ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 11, November 2015

An Approach for Efficient and Reliable Storage in Cloud Computing Environment

Mr. Santosh Ramesh Kadlag[1], Prof. Mayur C Akewar[2]

M.E Student, Dept. of Computer Engineering, RMD Sinhgad School of Engineering, Pune, Maharashtra India

Assistant Professor, Dept. of Computer Engineering, RMD Sinhgad School of Engineering, Pune, Maharashtra, India

ABSTRACT: Data de-duplication is one of the most important technique used for removing the identical copies of repeating data and it is used in the cloud storage for the purpose of reduce the storage space. Here only one copy for each file store. These files are owned by huge number of users. Keeping the multiple data copies with similar content de-duplication eliminates redundant data by keeping only one physical copy and refer other redundant data to that copy. Data de-duplication can be file level or block level. The duplicate copies of identical file eliminates by file level de-duplication. And block level de-duplication eliminates duplicate blocks of data that occur in non-identical files. In de-de-duplication storage space and bandwidth is reduced but reliability. This paper mainly concentrates on reducing storage while improving reliability, integrity and privacy of user's sensitive data. File is divided into fragments using Secrete Sharing Scheme. This system allocates the file fragments using T-colouring graph technique. T.-Colouring allows to store the nodes at certain distance so prevents attacks like guessing location of fragments. To maintain integrity we are providing the Third Party Auditor scheme which makes the audit of file stored at cloud and notifies the data owner about file status stored at cloud server. This system supports security challenges such as authorized duplicate check, integrity, data confidentiality and reliability.

KEYWORDS:Deduplication; Distributed storage system; Reliability; Secrete Sharing; Fragmentation; Replication, Public audit; File level de duplication; Block level de duplication; T-Coloring algorithm .

I. INTRODUCTION

Now a day due tremendous growth in information technology and digitization large amount of data is generating. So storage overhead is increases day by day. As storage overhead increases cost of storage also increases. Currently due to Smart City, e-governance large data is going to generate. [1] In future volume of data expected to reach 40 trillion gigabytes in 2020. Paper proposes a space management technique called data de duplication. To manage storage space on cloud is using de duplication has received much attention from industry and students who working on cloud platform. De duplication technique becomes more useful and essential due to tremendous increase in data.

De duplication systems improve the storage utilization but question arises about reliability of data. Here reliability decreases while improving storage utilization. To solve above challenge of reliability paper proposes a T-coloring algorithm .In this technique data fragments stored at certain distance by T-coloring algorithm [3]. Therefore attacker unable to guess location of fragments. Reliability and privacy to users sensitive data is achieved .Next question is about integrity. To solve this challenge paper proposes a Third Party Auditing Scheme [4]. De- duplication algorithms like file level de duplication, block level de duplication are explained in proposed system in detail. Propose systems also concentrate on improving integrity, reliability using Third Party Audit Scheme and T-coloring Algorithm.

II. RELATED WORK

There are various deduplication technique are available. Much work is done on this concept. But existing schemes are having some drawbacks including reliability, integrity. Proposed System reduces storage space and improves reliability and also provides integrity. Existing de-duplication systems have only considered a single-server setting. In Previous systems challenge of data privacy was there because user's data is sensitive and this data is outsourced by user to cloud. So privacy must be kept of user's data. In Proposed systems using T-coloring fragments are stored in such way that though attacker attacks on one fragment still he unable to get meaningful information [3]. Encryption is use to protect the confidentiality before outsourcing data into cloud. When encryption is apply over the data it makes de duplication impossible. The reason is that the traditional encryption mechanisms including public key encryption and

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 11, November 2015

symmetric key encryption require different users to encrypt their data with their own keys. As a result identical data copies of different users will lead to different cipher texts [1][2]. To solve this problem convergent encryption key approach is proposed. Convergent key generate same cipher text for identical data copy. But this technique still has limitations of reducing error resilience [2].

[1] The traditional deduplication methods are not flexible and applicable directly in distributed and multi-server systems. If the same short value is stored at a different cloud storage server to support a duplicate check by using a traditional de-duplication method, it cannot resist the collusion attack launched by multiple servers. In other words, any of the servers can obtain shares of the data stored at the other servers with the same short value as proof of ownership Therefore, how to protect both confidentiality and reliability while achieving de duplication in a cloud storage system is still a challenge. Proposed deduplication is distributed de duplication scheme with file level and block level duplication check.[1]

1) File-Level Deduplication: This algorithm finds redundancies between different files and removes these redundancies by eliminating such files.

2) Block-Level Deduplication: This algorithm discovers and removes redundancies between data blocks.

III. LITERATURE SURVEY

Sr. No	Paper Title	Review Of Paper(algorithms)
1	Jin Li, Xiao Feng Chen, Xining Huang, Shaohua Tang and Yang Xiang ,'Secure Distributed Deduplication Systems with Improved Reliability', IEEE Transactions on Computers Volume, pp.1-12 ,2015.[1]	Distributed File level and block level algorithms are explained in detail. Which check for De duplication Secrete Sharing Scheme used for fragmentations.
2	Jin Li, Yan Kit Li, Xiaofeng Chen, Patrick P.C. Lee, and Wenjing Lou ,'A Hybrid Cloud Approach for Secure Authorized Deduplication', IEEE transactions on parallel and distributed systems, vol. 26, no. 5pp. 1206-1216, May 2015.[2]	Convergent key approach is proposed to encrypt data before de duplication.
3	Mazhar Ali, Kashif Bilal,Samee U. Khan,Bharadwaj Veeravalli,Keqin Li,Albert Y. Zomaya,'DROPS: Division and Replication of Data in Cloud for Optimal Performance and Security',IEEE Transactions on Cloud Computing, pp. 1-15, 2015.[3]	Paper gives about DROPS Concept. Which used for fragmentation and Replication of data. T-Coloring algorithm for placing node is given. Reliability is achieved
4	Jian Liu, Kun Huang, Hong Rong, Huimei Wang, and Ming Xian,'Privacy-Preserving Public Auditing for Regenerating-Code-Based Cloud Storage',IEEE Transactions on Information Forensics and Security, Vol. 10, No. 7, July 2015.[4]	Auditing algorithms for achieving integrity is given.
5	Backialakshmi.N ,Manikandan.M ,'Survey based on Secure Authorized Deduplication Hybrid Cloud Approach', IJIRST –International Journal for Innovative Research in Science & Technology, Volume 1 ,Issue 9, pp. 164-165,Feb 2015.[5]	Deduplication Techniques are revived.

Tabe1: Literature Survey

In [1] authors explain file level de duplication and block level de duplication. Ramp Secrete Sharing Scheme is used for dividing data in chunks is explained in detail which allows kipping confidentiality without using encryption mechanism. In [2]authors explain convergent key approach to solve challenge of traditional encryption which generate different cipher text for same plain text. In[3]authors explain DROP's concept with T- coloring algorithm to provide

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 11, November 2015

high security to data .T-coloring allows to place node at certain distance which gives more confidentiality. In [4] auditing mechanism is explained. In [5] survey of de duplication is given.

IV. PROPOSED SYSTEM

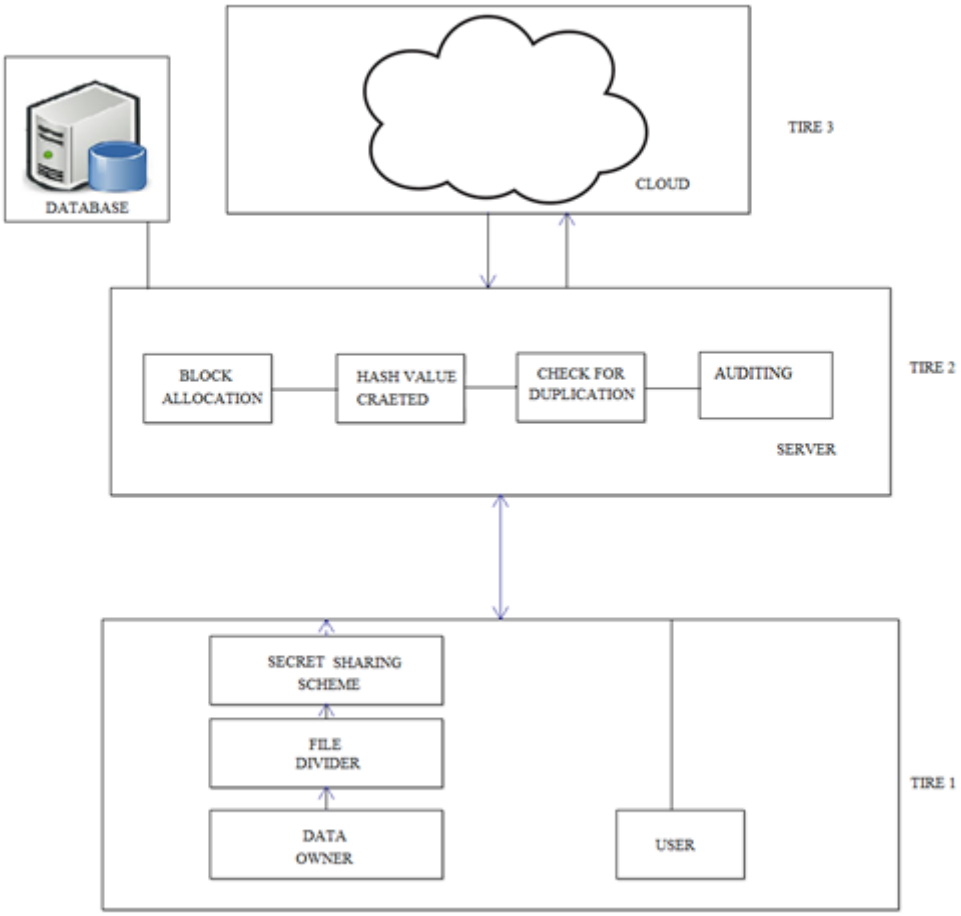

Fig1: System Architecture

A. *Design Work flow:*

A3 - Tire architecture is shown in Above Fig1.When data owner upload data it first fragmented using Secrete Sharing Scheme. Then Duplication Algorithm checks for File Level and Block Level Deduplication. To improve integrity and reliability T-Colouring and Auditing Algorithms are used. Whole process is explained below sections

- First upload the file
- Generate hash code for that file
- Divide that file into fragments using Secrete Sharing Scheme.
- Encrypt that fragments.
- Allocate that fragments on cloud server using T-coloring graph technique.
- While new file is trying to upload check for file level duplication and after that at block level duplication.
- Third Party Auditor checks the file status and notify the data owner about status.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 11, November 2015

B. *Description of the Proposed Algorithm:*

1) Secrete Sharing Scheme algorithm:[1]

Input: - Data
Process: -Data is split into chunks. A chunk is split into blocks, blocks are accessed incrementally
Encryption:- Ek(m) is a secure block cipher with block length n
K1, K2 is the primary and secondary key
A is a primitive element in F (2n)
N is the physical address of the chunk
The chunk key L is created as L = EK2 (N)
The i^{th} block key is i = aiL computed in F (2n)
The i^{th} block is encrypted as Ci= EK1 (Mi)
Out Put: - Encrypted Data

Now these fragments are allocated using T-coloring Algorithm and then file level and Block level Duplication is check.

2) T-Coloring Algorithm for fragments allocation:

[3] Once the file is split into fragments, the T-coloring methodology selects the cloud nodes for fragment placement .The selection is made by keeping an equal focus on both security and performance in terms of the access time. Select nodes that are most central to the cloud network to provide better access time. The DROPS methodology uses the concept of centrality to reduce access time. [3] The centrality determine how central a node is based on different measures

(a) Betweenness: The betweenness centrality of a node n is the number of the shortest paths, between other nodes, passing through n

(b) Closeness: A node is said to be closer with respect to all of the other nodes within a network, if the sum of the distances from all of the other nodes is lower than the sum of the distances of other candidate nodes from all of the other nodes. The lower the sum of distances from the other nodes, the more central is the node.

(c) Eccentricity centrality: The eccentricity of a node n is the maximum distance to any node from a node n . A node is more central in the network, if it is less eccentric .If all of the fragments are placed on the nodes based on the descending order of centrality, then there is a possibility that adjacent nodes are selected for fragment placement. Such a placement can provide clues to an attacker as to where other fragments might be present, reducing the security level of the data. To deal with the security aspects of placing fragments, we use the concept of T-coloring that was originally used for the channel assignment problem.

Generate a non-negative random number and build the set T starting from zero to the generated random number.
The set T is used to restrict the node selection to those nodes that are at hop-distances not belonging to T.
Assign colors to the nodes, such that, initially, all of the nodes are given the open color. Once a fragment is placed on the node, all of the nodes within the neighborhood at a distance belonging to T are assigned close color. In this process, we lose some of the central nodes that may increase the retrieval time but we achieve a higher security level. If somehow the intruder compromises a node and obtains a fragment, then the location of the other fragments cannot be determined.

3) File level Distributed De-duplication algorithm: [1] To support efficient duplicate check tags for each file will be computed and are sent to S-CSPs. To prevent a collusion attack launched by the S-CSPs, the tags stored at different storage servers are computationally independent and different. We now elaborate on the details of the construction as follows.

In our construction, the number of storage servers S-CSPs is assumed to be n with identities denoted by id_1, id_2, $\cdot \cdot$ id_n, respectively. Define the security parameter as 1^{λ} and initialize a secret sharing scheme SS = (Share, Recover), and a tag generation algorithm Tag Gen. the file storage system for the storage server is set to be $\perp$. File Upload. To upload a file F, the user interacts with S-CSPs to perform the de-duplication. More precisely, the user firstly computes and sends the file tag ϕF = Tag-Gen (F) to S-CSPs for the file duplicate check.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 11, November 2015

• If a duplicate is found, the user computes and sends ϕF idj = Tag-Gen′(F, idj) to the j-th server with identity idj via the secure channel for $1 \leq j \leq n$ (which could be implemented by a cryptographic hash function Hj(F) related with index

j). The reason for introducing an index j is to prevent the server from getting the shares of other S-CSPs for the same file or block, which will be explained in detail in the security analysis. If ϕF;idj matches the metadata stored with ϕF , the user will be provided a pointer for the shard stored at server idj .

• Otherwise, if no duplicate is found, the user will proceed as follows. He runs the secret sharing algorithm SS over F to get {cj} = Share (F), where cj is the j-th shard of F. He also computes ϕF;idj = Tag-Gen′(F, idj), which serves as the tag for the jth S-CSP. Finally, the user uploads the set of values {ϕF,cj , ϕF;idj } to the S-CSP with identity id_j via a secure channel. The S-CSP stores these values and returns a pointer back to the user for local storage.

File Download. To download a file F, the user first downloads the secret shares {cj} of the file from k out of n storage servers. Specifically, the user sends the pointer of F to k out of n S-CSPs. After gathering enough shares, the user reconstructs file F by using the algorithm of Recover ({cj}). This approach provides fault tolerance and allows the user to remain accessible even if any limited subsets of storage servers fail.

4)The Block-level Distributed De-duplication algorithm:

[1] In this section, we show how to achieve the fine-grained block-level distributed de-duplication. In a block-level de-duplication system, the user also needs to firstly perform the file-level de-duplication before uploading his file. If no duplicate is found, the user divides this file into blocks and performs block-level de-duplication. The system setup is the same as the file-level de-duplication system, except the block size parameter will be defined additionally. Next, we give the details of the algorithms of File Upload and File Download. File Upload. To upload a file F, the user first performs the file-level de-duplication by sending ϕF to the storage servers. If a duplicate is found, the user will perform the file-level de-duplication. Otherwise, if no duplicate is found, the user performs the block-level de-duplication as follows. He firstly divides F into a set of fragments {Bi} (where i = 1, 2, · · ·). For each fragment Bi, the user will perform a block-level duplicate check by computing ϕBi = Tag-Gen(Bi), where the data processing and duplicate check of block-level de-duplication is the same as that of file-level de-duplication if the file F is replaced with block Bi. Upon receiving block tags {ϕBi} the server with identity idj computes a block signal vector σBi for each i.

.•i) If $\sigma B_i=1$, the user further computes and sends ϕBi;j = Tag-Gen′(Bi, j) to the S-CSP with identity idj . If it also matches the corresponding tag stored, S-CSP returns a block pointer of Bi to the user. Then, the user keeps the block pointer of Bi and does not need to upload Bi.

• ii) If σBi=0, the user runs the secret sharing algorithm SS over Bi and gets {cij} = Share (Bi), where cij is the j^{th} secret share of Bi. The user also computes ϕBi;j for $1 \leq j \leq n$ and uploads the set of values {ϕF , ϕF;idj , cij , ϕBi;j} to the server idj via a secure channel. The S-CSP returns the corresponding pointers back to the user.

5) Auditing Algorithm:[4]

Key Gen: -This polynomial-time algorithm is run by the data owner to initialize its public and secret parameters by taking security parameter κ as input.

Delegation: -This algorithm represents the interaction between the data owner and proxy. The data owner delivers partial secret key *x* to the proxy through a secure approach.

Sig and Block Gen: -This polynomial time algorithm is run by the data owner and takes the secret parameter and the original file as input, and then outputs a coded block set, an authenticator set and a file tag.

Audit: -The cloud servers and TPA interact with one another to take a random sample on the blocks and check the data intactness in this procedure.

Challenge:-This algorithm is performed by the TPA with the information of the file as input and a challenge as output.

Proof gen: -This algorithm is run by each cloud server with input challenge, coded block set and authenticator set then

it outputs a proof.

Verify: -This algorithm is run by TPA immediately after a proof is received. Taking the proof, public parameter and the corresponding challenge *C* as input, it outputs 1 if the verification passed and 0 otherwise.

Repair: -In the absence of the data owner, the proxy interacts with the cloud servers during this procedure to repair the wrong server detected by the auditing process.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 11, November 2015

Claim for Rep:-This algorithm is similar with the Challenge algorithm in the Audit phase, but outputs a claim for repair.
Gen for Rep:-The cloud servers run this algorithm upon receiving the and finally output the block and authenticators set with another two inputs.
Block and Sig Re-Gen:-The proxy implements this algorithm with the claim and response from each server as input, and outputs a new coded block set and authenticator set.

V. CONCLUSION AND FUTURE WORK

Paper proposed the distributed de-duplication systems to improve the reliability of data while achieving the confidentiality, integrity of the users' outsourced data. Ramp Secrete Sharing Scheme is used to fragments the data and T-coloring algorithm is use to allocate that fragment to resist against guessing attack. Auditing algorithm improves integrity are explained in detail. Thus proposed system achieves de-duplication to reduce the storage space utilization and upload bandwidth and improves reliability and integrity. Here we consider only text data as a input for system. In future work is extended to have all type of data including image, multimedia as input to proposed system.

REFERENCES

1. Jin Li, Xiaofeng Chen, Xinyi Huang, Shaohua Tang and Yang Xiang ,"Secure Distributed Deduplication Systems with Improved Reliability", IEEE Transactions on Computers Volume, pp.1-12 ,2015
2. Jin Li, Yan Kit Li, Xiaofeng Chen, Patrick P.C. Lee, and Wenjing Lou ,"A Hybrid Cloud Approach for Secure Authorized Deduplication", IEEE Transactions on Parallel and Distributed systems, vol. 26, pp. 1206-1216, May 2015.
3. Mazhar Ali,Kashif Bilal, Samee U. Khan,,Bharadwaj Veeravalli,Keqin Li,Albert Y. Zomaya,"DROPS: Division and Replication of Data in Cloud for Optimal Performance and Security",IEEE Transactions on Cloud Computing, pp. 1-15, 2015.
4. Jian Liu, Kun Huang, Hong Rong, Huimei Wang, and Ming Xian, "Privacy-Preserving Public Auditing for Regenerating-Code-Based Cloud Storage", IEEE Transactions on Information Forensics and Security, Vol. 10, No. 7, 2015.
5. Backialakshmi.N,Manikandan.M ," Survey based on Secure Authorized Deduplication Hybrid Cloud Approach", IJIRST –International Journal for Innovative Research in Science & Technology, Vo1.1 ,Issue 9, pp. 164-165, 2015.

BIOGRAPHY

Mr. Santosh Ramesh Kadlag is a student in the Computer Engineering Department, RMD Sinhgad School of Engineering, Pune, Savitribai Phule Pune University, Maharashtra, India. He pursuing Master of Computer Engineering (ME) from SPPU, Pune, MS, India. His research interests are Cloud Computing, Networks Security etc…

Prof. Mayur C Akewar is assistant professor in the Computer Engineering Department, RMD Sinhgad School of Engineering, Pune, Savitribai Phule Pune University, Maharashtra, India. He received PG (M.Tech) in Computer Science and Engineering (ME) from Shri. Ramdeobaba College of Engineering & Management, Nagpur, India. His research interest includes Wireless Sensor Networks.