IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Provable Data Possession by Identifying Duplicate Outsourced Data Transfer in Cloud Computing

PENMESTA SATYA BHAVANI [#1], L. SOWJANYA [#2]

[#1] MSC Student, Master of Computer Science,

D.N.R. College, P.G.Courses & Research Center, Bhimavaram, AP, India.

[#2] Assistant Professor, Master of Computer Applications,

D.N.R. College, P.G.Courses & Research Center, Bhimavaram, AP, India.

Abstract

Cloud Computing is one of the practice of using a network of remote servers hosted on internet to store, access, retrieve data from remote machines not from local machines. As the data will be stored on remote server, the user will retrieve the data from that server at the time of need. So for that data storage and access the user need to pay the amount on rental basis, that is the main reason why the cloud is also known as PAUZ(I.e Pay As you Uze).As data is stored in remote system we need to have a facility to avoid the duplicate data not to be reside on that server systems, if this is not avoided the user need to pay excess amount for that duplicated storage. So in order to avoid this data duplication we need to apply a new principle called as Data Deduplication.This is one of the best data compression technique that was used for eliminating the duplicate copies of repeated data and this was widely used in recent cloud storage.

1. INTRODUCTION

Cloud computing is the use of computing resources (hardware and software) that are delivered as a service over a network (typically the Internet). The name comes from the common use of a cloud-shaped symbol as an abstraction for the complex infrastructure it contains in system diagrams. Cloud computing entrusts remote services with a user's data, software and computation. Cloud computing consists of hardware and software resources made available on the Internet as managed third-party services. These services typically provide access to advanced software applications and high-end networks of server computers.

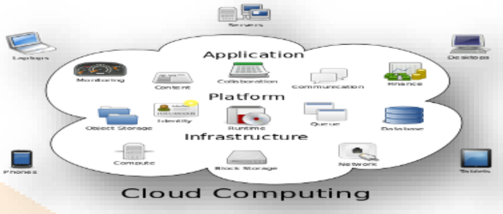


Fig: 1.1 Structure of cloud computing

PROBLEM STATEMENT

Here in this current application we try to address these limitations of current cloud server and we discuss about the major limitation in current cloud server is data is not encrypted by any means of encryption technique. All the data which is stored in either public cloud or private cloud is stored in a plain manner. By using the current cloud storage ,de-duplication of data can’t be minimized. The data owners only outsource their data storage by utilizing public cloud while the data operation is managed in private cloud.

In this proposed work, we enhance our system in security. Here we try to design the application with two main principles : One is encryption of data so that only valid users can able to access the data from the cloud server and other one is Access policies in which the cloud server can give access for the users like : Read, Write and Update and the same user who got access permission can able to control the data and remaining un-authorized users cant able to access the files. Also the cloud server can control the duplication of data not to be allowed into the system. It is restricted by disallowing same filenames not be used while uploading the data into the cloud server.The user is only allowed to perform the duplicate check for files marked with the corresponding privileges. In this proposed system as an extension we have implemented proposed storage on real cloud service like : DriveHq Cloud Service provider.For the first time we have implemented encryption of data that is stored in cloud without having any data duplication.

2. LITERATURE SURVEY

2.1 INRODUCTION

Literature survey is the most important step in software development process. Before developing the tool, it is necessary to determine the time factor, economy and company strength. Once these things are satisfied, ten next steps are to determine which operating system and language used for developing the tool. Once the programmers start building the tool, the programmers need lot of external support. This support obtained from senior programmers, from book or from websites. Before building the system the above consideration r taken into for developing the proposed system.

2.2 RELATED WORK

1) Fast and secure laptop backups with encrypted de-duplication

AUTHORS: P. Anderson and L. Zhang

Many people now store large quantities of personal and corporate data on laptops or home computers. These often have poor or intermittent connectivity, and are vulnerable to theft or hardware failure. Conventional backup solutions are not well suited to this environment, and backup regimes are frequently inadequate. This paper describes an algorithm which takes advantage of the data which is common between users to increase the speed of backups, and reduce the storage requirements. This algorithm supports client-end per-user encryption which is necessary for confidential personal data.

2) Message-locked encryption and secure deduplication.

AUTHORS: M. Bellare, S. Keelveedhi, and T. Ristenpart

We formalize a new cryptographic primitive, Message-Locked Encryption (MLE), where the key under which encryption and decryption are performed is itself derived from the message. MLE provides a way to achieve secure deduplication (space-efficient secure outsourced storage), a goal currently targeted by numerous cloud-storage providers. We provide definitions both for privacy and for a form of integrity that we call tag consistency. Based on this foundation, we make both practical and theoretical contributions. On the practical side, we provide ROM security analyses of a natural family of MLE schemes that includes deployed schemes. On the theoretical side the challenge is standard model solutions, and we make connections with deterministic encryption, hash functions secure on correlated inputs and the sample-then-extract paradigm to deliver schemes under different assumptions and for different classes of message sources. Our work shows that MLE is a primitive of both practical

3. Security proofs for identity-based identification and signature schemes.

AUTHORS: M. Bellare, C. Namprempre, and G. Neven

This paper provides either security proofs or attacks for a large number of identity-based identification and signature schemes defined either explicitly or implicitly in existing literature. Underlying these is a framework that on the one hand helps explain how these schemes are derived and on the other hand enables modular security analyses, thereby helping to understand, simplify, and unify previous work. We also analyze a generic folklore construction that in particular yields identity-based identification and signature schemes without random oracles.

4. A reverse deduplication storage system optimized for reads to latest backups

AUTHORS: C. Ng and P. Lee. Revdedup

Deduplication is known to effectively eliminate duplicates, yet it introduces fragmentation that degrades read performance. We propose RevDedup, a deduplication system that optimizes reads to the latest backups of virtual machine (VM) images using reverse deduplication. In contrast with conventional deduplication that removes duplicates from new data, RevDedup removes duplicates from old data, thereby shifting fragmentation to old data while keeping the layout of new data as sequential as possible. We evaluate our RevDedup prototype using a 12-week span of real-world VM image snapshots of 160 users. We show that RevDedup achieves high deduplication efficiency, high backup throughput, and high read throughput.

5. **Secure deduplication with efficient and reliable convergent key management**

AUTHORS: P. Lee, and W. Lou

Data deduplication is a technique for eliminating duplicate copies of data, and has been widely used in cloud storage to reduce storage space and upload bandwidth. Promising as it is, an arising challenge is to perform secure deduplication in cloud storage. Although convergent encryption has been extensively adopted for secure deduplication, a critical issue of making convergent encryption practical is to efficiently and reliably manage a huge number of convergent keys. This paper makes the first attempt to formally address the problem of achieving efficient and reliable key management in secure deduplication.

3. EXISTING SYSTEM

In the existing system all the cloud servers don’t have a facility like encrypting the data before it is stored into the cloud server. In the existing cloud either public or private the following are the limitations that take place in storing and retrieving the data.

LIMITATION OF EXISTING SYSTEM

The following are the main limitations of the existing system. They are as follows:

1. The major limitation in current cloud server is data is not encrypted by any means of encryption technique.
2. All the data which is stored in either public cloud or private cloud is stored in a plain manner.
3. By using the current cloud storage ,de-duplication of data cant be minimized.
4. The data owners only outsource their data storage by utilizing public cloud while the data operation is managed in private cloud

4. PROPOSED SYSTEM

In this proposed system, we enhance our system in security. Here we try to design the application with two main principles : One is encryption of data so that only valid users can able to access the data from the cloud server and other one is Access policies in which the cloud server can give access for the users like : Read, Write and Update and the same user who got access permission can able to control the data and remaining un-authorized users cant able to access the files. Also the cloud server can control the duplication of data not to be allowed into the system. It is restricted by disallowing same filenames not be used while uploading the data into the cloud server.

ADVANTAGES OF THE PROPOSED SYSTEM

The following are the advantages of the proposed system, they are as follows:

1. The user is only allowed to perform the duplicate check for files marked with the corresponding privileges.

2. In this proposed system as an extension we have implemented proposed storage on real cloud service like : DriveHq Cloud Service provider.

For the first time we have implemented encryption of data that is stored in cloud without having any data duplication

5. SOFTWARE PROJECT MODULES

Implementation is the stage where the theoretical design is converted into programmatically manner. In this stage we will divide the application into a number of modules and then coded for deployment. The front end of the application takes JSP,HTML and Java Beans and as a Back-End Data base we took My SQL data base. The application is divided mainly into following 4 modules. They are as follows:

1) Cloud Service Provider Module

2) Data User Module
3) Private Cloud Module
4) Secure Data De-duplication Module

Now let us discuss about each and every module in detail

5.1 CLOUD SERVICE PROVIDER

- In this module, we develop Cloud Service Provider module. This is an entity that provides a data storage service in public cloud.
- The S-CSP provides the data outsourcing service and stores data on behalf of the users.
- To reduce the storage cost, the S-CSP eliminates the storage of redundant data via deduplication and keeps only unique data.
- In this project, we assume that S-CSP is always online and has abundant storage capacity and computation power.

5.2 DATA USERS MODULE

- A user is an entity that wants to outsource data storage to the S-CSP and access the data later.
- In a storage system supporting deduplication, the user only uploads unique data but does not upload any duplicate data to save the upload bandwidth, which may be owned by the same user or different users.
- In the authorized deduplication system, each user is issued a set of privileges in the setup of the system. Each file is protected with the convergent encryption key and privilege keys to realize the authorized deduplication with differential privileges.

5.3 PRIVATE CLOUD MODULE

- Compared with the traditional deduplication architecture in cloud computing, this is a new entity introduced for facilitating user's secure usage of cloud service.
- Specifically, since the computing resources at data user/owner side are restricted and the public cloud is not fully trusted in practice, private cloud is able to provide data user/owner with an execution environment and infrastructure working as an interface between user and the public cloud.
- The private keys for the privileges are managed by the private cloud, who answers the file token requests from the users. The interface offered by the private cloud allows user to submit files and queries to be securely stored and computed respectively.

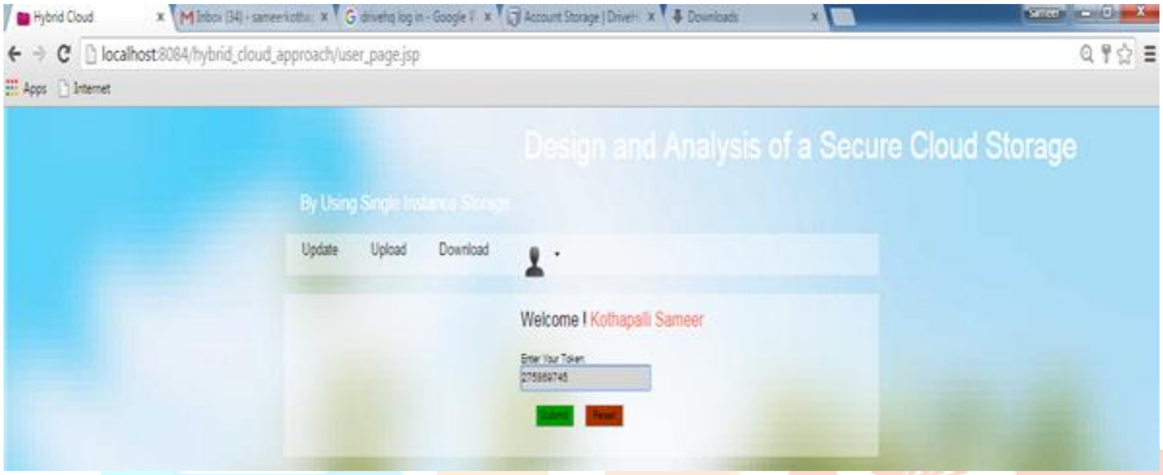
5.4 SECURE DE-DUPLICATION SYSTEM

- We consider several types of privacy we need protect, that is, unforgeability of duplicate-check token: There are two types of adversaries, that is, external adversary and internal adversary.

- As shown below, the external adversary can be viewed as an internal adversary without any privilege

6. OUTPUT RESULTS

1) USER ENTERING HIS TOKEN:



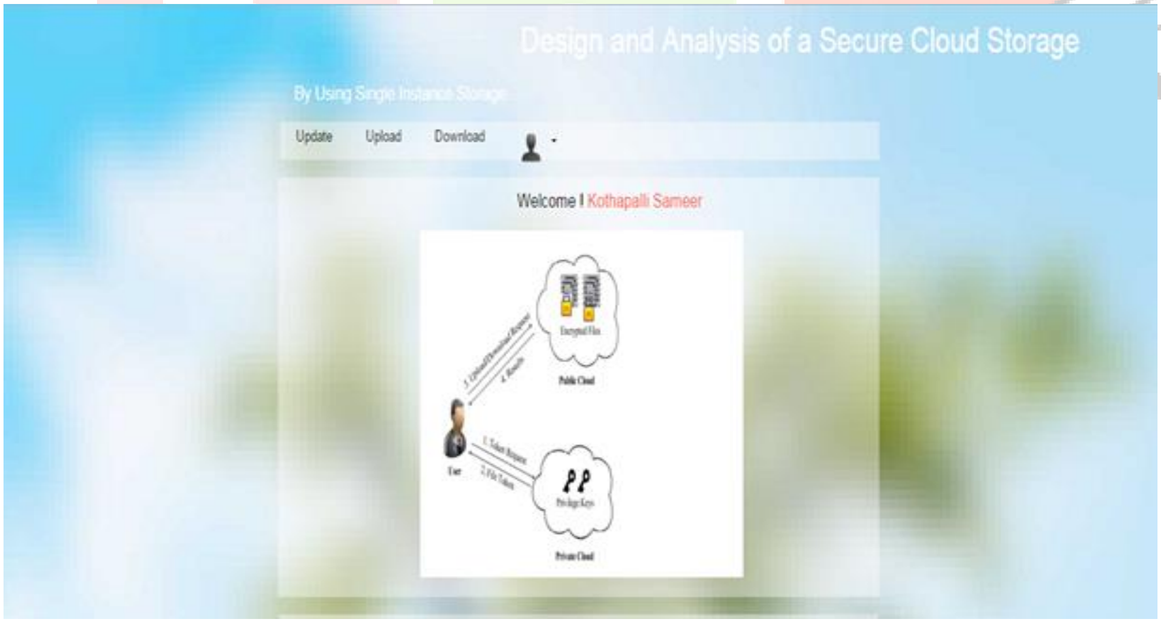
Design and Analysis of a Secure Cloud Storage

By Using Single Instance Storage

Welcome ! Kothapalli Sameer

Enter Your Token:
275869745

2) USER SUCCESSFULLY LOGIN TO THE CLOUD:

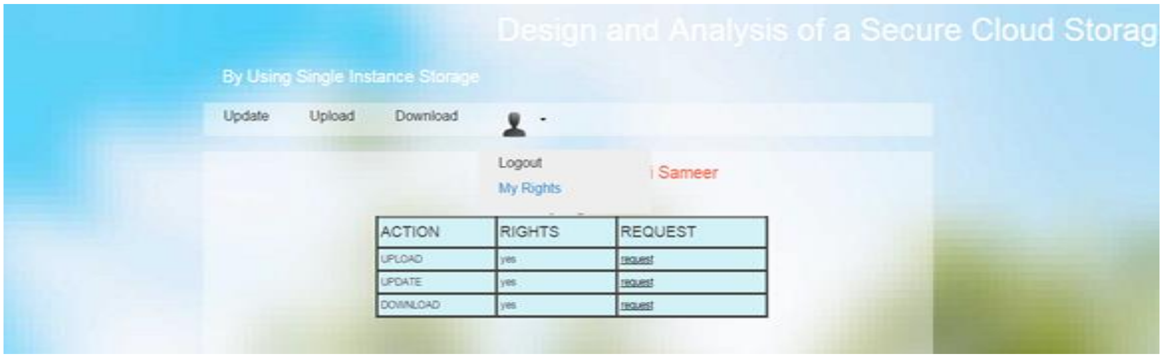


Design and Analysis of a Secure Cloud Storage

By Using Single Instance Storage

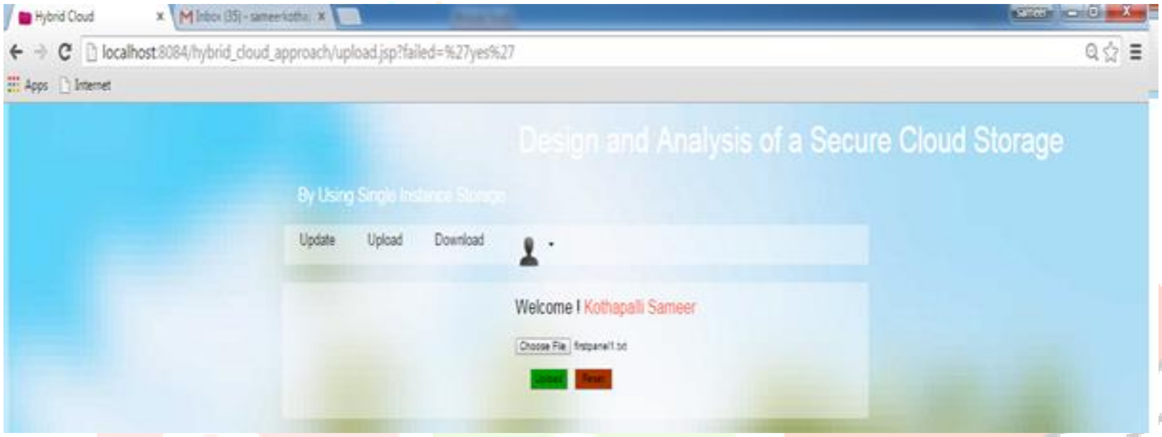
Welcome ! Kothapalli Sameer

3) USER CHECKING HIS RIGHTS ON CLOUD:


Design and Analysis of a Secure Cloud Storag
By Using Single Instance Storage
Update
Upload
Download
Logout
My Rights
Sameer

ACTION	RIGHTS	REQUEST
UPLOAD	yes	request
UPDATE	yes	request
DOWNLOAD	yes	request

4) USER UPLODING A FILE:

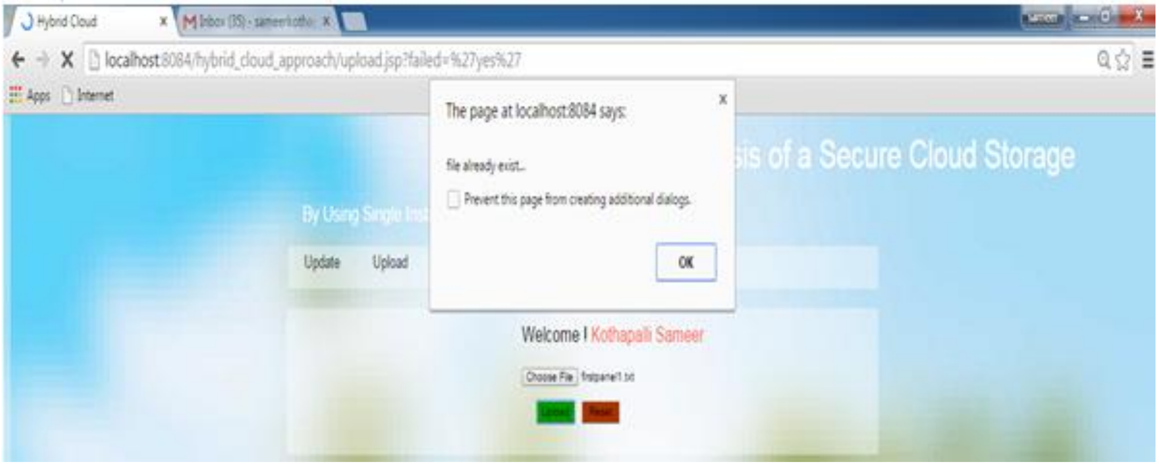


localhost:8084/hybrid_cloud_approach/upload.jsp?failed=%27yes%27

Welcome ! Kothapalli Sameer

firstpanel1.txt

5) WARNING MESSAGE FOR FILE ALREADY EXISTS:



localhost:8084/hybrid_cloud_approach/upload.jsp?failed=%27yes%27

The page at localhost:8084 says:

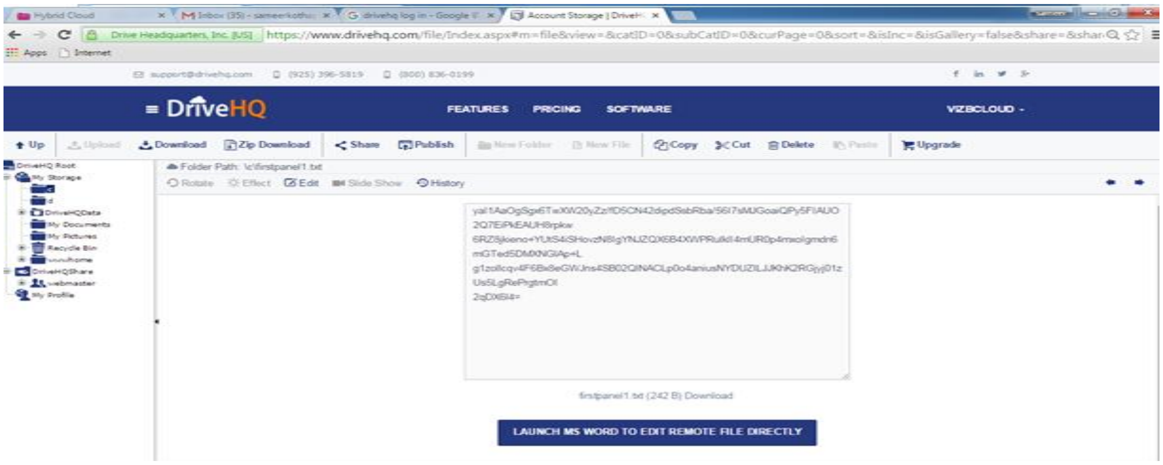
file already exist..

□ Prevent this page from creating additional dialogs.

Welcome ! Kothapalli Sameer

firstpanel1.txt

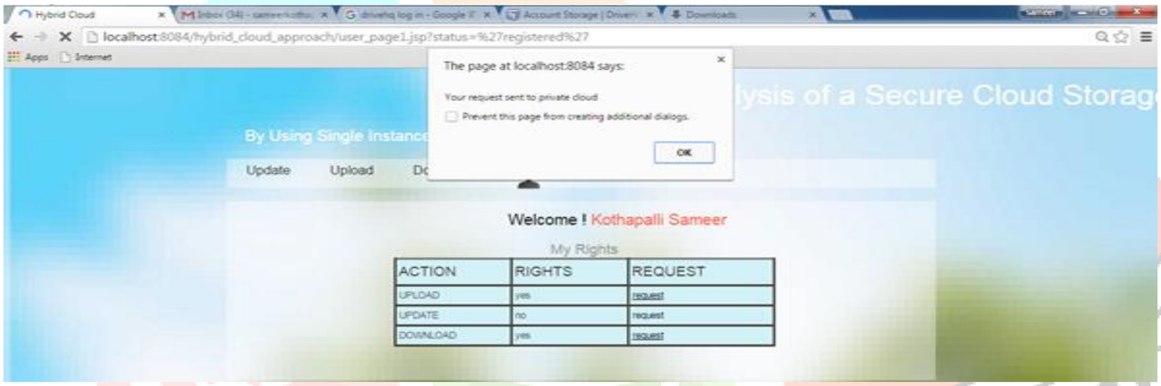
6) FILE IS STORED IN FORM OF ENCRYTION MANNEER IN CLOUD:



```
yal1AaOgSgx6TeXW20yZz!fD5CN42dipdSsbRba/56I7sMJGoaiQPy5FIAUO
2Q7EiPkEAUH8rpkw
6RZ8jkeno+YUtS4iSHovzN8IgYNJZQX6B4XWPRu8kl4mUR0p4mxolgmdn6
mGTed5DMXNGIAp+L
g1zoIlcqv4F6Bx8eGWJns4SB02QINACLp0o4aniusNYDUZILJJKhK2RGjyj01z
Us5LgRePrgtmOI
2qDX6I4=
```

firstpanel1.txt (242 B) Download

7) USER IS REQUESTING FOR UDATE ACTION



localhost:8084/hybrid_cloud_approach/user_page1.jsp?status=%27registered%27

The page at localhost:8084 says:

Your request sent to private cloud

□ Prevent this page from creating additional dialogs.

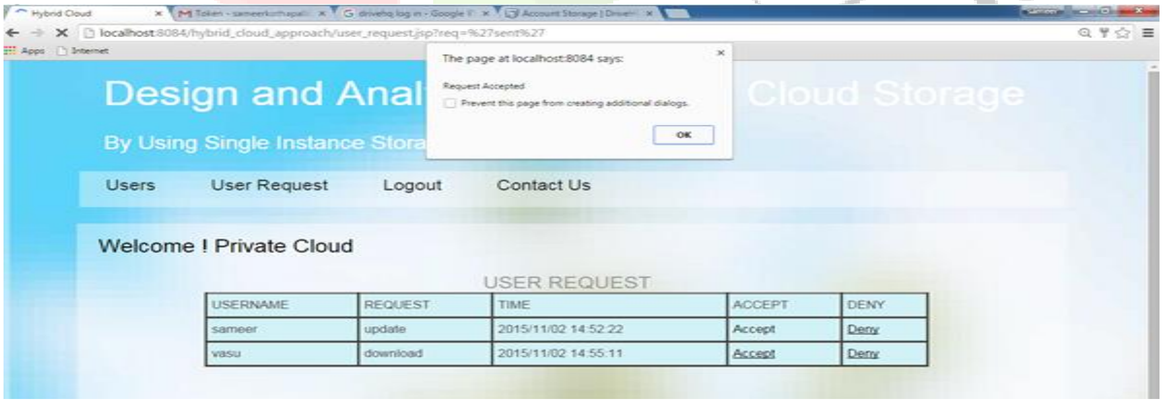
OK

Welcome ! Kothapalli Sameer

My Rights

ACTION	RIGHTS	REQUEST
UPLOAD	yes	request
UPDATE	no	request
DOWNLOAD	yes	request

8) PRIVATE CLOUD ACTIVATING THE USER TO DO UDATE:



localhost:8084/hybrid_cloud_approach/user_request.jsp?req=%27sent%27

The page at localhost:8084 says:

Request Accepted

□ Prevent this page from creating additional dialogs.

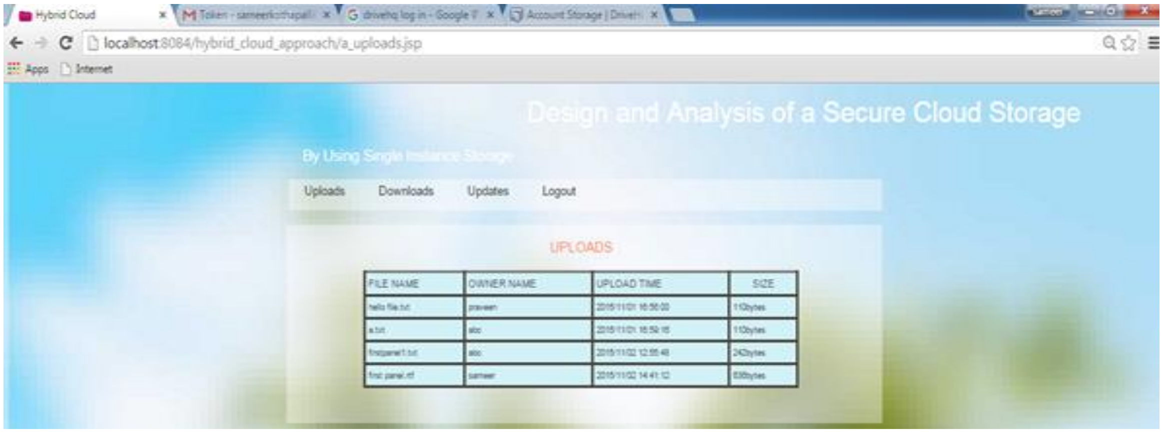
OK

Welcome ! Private Cloud

USER REQUEST

USERNAME	REQUEST	TIME	ACCEPT	DENY
sameer	update	2015/11/02 14:52:22	Accept	Deny
vasu	download	2015/11/02 14:55:11	Accept	Deny

9) ADMIN CAN SEE NUMBER OF UPLOADS FROM USERS:


Hybrid Cloud
Token - sameerkothapalli
drivehq log in - Google
Account Storage | DriveHQ
localhost:8084/hybrid_cloud_approach/a_uploads.jsp
Apps
Internet
Design and Analysis of a Secure Cloud Storage
By Using Single Instance Storage
Uploads
Downloads
Updates
Logout

UPLOADS

FILE NAME	OWNER NAME	UPLOAD TIME	SIZE
hello file.txt	praveen	2015/11/01 16:56:00	11Qbytes
a.txt	abc	2015/11/01 16:59:16	11Qbytes
firstpanel1.txt	abc	2015/11/02 12:55:48	242bytes
first panel.rtf	sameer	2015/11/02 14:41:12	536bytes

7. CONCLUSION

In this project, we have analyzed the design, implementation, and evaluation of a HYBRID CLOUD with data de-duplication at the end. And also data should be stored in the form of encrypted manner. Our proposed system uses this new de-duplication technique in order to main concept for resource allocations. As a proof of concept, we implemented a prototype of our proposed authorized duplicate check scheme and conduct test bed experiments on our prototype. We showed that our authorized duplicate check scheme incurs minimal overhead compared to convergent encryption and network transfer.

8. REFERENCES

[1] OpenSSL Project. http://www.openssl.org/.

[2] P. Anderson and L. Zhang. Fast and secure laptop backups with encrypted de-duplication. In *Proc. of USENIX LISA*, 2010.

[3] M. Bellare, S. Keelveedhi, and T. Ristenpart. Dupless: Serveraided encryption for deduplicated storage. In *USENIX Security Symposium*, 2013.

[4] M. Bellare, S. Keelveedhi, and T. Ristenpart. Message-locked encryption and secure deduplication. In *EUROCRYPT*, pages 296– 312, 2013.

[5] M. Bellare, C. Namprempre, and G. Neven. Security proofs for identity-based identification and signature schemes. *J. Cryptology*, 22(1):1–61, 2009.

[6] M. Bellare and A. Palacio. Gq and schnorr identification schemes: Proofs of security against impersonation under active and concurrent attacks. In *CRYPTO*, pages 162–177, 2002.

[7] S. Bugiel, S. Nurnberger, A. Sadeghi, and T. Schneider. Twin clouds: An architecture for secure cloud computing. In *Workshop on Cryptography and Security in Clouds (WCSC 2011)*, 2011.

[8] J. R. Douceur, A. Adya, W. J. Bolosky, D. Simon, and M. Theimer. Reclaiming space from duplicate files in a serverless distributed file system. In *ICDCS*, pages 617–624, 2002.

[9] D. Ferraiolo and R. Kuhn. Role-based access controls. In *15th NIST-NCSC National Computer Security Conf.*, 1992.

[10] GNU Libmicrohttpd. http://www.gnu.org/software/libmicrohttpd/.

[11] S. Halevi, D. Harnik, B. Pinkas, and A. Shulman-Peleg. Proofs of ownership in remote storage systems. In Y. Chen, G. Danezis, and V. Shmatikov, editors, *ACM Conference on Computer and Communications Security*, pages 491–500. ACM, 2011.

[12] J. Li, X. Chen, M. Li, J. Li, P. Lee, andW. Lou. Secure deduplication with efficient and reliable convergent key management. In *IEEE Transactions on Parallel and Distributed Systems*, 2013.

[13] libcurl. http://curl.haxx.se/libcurl/.

[14] C. Ng and P. Lee. Revdedup: A reverse deduplication storage system optimized for reads to latest backups. In *Proc. of APSYS*, Apr 2013.

[15] W. K. Ng, Y. Wen, and H. Zhu. Private data deduplication protocols in cloud storage. In S. Ossowski and P. Lecca, editors, *Proceedings of the 27th Annual ACM Symposium on Applied Computing*, pages 441–446. ACM, 2012.

[16] R. D. Pietro and A. Sorniotti. Boosting efficiency and security in proof of ownership for deduplication. In H. Y. Youm and Y. Won, editors, *ACM Symposium on Information, Computer and Communications Security*, pages 81–82. ACM, 2012.

[17] S. Quinlan and S. Dorward. Venti: a new approach to archival storage. In *Proc. USENIX FAST*, Jan 2002.

[18] A. Rahumed, H. C. H. Chen, Y. Tang, P. P. C. Lee, and J. C. S. Lui. A secure cloud backup system with assured deletion and version control. In *3rd International Workshop on Security in Cloud Computing*, 2011.

[19] R. S. Sandhu, E. J. Coyne, H. L. Feinstein, and C. E. Youman. Role-based access control models. *IEEE Computer*, 29:38–47, Feb 1996.

[20] J. Stanek, A. Sorniotti, E. Androulaki, and L. Kencl. A secure data deduplication scheme for cloud storage. In *Technical Report*, 2013.

[21] M. W. Storer, K. Greenan, D. D. E. Long, and E. L. Miller. Secure data deduplication. In *Proc. of StorageSS*, 2008.

[22] Z. Wilcox-O’Hearn and B. Warner. Tahoe: the least-authority filesystem. In *Proc. of ACM StorageSS*, 2008.