Dynamic and Public Auditing with Fair Arbitration for Cloud Data

R.Jothi[1], B.Abdul[2]

[1] *Student, Dept. of MCA, EAIMS*

[2] *Professor, Dept. of MCA, EAIMS, Tirupati, A.P.*

Abstract- **Cloud users no longer physically possess their data, so how to ensure the integrity of their outsourced data becomes a challenging task. Recently proposed schemes such as "provable data possession" and "proofs of retrievability" are designed to address this problem, but they are designed to audit static archive data and therefore lack of data dynamics support. Moreover, threat models in these schemes usually assume an honest data owner and focus on detecting a dishonest cloud service provider despite the fact that clients may also misbehave. This paper proposes a public auditing scheme with data dynamics support and fairness arbitration of potential disputes. In particular, we design an index switcher to eliminate the limitation of index usage in tag computation in current schemes and achieve efficient handling of data dynamics. To address the fairness problem so that no party can misbehave without being detected, we further extend existing threat models and adopt signature exchange idea to design fair arbitration protocols, so that any possible dispute can be fairly settled. The security analysis shows our scheme is provably secure, and the performance evaluation demonstrates the overhead of data dynamics and dispute arbitration are reasonable.**

Index Terms- **Integrity auditing, public verifiability, dynamic update, arbitration, fairness.**

I. INTRODUCTION

DATA oursourcing is a key application of cloud computing, which relieves cloud users of the heavy burden of data management and infrastructure maintenance, and provides fast data access independent of physical locations. However, outsourcing data to the cloud brings about many new security threats. Firstly, despite the powerful machines and strong security mechanisms provided by cloud service providers (CSP), remote data still face network attacks, hardware failures and administrative errors. Secondly, CSP may reclaim storage of rarely or never accessed data, or even hide data loss accidents for reputation reasons. As users no longer physically possess their data and consequently lose direct control over the data, direct employment of traditional cryptographic primitives like hash or encryption to ensure remote data's integrity may lead to many security loopholes.In particular, downloading all the data to check its integrity is not viable due to the expensive communication overhead, especially for large-size data files. In this sense, message authentication code (MAC) or signature based mechanisms, while widely used in secure storage systems, are not suitable for integrity check of outsourced data, because they can only verify the integrity of retrieved data and do not work for rarely accessed data (e.g., archive data). So how to ensure the correctness of outsourced data without possessing the original data becomes a challenging task in cloud computing, which, if not effectively handled, will impede the wide deployment of cloud services.

Data auditing schemes can enable cloud users to check the integrity of their remotely stored data without downloading them locally, which is termed as blockless verification. With auditing schemes, users can periodically interact with the CSP through auditing protocols to check the correctness of their outsourced data by verifying the integrity proof computed by the CSP, which offers stronger confidence in data security because user's own conclusion that data is intact is much more convincing than that from service providers. Generally speaking, there are several trends in the development of auditing schemes.

First of all, earlier auditing schemes usually require the CSP to generate a deterministic proof by accessing the whole data file to perform integrity check, e.g., schemes in [1], [2] use the entire file to perform modular exponentiations. Such plain solutions incur expensive computation overhead at the server side, hence they lack efficiency and practicality when dealing with large-size data. Represented by the ”sampling” method in ”Proofs of Retrievability” (PoR) [3] model and ”Provable Data Possession” (PDP) [4] model, later schemes [5], [6] tend to provide a probabilistic proof by accessing part of the file, which obviously enhances the auditing efficiency over earlier schemes.

Secondly, some auditing schemes [3], [7] provide private verifiability that require only the data owner who has the private key to perform the auditing task, which may potentially overburden the owner due to its limited computation capability. Ateniese el al. [4] were the first to propose to enable public verifiability in auditing schemes. In contrast, public auditing schemes [5], [6] allow anyone who has the public key to perform the auditing, which makes it possible for the auditing task to be delegated to an external third party auditor (TPA). A TPA can perform the integrity check on behalf of the data owner and honestly report the auditing result to him [8].

Thirdly, PDP [4] and PoR [3] intend to audit static data that are seldom updated, so these schemes do not provide data dynamics support. But from a general perspective, data update is a very common requirement for cloud applications. If auditing schemes could only deal with static data, their practicability and scalability will be limited. On the other hand, direct extensions of these static data oriented schemes to support dynamic update may cause other security threats, as explained in [6]. To our knowledge, only schemes in [6], [9], [10] provide built-in support for fully data dynamic operations (i.e., modification, insertion and deletion), but they are insufficient in providing data dynamics support, public verifiability and auditing efficiency simultaneously, as will be analyzed in the section of related work.

To address the fairness problem in auditing, we introduce a third-party arbitrator(TPAR) into our threat model, which is a professional institute for conflicts arbitration and is trusted and payed by both data owners and the CSP. Since a TPA can be viewed as a delegator of the data owner and is not necessarily trusted by the CSP, we differentiate between the roles of auditor and arbitrator. Moreover, we adopt the idea of signature exchange to ensure metadata correctness and provide dispute arbitration, where any conflict about auditing or data update can be fairly arbitrated.

Generally, this paper proposes a new auditing scheme to address the problems of data dynamics support, public verifiability and dispute arbitration simultaneously. Our contributions mainly lie in:

- We solve the data dynamics problem in auditing by introducing an index switcher to keep a mapping between block indices and tag indices, and eliminate the passive effect of block indices in tag computation without incurring much overhead.
- We extend the threat model in current research to provide dispute arbitration, which is of great significance and practicality for cloud data auditing, since most existing schemes generally assume an honest data owner in their threat models.
- We provide fairness guarantee and dispute arbitration in our scheme, which ensures that both the data owner and the cloud can not misbehave in the auditing process or else it is easy for a third-party arbitrator to find out the cheating party.

The rest of the paper is organized as follows. Section 2 introduces the system model, threat model and our design goals. In Section 3 and 4, we elaborate on our dynamic auditing scheme and arbitration protocols. Further, we present the security analysis and performance evaluation in Sections 5 and 6, respectively. Section 7 surveys the related work. Finally, Section 8 concludes the paper.

1.2 Existing System

Earlier auditing schemes usually require the CSP to generate a deterministic proof by accessing the whole data file to perform integrity check, such plain solutions incur expensive computation overhead at the server side, and hence they lack efficiency and practicality when dealing with large-size data. Recently proposed schemes such as “provable data possession” and “proofs of retrievability” are designed to audit static archive data and therefore lack of data dynamics support. Moreover, threat models in these schemes usually assume an honest data owner and focus on detecting a dishonest cloud service provider despite the fact that clients may also misbehave.

DISADVANTAGES

- Lack of data dynamics support.

- Lack efficiency and practicality when dealing with large-size data.

1.3 Proposed System

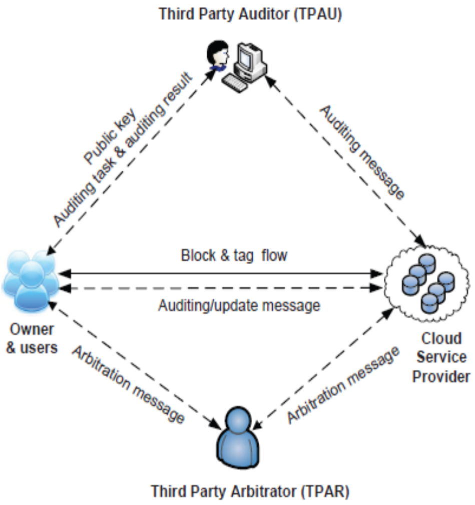
Generally, this paper proposes a new auditing scheme to address the problems of data dynamics support, public verifiability and dispute arbitration simultaneously. Our contributions mainly lie in:

- We solve the data dynamics problem in auditing by introducing an index switcher to keep a mapping between block indices and tag indices, and eliminate the passive effect of block indices in tag computation without incurring much overhead.
- We extend the threat model in current research to provide dispute arbitration, which is of great significance and practicality for cloud data auditing, since most existing schemes generally assume an honest data owner in their threat models.
- We provide fairness guarantee and dispute arbitration in our scheme, which ensures that both the data owner and the cloud cannot misbehave in the auditing process or else it is easy for a third-party arbitrator to find out the cheating party.

ADVANTAGES

- We can easily detect the misbehavior.
- Solve the data dynamics problem in auditing.

II. ARCHITECTURE DIAGRAM



The system and threat model.

III. MODULES

The project has been divided into 4 different modules:

1. Third Party Auditor(TPAU)
2. Owner & Users
3. Cloud Service Provider
4. Third Party Arbitrator(TPAR)

3.1 Third Party Auditor (Tpau):

Tpau will view the original data of a file, it will convert the data in to blocks, tpau will convert the data to encrypted form and it will add data to the server. User or owner will send request to tpau to view and download the data. Tpau will provide key permission to user to download the original data. If data is modified by user, then tpau permission is necessary to view or download the modified data. Tpau will provide verification permission to user, to see his or her verification status.

3.2 Owner OR Users:

In this user will upload data but he can’t see the data because tpau permission is necessary, so he will send request to tpau, tpau will convert the data into encrypt form and add to server, then user can see the data that to., encrypt data. To view or download decrypted data again he need tpau key permission after getting permission he or she can download the decrypted data. If user modifies the data and if user want to download or view that data again tpau permission is needed after getting permission he can download data. To see verification status of user, user needs tpau permission after getting permission user can see verification status.

3.3 Cloud Service Provider:

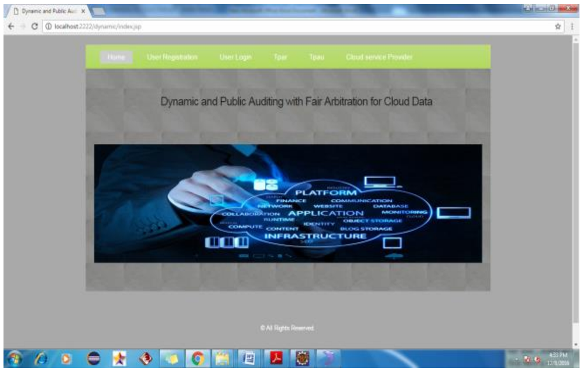
Csp is used to store data, when tpau add data to the server (csp) then only we can see data in csp and user can see the data. Csp can see the list of users and files and data.

3.4 Third Party Arbitrator (TPAR):

In this Tpar can see the user details.

IV. SCREENSHOTS

Home page:

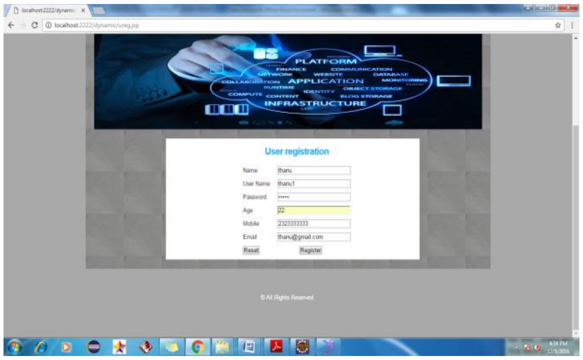


localhost:2222/dynamic/index.jsp

Dynamic and Public Auditing with Fair Arbitration for Cloud Data

© All Rights Reserved.

User registration page:



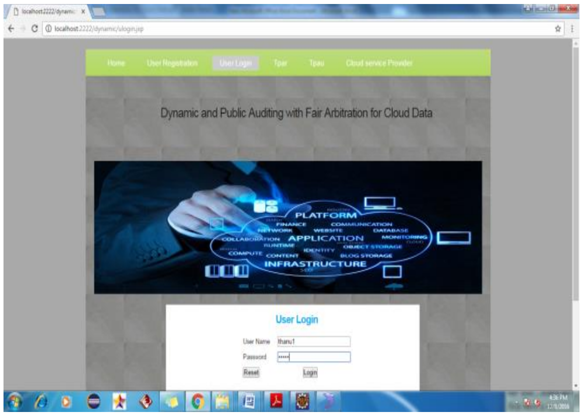
localhost:2222/dynamic/ureg.jsp

User registration

Name: thanu
User Name: thanu1
Password: ******
Age: 22
Mobile: 2323333333
Email: thanu@gmail.com

© All Rights Reserved.

User login page:



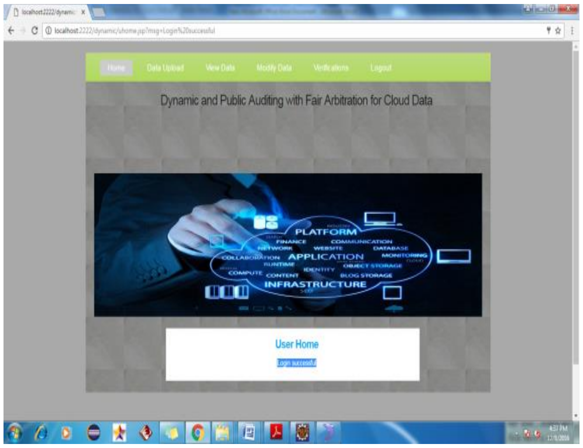
localhost:2222/dynamic/ulogin.jsp

Dynamic and Public Auditing with Fair Arbitration for Cloud Data

User Login

User Name: thanu1
Password: *****

User home page:



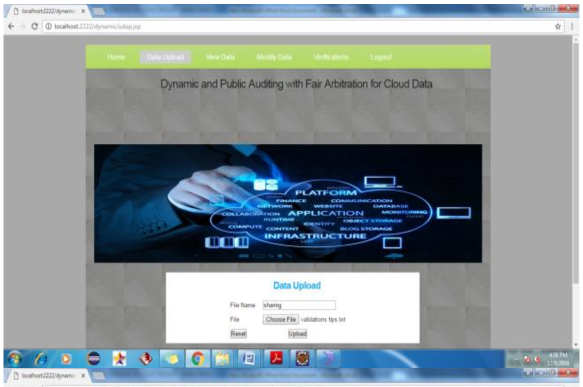
localhost:2222/dynamic/uhome.jsp?msg=Login%20successful

Dynamic and Public Auditing with Fair Arbitration for Cloud Data

User Home

Login successful

Data upload page:

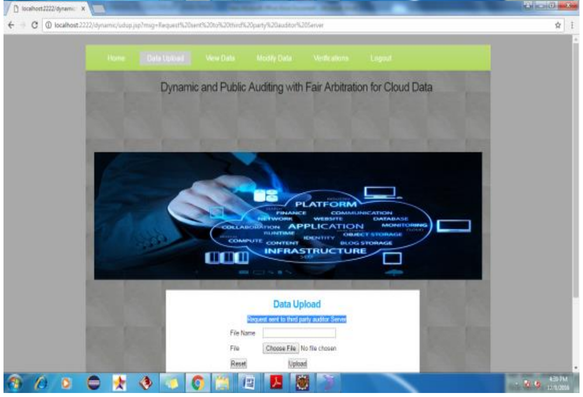


localhost:2222/dynamic/udup.jsp

Dynamic and Public Auditing with Fair Arbitration for Cloud Data

Data Upload

File Name: sharing
File: validations tips.txt



localhost:2222/dynamic/udup.jsp?msg=Request%20sent%20to%20third%20party%20auditor%20Server

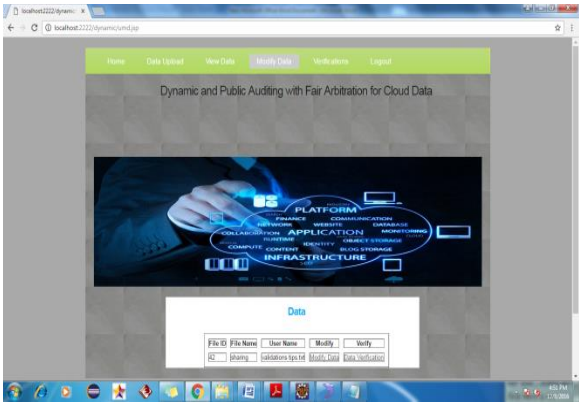
Dynamic and Public Auditing with Fair Arbitration for Cloud Data

Data Upload

Request sent to third party auditor Server

File Name:
File: No file chosen

In Modify data page, user can modify the data:



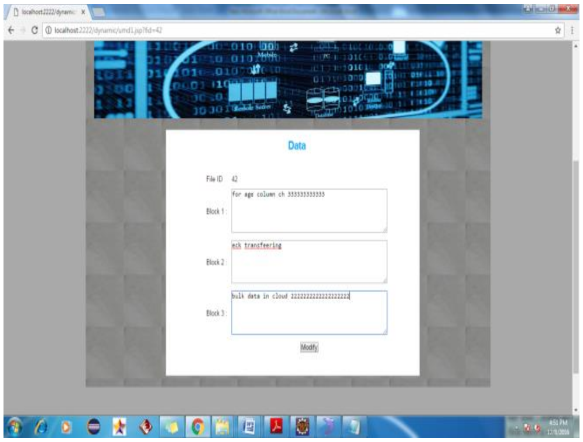
localhost:2222/dynamic/umd.jsp

Dynamic and Public Auditing with Fair Arbitration for Cloud Data

Data

File ID	File Name	User Name	Modify	Verify
42	sharing	validations tips.txt	Modify Data	Data Verification

(Add or delete) data to modify and click:

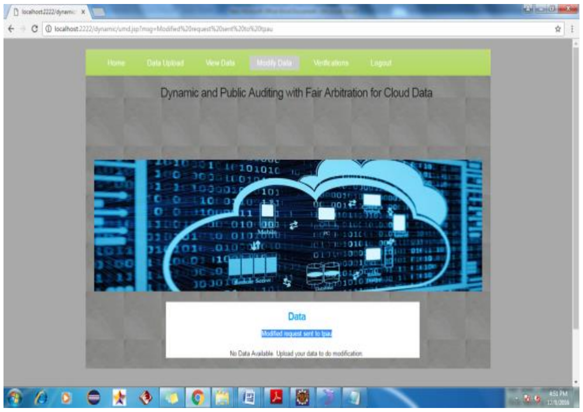


localhost:2222/dynamic/umd1.jsp?fid=42

Data

File ID: 42
Block 1: for age column ch 333333333333
Block 2: eck transfeering
Block 3: bulk data in cloud 2222222222222222222

Modified data request is sended to tpau:


localhost:2222/dynamic/umd.jsp?msg=Modified%20request%20sent%20to%20tpau
Home
Data Upload
View Data
Modify Data
Verifications
Logout

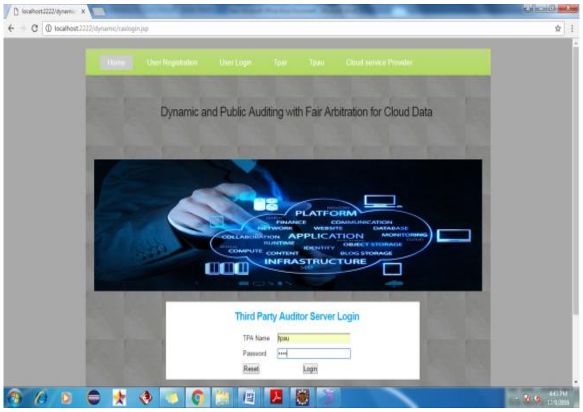
Dynamic and Public Auditing with Fair Arbitration for Cloud Data

Data

Modified request sent to tpau

No Data Available. Upload your data to do modification.

Tpau login page:



Dynamic and Public Auditing with Fair Arbitration for Cloud Data

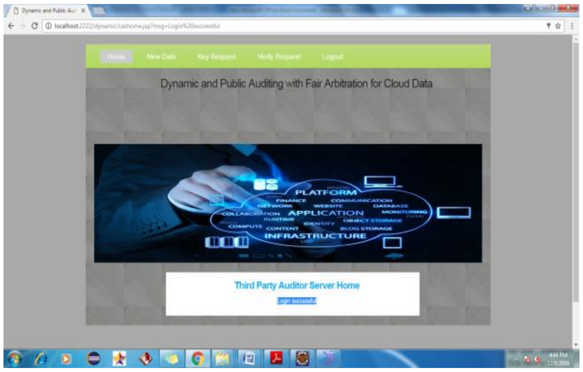
Third Party Auditor Server Login

TPA Name: tpau

Password: ****

Reset Login

Tpau home page:

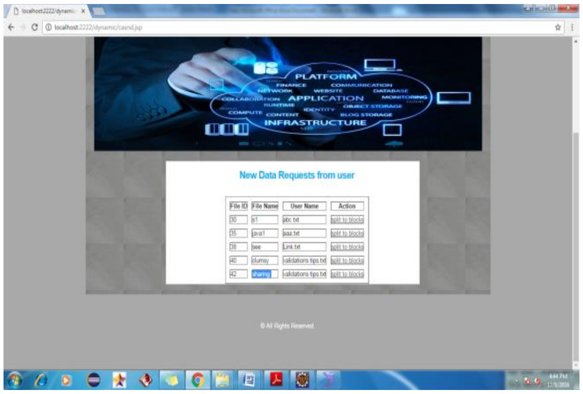


Dynamic and Public Auditing with Fair Arbitration for Cloud Data

Third Party Auditor Server Home

Login successful

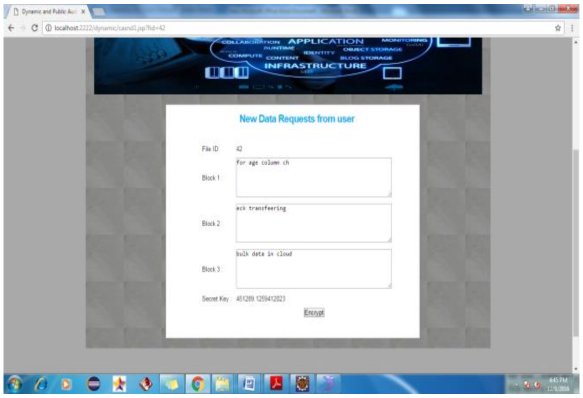
New data page:



New Data Requests from user

File ID	File Name	User Name	Action
30	s1	abc.txt	split to blocks
35	java1	aaa.txt	split to blocks
38	see	Link.txt	split to blocks
40	clumsy	validations tips.txt	split to blocks
42	sharing	validations tips.txt	split to blocks

Split to blocks, uploaded file will split into blocks



New Data Requests from user

File ID: 42

Block 1 : for age column ch

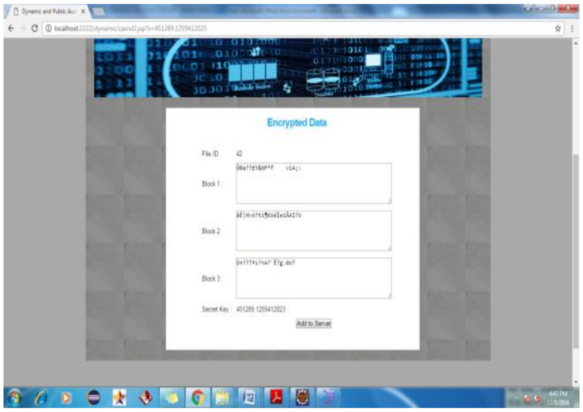
Block 2 : eck transfeering

Block 3 : bulk data in cloud

Secret Key : 451289.1259412023

Encrypt

Click on encrypt, it will convert into encrypt form



Encrypted Data

File ID: 42

Block 1 : Ùßa??ÉY&6P³f vlA¡¦

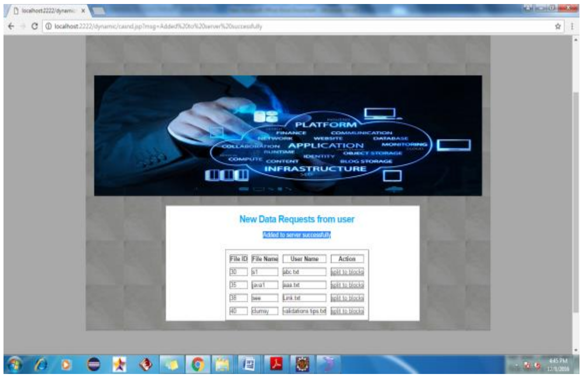
Block 2 : ã€|H>ó?±§ßöâÎ¤±ÃAI?V

Block 3 : Ù+???#s?+A?' É?g.èW?

Secret Key : 451289.1259412023

Add to Server

Click on add to server, so data will store in server

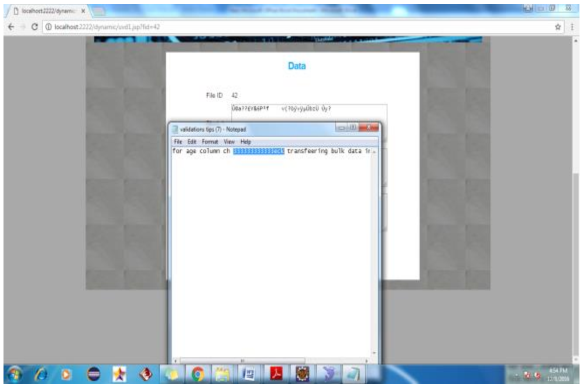

localhost:2222/dynamic/casnd.jsp?msg=Added%20to%20server%20successfully

New Data Requests from user

Added to server successfully

File ID	File Name	User Name	Action
30	s1	abc.txt	split to blocks
35	java1	aaa.txt	split to blocks
38	see	Link.txt	split to blocks
40	clumsy	validations tips.txt	split to blocks

User downloaded modified data


localhost:2222/dynamic/usd1.jsp?fid=42
validations tips (7) - Notepad
File
Edit
Format
View
Help

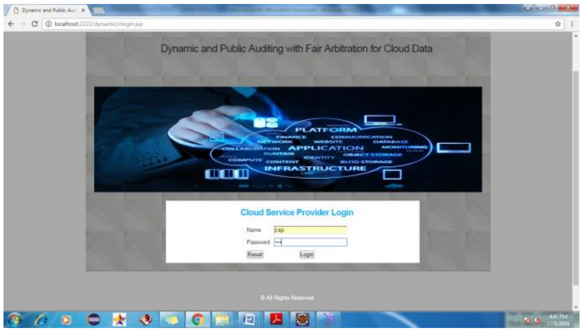
Data

File ID: 42

Ùßa??ÉY&6P³f v(?ôÿ÷ÿµÛbzÙ Ùy?

for age column ch ¶¶¶¶¶¶¶¶¶¶¶¶¶¶ transfeering bulk data in

Cloud service provider(csp) login:



localhost:2222/dynamic/clogin.jsp

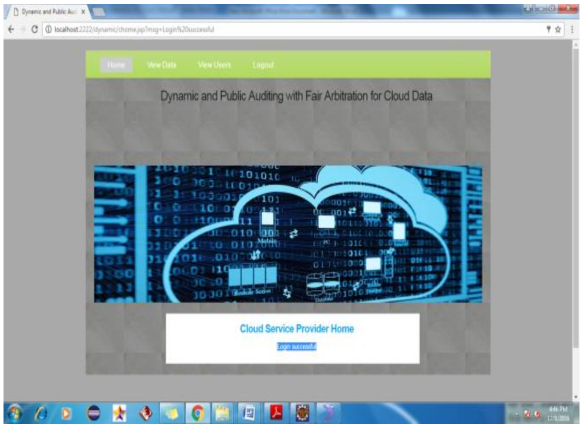
Dynamic and Public Auditing with Fair Arbitration for Cloud Data

Cloud Service Provider Login

Name: csp

Password: •••

Csp home page:

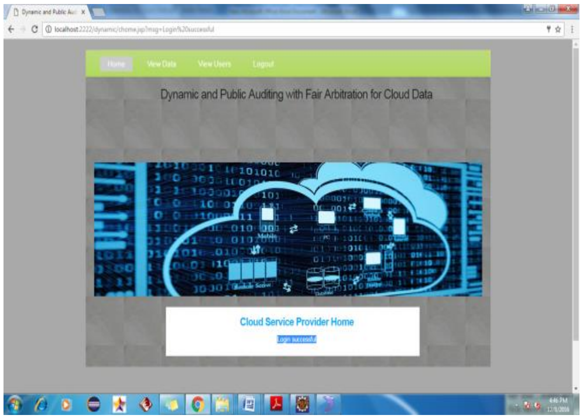


Dynamic and Public Auditing with Fair Arbitration for Cloud Data

Cloud Service Provider Home

Login successful

Click on view data:

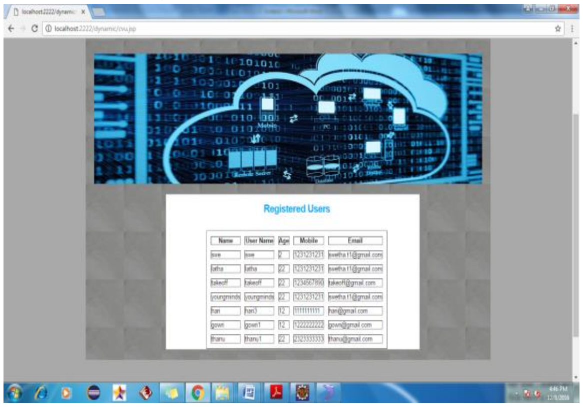


Dynamic and Public Auditing with Fair Arbitration for Cloud Data

Cloud Service Provider Home

Login successful

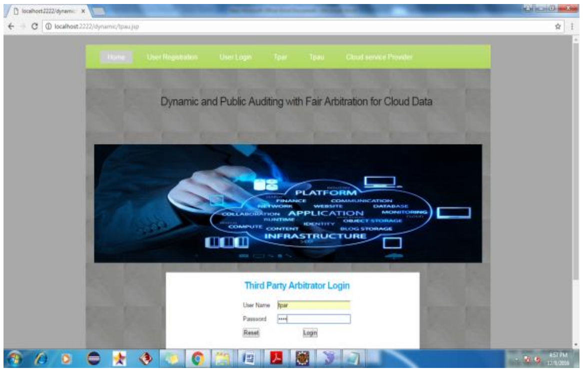
View users page:



Registered Users

Name	User Name	Age	Mobile	Email
swe	swe	2	1231231231	swetha.t1@gmail.com
latha	latha	22	1231231231	swetha.t1@gmail.com
takeoff	takeoff	22	1234567890	takeoff@gmail.com
youngminds	youngminds	22	1231231231	swetha.t1@gmail.com
han	han3	12	1111111111	han@gmail.com
gown	gown1	12	1222222222	gown@gmail.com
thanu	thanu1	22	2323333333	thanu@gmail.com

Tpar login page:



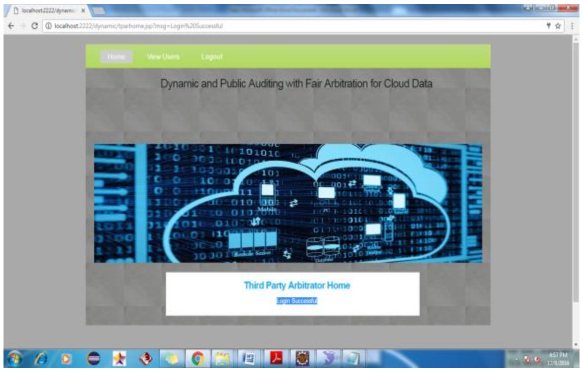
Dynamic and Public Auditing with Fair Arbitration for Cloud Data

Third Party Arbitrator Login

User Name: tpar

Password: •••

Tpar home page:



Dynamic and Public Auditing with Fair Arbitration for Cloud Data

Third Party Arbitrator Home

Login Successful

V. CONCLUSION

The aim of this paper is to provide an integrity auditing scheme with public verifiability, efficient data dynamics and fair disputes arbitration. To eliminate the limitation of index usage in tag computation and efficiently support data dynamics, we differentiate between block indices and tag indices, and devise an index switcher to keep block-tagindex mapping to avoid tag re-computation caused by block update operations, which incurs limited additional overhead, as shown in our performance evaluation. Meanwhile, since both clients and the CSP potentially may misbehave during auditing and data update, we extend the existing threat model in current research to provide fair arbitration for solving disputes between clients and the CSP, which is of vital significance for the deployment and promotion of auditing schemes in the cloud environment. We achieve this by designing arbitration protocols based on the idea of exchanging metadata signatures upon each update operation. Our experiments demonstrate the efficiency of our proposed scheme, whose overhead for dynamic update and dispute arbitration are reasonable.

REFERENCES

References for the project development were taken from the following books and web sites.

[1] Y. Deswarte, J.-J. Quisquater, and A. Sa¨ıdane, “Remote integrity checking,” in Proc. 5th Working Conf. Integrity and Intl Control in Information Systems, 2004, pp. 1–11.

[2] D. L. Gazzoni Filho and P. S. L. M. Barreto, “Demonstrating data possession and uncheatable data transfer.” IACR Cryptology ePrint Archive, Report 2006/150, 2006.

[3] A. Juels and B. S. Kaliski Jr, “Pors: Proofs of retrievability for large files,” in Proc. 14th ACM Conf. Computer and Comm. Security (CCS07), 2007, pp. 584–597.

[4] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, “Provable data possession at untrusted stores,” in Proc. 14th ACM Conf. Computer and Comm. Security (CCS07), 2007, pp. 598–609.

[5] H. Shacham and B. Waters, “Compact proofs of retrievability,” in Proc. 14th Intl Conf. Theory and Application of Cryptology and Information Security: Advances in Cryptology (ASIACRYPT 08), 2008, pp. 90–107.

[6] Q. Wang, C. Wang, J. Li, K. Ren, and W. Lou, “Enabling public verifiability and data dynamics for storage security in cloud computing,” in Proc. 14th European Conf. Research in Computer Security (ESORICS 08), 2009, pp. 355–370.

[7] M. A. Shah, R. Swaminathan, and M. Baker, “Privacy-preserving audit and extraction of digital contents.” IACR Cryptology ePrint Archive, Report 2008/186, 2008.

[8] C. Wang, K. Ren, W. Lou, and J. Li, “Toward publicly auditable secure cloud data storage services,” Network, IEEE, vol. 24, no. 4, pp. 19–24, 2010.

[9] C. Erway, A. K¨upc¸ ¨ u, C. Papamanthou, and R. Tamassia, “Dynamic provable data possession,” in Proc. 16th ACM Conf. Computer and Comm. Security (CCS 09), 2009, pp. 213–222.

[10] Y. Zhu, H.Wang, Z. Hu, G.-J. Ahn, H. Hu, and S. S. Yau, “Dynamic audit services for integrity verification of outsourced storages in clouds,” in Proc. ACM Symp. Applied Computing (SAC 11), 2011, pp. 1550–1557.