Efficient Secure Deduplication in the Cloud Using User-Defined Access Control

RAJASEKHAR BANDI [1], M SRAVANI [2]

[1] PG Scholar, Dept of CSE, Gokula Krishna College of Engineering, Sullurpeta, AP, India.

[2] Assistant Professor, Dept of CSE, Gokula Krishna College of Engineering, Sullurpeta, AP, India.

ABSTRACT_ Distributed storage as one of the main administrations of distributed computing which fundamentally works with cloud clients to re-appropriate their information to the cloud for capacity and offer them with approved clients. Secure deduplication has received a lot of attention in cloud storage because it can reduce communication overhead and storage space by eliminating redundancy in encrypted data. In terms of privacy and security, numerous current secure deduplication schemes typically concentrate on achieving the following properties: confidentiality of the data, consistency of the tags, management of access, and resistance to brute-force attacks However, as far as we are aware, none of them can simultaneously fulfill these four requirements. To defeat this deficiency, in this article, we propose a proficient secure deduplication plot that upholds client characterized admittance control. Particularly, our plan maximizes the elimination of duplicates without compromising cloud users' privacy or security by allowing only the cloud service provider to authorize data access on behalf of owners. Our authorized secure deduplication scheme, according to a comprehensive security analysis, prevents brute-force attacks while maintaining data confidentiality and tag consistency. In addition, extensive simulations demonstrate that our strategy outperforms the competing ones in terms of the efficiency of deduplication and the overheads associated with computation, communication, and storage.

1.INTRODUCTION

With the rapid growth of data volumes, there are increasing demands for safe places to store private data. An effective solution to this issue is to outsource big data to the cloud [1, 2]. Duplicate data still eats up a lot of storage space and network bandwidth and complicates data management, despite all the benefits of cloud computing [3]. The cloud storage provider is able to save storage space by storing only a single copy of the data that is owned by multiple owners thanks to the deduplication process [5], which is a process that identifies the same data by data similarity. However, there are still issues with dynamic ownership management and access control with the current deduplication schemes. First and foremost, many users encrypt data prior to uploading it to cloud storage to safeguard their privacy. Deduplication will be hampered if the same data is encrypted using different keys because the encryption key is generated at random. Some deduplication schemes suggest that the owners of the same file share the same encryption key to solve this issue [6–14]. However, the majority of them do not take into account the frequent dynamic ownership shifts in cloud storage services [15]. The cloud clients ought to be renounced from the legitimate possession list once they demand the distributed storage supplier for information cancellation/change. Second, numerous schemes [4], [15], and [19] were proposed to address dynamic ownership management by utilizing Authority Party (AP) or The Public Cloud Provider (P ub CSP) as trusted or semi-trusted third parties to perform proxy reencryption work. From one viewpoint, it very well may be challenging to execute a confided in outsider in commonsense applications [20]. On the other hand, when a third party conspires with unauthorized users, some schemes cannot withstand collusion attacks. In this paper, we propose a clever plan, which focuses on effectively taking care of the issue of deduplication with regular cloud client disavowal and new cloud client participating in distributed computing. Specifically, not the same as existing information deduplication strategies, which utilize either trusted/semi-believed outsider to do intermediary re-encryption work, our proposed plot plans a half breed cloud engineering, which incorporates a public cloud and further presents a confidential cloud. In our scheme's implementations, the introduced private cloud acts as both a data owner and a proxy to 1) manage the dynamic ownership when the real data owner is offline or revokes ownership, and 2) control access to outsourced data through re-encryption techniques. Moreover, we propose to improve our plan concerning productivity by 1) guaranteeing that the information proprietor performs encryption just when he/she is the underlying uploader; 2) introducing an entrance control procedure that confirms the legitimacy of the information clients before they download information; 3) requiring the cloud user to be on the ownership list before the public cloud server can send ciphertext to them As a result, the expense of extensive communication will decrease.

2. LITERATURE SURVEY

2.1) DaSCE: Data Security for Cloud Environment with Semi-Trusted Third Party

AUTHORS: Ali, M., Malik, S. and Khan, S., Off-site information capacity is a use of cloud that assuages the clients from zeroing in on information capacity framework. However, there are serious security concerns associated with outsourcing data to a third-party administrative control. Information spillage might happen because of assaults by different clients and machines in the cloud. Discount of information by cloud specialist organization is one more issue that is looked in the cloud climate. Therefore, elevated degree of safety efforts is required. Data Security for Cloud Environment with Semi-Trusted Third Party (DaSCE) is a data security system we propose in this paper that offers file assured deletion, key management, and access control. To manage the keys, the DaSCE employs Shamir's (k, n) threshold scheme, in which k shares out of n are required to generate the key. We use multiple key managers, each of which stores a single key share. Multiple key managers keep the cryptographic keys safe from a single point of failure. We (a) use the Satisfiability Modulo Theories Library (SMT-Lib) and the Z3 solver to verify the operation of DaSCE, (b) formally model and analyze the working of DaSCE using High Level Petri nets (HLPN), and (c) evaluate its performance based on the amount of time consumed by various operations. Key management, access control, and file assured deletion are all features of DaSCE that can be used effectively to protect outsourced data, as demonstrated by the findings.

2. 2) Control Cloud Data Access Privilege and Anonymity With Fully Anonymous Attribute-Based Encryption

AUTHORS: Jung, T., Li, X. Y., Wan, Z. and Wan, M

Although some cloud servers store data, which raises a number of privacy concerns, cloud computing is a revolutionary computing paradigm that enables flexible, on-demand, and low-cost resource utilization. To protect cloud storage, a number of approaches based on attribute-based encryption have been proposed. However, identity privacy and privilege control receive less attention than data content privacy and access control in most projects. AnonyControl, a semi-anonymous privilege control method, is presented in this paper to address the data privacy and user identity privacy concerns of existing access control methods. To prevent identity leaks, AnonyControl decentralizes authority, resulting in semi-anonymity. In addition, it extends file access control to privilege control, making it possible to fine-tune privilege management for all cloud data operations. Then, we present the AnonyControlF, which achieves complete anonymity and completely prevents identity leakage. Our performance evaluation demonstrates the viability of our schemes, and our security analysis demonstrates that, under the DBDH assumption, both AnonyControl and AnonyControl-F are secure.

3.PROPOSED SYSTEM

To overcome this challenge, in this paper, we design an efficient secure cross-user deduplication scheme with user-defined access control

In our authorized deduplication system, the proposed scheme should ensure that outsourced encrypted data can be available not only to the data owner but also to all authorized users selected by this data owner. On the contrary, unauthorized users cannot access the content of encrypted data outsourced by users of interest

3.1 IMPLEMENTATION

Data Provider

In this module, the data provider uploads their encrypted data in the Cloud server. For the security purpose the data owner encrypts the data file and then store in the server. The Data owner can have capable of manipulating the encrypted data file and performs the following operations: Register and Login, Upload Data, View Uploaded Files, View Secret Key Generated, Update cipher text.

Cloud Service Provider

The **Cloud** server manages which is to provide data storage service for the Data Owners. Data owners encrypt their data files and store them in the Server for sharing with data consumers. To access the shared data files, data consumers download encrypted data files of their interest from the Server and then Server will decrypt them. The server will generate the aggregate key if the end user requests for file authorization to access and performs the following operations such as Login, View and Authorize Users, View and Authorize Owners, View Files, View All Search Transactions, View All File Transactions, View All Top Searched, View Attackers, Search Requests, View Time Delay, View Throughput. Login, View Storage Server Files, View Secret Key, View End Users, View Data Owners, View Transactions, Un Revoke User, View De Dup Files, View Attackers, View Results, View Time Delay Results, View Throughput Results.

User

In this module, the user can only access the data file with the secret key. The user can search the file for a specified keyword. The data which matches for a particular keyword will be indexed in the cloud server and then response to the end user and performing the following operations Register and Login, Request Secret Key, Find Secret Key, Download.

KGC – responsible for Login, Generate secret key, View End Users, View Attackers.

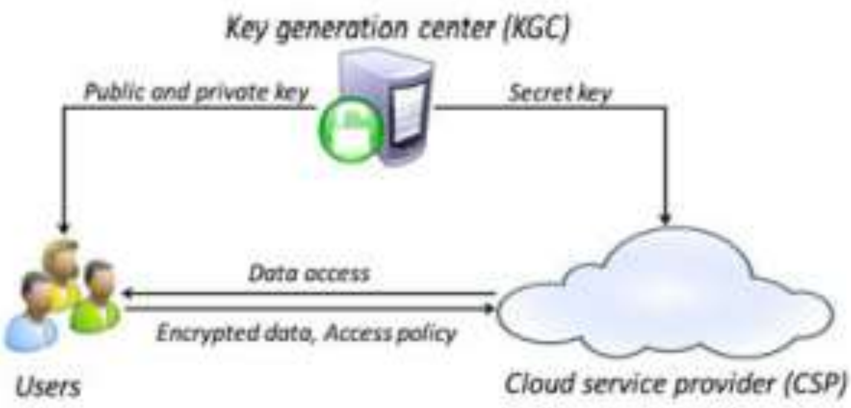
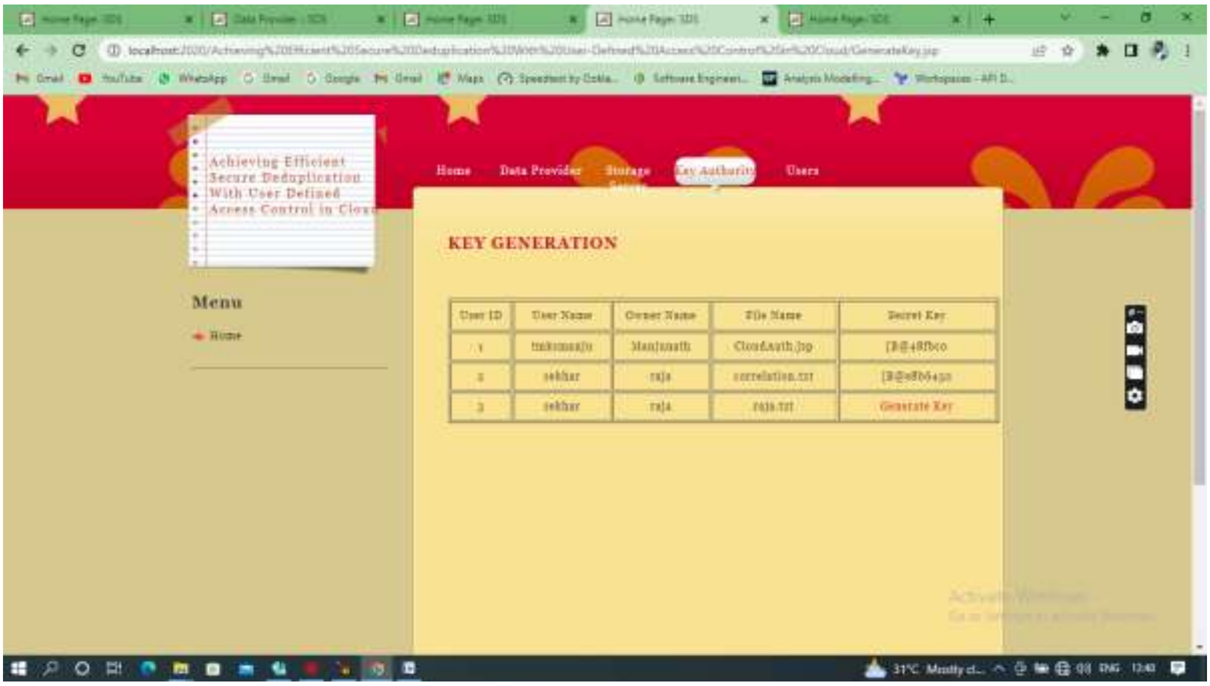

Key generation center (KGC)
Public and private key
Secret key
Data access
Encrypted data, Access policy
Users
Cloud service provider (CSP)

Fig 1: Architecture

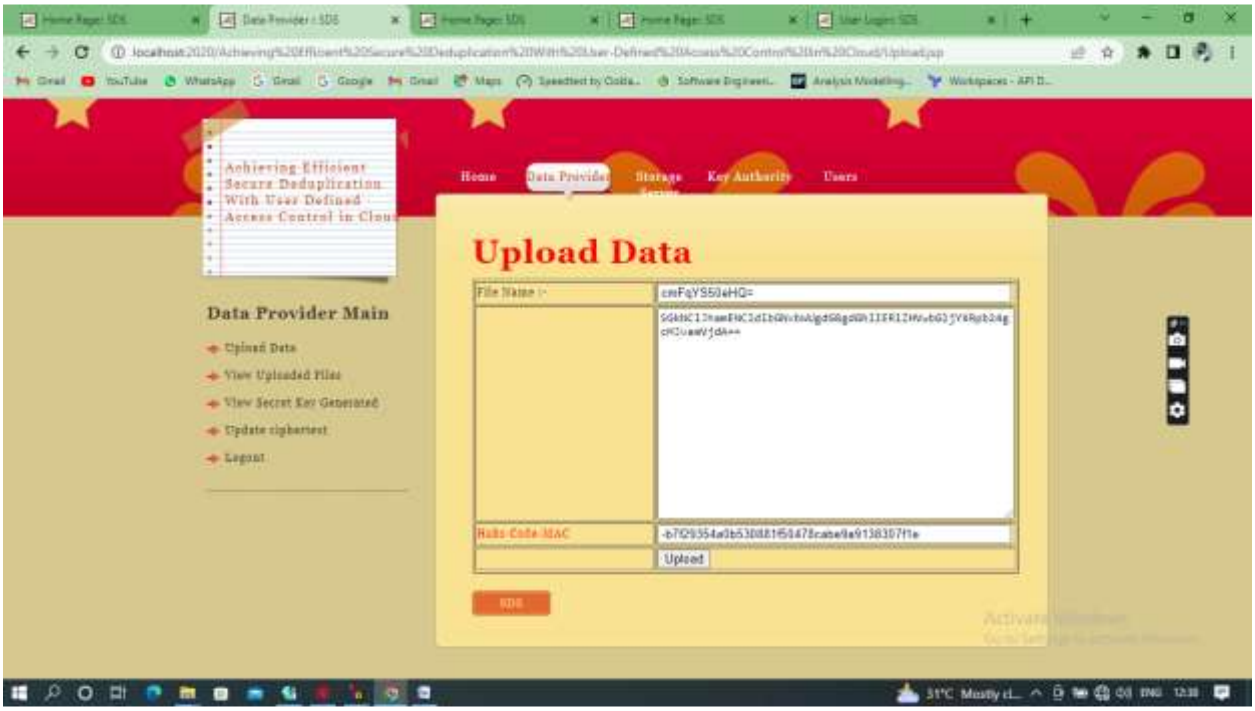
4.RESULTS AND DISCUSSION


Achieving Efficient Secure Deduplication With User Defined Access Control in Cloud
Home
Data Provider
Storage Server
Key Authority
Users
Menu
Home

KEY GENERATION

User ID	User Name	Owner Name	File Name	Secret Key
1	[illegible]	Manjunath	CloudAuth.jsp	[B@48fbc0
2	sekhar	raja	correlation.txt	[B@e8b64a3
3	sekhar	raja	raja.txt	Generate Key

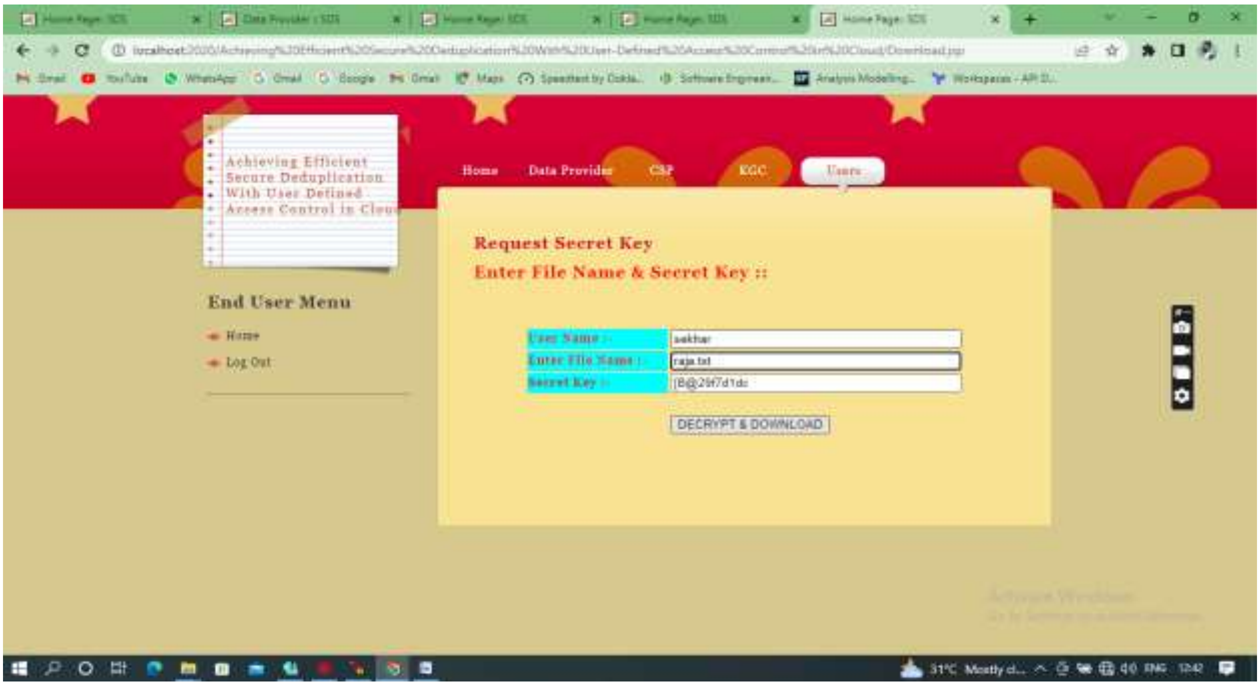
Fig 2:Generate key for user request



Upload Data

File Name :-	cmFqYS50eHQ=
	SGkhC1JhamFNC1dIbGNvbWUgdG8gdGhlIEFRlZHVubG1jYXRpb24g cHJvamVjdA==
Hash Code MAC	-b7f29354a0b530881f50478cabe9a9138307f1e
	Upload

Fig 3: Uploading file to cloud


Achieving Efficient Secure Deduplication With User Defined Access Control in Cloud
Home
Data Provider
CSP
KGC
Users
End User Menu
Home
Log Out

Request Secret Key

Enter File Name & Secret Key ::

User Name ::	sekhar
Enter File Name ::	raja.txt
Secret Key ::	[B@29f7d1dc

DECRYPT & DOWNLOAD

Fig 4: user downloading the file

5. CONCLUSION

In this paper, we present a user-defined access control efficient secure deduplication scheme. In particular, our method does not require the use of a hybrid cloud architecture or the addition of a second authorised server to achieve the authorised deduplication. Only the CSP in our system is capable of managing access rights on behalf of data owners without jeopardising data privacy. Also included in our plan is the Bloom filter, which effectively completes the duplicate check.According to thorough security analyses, our scheme can simultaneously achieve data confidentiality, access control, tag consistency, and resistance to brute-force attacks. Additionally, thorough performance analyses of file-level and chunk-level deduplication demonstrate the effectiveness of our plan in terms of deduplication efficiency, computational cost, communication overhead, and storage cost.

REFERENCES

[1] Joseph A Akinyele, Christina Garman, Ian Miers, Matthew W Pagano, Michael Rushanan, Matthew Green, and Aviel D Rubin. Charm: a framework for rapidly prototyping cryptosystems. Journal of Cryptographic Engineering, 3(2):111–128, 2013.

[2] IttaiAnati, Shay Gueron, Simon Johnson, and Vincent Scarlata. Innovative technology for cpu based attestation and sealing. In Workshop on hardware and architectural support for security and privacy (HASP), volume 13, page 7. ACM New York, NY, USA, 2013.

[3] Alexandros Bakas and AntonisMichalas. Modern family: A revocable hybrid encryption scheme based on attribute-based encryption, symmetric searchable encryption and SGX. In Secure Comm 2019, pages 472–486, 2019.

[4] Amos Beimel. Secure schemes for secret sharing and key distribution. PhD thesis, PhD thesis, Israel Institute of Technology, Technion, Haifa, Israel, 1996.

[5] John Bethencourt,AmitSahai,andBrentWaters. Ciphertext-policy attribute-based encryption. In S&P 2007, pages 321–334. IEEE, 2007.

[6] Victor Costan and Srinivas Devadas. Intel sgx explained. IACR Cryptology ePrint Archive, 2016(086):1–118, 2016.

[7] Ben Fisch, DhinakaranVinayagamurthy, Dan Boneh, and Sergey Gorbunov. IRON: functional encryption using intel SGX. In Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, CCS 2017, pages 765–782, 2017.

[8] Eiichiro Fujisaki and Tatsuaki Okamoto. Secure integration of asymmetric and symmetric encryption schemes. In Advances in Cryptology-CRYPTO 1999, pages 537–554. Springer, 1999.

[9] Vipul Goyal, Omkant Pandey, Amit Sahai, and Brent Waters. Attribute-based encryption for fine-grained access control of encrypted data. In ACM CCS 2006, pages 89–98. ACM, 2006.

[10] Jinguang Han, Willy Susilo, Yi Mu, Jianying Zhou, and Man Ho Allen Au. Improving privacy and security in decentralized ciphertext-policy attribute-based encryption. IEEE transactions on information forensics and security, 10(3):665–678, 2015.