IJARCCE

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

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 5, Issue 7, July 2016

Efficient Public Auditing Scheme for Shared Cloud Data Storage Using Multi Replica Merkel Hash Tree

Kaveri Sonawane[1], Prof. Ranjana Dahake [2]

Student, Computer Engineering, MET's IOE BKC, Nashik, University of Pune, City, India [1]

Asst. Professor, Computer Engineering, MET's IOE BKC, Nashik, University of Pune, City, India [2]

Abstract: Cloud computing is in demand in recent portion of decades as data transaction and storage it is exponentially increases as far as web based systems are concern. Also Cloud provides dynamic storage space and can handle big data dealing complexities. Though they can manage cloud big data complexities, it is also semi trusted entity. Cloud shows secure but snooping behaviour hence integrity of data and security is major part of concern. Also replication of data is required to improve data availability on cloud. Hence data replication process is provided by the cloud. Also there are cloud data auditing processes that keep the trace of data integrity. To do auditing process, complete dataset retrieval is not required. There are existing data auditing schemes that has flaws when concern to big data. These flaws can be like time consuming process of multiple replica update when verification is done, problem is simultaneous operations of authentication and auditing etc. Hence to rectify these problems proper solution can be developed for auditing process. To address these problems with big data in current public auditing, we proposed a multi-replica dynamic public auditing approach for shared user. For time problem we introduced SHA-1 algorithm instead of previous algorithm that gives significance time improvement that can bridge the gaps mentioned in above discussion.

Keywords: Cloud Computing, Public Data auditing, Big Data, Merkel Hash Tree.

I. INTRODUCTION

New distributed computing approach is adapted by cloud computing, which is helpful for big data processing and handling its complexity. There are many definitions are available for “Big Data”. It can be defined as the data 5-V attributes that is having volume, velocity and variety value, veracity these all are properties of big data. Data contain in a Big data can be structured, semi-structured or it can be relational data. Whereas, multi-structured data is referred as the data set this involves mixture of all these data sets. For solving big data problems cloud provides technological backbone [2] in IaaS (Infrastructure-as-a-Service), PaaS and SaaS. Cloud is great for big data application as it saves lot of time required for purchasing hardware its maintenance. Cloud computing are able to handles the complexities and big data streams of big data applications. For hosting the application security is major phase in the cloud [5], [8]. The dataset contain by the big data application are always dynamic in nature i.e. internet data. In many applications like social networks and business transactions, data updates are very frequent. Therefore cloud security mechanism, such as a public auditing scheme is very important to support dynamic data. Mainly three dimensions are concern in security aspect i.e. confidentiality, integrity and availability. Previous, public auditing schemes supports verification over data for updates. Such updates are managed by authenticated data structures (ADS) such as Merkle hash trees [4]. But in previous approach number of research gap is mentioned which is addressed over here. Existing research maintain several replicas in analysis of the big datasets. So there is problem in updating the datasets as it leads to update every replica. Therefore, public auditing scheme requires O(logn) communication complexity. To tackle these problem with big data in current public auditing, proposed scheme multi-replica dynamic public auditing (MuR-DPA) can bridge the gaps mentioned in above discussion. It is authenticated data structure used to store a data.

II. RELATED WORK

M. Armbrust et al. [2] have provided the simple form, they did the comparison between cloud computing and conventional computing. Authors also mark functional and non-functional opportunities of cloud storage. They have it in mind to minimize confusion by, providing simple figures to rectify the comparisons between of cloud and conventional computing. In their proposed system definition to identify certain installations very clearly as applications and non-examples of cloud Computing. R. Buyya, et al. [3]. They worked on the system architecture for market oriented allocation of resources within Clouds. They also provided the image for the creation of global Cloud exchange for trading services. For the blooming adoption of Cloud computing, like meta-negotiation infrastructure for global Cloud exchanges services like

DOI 10.17148/IJARCCE.2016.5774

third-party are enabled. Authors also specified the comparisons between High Performance Computing (HPC) workload and Internet based services workload. Furthermore, they were detecting the harnessing 'Storage Clouds' for high performance content delivery. C. Liu, J. et al. [4], they have performed the formal analysis of all types of fine-grained data updates. The scheme on they have worked on supports authorized auditing and fine-grained update requests. It also tries to reduce communication overheads for verifying small updates. Based on this scheme, they have also proposed an improvement for dramatically minimizing the communication difficulties for verifying small updates analysis on top of experimental results of this system explores that proposed system may not offer only enhanced security and flexibility, but it will also provides lower overhead for a large number of frequent small updates, such as applications in social media and business transactions. R. C. Merkle et al's[5] did the digital signature system based on conventional encryption. They proposed algorithm to sign and check the sign are rapid and require the small amount of memory. They have determined the exchange between size and memory required for signature. The advantage of this proposed system is to minimize the computational cost compared to arithmetic modular systems. A digital signature system is based on DES as DES runs faster than exponentiation modulo.

H. Shacham and B. Waters [6], have studied about the proof of irretrievability which having full proofs of security against arbitrary adversaries in the strongest model. In this first system was built from BLS signatures and second system builds PRF in standard model. In the proposed method, user breaks an erasure encoded file into n number of blocks. This method is called public verifiability. Authors have introduced new parameter to find the exchange between storage overhead and response length. Framework allows unforgeability and retrievability of the system.

E. Shi and C. Papamanthou[7], have point out blackbox application of ORAM (Oblivious RAM). They have proposed system with dynamic PoR scheme with constant client storage. This system requires cryptographic information. PoR (Proofs of Retriev- ability) ensures that the server maintains knowledge of all outsourced data blocks, PoR is used to proof authenticity and retrievability of the system. To rep- resent the portion of bandwidth cost Hierarchical FFT encoding is plotted in this methodology. This identifies the Computational overhead. S. Subashini and V. Kavitha [8] put a detail concept regarding different risk in cloud system. Internet services but also for the IT sector as a whole. Many issues exist, particularly related to service level agreement, they have described security, privacy and power efficiency in cloud based storage. This primarily focused on service delivery models issues of cloud computing. Later they discussed about common security issues, the security threats posed SaaS (Software as Service)delivery model, the security threats posed by the PaaS (Platform as Service) delivery model, the security threats posed by the PaaS (Platform as Service)de- livery model, describes the security threats posed by the IaaS (Infrastructure as Service) delivery model. To solve the problem related to theses many models Cloud Security Alliance (CSA) is used to group the solutions.

B. Wang et al's [9] have described Forward Secure ID-based Ring Signature. It is an essential tool for building cost effective genuine and anonymous data sharing system. It also provides unconditional anonymity. This proposed scheme involve the distinguishing features as well as data privacy, which is the SEM (security-mediator) does not learn anything about the data to be uploaded to the cloud. To evade the potential single point of failure multi-SEM model is used to extend the result of the system. B. Wang et al's [10], have proposed a new preserving public auditing mechanism is called as "Oruta"(One Ring to Rule them all). It helps to share data in untrusted cloud. To construct homomorphic authenticators it utilizes ring signature. Security properties of Oruta are implemented; these properties are based on bilinear maps and the right hand side (RHS). They also prove the security issues in untrusted cloud.

B. Wang, B. Li [11], have put a light on the method of public auditing to share the data with effective revocation of user. For that they utilize proxy re-signatures. This method, utilizes the idea of proxy re-signatures. Therefore it improve efficiency of user revocation as well computational resources are saved. In public auditing, public verifier can audit the integrity of shared data. In improves the scalability cloud data can be efficiently shared on large system. C. Wang et al's [12] have worked on privacy preserving public auditing system. To provide security of data on cloud this system handles multiple audit session for different users for their outsourced data files. Further, TPA is extended to perform audits for multiple users at a time and efficiently. In this system, auditing protocol is appropriately designed during the auditing process to prevent data from "flowing away" towards external parties. They also utilize the public key based homomorphic linear authenticator. P. Williams et al's[14], have introduced a new mechanisms to increase throughput, a generalization of ORAM(Oblivious RAM) democratization, and implementation of an efficient ORAM for secure parallel querying of existing ORAMs. It performs a transaction per second on a terabyte database in an average latency network (a first). They also give integrity assurance of the cloud data and discuss their demerits. PKC based homomorphic authenticator. This system accomplish efficient data dynamics, hence it improve the existing proof of storage models. It also manipulate, classic Merkle Hash Tree (MHT).

New type of data structure is proposed in system called as balanced update tree By Y. Zhang and M. Blanton [15], they have also provided a new solution for data possession which supports dynamic functionality, also to share data amongst multiple users. They select, Bloom-filter based ORAM to introduce PD-ORAM ("Parallel De-amortized ORAM"). It is applicable to the classic ORAM. It improves the performance of system with the elimination

DOI 10.17148/IJARCCE.2016.5774

of critical bottlenecks and drawback. Y. Zhu, H. Hu, et al.[16], authors have introduced construction of an efficient PDP (Provable Data Possession) scheme for distributed cloud storage. This scheme is based on homomorphic verifiable response and hash index hierarchy. They have proposed a scheme to support dynamic scalability on multiple storage servers. It requires zero knowledge of interactive proofs.

Q. Wang et al's [13], have considered verification schemes with public auditability in which any TPA can perform as verifier to evaluate quality from an objective and independent perspective. In this process proof of storage models manipulates the Merkle Hash Tree construction which is required for block tag authentication. Third Party Auditor (TPA) construct the system. It aims to develop Dynamic data operation support, Public auditability and Blockless verification of storage correctness assurance. B. Wang et al's[17] have worked on Efficient user revocation of user in shared group To ensure integrity of shared data, users in the group have to compute signatures on the blocks in shared data. Various data blocks in shared data are generally signed by different users due to data modifications performed by different users. If one of user is revoked from the group, the blocks which were previously signed by this revoked user have to re-signed by an existing user in the efficiently.

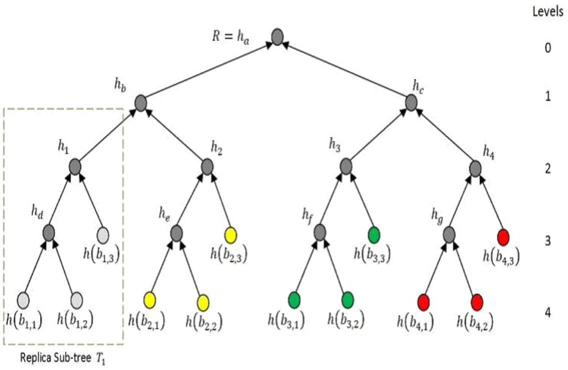
Chang Liu et al's [1] have worked for public auditing scheme which uses Top-down Levelled Multi-replica Merkle Hash Tree to store data which is stored on cloud. It provides the level values of nodes in MR-MHT.

III.PROCESSING STRTEGY

Public auditing scheme named MuR-DPA i.e. Multi-replica Dynamic Public Auditing which uses authenticated data structure based on the Merkle hash tree. To support data updates and authentication of block indices. MHT has a rank and level values in for each MHT nodes. All replicas of a same blocks are arranged into a same replica sub-tree. This arrangement allows efficiently verification of updates for multiple copies of replicas. Each MR-MHT is constructed based on blocks of a file, and its replicas, as well as a pre-defined cryptographic one way hash function H. Shared user strategy is adopted, in which user can share his own data on cloud amongst the group of user.

A. MHT Data Storage Strategy

In the figure. 1 example of MR-MHT is shown. It is constructed based on a file, that file is divided in to 4 blocks and for each block 3 replicas are maintained. Levels of nodes in tree are defined in a top-down order, i.e. the level of root node is defined as 0 and gradually increase as per height of tree is increases. Values stored in the leaf nodes are hash values of stored replica blocks. Value stored in a none-leaf node is combination of hash values of its child nodes and two more parameter one is the level of that node in a tree and another is maximum number of leaf nodes i.e. nodes in bottom level that can be reached from that particular node.



. Figure 1: Example of Merkel Hash Tree

B. Public Auditing Scheme

As compared to existing integrity verification and public auditing schemes, theoretical analysis predict that the proposed scheme reduces communication overhead for both update verification and integrity verification of cloud datasets with multiple replicas, but also provide improved security against dishonest CSS.

The Figure.2 three main actors in the system are shown client, third party auditor and cloud server. These three actors in system they do not fully trust each other. Authenticated data structures (ADS) such as MHT can enable other parties to verify the content and updates of data blocks. In this system multiple users are connected to the cloud.

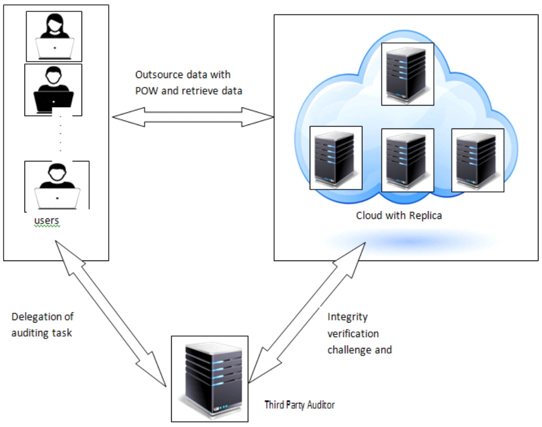


Figure.2. Relations between the participating parties cloud, TPA and Client.

Users are grouped together and can share data among group or can share their data individually. User having revocation rights can revoke user in the group. Creator of the group is having by default revocation rights and it has the rights that he can give revocation right to users in the group. User having access privileges can download/modify file. When user upload file on cloud original file with its backup copies are maintained on cloud servers. Here we can consider 3 replicas are created

DOI 10.17148/IJARCCE.2016.5774

for single file. File can be saved in terms of blocks. Cloud preserve block with its replica details. When data is uploaded on server its metadata is calculated using Merkle Hash Tree technique. Meta-data contains data about original block file as well as its replica. This data is uploaded to the TPA server. Data owner can check the file data integrity by sending verification request to the TPA. TPA sends challenge message to the cloud. Cloud generates proof of original data along with its replica. TPA verifies this proof and notifies verification details to the data owner. After checking the verification details data owner has facility to view and restore data backup copies from cloud.

IV. ALGORITHM

Following algorithms give detail steps for data update and verification [1]

A. Notation

F : File to be uploaded by the client to store on cloud.

m_i : The i^{th} file block of file.

$b_{i,j}$: The i^{th} block of replica F_j .

T: The replica merkel hash tree developed based on { m_i } .

Ti: The Replica-sub Tree of for block { m_i }.

$Ώ_i$: A set of tuples that are used as m_i's auxiliary authentication information (AAI).

R: Value stored in root of tree, this is hash value

$\sigma_{I,j}$: The homomorphic authenticator for $b_{i,j}$

sign $_{AUTH}$: Authorization signature for verification of TPA. numbers, headers and footers must not be used.

B. Data Updates and Verification

Insertion - I, update – M and deletion- D of data.

When data is get updated on cloud its replica of data need to be updated and verification data also need to be updated. At the client end file block is created and replicas are generated along with the request type – I/M or D.

Following algorithm represents the communication between client and cloud while updating the data.

Algorithm

Step 1: Client upload file with its block $m_{i,j}$ for new upload (I)/modification (M)/deletion(D)

Step 2: Cloud compute $b_{i,j}$ based on m_i then generate update notification as, {M/1_i,{ $b_{i,j}$ }} Or {D, I} {M/1_i,{ $b_{i,j}$ }} Or {D_i}

Step3: CSS locate subtree Ti, and Compute R’ with { $b_{i,j}$, $Ώ_i$ }.

For I/M: Creates new sub tree with $b_{i\ j}$ and update following indices:

{h ($b_{i,j}$) , $Ώ_i$, R’, sign)} M/I or {h ($b_{i,j}$) , h(b’ $_{ij}$) ,$Ώ_i$, R’, sign)}

For D: Delete Ti and updated indices and then compute R’.

Step 3: .Cloud compute σ' =(R, u, $b_{i,j}$)and sig’=(H(R’)) and send to TPA as metadata

C. Verification of data:

Verification of data is done with the verification of original data blocks as well as its replica present on CSS. Third party auditor TPA sends challenge message to CSS. CSS generates proof for the received request. It computes σ and μ for every block and its replica and sends the response to the TPA. TPA verifies the generated proof and generates a response as accept or reject. This response is conveyed to the user as verification result.

Following algorithm shows the communication between TPA and CSS.

Algorithm:

Step 1: TPA sends challenge request to CSS as:

{sign $_{AUTH}$,vi }

Step 2: CSS verify sign $_{AUTH}$ then compute,

$\mu\ i = \Sigma_j\ b_{i,j}$

$\sigma_j = \Pi_i\ \varepsilon_I\ \sigma_i$

Generate {μ_i, σ_j,sig}} and send to TPA

Step 3: TPA compute & verify R and indices of σ_j with σ'_j

Verify sig with R.

Step 4: TPA accepts, if all verifications passed else reject and generate audit report and send to the client

V. RESULT ANALYSIS

We have used 3 distinct systems connected in LAN. We have used ubuntu -15.04 Operating system for cloud and TPA environment. Hadoop 2.7.0 is installed on cloud for map reduce programming model on cloud node. We have built this system in java using jdk 1.7. For server side environment we have used apache tomcat 6.0. To store database we use mysql. For user side system we have used java swing application to communicate with cloud and TPA. A RESTful API is provided for third party web services.

A. Dataset:

To test data upload and verification for multiple files we have used Enron Dataset [18]. This dataset contains multiple files of different sizes varies from 1 kb to 1mb.

Performance Evaluation:

We have compared our system with multiple existing systems in terms of provided feature set. In the following table we have summarized the qualitative comparison among multiple systems.

TABLE I COMPARISON BETWEEN SYSTEMS

Characteristics	Existing System 1[17]	Existing System 2[1]	Propose d system
Blockless Verification	Yes	Yes	Yes
Stateless Verification	Yes	Yes	Yes
Infinite Verifications	Yes	Yes	Yes

DOI 10.17148/IJARCCE.2016.5774

Public Verifiability/ Auditability	Yes	Yes	Yes
Coarse-grained Verifiable Data Updating	Yes	Yes	Yes
Fine-grained Verifiable Data Updating	Yes	Capable	Capable
Variable-sized Data Blocks	Yes	Yes	Yes
Authorized Auditing	Yes	Yes	Yes
Authentication of Block Indices	No	Yes	Yes
Updating All Replicas at a time	No	Yes	Yes
Shared User Concept	Yes	No	Yes
User Revocation	Yes	No	Yes

We have provided multiple features to the user. Using user control panel user can upload and share the file among multiple user by creating a user group. Other user can download and edit the file. Owner has facility to delete uploaded files as well as can restore the file to cloud from replica copy. User has facility to revoke user from group.

VI. CONCLUSION

As per the different perspective analysis we can say about public auditing scheme, multiple existing systems already supports verification for data updates. But no auditing scheme verifies replica of data. Merkel hash tree data structure used for efficient data storage. It is also used replica sub-tree strategy to efficiently locate the data while updating process, while existing auditing techniques have some security issues. Updating overhead is reduces although multiple copies of original data is maintained due to efficient use of top down levelled MHT. It is also provide support for public auditing and authentication of block indices and verification of all replicas. There is need of such system that provide full data updates and authentication of block indices and verification of all replicas at once efficiently and at a same time. Shared data on cloud server minimizes the space requirement on cloud if many user wants to store same data on cloud they can share that data by storing only one copy of data. Efficient user revocation does the efficient management of users group on cloud. It also has the provision that if user doesn’t want to share his data he can upload data on cloud individually, this feature add more flexibility in approach.

ACKNOWLEDGMENT

We are thankful to all the authors of paper those, we have referred in our work.

REFERENCES

[1]. MuR-DPA: Top-down Levelled Multi-replica Merkle Hash Tree Based Secure Public Auditingfor Dynamic Big Data Storage on Cloud, Chang Liu, Rajiv Ranjan, Chi Yang, Xuyun Zhang, Lizhe Wang, Senior Member, IEEE, and Jinjun Chen, Senior Member, IEEE.

[2]. 2. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "A View of Cloud Computing," Communications of the ACM, vol. 53, pp. 50-58, 2010.

[3]. 3. R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg, and I. Brandic, "Cloud Computing and Emerging IT Platforms: Vision, Hype, and Reality for Delivering Computing as the 5th Utility," Future GenerationComputer Systems, vol. 25, pp. 599-616, 2009.

[4]. 4. C. Liu, J. Chen, L. T. Yang, X. Zhang, C. Yang, R. Ranjan, and K. Ramamohanarao, "Authorized Public Auditing of Dynamic Big Data Storage on Cloud with Efficient Verifiable Fine-grained Updates," IEEE Transactions on Parallel and Distributed Systems, in press, 2013.

[5]. 5. R. C. Merkle, "A Digital Signature Based on a Conventional Encryption Function," in Proceedings of A Conference on the Theory and Applications of Cryptographic Techniques on Advances in Cryptology (CRYPTO '87), 1987, pp. 369-378.

[6]. 6. H. Shacham and B. Waters, "Compact Proofs of Retrievability," in Proceedings of the 14th International Conference on the Theory and Application of Cryptology and Information Security (ASIACRYPT '08), 2008, pp. 90 – 107

[7]. 7. E. Shi, E. Stefanov, and C. Papamanthou, "Practical Dynamic Proofs of Retrievability," in Proceedings of the 2013 ACM SIGSAC Conference on Computer and Communications Security (CCS '13), 2013, pp. 325-336.

[8]. 8. S. Subashini and V. Kavitha, "A Survey on Security Issues in Service Delivery Models of Cloud Computing," Journal of Network and Computer Applications, vol. 34, pp. 1-11, 2010.

[9]. 9. B. Wang, S. S. M. Chow, M. Li, and H. Li, "Storing Shared Data on the Cloud via Security- Mediator," in 33rd IEEE International Conference on Distributed Computing Systems (ICDCS '13), Philadelphia, USA, 2013.

[10]. 10. B. Wang, B. Li, and H. Li, "Oruta: Privacy- Preserving Public Auditing for Shared Data in the Cloud," in Proceedings of the 2012 IEEE Fifth International Conference on Cloud Computing (CLOUD '12), Hawaii, USA, 2012, pp. 295-302.

[11]. 11. B. Wang, B. Li, and H. Li, "Public Auditing for Shared Data with Efficient User Revocation in the Cloud," in Proceedings of the 32nd Annual IEEE International Conference on Computer Communications (INFOCOM'13), Turin, Italy, 2013, pp. 2904-2912.

[12]. 12. C. Wang, Q. Wang, K. Ren, and W. Lou, "Privacy-Preserving Public Auditing for Data Storage Security in Cloud Computing," in Proceedings of the 29th Annual IEEE International Conference on Computer Communications (INFOCOM'10), San Diego, USA, 2010, pp. 1 - 9.

[13]. 13. Q. Wang, C. Wang, K. Ren, W. Lou, and J. Li, "Enabling Public Auditability and Data Dynamics for Storage Security in Cloud Computing," IEEE Transactions on Parallel and Distributed Systems, vol. 22, pp. 847 - 859, 2011.

[14]. 14. P. Williams, R. Sion, and A. Tomescu, "PrivateFS: A Parallel Oblivious File System," in Proceedings of the 2012 ACM Conference on Computer and Communications Security (CCS '12), 2012, pp. 977-988.

[15]. 15. Y. Zhang and M. Blanton, "Efficient DynamicProvable Possession of Remote Data via Update Trees," IACR Cryptology ePrint Archive 2012:291.

[16]. 16. Y. Zhu, H. Hu, G.-J. Ahn, and M. Yu, "Cooperative Provable Data Possession for Integrity Verification in Multi-Cloud Storage," IEEE Transactions on Parallel and DistributedSystems, vol. 23, pp. 2231-2244, 2012.

[17]. 17. B. Wang, B. Li, and H. Li, “Public Auditing for Shared Data withEfficient User Revoation in the Cloud,” in the Proceedings of IEEE INFOCOM 2013, 2013, pp. 2904–2912.

[18]. 18. Enron dataset “www.cs.cmu.edu/~./enron/”

DOI 10.17148/IJARCCE.2016.5774

BIOGRAPHIES

Kaveri Sonawane completed her graduation from K.K. Wagh Engineering College, Nasik, Maharashtra. Presently, she is Post-Graduate Student at MET Bhujbal Knowledge City Institute Of Engineering, Nasik, Maharashtra, India. Her research of interest includes Cloud, Big Data.

Prof. Ranjana Dahake presently she is working at MET Bhujbal Knowledge City Institute of Engineering, Nasik, Maharashtra, India as a Assistant Professor. She has presented/Publish papers at national and international conference as well in journals on various aspects of the computer engineering. Her research of interest includes image processing, cloud computing, data mining.

DOI 10.17148/IJARCCE.2016.5774