users of generating keys. Scheme proposed in this also supports dynamic data operations such as updating, deleting and appending at block level.

REFERENCES

[1]. C. Wang, S.S.M. Chow, Q. Wang, K. Ren, and W. Lou, "Privacy- Preserving Public Auditing for Secure Cloud Storage," IEEE Trans. Computers, preprint, 2012, doi:10.1109/TC.2011.245.

[2]. C. Wang, Q. Wang, K. Ren, and W. Lou, "Privacy-Preserving Public Auditing for Storage Security in Cloud Computing," Proc. IEEE INFOCOM, Mar. 2010.

[3] C. Wang, K. Ren, W. Lou, and J. Li, "Towards Publicly Auditable Secure Cloud Data Storage Services," IEEE Network Magazine, vol. 24, no. 4, pp. 19-24, July/Aug. 2010.

[4] C. Wang, Q. Wang, K. Ren, and W. Lou, "Ensuring Data Storage Security in Cloud Computing," Proc. 17th Int'l Workshop Quality of Service (IWQoS '09), pp. 1-9, July 2009.

[5] Toward Secure and Dependable Storage Services in Cloud Computing Cong Wang, , Qian Wang, Kui Ren, Ning Cao, and Wenjing Lou, IEEE TRANSACTIONS ON SERVICES COMPUTING, VOL. 5, NO. 2, APRIL-JUNE 2012.

[6] Secure Data transfer in Cloud Storage Systems using Dynamic Tokens. P.Srinivas * ,K. Rajesh Kumar # M.Tech Student (CSE), Assoc. Professor *Department of Computer Science (CSE) , Swarnandhra College of Engineering and Technology(SCET), International Journal of Research in Computer and Communication technology, IJRCCT, ISSN 2278-5841, Vol 2, Issue 1, January ,2013.

[7] 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.

[8] M.A. Shah, R. Swaminathan, and M. Baker, "Privacy-Preserving Audit and Extraction of Digital Contents," Cryptology ePrint Archive, Report 2008/186, http://eprint.iacr.org, 2008.

[9] G. Ateniese, R.D. Pietro, L.V. Mancini, and G. Tsudik, "Scalable and Efficient Provable Data Possession," Proc. Fourth Int'l Conf. Security and Privacy in Comm. Netowrks (SecureComm '08), pp. 1-10, 2008.

[10] C. Erway, A. Kupcu, C. Papamanthou, and R. Tamassia, "Dynamic Provable Data Possession," Proc. 16th ACM Conf. Computer and Comm. Security (CCS '09), pp. 213-222, 2009.

[11]H. Shacham and B. Waters, "Compact Proofs of Retrievability," Proc. 14th Int'l Conf. Theory and Application of Cryptology and Information Security: Advances in Cryptology (Asiacrypt '08), pp. 90- 107, 2008.

[12]K.D. Bowers, A. Juels, and A. Oprea, "Proofs of Retrievability: Theory and Implementation," Proc. ACM Workshop Cloud Computing Security (CCSW '09), pp. 43-54, 2009.

[13]R. Curtmola, O. Khan, R. Burns, and G. Ateniese, "MR-PDP: Multiple-Replica Provable Data Possession," Proc. IEEE 28th Int'l Conf. Distributed Computing Systems (ICDCS '08), pp. 411-420, 2008.

[14]Y. Dodis, S. Vadhan, and D. Wichs, "Proofs of Retrievability via Hardness Amplification," Proc. Si xth Theory of Cryptography Conf. (TCC '09), Mar. 2009.

[15]M. Castro and B. Liskov, "Practical Byzantine Fault Tolerance and Proactive Recovery," ACM Trans. Computer Systems, vol. 20, no. 4, pp. 398-461, 2002.

[16]VINITHA S P & GURUPRASAD E, "SECURE, DEPENDABLE AND SELECTIVE STORAGE SERVICES IN CLOUD COMPUTING", International Conference on Computer Science and Information Technology, 10th, March 2013, Hyderabad, ISBN: 978-93-82208-70-9.

Authors:

BODDU NANDA KISHORE pursuing M. Tech from Dadi Institute of Engineering and Technology, Anakapalli, Visakhapatnam, India. He obtained his MCA degree from Andhra University.

Vegi Srinivas received his M.Sc. and M.Tech.in Computer Science and Technologyfrom the Dept. of Computer Science and Systems Engineering, College of Engineering, Andhra University. He is currently doctoral candidate in the Department ofComputer Science & Engineering, JNTU, Kakinada, India. He is working as Associate Professor in the Dept. of Computer Science and Engineering, Dadi Institute of Engineering and Technology, Anakapalli, Visakhapatnam, India. His main areas of interests are Security, Privacy, Trust and Cloud Computing.

