|| Volume 4 || Issue 6 || June 2019 || ISSN (Online) 2456-0774
INTERNATIONAL JOURNAL OF ADVANCE SCIENTIFIC RESEARCH
AND ENGINEERING TRENDS

Attribute-Based Storage Supporting Secure De-duplication of Encrypted Data in Cloud using DROP Technique

Miss. Swati Dhokate[1], Prof. N. M. Shivale[2]

PG Student, Department of Computer Engineering, JSPM's BSIOTR, Wagholi, Pune[1]
Assistant Professor, Department of Computer Engineering, JSPM's BSIOTR, Wagholi, Pune[2]
swatidhokte@gmail.com[1], shivalenitin23@gmail.com[2]

Abstract- In public cloud storage system protecting the data and controlling the data access is a challenging issue. Cipher text Policy Attribute-Based Encryption (CP-ABE) has been adopted as a promising technique to provide flexible, fine-grained and secure data access control for cloud storage with honest-but-curious cloud servers. However numerous works have been proposed using CP-ABE scheme, in which the single attribute authority must execute the time-consuming user legitimacy verification and secret key distribution and hence it results in a single-point performance bottleneck when a CP-ABE scheme is adopted in a large-scale cloud storage system. Clients may be stuck in the waiting line for a long stretch to get their mystery keys, which results in low-efficiency of the framework. Even though the multi authority access control plans have been proposed, these plans still cannot conquer the disadvantages of single-point bottleneck and low efficiency; because of the way that each of the authority still autonomously deals with a disjoint characteristic set. In order to overcome this disadvantage, there has been proposed a novel heterogeneous framework to remove the problem of single point performance bottleneck and provide a more efficient access control scheme with an auditing mechanism. This framework employs multiple attribute authorities to share the load of user legitimacy verification. Meanwhile, in this scheme, a CA (Central Authority) is introduced to generate secret keys for legitimacy verified users and each of the authorities in this scheme manages the whole attribute set individually. This system makes performance improvement in key generation and also guarantees security requirement. Still there are some security loopholes in this system such as there is no protocol to verify owner and if the owner is compromised then he/she may put wrong data or information in the data server and users may get wrong data. There is no way to know who has used the data.

Keywords: *ABE, De-duplication, Cloud storage, CP-ABE scheme*

I INTRODUCTION

The most important and popular cloud service is data storage service. Cloud users upload personal or confidential data to the data center of a Cloud Service Provider (CSP) and allow it to maintain these data. Cloud computing schemes presented focus on warehoused data, where the outsourced data is kept unchanged over remote servers In cloud data storage system, users store their data in the cloud and longer possess the data locally. Thus, the correctness and availability of the multiple data files being on the distributed cloud servers must be guaranteed. In existing system, brute-force attack used to avoid multiple copies of dynamic when verifying multiple data copies, the overall system integrity check fails if there are on corrupted copies. This project Rewriting (HAR) algorithm to avoid de encrypted data stored in cloud based on ownership challenge and proxy re-encryption. It integrates cloud data de-duplication with access control. Scheme based on data owner-ship challenge and Proxy Re-Encryption (PRE) to manage encrypted data storage with deduplication.aim to solve the issue of de-duplication in the situation where the data holder is not available or difficult to get involve method using double encryption key for encrypted data stored in cloud. First the data owner provide the secret key to data user then authorized party (AP) send the secret key to data owner. Both AP key and private key generate the encrypted key for encryption. AES algorithm using the encrypt content stored in cloud.

The standard ABE system fails to achieve secure de-duplication 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, to the best of knowledge, existing constructions, for secure de-duplication are not built on attribute-based encryption.

Nevertheless, since ABE and secure de-duplication have been widely applied in cloud computing, it would be desirable to design a cloud storage system possessing both properties. The consider following scenario in the design of an attribute-based storage system supporting secure de-duplication 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. However, endowing such a tag checking ability to the private cloud is not sufficient to achieve de-duplication in the attribute-based storage system which employs CP-ABE for data encryption. In the proposed attributed-based system, the same file could encrypted to different cipher texts associated with different access policies, storing only one cipher text of the file means that users w hose attributes satisfy the access policy of a discarded cipher text (but not that of the stored cipher text) will be denied to access the data that they are entitled to. To overcome this problem, equip the private cloud with another capability named cipher text regeneration.

II LITERATURE SURVEY

In this paper, to solve the recognized critical security supplies for Keys outsourcing. I present cloud Key Bank, the first unified privacy and owner agreement enforced Key management framework. To implement cloud KeyBank, suggest a new Cryptographic embryonic SC-PRE and the consistent concrete SC-PRE arrangement. The security contrast and analysis prove that solution is adequate to support the identified three security requirements which are not be solve in old-style outsourced scenario. From the performance analysis, can see that explanation is not so efficient because it necessitates several seconds to answer a query on a database only 200 keywords. [1]

In this paper, Secured Cloud Maintenance with crypto policies has been widely used in all places, where a data provider outsources his/her encrypted data to a cloud service provider, and can share the data with users possessing specific credentials. However, the standard crypto system does not support secure de-duplication, which is crucial for eliminating duplicate copies of identical data in order to save storage space and network bandwidth. In this system, a new scheme called Advanced Cryptographic Standard (ACS) is introduced with secure de-duplication in a hybrid cloud setting, where a private cloud is responsible for duplicate detection and a public cloud manages the storage. That can be used to confidentially share data with users by specifying access policies rather than sharing decryption keys. 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. [2]

In this system, owner transfers the file with the attributes and access policy, accessing time, then transfers file check for checking whether the file is duplicate or not. On this if file is duplicate then the owner of the file will get proof of possession and if file is original then store it on cloud and once user request for file attribute authority can check the attributes of user then solely user can get key to access the file from cloud. The application will enable us to avoid redundant files in cloud which leads to save of storage space and network bandwidth. Also it improves the security notion as encryption algorithms are used and the misuse of files can be avoided as the sharing of files or data with other users is done by specifying the access policy rather than the decryption keys. [3]

In this system, the new distributed de-duplication systems with file-level and fine-grained block-level data de-duplication, higher reliability in which the data chunks are distributed across HDFS storage, reliable key management in secure de-duplication and the security of tag consistency and integrity were achieved. [4]

III PROBLEM STATEMENT

An inherent drawback of the existing approaches to achieve secure de-duplication is that they cannot satisfy the standard security definition for confidentiality such as semantic security. To solve this problem, a weaker security notion called privacy under chosen-distribution attacks was put forward under the assumption that the input message is sufficiently unpredictable. In this system present Division and Replication of Data in the Cloud for Optimal Performance and Security (DROPS) that judicially fragments user files into pieces and replicates them at strategic locations within the cloud.

IV SYSTEM ARCHITECTURE

An attribute-based storage system supporting secure de-duplication system is built under a hybrid cloud architecture, where a private cloud manipulates the computation and a public cloud manages the storage. Attribute based storage system supporting secure de-duplication 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. The Attribute Authority issues every user a decryption key associated with the set of attributes. The attribute based storage system check the duplication of the file. The duplication is not occur, the file is stored. If the duplication is occurring, the attribute

authority changes the ownership permission. In this system utilizing client accreditation's to check the confirmation of the client. In that cases cloud is available two sort of cloud such private cloud and open cloud. In private cloud store the client accreditation and in the open cloud client information present out. The system has utilized a half and half cloud construction modeling as a part of proposed. In this system have to need to mind the file name in record information duplication and information DE duplication is checked at the square level. On the other hand, client needs to recover his information or download the information record he have to download both of the document from the cloud server this will prompts perform the operation on the same record this abuses the security of the distributed storage. Division and Replication of Data in the Cloud for Optimal Performance and Security (DROPS) that collectively approaches the security and performance issues. In this project, DROPS methodology, divide a file into fragments, and replicate the fragmented data over the cloud nodes. Each of the nodes stores only a single fragment of a particular data file that ensures that even in case of a successful attack, no meaningful information is revealed to the attacker.

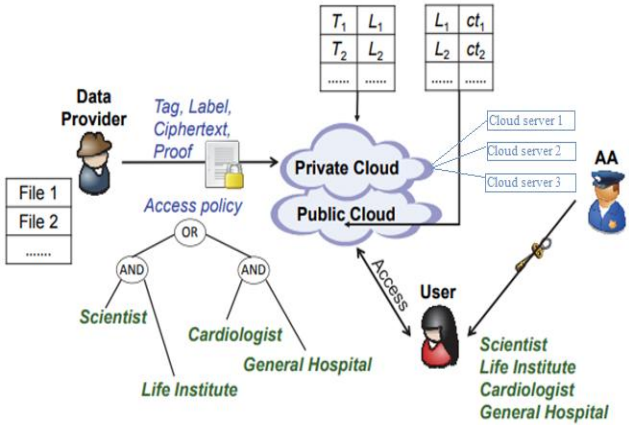

Figure 1: System Architecture

V ALGORITHM

Setup, Encrypt, Key Generation (KeyGen), and Decrypt.

1. Setup(1, λ) → (pars, msk). Taking the security parameter λ as the input, this setup algorithm outputs the public parameter pars and the master private key msk for the system.

2. Encrypt(pars, M, A)→ (skT , CT). Taking the public parameter pars, a message M and an access structure A over the universe of attributes as the input, this encryption algorithm outputs a trapdoor key skT and a tuple CT = (T, L, ct, pf), where T and L are the tag and the label associated with M respectively, ct is the ciphertext which includes the encryption of M as well as the access structure A, and pf is a proof on the relationship of tag T, label L and ciphertext ct. This algorithm is run by the data provider. Both skT and CT are forwarded to the private cloud. Note that skT can not be disclosed to any third party, so it must be sent to the private cloud in a secure manner.

3. KeyGen(pars, msk, A) → skA. Taking the public parameter pars, the master private key msk and an attribute set A as the input, this attribute-based private key generation algorithm generates an attribute based private key skA for the attribute set A.

4. Validity-Test(pars, CT)→ 1/0. Taking the public parameter pars and a tuple CT as the input, this validity testing algorithm parses CT as (T, L, ct, pf), and outputs 1 if pf is a valid proof for (T, L, ct) or 0 otherwise.

5. Equality-Test(pars, (T1, L1, ct1), (T2, L2, ct2)) → 1/0. Taking the public parameter pars and two tuples (T1, L1, ct1) and (T2, L2, ct2) as the input, this equality testing algorithm outputs 1 if both (T1, L1, ct1), (T2, L2, ct2) are generated from the same underlying message or 0 otherwise.

VI RESULTS

In traditional attribute systems, this does not allow to check duplication for the storage. To achieve this task proposed system takes leverage of ABE with de-duplication that utilizes the storage and bandwidth of cloud. In proposed system uses hybrid cloud to achieve ABE in cloud where private cloud is able to detect the duplicate data and public cloud for store the data.

To implement access structure basically leverages the LSSS system (Lewko-Waters algorithm [6]). E.g. Bob is project manager wants to assign one project with attributes "(Java AND (Software Developer OR Tester)" where "Java", "Software Developer" and "Tester" are attributes. In order to access the project report user should have these attributes, e.g. all Software developers and testers in department of Java.

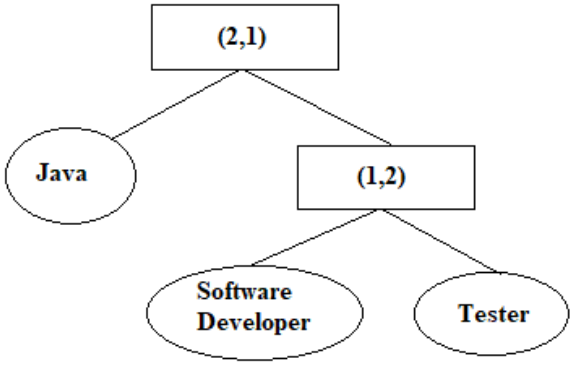

Figure 2: Threshold-gate access tree. Each leaf node is an attribute

From above example, Boolean formula form is (Java Λ (Software Developer V Tester)).

File Part placement Algorithm:

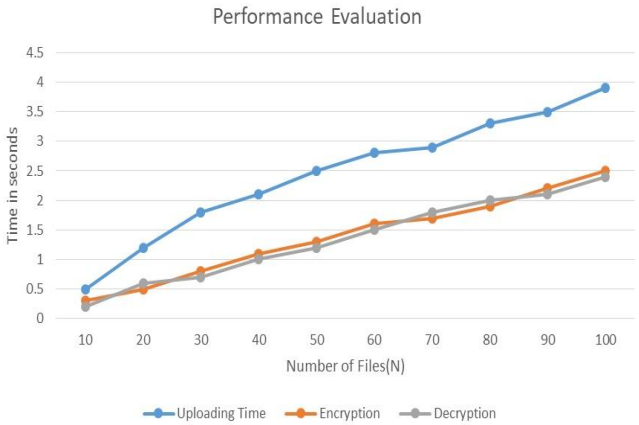

Figure 3: Graphical Form

In performance evaluation, system requires least time for uploading, while uploading file, files splits into parts which takes advantages of division to improve the system's performance and security. While splitting the files into fragment needs to check with existing uploaded files for duplication. When duplication found then users attribute structure should be checked. If attribute matched based on LSSS then matched parts of files are not written on cloud as they referenced to existing matched parts. Hence requires some more time for uploading. Encryption and decryption requires average computation time which increases linearly with number of attributes.

Figure 4: Admin Login

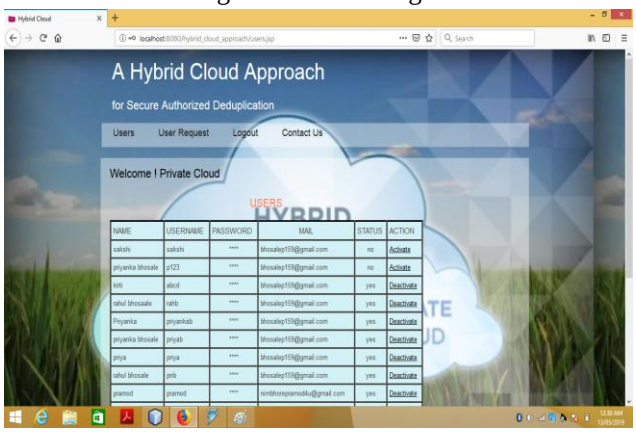

Figure 5: Activate User

Figure 6: Enter Token

Figure 7: Update Record

VII CONCLUSION

In this system proposed framework information DE duplication of record is done approves way and safely. In this system additionally proposed new duplication check system which produces the token for the private document. A proposed routine guarantees the information duplication safely. Future work includes efficient data ownership verification, scheme optimization with hardware acceleration at IoT devices for practical deployment, and development of a flexible solution to support de-duplication and data access controlled by either the data owner or its representative agent.

REFERENCES

[1] S. Kalaivani, "Secure Data Sharing in Cloud Computing using Revocable Storage Identity-Based Encryption"

[2] R. Lavanya, Jayanthi. S, "Advanced Crypto Standard to Secure the De-duplication of Data in Cloud"

[3] Chinmay Patil, Shubham Kasabe, "Attribute based Storage to avoid duplicate files on cloud"

[4] Mr. T. A. MohanaPrakash, "A Secure Access Policies Based Data Deduplication System"

[5] D. Quick, B. Martini, and K. R. Choo, Cloud Storage Forensics. Syngress Publishing / Elsevier, 2014. [Online].

Available: http://www.elsevier.com/books/cloud-storageforensics/ quick/978-0-12- 419970-5
[6] K. R. Choo, J. Domingo-Ferrer, and L. Zhang, "Cloud cryptography: Theory, practiceand future research directions," Future Generation Comp. Syst., vol. 62, pp. 51–53, 2016.
[7] K. R. Choo, M. Herman, M. Iorga, and B. Martini, "Cloud forensics: State-of the-art and fusture directions," Digital Investigation, vol. 18, pp. 77–78, 2016.
Y. Yang, H. Zhu, H. Lu, J.Weng, Y. Zhang, and K. R. Choo, "Cloud based data sharing with fine-grained proxy re-encryption," Pervasive and Mobile Computing, vol. 28, pp. 122–134, 2016.
[8] D. Quick and K. R. Choo, "Google drive: Forensic analysis of data remnants," J. Network and Computer Applications, vol. 40, pp. 179–193, 2014.