International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2013): 4.438

Authorized Auditing of Big Data Stored on Cloud with Auditability Aware Data Scheduling

Surapriya Swain[1], Prof. Saurabh Gupta[2]

[1]Department of Computer Engineering, GSMCOE, Balewadi, Savitribai Phule University,Pune,India

[2]Professor, Department of Computer Engineering, GSMCOE, Balewadi, Savitribai Phule University,Pune,India

Abstract: ***Cloud computing otherwise known as on demand computing. It provides the services over the internet. It has the provision of facilitating users to store and access their data in and from cloud server by sitting anywhere and on any device. Storing data in cloud server also opens up so many security threats as data is accessed over internet and client has no direct control over data once uploaded into cloud server. The risks are like authentication of client and integrity of data. To ensure the integrity of data there is need of frequent auditing of client's data in cloud server. Client hands over the task of auditing to the TPA which is also an external entity and which should be a trusted and authorized one. In the cloud environment requests come from user for uploading and editing of data, from TPA for integrity checking of data. So the server needs to serve the request in such a way that will utilize the time and resource efficiently. For this we have to make TPA audit aware data scheduling. Auditability aware data scheduling handle resource utilization properly. This paper provides an analysis on authorized auditing and auditability aware data scheduling in cloud server. Also focus on fine grained update request from user.***

Keywords: Cloud computing, Big Data, Authorized Public Auditing, Fine-grained Updates, TPA

1. Introduction

CLOUD computing is being intensively referred to as one of the most influential innovations in information technology in recent era. By using resource virtualization cloud delivers us computing resources and services in a pay-as-you-go mode. Today world is moving on digitization and cloud computing is best concept to handle big datasets. Various cloud computing services are categorized into Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS) and last one is Software-as-a-Service (SaaS).

Many international IT corporations now offer powerful public cloud services to users on a scale from individual to enterprise all over the world various examples of this are Amazon AWS and IBM Smart Cloud. As we know the current development and proliferation of cloud computing is rapidly growing, debates and hesitations on the usage of cloud still present. Data security and data privacy are some of the major concerns in the adoption of cloud computing. Users lose their direct control on data when they store data on cloud as compared to conventional systems.

In our proposed work we will address the problem of integrity verification for big data storage on cloud. We call this problem as data auditing when the verification is conducted by a trusted third party i.e. TPA. TPA is called as an auditor. From cloud users perspective it is named as auditing-as-a-service. In a remote verification scheme, the cloud storage server (CSS) cannot provide a valid integrity proof of a given proportion of data to a verifier unless all these data is intact. To ensure integrity of user data stored on cloud service provider, this support is of no less importance than any data protection mechanism deployed by the cloud service provider (CSP) [16], no matter how secure they seem to be, in that it will provide the verifier a piece of direct, trustworthy and real-timed intelligence of the integrity of the cloud user's data through a challenge request. It is especially recommended that data auditing should be conducted on a regular basis.

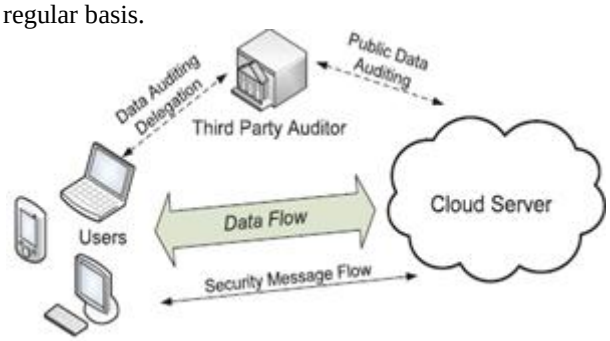

Figure 1: Relationship between participating Parties in Public Auditing Scheme

The three main contributions of our proposed work are described as follows
1. Authorized third party auditing
2. Fine grained dynamic data updates
3. Auditability aware data scheduling

2. Literature Survey

In past, lots of work has been done on cloud data security different techniques were used to provide security to cloud data but all the system is having some advantages and disadvantages. Existing methods for protecting user data include data encryption prior to storage and user authentication procedures prior to storage or retrieval of data after those building secure channels for data transmission over the cloud. In these existing systems the algorithms used are cryptographic and Digital signature based.

First work is by Ateniese et al[1] who consider public auditability in provable data possession model for ensuring possession of files on un trusted storages. Ateniese present a model in which RSA based homomorphic tags are used. With the help of this technique public auditability concept is

achieved. But the problem with this model is that it does not support dynamic data operation and also suffer security problems [8]. Another research by Wang, et al.[6] considered dynamic data storage in a distributed scenario which is a better idea. He proposed challenge response protocol can both determine the data correctness and locate possible errors but this model only considered partial support for dynamic data operations [9].

Kaliski presented a proof of retrievability model [2].This scheme can only be applicable to static data storage for verifying the integrity of data. The main disadvantage of this model as it does not support public auditability. Extended research on this done by Shacham. Shacham, et al [3] provided a scheme with stateless verification and the design is an improved PoR scheme with full proofs of security in the security model. In this model they use publicly verifiable homomorphic authenticators built from BLS signatures based on which the proofs can be aggregated into a small authenticator value by using this public irretrievability is achieved. The main concern comes in front with this is the authors only consider static data files which are not preferable because our main concern is about big data files [10]. Ateniese et al [17] proposed an extended scheme based on POR and PDP support only partial data dynamics and supporting limited number of challenges.

One research was there on MAC based scheme which has the disadvantages like the number of times a particular data file can be audited is limited by the number of secret keys that must be fixed a priori. So the problem arises here is once all possible secret keys are exhausted after that the user then has to retrieve data in full to recomputed and republish new MACs to TPA[10] .Here in this scheme TPA also has to maintain and update state between audits that is to keep track on the revealed MAC keys. It can only support static data and cannot efficiently deal with dynamic data at all. So this is a big issue to be solved when considering Big data.
HLA based scheme- There is need of system which can verify integrity of data without retrieving data blocks present. So there is another method presented that is HLA scheme was used for this purpose. The only difference between HLA and MAC is that HLA can be aggregated. The main issue with this system is that data can be retrieved only if linear combinations of same block are used [10]. More research is going on to support both static and dynamic data updates with higher efficiency. There is no support for different type of updates and for that work is going on to provide fine grained dynamic data updates facility.

3. Implementation Details

3.1 Problem Definition

In previous research it is shown that cloud environment provide various advantages by providing infrastructure as a service and maintenance as a service. It relieves the burden of user's task but security became a major concern in all time. User hire a TPA to check the integrity of data stored in cloud server. But again the problem arises whether user should trust or not on TPA . Another concern is related to the utilization of resources in cloud environment. There are number of resources as well as requests. There is no better way to serve the requests within a particular time and with available resource. There are also an increasing range of Information Communication Technology (ICT) vulnerabilities and threats that have to be effectively and efficiently managed. As a consequence, the confidentiality, integrity, availability and reliability of computerised data and of the systems that process, maintain and report these data are a major concern to audit.

Previously scheduling algorithms were performed in grid but reduces the performance by requiring advance reservation of resources. In cloud environment due to scalability of resources, manually allocate resources to task is not possible. Scheduling should be done in such a way that it will utilize the resources efficiently and also adopt the changes in environment configurations.

3.2 Proposed System

The various main steps of our proposed scheme is described as follows:

1. The client will generate keying materials via KeyGen and FileProc after that he upload the data to CSS. Different from previous schemes here in our scheme the client will store a RMHT as metadata.
2. After that the client will authorize the TPA by sharing a value sigAUTH.
3. Verifiable DataUpdating: the CSS performs the client'sfine-grained update requests via PerformUpdate on user's data.
4. Client runs VerifyUpdate to check whether CSS has performed the updates on both the data blocks and their corresponding authenticators (used for auditing) honestly.
5. Challenge, Proof Generation and Verification: Describes how the integrity of the data stored on CSS is verified by TPA via GenChallenge, GenProof and Verify.
6. In auditability aware data scheduling scheme we are clustering various tasks submitted in an application both from the user and auditor on the basis of their priority.

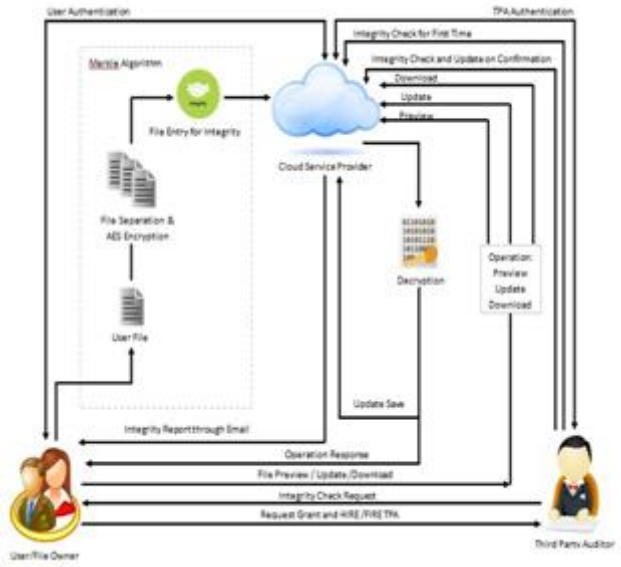

Figure 2: System Architecture

In our proposed work the three main components are User, TPA and CSP. When user wants to store its data in CSP, first time it should be authorized from CSP. User sends request to

CSP and if CSP grants the permission then he can store its data in cloud server. Otherwise the same message comes to register in CSP. When user data is uploaded in CSP it is not the actual data. First it is encrypted using the AES algorithm then it is broken into 3 separate parts and stored anywhere in the server. By this process data security is maintained that even CSP cannot read user's information. But for more security purpose User hires TPA (Third party auditor) who frequently audit user's information stored in CSP. TPA is not authorized to access the actual data. It will only get the hash value of the separated parts. By comparing the current hash value of the encrypted file with the previous hash value it finds whether integrity lost or not. If both value mismatches TPA informs the user and if the changes is done from the user site then TPA will ignore this. To find the hash value markele hash tree is used.

One more work in this system is to provide fine grained updates. As user file is separated into three parts instead of retrieving the whole file user can see the particular portion of the file to do changes. The file is available to be viewed and updated.

Finally for proper utilization of the resources the auditing process should be TPA aware which is called auditability aware data scheduling.

Data Integrity Verification by TPA Algorithm**:**

1. Start
2. Read data owner id(udoid)
3. If (doid ≠ udoid)
4. Stop
5. Read file name from CSP
6. Retrieve No. of blokes from TPA
7. Select the blocks number that the user want to verify.
8. Get the auxiliary information for block chal from TPA
9. Based on Auxiliary information generate new root for MHT
10. If (new root ≠ root) file modified
11. Else File not modified
12. Stop.

AES algorithm is used to store the data in encrypted form in cloud server. So when TPA does the auditing it only gets the illusion of original files. The values on which TPA calculates or check the integrity is actually the hash value of encrypted file calculated collectively. The TPA is only allowed to check the integrity nothing else. It checks the integrity for the first time then checks whether integrity preserved and lastly check whether integrity remains or lost. User can any time grant or revoke the privilege from TPA. User has the privilege to upload, download and edit data. User's edit request is also served for a particular part of the file instead of retrieving the whole file.

3.3 Auditability Aware Scheduling

As we know to the cloud server so many requests comes from user and TPA. User sends request to either upload file or view or to update file. TPA sends request to CSP only to check the integrity of file. So when user is updating the file, no need to check the integrity. CSP has to handle all the request in priority basis. Each request is assigned with some priority. Usually the requests comes from user is given more priority than those comes from TPA. User's work is done first. User's tasks are uploading file, preview file and update file. So when user is uploading TPA cannot check anything and after update of file it will check for integrity. There should be proper scheduling between the tasks and resource to improve the utilization and efficiency of resources. Some characteristics of priority based scheduling are:

1. Starvation can happen to low priority process.
2. The waiting time gradually increases for the equal Priority processes.
3. Higher priority process has smaller waiting time and response time.

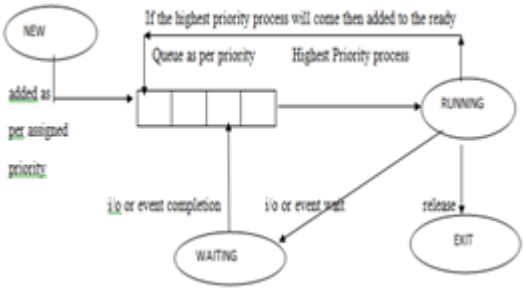

Figure 2: Priority based scheduling

Algorithm for scheduling:

1. for i = 0 to i <main queue-size
2. if priority (task i+1) > priority (task i)
3. then
4. add task i+1 in front of task i in the queue
5. end if
6. end for

In this scenario, if request comes from TPA to CSP for data auditing and at the same time user wants to update his data, then user will be able to update the data first. If two TPAs send request for auditing the same file then the request from first TPA will be served first.

4. Results

Many of project works developed previously which can only store data and share data between large numbers of users in a group. In our proposed work we have presented a third party auditing scheme to construct a secure data auditing mechanism. Also our work presents an auditability aware data scheduling scheme which is based on priority.

In this scheme the major merits are: (1) data security (2) privacy protection (3) Auditing details to the data owner (4) Auditability aware data scheduling.

Here we are going to evaluate the performance of our proposed scheme in terms of the computation overhead introduced by each operation. Request and resources are taken as the computing parameter. When the number of requests increases at the same time, it is to check whether they are served within a particular time. The waiting time is measured for each request.

Resources are coming from TPA and Users for different purposes. When the user requests are more then they will be

served first by the available resources. The TPA requests are served after user request are served.

We analyze the computation complexity for the following operations like system setup, new user grant, new file creation , file deletion and user revocation and file access. The graph shows the files which are uploaded and done editing. The graph is the plotting of time v/s file i.e which file take how much time for updating.

With an environment comprising of a specific software and hardware configuration the efficiency is measured. Hardware requirement includes a Pentium 4 processor, Hard disk of 20 GB and 512 mbdd Ram. Software requirement includes an operating system of Windows XP/7, Java as coding tool, MySQL as database and Netbeans tool.

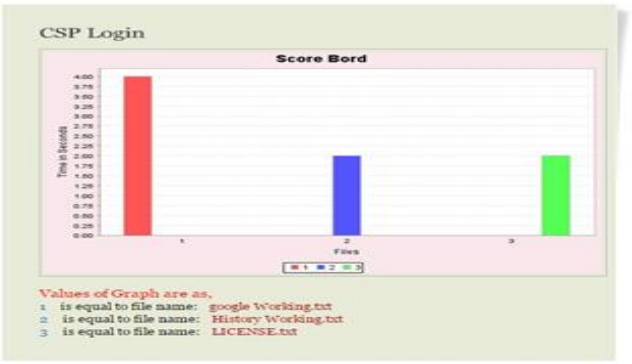

Figure 3: Result analysis

The above graph shows the files in X axis and time taken to serve the file operations in Y axis. It implies that numbers of files are in CSP and different file takes different time for various purpose. Files are the resources for which User and TPA sends different request like update or integrity check.

5. Conclusion and Future Work

Today the example of big computing paradigm in cloud computing is to store the big datasets. Important aspect for cloud user is cloud data security and privacy. In this paper we have provided an implementation of authorized auditing and efficient fine grained updates. We have also implemented auditability aware data scheduling which tries to utilize the cloud resources to serve the clients with maximum throughput. Our proposed system has three major components which provide a secure, authorized and efficient auditing of data and modification of data in cloud environment. Security is provided by restricting forged TPA from auditing user's data without user's concern. Efficiency is achieved by supporting small updates. This paper implemented a priority based scheduling algorithm to provide proper utilization of cloud resources among the tasks coming to CSS.

Our proposed work's result shows the result for text file. We can upload text file and provide fine-grained updates for those file as it is an universally accepted format. In future the work will focus for different types of file. For security purpose more layer of authentication to TPA will be provided.

6. Acknowledgement

I would like to express my thanks to my guide Prof. Saurabh Gupta for his highly appreciable support and encouragement also to my HOD Prof. Ratnaraj Kumar . Their guidance is a force behind the completion of this paper. I am grateful for all the suggestions and hints provided by him. My acknowledgment of gratitude to all who has supported to make it possible.

References

[1] R. Curtmola, O. Khan, R.C. Burns, and G. Ateniese, ''MR-PDP:Multiple-Replica Provable Data Possession,'' in Proc. 28th IEEE Conf. on Distrib. Comput. Syst. (ICDCS), 2008, pp. 411-420.
[2] A. Juels and B.S. Kaliski Jr., ''PORs: Proofs of Retrievability for Large Files,'' in Proc. 14th ACM Conf. on Comput. andCommun. Security (CCS), 2007, pp. 584-597.
[3] H. Shacham and B. Waters, ''Compact Proofs of Retrievability,'' in Proc. 14th Int'l Conf. on Theory and Appl. of Cryptol. and Inf. Security (ASIACRYPT), 2008, pp. 90-107.
[4] 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 Distrib. Syst., vol. 22, no. 5,pp. 847-859, May 2011.
[5] G.Ateniese, R.B. Johns,R. Curtmola, J.Herring, L. Kissner,Z. Peterson, and D. Song, ''Provable Data Possession at Untrusted Stores,'' in Proc. 14th ACM Conf. on Comput. andCommun. Security (CCS), 2007, pp. 598-609.
[6] C. Wang, Q. Wang, K. Ren, and W. Lou, ''Privacy Preserving Public Auditing for Data Storage Security in Cloud Computing,'' in Proc. 30st IEEE Conf. on Comput. andCommun. (INFOCOM), 2010, pp. 1-9.
[7] 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 Distrib. Syst., vol. 22, no. 5, pp. 847-859, May 2011
[8] Cong Wang, Sherman S.-M. Chow, Qian Wang, KuiRen, and Wenjing Lou, "Privacy-Preserving Public Auditing for Secure Cloud Storage," IEEE Transactions On Cloud Computing, Year 2013.
[9] C. Wang,"Toward publicly auditable secure cloud data storage services," IEEE Network , vol. 24, no. 4, pp. 19–24, 2010.
[10] G. Ateniese, R. Burns, R. Curtmola, Peterson, and D. Song, "Provable Data Possession at Untrusted Stores," Proc. 14th ACM Conf. Computer and Comm. Security (CCS '07), pp. 598-609, 2007.
[11] A. Juels "Pors: Proofs of retrievability for Large Files," Proc. 14th ACM Conf. Computer and Comm. Security (CCS '07), pp. 584-597, 2007. \
[12] Privacy preserving public auditing for Secure Cloud Storage", Cong Wang, Sherman S.-M. Chow, Qian Wang, Kui Ren.
[13] Zissis, D. and D. Lekkas, 2011. ,,,,Addressing Cloud Computing Security Issues,"" Future Gen. Comput. Syst., 28(3): 583-592.

[14] R. Curtmola, O. Khan, R.C. Burns, and G. Ateniese, ‘‘MR-PDP: Multiple-Replica Provable Data Possession,’’ in Proc. 28th IEEE Conf. on Distrib. Comput. Syst. (ICDCS), 2008, pp. 411-420.
[15] C. Wang, Q. Wang, K. Ren, and W. Lou, ‘‘Privacy-Preserving Public Auditing for Data Storage Security in Cloud Computing,’’ in Proc. 30st IEEE Conf. on Comput. and Commun. (INFOCOM), 2010, pp. 1-9.
[16] G. Ateniese, S. Kamara, and J. Katz, ‘‘Proofs of Storage From Homomorphic Identification Protocols,’’ in Proc. 15th Int’l Conf. on Theory and Appl. of Cryptol. and Inf. Security (ASIACRYPT),2009, pp. 319-333.
[17] G. Ateniese, R.D. Pietro, L.V. Mancini, and G. Tsudik, ‘‘Scalable and Efficient Provable Data Possession,’’ in Proc. 4th Int’lConf. Security and Privacy in Commun. Networking (SecureComm),2008,pp.1-10.

Author Profile

Mrs Surapriya Swain, received the B.E degree in Computer Science and Engineering from B.I.E.T in 2005, affiliated to Biju Pattanaik University Of Technology, Odisha, India. She is now pursuing her M.E (Computer Engineering) in G. S. Moze College of Engineering under Savitribai Phule Pune University.

Mr. Saurabh Gupta, working as an Assistant Professor with G. S. Moze College of Engineering under Savitribai Phule Pune University (Maharashtra, India). He has received his B.E(IT) degree from Agra University(UP, India) in 2004 and M.E from BITS Pilani(Rajstan, India) in the year 2009.