isrjournals

International Journal of Advanced Research in Computer Science Engineering and Information Technology

Volume: 6 Issue: 3 Mar,2017,ISSN_NO: 2321-3337

TO OVERCOME THE REDUNANCY IN PRIVATE CLOUD USING DEDUPLICATION ALGORITHM

M.Ruknudeen[1] ,S. Sharath[2] , M.Nagoor Imran[3] , V.Ramachandran[4]

Student, Dept. of Information Technology, Mohamed Sathak AJ College of Engineering,India.[1,2,3]

Asst.professor, Dept. of Information Technology, Mohamed Sathak AJ College of Engineering, India.[4]

ruknudenbasha@gmail.com[1],sharathcrazy754@gmail.com[2],nagoorimran54@gmail.com[3], it.ramachandran@msajce-edu.in[4]

Abstract—Dynamic Proof of Storage (PoS) is a useful cryptographic primitive that enables a user to check the integrity of outsourced files and to efficiently update the files in a cloud server. Although researchers have proposed many dynamic PoS schemes in single user environments, the problem in multi-user environments has not been investigated sufficiently. A practical multi-user cloud storage system needs the secure client-side cross-user deduplication technique, which allows a user to skip the uploading process and obtain the ownership of the files immediately, when other owners of the same files have uploaded them to the cloud server. To the best of our knowledge, none of the existing dynamic PoSs can support this technique. In this paper, we introduce the concept of deduplicatable dynamic proof of storage and propose an efficient construction called DeyPoS, to achieve dynamic PoS and secure cross-user deduplication, simultaneously. Considering the challenges of structure diversity and private tag generation, we exploit a novel tool called Homomorphic Authenticated Tree (HAT). We prove the security of our construction, and the theoretical analysis and experimental results show that our construction is efficient in practice.

Index Terms—Cloud storage, dynamic proof of storage, deduplication.

I.INTRODUCTION

Storage outsourcing [2] is becoming more and more attractive to both industry and academia due to the advantages of low cost, high accessibility, and easy sharing. As one of the storage outsourcing forms, cloud storage gains wide attention in recent years. Many companies, such as Amazon, Google, and Microsoft, provide their own cloud storage services, where users can upload their files to the servers, access them from various devices, and share them with the others. Although cloud storage services are widely adopted in current days, there still remain many security issues and potential threats. Data integrity is one of the most vital properties when a user outsources its files to cloud storage. Users should be convinced that the files stored in the server are not tampered. Traditional techniques [4] for protecting data integrity, such as message authentication codes (MACs) and digital signatures, require users to download all of the files from the cloud server for verification, which incurs a heavy communication cost. These techniques are not suitable for cloud storage services where users may check the integrity frequently, such as every hour. Thus, researchers introduced *Proof of Storage* (PoS) for checking the integrity without downloading files from the cloud server. Considering the challenges of structure diversity [5] and private tag generation, we exploit a novel tool called Homomorphic Authenticated Tree (HAT).Traditional techniques for protecting data integrity, such as message authentication codes (MACs) and digital signatures, require users to download all of the files from the cloud server for verification, which incurs a heavy communication cost.

Kun He,Jing Chen,Ruiying Du,Qianhong Wu,Xiang Zhang [1] Suggested that Proofs of Retrievability (POR) is a cryptographic method for remotely auditing the integrity of les stored in the cloud, without keeping a copy of the original les in local storage.We can periodically and remotely verify the integrity of data stored with using the authentication data, without retrieving back the data during a verifcation. Besides security, performances in communication, storage overhead and computation are major considerations.

Z. Xia, X. Wang, X. Sun, and Q. Wang [3] Focused on Cloud Computing, the long-held dream of computing as a utility ,has the potential to transform a large part of the IT industry ,making software even more attractive as a service and shaping the way IT hardware is designed and purchased. Developers with innovative ideas for new Internet services no longer require the large capital outlays in hardware to deploy their service or the human expense to operate it. They need not be concerned about over-provisioning [6] for a service whose popularity does not meet their predictions, thus wasting costly resources ,or under-provisioning for one that becomes wildly popular, thus missing potential customers and revenue.The Problem Statement is described in Section II.

II. PROBLEM STATEMENT

A sensible multi-user cloud storage system desires the secure client-side cross-user deduplication technique, that permits a user to skip the uploading method and obtain the possession of the files now, once different house owners of identical files have uploaded them to the cloud server. To the best of our data, none of the prevailing dynamic PoSs will support this system. during this paper, we tend to introduce the construct of deduplicatable dynamic proof of storage Associate in Nursing the propose an economical construction referred to as DeyPoS, to realize dynamic PoS and secure cross-user deduplication, at the same time. Considering the challenges of structure diversity and personal tag generation, we tend to exploit a novel tool referred to as Homo morphic documented Tree (HAT). we tend to prove the protection of our construction, and therefore the theoretical analysis and experimental results show that our construction is economical in observe. The Exising System is described in Section III.

III. EXISTING SYSTEM

A practical multi-user cloud storage system needs the secure client-side cross-user de duplication technique, which allows a user to skip the uploading process and obtain the ownership of the files immediately, when other owners of the same files have uploaded them to the cloud server. To the best of our knowledge, none of the existing dynamic PoSs can support this technique .Subsequent users need to convince the cloud server that they own the files without uploading them to the cloud server. Note that, these three phases (pre-process, upload, and deduplication) are executed only once in the life cycle of a file from the perspective of users. That is, these three phases appear only when users intend to upload files. If these phases terminate normally, i.e., users finish uploading in the upload phase, or they pass the verification in the deduplication phase, we say that the users have the ownerships of the files. In the *update* phase, users may modify, insert, or delete some blocks of the files. Then, they update the corresponding parts of the encoded files and the authenticated structures in the cloud server, even the original files were not uploaded by themselves. Note that, users can update the files only if they have the ownerships of the files, which means that the users should upload the files in the upload phase or pass the verification in the deduplication 4 phase. For each update, the cloud server has to reserve the original file and the authenticated structure if there exist other owners, and record the updated part of the file and the authenticated structure. The proposed system is described in Section IV.

IV.PROPOSED SYSTEM

Our system model considers two types of entities: the cloud server and users, as shown in Fig. 1. For each file, original user is the user who uploaded the file to the cloud server, while subsequent user is the user who proved the ownership of the file but did not actually upload the file to the cloud server. There are five phases in a deduplicatable dynamic PoS system: preprocess, upload, deduplication, update, and proof of storage. In the pre-process phase, users intend to upload their local files. The cloud server decides whether these files should be uploaded. If the upload process is granted, go into the upload phase; otherwise, go into the deduplication phase. In the upload phase, the files to be uploaded do not exist in the cloud server. The original users encodes the local files and upload them to the cloud server. In the deduplication phase, the files to be uploaded already exist in the cloud server. The subsequent users possess the files locally and the cloud server stores the authenticated structures of the files. Subsequent users need to convince the cloud server that they own the files without uploading them to the cloud server.The Modules are described below.

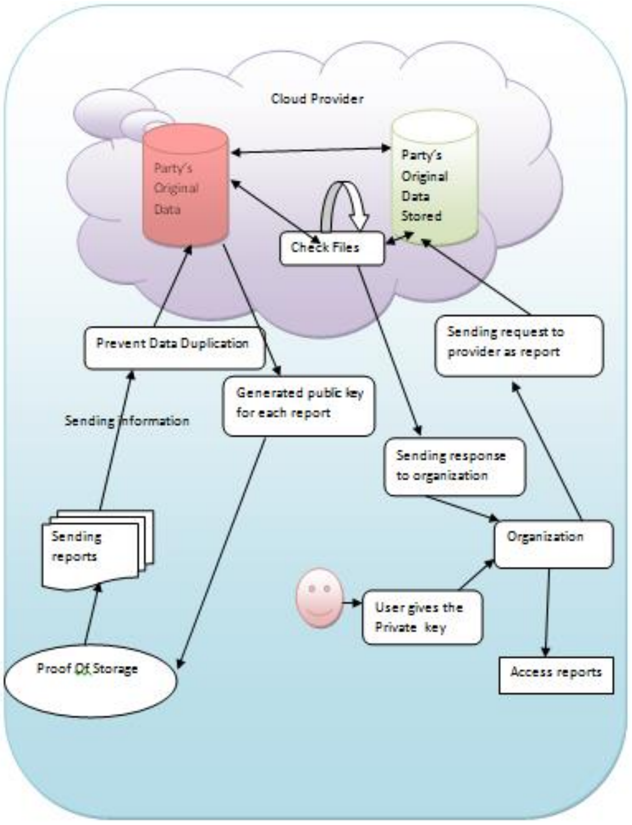

Cloud Provider
Party's Original Data
Party's Original Data Stored
Check Files
Prevent Data Duplication
Generated public key for each report
Sending request to provider as report
Sending information
Sending response to organization
Sending reports
Organization
User gives the Private key
Proof Of Storage
Access reports

Fig 1.System Architecture

1.User Interface Design

In this module we design the windows for the project. These windows are used to send a message from one peer to another. We use the Swing package available in Java to design the User Interface. Swing is a widget toolkit for Java. It is part of Sun Microsystems' Java Foundation Classes an API for providing a graphical user interface for Java programs. In this module mainly we are focusing the login design page with the Partial knowledge information. Application Users need to view the application they need to login through the User Interface GUI is the media to connect User and Media Database and login screen where user can input his/her user name, password and password will check in database, if that will be a valid username and password then he/she can access the database is shown in Fig 2.

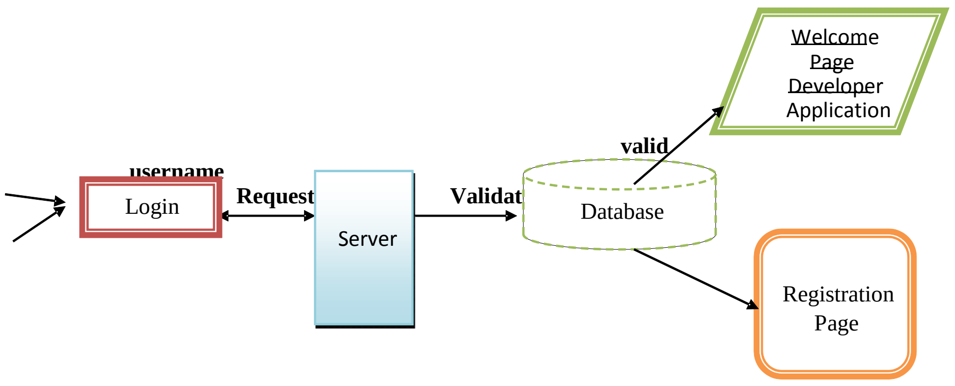


Fig 2.User Interface Design

2.Data Upload

This is the module for uploading owner’s files or documents into the virtual machines. These constraints serve a dual purpose as they can introduce high-level policies and assist in administration tasks. The user send the file to cloud send the Data so upload the file or Data. Given that we rely on network services for our most security-critical data. A source wants to securely send a message to a set of receivers over a cloud network with unit-capacity edges,in the presence of a cloud user is shown in Fig 3.

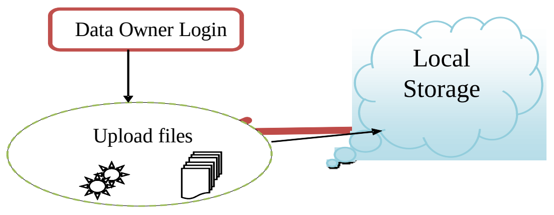


Fig 3.Data Upload

3. Unique key generation for each file upload

This is the module for creating into the Key Generate Data or documents into the virtual machines cloud. These constraints serve a dual purpose as they can introduce high-level policies and assist in administration tasks. Uploaded files retrieve purpose at the time only giving key otherwise not retrieves. So important to any file generate secret key to client. Generate all keys random wise at the time upload data is shown in Fig 4.

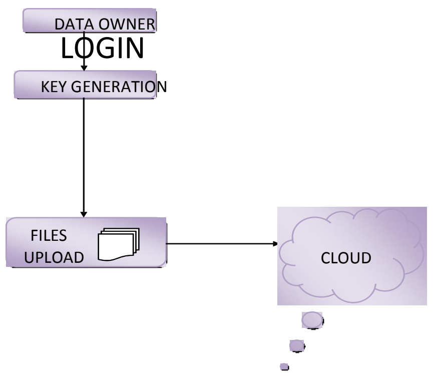


Fig 4.Unique key generation for each file upload

4. Avoiding Deduplicate file

In this module we give avoid the de duplicate data in the cloud. In the pre-process phase, users intend to upload their local files. The cloud server decides whether these files should be uploaded. If the upload process is granted, go into the upload phase; otherwise, go into the de duplication phase. In the upload phase, the files to be uploaded do not exist in the cloud server. The original users encodes the local files and upload them to the cloud server. In the de duplication phase, the files to be uploaded already exist in the cloud server. The subsequent users possess the files locally and the cloud server stores the authenticated structures of the files is shown in Fig 5.

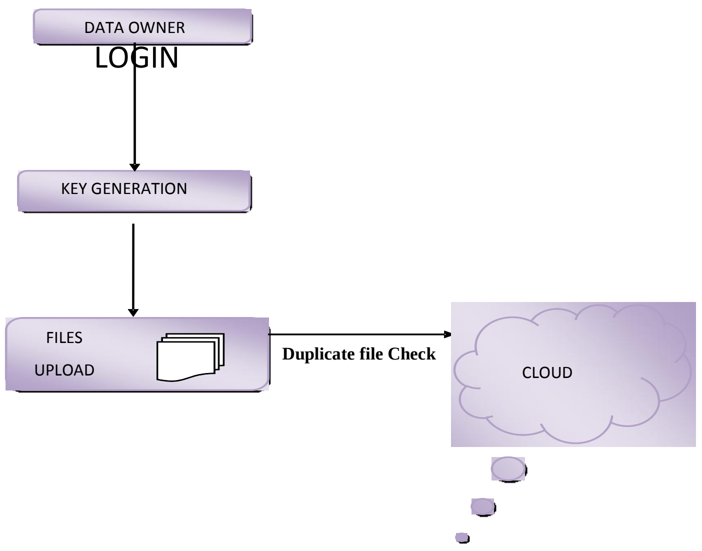


Fig 5.Avoiding Duplicate file

5. Proof of storage in Multi users

In this module summarization is important to proof of storage in multi user environments in the cloud. A practical multi-user cloud storage system needs the secure client-side cross-user de duplication technique, which allows a user to skip the uploading process and obtain the ownership of the files immediately, when other owners of the same files have uploaded them to the cloud server. To the best of our knowledge, none of the existing dynamic proof of storage can support this technique is shown in Fig 6. The Implementation is described in following section V.

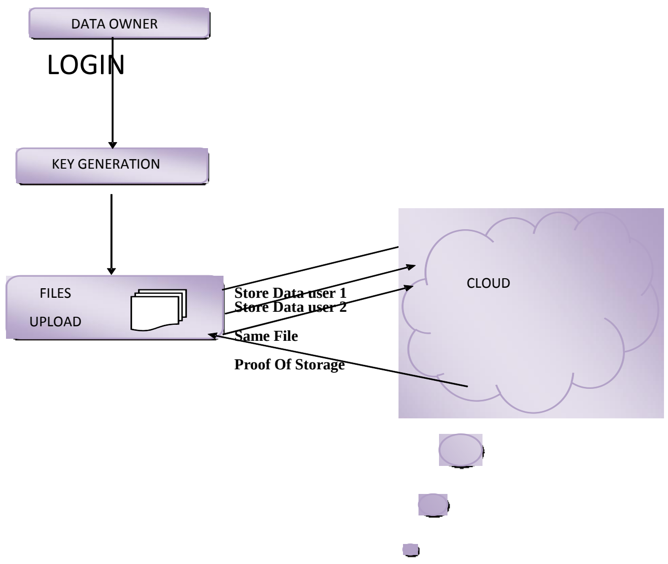


Fig 6. Proof of storage in multi users

V.IMPLEMENTATION

Original user is the user who uploaded the file to the cloud server, while subsequent user is the user who proved the ownership of the file but did not actually upload the file to the cloud server. The Algorithm for Deduplication and Module diagram is described below.

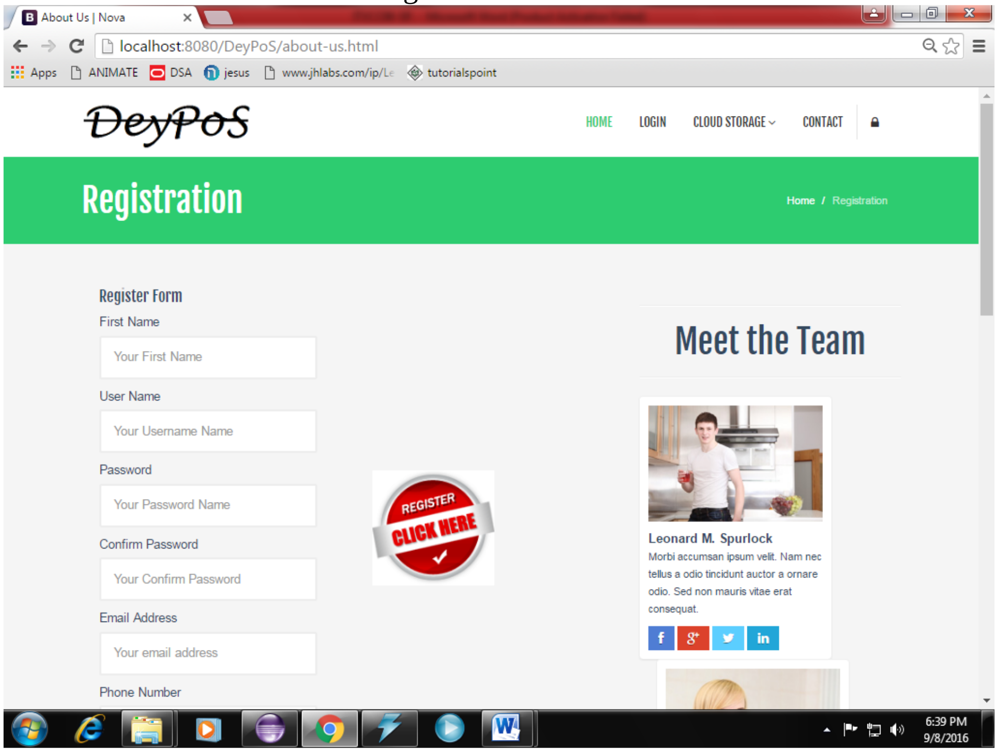
ALGORITHM:

The deduplication proving Algorithm

```
1: procedure DEDUPPROVE($\alpha s$, kc, $\alpha c$, $\{c1, . . . , cn\}$, I ,Q)
2: $c \leftarrow 0, t \leftarrow \emptyset, \zeta \leftarrow 1, l \leftarrow 1$
3: while $\zeta \leq n$ do
4: $\delta \leftarrow 0$
5: while $\zeta <$ ɩjl do
6: $\delta \leftarrow \delta + c\_ , \zeta \leftarrow \zeta + 1$
7: pop the first element in Q
8: $t \leftarrow t \cup \{fkc\ (iklikvi) + \alpha c \alpha s \delta\}$
9: $c \leftarrow c + c\_$
10: $l \leftarrow l + 1, \zeta \leftarrow \zeta + 1$
11: return c, t
```

Volume: 6 Issue: 3 Mar,2017,ISSN_NO: 2321-3337

The Module User interface design is shown below.


About Us | Nova
localhost:8080/DeyPoS/about-us.html
Apps ANIMATE DSA jesus www.jhlabs.com/ip/Le tutorialspoint
DeyPoS
HOME LOGIN CLOUD STORAGE CONTACT
Home / Registration
REGISTER CLICK HERE
6:39 PM 9/8/2016

Registration

Register Form

First Name: Your First Name

User Name: Your Username Name

Password: Your Password Name

Confirm Password: Your Confirm Password

Email Address: Your email address

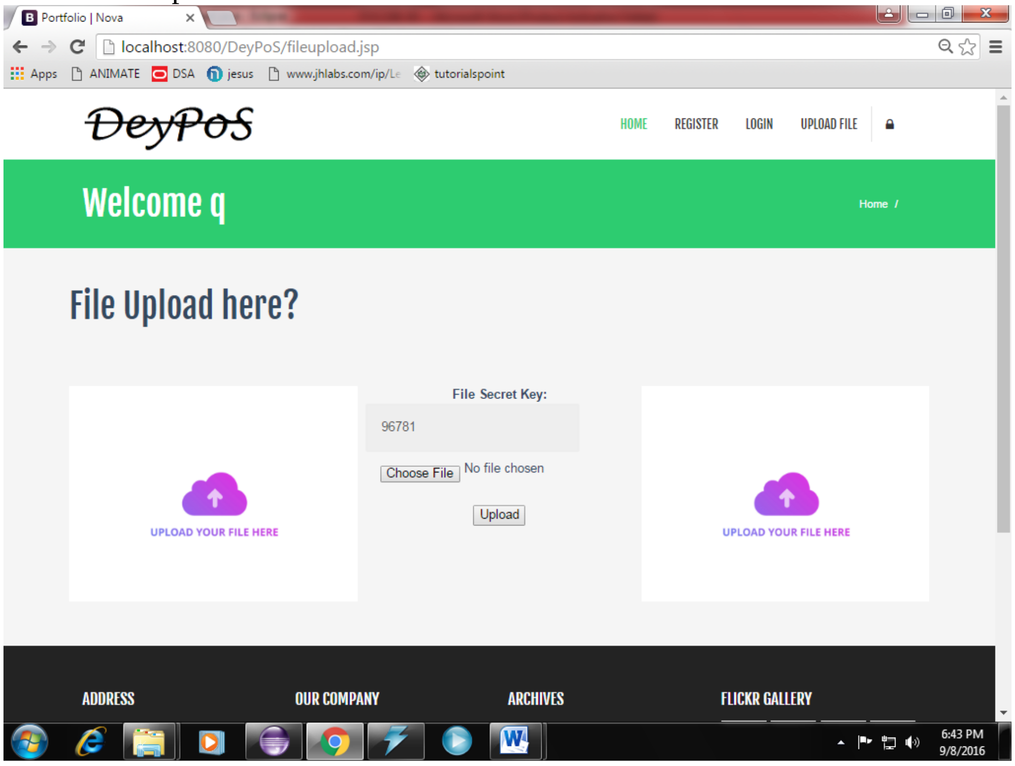
Phone Number

Meet the Team

Leonard M. Spurlock

Morbi accumsan ipsum velit. Nam nec tellus a odio tincidunt auctor a ornare odio. Sed non mauris vitae erat consequat.

The Module Data upload is shown below.



Welcome q

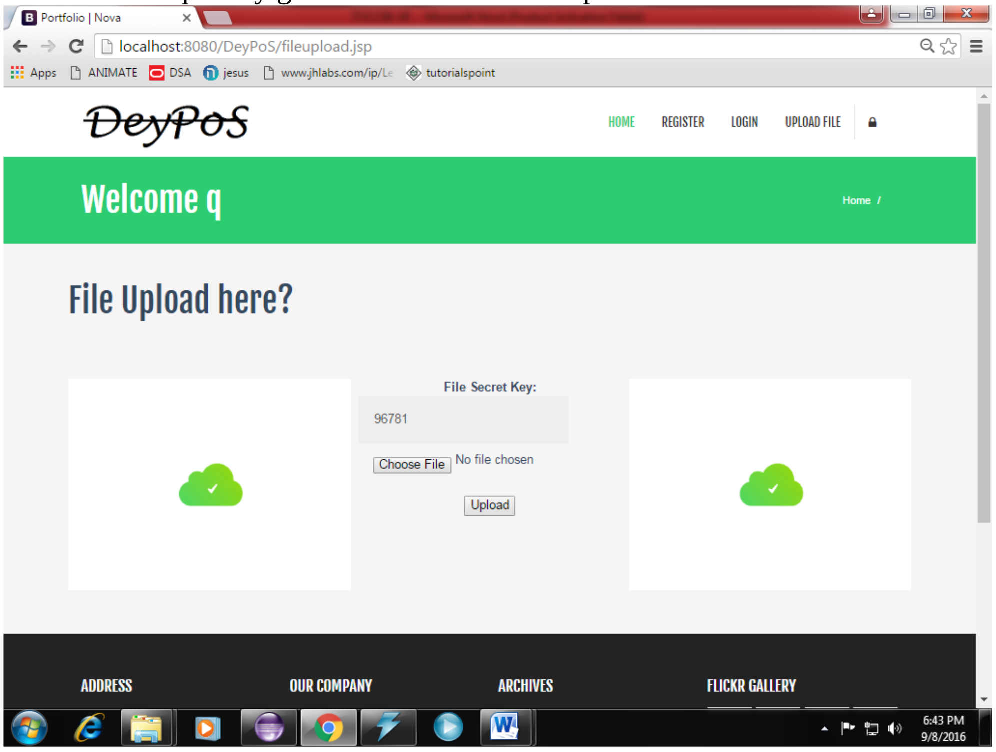
File Upload here?

File Secret Key: 96781

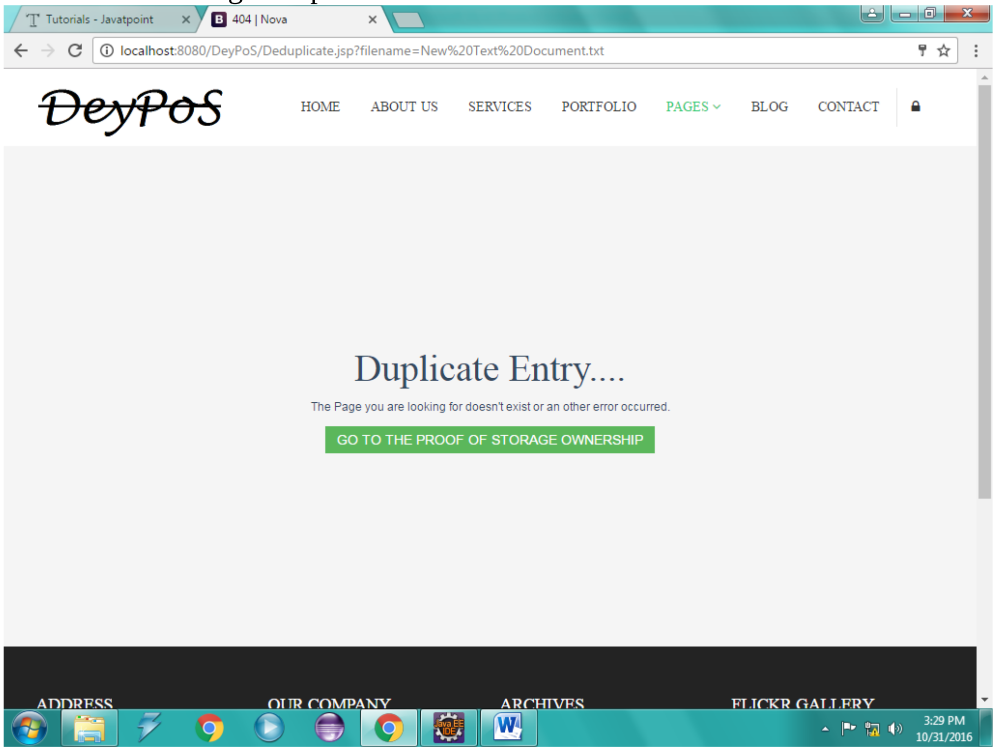
Choose File No file chosen

Upload

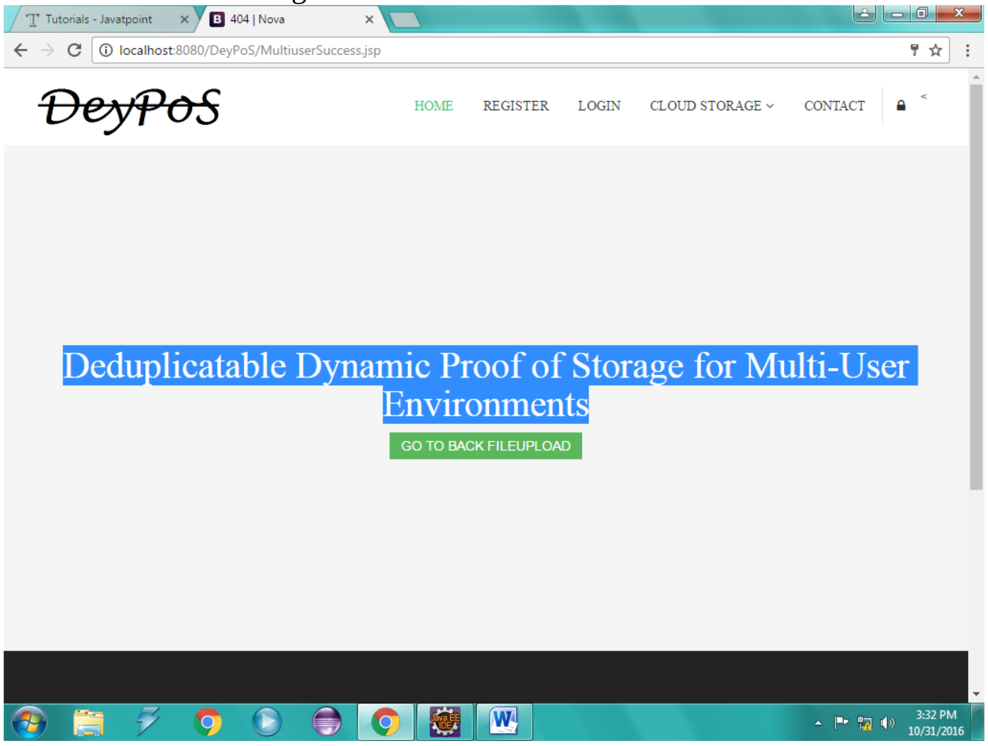
The Module Unique key generation for each file upload is shown below.



The Module Avoiding deduplicate file is shown below.


Tutorials - Javatpoint
404 | Nova
localhost:8080/DeyPoS/Deduplicate.jsp?filename=New%20Text%20Document.txt
DeyPoS
HOME
ABOUT US
SERVICES
PORTFOLIO
PAGES
BLOG
CONTACT
Duplicate Entry....
The Page you are looking for doesn't exist or an other error occurred.
GO TO THE PROOF OF STORAGE OWNERSHIP
ADDRESS
OUR COMPANY
ARCHIVES
FLICKR GALLERY
3:29 PM
10/31/2016

The Module Proof of storage in multiusers is shown below.



VI. CONCLUSION

We proposed the comprehensive requirements in multi-user cloud storage systems and introduced the model of de duplicable dynamic PoS. We designed a novel tool called HAT which is an efficient authenticated structure.Our Future work is based on HAT, we proposed the first practical deduplicatable dynamic PoS scheme called DeyPoS and proved its security in the random oracle model. The theoretical and experimental results show that our DeyPoS implementation is efficient, especially when the file size and the number of the challenged blocks are large.

REFERENCE

[1] Kun He,Jing Chen,Ruiying Du,Qianhong Wu,Xiang Zhang,"DeyPos:Deduplicatable Dynamic Proof of Storage for Multi-User Environments" in April 29,2016
[2] S. Kamara and K. Lauter, "Cryptographic cloud storage," in Proc. of FC, pp. 136–149, 2010.
[3] Z. Xia, X. Wang, X. Sun, and Q. Wang, "A Secure and Dynamic Multi-Keyword Ranked Search Scheme over Encrypted Cloud Data," IEEE Transactions on Parallel and Distributed Systems, vol. 27, no. 2, pp. 340–352, 2016.
[4] Z. Xiao and Y. Xiao, "Security and privacy in cloud computing," IEEE Communications Surveys Tutorials, vol. 15, no. 2, pp. 843–859, 2013.
[5] C. A. Ardagna, R. Asal, E. Damiani, and Q. H. Vu, "From Security to Assurance in the Cloud: A Survey," ACM Comput. Surv., vol. 48, no. 1, pp. 2:1–2:50, 2015.
[6] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, "Provable data possession at untrusted stores," in Proc. of CCS, pp. 598–609, 2007.

BIOGRAPHY

Ruknudeen.M pursuing his B.tech in IT department in Mohamed Sathak A.J. College of Engineering, Anna University in Chennai, India, in March 2017.

Sharath.S pursuing his B.tech in IT department in Mohamed Sathak A.J. College of Engineering, Anna University in Chennai, India, in March 2017.

Nagoor Imran.M pursuing his B.tech in IT department in Mohamed Sathak A.J. College of Engineering, Anna University in Chennai, India, in March 2017.

Ramachandran.V.He is currently working as Assistant Professor in Department of Information Technology in Mohamed Sathak A.J. College of Engineering,Chennai..