Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, Issue 6, June : 2023

SECURE DATA TRANSACTION IN CLOUD USING BLOOM FILTERINGTECHNIQUE

K BHAVYA SINDHU[1], B GANESH[2]

[1]PG Scholar, Department of Computer Science and Engineering, Avanthi Institute of Engineering and Technology, Visakhapatnam, Ap, India
[2]Assistant Professor, Department of Computer Science and Engineering, Avanthi Instituteof Engineering and Technology, Visakhapatnam, Ap, India

Abstract—
With the rapid growth of cloud storage, a growing number of data owners are opting to outsource their data to a cloud server, which can significantly reduce local storage overhead. Because different cloud service providers provide varied levels of data storage service, such as security, dependability, access speed, and costs, cloud data transfer has become a must-have for data owners looking to switch cloud service providers. As a result, data owners' key issue is how to securely migrate data from one cloud to another while also permanently deleting the transferred data from the original cloud. In this study, we propose a new counting Bloom filter-based approach to overcome this problem. Not only can the suggested approach provide secure data transport, but it can also ensure permanent data destruction. Furthermore, the suggested approach can meet public verifiability requirements without the use of a trusted third party. Finally, we provide a simulation implementation to demonstrate our proposal's feasibility andefficiency.
Index Terms— TOP-K, MULTICLOUD, SEMANTIC SEARCH, BLOOM FILTER

I Introduction

Cloud computing is the fusion and evolution of parallel computing, distributed computing, and grid computing as a new computer paradigm. Cloud storage is one of the most appealing cloud computing services, as it may provide users with convenient data storage and business access services by bringing together a large number of dispersed storage devices in a network. Users can outsource their data to a cloud server using cloud storage, which can significantly minimize local hardware/software overhead and human resource investments. Cloud storage has become widely used ineveryday life and at work as a result of its appealing benefits. As a result, more and more resource-constrained customers, such as individuals and businesses, are opting for cloud storage services. Despite its many benefits, cloud storage is bound to introduce new security issues as a result of the separation of outsourced data ownership and management, such as data confidentiality, data integrity, and data availability. data erasure These issues, particularlythose related to data destruction, could stymie public adoption of cloud storage if not addressed properly. Data deletion, as the final stepof the data life cycle, directly decides whether the data life cycle can be brought to a satisfactory conclusion, which is critical for data security and privacy preservation. Data deletion, on the other hand, receives far less attention than data integrity, which has been well researched and solved. Although some verified deletion strategies for outsourced data in the cloud computing environment have been developed, there are still certain issues and concerns that need to be addressed immediately. Cloudsfer, an outsourced data transfer tool,has been built to ensure secure data migration by applying cryptographic algorithms to prevent data privacy disclosure throughout the transfer process. However, processing cloud data migration and deletion still has some security issues.To begin, the cloud server may only migrate a portion of the data to save network traffic, or it may transfer all of the data. could deliver unrelated data in order to deceive the data owner Second, certain data blocks may be lost during the transmission process due to network instability. Meanwhile, the adversary may destroy the data blocks that have been sent. As a result, throughout the migration process, the transferred data may become contaminated. Last but not least, the originating cloud server may keep the transmitted data for the purpose of extracting the implicit benefits. From the perspective of the data owners, the data reservation is unanticipated. In short, the cloud storage service is cost-effective, but it necessarily faces major security issues, particularly in terms of safe data transfer, integrity verification, and verifiable deletion. If these issues are not addressed properly, the public may be hesitant to embrace and use cloud storage services.Contributions We investigate the issues of safe data transfer anddeletion in cloud storage in this paper, with a focus on achieving public verifiability. Then we offer a Bloom filter-based counting system that not only allows for provable data transmission between two clouds but also allows for publically verifiable data deletion. The verifier (the data owner and the target cloud server) can discover malicious acts by evaluating the returned transfer and deletion evidences if the originating cloud server does not migrate or remove the data honestly. Furthermore, unlike previous alternatives, our suggested scheme does not require the use of a

Trusted third party (TTP). We also use security analysis to show that our new approach may meet the needed design goals. Finally, simulation trials demonstrate that our novel concept is effective and feasible.

2 Literature survey

In récent décades, secure data rassure has been extensive resache, résultant in a Walt of littératures. Based on the deletion methods utilisez, existing data deletion strategies can be categorised into three types. The first method for deleting data is unlinking, which is the most efficient and straightforward method. The underlying file system's relationship to the file is specifically removed. The user is subsequently given a one-bit answer (Succès or Failure) indicating the success or failure of the data deletion operation. Although unlinking removes a file's link, the content of the file remains on the disc. As a result, the attacker can easily recover the "lost" data by scanning the necessary disc using a tool. As a result, the data was removed. It's possible that the response is deceptive. The second method for erasing data is overwriting, which can erase the file's content. The primary idea is to use random data instead of physical media.In 2010, Perito and Tsudik provided proofs of safe erasing as a novel technique. (PoSE-s). To overwrite the matched discs in their scheme, a succession of random patterns are provided. As proof of data destruction, the same string of patterns is returned. Luo et al. [23] created a one-of-a-kind outsourced data erasing method that does thefollowing: This work is licenced under the Creative Commons Attribution 4.0

License. See \shttps://creativecommons.org/licenses/by/4.0/ for additional information This menuscrit has been Accepted for publication in a future issue of This journal, but it is currently under review. It's possible that the content will change before it's published. Publicly Verifiable and Efficient Fine-Grained Data Deletion Scheme in Cloud Computing, Yang et al., Publicly Verifiable and Efficient Fine-Grained Data Deletion Scheme in Cloud Computing, Yang et al., IEEE AccessC, DOI10.1109/ACCESS.2020.2997351. Publicly Verifiable, Yang et al. The data can be removed by replacing it with some random data. In other words, the overwriting was disguised as a data update operation. Meanwhile, they achieved data deletion resultchecking using a challenge-response method. The verification may, however, fail due to network delays. Furthermore, a number of standards have specified overwriting deletion (such as [24]). The final method of data deletion is to destroy the data decryption key. The primary idea is to obliterate data encryption keys in order to make the system more secure. Xue et al.[19] investigated the goal of secure data deletion and proposed a key-policy attribute-based encryption technique that allows for fine-grained data access control and assured deletion. They achieve data deletion by eliminating the attribute and achieving verifiability via Merkle hash tree (MHT), but their technique requires a trusted authority. Du et al.[20] devised the Associated deletion strategy for multi-copy (ADM), which achieves data integrity verification and proven deletion through the use of a pre-deleting sequence and MHT. Their solution, however, necessitates the use of a TTP to maintain the data keys. Yang et al.[21] presented a Blockchain-based cloud data deletion technique in 2018, in which the cloud performs the deletion and publishes the corresponding deletion proof on Blockchain. The deletion outcome can then be checked by any verifier by confirming the deletion evidence. Additionally, they eliminate the bottleneck of requiring a TTP. Although all of these techniques can guarantee data destruction, they cannot guarantee secure data transport.

.

3 Implementation Study

- The client uses the Security Service module to upload the data. AES and Counting Bloom Filter Algorithms are included in the Security Service module.
- The data is subsequently encrypted with a 128-bit key using the AES technique. After generating a public key that is connected with the encrypted data, the data is uploaded to the cloud. In the cloud, a data record is kept in the form of an array and a SQL table.
- A copy of a public key is transmitted to the Third Party Auditor (TPA) when it is generated, and the TPA keeps it in its own database.
- The client checks for the availability of the key in the Bloom Filter array stored on the cloud while sending a verification request to the TPA. If the key is present, the data's integrity is jeopardised; if the key is not present, the data's integrity is jeopardised.

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, Issue 6, June : 2023

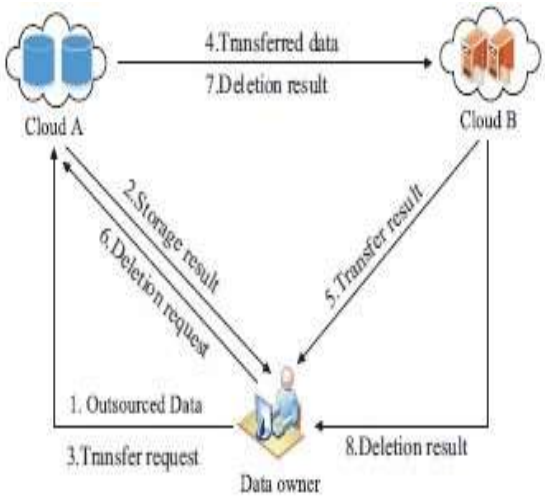

FIG1: - ARCHITECTURE Diagram

4 Proposed Approach

The method is built on a counting Bloom filter-based scheme that not only allows for demonstrable data transmission between two clouds but also allows for publicly verifiable data eradication. Unlike previous alternatives, our suggested scheme does not require the useof a Trusted third party (TTP). Through security research, our new solution can meet the intended design goals.

ADVANTAGES OF PROPOSED SYSTEM

- To encrypt or decrypt data using a 128-bit key, which will then be stored or retrieved using a cloud storage service.
- To ensure the integrity of data stored in the cloud, a datastructure that stores the element utilising is used. has
- Bloom filters are compressed data structures that describe a set's probability to allow membership queries. It answers the questions "Is an element X present in the set S hing functions?" and "Is an elementX present in the set S hing functions?"

5 Algorithms Used

The Bloom filter (BF)

is a space-efficient data structure invented by Burton Howard Bloom in 1970 for testing if a set includes a given element. This is intended to determine whether an element is present in a set quickly and effectively. No of how many elements the set and the BF contain, inserting an element or verifying that an element belongs to the set incurs a fixed time overhead. A BF denotes the first bit of an array of m bits, all of which are set to 0. The insertion passes an element through k separate hash functions, each of which maps the element to one of the m array locations, which are subsequently set to 1. If all places acquired by assessing the hash evaluations are set to 1, the element is regarded to be in the BF when querying the BF on it. The first secret key sk generated by a BFE scheme's generation method corresponds to an empty BF. Encryption takes a message M and a public key pk, samples a random element s (which acts as a tag for the ciphertext) corresponding to the BF's universe U, and encrypts a message using pk with regard to the k places defined in the BF by s.

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, Issue 6, June : 2023

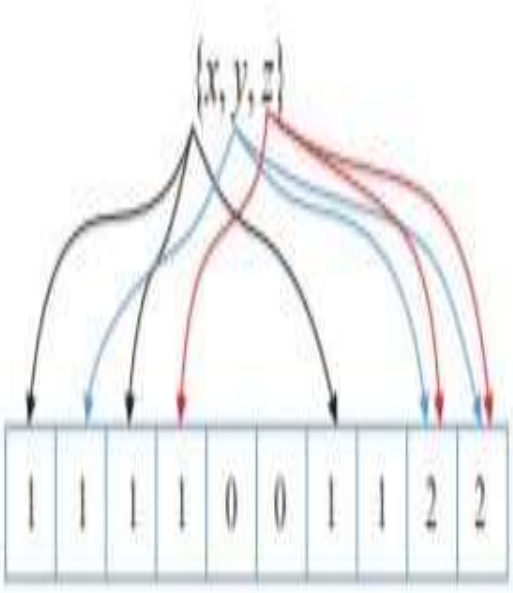

Fig 2:-Example of Bloom filter

A BF can be thought of as an m-byte bit array with k hash functions: hi(): 0, 1, 0, 1, m. To insert an element, we must set thegroup of k bits to 1, with hash values h1(x),... ,hk
determining the positions of these bits (x). As illustrated in Fig 2.2.1, membership tests are implemented by performing the samehash calculations and reporting success if all of the appropriate places are one.

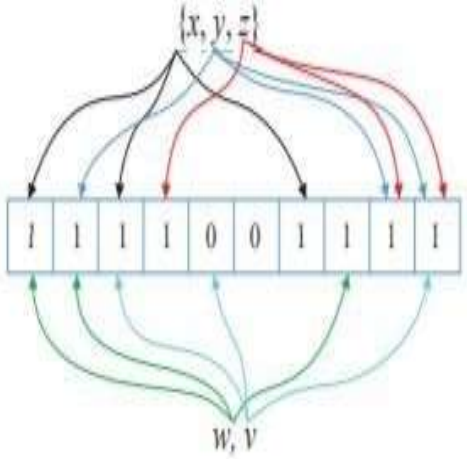

Fig 3 Example of Counting Bloom filter

It's worth noting that the BF contains a false positive, meaning that even though all of the k bits connected to w are one, w doesnot belong to the set with a little probability.
However, we may minimise the probability by adjusting factors such as the number of hash functions k, the length of the BF m, and the amount of elements n. Furthermore, if the parameters aresuitable, the probability will be so minimal that it will be inconsequential. Furthermore, BF is unable to remove an element from the data collection. The Counting Bloom filter (CBF) is proposed as a solution to this flaw. CBF is a version of BF that replaces every "bit" position with a counter cell count, as seen in Fig 2.2.2. To insert an element y, we must increase the k relevantcounters by one; the counters' indices are similarly defined by thecounters' indexes.
h1(y), h2(y), · · · , hk(y). On the contrary, the element deletion operation is simply to decrease the k Secure Data Transfer and Deletion from Counting Bloom Filter in Cloud Computing 275 corresponding counters by one

AES Algorithm:

The Advanced Encryption Standards (AES) algorithm uses a symmetric key. The block cypher algorithm is used by the MAC. Ablock cypher is an algorithm that encrypts and decrypts data into 128-bit blocks using 128/192/256-bit keys.

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, Issue 6, June : 2023

Secret key algorithms are another name for symmetric key algorithms. Because these algorithms often use a single key that is kept secret by the systems involved in the encryption and decryption operations, this is the case. Symmetric key algorithms are cryptography[6] techniques that employ the same cryptographic keys for both plain text encryption and cypher text decryption. The keys could be identical, or there could be a simple transformation that connects them. In practise, the keys constitute a shared secret between two or more parties that can be utilised to maintain a secure data transmission. One of the fundamental disadvantages of symmetric-key encryption over public-key encryption is that both parties must have access to the secret key (Also known as asymmetric-key encryption).

6 Results and Evolution Metrics

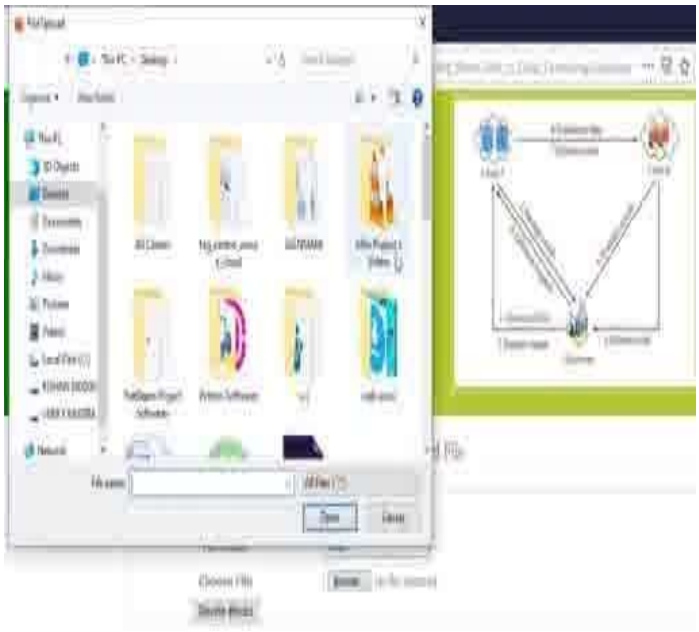

Fig 4: - home page of the website

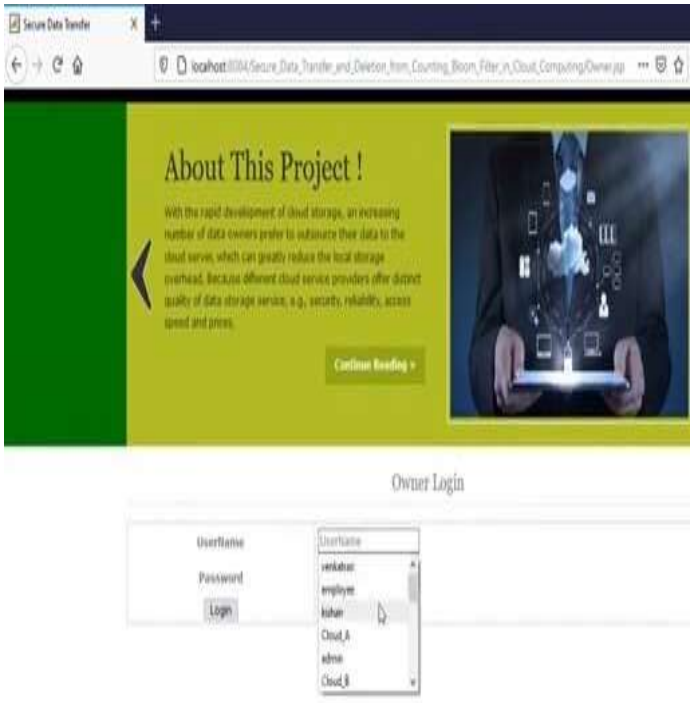

Fig 5 login page for owner

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 52, Issue 6, June : 2023

Fig 6 migrating the uploaded file

7. Conclusion

The data owner in cloud storage does not believe that the cloud server will carry out data transfer and delete activities honestly. We propose a CBF-based secure data transfer technique that can also perform verifiable data erasure to tackle this challenge. In our method, cloud B may verify the integrity of the sent data, ensuring that the data is completely migrated. Furthermore, cloud A shouldemploy CBF to generate a deletion evidence after deletion, which would be used by the data owner to validate the deletion result. As aresult, cloud A cannot act maliciously and successfully defraud the data owner. Finally, the results of the security analysis and simulationshow that our idea is both secure and feasible.

8 References

[1] C. Yang and J. Ye, "Secure and efficient fine-grained data accesscontrol scheme in cloud computing", Journal of High Speed Networks, Vol.21, No.4, pp.259–271, 2015.

[2] X. Chen, J. Li, J. Ma, et al., "New algorithms for secure outsourcing of modular exponentiations", IEEE Transactions onParallel and Distributed Systems, Vol.25, No.9, pp.2386–2396, 2014.

[3] P. Li, J. Li, Z. Huang, et al., "Privacy-preserving outsourced classification in cloud computing", Cluster Computing, Vol.21, No.1,pp.277–286, 2018.

[4] B. Varghese and R. Buyya, "Next generation cloud computing: New trends and research directions", Future Generation Computer Systems, Vol.79, pp.849–861, 2018.

[5] W. Shen, J. Qin, J. Yu, et al., "Enabling identity-based integrity auditing and data sharing with sensitive information hiding for securecloud storage", IEEE Transactions on Information Forensics and Security, Vol.14, No.2, pp.331–346, 2019.

[6] R. Kaur, I. Chana and J. Bhattacharya J, "Data deduplication techniques for efficient cloud storage management: A systematic review", The Journal of Supercomputing, Vol.74, No.5, pp.2035– 2085, 2018.

[7] Cisco, "Cisco global cloud index: Forecast and methodology, 2014–2019", available at: https://www.cisco.com/c/en/us-
/solutions/collateral/service-provider/global-cloud-index-gci/ white- paper-c11-738085.pdf, 2019-5-5.

[8] Cloudsfer, "Migrate & backup your files from any cloud to any cloud", available at: https://www.cloudsfer.com/, 2019-5-5.

[9] Y. Liu, S. Xiao, H. Wang, et al., "New provable data transfer from provable data possession and deletion for secure cloud

storage", International Journal of Distributed Sensor Networks, Vol.15, No.4,pp.1–12, 2019.

[10] Y. Wang, X. Tao, J. Ni, et al., "Data integrity checking with reliable data transfer for secure cloud storage", International Journalof Web and Grid Services, Vol.14, No.1, pp.106–121, 2018.

[11] Y. Luo, M. Xu, S. Fu, et al., "Enabling assured deletion in the cloud storage by overwriting", Proc. of the 4th ACM International Workshop on Security in Cloud Computing, Xi'an, China, pp.17–23,2016.

[12] C. Yang and X. Tao, "New publicly verifiable cloud data deletion scheme with efficient tracking", Proc. of the 2th International Conference on

[13] Security with Intelligent Computing andBig-data Services, Guilin, China, pp.359–372, 2018.

[14] Y. Tang, P.P Lee, J.C. Lui, et al., "Secure overlay cloud storage with access control and assured deletion", IEEE Transactions on Dependable and Secure Computing, Vol.9, No.6, pp.903–916, 2012.

[15] Y. Tang, P.P.C. Lee, J.C.S. Lui, et al., "FADE: Secure overlay cloud storage with file assured deletion", Proc. of the 6thInternational Conference on Security and Privacy in CommunicationSystems, Springer, pp.380-397, 2010.

[16] Z. Mo, Y. Qiao and S. Chen, "Two-party fine-grained assured deletion of outsourced data in cloud systems", Proc. of the 34th International Conference on Distributed Computing Systems, Madrid, Spain, pp.308–317, 2014.

[17] M. Paul and A. Saxena, "Proof of erasability for ensuring comprehensive data deletion in cloud computing", Proc. of the International Conference on Network Security and Applications, Chennai, India, pp.340–348, 2010.

[18] A. Rahumed, H.C.H. Chen, Y. Tang, et al., "A secure cloud backup system with assured deletion and version control", Proc. of the 40th International Conference on Parallel Processing Workshops,Taipei City, Taiwan, pp.160–167, 2011.

[19] B. Hall and M. Govindarasu, "An assured deletion technique forcloud-based IoT", Proc. of the 27th International Conference on Computer Communication and Networks, Hangzhou, China, pp.1–8,2018.

[20] L. Xue, Y. Yu, Y. Li, et al., "Efficient attributebased encryption with attribute revocation for assured data deletion", Information Sciences, Vol.479, pp.640–650, 2019.

[21] L. Du, Z. Zhang, S. Tan, et al., "An Associated Deletion Scheme for Multi-copy in Cloud Storage", Proc. of the 18th International Conference on Algorithms and Architectures for Parallel Processing, Guangzhou, China, pp.511–526, 2018.

[22] C. Yang, X. Chen and Y. Xiang, "Blockchain-based publicly verifiable data deletion scheme for cloud storage", Journal of Network and Computer Applications, Vol.103, pp.185–193, 2018.

[23] Y. Yu, J. Ni, W. Wu, et al., "Provable data possession supporting secure data transfer for cloud storage", Proc. of 2015 10th International Conference on Broadband and Wireless Computing, Communication and Applications(BWCCA 2015), Krakow, Poland,pp.38–42, 2015.

[24] J. Ni, X. Lin, K. Zhang, et al., "Secure outsourced data transfer with integrity verification in cloud storage", Proc. of 2016 IEEE/CICInternational Conference on Communications in China, Chengdu, China, pp.1–6, 2016.

[25] L. Xue, J. Ni, Y. Li, et al., "Provable data transfer from provable data possession and deletion in cloud storage", Computer Standards & Interfaces, Vol.54, pp.46–54, 2017.

[26] Y. Liu, X. Wang, Y. Cao, et al., "Improved provable data transfer from provable data possession and deletion in cloud storage", Proc. of Conference on Intelligent Networking and Collaborative Systems, Bratislava, Slovakia, pp.445–452, 2018.