Deduplication of encrypted data in cloud

Sri Priya. N[1], Sahil. N[2], Vamsi Kiran Reddy. N[3], Srinadh. N[4]
Department of Computer Science and Engineering [1,2,3,4], Prathyusha Engineering College [1,2,3,4]

ABSTRACT: Attribute-based encryption (ABE) has been widely used in cloud computing where a data provider outsources their encrypted data to a cloud service provider and can share the data with users possessing specific credentials. However, the standard ABE system does not support secure Deduplication, which is crucial for eliminating duplicate copies of identical data in order to save storage space and network bandwidth. Here, we present an attribute-based storage system with secure Deduplication in a hybrid cloud setting, where a private cloud is responsible for duplicate detection and a public cloud manages the storage. Compared with the prior data Deduplication systems, this system has two advantages. Firstly, it can be used to confidentially share data with users by specifying access policies rather than sharing decryption keys. Secondly, it achieves the standard notion of semantic security for data confidentiality while existing systems only achieve it by defining a weaker security notion. In addition, we put forth a methodology to modify a cipher-text over one access policy into cipher-texts of the same plaintext but under other access policies without revealing the underlying plaintext.

1. INTRODUCTION TO THE PROJECT

Cloud computing greatly facilitates data providers who wants to outsource their data to the cloud without disclosing their sensitive data to external parties and would like users with certain credentials to be able to access the data. This requires data to be stored in encrypted forms with access control policies such that no one except users with attributes of specific forms can decrypt the encrypted data. An encryption technique that meets this requirement is called attribute-based encryption (ABE), where a user's private key is associated with an attribute set, a message is encrypted under an access policy over a set of attributes, and a user can decrypt a cipher-text with private key if set of attributes satisfies the access policy associated with this cipher-text. However, the standard ABE system fails to achieve secure Deduplication, which is a technique to save storage space and network bandwidth by eliminating redundant copies of the encrypted data stored in the cloud. On the other hand, existing constructions for secure Deduplication is not built on attribute-based encryption. Nevertheless, since ABE and secure Deduplication have been widely applied in cloud computing, it would be desirable to design a cloud storage system possessing both properties.

2. SCOPE OF THE PROJECT

This System can be used as an application to detect duplication of data in cloud to avoid repeaters. This mechanism helps to save space in Cloud server. Cloud computing provides seemingly unlimited "virtualized" resources to users as services across the whole Internet, while hiding platform and implementation details. In cloud service providers offer both highly available storage and massively parallel computing resources at relatively low costs. As cloud computing becomes

We consider the following scenario in the design of an attribute-based storage system supporting secure Deduplication of encrypted data in the cloud, in which the cloud will not store a file more than once even though it may receive multiple copies of the same file encrypted under different access policies. A data provider, Bob, intends to upload a file *M* to the cloud, and share *M* with users having certain credentials. In order to do so, Bob encrypts *M* under an access policy A over a set of attributes, and uploads the corresponding cipher-text to the cloud, such that only users whose sets of attributes satisfying the access policy can decrypt the cipher-text. Later, another data provider, Alice, uploads a cipher-text for the same underlying file *M* but ascribed to a different access policy A. Since the file is uploaded in an encrypted form, the cloud is not able to discern that the plaintext corresponding to Alice's cipher-text is the same as that corresponding to Bob's, and will store *M* twice. Obviously, such duplicated storage wastes storage space and communication bandwidth.

prevalent, an increasing amount of data is being stored in the cloud and shared by users with specified privileges, which define the access rights of the stored data. One critical challenge of cloud storage services is the management of the ever-increasing volume of data. Data De-duplication is a specialized data compression technique for eliminating duplicate copies of repeating data in storage. The technique is used to improve storage utilization and can also be applied to network data transfers to reduce the number of bytes that must be sent. Instead of keeping multiple data copies with the

same content, De-duplication eliminates redundant data by keeping only one physical copy and referring other redundant data to that copy. Deduplication can take place at either the file level or the block level. For file level De-duplication, it eliminates duplicate copies of the same file. Deduplication can also take place at the block level, which eliminates duplicate blocks of data that occur in non-identical files.

3. EXISTING SYSTEM

In Existing System,

- Existing data De-duplication systems, the private cloud is occupied as a different to allow data owner/users to securely perform duplicate check with differential privileges.
- In architecture is practical and has involved much interest from researchers.
- The data owners only outsource their data storage by utilizing public cloud while the data operation is managed in private cloud.

4. PROPOSED SYSTEM

In this paper we proposed

- In using advanced De-duplication system supporting authorized duplicate check. In this new De-duplication system, a hybrid cloud architecture is introduced to solve the problem.
- The private keys for privileges will not be issued to users directly, which will be kept and managed by the private cloud server instead.
- In this way, the users cannot share these private keys of privileges in this proposed construction, which means that it can prevent the privilege key sharing among users in the above straightforward construction. To get a file token, the user needs to send a request to the private cloud server.
- The private cloud server will also check the user's identity before issuing the corresponding fill token to the user.
- The authorized duplicate check for this file can be performed by the user with the public cloud before uploading this file.
- Based on the results of duplicate check, the user either uploads this file or runs PoW.

5. ADVANTAGES

- In this proposal, faster recovery of encrypted data is performed.
- It effectively increases the network bandwidth.
- It deletes the duplicate files.
- It is internally built with high security.

6. LIMITATIONS

- Problems such as construction of authorized De-duplication has several serious security problems, which are listed below
- Each user will be issued private keys for their corresponding privileges.
- A restriction makes the authorized De-duplication system unable to be widely used and limited.

7. STUDY OF THE SYSTEM

Feasibility Study:

- A Feasibility Study is the analysis of a problem to determine if it can be solved effectively.
- The results determine whether the solution should be implemented.
- This activity takes place during the project initiation phase and is made before significant expenses are engaged.

Technology and system feasibility

The assessment is based on an outline design of system requirements in terms of Input, Processes, Output, Fields ,Programs, and Procedures. This can be quantified in terms of volumes of data ,trends, frequency of updating, etc. in order to estimate whether the new system will perform adequately or not this means that feasibility is the study of the based in outline**.**

Economic feasibility

Economic analysis is the most frequently used method for evaluating the effectiveness of a new system. More commonly known as benefit analysis**,**the procedure is to determine the benefits and savings that are expected from a candidate system and compare them with costs. If benefits outweigh costs, then the decision is made to design and implement the system. An entrepreneur must accurately weigh the cost versus benefits before taking an action.

Legal feasibility

Determines whether the proposed system conflicts with legal requirements, e.g. a data processing

system must comply with the local Data Protection Acts.

8. NUMBER OF MODULES

A module is a bounded contiguous group of statements having a single name and that can be treated as a unit. In other words, a single block in a pile of blocks.

8.1 Authorization control creation and Key Generation:

Authorized user is able to use his/her individual private keys to generate query for certain file and the privileges he/she owned with the help of private cloud, while the public cloud performs duplicate check directly and tells the user if there is any duplicate.

Unauthorized users without appropriate privileges or file should be prevented from getting or generating the file tokens for duplicate check of any file stored at the S-CSP. In system, the S-CSP is honest but curious and will honestly perform the duplicate check upon receiving the duplicate request from users. The duplicate check token of users should be issued from the private cloud server in our scheme.

8.2 Owner uploading and Built Hybrid Cloud:

In this new De-duplication system, a hybrid cloud architecture is introduced to solve the problem. The private keys for privileges will not be issued to users directly, which will be kept and managed by the private cloud server instead. In this way, the users cannot share these private keys of privileges in this proposed construction, which means that it can prevent the privilege key sharing among users in the above straightforward construction. To get a file token, the user needs to send a request to the private cloud server.. To perform the duplicate check for some file, the user needs to get the file token from the private cloud server. The private cloud server will also check the user's identity before issuing the corresponding file token to the user. The authorized duplicate check for this file can be performed by the user with the public cloud before uploading this file. Based on the results of duplicate check, the user either uploads this file or runs PoW.

8.3 Detect Deduplication:

Convergent encryption provides data confidentiality in De-duplication. A user derives a convergent key from each original data copy and encrypts the data copy with the convergent key. In addition, the user also 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. To detect duplicates, the user first sends the tag to the server side to check if the identical copy has been already stored. Note that both the convergent key and the tag are independently derived and the tag cannot be used to deduce the convergent key and compromise data confidentiality. Both the encrypted data copy and its corresponding tag will be stored on the server side.

8.4 Key Exchanging:

The private keys for the privileges are managed by the private cloud, the file token requests from the users. The interface offered by the private cloud allows user to submit files and queries to be securely stored and computed respectively. The private cloud server will also check the user's identity before issuing the corresponding file token to the user. The authorized duplicate check for this file can be performed by the user with the public cloud before uploading this file.

8.5 Verification and File Retrieving:

A symmetric key x for each user will be select and set of keys will be sent to the private cloud. An identification protocol equals to proof and verify is also defined, where Proof and Verify are the proof and verification algorithm respectively. In each user *U* is assumed to have a secret key *to* perform the identification with servers.

Assume that user *U* has the privilege set *PU*. It also initializes a PoW protocol POW for the file ownership proof. The private cloud server will maintain a table which stores each user's public information and its corresponding privilege.

It first sends a request and the file name to the S-CSP. Upon receiving the request and file name, the S-CSP will check whether the user is eligible to download *file*. If failed, the S-CSP sends back an abort signal to the user to indicate the download failure. Otherwise, the S-CSP returns the corresponding cipher text *CF*, upon receiving the encrypted data from the S-CSP, the user uses the key stored locally to recover the original file.

9. CONCLUSION

Attribute-based encryption (ABE) has been widely used in cloud computing where data providers outsource their encrypted data to the cloud and can share the data with users possessing specified credentials. On the other hand, De-duplication is an important technique to save the storage space and network bandwidth, which eliminates duplicate copies of identical data. However, the standard ABE systems do not support secure De-duplication, which makes them costly to be applied in some commercial storage services. In this paper, we

presented a novel approach to realize an attribute-based storage system supporting secure De-duplication. Our storage system is built under a hybrid cloud architecture, where a private cloud manipulates the computation and a public cloud manages the storage. The private cloud is provided with a trapdoor key associated with the corresponding cipher-text, with which it can transfer the cipher-text over one access policy into cipher-texts of the same plaintext under any other access policies without being aware of the underlying plain-text. After receiving a storage request, the private cloud first checks the validity of the uploaded item through the attached proof. If the proof is valid, the private cloud runs a tag matching algorithm to see whether the same data underlying the cipher-text has been stored. If so, whenever it is necessary, it regenerates the cipher-text into a cipher-text of the same plaintext over an access policy which is the union set of both access policies. The proposed storage system enjoys two major advantages. Firstly, it can be used to confidentially share data with other users by specifying an access policy rather than sharing the decryption key. Secondly, it achieves the standard notion of semantic security while existing De-duplication schemes only achieve it under a weaker security notion.

REFERENCES:

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

[2]. M. Bellare, S. Keelveedhi, and T. Ristenpart. Message-locke encryption and secure De-duplication. In *EUROCRYPT*, pages 296 – 312, 2013.

[3]. M. Bellare, C. Namprempre, and G. Neven. Security proofs for identity-based identification and signature schemes. *J. Cryptology*, 22(1):1–61, 2009.

[4]. M. Bellare and A. Palacio. GQ and schnorr identification schemes: Proofs of security against impersonation under active and concurrent attacks. In *CRYPTO*, pages 162–177, 2002.

[5]. 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.

[6]. D. Bhagwat, K. Eshghi, D.D.E. Long, and M. Lillibridge. Extreme binning: Scalable, parallel De-duplication for chunk-based file backup. In Proc. IEEE MASCOTS, Sep 2009.

[7]. Burton H. Bloom. Space/time trade-offs in hash coding with allowable errors. Comm. of the ACM, 13(7):422–426, 1970. [5] A.T. Clements, I. Ahmad, M. Vilayannur, and J. Li. Decentralized De-duplication in SAN cluster file systems. In Proc. USENIX ATC, Jun 2009.