International Journal of Recent Trends in Engineering & Research
ISSN(ONLINE) : 2455-1457
IMPACT FACTOR : 3.344

Re-encryption based key management with de-duplication mechanism for cloud

Akshat Vaidya[1], Parth Bhimani[2], Debashish Dwivedi[3],Mayur Phanse[4]
[1,2,3,4] Department Of IT, AISSMS IOIT

Abstract— Data outsourcing to the cloud is profitable for reasons of scalability, economy, and accessibility, but the challenges in security still remain.
As the business organizations move more to the cloud environment the data and therefore the load on the cloud will keep on increasing; therefore an effective storage mechanism is needed so that the data redundancy is reduced. One of the major form of redundant data on the cloud is dublicate data (especially media).
This paper aims to provide ways to improve data security on the cloud and to reduce data redundancy on the cloud by implementing De-duplication.
Keywords— Re-encryption, Attribute-based encryption, Cloud security, Cloud computing security, key management, de-duplication.

I. INTRODUCTION

Cloud computing is growing exponentially per day due to its clear advantages over traditional computing and yet many of the organizations are still reluctant to migrate completely to cloud platform. This is mainly due to the fact that along with wide range of advantages of cloud there are some security issues in the cloud scenario which are yet to be completely addressed. These issues cannot be overlooked because security of the data is the most important attribute while migrating to a third party cloud. This work enhances cloud security by encrypting the data and implementing attribute based encryption. Key management is handled jointly by the cloud provider and cloud administrator. OTP technique is also added in order to verify the authenticity of cloud users and validate them to access the data. The proposed model is highly secure, scalable and efficient. Data redundancy is reduced with the help of de-duplication mechanism. De-duplication is implemented using a secure hash function which checks for duplicate files before uploading any file on the server. Finally an implementation on Apache Tomcat server is used to verify the security and efficiency of the system

The upcoming sections of this paper describe a new approach of key management comprising a data de-duplication mechanism. Section 3 describes the architecture and the working of the proposed system. Finally advantages of the system are stated along with the conclusion in last section.

II. LITERATURE SURVEY

Data storing on the cloud is appropriate for any type of application that requires data to be saved in storage and given access to many users. Clients that approach a cloud provider generally only pay for the amount of storage, related processing, and the amount of network communication used up; they do not bear the cost and maintenance of an in-house solution.
Additionally, the provider offers the advantages of replication and automatic backup to ensure the longevity, safety, and accessibility of the data. A major concern which is not adequately addressed in real is that data is stored in the clear; it may be read and accessed by a cloud admin without the client knowing.

A cloud admin may not be trusted, although the presence of high security obligations, if data security is not further put through technical methods. Additionally the risk is that sensitive data carry the

constant chance of being intercepted by an unauthorized person despite security promised by the provider. Hence, it is useful to use software techniques, like encryption key management, which ensures the confidentiality of cloud data. It is especially crucial to guard sensitive user data such as e-mails, credentials.

III. PROPOSED METHODOLOGY

The Proposed system has enhanced the security of cloud storage with the help of encryption mechanism. The data owner is able to give importance to files before uploading them on the server and only the important files require data owner permissions. The model also has enhanced the security of cloud storage with the help of encryption mechanism. The data owner is able to give importance to files before uploading them on the ons. To reduce data redundancy de-duplication mechanism is included which prevents data owners for uploading duplicate files. The duplication is checked at the file content level and not just on the names of the files i.e. two files with same names but different data can be uploaded but two files with different names but same data cannot be uploaded on the server.

1.Architecture

The proposed system uses star topology with the server being in connection with each node.

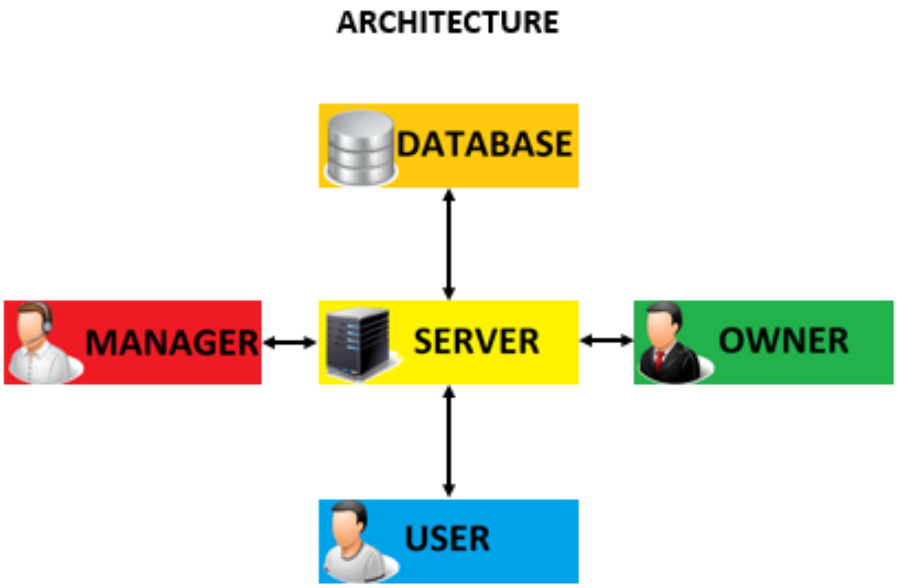

Figure 1 Architecturee

The manager node is added unlike the traditional models to stop unwanted users to register a profile and spam the owner for file access requests.
To be eligible for requesting the access to the uploaded file the user must create a profile which is only functional after the manager verifies and activates it.

2. Re-encryption based key management

The working of the system is as follows:
The owner uploads the file and provides the encryption key with an option to use extra security, the server then encrypts the file with the key provided by the user.
The user makes a profile which is checked and gets activated by the manager.
After activation the user requests for the file to the user.
The increased security is made available by adding certain steps to the current models;

The server waits for the owner to grant the permission for the user to access the uploade file, when the data owner accepts the sever proceeds with the decrytion on the file in the database and provides the user with download link to the requested file.
Simultaneously the server sends the key to the requesting user via his registered e-mail, which ensures data confidentiality and only the user meant to recieve the file has access to it.
The downloded file is only accessable after the user decrypts it with the key recieved from the server.

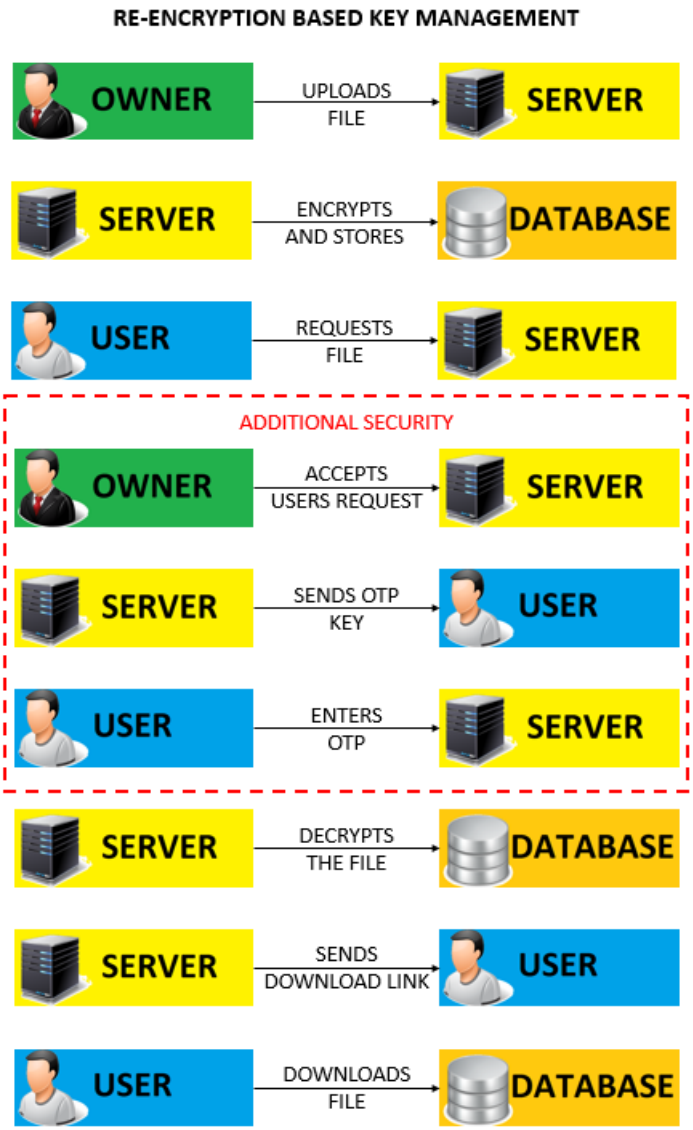

Figure 2 Working

3.De-duplication

De-duplication helps to remove uploading of same files to the server; it automatically rejects all the similar files.
De-duplication uses hashing algorithms to speed up the checking process for the replications of files. It does so by creating a hash digest of every file before storing it in the database. Whenever a new file is uploaded the server creates and checks for matching hash from the previously uploaded files. If the file is already present in the Database there is no need for storing the file again, therefore the file is deleted from memory, thus eventually reducing redundant data and saving memory.

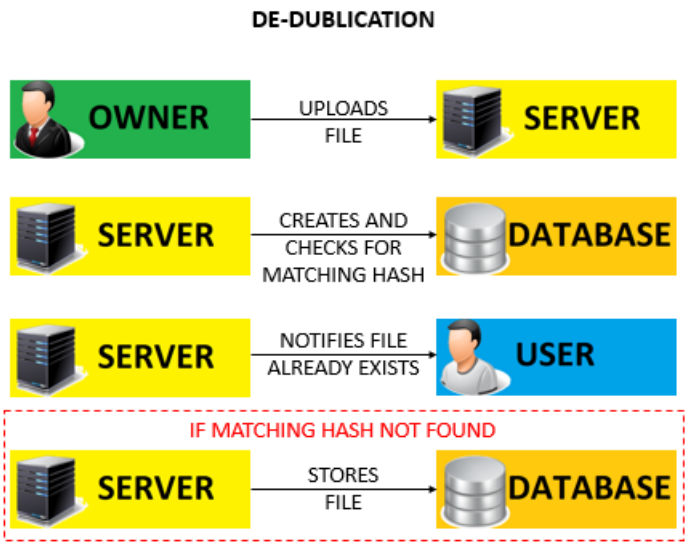

Figure 3 De-dublication mechanism

IV. ADVANTAGES

The proposed system gives following advantages over the existing systems:

1. System provides security to the confidential data of organizations and provides better access control. Due to this unauthenticated users cannot gain access to confidential data. The data is stored on a cloud and only the users with current session key and valid credentials can get access to the data stored on the cloud.
2. The de-duplication mechanism included in this proposed system reduces data redundancy by efficient data storage on the cloud. This mechanism prevents uploading of multiple copies of the same file.

V. CONCLUSION

The proposed model provides better additions to the cloud by efficient data storage and options of better security.

REFERENCES

1. P.K. Tysowski and M.A. Hasan, "Hybrid Attribute-Based Encryp-tion and Re-Encryption for Scalable Mobile Applications in Clouds," Technical Report 13, Centre for Applied Cryptographic Research (CACR), Univ. of Waterloo, 2013.
2. R.L. Rivest, A. Shamir, and L. Adleman, "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems,"Comm. ACM, vol. 26, no. 1, pp. 96-99, Jan. 1983.
3. N. Balasubramanian, A. Balasubramanian, and A. Venkataramani, "Energy Consumption in Mobile Phones: A Measurement Study and Implications for Network Applications," Proc. Ninth ACM SIGCOMM Conf. Internet Measurement Conf. (IMC '09),pp. 280-293, 2009.
4. J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-Policy Attribute-Based Encryption,"Proc. IEEE Symp. Security and Privacy (SP '07),pp. 321-334, 2007.
5. Tassanaviboon and G. Gong, "OAuth and ABE Based Authorization in Semi-Trusted Cloud Computing: Aauth,"Proc. Second Int'l Workshop Data Intensive Computing in the Clouds (DataCloud-SC '11),pp. 41-50, 2011.
6. X. Liang, R. Lu, and X. Lin, "Ciphertext Policy Attribute Based Encryption with Efficient Revocation," Technical Report BBCR, Univ. of Waterloo, 2011.
7. J. Hur and D.K. Noh, "Attribute-Based Access Control with Efficient Revocation in Data Outsourcing Systems,"IEEE Trans. Parallel and Distributed Systems,vol. 22, no. 7, pp. 1214 1221, July 2011.
8. Prof. Rakesh Mohanty, Niharjyoti Sarangi, Sukant Kumar Bishi, "A Secured Cryptographic Hashing Algorithm" VSSUT, Burla, Orissa, India

