IJIRCCE

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

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

Volume 9, Issue 7, July 2021

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 7.542

International Journal of Innovative Research in Computer and Communication Engineering

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.542

|| Volume 9, Issue 7, July 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0907228 |

Data Migration and Deletion between multiple Cloud Servers Using Bloom Filter

Mr. Chandrabhan Dashrath Kumare, Prof. Prathmesh S. Powar

Dept. of Computer Science and Engg., Ashokrao Mane Group of Institutions, Kolhapur, Maharashtra, India

Dept. of Computer Science and Engg., Ashokrao Mane Group of Institutions, Kolhapur, Maharashtra, India

ABSTRACT: The utilization of distributed storage, an expanding number of information proprietors like to re-appropriate their information to the cloud worker, which decrease the neighborhood stockpiling overhead. Diverse cloud specialist co-ops offer unmistakable nature of information stockpiling administration, So cloud information move has become a central necessity of the information proprietor to change the cloud specialist co-ops. Henceforth, how to safely relocate the information starting with one cloud then onto the next and for all time erase the moved information from the first cloud turns into an essential worry of information owners.With the speedy improvement of cloud limit, a growing number of data owners like to re-fitting their data to the cloud specialist, which can altogether decrease the local accumulating overhead. Since different cloud expert associations offer undeniable nature of data amassing organization, e.g., security, trustworthiness, access speed and expenses, cloud data move has gotten a focal essential of the data owner to change the cloud expert associations. Accordingly, how to securely migrate the data starting with one cloud then onto the next and everlastingly eradicate the moved data from the primary cloud transforms into a fundamental concern of data owners. To handle this issue, we foster another including Bloom channel based arrangement. The proposed contrive not solely can achieve secure data move yet also can comprehend enduring data abrogation. Besides, the proposed plan can satisfy the public proof without requiring any trusted in untouchable. Finally, we besides develop an amusement execution that delineates the good judgment and viability of our recommendation.

Cloud computing, an emerging and very promising computing paradigm, connects large-scale distributed storage resources, computing resources and network bandwidths together. By using these resources, it can provide tenants with plenty of high-quality cloud services.Due to the attractive advantages, the services (especially cloud storage service) have been widely applied, by which the resource-constraint data owners can outsource their data to the cloud server, which can greatly reduce the data owners' local storage overhead[5,6]. Because of the promising market prospect, an increasing number of companies (e.g., Microsoft, Amazon,Alibaba) offer data owners cloud storage service with different prices, security, access speed, etc.

To enjoy more suitable cloud storage service, the data owners might change the cloud torage service providers. Hence, they might migrate their outsourced data from one cloud to another, and then delete the transferred data from the original cloud. According to Cisco[7], the cloud traffic is expected to be 95% of the total traffic by the end of 2021, and almost 14% of the total cloud traffic will be the traffic between different cloud data centers. Foreseeably, the outsourced data transfer will become a fundamental requirement from the data owners' point of view.

To realize secure data migration, an outsourced data transfer app, Cloudsfer, has been designed utilizing cryptographic algorithm to prevent the data from privacy disclosure in the transfer phase. But there are still some security problems in processing the cloud data migration and deletion. Firstly, for saving network bandwidth, the cloud server might merely migrate part of the data, or even deliver some unrelated data to cheat the data owner [9]. Secondly, because of the network instability, some data blocks may lose during the transfer process. Meanwhile, the adversary may destroy the transferred data blocks [10]. Hence, the transferred data may be polluted during the migration process. Last but not least, the original cloud server might maliciously reserve the transferred data for digging the implicit benefits

I. RELATED WORK

Xue et al.[9] studied the goal of secure data deletion, and put forward a key-policy attribute based encryption scheme, which can achieve data fine-grained access control and assured deletion. They reach data deletion by removing the attribute and use Merkle hash tree (MHT) to achieve verifiability, but their scheme requires a trusted authority.

Du et al.[2] designed a scheme called Associated deletion scheme for multi-copy (ADM), which uses pre-deleting sequence and MHT to achieve data integrity verification and provable deletion. However, their scheme also requires a TTP to manage the data keys.

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.542

|| Volume 9, Issue 7, July 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0907228 |

In 2015, Yu et al. [3] presented a Provable data possession (PDP) scheme that can also support secure data migration.

In 2018, Yang et al. [8] presented a Block chain-based cloud data deletion scheme, in which the cloud executes deletion operation and publishes the corresponding deletion evidence on Block chain. Then any verifier can check the deletion result by verifying the deletion proof. Besides, they solve the bottleneck of requiring a TTP. Although these schemes all can achieve verifiable data deletion, they cannot realize secure data transfer.

II. IMPLEMENTATION

2.1 Key Generation

Let G1, G2 be the cyclic groups of prime order p, let g be a generator of G1, and e : G1 × G1 → G2 be a map with the following properties.

(1) Bilinearity: e(ga,gb) = e(g,g)ab, a,b ∈ Zp.

(2) Non-degeneracy: There exist x,y ∈ G1 such that e(x,y) = 1.

(3) Computable: For all x,y ∈ G1, e(x,y) has to be computable in an efficient manner.

Decisional Modified Bilinear Diffie–Hellman (DMBDH) assumption. Given g,gx ,gy ,gz ∈ G1 for unknown random x,y,z,r ∈ Z∗ p. The DMBDH assumption is that no polynomial-time adversary is able to distinguish the tuple (gx ,gy ,gz,e(g,g) xy z) from a random tuple (gx ,gy ,gz,e(g,g)r) with more than a negligible advantage,

Pr A gx ,gy ,gz ,e(g,g) xy z = 1 − Pr A gx ,gy ,gz ,e(g,g)r = 1 ε.

Initialization: The adversary the identity, α, that he wishes to be challenged upon. Setup: The challenger runs the setup phase of the algorithm and tells the adversary the public parameters. Phase 1: The adversary is allowed to issue queries for private keys for many identities, γj , where ∀j , |γj ∩ α| < d. Challenge: The adversary submits two equal length messages M0, M1. The challenger flips a random coin, b, and encrypts Mb with α. The ciphertext is passed to the adversary. Phase 2: Phase 1 is repeated. Guess: The adversary outputs a guess b of b. The advantage of an adversary in this game is defined as

Pr|b = b| − 1 2

In the original attribute-based access control model, for the purpose of reducing the amount of computation carried out by participants, an authority is usually employed to generate the users' private key. In this model, all the private keys, which are used to recover the message M, are generated by authority. When some member is removed, the message should be re-encrypted. Original attribute-based access control model. Here are some notations.

k: security parameter

ω : set of attributes needed for decryption

ω: set of user's attributes

sk: private key

PK: public key

E(·): encryption algorithm

D(·): decryption algorithm s: side information

R(·, ·): matching relation of two elements

f (·): key generation algorithm

2.2 Data encryption

The original attribute-based access control model can be described as follows.

Setup: Authority generates sk, pk by k and the set of all attributes.

Encryption: Encryptor generates a set of ciphertext {C = E(M),ω ,s, pk} under pkand ω

First, a random value b ∈ Z∗ p is chosen by encryptor, and encryptor computes

m t=1(gi)bxtcit = gby , 1 t m.

The ciphertext is as: E = Hi,ω ,C = Mgby ,E = bYs , Ei = T s i i∈ω .

Encryptor sends Hi and E to data manager. Data manager checks whether Hi is belonging to the set S.

2.3 Data outsourcing:

The cloud A stores D and generates storage proof. Then the data owner checks the storage result and deletes the local backup. i) Upon receiving data set D and file tag tagf , the cloud A stores D, and uses the indexes (a1, a2, · · · , an) to construct a counting Bloom filter CBFs, where i = 1, 2, · · · , n. Meanwhile, the cloud A stores tagf as the index of D. Finally, the cloud A computes a signature sigs = SignSKA (storage‖tagf ‖CBFs‖Ts), and sends the proof λ = (CBFs, Ts, sigs) to the data owner, where Sign is a ECDSA signature algorithm, Ts is a timestamp.ii) On receipt of storage proof λ, the data owner checks its validity. Specifically, the data owner first checks the validity of the signature sigs. If sigs is invalid, the data owner quits and outputs failure;

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.542

|| Volume 9, Issue 7, July 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0907228 |

otherwise, the data owner randomly chooses half of the indexes from (a1, a2, · · · , an) to check the correctness of the CBFs.If the CBFs is not correct, the data owner quits and outputs failure; otherwise, the data owner deletes the local backup.The cloud A stores D and creates stockpiling evidence.
Then, at that point the information proprietor checks the capacity result and erases the nearby reinforcement.
I) Upon getting informational index D and document tag tagf , the cloud A stores D, and utilizations the lists (a1, a2, · · , an) to build an including Bloom channel CBFs, where I = 1, 2, · · , n. In the interim, the cloud A stores tagf as the list of D. At last, the cloud A registers a mark sigs = SignSKA (storage‖tagf ‖CBFs‖Ts), and sends the
evidence λ = (CBFs, Ts, sigs) to the information proprietor, where Sign is an ECDSA signature calculation, Ts is a timestamp.
ii) On receipt of capacity evidence λ, the information proprietor checks its legitimacy. In particular, the information proprietor first checks the legitimacy of the mark sigs. In case sigs is invalid, the information proprietor stops and yields disappointment; in any case, the information proprietor haphazardly picks half of the lists from (a1, a2, · · , an) to check the rightness of the CBFs. In the event that the CBFs isn't right, the information proprietor stops and yields disappointment; in any case, the information proprietor erases the nearby reinforcement.

2.4Data transfer and Deletion

When the data owner wants to change the service provider, he migrates some data blocks, even the whole file from the cloud A to the cloud B. i) Firstly, the data owner generates the index set of block indices ϕ, which will identify the data blocks that need to be migrated. Then the data owner computes a signature sigt = SignSKO (transfer‖tagf ‖ϕ‖Tt),. After that the data owner generates a transfer request Rt = (transfer, tagf , ϕ, Tt, sigt), and then sends it to the cloud A. Meanwhile, the data owner sends the hash values {Hi}i∈ϕ to the cloud B. The data owner might require the cloud A to delete some data blocks when they have been transferred to the cloud B successfully. Firstly, the data owner computes a signature sigd = SignSKA (delete‖tagf ‖ϕ‖Td), where Td is a timestamp. Then the data owner generates a data deletion request Rd = (delete, tagf , ϕ, Td, sigd) and sends it to cloud A. ii) Upon receiving Rd, the cloud A checks Rd. If Rd is invalid, the cloud A quits and outputs failure; otherwise, the cloud A deletes the datablocks {(ai ,Ci)}i∈ϕ by overwriting. Meantime, the cloud A removes indexes {aq}q∈ϕ from the CBFs and obtains a new counting Bloom filter CBFd. Finally, the cloud A computes a signature sigda = Sign(delete‖Rd‖CBFd), and returns the data deletion evidence τ = (sigda, CBFd) to the data owner.At the point when the information proprietor needs to change the serviceprovider, he relocates a few information blocks, even the wholefile from the cloud A to the cloud B.

I) Firstly, the information proprietor produces the file set of square files φ, which will recognize the information hinders that should be moved. Then, at that point the information proprietor figures a

signaturesigt = SignSKO (transfer‖tagf ‖φ‖Tt), where Tt is a timestamp. After that the information proprietor produces an exchange demand Rt = (move, tagf , φ, Tt, sigt), and

then, at that point sends it to the cloud A. In the mean time, the information proprietor sends the hash esteems {Hi}i∈φ to the cloud B.

ii) On receipt of the exchange demand Rt, the cloud A checks the legitimacy of Rt. In case Rt isn't substantial, the cloud A stops and yields disappointment; in any case, the cloud

A processes a mark sigta = SignSKA (Rt‖Tt), and sends the information blocks {(ai

, Ci)}i∈φ to the cloud B, alongside the mark sigta and the exchange demand Rt.

Move check The cloud B needs to check the accuracy of the exchange and returns the exchange result to the information proprietor. I) Firstly, the cloud B checks the legitimacy of the

move demand Rt and mark sigta. If not both ofthem are legitimate, the cloud B stops and yields disappointment; in any case, the cloud B watches that whether the condition

Hello there = H(tagf ‖ai‖mi) holds, where I □ φ. In the event that Hi≠H(tagf ‖ai‖Ci), the cloud B requires the cloud A to send(ai, Ci) once more; something else, the cloud B goes to

Step ii) The cloud B stores the squares {(ai, Ci)}i∈φ, and utilizations the records {ai}i∈φ to develop another checking Bloom channel CBFb. Then, at that point the cloud B registers a mark sigtb = SignSKB (success‖tagf

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.542

|| Volume 9, Issue 7, July 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0907228 |

||φ||Tt||CBFb).Finally, the cloud B returns the exchange evidence π =(sigta,sigtb, CBFb) to the information owner.iii) Upon receipt of π, the information proprietor checks the exchange result. To be explicit, the information proprietor checks the legitimacy of the mark sigtb. In the interim, the information proprietor arbitrarily picks half of the lists from set φ to check the rightness of the tallying Bloom channel CBFb. In the event that and just if every one of the checks pass, the information proprietor believes the exchange evidence is substantial, and the cloud B stores the moved information genuinely.
The information proprietor may require the cloud A to erase a few information blocks when they have been moved to the cloud B effectively.

I) Firstly, the information proprietor registers a mark sigd = SignSKA (delete||tagf ||φ||Td), where Td is a timestamp. Then, at that point the information proprietor creates an information cancellation demand Rd = (erase, tagf , φ, Td, sigd) and sends it to cloud A.

ii) Upon getting Rd, the cloud A checks Rd. In case Rd is invalid, the cloud A stops and yields disappointment; otherwise,the cloud An erases the information blocks {(ai, Ci)}i∈φ by overwriting. Interim, the cloud An eliminates lists {aq}q∈φ from the CBFs and gets another tallying Bloom channel CBFd. At last, the cloud A processes a

signaturesigda = Sign(delete||Rd||CBFd), and returns the information cancellation proof τ = (sigda, CBFd) to the information proprietor.

iii) After getting τ , the information proprietor checks the mark sigda. In case sigda is invalid, the information proprietor stops and yields disappointment; in any case, the information proprietor arbitrarily picks half of the lists from φ to check the conditions CBF(aq) = 0 and decides whether aq has a place with the CBFd. On the off chance that the conditions hold, the information proprietor trusts τ is legitimate.

III. RESULT ANALYSIS

To mimic the information move, we increment the quantity of moved information blocks from 10 to 80 with a stage for 10. For straightforwardness, we fix n = 150 and overlook the correspondence overhead, as displayed in Fig.2 . The time cost increments with the quantity of moved information blocks.

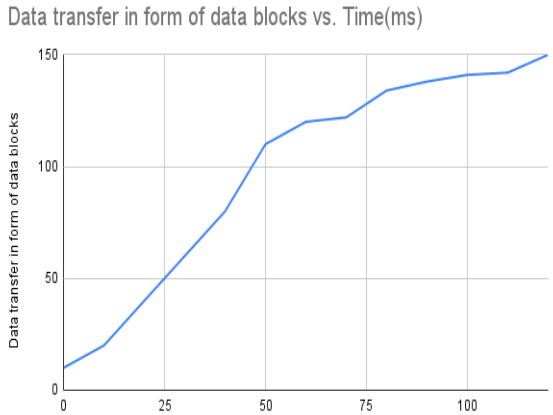

Figure 2. Time taken to transfer blocks

IV. CONCLUSION

We have successfully implemented Key Generation Data encryption,Data outsourcing, Data transfer and Deletion. n distributed storage, the information proprietor does not really accept that that the cloud worker may execute the information move and cancellation tasks sincerely. To address this issue, we propose a CBF-based secure information move conspire, which can likewise acknowledge obvious information erasure. In our plan, the cloud B can check the moved information respectability, which can ensure the information is completely relocated. Also, the cloud An ought to embrace CBF to create an erasure proof get-togethers, which will be utilized to confirm the erasure result by the information proprietor. Consequently, the cloud A can't act vindictively and cheat the information proprietor

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 7.542

|| Volume 9, Issue 7, July 2021 ||

| DOI: 10.15680/IJIRCCE.2021.0907228 |

effectively. At last, the security examination and reproduction results approve the security and practicability of our proposition, individually.

REFERENCES

[1] C. Yang and J. Ye, "Secure and efficient fine-grained data access control 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 on Parallel 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 secure cloud 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 Journal of Web and Grid Services, Vol.14, No.1, pp.106–121, 2018.

INNO SPACE
SJIF Scientific Journal Impact Factor
Impact Factor: 7.542

doi cross ref

ISSN INTERNATIONAL STANDARD SERIAL NUMBER INDIA

निस्केयर NISCAIR

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