May-June 2020
ISSN: 0193-4120 Page No. 4324-4328

Client side Secure Image Deduplication for Optimized Storage

Prof. Priyanka Bharti[1], Shreya Karbhari[2], Sonia Tripathi[3], Sparsh Gusain[4], Tejeshwini M S[5]

[1,2,3,4,5]School of C&IT, REVA University, Bangalore, India
[1]priyankabharti@reva.edu.in, [2]shreyakarbhari1008@gmail.com, [3]tripathisonia259@gmail.com,
[4]sparshoff2@gmail.com, [5]tejums665@gmail.com

Article Info
Volume 83
Page Number: 4324-4328
Publication Issue:
May - June 2020

Article History
Article Received:*19 November 2019*
Revised: *27 January 2020*
Accepted: *24 February 2020*
Publication: *12 May 2020*

Abstract

We all know the ability of images. "Pictures tell a thousand words". And during this era of digitization which has given the benefit to capture top quality images at every moment by anyone making these pictures to be used as a means of visual communication and expression. As a result it takes great efforts to sort, store and maintain huge amount of high quality images which demands equally huge storage drives. Therefore there is a requirement for a technique which allows the user to save lots of his/her images to the drive, yet serving the aim of an ingenious, cost-effective and proficient storage. Hence this paper meets all the difficulties of having duplicate images in our storage and proposes an answer for an efficient use of space. In this paper, we impart a secure Deduplication scheme for near alike pictures using DICE. It stands for Dual Integrity Convergent Protocol. It presents a method to achieve secure image deduplication at the block level. The numbers of blocks that are stored in the cloud are undersized in number as the larger is the similarity between images.

Keywords**:** *Cloud Computing, Hashcode generation, Image deduplication, Logical Block Addressing, MD5 algorithm, Multilevel Indexing*

1. Introduction

Cloud Computing has developed in recent years as a fantastic innovation to serve a wide scope of big business IT capabilities. The cloud, where providers provide a wide range of foundation services opened by means of an Internet. It is a cost-effective and scalable option for backing up and archiving data. With the advancement in the cloud technology the users are provided with benefits of huge storage, accessibility and availability of data which in turn has led to an increase in the digital data. Thus, management of this data is the major concern for the cloud service providers (CPS). Cloud service providers (CSPs) rely upon deduplication techniques [3][5] to dispense with comparative information and thus decline data transfer capacity, bandwidth and storage requirements. In any case, it is equally significant for CSPs to guarantee the protection and security of client's information. To unwind both these issues, secured data deduplication [4][5]came into picture. Figuring out copy duplicates in the picture and video information is a significant challenge [1].The main purpose of the paper is to provide a storage protocol for images to avoid redundancy [6] of duplicate or similar kind of images in order to achieve efficient memory storage, low cost and increase in uploading speed.

May-June 2020
ISSN: 0193-4120 Page No. 4324-4328

2. Literature Survey

With the advent of cloud computing, within the previous years a lot of studies have been done on the deduplication process and have been determining a more efficient system to forestall duplicate-faking and to give trustworthiness check at customer side. Following are the research have been done on the existing method.

"**DICE: A dual integrity convergent encryption protocol for client-side secure data Deduplication"[1].** This paper was presented by **A. Agarwala, P. Singh and P. K. Atrey**. The prime focus is to evade copy faking and eradication attack and to maintain the integrity at client side and server side.

"Message locked encryption and secure Deduplication"[2]. This paper was presented by **M. Bellare, S. Keelveedhi and T. Ristenpart.** It provides deduplication of data and encryption which is generally in the text form. It also provides definition for privacy and form of integrity. Accordingly, multimedia information, for example, pictures and videos haven't been given a lot of consideration.

"**A Secure data Deduplication scheme for cloud storage"[3].** This was presented by **J. Stanek, A. Sorniotti, E. Androulaki and L. Kencl**. In here, itis based on single popularity of the system which detects files as popular or unpopular, complete privacy is provided with unpopular file. But it has disadvantage that, for popular files complete privacy is not present. Also, deletion of any content is challenging. Hence, we are employing a technique called multilevel indexing for the process of deduplication leading to a reduced computation time. This method also enhances the efficiency of process by faster uploading/downloading speeds.

3. Objectives

The primary reasonof this paper is to expel the copy pictures consequently diminishing the distributed storage and expanding the capacity profiency of the system. The other main objectives of the paper are as follows:

1. Ease the process of identification and accessing of the image
2. Increase in uploading/downloading speeds
3. Reduction in the cost and space required for the image storage

4. Proposed System

1. Architectural Design of Image Uploading

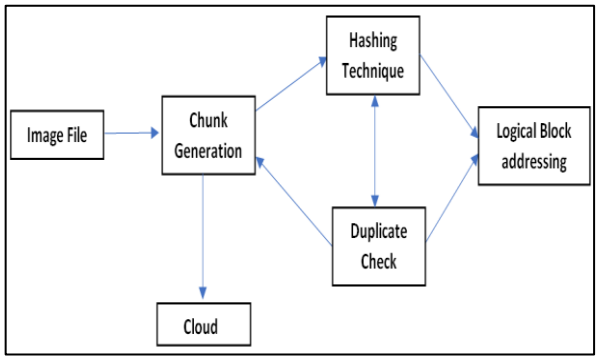


Figure 1: Architecture Design of Image Uploading

2. Architectural Design of Image Downloading

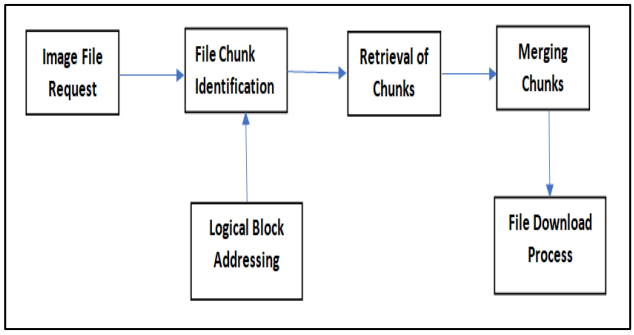


Figure 2: Architectural design of image downloading

The above proposed system consists of two phases, image uploading and image downloading. While uploading an image (Fig 1), the image is divided into chunks, and for every chunk unique hash code is generated using hashing technique. LBA(Logical Block Addressing) is used for the identification of the chunks if already present within the file or not, and it is used as temporary storage to store chunks with respect to the hashcodes. Drive HQ is a cloud storage which is used to store the chunks of image file that are uploaded by the user. Downloading process gets initiated when the user makes a request. While downloading the image (Fig 2), image identification and chunk identification is done using the Logical Block Addressing and the respective chucks of the images are retrieved and merged in the original form.

5. Methodology

The different stages involved in efficiently uploading the images to the cloud are as follows:

Step 1: Uploading raw data

In this paper, image is considered as the raw data. The images can be of the format .jpeg or .png. The user has to

May-June 2020
ISSN: 0193-4120 Page No. 4324-4328

login into his profile to be able to upload the images. He/She has to upload one image at a time.

Step 2: Image Processing

In here, the image file being uploaded is split into varied number of chunks. The image is split using Java image io package. We assign variables which hold values of the number of rows and columns, chunk width and chunk height that the image is going to be divided in. An array is then used to store the image chunks, then initializing the image array with image chunks and hence the image chunk is drawn.

Step 3: Hashing Algorithm

This algorithm is used to calculate a fixed size bit value from the image. The hashing algorithm which is being used in this paper is MD5 message digest algorithm. It is an algorithm that takes input of some random length and gives an output in the form of a message digest which is of length 128 bits. These are the following steps followed in this algorithm-

- Initially, padding is done to the original message in order to make the bit length to 512.
- After padding is done, 64 bits are inserted at the end which is used to record the original input length as modulo 2^64 resulting into a message of length multiple of 512 bits.
- MD5 uses the initial value assigned to a buffer which contains of words - A, B, C, D which are each 32 bits long. These are initiated as:

word A- 01 23 45 67, word B- 89 ab cd
word C: fe dc ba 98, word D: 76 54 32 10

- Using an auxiliary buffer, the contents of the four buffers are mixed and 16 rounds are performed processing the message in a 16-word block.

Then apply the logical operators to the input bits.

F(X,Y,Z) = (X and Y) or (not(X) and Z)
G(X,Y,Z) = (X and Y) or (Y and not(Z))
H(X,Y,Z,) = X xor Y xor Z
I(X,Y,Z) = Y xor (X or not(Z))

Finally, these buffers contain the output beginning with least bit A and terminating with superior bit D.

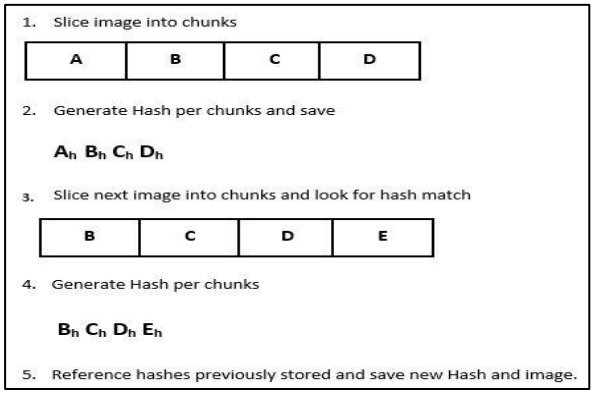


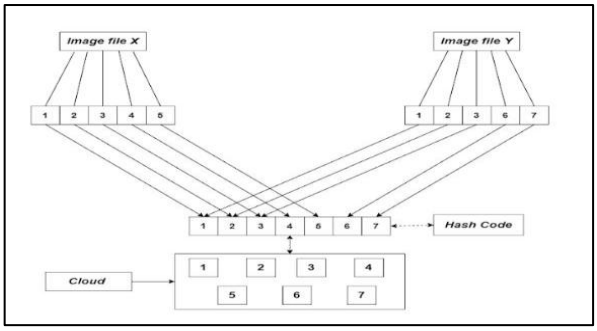
Figure 3: Hash based Deduplication

Step 4: Logical Block Addressing

It is a simple linear addressing format. For each individual chunk, hashcodes are generated after passing through the hashing algorithm- MD5. Every chunk has a unique hashcode. The Logical Block Addressing stores all the chunks against its respective hashcodes. It also logs which chunk belongs to which file. The chunks which are identical to the chunks of any other image, the instance of that particular chunk is increased in the record. The same is the case when the user tries to upload an image that consists a chunk which is pure identical to the chunk which has already been recorded, and then the number of instances in the record is decreased.

Step 5: Deduplication Check

Data deduplication is a technique for wiping out copy duplicates of recursive information. This method is used to reduce capacity utilization and storage. As the process proceeds, a comparison of the chunks of the same/different images is done against the original copy and at whatever point there is a match, the nearness repetitive piece is acknowledged by increasing the number of instances against the stored in the Logical Block Addressing. Multilevel indexing is the technique used to ease the process of comparing the hashcodes of the chunks. For instance, an image consists of two chunks- C1 and C2 with hashcodes-



C1- 26C2290328ACE54D6BC8A6A0F8C69CEE
C2- 2BD01C5F0119FB091C4D3F98DE953C6B

Chunks of another image are of the following hashcodes-

F1 - FF21493BA451E7FCE781167B40DADD6F
F2 - 2AE13531AB62834608208E1D943D4BD4

Chunk F1 is straightaway overruled because it's first character itself does not match. This technique proves to be of great worth as it reduces the access and retrieval time by ounces.

Step 6: Retrieval of images

The downloading of any image in the cloud happens when the user requests for it. In the cloud environment, all the pictures are stored in terms of blocks. The LBA (Logical Block Addressing) accounts for all chunks and its respective hashcodes. Therefore file chunk identification is done in the Logical Block Addressing. The respective chunks of the demanded images are

May-June 2020
ISSN: 0193-4120 Page No. 4324-4328

retrieved. The divided chunks are then merged according to the LBA and thus the original image gets downloaded in the user's machine. Whenever the user from the client side tries to download an image, the LBA (Logical Block Addressing) which accounts for all the chunks that are part of that demanded image serves the purpose.

6. Modules Identified

• Chunk Splitting and Chunk Merging Process

In this module, the images are split into varied chunks. In the process of chuck merging, the client receives the original image when the chunks are merged and downloaded from the cloud.

• Hashing technique

Using MD5 algorithm, the chunks of the images are converted into unique hashcodes. For each and every unique hashcode, keys are generated. Respective< key, value> pairs are formed where value is the respective hashcode. Based on these values it verifies if the chunk is already uploaded to the cloud. If so, its instance is increased in the LBA by mapping to the current block.

• Deduplication check

This happens during the mapping process of the chunks with hashcode. Mapping the whole 128 bits of chunk with hash code to another chunk with hashcode does nothing but increase of the computation time. Hence the hashcode is divided into levels, where first level matches then it moves to second or else discards it. Only unique chunks get uploaded to cloud.

. **• Upload and Download Process**

Within the upload phase, the desired image after splitting into chunks is uploaded to the cloud provided the chunks are unique. While downloading the file, with the help of Logical Block Addressing blocks are downloaded to the server, they are merged and sent to the client in the original form.

7. Result and Discussion

The paper incorporates a technique where it helps the users to achieve image deduplication when uploading it to the cloud instead of manually deleting the image.

- Multiple users will be ready to upload their images into the cloud storage and users can keep track of their logs. In the update stage, clients can insert, modify or erase some blocks of the image files.
- The expected output is the efficient memory storage which means that while uploading identical images, it should occupy the one image size only. For instance, one elephant image is uploaded by user A. Same image is uploaded by user B. If the image size is 100 MB each, then only 100 MB of space is consumed rather than occupying 200 MB-double the size. As a result, reduction of the occupied space.
- The multilevel indexing technique which takes place during mapping of hashcodes. The hashcode is divided and compared into levels which results in increased computation speed and also increased efficiency of the system.

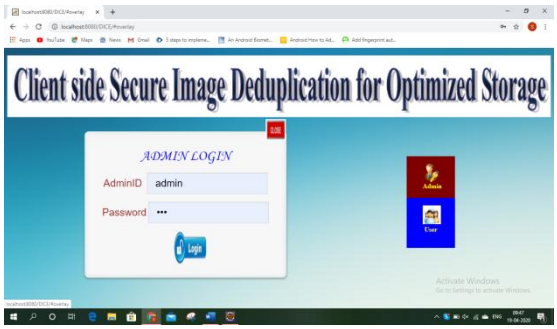


Figure 4: Login page of Admin

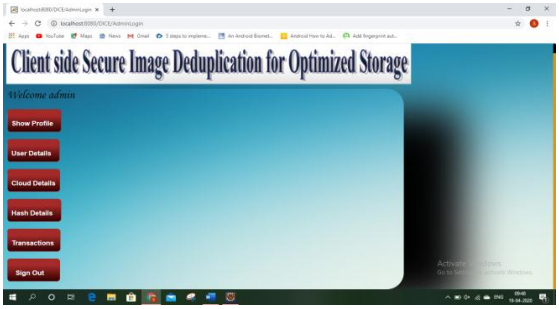


Figure 5: Admin's Profile

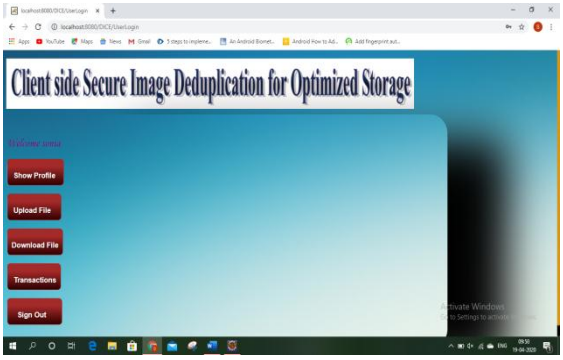


Figure 6: Users Profile

8. Conclusion and Future Scope

In this paper, a strategy to perform secure image deduplication at the block level provided supporting the DICE convention. We conclude that, the more is the similitude between the images, the smaller number of blocks gets stored at the cloud i.e., only unique image chunks get uploaded in the cloud resulting in less storage consumption. This paper presents faster uploading/ downloading speeds and efficiency compared to the previous works.

Future scope of this system is to apply various operation on picture like cropping, resizing, filter effects, background and templates, lighting conditions with

May-June 2020
ISSN: 0193-4120 Page No. 4324-4328

assorted document arranges. Also, to make the user capable of uploading multiple images at the same time.

References

[1] A. Agarwala, P. Singh, and P. K. Atrey, "DICE: A dual integrity convergent encryption protocol for client side secure data deduplication," in *IEEE International Conference on Systems, Man, and Cybernetics, Banff, Canada, 2017, pp.* 2176–2181.

[2] M. Bellare, S. Keelveedhi, and T. Ristenpart, "Message locked encryption and secure deduplication," in *Advances in Cryptology – 32nd Annual International Conference on the Theory and Applications of Cryptographic Techniques*, Athens, Greece, 2013, pp. 296–312.

[3] J. Stanek, A. Sorniotti, E. Androulaki, and L. Kencl, "A secure data deduplication scheme for cloud storage," in *Financial Cryptography and Data Security, Berlin, Heidelberg, 2014*, pp. 99–118.

[4] D. Koo, J. Hur, and H. Yoon, "Secure and efficient deduplication over encrypted data with dynamic updates in cloud storage," in *Frontier and Innovation in Future Computing and Communications, Dordrecht, 2014, pp. 229–235*.

[5] M. W. Storer, K. Greenan, D. D. Long, and E. L. Miller, "Secure data deduplication," in *Proceedings of the 4th ACM International Workshop on Storage Security and Survivability*, Fairfax, Virginia, USA, 2008, pp. 1–10.

[6] J. R. Douceur, A. Adya, W. J. Bolosky, P. Simon, and M. Theimer, "Reclaiming space from duplicate files in a serverless distributed file system," in *The 22nd International Conference on Distributed Computing Systems, Vienna, Austria*,,2002, pp. 617–624.