OPEN ACCESS

International
Journal
Of Modern Engineering Research (IJMER)

Authenticated and unrestricted auditing of big data space on cloud through valid and efficient granular updates

Suhas. S[1], Ramani. S[2]
[1, 2] *School of Computer Science and Engineering, Vellore Institute of Technology, Vellore, India*

Abstract: *Cloud unlocks a different era in Information technology where it has the capability of providing the customers with a variety of scalable and flexible services. Cloud provides these services through a prepaid system, which helps the customers cut down on large investments on IT hardware and other infrastructure. Also according to the Cloud viewpoint, customers don't have control on their respective data. Hence security of data is a big issue of using a Cloud service. Present work shows that the data auditing can be done by any third party agent who is trusted and known as auditor. The auditor can verify the integrity of the data without having the ownership of the actual data. There are many disadvantages for the above approach. One of them is the absence of a required verification procedure among the auditor and service provider which means any person can ask for the verification of the file which puts this auditing at certain risk. Also in the existing scheme the data updates can be done only for coarse granular updates i.e. blocks with the uneven size. And hence resulting in repeated communication and updating of auditor for a whole file block causing higher communication costs and requires more storage space. In this paper, the emphasis is to give a proper breakdown for types of fixed granular updates and put forward a design that will be capable to maintain authenticated and unrestricted auditing. Based on this system, there is also an approach for remarkably decreasing the communication costs for auditing little updates.*

Index Terms: *Cloud Computing, Big Data, Fine grained updates, Data Security, Authenticated Auditing.*

I. Introduction

Cloud Computing is setting the new trend in the present IT field. It is a latest computing framework derived from grid computing, parallel computing, grid computing, virtualization and utility computing [1]. Through virtualization of resources, cloud computing can bring services and resources required to the customers in a pay-per-use mode. Cloud Computing provides services in three ways and they are IaaS-Infrastructure as a Service, PaaS-Platform as a Service, SaaS-Software as a Service [2]. Big Data Analytics is the newly researched in the present IT sector. Big Data usually refers to the term for the set of data that is very huge in volume and also so complex that it is hard to process the data through normal data processing tools [3]. And Cloud Service Storage (CSS) is the largest source of dynamic and traditional Big Data storage. Big Data is usually referred by 3 Vs – high volume, high variety and high velocity. High volume refers to huge amount of datasets. High variety refers to different types of data that make up the dataset. And high velocity refers to constant change or updating of datasets [4]. Hence Cloud and Big data are the newest research fields in the IT sector. Even with all the present development and research of cloud computing is rapid and efficient there is always a debate and uncertainty on the usage. One such concern for the users is of Data Privacy/Security [5, 6]. Contrast to the usual systems, users doesn't have a control on their data.

In this paper, the emphasis is on investigating the various problems of reliability verification for the big data space on cloud. In real world the above problem is commonly known as data auditing or "Auditing_As_A_Service" through a cloud user's point of view and it is done through a verified third party auditing service [7, 8]. In an isolated authentication system, the valid reliability proof cannot be given by the Cloud Storage Server (CSS) to a trusted verifier until all the data is in one piece. And no matter how much secure the data security mechanism provided by the Cloud Service Provider (CSP) it is recommended to use a challenge request for auditing and also that data auditing is to be done on a regular basis for users who have highly varying data from time to time. Also for the users with very high security demands for their data the above process is recommended.

Literature contributions can be outlined as follows:

1) In this paper we put forth a formal analysis of various types of fine granular data updates on uneven sized blocks of file dynamically in a data set.

2) For an efficient security system, we integrate an extra authentication method. The goal of this system is to remove dangers of unauthenticated auditing requests from illegal third party auditors (TPA's). The above process can concluded into a single term as Authenticated Auditing.

3) We also examine how to enhance the effectiveness of validating small updates that are very frequent which form the core of some cloud and big data applications like social networks. Experimental outcomes of proposed system show that it can considerably reduce communication costs as compared to existing methods.

Paper Layout: The remainder of this paper is categorized as follows. Part 2 deals with Literature Survey. Part 3 gives details of existing system and problem statement. Part 4 gives details on the proposed system and also analysis on how to support requests for fine granular updates. Part 5 describes analysis of security for the design. Part 6 deals with the experimental results and part 7 describes conclusion and future enhancements.

II. Literature Survey

Scalable and elasticity are the two main feature of the cloud which can be treated as advantages when compared to traditional systems. Also efficiency in sustaining dynamic data updates is also important. Protection and Privacy of dynamic big data has been a matter to study in the past [9, 10, 12, 15]. Our focus is mainly on frequent and small data updates. This is a important factor since it exists in real world cloud applications like social networks and business transactions. Also sometimes users in the cloud may decide to divide big datasets into smaller datasets for easy management and processing uses. But among all the problems in today's cloud is of security or privacy [5,6,11]. Data privacy and security also the most raised fears about using the cloud since the users does not has a direct control over their data [6,13].

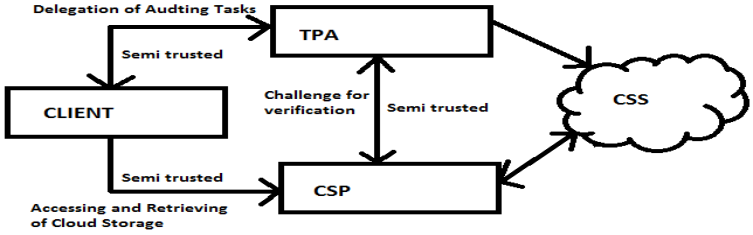
There were many solutions proposed for the above problems of security and small dynamic data updates over a cloud. The first model was Proof Of Verifiability by Jules [15]. But the disadvantage in this scheme is that it can be applied only to static data storage. There were other schemes like Proved data Possession which consists of verifiability tags which computed tags through which authenticator can authenticate the integrity of a part of a file that is outsourced. There was a scheme that used BLS method for public auditing purpose [16]. It had RSA algorithm for encryption and decryption. The production and authentication of validity proofs were same as verifying and signing of BLS signatures. After that both the Proof of verifiability and proved data possession were unified into single model name POR. Since public audit ability and coarse grained updates cannot be supported by default in the above systems there is a need for a new system. The latest one is to divide files into different blocks. But this results in expensive communication difficulties and storage costs. And hence there is need for support for dynamic and frequent small updates.

III. Problem Statement and Analysis

One of the reasons for the cloud to be so popular is that the elasticity feature of it plays a big role in bringing cost-effectiveness to the picture. An example can be considered of a mobile company using cloud platform to provide online video on demand services to customers. The no. of customers watching could drastically vary from millions to a mere hundreds according to the video. To provide for this variation demand the company has to purchase maximum no. of hardwares and other processing units. But even though the overall system is working fine, most of the time some part of the system is idle and does not account to any workload. Hence here is a situation where cloud can save millions of investment capital by providing elastic feature.

Some of the cloud providers are Amazon S3 or EC2, Microsoft Azure and others. Hence we can see that how scalability and elasticity enhance the support of dynamic data updates and are the core feature of cloud computing. A big data application is a collection of related and unrelated data where the datasets are huge in number. One such example of a big data can be given as cloud storage. A lot of big data applications handle datasets which are small and there is a frequent updation of information on the datasets. Hence usually users of the cloud divide this huge amount of datasets into small part of data before uploading to the storage. Only few auditing schemes support small updates over big data and also the security of these schemes is not assured.

The following diagram depicts the role of the entities involved in the scheme



In the above diagram the client has access to the cloud storage via a service provider provided by a Cloud Service Provider. The client does not have complete trust on the service provider. He also does not trust the auditor completely. When the client asks for the third party auditor to do some auditing tasks the auditor has to access the data to be audited through the Cloud server storage through the service provider. The Cloud service provider asks for challenge verification. This challenge verification is based on RSA algorithm and BLS signature scheme and is created in accordance with the client. So that the client can pre determine as to which auditor can be given data without any security concerns for auditing.

Some of the present auditing methods have the capability to support full data dynamics. Only insertion, updations and deletions on same-sized blocks are supported. There is a need for full dynamic support of variable –sized block updates also known as coarse-grained updates. Coarse grained block updates are always more complicated than fine granular updates. For every insert operation there is a new block created by the Cloud Server Storage. But when there are many no. of insertions and updates to be performed, the storage wasted and required is huge. This situation can be fixed if fine granular data updates are supported and hence efficiency is improved. There is always a huge amount of communication overhead involved in verifying large no. of coarse-grained updates. If the fine granular updates are supported it can provide extra flexibility and also improves the efficiency.

The main assumption of this paper is that the Cloud Server Storage will provide honest data query services to all the clients. Also if a user has to retrieve a particular part of his/her data which is stored on the Cloud Server Storage, CSS should not give an incorrect answer. This assumption can be concluded into a single word as Reliability. This is a fundamental service guarantee of Cloud Server Storage.

IV. The Proposed system

We use a data structure called Hashed Rank Merkle Tree. It is same as a binary tree; each node has a max of 2 nodes. It can be considered as a complete binary tree where each non leaf node has 2 child nodes. Info at a node N in a hash tree T can be represented as (h,R_n) in which h is the hash value and R is the rank of that particular node. For leaf node L.N which has data d we can represent as H=H(di),rn= Si, the parent node can be constructed as Np= (h(H1||H2),(rn1+rn2) || is the concatenation operator.

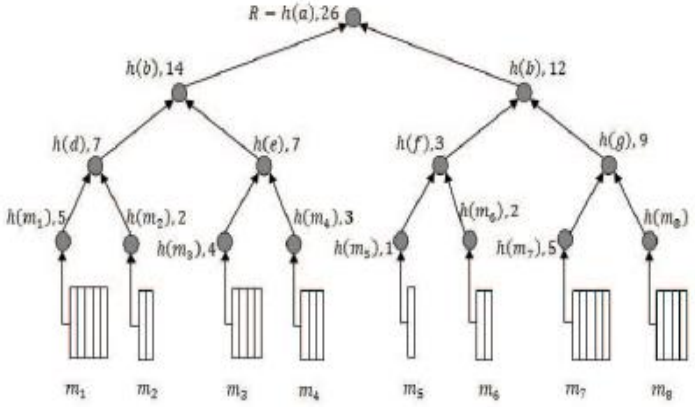


Fig 2. A typical example of Merkle Hash Tree

The leaf contains set of hash values selected from all of its upper levels since the root value can be calculated through (mi, O). The update operations defined in our scheme are as follows. Modification is an operation which can be done in two ways- partial and complete. Partial modification deals with a part of a certain block that needs to be updated. And whole modification deals with the situation where a whole block needs to be updated from the tree structure. There is also a operation of insertion if a whole block needs to be inserted on the tree structure for containing new data and split block operation to remove a part of data in a block and is to be replaced with a new block.

The authenticated auditing scheme is carried out using a series of algorithms. Key generation algorithm is used to generate security information like private key, public key and signature for encryption and decryption purposes. This task is carried out by RSA and BLS techniques. The user's data will be stored in the form of a Hashed Merkle tree as metadata. The user will authorize the auditor of his choice by sharing the signature that was generated. File Preprocessing deals with uploading and processing of the file on the Cloud Servicer Storage. The CSS makes that the file has been uploaded by the appropriate user with enough permission.

Challenge algorithm and verifiable update algorithm is done to verify the integrity of the user. After the user uploads the data onto the CSS it completes the update demands through update operation and then user does

update verification to verify whether the CSS has completed the updates on the correct data blocks and their particular authenticators.

Following the above operations in the proposed scheme there is a need to define a fine granular update demand for a file which is divided into n different sized blocks, in which each block consists of segments of a fixed size. If there is a Merkle hash tree that must have updated with operation discussed earlier for CSS to send the root R for the user to validate the integrity of the operation.

After the analysis done above, we can see that a large amount of small updates, be it insert, or deletion or updation. And also for each operation the partial modification is invoked every time and hence this is a big communication overhead. Therefore the emphasis is also on optimizing the partial modification operations and hence making the system efficient.

Even though the proposed system can support the fine granular updates, the user must retrieve the whole file block from the CSS for finishing the authentication work. This means that the user is the only entity who has the private key for authentication. But the user does not have the frequent update factor stored locally. Hence the extra communication overhead will be very expensive for huge no. of frequent updates.

For the split block operation the procedure is same as the basic scheme since there is no new data put into the Merkle hash tree T. hence getting back the whole data block cannot be avoided and is inefficient. For the other operations there is no old data in new blocks and therefore the other procedure remains the same.

The strategy applied here is based on RSA algorithm and can be used for Proved Data Possession which in turn can be used to achieve authenticated auditing and fine granular update requests. This process will be easier since RSA supports coarse grained updates. The auditing also can be applied in batch jobs since there is no change in authenticators and verifiers. This strategy can be used into our system so as to avoid TPA from browsing through the authenticated file segments via sequences of integrity checks for a file block. Also we can reduce the amount of challenges for the same set of file blocks. When there are frequent updates the attack's success rate of authorized access is very low since there is a high chance that challenged blocks are already updated.

V. Analysis of Security

In the verification procedure of the scheme the emphasis is on preventing the CSS cheating the valid TPA about the status of user's data. This is the same concept as Proved Data Possession – PDP. Aside from the new authentication process the only variation in comparison to earlier systems is that of the Merkle Hash Tree and different sized blocks. Also the security of this scheme is enhanced by the use of a signature scheme. Usually the schemes with signatures have a greater efficiency and are also more secure. An invalid or unauthorized TPA is an outside auditor who wants to challenge the client's data stored onto the CSS without the permission of the client. This is not available in the earlier auditing schemes. Hence with the unique authentication process no outside TPA without the user's permission can be able to audit the data without his permission. For even higher security demands the user add a authentication message to make each auditing unique so as to avoid mix up results of auditing work between different auditors. But this enhancement has its own limitations and user has to be online for most of the time.

The second part of the security analysis is to verify the updates over the client's data stored on the CSS. The main analysis to be done here is whether the CSS (partially trusted) has carried out the data updates correctly or not. In the process of updating the data the CSS must be able to honestly provide with the report on updates done on the data correctly. Also the CSS should be able to reduce the communication overhead for this process.

VI. Experiment results

We conducted our experiments on the Amazon S3 cloud, a pay as you go type cloud service provider. The user first has to register onto the cloud and then he can upload the files into a folder called as a bucket. The user uploads his files onto the CSS and subsequently the keys and signatures are generated for encryption and decryption purposes. This information is shared with the TPA of his/her choice. This means that the user has chosen the particular and trusted auditor for auditing tasks over his data hence enhancing security and privacy of the data. After this phase the auditor with the proper credentials challenges the CSS for integrity of the data.

The CSS after confirming the details of the challenging TPA forward the data for auditing tasks. The whole outline of the operation is done with accordance with the user and reports are generated after the auditing tasks.

The results generated show reduction in communication overhead compared to the existing scheme. For each update done on the data the report is generated. The report contains information about the total amount of data retrieved for the existing scheme and proposed scheme.

VII. Conclusion and Future Enhancements

In this paper, the emphasis is to give a proper breakdown for types of fixed granular updates and put forward a design that will be capable to maintain authenticated and unrestricted auditing. Based on this system, there is also an approach for remarkably decreasing the communication costs for auditing small updates. Analysis and experimental results have shown that this scheme can meet high security demands while also providing flexibility and elasticity. Also adding the feature of reducing communication overheads for bid data applications through more frequent amount of small updates which can be applied in social networks and business transactions. The future enhancement for this paper can be improving server side security issues while providing confidentiality and data availability. Also the plan is to investigate modifications in authenticated auditing for better security. Also the enhancement on meeting the Quality of service metrics such as data security, storage and computation can be done.

REFERENCES

[1] Zhang Xin, Lai Song-qing, Liu Nai-wen, "Research on cloud computing data security model based on multi-dimension," Information Technology in Medicine and Education (ITME), 2012 International Symposium on (Volume:2), 3-5 Aug. 2012

[2] M. 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, no. 4, pp. 50-58, 2010.

[3] Garlasu D, Sandulescu V, Halcu I, Neculoiu G, Grigoriu O, Marinescu M, Marinescu V, "A big data implementation based on Grid computing," Roedunet International Conference (RoEduNet), 2013 11th .

[4] Malik P, "Governing Big Data: Principles and practices", IBM Journal of Research and Development (Volume:57 , Issue: 3/4), May-July 2013.

[5] J. Yao, S. Chen, S. Nepal, D. Levy and J. Zic, "TrustStore: Making Amazon S3 Trustworthy with Services Composition," in Proceedings of the 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing (CCGRID '10), pp. 600-605, 2010.

[6] D. Zissis and D. Lekkas, "Addressing Cloud Computing Security Issues," Future Generation Computer Systems, vol. 28, no. 3, pp. 583-592, 2011.

[7] Jing Yu, Bin Xu, Yong Shi "The Domain Knowledge Driven Intelligent Data Auditing Model," Web Intelligence and Intelligent Agent Technology (WI-IAT), 2010 IEEE/WIC/ACM International Conference on (Volume:3), Aug. 31 2010-Sept. 3 2010, Page(s): 199 – 202.

[8] Cong Wang, Qian Wang, Kui Ren, Wenjing Lou "Privacy-Preserving Public Auditing for Data Storage Security in Cloud Computing," INFOCOM, 2010 Proceedings IEEE, 14-19 March 2010, Page(s): 1 - 9

[9] 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, no. 5, pp. 847 - 859, 2011.

[10] G. Ateniese, R.D. Pietro, L.V. Mancini and G. Tsudik, "Scalable and Efficient Provable Data Possession," in Proceedings of the 4th International Conference on Security and Privacy in Communication Netowrks (SecureComm '08), pp. 1-10, 2008.

[11] X. Zhang, L.T. Yang, C. Liu and J. Chen, "A Scalable Two-Phase Top-Down Specialization Approach for Data Anonymization using MapReduce on Cloud," IEEE Transactions on Parallel and Distributed Systems, In Press, 2013.

[12] C. Erway, A. Küpçü, C. Papamanthou and R. Tamassia, "Dynamic Provable Data Possession," in Proceedings of the 16th ACM Conference on Computer and Communications Security (CCS'09), pp. 213-222, 2009.

[13] S.E. Schmidt, "Security and Privacy in the AWS Cloud," Presentation on Amazon Summit Australia, 17 May 2012, Sydney, 2012,

[14] Y. He, S. Barman and J.F. Naughton, "Preventing Equivalence Attacks in Updated, Anonymized Data," in Proceedings of the 27th IEEE International Conference on Data Engineering (ICDE '11), pp. 529-540, 2011.

[15] A. Juels and J. B. S. Kaliski, "PORs: Proofs of Retrievability for Large Files," in Proceedings of the 14th ACM Conference on Computer and Communications Security (CCS '07), pp. 584-597, 2007.

[16] D. Boneh, H. Shacham and B. Lynn, "Short Signatures from the Weil Pairing," Journal of Cryptology, vol. 17, no. 4, pp. 297-319, 2004.