Secure Authorized De-duplication Using Hybrid Cloud

Shilpa Giri

Abstract— **Data de-duplication is most widely used for eliminating data redundancy. Instead of saving the files having the same data, it stores single file thus reduces the memory space usage. In many organizations, their database contains the data de-duplication. For example, the different user saves the same file having the same data at different, different spaces. De-duplication eliminates these all extra copies by saving just one copy of the data and replacing the other all copies with pointers that lead back to the original copy. It is one of the compression techniques so that we can improve the bandwidth efficiency and storage utilization. Cloud contains no. of resources having no. of applications, software, and storage, so data redundancy may be there. So could computing uses Data De-duplication technique. It reduces the data management and storage problem. Data de-duplication protects the confidentiality of sensitive data. Data de-duplication uses convergent encryption technique to encrypt the data before uploading in the cloud. Companies frequently use de-duplication in backup and disaster recovery applications. Here we trying to use authorized de-duplication check, combine with convergent encryption for providing security to sensitive data using hybrid cloud computing.**

Index Terms— **Authorized duplicate check, Convergent encryption, De-duplication, Hybrid cloud.**

I. Introduction

In today's world everyone uses the Internet most widely, so cloud computing is the most important technique, used by a Communication network like internet. So the business storage is done at very low cost. We can use cloud computing to store different types of data from many different fields like government, enterprise or anyone can store their personal data also. Cloud computing is nothing but sharing of resources so, without any background implementation details, users can share as well as access the different resources. As we know the most important issues of clod computing are related to memory management and the security of sensitive data. So the main drawback of cloud storage is data duplication which is increasing day by day. To reduce the memory management problem and to improve the storage space data de-duplication is an important technique that should be used by cloud computing. Recently all storage systems use the data de-duplication technique widely, so it is becoming most popular in many organizations. Data compression also uses the same technique that is data de-duplication for reduce the data redundancy and going to store only single copy of that file. Data de-duplication is done by two ways one is File level and another is Block level. In File level approach we can eliminate the identical files from the storage space and in block level approach we can delete some amount of data i.e the block of data from the files which are not similar Data de-duplication decrease the storage needs up to 90-95% backup application and for standard file system it is 68% But the main problem is security of data and privacy of that data form hackers. To secure the data from attackers users uses the encryption and decryption technique. They encrypt the data before uploading the data in the cloud. They use the private key to perform encryption and decryption on the cloud. Such that to perform the encryption user first generates the convergent key using that key it encrypt the data. To give the proof that user wants the same file proof of ownership protocol is used which prevents the unauthorized access when de-duplication found. After giving the proof server provides the pointer to that user showing the same file that's why that user no need to upload the same file again n again If user wants any file then it just download the file form the cloud and using that convergent key he just decrypt the file.

II. Related Work

Data de-duplication is a technique for eliminating duplicate copies of data, and has been broadly used in cloud storage to reduce upload bandwidth and storage space. Predicting as it is, a coming up challenge to perform secure de-duplication in cloud storage. Although convergent encryption has been widely adopted for secure de-duplication, and it is a critical issue of making convergent encryption practical for reliably and efficiently manage a huge number of convergent keys. Our paper makes the first and best attempt to formally address the problem of achieving efficient and reliable key management in secure de-duplication. We firstly introduce a standard methodology in which each and every user has their own separate master kay for encrypting the convergent keys and outsourcing them to the cloud storage. However, we had a baseline or standard key management scheme which creates a huge number of keys as the users growing rapidly and requires users to enthusiastically protect the master keys. Storage efficiency functions such as compression and de-duplication afford storage providers' better utilization of their storage backend and the ability to serve more customers with the same infrastructure. Data de-duplication is the process by which a storage provider only stores a single copy of file owned by several of its users. There are four different de-duplication strategies, depending on whether de-duplication happens at the client side (i.e. before the upload happens) or at the server

Manuscript received Dec, 2014.

Shilpa Giri, Department of Computer Engineering, Savitribai Phule Pune University/ G.H Raisoni College of Engineering & Management, Wagholi Pune Pune INDIA.

side, and whether de-duplication happens at a block level or at file level. De-duplication is most reinforcing when it is triggered at the client side, as it also saves upload bandwidth. For these reasons, de-duplication is a critical enabler for a number of popular and successful storage services that offer a cheap, remote storage to the broad public by performing client-side de-duplication, thus saving both the storage costs and network bandwidth. Well the data de-duplication is disputably one of the main reasons why the prices for cloud storage and cloud backup services have dropped so sharply. Unfortunately, de-duplication loses its effectiveness in conjunction with end-to-end encryption. End-to-end encryption in a storage system is the process by which data is encrypted at its source prior to ingress into the storage system. It is becoming an increasingly prominent requirement due to both the number of security incidents linked to leakage of unencrypted data and the tightening of sector-specific laws and regulations. Clearly, if semantically secure encryption is used, file de-duplication is impossible, as no one apart from the owner of the decryption key can decide whether two cipher texts correspond to the same plaintext.

III. Proposed Work

In previous de-duplication systems cannot support differential authorization of duplicate check, which is having importance in many of the applications. In such an authorized de-duplication system, each user is issued a set of privileges during system initialization.

The overview of the cloud de-duplication is as follow:

A. *POST-PROCESS DE-DUPLICATION*

Post-process de-duplication is frequently used in most of the backup applications, and the virtual tape libraries, where the reduction of backup time is always required. With that this method turns out to be absolutely ineffective in case of rapid data recovery is needed to user because when a client addresses the storage, our system can be busy with de-duplication of the next portion of data. Post-process de-duplication used to check the data redundancy so for that new data is first stored on the storage device and then the process is checking for the data duplication on that storage device. Hash calculations is used to check the data de-duplication so here is we are not using the hash calculations and data redundancy is checking before data stored on that device so it improves the performance here policy based implementation used so that user has the ability to defer optimization on "active" files, or to process files based on type and location. One problem is that we unnecessarily store the data and storage device becomes nearly full.

B. *IN-LINE DE-DUPLICATION*

Inline de-duplication is the most beneficial and the economic method for doing the de-duplication. The method significantly reduces our raw disk capacity is required in the system since the disk gets full, or not yet de-duplicated data set would be never written to the disk. Within line de-duplication hash value is calculated on that storage device as data is submitted on that device in real time. It is using block level if that device highlights some block of data. Then it does not store the new In-line de-duplication is more beneficial than post process de-duplication as it consumes the less space on storage device. On the non-positive side, it is often said that because lookups and hash calculations takes so long, that the data consumption can be slower thus reducing the backup magnitude of the device. Nevertheless, certain dealers with in-line de-duplication have verified tools with likely same results to their post-process de-duplication counterparts. Post-process and in-line de-duplication methods are often greatly questioned.

SOURCE VERSUS TARGET DE-DUPLICATION

Another way to think about data de-duplication is by where it occurs. When the de-duplication occurs on the nearby to the data creation, it is called as "source de-duplication". When it occurs on the nearby the data is saved, it is often called "target de-duplication". Source de-duplication assures that data on the data source is de-duplicated and it generally takes place directly within the file system. The file system will timely scan the new files making hashes and match them to hashes of remaining files. A Hybrid Cloud Approach for Secure Authorized de-duplication done when data files with matching hashes are located then the file copy is replaced with a new file which points to the old file. However, the repeated files are considered to be distinct entities and if one of the repeated files is later altered, and then using a method called Copy on-write. Then the replica of that file or altered block is created. This de-duplication process is clear to each and every user and to the backup applications. To back up a de-duplicated file system, frequently cause repetition which resulting in the backups being larger than the source data. Target de-duplication is the procedure of eliminating duplicates of data in the secondary store. Generally this will be a backup store such as a data warehouse or a virtual tape store. One of the most widely used technique of data de-duplication operation works by equating portions of data to identify duplicates. For this purpose, each portion of data is set proof of identity, designed by the software, normally consuming cryptographic hash functions. In many operations, the presumption is made that if the credentials is duplicate, the data is duplicate, although this is not correct in all circumstances due to the pigeonhole principle; other operations do not undertake that two blocks of data with the same identifier are equal, but in reality it verified that data with the same identification is matching. If the software either concludes that a specified proof of identity at present be existent in the de-duplication namespace or actually confirms the identity of the two blocks of data, depending on the operation, then it will substitute that duplicate portion with a link. Once the information has been de-duplicated, upon read back of the file, wherever a linking is found, the method simply substitutes that connection link with the referenced data portion. The de-duplication process is proposed to be apparent to consumers and applications.

IV. PROPOSED METHOD

In the proposed system we are achieving the data de-duplication by providing the proof of data by the data owner. This kind of proof is used at the time of uploading of the file in the system. Each file uploaded to the cloud storage is also restricted by a set of rights to identify which type of users is permissible to accomplish the duplicate check and which type of user access the files. Such that, the user needs to take this file and his own privileges as his inputs before submitting his duplicate check request for some file. Whereas the user is able to find a duplicate for this file if and only if there is a copy of this file and a matched privilege stored in cloud.

Proposed Method Block diagram:

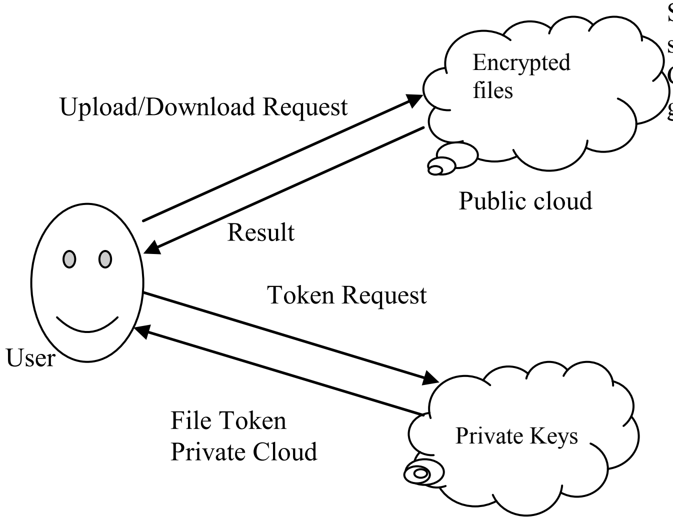

Fig.1 block diagram of proposed method

1. User Module is the module in which the user are having authentication and security to access the detail which is offered in the ontology system before opening or examining the details users should have the account in that otherwise they should register first.

2. Secure De-duplication System to support authorized de-duplication, the tag of file 'F' will be determined by the file 'F' and the privilege .To show the difference with traditional notation tag, we call it file token. To support authorized access a secret key "kp" will be bounded with a privilege p to generate a file token. Let $'F:p=TagGen(F,kp) denote the token of 'F' that is only allowed to access by user with privilege 'p'. In other words, the token $'F:p could only be computed by the users with privilege 'p'. As a result if file is uploaded by a user with duplicate token $'F:p then a duplicate check sent from another user will be successful if and only if he also has the file 'F' and privilege 'p'. Such that a token generation function could be easily implemented as H(F,kp), where H(_) denotes the cryptographic function.

3. Security of Duplicate Check Token as we had consider several types of privacy so far and there is need to protect, that is, unforged ability of duplicate check token: There are two types of adversaries that is external adversary and internal adversary. As we know, the external adversary can be viewed as internal adversary without any privilege 'p' on any file F where p doesn't match p'. Moreover it also requires that if the adversary does not make a request of token with its own privilege from private cloud server, it cannot counterfeit and output a valid duplicate token p on any F that has been queried.

4. Send Key once the key request is received, the sender can send the key or on the other hand the user can decline it. With this key and request id which was generated at the time of sending key request the receiver can decrypt the message.
We implement a prototype of the proposed authorized de-duplication system, in which we model three entities as separate C++ programs. A Private Server program is used to model the private cloud which manages the private keys and handles the file token computation. A Client program is used to model the data users to carry out the file upload process. A Storage Server program is used to model the S-CSP which stores and de-duplicate files. Our implementation of the Client provides the following function calls to support token generation and de-duplication along the file upload process.

- FileTag (File) – the process computes SHA-1 hash of the File as the File Tag;
- TokenReq (Tag, UserID) – the process requests the Private Server for File Token generation with the User ID and the File Tag;
- DupCheckReq (Token) – the process requests the Storage Server for the Duplicate Check of the File by sending the file token received from the private server;
- ShareTokenReq (Tag, {Priv.}) – the process requests the Private Server to generate the Share File Token with Target Sharing Privilege Set and the File Tag;
- FileEncrypt (File) – the process encrypts the File with Convergent Encryption using 256-bit of the AES algorithm in cipher block chaining (CBC) mode, where the convergent key is from SHA-256 Hashing of the file;
- FileUploadReq(FileID, File, Token) – the process uploads the File Data to the Storage Server if the file is Unique and the updates of the File Token is stored. Our implementation of the Private Server includes corresponding request handlers for the token generation and maintains a key storage with Hash Map.
- TokenGen(Tag, UserID) – the process loads the associated privilege keys of the user and generate token with HMAC-SHA-1.

V. Conclusion

Cloud computing has reached a maturity that leads it into a generative phase. This means that most of the main issues with cloud computing have been addressed to a degree that clouds have become interesting for full commercial exploitation. This however does not mean that all the problems listed above have actually been solved, only that the according risks can be endured to a certain degree. The Cloud computing is therefore still as much a research topic, as it is a market offering. For the better confidentiality and security in cloud computing we have proposed a new de-duplication constructions supporting authorized duplicate check in hybrid cloud architecture, in which the duplicate-check tokens of files are generated by the private cloud server with private keys. The proposed system includes proof of data owner so it will help to implement better security issues in cloud computing.

REFERENCES

[1] M. Bellare, S. Keelveedhi, and T. Ristenpart. Dupless: Serveraided encryption for deduplicated storage. In USENIX Security Symposium, 2013.

[2] P. Anderson and L. Zhang. Fast and secure laptop backups with encrypted de-duplication. In Proc. of USENIX LISA, 2010.

[3] J. Li, X. Chen, M. Li, J. Li, P. Lee, andW. Lou. Secure de-duplication with efficient and reliable convergent key management. In IEEE Transactions on Parallel and Distributed Systems, 2013.

[4] S. Halevi, D. Harnik, B. Pinkas, and A. Shulman-Peleg. Proofs of ownership in remote storage systems. In Y. Chen, G. Danezis, and V. Shmatikov, editors, ACM Conference on Computer and Communications Security, pages 491–500. ACM, 2011.

[5] J. Li, X. Chen, M. Li, J. Li, P. Lee, andW. Lou. Secure de-duplication with efficient and reliable convergent key management. In IEEE Transactions on Parallel and Distributed Systems, 2013.

[6] C. Ng and P. Lee. Revdedup: A reverse de-duplication storage system optimized for reads to latest backups. In Proc. of APSYS, Apr 2013.

[7] C.-K Huang, L.-F Chien, and Y.-J Oyang, “Relevant Term Suggestion in Interactive Web Search Based on Contextual Information in Query Session Logs,” J. Am. Soc. For Information science and Technology, vol. 54, no. 7, pp. 638-649, 2003.

[8] S. Bugiel, S. Nurnberger, A. Sadeghi, and T. Schneider. Twin clouds: An architecture for secure cloud computing. In Workshop on Cryptography and Security in Clouds (WCSC 2011), 2011.

[9] W. K. Ng, Y. Wen, and H. Zhu. Private data de-duplication protocols in cloud storage. In S. Ossowski and P. Lecca, editors, Proceedings of the 27th Annual ACM Symposium on Applied Computing, pages 441–446. ACM, 2012.