Using Downtoken Secure Group Data Sharing On Cloud

Mayuri Shinde[1], Nirmala Sawant[2], Jyoti Rakshe[3], Aishwarya Manjare[4]

[1,2,3,4]*Student, Dept. Of Computer Engineering, PKTC PUNE, Maharashtra, India*

Abstract:- *Nowadays Biggest Problem is that sharing the data over a public cloud storage is Privacy. An important function is data sharing in public cloud storage. The problem solution for secure data transfer on the cloud is to encrypt the data before sharing and using decryption keys decrypting it by the receiver. The key challenge is that to designing such an encryption scheme lies in the efficient management of encryption keys. To avoiding this practical problem, we propose the concept of Key Aggregate Search Encryption (KASE) in which data owner only need to distribute a single trapdoor to a group of users for sharing many of documents and the user only needs to give a trapdoor to the cloud for querying and searching the shared many of documents.*

***Key Words*: Group data sharing, an, traceability, key agreement, Symmetric Balanced Incomplete Block Design (SBIBD).**

1. INTRODUCTION

Group data sharing on the cloud is a very big trend in today's world. For security concern use of cryptographic schemes like encryption and decryption. The problem in that when we want to share a group of files with selected users from a group then we need different encryption keys for different files hence we need to the large database for storing this secure keys and it becomes very complex so in our system use a KASE (Key Aggregate Search Encryption).

In KASE sender share single DT (Download Token) with the number of receivers.DT has receiver's id and a group of files which receiver want. And receivers only need to upload DT on the cloud. Then cloud checks the DT and some public data for file search and sends the result to the receivers.Using DT we can send the group of documents to selected users of a group.

2. EXISTING SYSTEM

Primary User wants to upload a large collection of files to the cloud storage, which is meant every user from group to get access to this files.

Suppose those files contain highly sensitive information that should only be accessed by authorized users, and Secondary User is one of the directors and is thus authorized to view documents. Due to concerns about potential Data leakage in the cloud, Primary User encrypts these documents with different keys and generates keyword cipher texts before uploading to the cloud storage. Primary User then uploads and shares those files with the directors using the sharing functionality of the cloud storage.

In order for Secondary User to view the files, need different decryption keys for different files.When files are sufficiently large, the key distribution and storage, as well as the trapdoor generation, may become too expensive for the client-side device, which basically defies the purpose of using cloud storage.

3. PROBLEM STATEMENT

In our system, User Upload Encrypted Files on Cloud Server with a generated Download Token to share the files securely from one user to another. When user share file to another user then a generate Download token and send to another user for the File Downloading. Download token also uses for search files on the cloud.

4. PROPOSED SYSTEM

In this section, we present our scheme in detail. Benefiting from the KASE the presented scheme the presented scheme can be applied for group data sharing with low communication and computation complexities. Our scheme can be divided into five parts: initialization, key generation, fault detection, file generation and key update, file access and traceability.

5. ARCHITECTURE

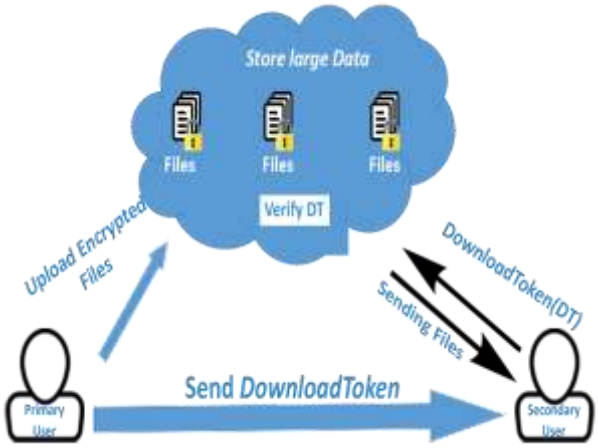

Fig -1: System Architecture

In Fig 1. Primary user uploads encrypted files on cloud server with a generated Download Token to share the files securely from one user to another number of secondary users.The primary user generates Download token and sends to the secondary user. The secondary user uploads download token on the cloud. At cloud verify DT and according to that sends to the secondary user for the File Downloading.

6. FLOW DIAGRAM

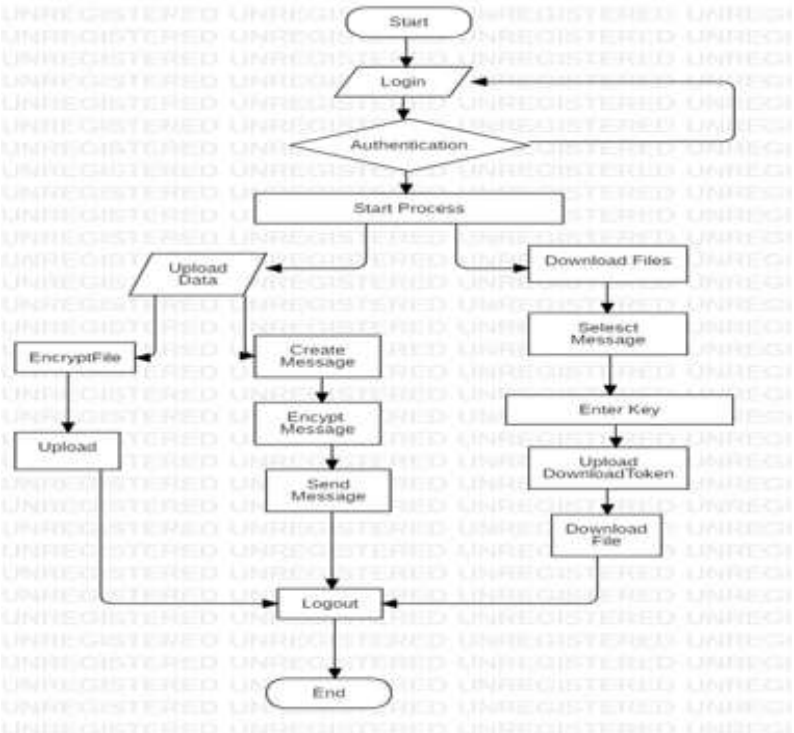

Fig -2: Flow Diagram

7. ADVANTAGES

Confidentiality: The confidentiality of the outsourced Data Is Preserved.

Dynamic changes: Arbitrary number of users and dynamic changes are supported.

Low energy: consumption and resource sharing.

Efficient: Client authorizes access to his/her data for many clients with secure and efficient at the same time.

8. FUTURE SCOPE

In cloud storage, the number of ciphertexts usually grows rapidly. So we have to reserve enough cipher text classes for the future extension. Otherwise, we need to expand the public-key. Although the parameter can be downloaded with cipher texts. In future we can add this concept on Android application hence it will be more efficient as compared to proposed system.

9. CONCLUSION

We present a secure and fault-tolerant key agreement for group data sharing in a cloud storage scheme. Our scheme can support the traceability of user identity in an anonymous environment. In terms of dynamic changes of the group member, taking advantage of the key agreement and efficient access control, the computational complexity and communication complexity for updating the common conference key and the encrypted data are relatively low.

REFERENCES

[1] Zhou, V. Varadharajan, and M. Hitchens, Cryptographic role-based access control for secure cloud data storage systems, Information Forensics and Security IEEE Transactions on, vol. 10, no.11, pp. 23812395, 2015.

[2] Manghui Tu, Peng Li, Ling Yen, Bhavani the raising- ham, and Latifur Khan. Data objects replication in the data grid. IEEE TRANSACTIONS ON DEPENDABLE AND SECURE COMPUTING, 5(4), 2008.

[3] William Stallings.Cryptography and Network Security: Principles and Practices, 4th edition. Pearson Education, Inc., 2006.

[4] S. A. Weil, S. A. Brandt, E. L. Miller, and C. Maltzahn. Crush Con-trolled, scalable, decentralized placement of replicated data. In Proc. of ACM/IEEE Con. on Supercomputing, 2006.

[5] C. Asmuth and J. Bloom. A modular approach to key safeguarding.IEEE Transactions on Information Theory, 29:208210, 1983.

[6] E. Dawson and D. Donovan.The breadth of Shamir as secret-sharing scheme.Computers and Security, 13:69 78, 1994.

[7] S. Lakshmanan, M. Ahamad, and H. Venkateswaran. Responsive security for stored data.IEEE Transactions on Parallel and Distributed Systems, 14(9), 2003.

[8] T. Rabin and M. Ben-Or.Veri able secret sharing and multiparty protocols with honest majority (extended abstract) Communication of The ACM, 1989.