ISSN: 2321-7782 (Online)
Impact Factor: 6.047

Volume 4, Issue 5, May 2016

International Journal of Advance Research in Computer Science and Management Studies

Research Article / Survey Paper / Case Study

Available online at: www.ijarcsms.com

Multimedia and Text Content Protection through Simple Signature Generation in Cloud Environment

Swathi Sridharan[1]
MTech CMRIT
Bengaluru – India

Swathi Y[2]
CSE CMRIT
Bengaluru – India

Abstract: ***A new approach to protect our videos and large text documents is discussed here. Our design aims to provide a security for your content (here content refers to videos, multimedia and large text documents) in the online world so that no other person would copy the same and claim it as there content. The system is cloud based all the computing of the content is done in the cloud. This system has three main components (i)Signature Generation of the video , large text files or any document (ii) Comparison of the signature of your reference content with the other content which might possible been copied from the original content , (iii)Possible outcome of success or failure is known and for any altered content level of copy can be detected The comparison is fast and is in the cloud .Hence our model not only can detect any pirated content but also gives level of copy for any altered content in an online process.***

Keywords: Videos, images, multimedia, large text documents, Signature Generation, level of copy, cloud, online, content protection.

I. INTRODUCTION

Latest developments technologies as well as the availability of online free hosting sites have made it very easy to duplicate copyrighted materials such as videos, images, important documents and music clips files. Illegally redistributing these contents over the Internet will result in great loss in terms of revenues for content owners. Finding these illegally made copies over the Internet is a very complex and expensive task as the volume of data available over the internet is huge and comparing content to match and identify copies is very complex.

Here we present a novel system for any file content protection on cloud infrastructures. The system can be used to protect various multimedia content types like videos, images, music or any large text files. The system can run on private clouds, public clouds, or any combination of public-private clouds. The design is cost effective because it uses the computing resources on demand. The design can be scaled up and down to support varying amounts of file's content being protected.

The contributions of this paper are as follows.

- Complete multi-cloud system for multimedia content protection. The system can support different types of file content and can effectively use varying computing resources.
- A novel method for generating signature for any multimedia or large text files which is simple and effective.

II. EXISTING SYSTEM/RELATED WORK

The problem of protecting various types of multimedia content has attracted significant attention from academia and industry.

One approach to this problem is using watermarking , in which some distinctive information is embedded in the content itself and a method is used to search for this information in order to verify the authenticity of the content.

Watermarking requires inserting watermarks in the multimedia objects before releasing them as well as mechanisms/systems to find objects and verify the existence of correct watermarks in them. Thus, this approach may not be suitable for already-released content without watermarks in them.

Signatures or fingerprints extraction technique is another important one. Signatures are also created from query (suspected) objects downloaded from online sites. Then, the similarity is computed between original and suspected objects to find potential copies.

Many previous works proposed different methods for creating and matching signatures. These methods can be classified into four categories: spatial, temporal, color, and transform-domain. Spatial signatures (particularly the block-based) are the most widely used. However, their weakness is the lack of resilience against large geometric transformations. Temporal and color signatures are less robust and can be used to enhance spatial signatures. Transform-domain signatures are computationally intensive and not widely used in practice.

III. PROPOSED SYSTEM

The proposed system uses spatial signature techniques.

There are two main components, the cloud server and the clients. The client users are given provision to store their files in the cloud. But how safe will their data or copy write contents be from any third party ? Chances are that the files can be viewed and copied.

To prevent this copying and duplication of data we have come up with a simple content protection method that can assure the user that no matter who views the file, they will not be able to replicate the data in the system.

Example. You tube videos can be seen by everyone but our system will not allow any two users to have the same videos. The credit always goes to the owners and not anyone else.

The working of the system is simple and can be explained as follows:

The client users upload and save their files in the server, during this process the unique signatures of that file is generated and stored in another file which is placed in the server. When another user also uploads any file signatures are generated for his file too and before it is placed in the server , the signatures are compared with the signature files that are already placed in the server to know the potential copied file. If the signature is not matching with any file then the file is said to be unique and its signature copy is placed in the server. If suppose the signature matches to any one file already in the server then the level or percentage of copy is determined. The file's acceptance is determined based on the user permissible level of copy. If the level of copy is higher than the user set limit then the file cannot be uploaded into the server.

The process of signature generation is fairly easy and can be determined as follows:

The client user's file is divided equally based on the size, say one kilobyte each. These small files are given to the algorithms to generate the ASCI equivalent of the bits, this ASCI code serves as an input to the MD5 algorithm that reduces the size of signature generated to a standard uniform length.

This generated signatures are stored in the signature file representing that particular file. This file is stored in the cloud server if the generated set of signatures are unique and do not match with any other set of signatures in any other file in the cloud. If suppose the set of signatures match then they are not stored into the cloud as there is already a copy existing in the cloud, allowing this file to be stored in the cloud is like allowing multiple copy of the files to be stored which is considered as wastage of precious memory.

The algorithm used to generate the signatures in this system is unique and the supporting algorithm used are the MD5 algorithm to maintain the uniformity in the signature and the KMP algorithm for the string matching techniques.

The algorithms used are as follows:

A. MD5 Algorithm:

The MD5 message-digest algorithm is a largely used cryptographic hash perform producing a 128-bit hash worth, as a rule expressed in text layout as a 32-digit hexadecimal number. MD5 has been utilized in a vast kind of cryptographic applications and can also be mostly used to verify information integrity.

MD5 is a one-method operate, it is neither encryption nor encoding. It are not able to be reversed other than by using brute-force attack. MD5 techniques a variable-size message into a fixed-length output of 128 bits. The enter message is broken up into chunks of 512-bit blocks the message is padded so that its size is divisible by way of 512.

The padding works as follows:

First a single bit, 1, is appended to the end of the message. This is followed by as many zeros as are required to bring the length of the message up to 64 bits fewer than a multiple of 512. The remaining bits are filled up with 64 bits representing the length of the original message, modulo 2^{64}.

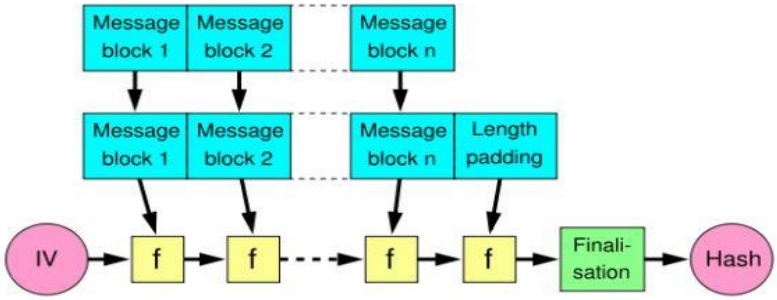

Figure 1 : Padding of messages

Example the text India-Bengaluru is coded as “4200ea5fbb204c4f32b92ce8bf092350” in MD hash.

One of the main uses for cryptographic hashing is for verifying the contents of a message or file after transfer. Another use for hashes is in the storage of passwords. Storing passwords as clear text is a bad idea, for obvious reasons so instead they are converted to hash values.

When a user inputs a password it is converted to a hash value, and checked against the known stored hash. As hashing is a one-way process, provided the algorithm is sound then there is theoretically little chance of the original password being deciphered from the hash.

B. KMP Algorithm:

The Knuth–Morris–Pratt string shopping algorithm (KMP algorithm) searches for occurrences of a "word" say W within a primary "text string" say S with the aid of using the commentary that after a mismatch happens, the word itself embodies enough knowledge to assess where the following match would start, for that reason bypassing re-examination of beforehand matched characters.

A string matching algorithm wants to find the starting index in string that matches the search word. The algorithm looks for a character match at successive values of the index, the position in the string being searched. If the index reaches the end of the string then there is no match, in which case the search is said to "fail".

At each position the algorithm first checks for equality of the first character in the word being searched. If a match is found, the algorithm tests the other characters in the word being searched by checking successive values of the word position index. The algorithm retrieves the character in the word being searched and checks for equality of the expression. If all successive characters match at any given position, then a match is found at that position in the search string.

IV. WORKING OF THE SYSTEM

The working of our system can be explained considering an example:

Consider any file of 100kb size,(here for the example lets consider a text file). This file say File1.txt is owned by a user User1. The duplicate copy of this file File2 is owned by User2 with a copy level of 70%.

When the User1 uses the cloud service, he uploads his file through a client portal . His file gets accepted is there was no previous copy of the file in the cloud server. This process can be explained in the following steps:

1. User 1 logs in through client portal with a valid email ID .
2. The server is open for the connections.
3. He uploads the File1 through the portal.
4. The background process is as follows:
    - The file gets transferred in bits over the internet transportation layer. Here the bits are divided into blocks of same size say 1kb each . So a 100kb file will have 100 blocks each block containing equal amounts of bits.
    - These block's contents are converted into ASCI values of their equivalent bits.
    - Now all the blocks will contain information in the ASCI level.
    - This ASCI values act as signatures but they are too long for a signature, hence they are parsed by the MD5 algorithm.
    - As we know the MD5 algorithm takes in 128 bits at a time and generate a 32 bit uniform signature.
    - The signature generation happens in a parallel distributed way where in every kb generates a signature in parallel (level of parallelism depends on the machine)
    - The signatures generated will be stored in the file say File1.Signature.
    - This Signature file will be stored in the cloud in case of a genuine file is being uploaded only.
    - This signature generation happens in an intermediate part of the cloud and only valid files are uploaded to the main cloud.
    - This signature file is used to check against the duplication files when stored in the main cloud using the KMP algorithm.
5. The owner of the file File1 gets a success message once his file has been validated and stored in the cloud.
6. An email is sent to the owner's ID stating the success of the file and its genuinely.
7. Now when the User2 tries to upload the File2 with about 70% copy from the original file, he follows the same steps from 1 to 3.
8. The background process is follows:
    - The file is again divided into small chunks of blocks and follows the same process of signature generation.
    - As we know the signature generation happens in the intermediate part of the cloud, the signature comparison also happens in this level only. Only when the validity is tested and obtained the file will be uploaded to the main cloud.

- The signature comparison happens with all the files that are there in the cloud in a parallel distributed way , this speeds up the process of checking the files in case there are many file that need to be compared.
- If there is a match of signatures then the file will not be uploaded to the main cloud and a failed notification is sent to the user uploading the file.

9. When the file has been proved to be copied by the level of copy feature, the user receives a notification saying the file uploading failed as there is a original copy that resides in the server .
10. The owner of the original file is sent a notification and an email that his file is being duplicated by the other user with the mentioned level of copy.
11. Hence duplicate file cannot be uploaded to the server.
12. The original file owners are notified whenever anyone copies their content and tries to upload them as their own. And the user who is trying to upload the file also gets a mail stating as to who owns the original copy of the file.
13. Multiple clients can communicate with the server at the same time.
14. Any user cannot login into the client portal more than once in the client portal, if so , a notification saying "This user has already been logged in " is displayed.
15. Only authorized users are allowed to access the cloud through the client portal , there cannot be any users trying to access the cloud without the client portal. In such case the access is denied.

V. IMPLEMENTATION

This paper is implemented and running successfully. The following screenshots are obtained while this system was running.

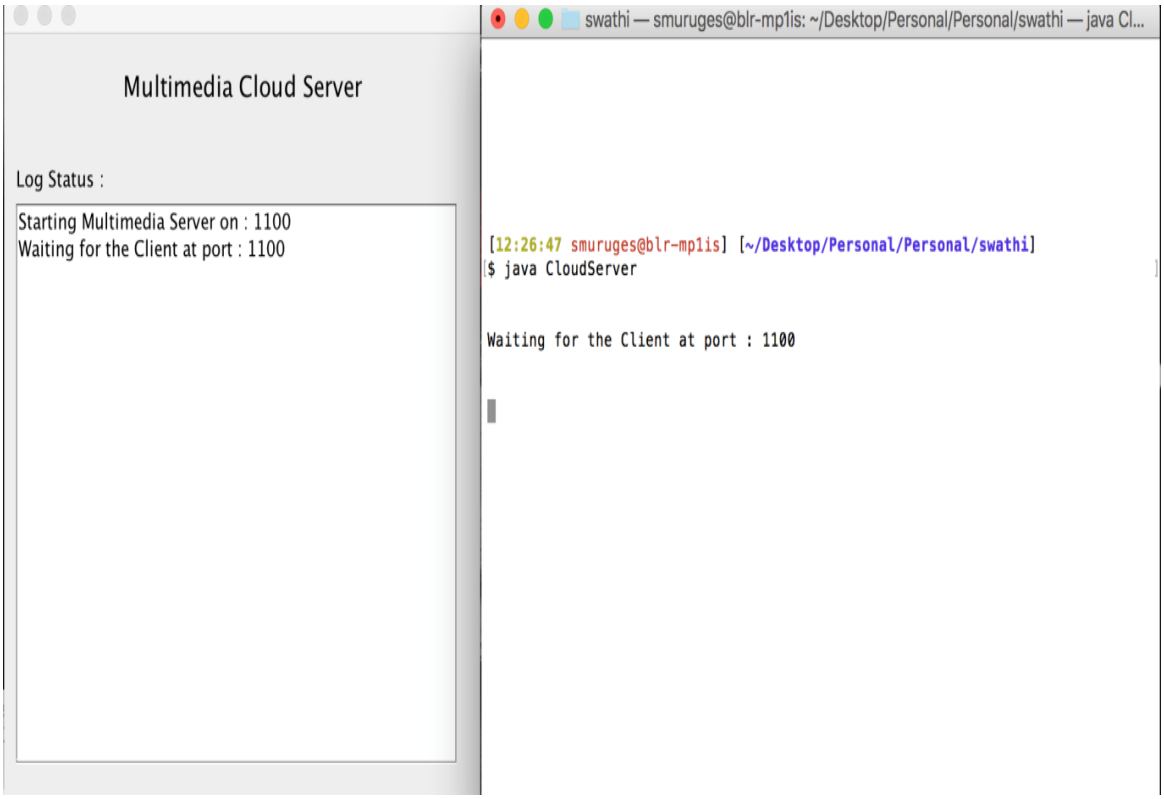

Figure 2: Cloud Server

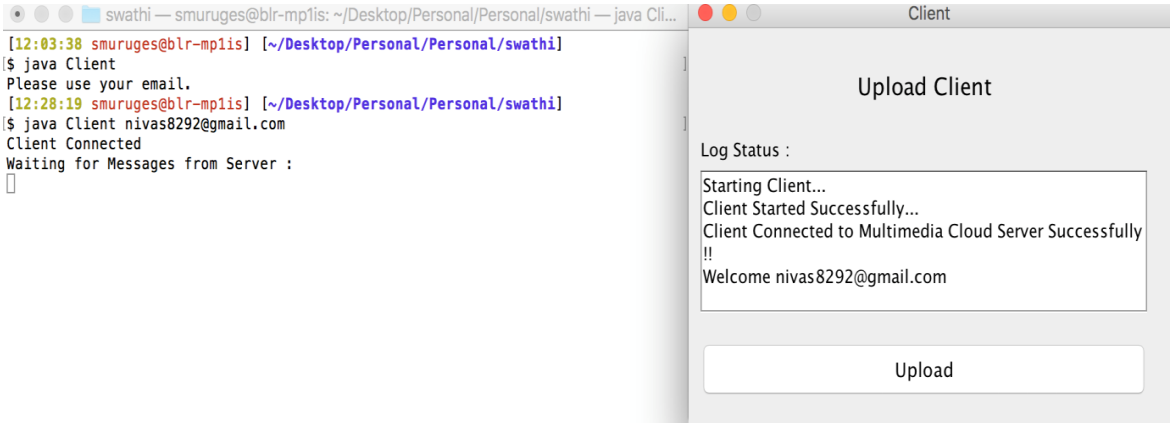

Figure 3: Clients

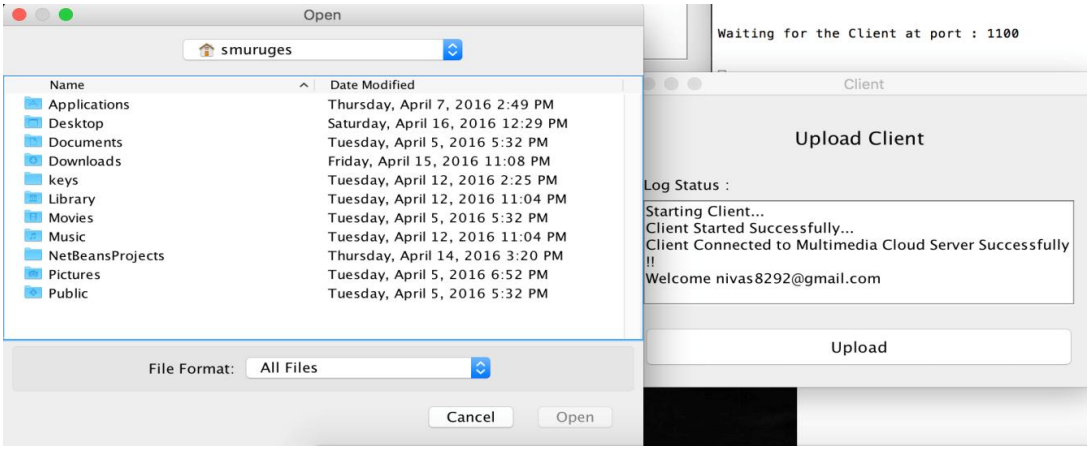

Figure 4: Clients uploading the file

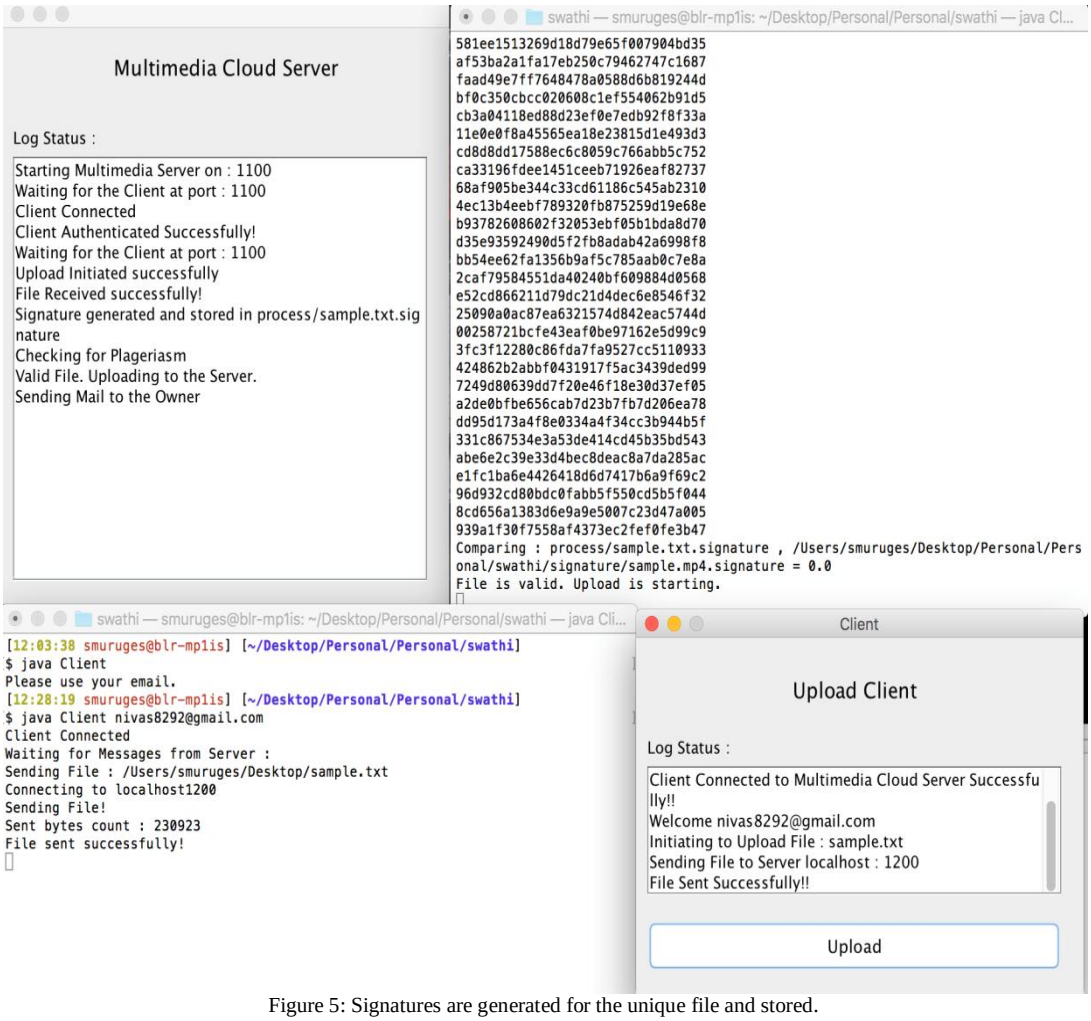

Figure 5: Signatures are generated for the unique file and stored.

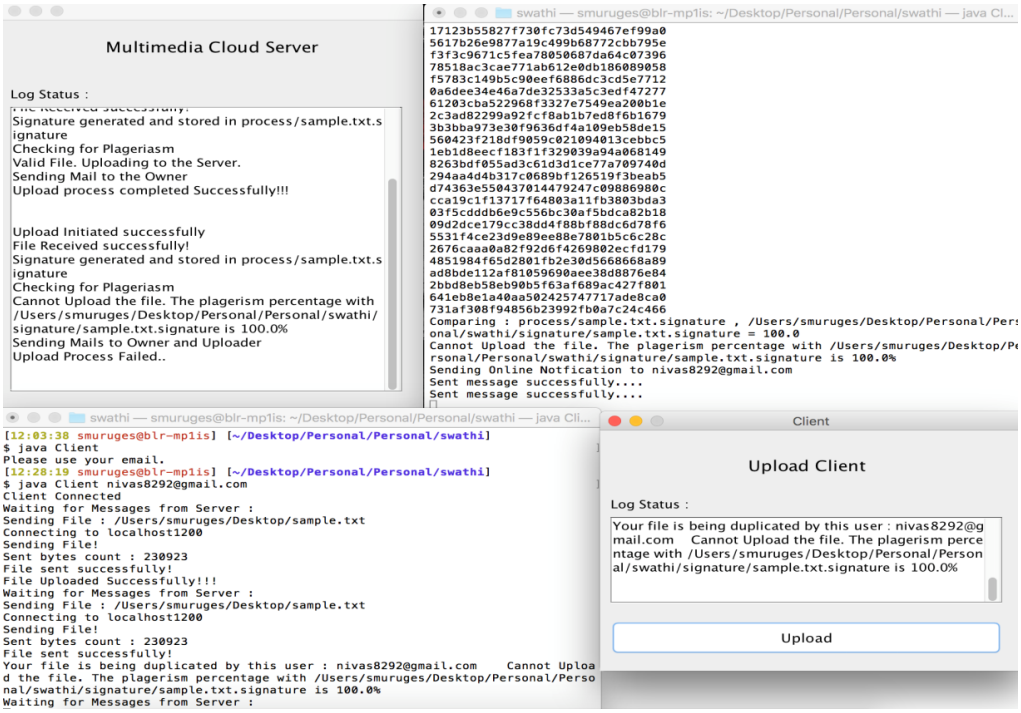

Figure 6: Signatures generated for the copied file and file cannot be uploaded, The level of copy is determined.

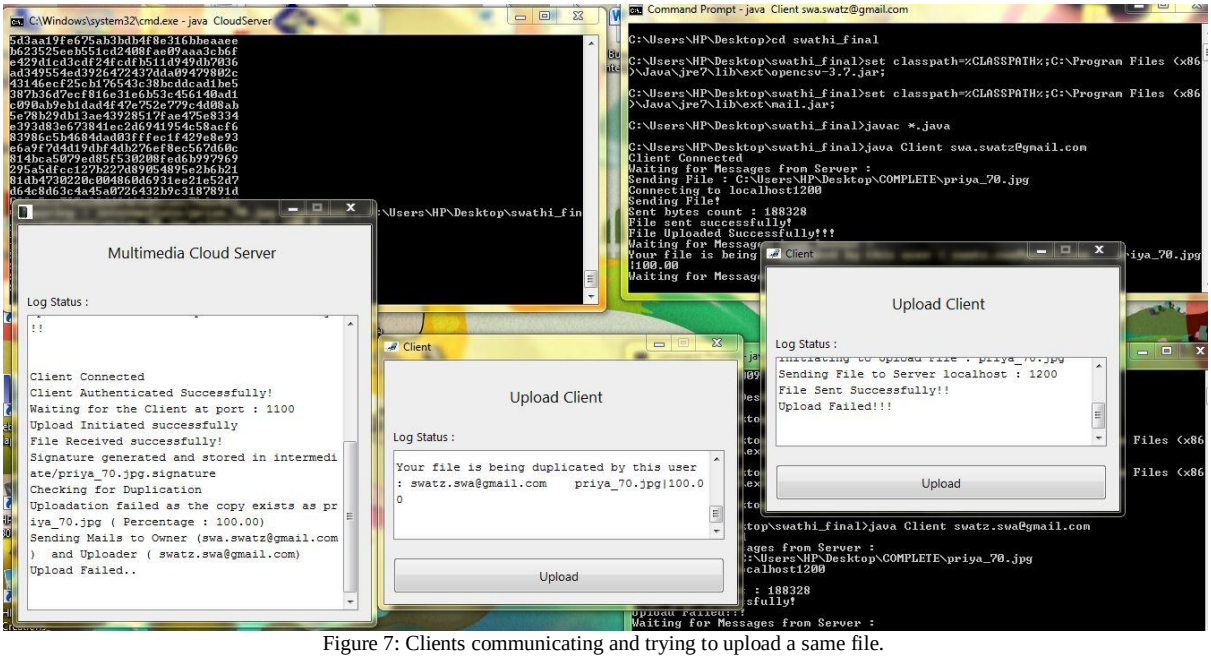

Figure 7: Clients communicating and trying to upload a same file.

VI. CONCLUSION

This system is used to detect and prevent the possibilities of any other third party trying to violate any copyrighted material. The Material here refers to any type of file like a text doc, video, images, etc. The system intends to protect the contents of any file irrespective of the type of file for files are available on a cloud platform. The contents of the file cannot be replicated, If the contents are copied then it should not be allowed to be uploaded into the cloud facility. Unique technique is employed to protect the contents of the file.

Signatures are generated for every file which is unique to that file. Signature generation and comparison uses well known algorithms effectively. The algorithms used are simple and effective. The duplicate copies of the file contents are determined by analyzing these signatures. The level or the percentage of copy is determined from which the contents were copied.

The owner of the original file is notified when anyone tries to use his contents. The cloud facility is made secure and only unique copies of the files can exists in the system at any point of time. Copied or duplicate contents are blocked form uploading

them to the cloud facility. The cloud resources are provided by the cloud service provider and the users data is available during anytime. The uploaded users file is safe and secure in the cloud facility. By using this technique the owners of the files can be at peace knowing that their data is safe and non-replicable any time by anyone.

ACKNOWLEDGEMENT

I take this opportunity to express my gratitude to all those people who have been directly and indirectly with me during the completion of this thesis.

I pay special thanks to my guide Associate Professor and HOD-CSE Mrs Swathi Y who was very supportive and compassionate throughout the preparation of this thesis. Finally I would like to acknowledge my profound gratitude to Almighty and my family. It would not have been possible to accomplish this without their blessings and their constant support.

References

1. Hampapur, K. Hyun, and R. Bolle, "Comparison of sequence matching techniques for video copy detection,",2011.
2. N. Khodabakhshi and M. Hefeeda, "Spider: A system for finding 3D video copies,", 2014.
3. H. Liao, J. Han, and J. Fang, "Multi-dimensional index on hadoop distributed file system,",2013.
4. J.Lu, "Video fingerprinting for copy identification: From research to industry applications,"2013.
5. W. Lu, Y. Shen, S. Chen, and B. Ooi, "Efficient processing of knearest neighbor joins using MapReduce,",2011.
6. V. Ramachandra, M. Zwicker, and T. Nguyen, "3D video fingerprinting,",2013.
7. A.Stupar, S. Michel, and R. Schenkel, "Rank reduce– processing k-nearest neighbor queries on top of mapreduce, 2012.
8. K. Tasdemir and A. Cetin, "Motion vector based features for content based video copy detection",2011.
9. Securing Digital Video: Techniques for DRM and Content Protection – Eric Diehl, 2012.
10. Multimedia Systems and Content-based Image Retrieval, Sagarmay Deb, 2012.

AUTHOR(S) PROFILE

Ms. Swathi Sridharan, Completed Bachelors degree in Computer Science and Engineering from Global Academy of technology, Bengaluru. With a prior work experience as a Systems Engineer at Infosys Technologies Pvt Ltd from 2011 to 2012 and as a Research Assistant at Nanyang Technological University (NTU) , Singapore from 2012 to 2013. And currently pursuing Masters in technology in Computer Since and Engineering, at CMR Institute of Technology.

Mrs. Swathi Y, is currently working as Associate Professor and Head of the Department in Computer Science, CMRIT from past 8 years. She worked in SAPLABs as BI consultant for 5years and currently pursuing Ph.D in the area of Network Security. She has completed her M.Tech from VTU and her research areas are Network Security, Game Theory, Algorithms.

