AN EFFICIENT, SECURE DEDUPLICATION DATA STORING IN CLOUD STORAGE ENVIRONMENT

Amaresh[1], Manjunath G S[2]

[1]*Student, Computer science and Engineering, Acharya Institute of Technology, Bengaluru, India*
[2]*Assistant professor, Computer science and Engineering, Acharya Institute of Technology, Bengaluru, India*

Abstract

Most of the IT industry heading towards the cloud –based storage services .cloud storage is witnessed to weak in security and privacy for public cloud environments .To tackle these security challenges ,we propose a new deduplication at client side for secure data storing and data sharing among the cloud users through public cloud .our proposal is mainly concerned on the owner of the file is encrypted the data that he intended to upload to the cloud by applying per data key, so data access controlled by data owner and log file which contains retrieval rights of the cloud users ,an authorised user can decipher an encrypted file with his private key.

Keywords: *convergent encryption, cloud storage, deduplication, privacy, Merkle-tree*

----------***----------

1. INTRODUCTION

Recent days, ever increasing of digital information insist on having new storage and network capacity demands along with the cost effectiveness of storage and network traffic rate.

As such, use of remote storage is got attention, namely cloud based services because it gives cost effective architecture and is pay as you go model. Client side deduplication is a scheme to reduce the consumption of storage space along with the bandwidth used to transfer duplicate file, this already applied some service providers namely Memopal, DropBox ([2][3]).

In spite of some noteworthy advantage's in saving resources, client side deduplication pose many security problems due to multi-owner data possession challenges [1], most of the attackers attack only for bandwidth consumption as well as privacy. Lately, weaken these problems, a lot work has been done under different security models [1] [3] [4] [8] [6] .these security models focus on POW (power of ownership) and onto the server to check the test user ownership, based on content and short Hash value. These all models build trust most of the requirements, say light weight verification.

Our paper introduces new cryptographic approach for an efficient, secure POW (power of ownership), on the basis of combined effect of convergent encryption [5] and the Merkle-tree [7], for climbing the level of data storage security in the cloud environments, giving option to share cloud users. Our spotlight contains derive the unique identifier of outsourced data in using markle base tree over enciphered data. Other side identifier serve to make sure that only copy of data is stored along with dynamic sharing of data with controlled access to users.

2. BACKGROUNDS

The proof of ownership (PoW) is found by Halevi [8].It is request-response protocol make sure that requesting object is data owner of the file (F) outsourced, on the basis of short value .whenever the owner wants to outsource a file, he must calculate the hash value H=Hash (F) and send to the cloud server to check the duplicate file has been found or not. The server database checks the short hash value of data file to ensure the uniqueness of file in the cloud environment. If the short value matched with current hash value then by sending the response message as duplicate file otherwise allowing to cloud to store file by saving hash value into the database of serve. This client side deduplication, referred to as hash-as-a-proof [6], poses a many security problems to cloud users.

2.1 Security Investigation

In spite of noteworthy resources rescuing, Pow schemes imports many security demands that may lead to sensitive data

- **Uncovering of Confidentiality**-Existing Pow schemes introduces confidentiality burden because of client uses a static-a hash a proof. For example if unauthorised user has a small static client side hash value, so he can easily fool cloud server as data owner demanding to upload intended file, then ,he advantage connection to data by giving a hash proof.
- **Breaching the privacy**-The outsourced data file must be ensuring that, cloud servers inadequate to entry outsourced data or user profiles.
- **Replace poisoned file-**The data file is encrypted on client side by selecting the random key to encrypt the file, the cloud server inadequate to check the consistency between outsourced file with current hash proof indeed given (Hash(F),Enc_k(F)),the storage server unable to verify.

The adversary user upload hash value, he demands the original file, then replace his poison file with original.

2.2 Relevant Works

Table 1: A survey on PoW scheme in deduplication.

Author	Title	Advantage	Dis-advantage
R di pietro [1]	Boosting efficiency security in PoW or deduplication	Problems of multitenant environments are solved by convergent encryption	But does not point out the data leakage
Halevi et al [3]	Proof of ownership in remotes storage systems	*It* overcomes the data leakage by providing the siblings of valid path.	But, it is random function of hash a proof.
Jia at al [6]	Weak-leakage resistant client side deduplication	Address confidentiality preservation concern in cross-user client side deduplication	Does not support malicious adversary.
Ng et al [4]	Private data deduplication in cloud storage	PoW scheme over encrypted data file of fixed block has unique commitment hash a proof. No need to reveal any information.	Computational cost requiring a generation of all commitments.

3. SYSTEM ARCHITECTURE

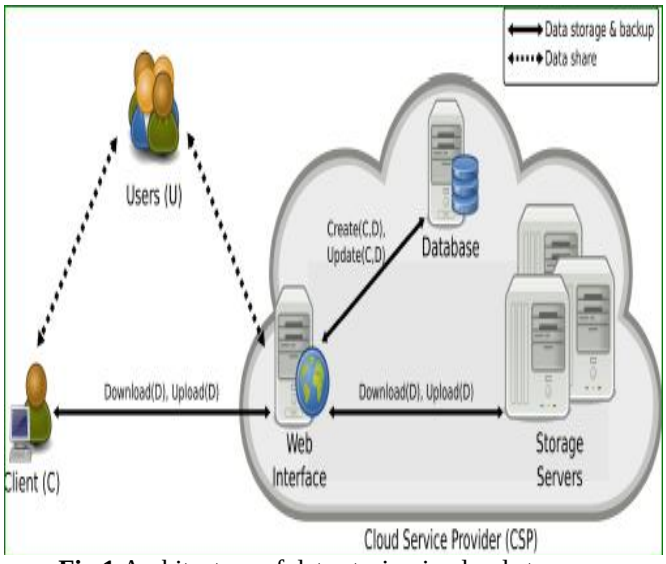

Fig 1.Architecture of data storing in cloud storage environment

It consists of fallowing entities which describes the how all entities are communicating each other through network architecture for cloud storage.

A CSP has virtualised provisioning of resources to control distributed cloud servers and which consist of cloud database to maintain user profiles as well as unique identifier of any file that we outsourced

Clients are the actual data owner he may be single or multiple .he can upload, download, giving permission to other cloud users. All permission's are integrated in the metafile which is intended to store to cloud.

All the restrictions are done by the client of data, he is responsible for integrating the access rights to cloud users, if the cloud user is authorised then he able to retrieval of data.

4. OUR PROOF OF OWNERSHIP IDEA

Idea of secure client side deduplication

- Encryption Key Extracted
- Markel-based tree over Encrypted Data
- Unique Identifier is extracted.
- Encrypt Decipher key with public key of cloud users.
- Integrate by data owner in user log file.

Our proof consists of mainly above five steps at client side to resistant to all weakness posed earlier schemes of PoW.

Encrypted key is extracted on file which intended to outsource into the cloud by applying one way hash function that is use of convergent encryption, on the other side key is acted as enciphering key of file and the file is encrypted. After the encryption accomplished on data file, the data owner has to derive a uniqueness identifier of the data file by applying Merkle tree over enciphered data file.

On the other side, to care of data from public cloud users of unauthorised entities access, enciphering the decryption key by using public key encryption like to say encrypt the deciphering key using public key of cloud user. the key then unified by cloud client in cloud user log file(metadata) and it is outsourced to cloud database ,it makes sure of confidentiality posed to malicious cloud users, and all access controlled by data owner.

4.1 Methodology

Client side: select the file (D) which is intended to outsource, apply one way hash function in order to extract the enciphering key, after derived the key must apply symmetric key encryption to original data file (D) and then run Merkle-tree over encrypted file to extract the unique identifier of the data file.

The enciphering key is encrypted with asymmetric key encryption from public key of cloud user. After successfully done all these must be store in user metadata file into cloud database, subsequent storage checks the uniqueness in cloud database if found it stops transferring of encrypted file to cloud storage server by saving the bandwidth of network traffic. And it ensures the highest level of privacy to cloud clients.

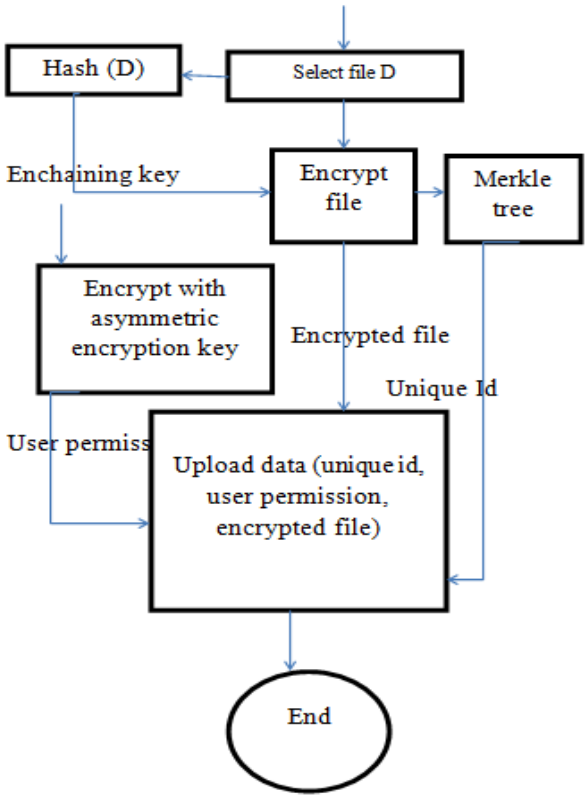

Fig 2.Client side deduplication

Cloud Storage: The cloud server checks the requested client is authorized, if he is the authorised can upload or download file from cloud server ,if client wants to upload the file checks the unique id of in the cloud database if it found then it display text duplicate no need to upload, otherwise allow to upload into cloud storage server.

And it checks against the unauthorised cloud users sharing of cloud data file, if the cloud user is not malicious user then allow to access data file.

If the file is not duplicate one then it intended to store the encrypted data file to cloud server and unique id, user permission's to cloud database.

4.2 Premise

The assumptions are made of our model are

- Establishment of secure channel between the client and CSP
- And uses of hash functions to enchaining key extraction
- Merkle tree gives the root value that is unique identifier extraction. It is file divided into number of blocks and find a hash for each block ,finally subsequent file root is created.

4.3 Cloud Storage

- Whenever the data owner need to upload the file to cloud. Client must derive the enciphering key from data file key_{file}, by applying hash function H() .
- And encrypt the file based on symmetric encryption algorithm.
- Extract the data identifier from Merkle-Tree over encrypted file.
- The identifier which must be unique in entire cloud database associated with that cloud client.

4.4 Cloud Share

The client outsourced data is sharing among the cloud users who are associated with that cloud owners, only authorised users can only get data that outsourced by data owner.

- The users should not connected to the cloud during depositing of the data file to cloud, the data owner integrate the access rights into the metadata file which present in cloud database.
- Even data owner can intimate URI to the cloud user after the deposit data itself, or store the cloud database with metadata file.
- Cloud User can access the data file whenever he need but must be authorised to access.

5. CONCLUSION

The demand for secured storage in the cloud and the fetching properties of convergent encryption makes integrate them, thus leads to more attractive solution to outsource of data storage along with more secure, efficient.

Our result combines the feature of cryptographic usage of both symmetric encryption and asymmetric encryption used for enciphering the data file and for meta data files, respectively due to the maximize the security towards privacy information to tackle several intrusions., and most appreciation job from the Merkle tree properties, this helps to data deduplication, as it lead to an pre-verification of data

presence in cloud servers, which is saves bandwidth. Also, the solution is shown to be resistant to unauthorized access to data and it maintains privacy during sharing process.

At last we know every solution as its own barrier still need to face some challenges, yet to discover issues to outsource the data into the cloud.

REFERENCES

[1] R. Di Pietro and A. Sorniotti. Boosting efficiency and security in proof of ownership for deduplication. In Proceedings of the 7th ACM Symposium on Information, Computer and Communications Security, ASIACCS ’12, pages 81–82, New York, NY, USA, 2012. ACM.

[2] M. Dutch. Understanding data deduplication ratios. SNIA White Paper, June 2008.

[3] D. Harnik, B. Pinkas, and A. Shulman-Peleg. Side channels in cloud services: Deduplicationn cloud storage. IEEE Security And Privacy, 8(6):40–47, 2010.

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

[5] C. Wang, Z. guang Qin, J. Peng, and J. Wang. novel encryption scheme for data deduplication. In Communications, Circuits and Systems (ICCCAS), 2010 International Conference on, pages 265–269, 2010.

[6] J. Xu, E.-C. Chang, and J. Zhou. Weak leakage-resilient client-side deduplication of encrypted data in cloud storage. In Proceedings of the 8th ACM SIGSAC symposium on Information, computer and communications security, ASIA CCS ’13, pages 195–206, New York, NY, USA, 2013. ACM.

[7] R. C. Merkle. A digital signature based on a conventional encryption function. In A Conference on the Theory and Applications of Cryptographic Techniques on Advances in Cryptology, CRYPTO ’87, pages 369–378, London, UK, UK, 1988. Springer-Verlag

[8] M. W. Storer, K. Greenan, D. D. Long, and E. L. Miller. Secure data deduplication. In Proceedings of the 4th ACM International Workshop on Storage Security and Survivability, StorageSS ’08, pages 1–10, New York, NY, USA, 2008. ACM

[9] http://github.com/openstack/swift.

BIOGRAPHIES

Amaresh is a received degree BE degree in computer science from VTU, Belagavi. Now perceiving M Tech (CSE) degree at Acharya institute of technology, Bengaluru.

Manjunath G S is Assistant professor of computer science at Acharya Institute of Technology, Bengaluru and received M Tech Degree computer science from same institute.