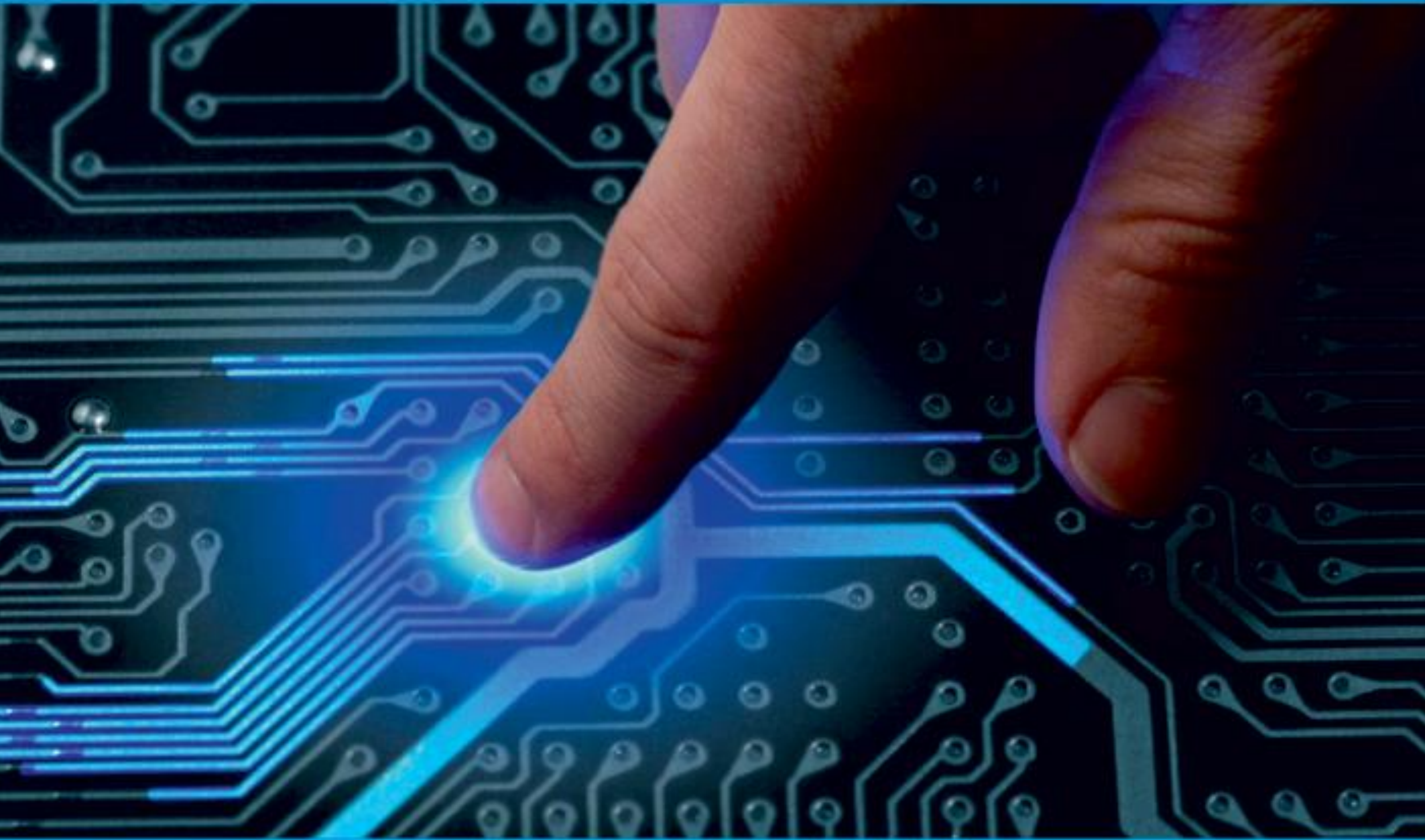
IJIRCCE

e-ISSN: 2320-9801 | p-ISSN: 2320-9798



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

Volume 12, Issue 5, May 2024

INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA

Impact Factor: 8.379

International Journal of Innovative Research in Computer and Communication Engineering

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 5, May 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1205216 |

Efficient Key De Duplication Using Identity Based Broadcast Encryption

Dr. D J Samatha Naidu, P. Bhargavi

Department of MCA, Annamacharya PG College of Computer Studies, Rajampet, Andhra Pradesh, India

Department of MCA, Annamacharya PG College of Computer Studies, Rajampet, Andhra Pradesh, India

ABSTRACT: Deduplication, which can save storage cost by enabling us to store only one copy of identical data, becomes unprecedentedly significant with the dramatic increase in data stored in the cloud.To protect the confidentiality of sensitive data while supporting deduplication, the convergent encryption technique has been proposed to encrypt the data before outsourcing. To better protect data security, this paper makes the first attempt to formally address the problem of authorized data deduplication. The user is only allowed to perform the duplicate check for files marked with the corresponding privileges. A novel client-side deduplication protocol named KeyD without such an independent key management server by utilizing the identity-based broadcast encryption (IBBE) technique Reduce the storage size of the tags for integrity check. To enhance the security of deduplication and protect the data confidentiality.

KEYWORDS: Duplication, Efficient

I. INTRODUCTION

The stored data is growing intensely with the advent of the era of Big Data. We need to constantly increase the storage devices if we continue using the traditional storage way. Alternatively, more and more users are prone to outsource their storage to cloud such as Amazon Web Services (AWS) for economic savings. The ever-increasing data and users, coupled with multiple backup and other factors, result in more and more duplication of files or blocks in the cloud. In order to improve the storage efficiency in the payas-you-go model , deduplication operation is adopted for eliminating duplicate copies of redundant data on the cloud side. Consider an example that m users outsource the same data copies1 of n TB to the CSP. With data deduplication, only one copy is actually stored in the cloud, and the subsequent instances are referenced back to the saved copy for reducing storage roughly from Mn to n TB. However, in order to protect the safety of the outsourced data, they are usually encrypted by their owners before outsourced to the CSP. Then it comes the problem, how can the CSP perform deduplication when these same data copies are encrypted into different ciphertexts by different users' Convergent encryption (CE), which encrypts a data copy with a convergent key derived by computing the cryptographic hash value of the content of the data copy itself and thus can produce identical ciphertext from identical plaintext, brings the hope to realize deduplication while ensuring data confidentiality. This property of convergent encryption allows the CSP to perform deduplication on encrypted data. In particular, users encrypt their data copies using corresponding convergent keys and outsource encrypted data to the CSP They just need to keep convergent keys locally so that they can later restore the data. However, the number of convergent keys increases linearly with the number of data copies since a data copy corresponds to a convergent key. As we all know, in practical file storage systems such as Google File System GFS and Hadoop Distribute File System HDFS data files are usually divided into fine-grained blocks to facilitate deduplicati management,which makes the convergent key storage even more serious. Suppose the n TB data in the above example is divided into blocks of size 4 KB each, and that each convergent key is the hash value of SHA-256.

II. LITERATURE SURVEY

[1] Reclaiming Space from Duplicate Files in a Server less Distributed File System
AUTHORS: J.R. Douceur, A. Adya, W.J. Bolosky, D. Simon, and M. Theimer.
The Far site distributed file system provides availability by replicating each file onto multiple desktop computers. Since this replication consumes significant storage space, it is important to reclaim used space where possible. Measurement of over 500 desktop file systems shows that nearly half of all consumed space is occupied duplicate files. We present a mechanism to eclaim space from this incidental duplication to make it available for controlled file replication. Our mechanism includes: convergent encryption, which enables duplicate files to be coalesced into the space of a single file, even if the files are encrypted with different users' keys; and SALAD, a Self-Arranging Lossy Associative

Database for aggregating file content and location information in a www.jespublication.com decentralized, scalable, fault-tolerant manner. Large-scale simulation experiments show that the duplicate-file coalescing 7 SRS Secure key deduplication with identity based on broadcast encryption system is scalable, highly effective, and fault-tolerant.

[2] Secure Deduplication with Efficient and Reliable Convergent Key Management
AUTHORS: J. Li, X. Chen, M. Li, J. Li, P.P.C. Lee, and W. Lou
Data deduplication is a technique for laminating duplicate copies of data, and has been widely used in cloud storage to reduce storage space and upload bandwidth. Promising as it is, an arising challenge is to perform secure deduplication in cloud storage. Although convergent encryption has been extensively adopted for secure deduplication, a critical issue of making convergent encryption practical is to efficiently and reliably manage a huge number of convergent keys. This paper makes the first attempt to formally address the problem of achieving efficient and reliable key management in secure deduplication. We first introduce a baseline approach in which each user holds an independent master key for encrypting the convergent keys and outsourcing them to the cloud. However, such a baseline key management scheme generates an enormous number of keys with the increasing number of users and requires users to dedicatedly protect the master keys. To this end we propose De key a new construction in which users do not need to manage any keys on their own but instead securely distribute the convergent key shares across multiple servers. Security analysis demonstrates that De key is secure in terms of the definitions specified in the proposed security model. As a proof of concept, we implement De key using the Ramp secret sharing scheme and demonstrate that De key incurs limited overhead in realistic environments.

[3] Multiple Ramp Schemes AUTHORS: A.D. Santis and B. Masucci A (t,k,n,S)
Ramp scheme is a protocol to distribute a secret s chosen in S among a set P of n participants in such a way that: sets of participants of cardinality greater than or equal to k can reconstruct the secret s; sets of participants of cardinality less than or equal to t have no information on s, whereas sets of participants of cardinality greater than t and less than k might have "some" information on s. In this correspondence we analyze multiple ramp 8 SRS Secure key deduplication with identity based on broadcast encryption schemes, which are protocols to share many secrets among a set P of participants, using different ramp schemes. In particular, we prove a tight lower bound on the size of the www.jespublication.com shares held by each participant and on the dealer's randomness in multiple ramp schemes.

III. EXISTING ALGORITHMS

Identity-based broadcast encryption (IBBE) is a cryptographic scheme that allows a sender to broadcast a message to a specific set of recipients identified by their identities. In the context of efficient key deduplication using IBBE, the goal is to minimize the storage overhead and computational cost associated with managing multiple keys for the same recipient. One existing algorithm that can be used for efficient key deduplication in IBBE is the Boneh-Franklin Identity-Based Encryption (BF-IBE) scheme. In BF-IBE, each user is identified by a unique identity string, and a master key generator generates private keys for each user based on their identities. However, in the basic form of BF-IBE, there is no inherent mechanism for key deduplication. To achieve efficient key deduplication in IBBE, one approach is to extend existing IBBE schemes like BF-IBE with additional mechanisms for key management.

IV. SYSTEM ARCHITECTURE

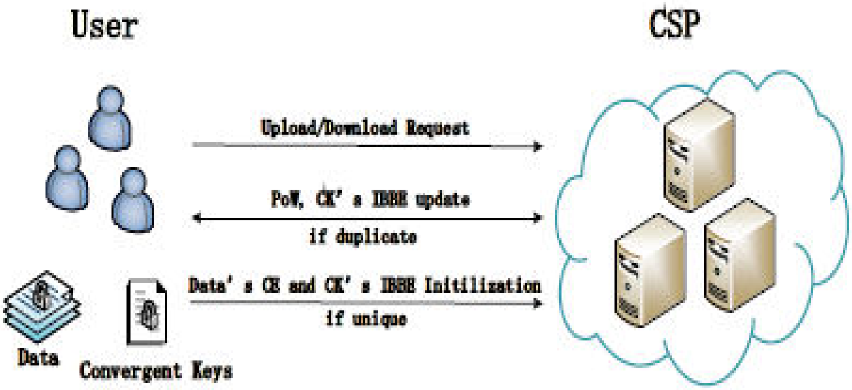


Fig: System Architecture

V. PROPOSED ALGORITHMS

Step-1: Setup phase

The authority generates system parameters:

- Master secret Key
- Public parameters

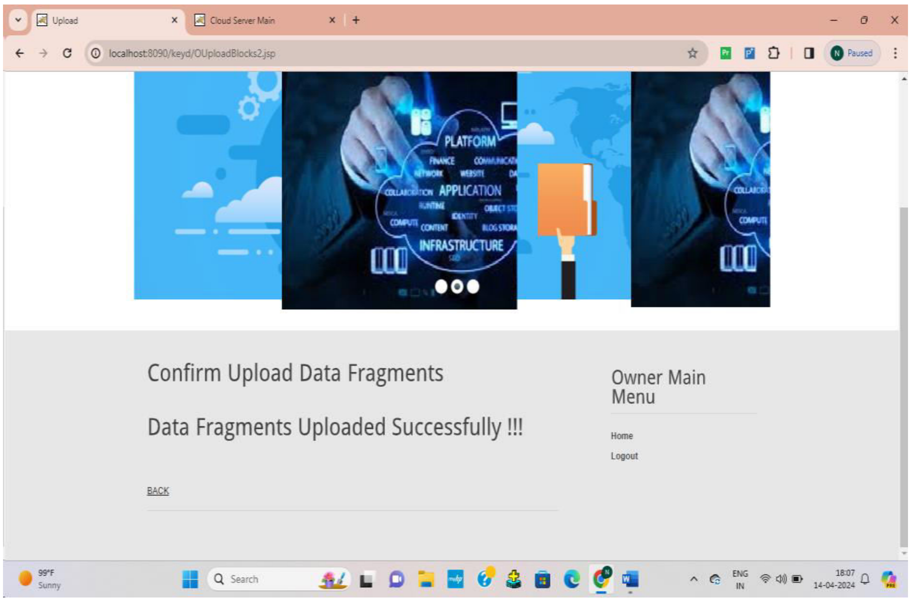
Step-2: Identity Registration

- Each user registers their identity with the authority.
- The authority generates a private key for each identity using the master secret key

Step-3: Key distribution

- When a sender wants to broadcast a message, they specify the identities of the intended receipents
- The authority retrieves the decryption keys corresponding to the requested identities.
- To ensure key de-duplication:
- The authority maintains a mapping of identities to decryption keys.
- If multiple users share the same identity, the authority sends only one copy of the decryption key to all users with that identity.

Data successful page



Confirm Upload Data Fragments

Data Fragments Uploaded Successfully !!!

VI. CONCLUSION

In this paper, we propose a secure client-side deduplication scheme to effectively manage convergent keys. Data deduplication in our design is achieved by interactions between data owners and the Cloud Service Provider (CSP), without participation of other trusted third parties or Key Management Cloud Service Providers. The security analysis shows that our ensures the confidentiality of data and security of convergent keys, and well protects the user ownership privacy at the same time. Experimental results demonstrate that the security of our scheme is not at the expense of the performance..

REFERENCES

[1] Amazon, "Aws global infrastructure," in https://aws.amazon.com/ aboutaws/global-infrastructure/, 2017.
[2] C. Metz, "Facebook tackles (really) big data with project prism," in https://www.wired.com/2012/08/facebook-prism/, 2012.
[3] K. V. SHVACHKO, Y. Aahlad, J. Sundar, and P. Jeliazkov, "Geographically-distributed file system using coordinated namespace replication," in https://www.google.com/patents/WO2015153045A1?cl=zh, 2014.
[4] C. Liao, A. Squicciarini, and L. Dan, "Last-hdfs: Location-aware storage technique for hadoop distributed file system," in IEEE International Conference on Cloud Computing (CLOUD), 2016.
[5] N. Paladi and A. Michalas, "“one of our hosts in another country”: Challenges of data geolocation in cloud storage," in International Conference on Wireless Communications, Vehicular Technology, Information Theory and Aerospace & Electronic Systems (VITAE), 2014, pp. 1–6.
[6] Z. N. Peterson, M. Gondree, and R. Beverly, "A position paper on data sovereignty: The importance of geolocating data in the cloud." in HotCloud, 2011.
[7] A. Squicciarini, D. Lin, S. Sundareswaran, and J. Li, "Policy driven node selection in mapreduce," in 10th International Conference on Security and Privacy in Communication Networks (SecureComm), 2014.

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

9940 572 462 6381 907 438 ijircce@gmail.com

www.ijircce.com

Scan to save the contact details