RESEARCH ARTICLE OPEN ACCESS

Implementation Idea for Secure Data Deduplication Using Hybrid Cloud Approach

Ms. Deepali C. Ghosalkar
M.E
Department of Computer Science and Engineering
Vidyalankar Institute of Technology
Mumbai - India

ABSTRACT

One critical challenge of cloud storage services is the management of the ever increasing volume of data. To make data management scalable in cloud computing, data deduplication is one of the important compression technique for eliminating duplicate copies of repeating data.

In traditional encryption systems, identical data copies of different users will lead to different ciphertexts after encryption, making deduplication impossible.

In this paper, convergent encryption has been proposed for data deduplication feasible. Data deduplication is popular technique used in cloud storage. The main purpose of data deduplication is to reduce the amount of storage space and saves bandwidth.

Keywords: - Deduplication, secured data deduplication, convergent encryption, hybrid cloud.

I. INTRODUCTION

Data deduplication [5] is a technique for eliminating duplicate copies of data, and has been widely used in cloud storage to reduce storage space and upload bandwidth. It is an arising challenge is to perform secure deduplication in cloud storage.

In traditional encryption system, each user was using different encryption keys to encrypt the data so it produces different ciphertext for same files. But in proposed deduplication, each user is using the convergent encryption to encrypt the data so it produces same ciphertext for same files, making deduplication possible.

Convergent encryption [6], is a cryptosystem that produces identical ciphertext from identical plaintext files. This has applications in cloud computing to remove duplicate files from storage.

II. EXISTING SYSTEM

Identical data copies of different users will lead to different ciphertexts, making deduplication impossible. In traditional encryption, while providing data confidentiality, is incompatible with data deduplication. Especially traditional encryption requires different users to encrypt their data with their own key. This systems cannot support differential authorization duplicate check, which is important in many applications.

III. PRPOSED SYSTEM

Convergent encryption has been proposed to enforce data confidentiality while making deduplication feasible. It encrypts/decrypts a data copy with a convergent key, which is obtained by computing the cryptographic hash value of the content of the data copy.

After key generation and data encryption, users retain the keys and send the ciphertext to the cloud. Since the encryption operation is deterministic and is derived from the data content, identical information copies will generate the same convergent key and hence the same ciphertext.

IV. SYSTEM ARCHITECTURE

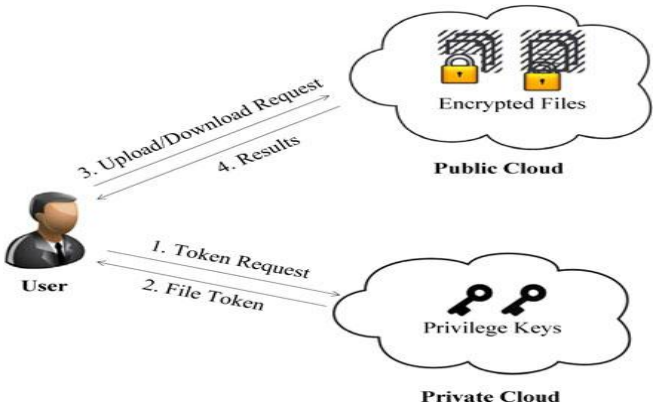

Fig. 1 Architecture of Authorized Deduplication

V. SECURE DATA DEDUPLICATION

Mostly Small Scale to Medium Scale companies outsource their data to S-CSP. Because Small Scale to Medium Scale companies cannot offered the storage cost. So service providers provides storage space to these companies on rent basis. S-CSP provides authority on the basis of their privileges. There are three entities defined in our system, that is, users, private cloud and S-CSP in public cloud. The S-CSP performs deduplication by checking if the contents of two files are the same and stores only one of them. The access right to a file is defined based on a set of privileges [7]. The exact definition of a privilege varies across applications. For example, we may define a role based privilege according to designation (e.g., Senior Engineer, Team Lead and Manager), or we may define a time based privilege that specifies a valid time period (e.g., 2016-01- 01 to 2016-01-31) within which a file can be accessed. A user, say Deep, may be assigned two privileges "Director" and "access right valid on 2016- 01-01", so that she can access any file whose access role is "Manager" and accessible time period covers 2016- 01- 01.

Each privilege is represented in the form of a short message called token. Each file is associated with some file tokens. A user computes and sends duplicate-check tokens to the public cloud for authorized duplicate check. Users have access to the private cloud server, a semi trusted third party which will aid in performing deduplicable encryption by generating file tokens for the requesting users. Users are also provisioned with per-user encryption keys and credentials.To prevent unauthorized access, a secure proof of ownership (POW) protocol [3] is also needed to provide the proof that the user indeed owns the same file when a duplicate is found.

VI. MODULES

1. User Module

In this module, Users are having authentication and security to access the system.

A user outsource their data for storage to the S-CSP and access the data later. To save storage space and to save bandwidth S-CSP will allow single copy of data using deduplication technique. Each user is issued a set of privileges in the setup of the system to maintain authorized deduplication. Each file is protected with the convergent encryption key and privilege keys to realize the authorized deduplication with differential privileges.

File Tag (File) -

Proposed system calculates File Tag using SHA-1 Algorithm. File Tag will be same for same file.

Token Req(Tag, UserID) -

User will request to Private Server for Token generation. Proposed system will generate the Token using File Tag and User ID.

Dup Check Req (Token) -

After receiving the Token from Private Serve, User will request to the Public Server to check duplicate token.

Share Token Req (Tag, {Priv.}) -

It requests the Private Server to generate the Share File Token with the File Tag and Target Sharing Privilege Set.

File Encrypt (File) -

Proposed system encrypts the File with Convergent Encryption using 256-bit AES algorithm, where the convergent key is from SHA-256 Hashing of the file.

File UploadReq (FileID, File, Token) –

It uploads the File Data to the Storage Server if the file is Unique and updates the File Token stored.

2. Private Server

Private Server includes corresponding request handlers for the token generation and maintains a key storage with Hash Map.

A private cloud is a particular model of cloud computing that involves a distinct and secure cloud based environment in which only authorized users can operate. Private Cloud provides higher security and privacy. With so many services like iCloud and Dropbox getting hacked these days, it's no surprise that more people want to pull their data off the cloud. Instead of missing out on those great syncing features, though, we can create our own cloud storage service that we control with a service called ownCloud.

Token Gen(Tag, UserID) -

Private Server generates the Token using HMAC-SHA-1 algorithm and privilege keys.

ShareTokenGen(Tag, {Priv.}) -

Share token with the corresponding privilege keys of the sharing privilege set with HMAC-SHA-1 algorithm.

3. Public Server (Cloud Storage Server)

Storage Server provides deduplication and data storage with following handlers and maintains a map between subsisting files and associated token with Hash Map.

This provides storage service for data in public cloud. The S-CSP provides the data outsourcing service and stores data on behalf of the users. With the help of data deduplication system we can reduce the cost of storage. The storage server should have large space so that users can outsource their data. Amazon EC2, Dropbox etc. Are the examples of large storage servers.

DupCheck (Token) -

This process checks Token with existing Tokens.

FileStore (FileID, File, Token) -

This process stores the File on Disk and updates the Mapping.

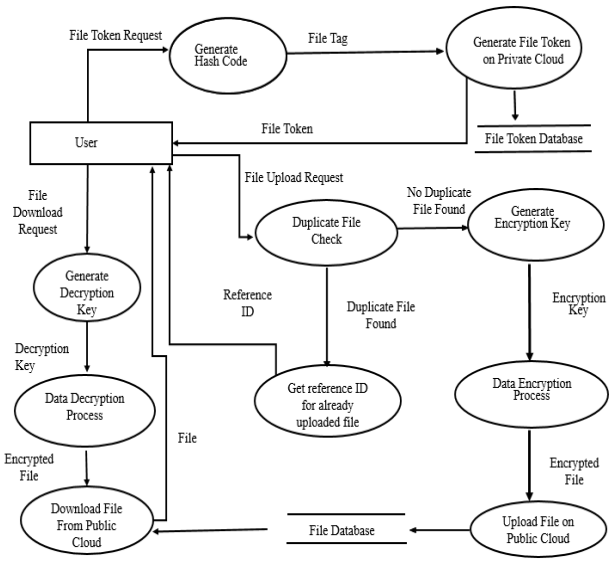

Fig. 2 Flow of Execution of The Proposed System

Hybrid Cloud approach is nothing but the combination of Public Cloud and Private Cloud. When user wants to store any file in Public Cloud, first he send a request to Private Cloud for File Token. Then Private Cloud generates File Token and gives to User. After getting the File Token, User can send to Public Cloud for Deduplication, if File Token already is there then Public Cloud check Owner of the File, if he/she is authenticated person then Public cloud sends a existing File reference id to user. So like that we can achieve Deduplication in Cloud. Suppose, if File Token is not available in Public Cloud then User can be perform convergent encryption and store the file in Public Cloud.

VII. RELATED WORK

The basic idea is that we can limit the damage of stolen data if we decrease the value of that stolen information to the attacker. We can achieve this through a 'preventive' disinformation attack. We posit that secure deduplication services can be implemented given two additional security features: User Behaviour Profiling and Decoy [1].

We present *FadeVersion*[2] , a secure cloud backup system that supports both *version control* and *assured deletion*. *FadeVersion* allows fine-grained assured deletion, such that cloud clients can specify particular versions or files on the cloud to be assuredly deleted, while other versions that share the common data of the deleted versions or files will remain unaffected.

In DupLESS[4], clients encrypt under message-based keys obtained from a key-server via an oblivious PRF protocol. It enables clients to store encrypted data with an existing service, have the service perform deduplication on their behalf, and yet achieves strong confidentiality guarantees. Deduplicated storage can achieve performance and space savings close to that of using the storage service with plaintext data.

VIII. FUTURE WORK

Hash Value computation for large File Size will be performance hit. So further in future work we can use Attribute-Based Encryption for encrypting the File Data. To compute the File Tag, instead of using whole file data we will use File Attributes.

IX. CONCLUSION

With the help of Hybrid Cloud Approach and Convergent Encryption we achieved secured data deduplication. To find duplicate data Proof of Ownership (PoW) protocol is used. Which gave reference of file which is already present on Public Cloud.

REFERENCES

[1] J. Li, X. Chen, M. Li, J. Li, P. Lee, and W. Lou, "Secure deduplication with efficient and reliable convergent key management," in Proc. IEEE Trans.ParallelDistrib.Syst.,http://doi.ieeecomputersociety.org/10.1109/TPDS.2013.284,2013.

[2] A. Rahumed, H. C. H. Chen, Y. Tang, P. P. C. Lee, and J. C. S. Lui, "A secure cloud backup system with assured deletion and version control," in Proc. 3rd Int. Workshop Secutiry Cloud Comput., 2011,pp. 160–167.

[3] "Proofs of Ownership in Remote Storage Systems" Shai Halevi1, Danny Harnik2, Benny Pinkas3, and Alexandra Shulman-Peleg2 1IBM T. J. Watson Research Center, 2IBM Haifa Research Lab, 3Bar Ilan University.

[4] M. Bellare, S. Keelveedhi, and T. Ristenpart, "Dupless: Serveraided encryption for deduplicated storage," in Proc. 22nd USENIX Conf. Sec. Symp., 2013, pp. 179–194.

[5] S. Quinlan and S. Dorward, "Venti: A new approach to archival storage," in Proc. 1st USENIX Conf. File Storage Technol., Jan. 2002.

[6] J. R. Douceur, A. Adya, W. J. Bolosky, D. Simon, and M. Theimer, "Reclaiming space from duplicate files in a serverless distributed file system," in Proc. Int. Conf. Distrib. Comput. Syst., 2002.

[7] Jan Stanek, Alessandro Sorniotti, Elli Androulaki, and Lukas Kencl, "A Secure Data Deduplication Scheme for Cloud Storage," Tech Rep. IBM Reasearch, Zurich.