A REVIEW ON RELIABLE DEDUPLICATION AND PUBLIC AUDITING IN CLOUD STORAGE

TIPPANAVENI NAGARAJU
M.Tech, CSE, Holy Mary Institute of Technology and Science, Bogaram (V), Kessara (M), Medchal-501301, Telangana.

BIRRU DEVENDER
Associate Professor, Holy Mary Institute of Technology and Science, Bogaram (V), Kessara (M), Medchal-501301, Telangana.

ABSTRACT
As the cloud computing becomes more popular for the virtually storing data because it provides the big storage space as well as provides the portability and mobility functions. Whenever we are moving around the world we can access the cloud data by setting up internet connection in between our device and cloud server. Data de-duplication is a technique for eliminating duplicate copies of data, and has been widely used in cloud storage to reduce storage space and upload bandwidth. However, there is only one copy for each file stored in cloud even if such a file is owned by a huge number of users. As a result, de-duplication system improves storage utilization while reducing reliability. Furthermore, the challenge of privacy for sensitive data also arises when they are outsourced by users to cloud. Aiming to address the above security challenges, this paper makes the first attempt to formalize the notion of distributed reliable deduplication system. In this paper new distributed deduplication systems with higher reliability in which the data chunks are distributed across multiple cloud servers is being praposed.

1. INTRODUCTION
Hiding platform and implementation details unlimited virtualized resources provided to the users as a service is a cloud computing. Presently cloud service provided to the users offered high available storage and massively parallel computing of resources at relatively low costs. But the question is about the cloud users with different privileges store data on cloud is a most challenge issue in managing cloud data storage system. Now a day's cloud becomes an attractive trend toward the people. Cloud provides the space for user to store data in virtualized pool storage which may be accessible wherever you want with the help of internet capable device. This portability as well as the scalable services provided by the cloud affect that peoples preferred to store their personal data on cloud storage. Cloud also beneficial for cost saving .It provides the resource sharing feature. There are some emerging needs which fail by cloud such as auditing integrity of files by client and detecting the duplicate files of data by cloud server. Cloud server minimizes the heavy load of storing and maintaining data stored on cloud. As compares to traditional storage cloud storage store the data at uncertain storage domain without any control of client. This grows the security concerns of the client data. Cloud storage is susceptible to security threats from both outside and inside of the cloud [1].Some clouds hides the loss of data from client in order to maintain their reputation. Another one more serious problem is that in order to save the space as well as money some cloud discards the rarely used data of ordinary clients. Here the issue arises how to recover these problems of integrity verification of client data. Second problem is that secure deduplication. Most of the peoples uses the cloud and upload the data on cloud and this

activity gives results of duplicates of the same data/files on server which exceeds the storage of cloud. Then here needs the cloud to store only deduplicate data that is keep only the single copy of data on cloud. If any user request to access same file which already on server giving him/her the reference of that file and avoid duplication of same file. This action of deduplicating files may lead to security threats [3][2].Some potential security threats are arises while client upload the file towards the cloud that cloud tell that file is already exist on cloud these files are sensitive sometimes . From these types of attack client should solely get that file which is permitted to him/her [3].Second problem is that cloud server efficiently confirm that client owns the uploaded file by creating link to that file. Here aim is that to auditing integrity and dedulication .This can be implement by using SecCloud and SecCloud. SecCloud used for secure cloud. It fixes the issue of previous work of heavy load of computational time. SecCloud generates the tags which we upload to the cloud. It also prevents the leakage of information side channel. SecCloud+ it uses the encryption so data becomes more confidential on cloud. SecCloud uses convergent key encryption in order to defend to the attacker. It prevents dictionary attacks [4].

MOTIVATION:

Integrity Auditing: It verifies the accuracy or the correctness of data uses verification method that is public verification and stateless verification. In public verification it allows anyone not only the client for verification. Stateless verification eliminates the state information maintenance.

Secure Deduplication: The process which used for keep the single copy of file in order to save space on cloud duplicate file not allowed to store on cloud by performing duplicate check. Cost Effective: It is does not take any extra cost to download as well as uploading file operation. Also it does not change the upload download mechanism.

Related Work:

There are many strategies for deduplication is proposed, Server side deduplication, client side deduplication and file level, block level. These are the stratagies where deduplication can be done. Bellare et al introduced message locked encryption as well as its application.

Li shown the Key Management issue in block-level deduplication by sending these keys over multiple servers after encrypting the files. Bellare et al showed how to protect data confidentiality.

The initial problem is integrity auditing. The cloud server minimizes the huge load from users to storing and maintaining those bulk amounts of data. The difference is that in traditional storage and cloud storage as shown below. Cloud refers to use of internet to transfers the file from client to uncertain space which is not under the control of user. This result in increasing the client integrity of their data.

The second problem is secure deduplication. In the recent days many users prefer cloud storage to store their data on cloud. This result in large amount of duplicate data on

cloud. This becomes the issue of storage space as well as to maintain it. By the EMC survey there is 75% of the digital data on cloud are duplicated file. Actually, this action of deduplication would cause to• many threats potentially affecting the storage system, for example if there is already exist file and client uploading same file that time server suggest to client the file is already existed do not need to upload the same file. This file may be sensitive some times.

DISADVANTAGE

Data reliability is actually a very critical issue in a• deduplication storage system because there is only one copy for each file stored in the server shared by all the owners. Most of the previous deduplication systems have only been• considered in a single-server setting. The traditional deduplication methods cannot be directly• extended and applied in distributed and multi-server systems.

PROPOSED SYSTEM:

This paper presents a system that encounters the below mentioned two problems in cloud. The first problem is integrity auditing. The cloud server is able to relieve clients from the heavy burden of storage management and maintenance. The most difference of cloud storage from traditional in-house storage is that the data is transferred via Internet and stored in an uncertain domain, not under control of the clients at all, which inevitably raises clients great concerns on the integrity of their data. These concerns originate from the fact that the cloud storage is susceptible to security threats from both outside and inside of the cloud [1], and the uncontrolled cloud servers may passively hide some data loss incidents from the clients to maintain their reputation. What is more serious is that for saving money and space, the cloud servers might even actively and deliberately discard rarely accessed data files belonging to an ordinary client. 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 secure deduplication. 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 [2], 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 [3][2], 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 data) is solely based on a static, short value (in most cases the hash of the file) [3]. Thus, the second problem is generalized as 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.

ADVANTAGES

Distinguishing feature of our proposal is that data integrity, including tag consistency, can be achieved.

No existing work on secure deduplication can properly address the reliability and tag consistency problem in distributed storage systems.

The proposed constructions support both file-level and block-level deduplications.

Security analysis demonstrates that the proposed deduplication systems are secure in terms of the definitions specified in the proposed security model. In more details, confidentiality, reliability and integrity can be achieved in proposed system. Two kinds of collusion attacks are considered in our solutions. These are the collusion attack on the data and the collusion attack against servers. In particular, the data remains secure even if the adversary controls a limited number of storage servers.

This deduplication system has been implemented using the Ramp secret sharing scheme that enables high reliability and confidentiality levels. The evaluation results demonstrate that the new proposed constructions are efficient and the redundancies are optimized and comparable with the other storage system supporting the same level of reliability.

File Confidentiality: The design goal of file confidentiality requires preventing the cloud servers from accessing the content of files. Specially, we require that the goal of file confidentiality needs to be resistant to "dictionary attack". That is, even the adversaries have pre-knowledge of the "dictionary" which includes all the possible files; they still cannot recover the target file

Secure Deduplication: Deduplication is a technique where the server stores only a single copy of each file, regardless of how many clients asked to store that file, such that the disk space of cloud servers as well as network bandwidth are saved. However, trivial client side deduplication leads to the leakage of side channel information. For example, a server telling a client that it need not send the file reveals that some other client has the exact same file, which could be sensitive information in some case.

Encryption & Decryption: Encryption and decryption provides data confidentiality in deduplication. A user (or data owner) derives a convergent key from the data content and encrypts the data copy with the convergent key. In addition, the user derives a tag for the data copy, such that the tag will be used to detect duplicates. Here, we assume that the tag correctness property holds, i.e., if two data copies are the same, then their tags are the same. Formally, a convergent encryption scheme can be defined with four primitive functions:

Integrity Auditing: The first design goal of this work is to provide the capability of verifying correctness of the remotely stored data. The integrity verification further requires two features:

1. Public verification, which allows anyone, not just the clients originally stored the file, to perform verification;

2. Stateless verification, which is able to eliminate the need for state information maintenance at the verifier side between the actions of auditing and data storage.

CONCLUSION

It can be concluding that the distributed deduplication systems to improve the reliability of data while achieving the confidentiality of the users' outsourced data without an encryption mechanism. Four constructions were proposed to support file-level and fine-grained block-level data deduplication. The security of tag consistency and integrity were achieved. This deduplication system has been implemented using the Ramp secret sharing scheme and demonstrated that it incurs small encoding/decoding overhead compared to the network transmission overhead in regular upload/download operations.

REFERENCES:

[1] Jingwei Li, Jin Li, Dongqing Xie and Zhang Cai, "Secure Auditing and Deduplicating Data in Cloud", IEEE TRANSACTIONS ON COMPUTERS VOL: PP NO: 99 YEAR 2015.

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

[3] 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.

[4] S. Keelveedhi, M. Bellare, and T. Ristenpart, "Dupless: Serveraided encryption for deduplicated storage," in Proceedings of the 22Nd USENIX Conference on Security, ser. SEC'13. Washington, D.C.: USENIX Association, 2013, pp. 179–194.

[5] J. Li, X. Chen, M. Li, J. Li, P. Lee, and W. Lou, "Secure deduplication with efficient and reliable convergent key management," IEEE Transactions on Parallel and Distributed Systems, vol. 25, no. 6, pp. 1615–1625, June 2014.

[6] Y. Zhu, H. Hu, G.-J. Ahn, and M. Yu, "Cooperative provable data possession for integrity verification in multicloud storage," IEEE Transactions on Parallel and Distributed Systems, vol. 23, no. 12, pp. 2231–2244, 2012.

[7] W. K. Ng, Y. Wen, and H. Zhu, "Private data deduplication protocols in cloud storage," in Proceedings of the 27th Annual ACM Symposium on Applied Computing, ser. SAC '12. New York, NY, USA: ACM, 2012, pp. 441–446.

[8] J. Li, X. Tan, X. Chen, and D. Wong, "An efficient proof of retrievability with public auditing in cloud computing," in 5th International Conference on Intelligent Networking and Collaborative Systems (INCoS), 2013, pp. 93–98.

[9] 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: Advances in Cryptology, ser. ASIACRYPT '08. Springer Berlin Heidelberg, 2008, pp.90–107.

[10] Q. Wang, C. Wang, J. Li, K. Ren, and W. Lou, "Enabling public verifiability and data dynamics for storage security in cloud computing," in Computer Security – ESORICS 2009, M. Backes and P. Ning, Eds.,vol. 5789. Springer Berlin Heidelberg, 2009, pp. 355–370.