Online-Cloud Based Data Security

Anjali Dange[1], Satish Sahare[2], Shubham Amnerkar[3], Utkarsh Fopare[4], Vijay V. Chakole[5]

Electronics Department, K.D.K.C.E, Nagpur[1,2,3,4,5]

Abstract: We propose a scheme to deduplicate encrypted data stored in cloud based on ownership challenge and proxy re-encryption. It integrates cloud data deduplication with access control. We evaluate its performance based on extensive analysis and computer simulations. The superior efficiency and effectiveness of the scheme for potential practical deployment, especially for big data deduplication in cloud storage is clearly seen in and around the scope of realization of the concept of cloud security

Keywords: Re-encryption, Simulations, Cloud computing, Data storage, Data deduplication

I. INTRODUCTION

Data deduplication is one of the hottest technologies in storage right now because it enables companies to save a lot of money on storage costs to store the data and on the bandwidth costs to move the data when replicating it offsite for DR. If you can deduplicate what you store, you can better utilize your existing storage space, which can save money by using what you have more efficiently. If you store less, you also back up less, which again means less hardware and backup media. The business benefits of data deduplication include:

If you store less, you also send less data over the network in case of a disaster, which means you save money in hardware and network costs over time.

If you intend to move large amounts of data over a network and provide access to that data as a service, you need to be cognizant of network bandwidth requirements, data security and the total IT costs of providing those services to end users, especially when providing services for data storage and DR protection.
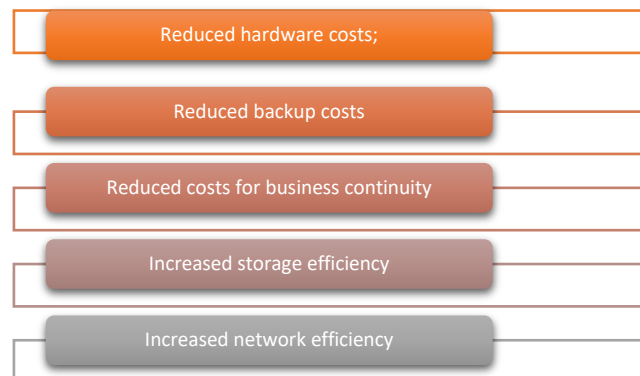
II. AIM AND OBJECTIVE

A scheme based on data ownership challenge and Proxy Deduplication to manage encrypted data storage with deduplication can be realized and we aim to solve the issue of deduplication in the situation where the data holder is not available or difficult to get involved.

We motivate to save cloud storage and preserve the privacy of data holders by proposing a scheme to manage encrypted data storage with deduplication. Our scheme can flexibly support data sharing with deduplication even when the data holder is offline, and it does not intrude the privacy of data holders. We propose an effective approach to verify data ownership and check duplicate storage with secure challenge and big data support.

We integrate cloud data deduplication with data access control in a simple way, thus reconciling data deduplication and encryption.

We prove the security and assess the performance of the proposed scheme through analysis and simulation. The results show its efficiency, effectiveness and applicability. Objective is an effective and flexible distributed scheme with explicit dynamic data support to ensure the correctness of users' data in the cloud. whenever data corruption has been detected during the storage correctness verification, our scheme can almost guarantee the simultaneous localization of data errors, i.e., the identification of the misbehaving server(s).

1. Compared to many of its predecessors, which only provide binary results about the storage state across the distributed servers, the challenge-response protocol in our work further provides the localization of data error.

2. Unlike most prior works for ensuring remote data integrity, the new scheme supports secure and efficient dynamic operations on data blocks, including: update, delete and append.

3. Extensive security and performance analysis shows that the proposed scheme is highly efficient and resilient against Byzantine failure, malicious data modification attack, and even server colluding attacks.
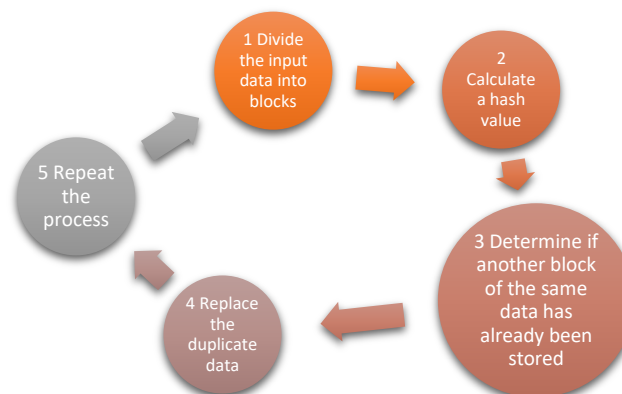
III. PROPOSED WORK

Data deduplication works by comparing objects (usually files or blocks) and removes objects (copies) that already exist in the data set. All the processes which are not unique are removed in this method.
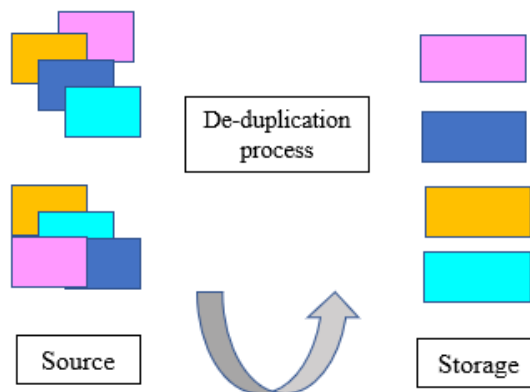
In Data deduplication method we divide the input data into blocks and a hash value is calculated for each of these blocks. Then using these hash values, we can determine whether another block of same data has already been stored. If a similar data file is found then replace the duplicate data with a reference to the object already present in the database.

IV. FLOW CHART

In simplified terms, data deduplication compares objects (usually files or blocks) and removes objects (copies) that already exist in the data set. The deduplication process removes blocks that are not unique. Simply put, the process consists of four steps:

Once the data is chunked, an index can be created from the results, and the duplicates can be found and eliminated. Only a single instance of every chunk is stored. The actual process of data deduplication can be implemented in a
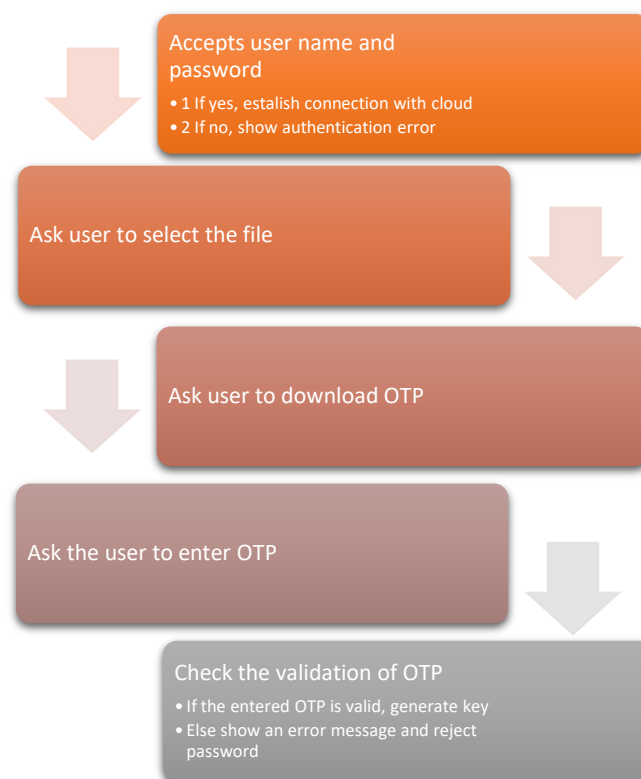
number of different ways. You can eliminate duplicate data by simply comparing two files and making the decision to delete one that is older or no longer needed. But real deduplication solutions use more sophisticated methods, where the actual math involved can make your head spin.

More intelligent file-level deduplication methods actually look inside individual files and compare differences within the files themselves, or compare updates to a file and then just store the differences as a "delta" to the original file. File versioning associate's updates to a file and just stores the deltas as other versions. File-based hashing actually creates a unique mathematical "hash" representation of files, and then compares hashes for new files to the original. If there is a hash match, you can guarantee the files are the same, and one can be removed.

A lot of backup applications have the versioning capability, and you may have heard it called incremental or differential backup. Some backup software options (Tivoli Storage Manager is a good example) always use the versioning method to speed backup. You do a full back up the first time, and from then on, only the changes in the data need to be stored. IBM calls this "progressive" backup.

Other software solutions use similar techniques to reduce wide area network (WAN) requirements for centralized backup. Intelligent software agents running on the client (desktop, laptop or workstation) use file-level versioning or hashing at the client to send only delta differences to a central site. Some solutions actually send all data updates to the central site and then hash the data once it arrives, storing only the unique data elements.

V. ALGORITHM

Accepts user name and password
• 1 If yes, estalish connection with cloud
• 2 If no, show authentication error

Ask user to select the file

Ask user to download OTP

Ask the user to enter OTP

Check the validation of OTP
• If the entered OTP is valid, generate key
• Else show an error message and reject password

An example of this method is comparing the name, size, type and date-modified information of two files with the same name being stored in a system. If these parameters match, you can be pretty sure that the files are copies of each other and that you can delete one of them with no problems. Although this example isn't a foolproof method of proper data deduplication, it can be done with any operating system and can be scripted to automate the process, and best of all, it's free. Based on a typical enterprise environment running the usual applications, you could probably squeeze out between 10 percent to 20 percent better storage utilization by just getting rid of duplicate files.

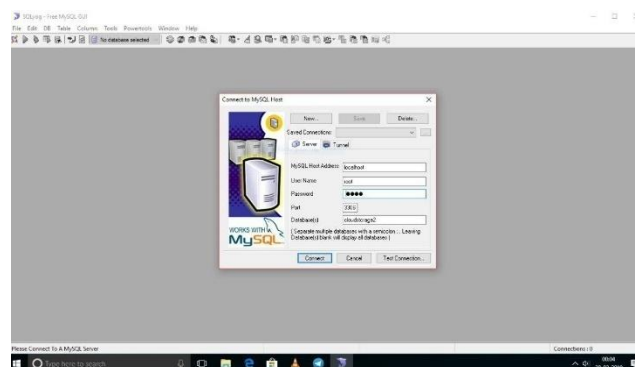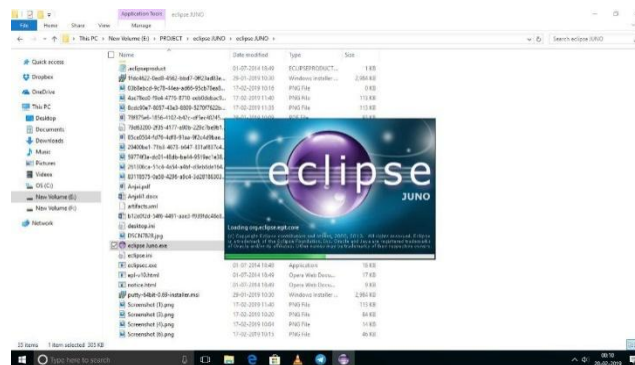## VI.SYSTEM REQUIREMENTS & EXPERIMENTAL SCREENSHOTS

Hardware

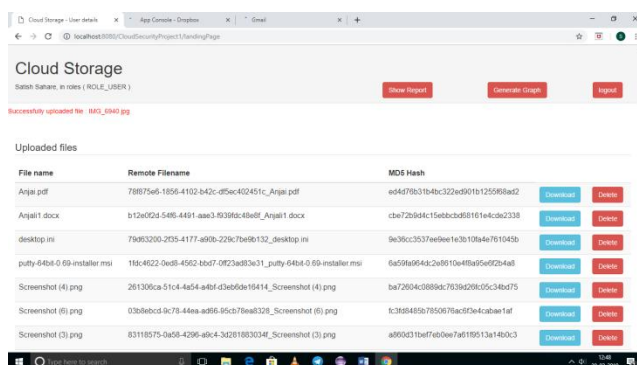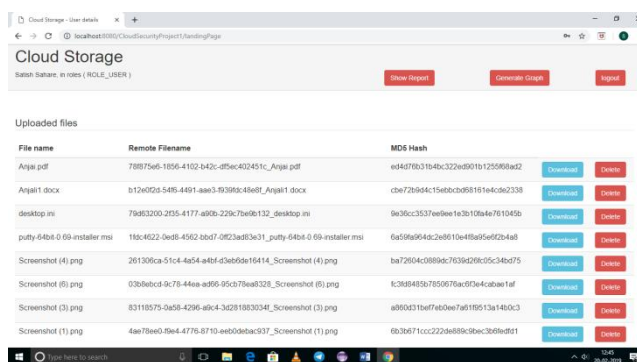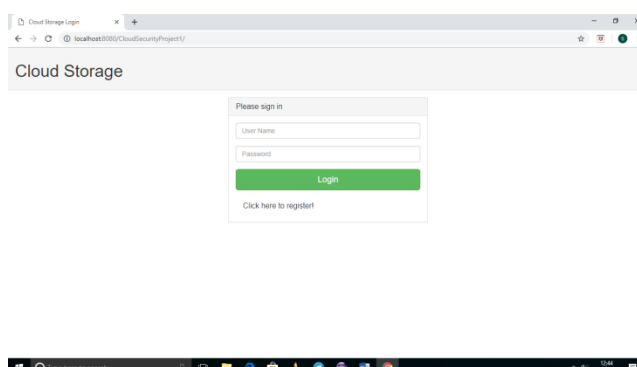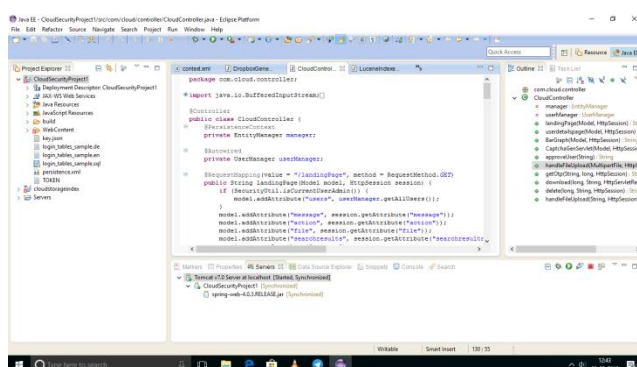| Number	Description	Alternatives (If available)
1	PC with 100 GB hard-disk and 2 GB RAM	Not-Applicable
	PCs in Network	

Fig 3. Hardware specifications

Software

Number	Description	Alternatives (If available)
1	Windows 7/8/XP with MS-office	Not Applicable
2	Java (1.7),	Java 1.6,
3	Eclipse Juno	Neat Bean 7.4
4	DropBox Cloud	
5	Tomcat server 7	
6	MySQL Database Server 5.5	

Fig 4. Software specifications

IJARCCE

International Journal of Advanced Research in Computer and Communication Engineering
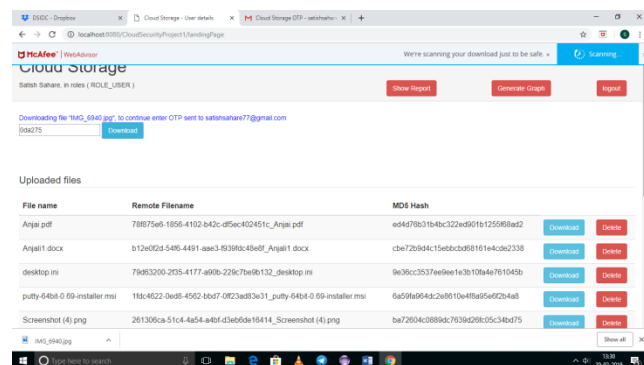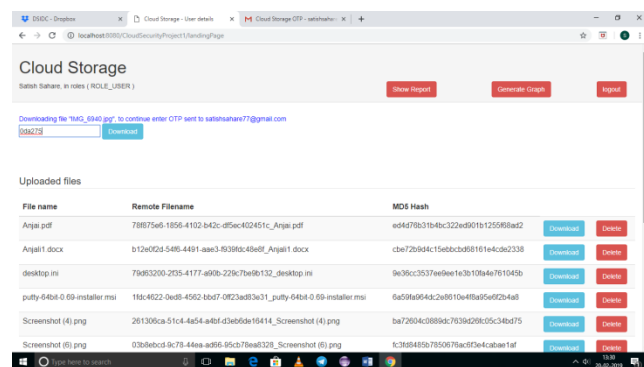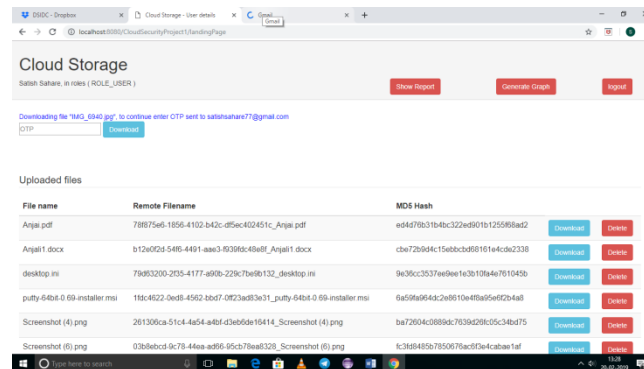
V. CONCLUSION

In this current work the problem of data security in cloud data storage has been investigated. To ensure the correctness of client data in cloud data storage, the proposed method encrypts data and stores it in cloud and user is allowed for modification of data. Through data security and performance analysis, it shows that work is highly efficient and avoid unauthorized users from accessing the data.

The commercial cloud storage services are in favor of the cross-user client-side fixed-size-chunk-level data deduplication to reach the highest deduplication gain. In particular, the cross-user data deduplication sees the cloud storage as a virtually single disk shared by all of the cloud users, deduplicating all of the files from different users and consequently maximizing the deduplication opportunity. With the results and analysis following conclusions are made deduplication opportunity.

REFERENCES

[1] Youssef Gahi, Mouhcine Guennoun, Hussein T. Mouftah,"Big Data Analytics: Security and Privacy Challenges", IEEE Symposium on Computers and Communication (ISCC), Messina, Italy, June 2016, pp 15-17.

[2] Laila Fetjah, Karim Benzidane, Hassan El Alloussi Othman El Warrak, Said Jai- Andaloussi," Toward a Big Data Architecture for Security Events Analytic", IEEE 3rd International Conference on Cyber Security and Cloud Computing , Beijing, China,2016,pp 1-7.

[3] Natalia Miloslavskaya and Aida Makhmudova, "Survey of Big Data Information Security", 4th International Conference on Future Internet of Things and Cloud Workshops, Vienna, Austria, Aug 2016,pp 4-9.

[4] Suliman A. Alsuhibany," A Space-and-Time Efficient Technique for Big Data Security Analytics", vol. 46, no. 2, Riyadh, Saudi Arabia ,pp.241-284, 2016.

[5] Hai-ting Cui, "Research on the Model of Big Data Serve Security in Cloud Environment", First IEEE International Conference on Computer Communication and the Internet, Wuhan, China,Oct 2016, pp 1-16.

[6] Cong Wang, Qian Wang, and Kui Ren," Ensuring Data Storage Security in Cloud Computing", US National Science Foundation,2015, pp 1-4

[7] Raj Kumar, "Research on Cloud Computing Security Threats using Data Transmission" International Journal of Advanced Research in Computer Science and Software Engineering, India Volume 5, Issue 1, January 2015, pp. 399-402.

[8] Saakshi Narula," Cloud computing security: Amazon web service", Fifth International Conference on Advanced Computing & Communication Technologies, Haryana, India,Feb 2015,pp 699-703

[9] Changyou Guo and Xuefeng Zheng "The Research of Data Security Mechanism Based on Cloud Data Storage Architecture", International Journal of Security and Its Applications, China, Vol. 9, No. 3 (2015), pp. 363-370

[10] Ahmed Dheyaa Basha, Irfan Naufal Umar, and Merza Abbas," Storage in cloud as a module", International Journal .30,Vol. 4, No. 1, January 2014, pp 160-163

[11] Yunchuan Sun, and Junsheng Zhang," Data Security and Privacy in Cloud Computing", International Journal of Distributed Sensor Networks ,Japan, Volume 5,2014,pp 9-20.

[12] Buyya," Introduction to module interactions in a system on Cloud Computing",IEEE transactions on cloud computing, Italy, vol. 1, no. 1, january-june 2013,pp 2424-2444

[13] Zohreh Sanaei, "Heterogeneity in Cloud Computing: Taxonomy and Open system modules",Bellingham, IEEE Communications Surveys & Tutorials, Vol. 16, No. 1, 2013, pp 40-45.

[14] Young-Sae Song ,"Storing Big Data- The Rise of the Storage Cloud",Maldives,pp. 504- 511, 2012.

[15] D. L. G. Filho and P. S. L. M. Barreto, "Demonstrating Data Possession and Uncheatable Data Transfer" Italy, Cryptology ePrint Archive, Volume 78, Issue 5, September 2012, pp 1345-1358

[16] Intel IT center, "Peer Research Big Data Analytics "Intel's IT Manager Survey on How Organizations Are Using Big Data", USA , Aug 2012, pp 46-55.

[17] Vahid Ashktorab and Dr. Kamran Zamanifar ," A Survey on Cloud Computing and Functional Solution Providers", International Journal of Application or Innovation in Engineering & Management (IJAIEM), Iran ,Vol 1, issue 2,Oct 2012, pp 1191-1201.

[18] T. S. J. Schwarz and E. L. Miller, "Store, Forget, and Check: Using Algebraic Signatures to Check Remotely Administered Storage," Egypt, Proc. of ICDCS '06, pp. 12–12, 2011

[19] I.Somervile, Software engineering, 9th edition, vol.ISBN 9780 1370 35151.Adison Wisley publishing company, 2010

[20] Q. Wang, K. Ren, W. Lou, and Y. Zhang, "Dependable and Secure Data Storage Architecture," Rio de Janeiro, Brazil, Apr 2009, pp 688-692.

[21] G. Ateniese, R. D. Pietro, L. V. Mancini, and G. Tsudik,"Scalable and Efficient Provable Data Possession," Istanbul, Turkey Proc. of SecureComm '08, pp. 1–10, 2008.

[22] K. D. Bowers, A. Juels, and A.Oprea, "HAIL: A High-Availability and Integrity Layer for Cloud Storage," Cryptology ePrint Archive, USA, 2008, pp 102-113.

[23] R. Curtmola, O. Khan, R. Burns, and G. Ateniese, "MRPDP: Multiple- Replica Provable Data Possession", USA , pp. 411–420, 2008.