Published by :
http://www.ijert.org

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 8 Issue 04, April-2019

An Advanced Greedy Storage & Multi-Cloud Based Public Auditing Mechanism Integrating Data Risk Management

Nivedita Kasturi
Department of Computer Science and Engineering
PESIT Banglore South Campus
Banglore,Karntaka,India

Dr. S G Totad
School of Computer Science and Engineering
KLE Technological University
Hubli,Karnataka,India

***Abstract*—Data and Cloud are closely related to each other, as data is growing abundantly its necessary to manage storage efficiently and also allow multiple application s to access data effectively. There is a risk in managing and storing the data at cloud, we are trying to make cloud storage efficient by greedy storage and also trying to reduce the data risk. Paper is an preliminary work in this regard and literature survey is conducted to find the possible existing work in multi cloud based public auditing for data risk management.**

Keywords— Greedy storage, cloud computing, multi cloud, data risk, public auditing.

I. INTRODUCTION

Cloud computing is being intensively referred to as one of the most influential innovations in information technology in recent years. With resource virtualization, cloud can deliver computing resources and services in a pay-as-you-go mode, which is envisioned to become as convenient to use similar to daily-life utilities such as electricity, gas, water and telephone in the near future. These computing services can be categorized into Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS) and Software-as-a-Service (SaaS). Cloud Service providers offer users efficient and scalable data storage services with a much lower marginal cost than traditional approaches. Cloud storage provides customers with benefits, ranging from cost saving and simplified convenience, to mobility opportunities and scalable service. It is routine for users to leverage cloud storage services to share data with others in a group, as data sharing becomes a standard feature in most cloud storage offerings, including Drop box, iCloud and Google Drive. These great features attract more and more customers to utilize and storage their personal data to the cloud storage: according to the analysis report, the volume of data in cloud is expected to achieve 40 trillion gigabytes in 2020. Even though cloud storage system has been widely adopted, it fails to accommodate some important emerging needs such as the abilities of auditing integrity of cloud files by cloud clients and detecting duplicated files by cloud servers.

The integrity of data in cloud storage, however, is subject to skepticism and scrutiny, as data stored in the cloud can easily be lost or corrupted due to the inevitable hardware/ software failures and human errors [7]. To make this matter even worse, cloud service providers may be reluctant to inform users about these data errors in order to maintain the reputation of their services and avoid losing profits. Therefore, the integrity of cloud data should be verified before any data utilization, such as search or computation over cloud data. Data security/privacy is one of the major concerns in the adoption of cloud computing [8].

Compared to conventional systems, user's loses their direct control over their data. Recently, many mechanisms have been proposed to allow not only a data owner itself but also a public verifier to efficiently perform integrity checking without downloading the entire data from the cloud, which is referred to as public auditing[9]. From cloud users' perspective, it may also be called 'auditing-as-a-service'. To date, extensive research is carried out to address this problem [9], [10], [11]. In these mechanisms, data is divided into many small blocks, where each block is independently signed by the owner; and a random combination of all the blocks instead of the whole data is retrieved during integrity checking. A public verifier could be a data user (e.g., researcher) who would like to utilize the owner's data via the cloud or a third party auditor (TPA) who can provide expert integrity checking services. Existing public auditing mechanisms can actually be extended to verify shared data integrity. Considering the large size of the outsourced data files and the clients' constrained resource capabilities, the first problem is generalized as how can the client efficiently perform periodical integrity verifications even without the local copy of data files.

The second problem is greedy storage. In greedy storage we can think of two options such as secure deduplication and compression. The rapid adoption of cloud services is accompanied by increasing volumes of data stored at remote cloud servers. Among these remote stored files, most of them are duplicated: according to a recent survey by EMC [12], 75% of recent digital data is duplicated copies. This fact raises a technology namely deduplication, in which the cloud servers would like to deduplicate by keeping only a single copy for each file (or block) and make a link to the file (or block) for every client who owns or asks to store the same file (or block). Unfortunately, this action of deduplication would lead to a number of threats potentially affecting the storage system [12][13], for example, a server telling a client that it (i.e., the client) does not need to send the file reveals that some other client has the exact same file, which could be sensitive sometimes. These attacks originate from the reason that the proof that the client owns a given file (or block of

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 8 Issue 04, April-2019

data) is solely based on a static, short value (in most cases the hash of the file) [13]. Also, with normal storage data may occupy more storage space quickly compared to compressed files thus may result in filling the space completely soon. Thus, the second problem is generalized as after compression and deduplication, how can the cloud servers efficiently confirm that the client (with a certain degree assurance) owns the uploaded file (or block) before creating a link to this file (or block) for him/her.

The third problem is how to restore original data in case of anomalies or disaster. No matter how efficient and successful our integrity auditing system is, but end of the day user needs their original data back to them that too intact. Privacy and security while performing on data auditing can ensure proper resulted to the user but still to get another copy of data which is untouched is a matter of concern as there is no local copy with the user. Also servers to maintain confidentiality can't keep another copy in raw format. There has to be another copy on another server so that attack or malwares should not affect other server data. Also with each update operation on blocks the original file has should be regenerated to preserve proof of integrity. So the third problem generalized is how to retrieve original data back in case of anomalies.

We investigate the problem of data storage efficiency via deduplication and compression, attribute based encryption, multi-cloud data management, data and user privacy, data integrity public auditing, batch auditing and most importantly data recovery by users and service providers. Our goal is to design a cloud architecture that satisfies primary and important needs such as privacy, security, performance and risk management. We propose a system to resolve above problems and design an architecture which can set a new benchmark in the field of data integrity maintenance on cloud keeping security and privacy as key component.

II. LITERATURE SURVEY

[1] Jingwei Li; Jin Li; Dongqing Xie; Zhang Cai, "**Secure Auditing and Deduplicating Data in Cloud**," *IEEE Transactions On Computers*, vol. PP, no. 99, pp. 11, January 2015.

As the cloud computing technology develops during the last decade, outsourcing data to cloud service for storage becomes an attractive trend, which benefits in sparing efforts on heavy data maintenance and management. Nevertheless, since the outsourced cloud storage is not fully trustworthy, it raises security concerns on how to realize data deduplication in cloud while achieving integrity auditing. In this work, we study the problem of integrity auditing and secure deduplication on cloud data. Specifically, aiming at achieving both data integrity and deduplication in cloud, we propose two secure systems, namely SecCloud and SecCloud+. SecCloud introduces an auditing entity with a maintenance of a MapReduce cloud, which helps clients generate data tags before uploading as well as audit the integrity of data having been stored in cloud. Compared with previous work, the computation by user in SecCloud is greatly reduced during the file uploading and auditing phases. SecCloud+ is designed motivated by the fact that customers always want to encrypt their data before uploading, and enables integrity auditing and secure deduplication on encrypted data.

[2] Hao Jin; Hong Jiang; Ke Zhou, "**Dynamic and Public Auditing with Fair Arbitration for Cloud Data**," *IEEE Transactions On Cloud Computing*, vol. PP, no. 99, pp. 1, February 2016.

Cloud users no longer physically possess their data, so how to ensure the integrity of their outsourced data becomes a challenging task. Recently proposed schemes such as "provable data possession" and "proofs of retrievability" are designed to address this problem, but they are designed to audit static archive data and therefore lack of data dynamics support. Moreover, threat models in these schemes usually assume an honest data owner and focus on detecting a dishonest cloud service provider despite the fact that clients may also misbehave. This paper proposes a public auditing scheme with data dynamics support and fairness arbitration of potential disputes. In particular, we design an index switcher to eliminate the limitation of index usage in tag computation in current schemes and achieve efficient handling of data dynamics. To address the fairness problem so that no party can misbehave without being detected, we further extend existing threat models and adopt signature exchange idea to design fair arbitration protocols, so that any possible dispute can be fairly settled. The security analysis shows our scheme is provably secure, and the performance evaluation demonstrates the overhead of data dynamics and dispute arbitration are reasonable.

[3] Hui Tian; Yuxiang Chen; ChinChen Chang; Hong Jiang; Yongfeng Huang; Yonghong Chen; Jin Liu, "**Dynamic HashTable Based Public Auditing for Secure Cloud Storage**," *IEEE Transactions On Services Computing*, vol. PP, no. 99, pp. 11, December 2015.

Cloud storage is an increasingly popular application of cloud computing, which can provide ondemand outsourcing data services for both organizations and individuals. However, users may not fully trust the cloud service providers (CSPs) in that it is difficult to determine whether the CSPs meet their legal expectations for data security. Therefore, it is critical to develop efficient auditing techniques to strengthen data owners' trust and confidence in cloud storage. In this paper, we present a novel public auditing scheme for secure cloud storage based on dynamic hash table (DHT), which is a new twodimensional data structure located at a third parity auditor (TPA) to record the data property information for dynamic auditing. Differing from the existing works, the proposed scheme migrates the authorized information from the CSP to the TPA, and thereby significantly reduces the computational cost and communication overhead. Meanwhile, exploiting the structural advantages of the DHT, our scheme can also achieve higher updating efficiency than the stateoftheart schemes. In addition, we extend our scheme to support privacy preservation by combining the homomorphic authenticator based on the public key with the random masking generated by the TPA, and achieve batch auditing by employing the aggregate BLS signature technique. We formally prove the security of the proposed scheme, and evaluate the auditing performance by detailed experiments and comparisons with the existing ones. The results demonstrate that the proposed scheme can effectively achieve

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 8 Issue 04, April-2019

secure auditing for cloud storage, and outperforms the previous schemes in computation complexity, storage costs and communication overhead.

[4] Tao Jiang; Xiaofeng Chen; Jianfeng Ma, "**Public Integrity Auditing for Shared Dynamic Cloud Data with Group User Revocation**," *IEEE Transactions On Computers*, vol. PP, no. 99, pp. 11, January 2015.

The advent of the cloud computing makes storage outsourcing become a rising trend, which promotes the secure remote data auditing a hot topic that appeared in the research literature. Recently some research consider the problemof secure and efficient public data integrity auditing for shared dynamic data. However, these schemes are still not secure against the collusion of cloud storage server and revoked group users during user revocation in practical cloud storage system. In this paper, we figure out the collusion attack in the exiting scheme and provide an efficient public integrity auditing scheme with secure group user revocation based on vector commitment and verifierlocal revocation group signature. We design a concrete scheme based on the our scheme definition. Our scheme supports the public checking and efficient user revocation and also some nice properties, such as confidently, efficiency, countability and traceability of secure group user revocation. Finally, the security and experimental analysis show that compared with its relevant schemes our scheme is also secure and efficient.

[5] Jian Liu; Kun Huang; Hong Rong; Huimei Wang; Ming Xian, "**Privacy Preserving Public Auditing for Regenerating CodeBased Cloud Storage**", *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 7, pp. 1513 1528, July 2015.

To protect outsourced data in cloud storage against corruptions, adding fault tolerance to cloud storage together with data integrity checking and failure reparation becomes critical. Recently, regenerating codes have gained popularity due to their lower repair bandwidth while providing fault tolerance. Existing remote checking methods for regenerating coded data only provide private auditing, requiring data owners to always stay online and handle auditing, as well as repairing, which is sometimes impractical. In this paper, we propose a public auditing scheme for the regenerating codebased cloud storage. To solve the regeneration problem of failed authenticators in the absence of data owners, we introduce a proxy, which is privileged to regenerate the authenticators, into the traditional public auditing system model. Moreover, we design a novel public verifiable authenticator, which is generated by a couple of keys and can be regenerated using partial keys. Thus, our scheme can completely release data owners from online burden. In addition, we randomize the encode coefficients with a pseudorandom function to preserve data privacy. Extensive security analysis shows that our scheme is provable secure under random oracle model and experimental evaluation indicates that our scheme is highly efficient and can be feasibly integrated into the regenerating codebased cloud storage.

[6] Ayad F. Barsoum; M. Anwar Hasan, "**Provable Multicopy Dynamic Data Possession in Cloud Computing Systems**," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 3, pp. 485 497, March 2015.

Increasingly more and more organizations are opting for outsourcing data to remote cloud service providers (CSPs). Customers can rent the CSPs storage infrastructure to store and retrieve almost unlimited amount of data by paying fees metered in gigabyte/month. For an increased level of scalability, availability, and durability, some customers may want their data to be replicated on multiple servers across multiple data centers. The more copies the CSP is asked to store, the more fees the customers are charged. Therefore, customers need to have a strong guarantee that the CSP is storing all data copies that are agreed upon in the service contract, and all these copies are consistent with the most recent modifications issued by the customers. In this paper, we propose a mapbased provable multicopy dynamic data possession (MBPMDDP) scheme that has the following features: 1) it provides an evidence to the customers that the CSP is not cheating by storing fewer copies; 2) it supports outsourcing of dynamic data, i.e., it supports blocklevel operations, such as block modification, insertion, deletion, and append; and 3) it allows authorized users to seamlessly access the file copies stored by the CSP. We give a comparative analysis of the proposed MBPMDDP scheme with a reference model obtained by extending existing provable possession of dynamic singlecopy schemes. The theoretical analysis is validated through experimental results on a commercial cloud platform. In addition, we show the security against colluding servers, and discuss how to identify corrupted copies by slightly modifying the proposed scheme.

III. OBJECTIVES

Cloud servers possess several security and integrity threats on user data and that is the reason that public auditing is studied in the past numerously but still efficiency and security is an area of research. We need to take care of various parameters such as privacy, availability, security, scalability, reliability and efficiency in the system. Key objectives of our research work are: to maintain privacy of data so that it is not disclosed to unauthorized users; to derive a mechanism which could deliver correct data to the user whenever it is requested; to maintain security of data using encryption techniques and maintain integrity of data by providing data recovery options to data owners and CSPs so that in case of integrity check fails on data, there would be option to recover original data back; to make system reliable in terms of privacy, integrity, security, transparency, availability and efficiency. to remove redundancy of data on server and enhance efficiency of system by enabling quick operation on block data and to combine compression technique to attain goal of greedy storage.

IV. PROBLEM STATEMENT

Greedy data storage is a demand of today for every IaaS cloud provider. Compression techniques are used widely to maintain efficiency in storage but along with it deduplication is not studied in past. We find this combination can make big difference in the current trend of data storage on cloud servers. With numerous public auditing mechanisms proposed in the past, checking for data integrity on server, we have very limited or no study on data recovery options in case of anomalies. It is obviously a no matter of satisfaction

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 8 Issue 04, April-2019

that data owner can just get to know that their data is manipulated on server, but additionally there should be facility for them to make sure genuine data can be recovered. Also it could be of much help for CSPs if they itself also can do batch audit to find anomalies in data and do data recovery. It would give them ace over other untrusted servers. To the best of our knowledge this problem has not been identified, therefore, we propose to investigate it and provide solution for it using proposed methodology while accomplishing above mentioned objectives.

V. METHODOLOGY

Our methodology to attain a system which is efficient in storage maintenance, secure in terms of data security and privacy, transparent in terms of data integrity validation and reliable in terms of data risk management, we intend to design an innovative architecture for cloud environment where storage efficiency can be maintained through combination of compression and deduplication techniques. Security of data and user privacy can be maintained by CipherText Policy-Attribute Based Encryption (CP-ABE) system so that data can be kept in encrypted form on the server and can be accessed via Role Based Access Control (RBAC). For a light-weight audit mechanism, hashing of data could be a good choice for audit as a little change in data can alter the entire hash. We would make block level data auditing by public auditing mechanism on encrypted cloud data saved on multiple cloud severs. But our main concentration will on data audit and recovery options to CSUs and batch audit facility for CSPs so that faults can be recognized and faulty data can be replaced with original data. Enhancements on existing techniques would make system more efficient and reliable.

VI. POSSIBLE OUTCOME

Outcome of our research would be an efficient and secure cloud based system for users where not only public auditing but data recovery will also be provided as data risk management option to the data owners. Data Compression and Deduplication technique would be used to minimize the redundancy of data on server and efficient storage system. We allow public auditing on cloud data upon data owner request, keeping owner and file information confidential due to privacy concerns. We ensure complete safety of user data using multi-cloud storage and replication, so that in case of failed integrity result, data owners can recover their data back. Also we allow CSPs to make a batch audit on data so that faults can be found and original data can be recovered. This system will be executed on web in real time and results will be demonstrated in the form of graphs and charts.

REFERENCES

[1] Jingwei Li; Jin Li; Dongqing Xie; Zhang Cai, “Secure Auditing and Deduplicating Data in Cloud,” IEEE Transactions On Computers, vol. PP, no. 99, pp. 11, January 2015.

[2] Hao Jin; Hong Jiang; Ke Zhou, “Dynamic and Public Auditing with Fair Arbitration for Cloud Data,” IEEE Transactions On Cloud Computing, vol. PP, no. 99, pp. 1, February 2016.

[3] Hui Tian; Yuxiang Chen; ChinChen Chang; Hong Jiang; Yongfeng Huang; Yonghong Chen; Jin Liu, “Dynamic HashTable Based Public Auditing for Secure Cloud Storage,” IEEE Transactions On Services Computing, vol. PP, no. 99, pp. 11, December 2015.

[4] Tao Jiang; Xiaofeng Chen; Jianfeng Ma, “Public Integrity Auditing for Shared Dynamic Cloud Data with Group User Revocation,” IEEE Transactions On Computers, vol. PP, no. 99, pp. 11, January 2015.

[5] Jian Liu; Kun Huang; Hong Rong; Huimei Wang; Ming Xian, “Privacy Preserving Public Auditing for Regenerating CodeBased Cloud Storage", IEEE Transactions on Information Forensics and Security, vol. 10, no. 7, pp. 1513 1528, July 2015.

[6] Ayad F. Barsoum; M. Anwar Hasan, “Provable Multicopy Dynamic Data Possession in Cloud Computing Systems,” IEEE Transactions on Information Forensics and Security, vol. 10, no. 3, pp. 485 497, March 2015.

[7] 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,” Communication of the ACM, vol. 53, no. 4, pp. 50–58, 2010.

[8] D. Zissis and D. Lekkas, “Addressing Cloud Computing Security Issues,” Future Gen. Comput. Syst., vol. 28, no. 3, pp. 583-592, Mar. 2011.

[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 Trans. Parallel Distrib. Syst., vol. 22, no. 5, pp. 847-859, May 2011.

[10] G. Ateniese, R. Burns, R. Curtmola, J. Herring, O. Khan, L. Kissner, Z. Peterson, and D. Song, “Remote Data Checking Using Provable Data Possession,” ACM Trans. Inf. Syst. Security, vol. 14, no. 1, May 2011, Article 12.

[11] Y. Zhu, H. Hu, G.-J. Ahn, and M. Yu, “Cooperative Provable Data Possession for Integrity Verification in Multi-Cloud Storage,” IEEE Trans. Parallel Distrib. Syst., vol. 23, no. 12, pp. 2231-2244, Dec. 2012.

[12] J. Yuan and S. Yu, “Secure and constant cost public cloud storage auditing with deduplication,” in IEEE Conference on Communications and Network Security (CNS), 2013, pp. 145–153.

[13] S. Halevi, D. Harnik, B. Pinkas, and A. Shulman-Peleg, “Proofs of ownership in remote storage systems,” in Proceedings of the 18th ACM Conference on Computer and Communications Security. ACM, 2011, pp. 491–500.