NATIONAL CONFERENCE ON EMERGING TRENDS IN ENGINEERING AND TECHNOLOGY.
NCETET-2017, 7st March 2017, BVCOE, Kolhapur.
ISSN: 2231-5063.

PRIVACY PRESERVING PUBLIC AUDITING WITH DATA DEDUPLICATION IN CLOUD COMPUTING

Ms. Neha M. Gondkar, Mr. Rahul P. Mirajkar
P.G., student, Department of Computer Engineering, Bharati Vidyapeeth's College of Engineering, Kolhapur , India.
Assistant Professor, Department of Computer Science & Engineering, Bharati Vidyapeeth's College of Engineering, Kolhapur, India.

Abstract: Storage represents one of the most commonly used cloud services. Data integrity and storage efficiency are two key requirements when storing users' data. Public auditability, where users can employ a Third Part Auditor (TPA) to ensure data integrity, and efficient data deduplication which can be used to eliminate duplicate data and their corresponding authentication tags before sending the data to the cloud, offer possible solutions to address these requirements. Here, we propose a privacy- preserving public auditing scheme with data deduplication. Our analytical and experimental results show the efficiency of the auditing by reducing the number of pairing operations need for the auditing.

Keywords: Deduplication, CSP(Cloud service provider),TPA(Third party auditor) .

IX. INTRODUCTION

In today's world everything is associating with internet in some or other ways. People are using different applications for there ease in day today work. These applications are nothing but generating and playing with big amount of data. In many applications users may or may not need this data in future. Not all of them can afford to store and manage the large amount of data. Cloud service providers provide infinite amount of space to these users to store there on cloud. There are many aspects regarding the storage of the data on cloud such as management of the data, security of the data, processing on stored data etc. One of the dominant aspect is security. Cloud service provider only stores the data but many users want their data to be secure many times. For example the application storing the data about the call details of the person may demand the security of the data. So they need some encryption algorithm which makes there data secure hence the stored data is generally in encrypted form.

Though providers have very large storage space they need to have utilization of this space as high as possible. To increase storage efficiency, storage providers often identify and remove redundant data and keep only one copy of each file (file-level deduplication) or block (block-level deduplication). Data deduplication may occur before the data is transmitted to the cloud or after data transmitted.Different users can store data under different encryption technique so there are very good chances to find duplicate data on cloud which will definitely reduce the utilization of space. Data dedupliction scheme can solve this issue. Dedupliction eliminates duplicate copies of data. To make deduplication possible we need some confluent encryption scheme which will generate exactly same ciphertext for same files .

One more significant functionality that users want with huge data storage is mechanism to access that data. Access control gives access to data only to the users who are authorized to access the particular service. For example nowadays users share their personal data on social networking applications and they assign authority to only limited group of people to access their data. Here implementation of some access control mechanism can take care of this.

Again user always need the exact copy of data which he stores on cloud. Therefore privacy of the stored data is very necessary. Privacy can be determined by checking the integrity of the data. At any point of time user can feel that the data has been altered and he may want to crosscheck for the integrity of the data. Here in this case user do not have trust on provider. If provider provides the privacy of data then the user can be assure to outsource there data on cloud.

Existing System

Mingjun Wang [1] explains how duplicate data stored on cloud under different encryption scheme decrease the utilization of storage space specially for big data storage which results from IOT applications. It explains the

deduplication technique calculated with hash value of file which will create same signature for same file. Here author considers three basic entities, data owner, data holder/user and cloud service provider. Pasquale Puzio and Refik Molva [2] did block level deduplication which also provides data confidentiality. It explains why having only convergent encryption is not enough secure solution for cloud users. As deduplication is block level here it needs key for every block so it provides efficient solution for management of keys. It also explains simple retrieval protocol for stored data. Cong Wang, Sherman S.M. Chow, Qian Wang, Kui Ren, and Wenjing Lou proposed [3] scheme which perform the data integrity check. Here they have explained public auditing and batch auditing mechanism in detail.

X. PROPOSED SYSTEM

In proposed architecture the data outsourced to the cloud server is encrypted using Attribute Based Encryption (ABE) which maintains privacy of the data. Deduplication is performed on the stored data which removes duplicate copies of the data and maintains single copy. e.g. If two users upload same document then only one copy will be stored which will contain the owner list of specific document. So data flexibility with low ownership cost is achieved. Access control with multi authority mechanism implements protocol for access to the stored data. It ensures the authorized and secure access. To preserve data integrity in case where the cloud server itself is not trusted server, system provides third party auditor checking of the data which checks if the data is altered or removed. With the help of this functionality the user will be rest assured about the correctness of their data.

A] Block Diagram

The Block diagram as shown in fig [1].

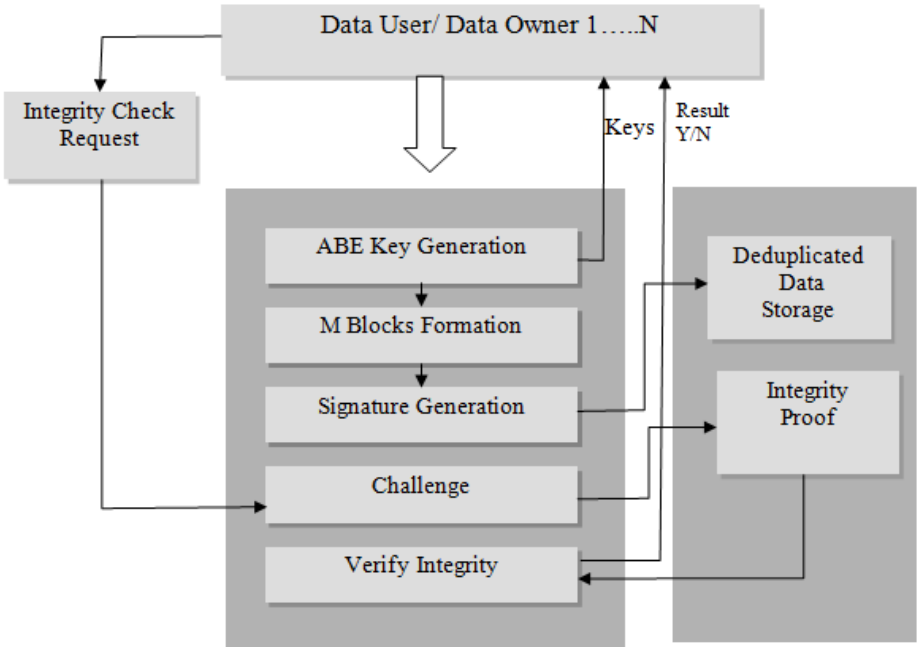

Fig 1: Block Diagram of System

The System contains four steps as follows:

1. File upload/Data upload and setup

The data owner who wants to uploads file to the server make a request for keys to audit server who generates keys. Now owner/user is provided with the pair of keys. Public key for encryption and the secret key for decryption. In setup process after key generation the file preprocessing takes place. Here the file is divided into M fixed size blocks.

Algorithm For key generation
$G_1,G_2...G_t$: multiplicative cyclic groups of prime order p.
$\{G_1,g_1\}$: pair of multiplicative cyclic group of prime no p and its generator.
x,u : variables
g : generator
$m_1,m_2....m_n$: data blocks
n :number of blocks
ssk : private key
spk : public key
Let $m_1,m_2....m_n \in Z_p$

1. User send request to KeyGen to generate public and secrete attributes.
2. Choose random key pair (spk,ssk)
3. For random x from Zp and random element u fromG_1 compute $v = g^x$
4.Secrete parameters = (x,ssk)
5.Public parameters = (v,u,spk)

2. Signature generation

The signature generation is a process that generate the verification value or metadata. This value is used by auditor for comparison to check correctness of data.

Algorithm for signature generation
F : File / Data
$m_1,m_2....m_n$: data blocks
n : number of blocks
H(.) : cryptographic hash function
ssk : private key
spk : public key
$G_1,G_2...G_t$: multiplicative cyclic groups of prime order p.
$\{G_1,g_1\}$: pair of multiplicative cyclic group of prime no p and its generator.
Let $m_1,m_2....m_n \in Z_p$
name : random variable $\in Z_p$
i : identifier for F
ϕ : set of authenticator.

1. Input : File= $\{m_j\}$
2. User runs SigGen to calculate authenticator A.
3. For each element i
4. Set W_i= name|| i
5.Ai←$H((W_i)\times u^{mi})^x \in G$.
6. Compute $\phi = \{\sigma_i\}_{1\le i\le n}$
7. Compute File tag t.
8.t :name || $SSig_{ssk}$(name)
9. Output : $SSig_{ssk}$(name) :Signature for name by private key ssk

3. Key Distribution and Data Deduplication

Deduplication process do not allow the duplicate copy of a data to reside in cloud storage.
when user has file to store on cloud server calculates hash value of each block and then it compares to detect duplicate ones. If it found the duplicate copy then it imply add the user in the owner list L associated with that block.
File is of the form F ={ m, Ϭ,L}
m : data block
Ϭ : Signature
L : group of owners or users.

4. Integrity Verification

Here data integrity check is performed at in case if the cloud server is not trusted. The data owner can ask for integrity check request and generates the request to the third party auditor. Third party auditor randomly picks the specific blocks and generates challenge to the server. According to server response the correctness is verified and owner gets response.

Algorithm for integrity checking

1.Input : File F with (ϕ,t)
2. Compare its signature.
3. If fails then quit.
4. Construct random challenge message.
5. Pick random element subset I = $\{e_1,...e_c\}$ of set [1..n]
6. For each element e
7. Choose random value V_e to generate position.
8. Prepare message chal = $\{(e ,V_e)\}$ which specify blocks that are to be checked.
9. Send chal msg to cloud server.
10.Cloud server generates response proof to auditor.
11.Auditor computes and verify the correctness.
12. If match found
output Y
else
output N.

IV. EXPERIMENTAL RESULTS

We can say that the data owner's computation complexity is *O*(1). The computation complexity of cloud service provider is *O*(*n*), say *n* is the number of data holders, but normally a cloud service provider has sufficient computation capability. This fact says that our system is computationally more efficient. As the proposed scheme has similar computation complexity to the scheme presented elsewhere.Kkey generation can be done at two places one at the data owner's which will take more computation time, second is third party which totally depend on a trusted third party. The system deployment and operation costs should be much lower than in the other one.

XI. CONCLUSION

We hope to construct model for cloud data which will provide efficient storage ,security of data, integrity of the data ,authorization over access. As we are combining different algorithms together ie. encryption, deduplication, access control, integrity checking etc. this will make our model definitely more efficient and secure.

XII. REFERENCES

1. Zheng Yan, Mingjun Wang, Yuxiang Li and V. Vasilakos , "Encrypted Data Management with Deduplication in Cloud Computing" IEEE Cloud Computing , 2016.

2. Pasquale Puzio, Refik Molva, Melek ¨Onen and Sergio Loureiro, " ClouDedup: Secure Deduplication with Encrypted Data for Cloud Storage", Cloud Computing Technology and Science,IEEE 5th International Conference ,2013

3. Cong Wang, Sherman S.M. Chow, Qian Wang, Kui Ren, and Wenjing Lou, " Privacy-Preserving Public Auditing for Secure Cloud Storage",IEEE Transaction on Computers, Vol2,2013.

4. J. Li et al., "A Hybrid Cloud Approach for Secure Authorized Deduplication," IEEE Trans. Paral- lel Distributed Systems, vol. 26, no. 5, 2015, pp. 1206–1216.

5. John Bethencourt,Amit Sahai and Brent Waters, " Ciphertext-Policy Attribute-Based Encryption", IEEE Transaction on Computers,2014

6. Mr. Imran D. Tamboli1, Prof. Ranjana R. Badre, Prof. Rajeshwari M. Goudar, " A Survey-Decentralized Access Control with Anonymous Authenti- cation and Deduplication of Data Stored in Clouds" International Journal Of Engineering And Computer Science Volume 4 ,2015

7. Ms. Deepali C. Ghosalkar," Implementation Idea for Secure Data Deduplication Using Hybrid Cloud Approach " International Journal of Computer Science Trends and Technology (IJCST) – Volume 4, 2016.

8. Z. Wan, J. Liu, and R.H. Deng, "HASBE: A Hierarchical Attribute-Based Solution for Flexible and Scalable Access Control in Cloud Computing," IEEE Trans. Information Forensics and Security, vol. 7, 2, 2012.

9. Taeho Jung, Xiang-Yang Li, Zhiguo Wan and Meng Wan, " Privacy Preserving Cloud Data Access With Multi-Authorities",INFOCOM,2013,Proceeding IEEE,2013.

Vijay Ram Ghorpade
Digitally signed by Vijay Ram Ghorpade
Date: 2023.05.20 15:48:00 +05'30'