International Journal of Emerging Research in Management &Technology
ISSN: 2278-9359 (Volume-4, Issue-6)

Research Article

June 2015

Optimising Cloud Data Storage with Secure and Reliable Source Side Deduplication

Shwetha G.S.*, Anand Kumar K. R.
SJBIT, Bangalore,
India

Abstract:

Cloud computing is an prevalent data interactive paradigm for processing of large amounts of data and storage without considering the local infrastructure limitations. The advent of cloud storage enables the organizations and enterprises to outsource their data to third party cloud service providers (CSP). Though the services provided by Cloud has many advantages, the users voluntarily give up the physical possession of their outsourced data which inevitably poses new security and privacy risks and yet another challenge is the management of ever increasing volume of data for CSP. In order to deal with these security issues, a new secure storage with deduplication scheme has been adopted. In order to provide security of outsourced data against malicious users and curious CSP's, a new convergent encryption technique is proposed. Every Client encrypts the file before uploading and the encrypted file is input to Hash algorithm which generates a unique identifier for every file. The Client specifies the authorized users and their access rights in a metafile uploaded to the cloud and user can decrypt the downloaded file with his secret key.

Keywords: Cloud Storage, Deduplication, Integrity, Security, Privacy.

I. INTRODUCTION

Cloud Computing is defined as "A large-scale distributed computing paradigm that is driven by economies of scale, in which a pool of abstracted, virtualized, dynamically scalable, managed computing power, storage, platforms, and services are delivered on demand to external customers over the Internet[1]".It is the means of delivering any and all information technology components from computing power to computing infrastructure, applications ,business processes and collaboration actually delivering IT as a service. Cloud computing is an advanced style of computing where applications, data and resources are provided to users as a service over the web .

Moving the data onto the cloud offers significant advantages in resource saving and provides great convenience to users as they don't have to worry about the complexities of hardware, software and their maintenance. With the potentially infinite storage space offered by cloud providers, users tend to use as much as space as they can and vendors constantly look for techniques aimed to minimize redundant data and maximize space savings. .Technology is changing every day and organizations are expected to adopt to the changes and transform enterprise IT with self-service, charge back, service catalogs, resource orchestration, complete application provisioning hybrid IT, reservations, etc.A technique which has been widely used and adopted is deduplication. Deduplication is a technique that stores only a single copy of each file on a storage server regardless of how many clients ask to store that file.

Despite the significant advantages, it brings several new security issues towards the user's outsourced data and confidentiality is guaranteed through encryption. Unfortunately, deduplication and encryption are two conflicting technologies. While the aim of deduplication is to detect identical data segments and store them only once, the result of encryption is to make two identical data segments indistinguishable after being encrypted. This means that if data is encrypted in a standard way cloud, the cloud storage provider cannot apply deduplication since two identical data segments will be different after encryption. On the other hand if data is not encrypted by users, confidentiality cannot be guaranteed and data are not protected against curious cloud storage providers.

In order to meet the two conflicting requirements a technique called convergent encryption has been proposed in which the data to be uploaded is encrypted with the key generated from the hash of its contents. Convergent encryption is a good candidate to achieve confidentiality and deduplication at the same time. The security of the this system relies on its new architecture where in addition to basic storage provider; a metadata manager is defined. The confidentiality is achieved through convergent encryption and the metadata manager is responsible for key management task. Thus, the underlying deduplication is performed at the file level and at the client side.

The remainder of the work is organized as follows. First in section II a brief background about deduplication and convergent encryption has been explained. Section III provides an overview of the related work. Section IV describes the cloud storage architectures used. Section V describes the proposed system and then followed by security analysis. Finally Section VII presents conclusion and future work.

II. BACKGROUND

A. Deduplication

In deduplication process, unique chunks of data or byte patterns are identified and stored during analysis. As the analysis continues, other chunks are compared to the stored copy and whenever a match occurs, the redundant chunk is replaced

with a small reference that points to the stored chunk. There are mainly two categories of deduplication: file-level deduplication and block-level deduplication. In block level deduplication the block size can either be fixed or variable. It is also categorized based on the location at which the deduplication is performed: if the data is deduplicated at the server side then it is called target based deduplication otherwise source-based deduplication. In target-based deduplication the file/data is first sent to the server and then the deduplication is performed whereas in source-based deduplication, the client hashes the data to be uploaded and sends the unique hash value to the cloud server to check for uniqueness of the data. While deduplication at the client side can achieve bandwidth savings, it unfortunately can make the system vulnerable to side-channel attacks. On the other hand, by deduplicating data at the storage server, the system is protected against side channel attacks but the does not decrease the communication overhead.

B. Convergent Encryption

The basic idea of convergent encryption (CE) is to derive the encryption key from the hash of the plaintext. The simplest implementation of convergent encryption can be defined as follows: Alice derives the encryption key from her message such that K = H (M), where H is a cryptographic hash function; she can encrypt the message with this key, hence: C = E (K; M) = E (H (M); M), where E is a block cipher. By applying this technique, two users with two identical plaintexts will obtain two identical cipher texts since the encryption key is the same; hence the cloud storage provider will be able to perform deduplication on such cipher texts. Furthermore, encryption keys are generated, retained and protected by users. As the encryption key is deterministically generated from the plaintext, users do not have to interact with eachother for establishing an agreement on the key to encrypt a given plaintext. Therefore, convergent encryption seems to be a good candidate for the adoption of encryption and deduplication in the cloud storage domain.

III. RELATED WORK

Many systems have been developed to provide secure storage but traditional encryption processes are not suitable for deduplication process. Most works do not consider security as a concern for deduplicating systems. However Zhifengxiao et al. [1] systematically studied the security and privacy challenges in cloud computing environment based on the attribute driven methodology. The authors identified the most representative security/privacy attributes and also the vulnerabilities, which may be exploited by adversaries in order to perform various attacks and some of the defense strategies were also discussed.

Wenjing Lou et al. [2] focused on the cloud data storage security. The authors proposed an effective and flexible scheme by utilizing the homomorphic token with distributed verification of erasure-code data which achieves the integration of storage correctness insurance and data error localization.

Cond Wang et al. [3] developed a flexible distributed storage integrity auditing mechanism which allows the user to audit the cloud storage with very lightweight communication and computation cost. The auditing result not only ensures strong cloud storage correctness guarantee, but also simultaneously achieves fast error localization.

Luca Ferretti et al. [4] designed a novel architecture that integrates cloud database services with data confidentiality and possibility of executing concurrent operations on encrypted cloud database and it also eliminates the intermediate proxies that limit the elasticity, availability and scalability properties that are intrinsic in cloud-based solutions.

Hong Liu et al. [5] implemented a shared authority based privacy-preserving authentication protocol to address the privacy issues of cloud storage; the proposed protocol is attractive for multi-user collaborative cloud applications.

KanYag et al. [6] proposed the Cipher text-Policy Attribute-based Encryption (CP-ABE) to control the data access in cloud storage. The authors proposed the efficient and revocable data access control scheme for multi-authority cloud storage systems. The method described achieved both forward and backward security

Douceur et al. [7] studied the problem of Deduplication in multi-tenant environment. The authors proposed the use of convergent encryption i.e., deriving keys from the hash of plain text with the attempt to combine data confidentiality with the possibility of data deduplication. Then storer et al. [8] pointed out some security problems and presented a security model for secure data Deduplication. However, the two protocols focus on server-side Deduplication and do not consider data leakage settings, against malicious users.

Halevi et al. [9] the concept of *proof of ownership* (POW) was introduced in order to prevent the private data leakage. These schemes involve the server challenging the client to present the valid sibling paths for a subset of a merkle tree leaves.

Ng et al. [10] proposed a POW scheme over encrypted data. The file is divided into fixed-size blocks, where each block has a unique commitment. Hence, the owner has to prove the possession of a data chunk of precise commitment, with no need to reveal any secret information. However this scheme introduces a high computation cost.

IV. CLOUD ARCHITECTURE

A. Architecture

Fig. 1 illustrates the descriptive network architecture for cloud storage. It relies on the following entities for good management for client data.

- *Cloud Service Provider (CSP):* a CSP has significant resources to govern distributed cloud storage server and to manage its database servers. It also provides virtual infrastructure to host application services. These services can be used by the client to manage his datastored in the cloud servers.
- *Client:* a client makes use of provider's resources to store, retrieve and share data with multiple users. A client can be either an individual or an enterprise.

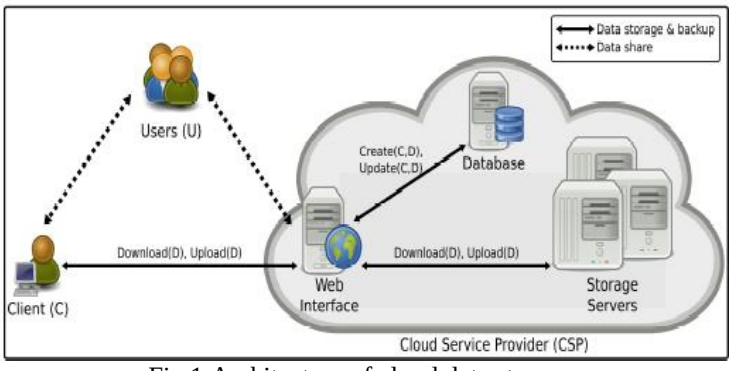

Fig 1 Architecture of cloud data storage

- *Users:* the users are able to access the content stored in the cloud, depending on their access rights which are authorizations granted by the client, like the rights to read, write or re-store the modified data in thecloud.

In practice, the CSP provides a web interface for the client to store data into a set of cloud servers, which are running in a cooperated and distributed manner. In addition, the webinterface is used by the users to retrieve, modify and restore data from the cloud, depending on their access rights. Moreover, the CSP relies on database servers to map client identities to their stored data identifiers and group identifiers.

B. Security Requirements

When outsourcing data to a third party, providing confidentiality and privacy becomes more challenging and conflicting. Privacy is a critical concern with regards to cloud storage due to the fact that clients' data reside among distributed public servers. Therefore, there are potential risks where the confidential information (e.g., financial data, health record) or personal information (e.g., personal profile) is disclosed. Meanwhile, confidentiality implies that client's data have to be kept secret from both cloud provider and other users.

Confidentialityremainsasoneofthegreatestconcerns.Thisislargelyduetothefactthatusersoutsourcetheirdataoncloudservers, which are controlled and managed by potentially untrusted CSPs. That is why, it is compulsory to provide secrecy by encrypting data before their storage in cloud servers while keeping the decryption keys out of the reach of CSP and any malicious user. For designing the most suitable security solutions for cloud storage, we are considering an honest but curious cloud provider, as a threat model. That is, it honestly performs the operations defined by our proposed scheme, but it may actively attempt to gain the knowledge of the outsourced data. In addition, an attacker can be either a revoked user with valid data decryption keys, an unauthorized group member or a group member with limited access rights. Therefore, secure data sharing should support flexible security policies including forward and backward secrecy.

C. Assumptions

Our solution considers the following assumptions .First; we assume that there is an established secure channel between the client and the CSP. This secure channel supports mutual authentication and data confidentiality and integrity. Hence, after successfully authenticating with the CSP, thesecloud users share the same resources in a multi-tenant environment.

Second, our solution uses the hash functions in the generation of the enciphering data keys. Hence, we assume that these cryptographic functions are strongly collision resistant, as it is an intractable problem to find the same output for different data files.

V. PROPOSAL FOR SECURE DATA STORAGE, BACKUP, SHARING

The proposed scheme performs Client side deduplication and is based on three different scenarios: Storage Backup, and sharing schemes.

A. Cloud Data Storage

When a client wants to store a new data file f in the cloud, he derives the enciphering key kf from the data contents, based on a one-way hash function H (). Note that data are stored enciphered in cloud servers, based on a symmetric algorithm. Hence, the data owner has to encipher the data, file that he intends to outsource. Then, he generates the data identifier ID. That is, it is the Hash over encrypted data. This identifier, associated to the file, must be unique in the CSP database.

Thus, the client starts the storage process by sending a ClientRequestVerif message to verify the uniqueness of the generated ID to his CSP.

New Data File Storage: The storage process consists in exchanging the four following messages:

- *ClientRequestVerif*: this first message contains the generated data identifier ID. This message is a request for the verification of the uniqueness of the ID. The CSP replies with a ResponseVerif message to validate or invalidate the claimed identifier.
- *ResponseVerif*: this acknowledgement message is generated by the CSP to inform the client about the existence of the requested MTf in its database.

- *ClientRequestStorage*: this message is sent by the client. If the file does not exist in the cloud servers, the client sends the file that he intends to store in the cloud, and the data decrypting key kf enciphered with the public keys of authorized users. Then, the enciphered kf is included in the Meta data of the file and it serves as an access rights provision.
- *ResponseStorage*: this acknowledgement message, sent by the CSP, is used to confirm to the client the success of his data storage.

B. Cloud Data Backup

The data backup process starts when the client requests for retrieving the data previously stored in the cloud. The data backup process includes the following messages:

- *ClientRequestBackup:* it contains the URL of the requested data that the client wants to retrieve. Upon receiving this client request, the CSP verifies the client ownership of the claimed file and generates a ResponseBackup message.
- *ResponseBackup*: in his response, the CSP includes the encrypted outsourced data kf (f).Upon receiving the ResponseBackup message, the client first retrieve the file metadata and deciphers the data decrypting key kf , using his secret key. Then, he uses the derived key to decrypt the request data file.

C. Cloud Data Sharing

In data sharing process, the client outsources his data to the cloud and authorizes a group of users to access the data. Users' access rights are granted by the data owner and managed by the CSP. That is, these access rights are also included in the metadata file. In addition, the CSP is in charge of verifying each recipient access permissions before sending him the outsourced data.

Each member of the group can start the data sharing process based on the two following messages:

- *UserRequestAccess:* This message contains the URL of the requested file. When receiving this message, the CSP searches for the read/write permissions of the recipient, and then, he generates a Response Access message.
- *ResponseAccess:* the CSP includes, in its response, the enciphered file kf (f). Upon receiving this message, each recipient retrieves the data decrypting key from user metadata. That is, he deciphers the associated symmetric key with his own private key. Then, he performs a symmetric decryption algorithm to retrieve the plaintext.

Our proposal provides a strong solution to improve the confidentiality of data in the cloud. In addition, the access to outsourced data is controlled by two processes. First, there is a traditional access list managed by the CSP. Second, the client has to own the private decrypting key to get the secret needed to retrieve the symmetric key compulsory needed to decipher data.

VI. SECURITY DISCUSSION

In this section an informal security analysis of the proposal is discussed.In addition, the possible refinements that could be made to mitigate other threats are also discussed.

Data confidentiality – In the present model, it is proposed to outsource encrypted data to remote storage servers. That is, data is stored enciphered in the cloud, based on a symmetric encryption algorithm using a per data key. This enciphering key is content related information, ensuring data deduplication in remote servers. Thus, the confidentiality of outsourced data is twofold.First; we ensure confidentiality preservation against malicious users. On one hand, when a user wants to store new data in the cloud, he has to send the data identifier ID, based on the encrypted file. Hence, this dual data identifier protection provides better secrecy to data outsourcing issue.

Second, we enhance data confidentiality against curious servers. That is, the data owner outsources encrypted contents. Then, he enciphers the decrypting key relying on an asymmetric scheme, in order to ensure efficient access control. As such, the CSP is also unable to learn the contents of stored data in his public servers.

Privacy – based on a cryptographic solution to keep data content secret, sensitive information are generally included in metadata whose leakage is a critical concern in a multi-tenant environment. Thus, our model mitigates to such privacy violation issue. On one side, the CSP identifies clients as data owners, while outsourcing the same content in remote servers. However, the cloud server cannot bind the consistency between the plaintext information and these data owners, as he has only access to hashed identifiers and encrypted contents. Consequently, he is unable to build user profiles, based on the received identifiers.

VII. CONCLUSIONS

The growing need for secure cloud storage services and the attractive properties of cryptography lead to the innovative solution to the data outsourcing security issue.

Our solution is based on a cryptographic usage of symmetric encryption used for enciphering the data file and asymmetric encryption for Meta data files, due to the highest sensibility of this information towards several intrusions. Besides, our solution is also shown to be resistant to unauthorized access to data and to any data disclosure during sharing process, providing two levels of access control verification.Finally, we believe that cloud data storage security is stillfull of challenges and of paramount importance, and manyresearch problems remain to be identified.

REFERENCES

[1] Zhifeng Xiao, Yang Xiao, *Security and Privacy in Computing*, IEEE Communications Surveys and &Tutorials, Vol. 15, No. 2, pp. 843- 859, Second quarter 2013.

[2] Cong Wang, Qian Wang, KuiRen, Wenjing Lou, *Ensuring Data Storage in Cloud Computing*, in Proc. Of IWQoS'09, pp. 1-9, July 2009.

[3] Cong Wang, Qian Wang, KuiRen, Wenjing Lou, *Towards Secure and Dependable Storage Services in Cloud Computing*, IEEE Transactions in Services Computing , Vol. 5, No. 2, pp.220-232, 2012.

[4] Luca Ferreti, Michele Colajanni, MircoMarchetti, *Distributed, Concurrent and Independent Access to Encrypted Cloud Databases*, IEEE Transactions in Parallel and Distributed Systems, Vol. 25, No. 2, pp. 437-446, Feb 2014.

[5] Hing Liu, HuanshengNing, QingxuXiong, Laurence T. Yang, *Shared Authority Bassed Privacy-preserving Authentication Protocol in Cloud Computing*, IEEE Transactions in Parallel and Distributed Systems, Vol: pp:99, 2014.

[6] Kan Yang, XiaohuaJia, *Expressive, Efficient and Revocable Data Access Control for Multi-Authority Cloud Storage*, IEEE Transactions in Parallel and Distributed Systems, Vol. 25, No. 7, pp1735-1745, July 2014.

[7] J. R. Douceur, A. Adya, W. J. Bolosky, D. Simon an M. Theimer, *Reclaiming Space from Duplicate files in a Serviceless Distributed File System*, in Proc. of 22nd International Conference on Distributed Computing Systems (ICDCS-2012).

[8] M. Dutch, *Understanding Data Duplication Ratios*, SNIA White Paper, June 2008.

[9] T. G. et.al, *GNU Multiple Precision Arithmetic Library*, 4.1.2, December 2002.

[10] S. Halevi, D. Harnik, B. Pinas, A. Shulman-Peleg, *Proofs of Ownership in Remote Storage Systems*, in Proc. Of the 18th ACM Conference on Computer and Communications Security, CCS'11, pp-491-500, New York, NY, USA, 2011.

[11] Danny Harnik, Benny Pinkas and Alexander Shulman-Peleg, Side,*Channels in Cloud Services: Deduplication in Cloud Storage, Security and Privacy*, IEEE, 8(6):40-47, 2010.

[12] Mihir Bellare, AlexandraBoldyreva and Adam ONeill.Deterministic and efficiently searchable encryption.In Advances in Cryptology-CRYPTO 2007,pages 535-552.Springer,2007.

