SECURING CLOUD DATA UNDER KEY EXPOSURE

E. Venkatesh[1], Ms. E. Durga Nandhini[2], Ms. Sarika Jain[3], Dr. S. Geetha[4]

[1]M. Sc – CFIS, Department of Computer Science and Engineering, Dr. M.G.R Educational and Research Institute, Chennai 600 095, Tamilnadu, India

[2,3]Center of Excellence in Digital Forensics, Perungudi, Chennai 600 089, Tamilnadu, India

[4]Professor and Head, Department of Computer Science and Engineering, Dr. M.G.R Educational and Research Institute, Chennai 600 095, Tamilnadu, India

ABSTRACT

Recent news reveals a powerful attacker which breaks data confidentiality by acquiring cryptographic keys, by means of coercion or backdoors in cryptographic software. Once the encryption key is exposed, the only viable measure to preserve data confidentiality is to limit the attacker's access to the cipher text. This may be achieved, for example, by spreading cipher text blocks across servers in multiple administrative domains—thus assuming that the adversary cannot compromise all of them. Nevertheless, if data is encrypted with existing schemes, an adversary equipped with the encryption key, can still compromise a single server and decrypt the cipher text blocks stored therein. In this paper, we study data confidentiality against an adversary which knows the encryption key and has access to a large fraction of the cipher text blocks. To this end, we propose Bastion, a novel and efficient scheme that guarantees data confidentiality even if the encryption key is leaked and the adversary has access to almost all cipher text blocks. We analyze the security of Bastion, and we evaluate its performance by means of a prototype implementation. We also discuss practical insights with respect to the integration of Bastion in commercial dispersed storage systems. Our evaluation results suggest that Bastion is well-suited for integration in existing systems since it incurs less than 5% overhead compared to existing semantically secure encryption modes.

1. INTRODUCTION

The world recently witnessed a massive surveillance program aimed at breaking users' privacy. Perpetrators were not hindered by the various security measures deployed within the targeted services. For instance, although these services relied on encryption mechanisms to guarantee data confidentiality, the necessary keying material was acquired by means of backdoors, bribe, or coercion. In addition to the public and governmental outrage, another immediate reaction from the industry was an even larger apprehension to use third-party services, and in particular cloud services. If the encryption key is exposed, the only viable counter-measure is to limit the adversary's access to the ciphertext, e.g., by spreading it across multiple administrative domains, in the hope that the adversary cannot compromise all of them. However, this countermeasure does not entirely solve the problem. Even if the data is encrypted and dispersed across different administrative domains, an adversary equipped with the appropriate keying material can compromise a single server and decrypt ciphertext blocks stored therein.

2. EXISTING SYSTEM

Perpetrators were not hindered by the various security measures deployed within the targeted service. For instance, although these services relied on encryption mechanisms to guarantee data confidentiality, the necessary keying material was acquired by means of backdoors, bribe, or coercion If the encryption key is exposed, the only viable means to guarantee confidentiality is to limit the adversary's access to the cipher text, e.g., by spreading it across multiple administrative domains, in the hope that the adversary cannot compromise all of them. However, even if the data is encrypted and dispersed across different administrative domains, an adversary equipped with the appropriate keying material can compromise a server in one domain and decrypt cipher text blocks stored therein.

3. PROPOSED SYSTEM

We study data confidentiality against an adversary which knows the encryption key and has access to a large fraction of the cipher text blocks. The adversary can acquire the key either by exploiting flaws or backdoors in the key-generation software, or by compromising the devices that store the keys. As far as we are aware, this adversary invalidates the security of most cryptographic solutions, including those that protect encryption keys by means of secret-sharing (since these keys can be leaked as soon as they are generated, we propose Bastion, a novel and efficient scheme which ensures that plaintext data cannot be recovered as long as the adversary has access to at most all but two cipher text blocks, even when the encryption key is exposed. Bastion achieves this by combining the use of standard encryption functions with an efficient linear transform. In this sense, Bastion shares similarities with the notion of all-or-nothing transform. An AONT is not an encryption by itself, but can be used as a pre-processing step before

INTERNATIONAL JOURNAL OF PROGRESSIVE
RESEARCH IN ENGINEERING MANAGEMENT
AND SCIENCE (IJPREMS)

www.ijprems.com
editor@ijprems.com

Vol. 03, Issue 04, April 2023, pp : 307-316

e-ISSN :
2583-1062

Impact
Factor :
5.725

encrypting the data with a block cipher. This encryption paradigm—called AON encryption— was mainly intended to slow down brute-force attacks on the encryption key.

4. BLOCK DIAGRAM
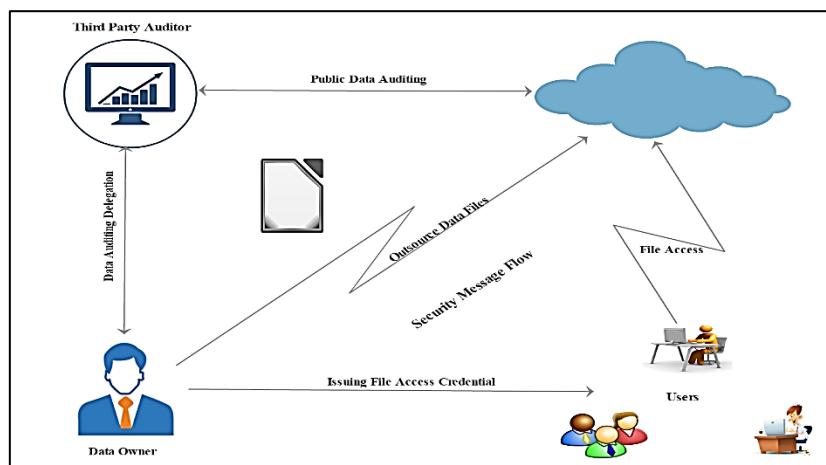
Figure 1. Block Diagram

5. OBJECTIVES

Our work shares similarities with the notion of shared key deniable encryption. An encryption scheme is "deniable" if—when coerced to reveal the encryption key—the legitimate owner reveals "fake keys" thus forcing the cipher text to "look like" the encryption of a plaintext different from the original one—hence keeping the original plaintext private. Deniable encryption therefore aims to deceive an adversary which does not know the "original" encryption key but, e.g., can only acquire "fake" keys. Our security definition models an adversary that has access to the real keying material. Secret sharing schemes allow a dealer to distribute a secret among a number of shareholders, such that only authorized subsets of shareholders can reconstruct the secret. In threshold secret sharing scheme, the dealer defines a threshold t and each set of shareholders of cardinality equal to or greater than t is authorized to reconstruct the secret. Secret sharing guarantees security against a non-authorized subset of shareholders; however, they incur a high computation/storage cost, which makes them impractical for sharing large files.

6. LITERATURE SURVEY

Qian Wang et.al, cloud computing has been envisioned as the next-generation architecture of IT Enterprise. It moves the application software and databases to the centralized large data centers, where the management of the data and services may not be fully trustworthy. This unique paradigm brings about many new security challenges, which have not been well understood. This work studies the problem of ensuring the integrity of data storage in Cloud Computing.

We first identify the difficulties and potential security problems of direct extensions with fully dynamic data updates from prior works and then show how to construct an elegant verification scheme for the seamless integration of these two salient features in our protocol design.

Peng Xu, et.al, public-key encryption with keyword search (PEKS) is a versatile tool. It allows a third party knowing the search trapdoor of a keyword to search encrypted documents containing that keyword without decrypting the documents or knowing the keyword. However, it is shown that the keyword will be compromised by a malicious third party under a keyword guess attack (KGA) if the keyword space is in a polynomial size. A malicious searcher can no longer learn the exact keyword to be searched even if the keyword space is small. We propose a universal transformation which converts any anonymous identity-based encryption (IBE) scheme into a secure PEFKS scheme. Following the generic construction, we instantiate the first PEFKS scheme proven to be secure under KGA in the case that the keyword space is in a polynomial size.

7. UML DIAGRAMS
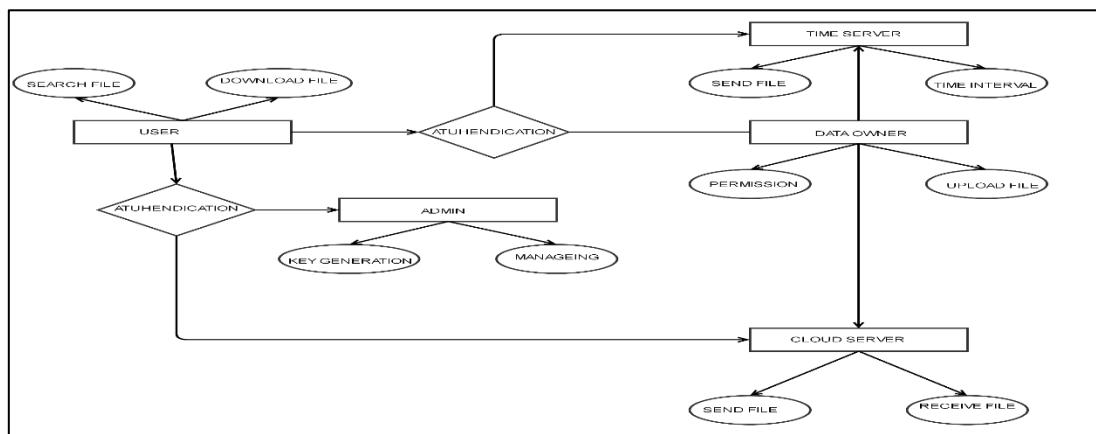
**Figure 2.** Entity Relationship Diagram
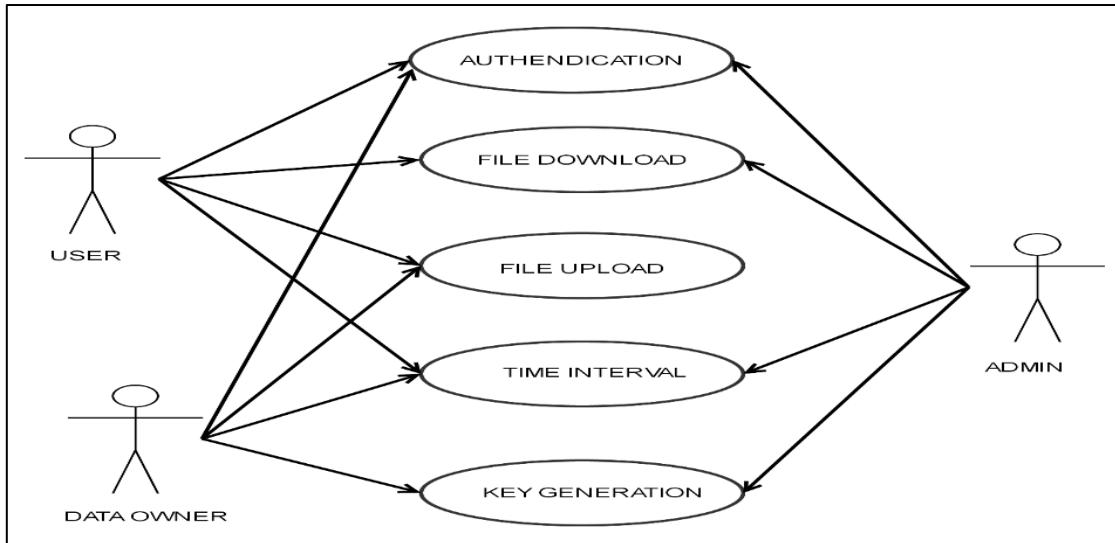

Figure 3. Use Case Diagram
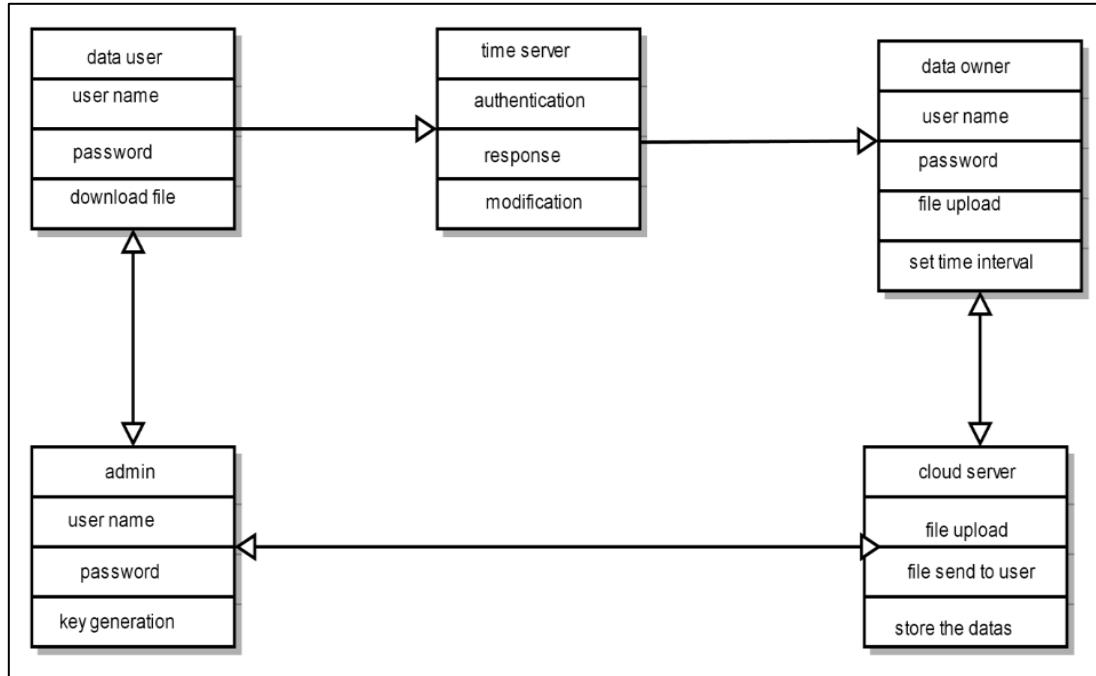
Figure 4. Class Diagram

**Data Flow Diagram**

Level 0

Level 1
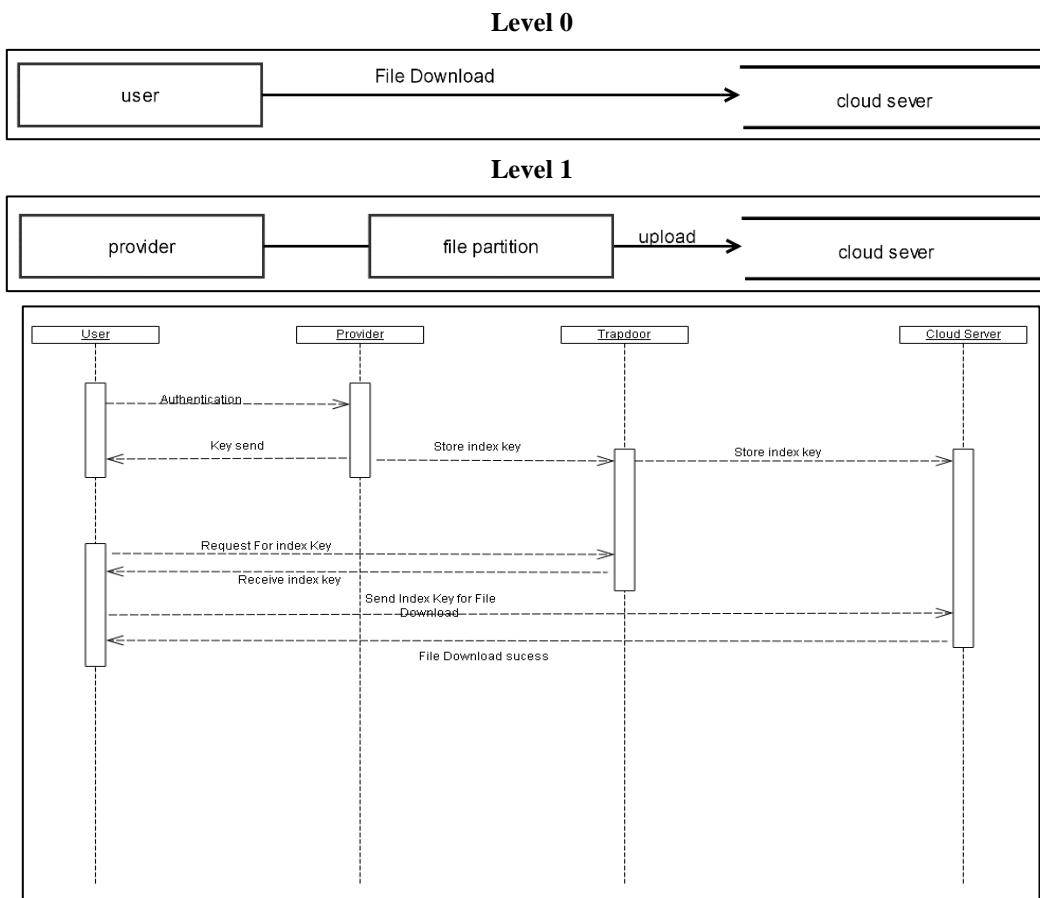
Figure 5. Sequence Diagram

8. MODULES

Module Description

- Data Owner Module
- Data User Module
- Admin Module.
- Time Server Module
- Cloud Server Module
- Secure Self-Destruction Scheme Module

Data Owner Module

Data owner can provide data or files that contain some sensitive information, which are used for sharing with his/her friends. All these shared data are outsourced to the cloud servers to store. And generate the three-attribute key for the data user. This attribute key is using for security. Ans additionally data owner allocates the file access time interval. For when data user will access the file in that particular time interval only. Data owner only a decryption the file and upload the file into a cloud sever.

Data User Module

The data users are having a registered users only when data user want to file from the cloud server that time send file request to the data owner. the data owner file upload for the that data user here security keys are generated from the data user mail and mobile number. that is unique to every user. and time interval also allocate from the data owner. That time interval only user can access the file from the cloud server.

Admin Module

In here admin have a monitoring the all-work flow from the file owner and file user activity and also all the private keys, and attribute keys are generated from the file owner and admin side that for security of the files. Admin has only sent the key generation for file uploading for data owner and generate the key for file download from the data user its used for the unauthorized user not access the file from the cloud server without key.

Time Server Module

The time server module is a most important module of in this project. Here file owner allocates the time interval in this module. When file owner uploads the file into the cloud server that time owner allocate the file download interval time. This is used for the other user can't access the file from the cloud server. Current system time will cross the allocate the ending time while that file will be deleted from the cloud.

Cloud Server Module

The cloud server module is when user and file owner registered that information and file owner upload the files details that type of all information are stored in this cloud server. Here data owner /user / admin data bases are stored in cloud sever and file upload to cloud sever and file retrieve from the cloud sever only. This is main part of our project. when unauthorized used cant accesses the file without authentication in cloud sever.

Secure Self-Destruction Scheme Module

Here we are using the generating the attribute from the user's information. For ex mail and mobile number are taken and generate the key. That key is generated from the file owner when file owner upload to the cloud server that time this operation will performed. These attributes are mostly used for the security for the files in cloud.

Cloud Computing

The term Cloud refers to a Network or Internet. In other words, we can say that Cloud is something, which is present at remote location. Cloud can provide services over network, i.e., on public networks or on private networks, i.e., WAN, LAN or VPN. Applications such as e-mail, web conferencing, customer relationship management (CRM), all run in cloud.

Basic Concepts

There are certain services and models working behind the scene making the cloud computing feasible and accessible to end users. Following are the working models for cloud computing:

● Deployment Models

● Service Models

9. DEPLOYMENT MODELS

Deployment models define the type of access to the cloud, i.e., how the cloud is located? Cloud can have any of the four types of access: Public, Private, Hybrid and Community

Public Cloud -The Public Cloud allows systems and services to be easily accessible to the general public. Public cloud may be less secure because of its openness, e.g., e-mail.

Private Cloud-The Private Cloud allows systems and services to be accessible within an organization. It offers increased security because of its private nature.

Community Cloud -The Community Cloud allows systems and services to be accessible by group of organizations.

Hybrid Cloud -The Hybrid Cloud is mixture of public and private cloud. However, the critical activities are performed using private cloud while the non-critical activities are performed using public cloud.

10. SERVICE MODELS

Service Models- are the reference models on which the Cloud Computing is based. These can be categorized into three basic service models as listed below:

• Infrastructure as a Service (IaaS)

• Platform as a Service (PaaS)

• Software as a Service (SaaS)

There are many other service models all of which can take the form like XaaS, i.e., Anything as a Service. This can be Network as a Service, Business as a Service, Identity as a Service, Database as a Service or Strategy as a Service.

Iaas- IaaS provides access to fundamental resources such as physical machines, virtual machines, virtual storage, etc., Apart from these resources, the IaaS also offers:

● Virtual machine disk storage

● Virtual local area network (VLANs)

● Load balancers

● IP addresses

● Software bundles

Paas

PaaS offers the runtime environment for applications. It also offers development & deployment tools, required to develop applications. PaaS has a feature of point-and-click tools that enables non-developers to create web applications. Google's App Engine, Force.com are examples of PaaS offering vendors. Developer may log on to these websites and use the built-in API to create web-based applications. But the disadvantage of using PaaS is that the developer lock-in with a particular vendor. For example, an application written in Python against Google's API using Google's App Engine is likely to work only in that environment. Therefore, the vendor lock-in is the biggest problem in PaaS. The following diagram shows how PaaS offers an API and development tools to the developers and how it helps the end user to access business applications.

Saas

Software as a Service (SaaS) model allows to provide software application as a service to the end users. It refers to a software that is deployed on a hosted service and is accessible via Internet. There are several SaaS applications, some of them are listed below:

- Billing and Invoicing System
- Customer Relationship Management (CRM) applications
- Help Desk Applications
- Human Resource (HR) Solutions

11. SCREENSHOTS

Homepage
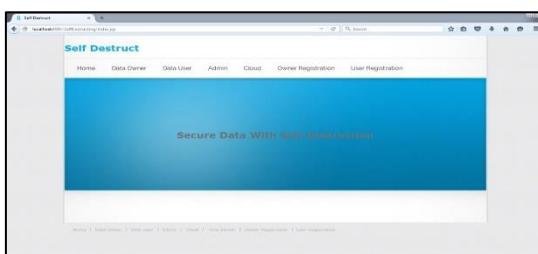
Data Owner Login
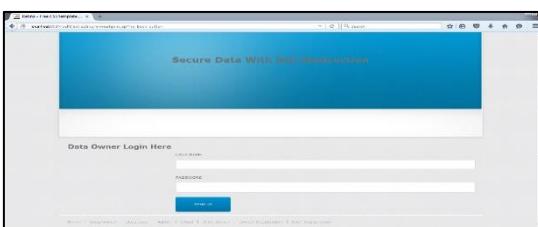
Data User Login
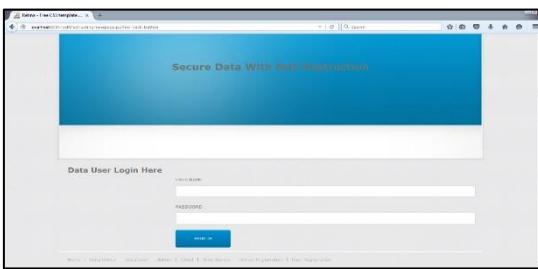
Admin Login
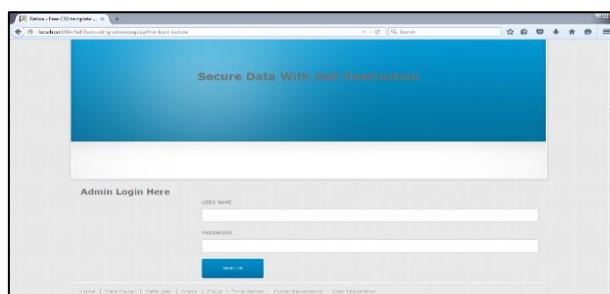
Cloud Login
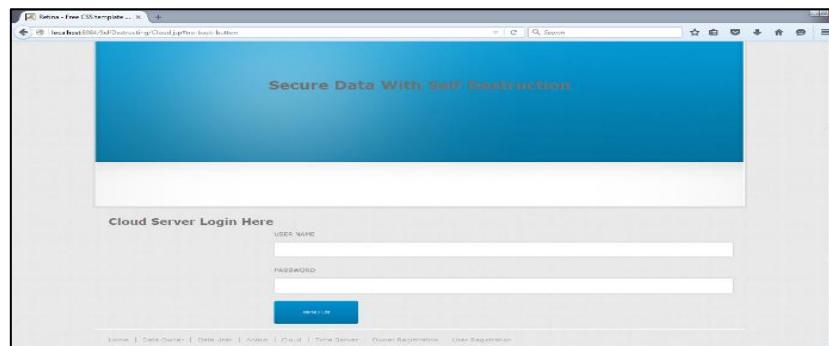
Owner Registration
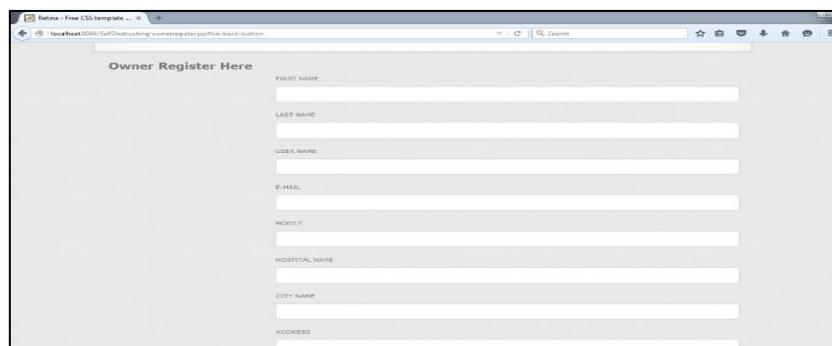
User Registration
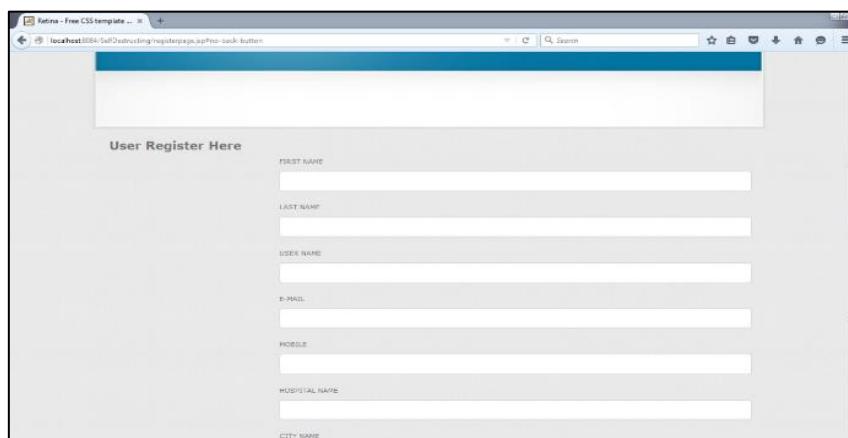
User Request
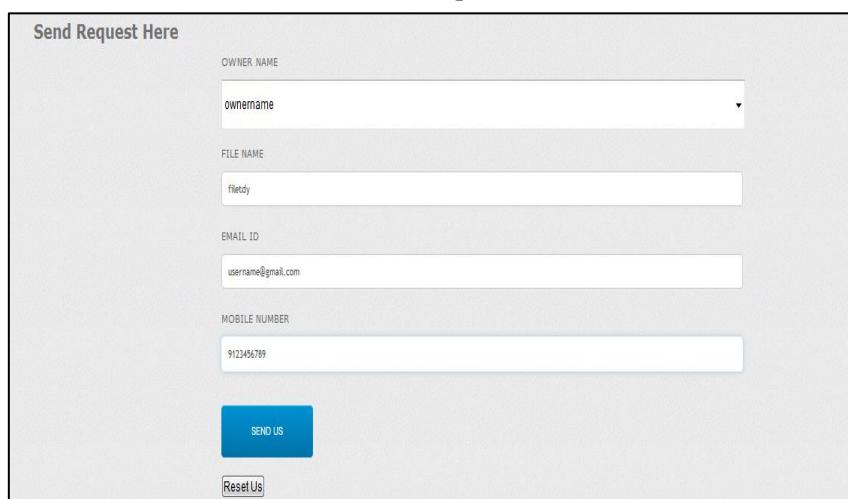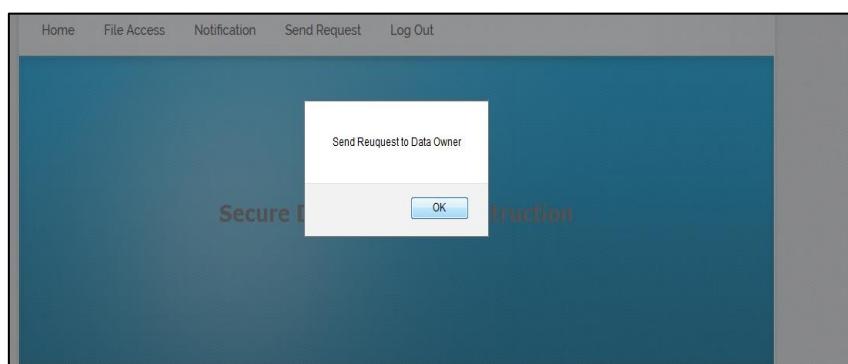
Owner View Request
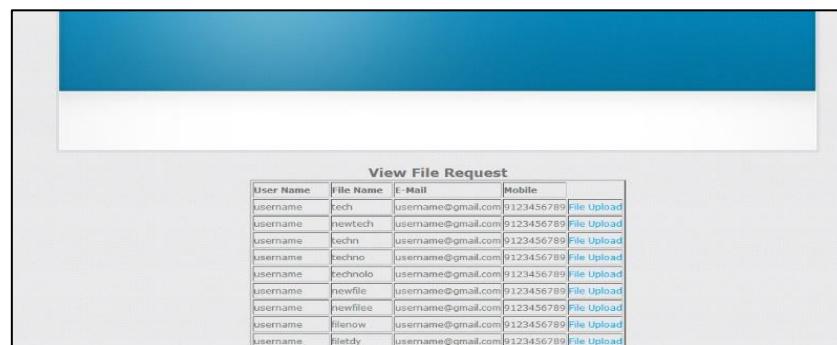
Owner Key Request
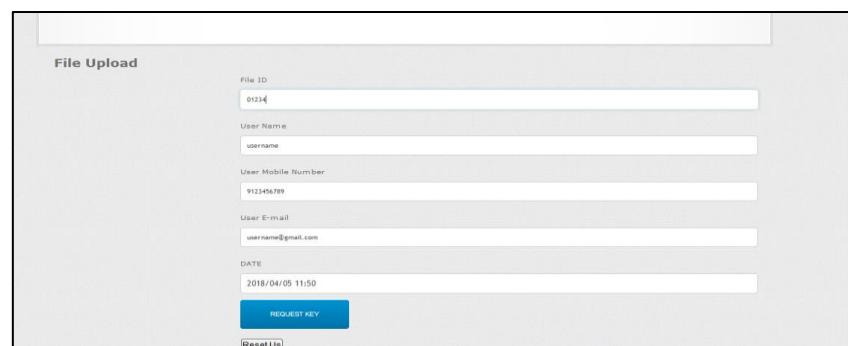
Owner Time Allocate

Owner File Upload

Data User Access

Data User Time Access

Data User File Download

12. CONCLUSIONS

With the rapid development of versatile cloud services, a lot of new challenges have emerged. One of the most important problems is how to securely delete the outsourced data stored in the cloud. In this paper, we proposed a novel KP-TSABE scheme which is able to achieve the time specified cipher text in order to solve these problems by implementing flexible fine-grained access control during the authorization period and time-controllable self-destruction after expiration to the shared and outsourced data in cloud computing. We also gave a system model and a security model for the KP-TSABE scheme. Furthermore, we proved that KP-TSABE is secure under the standard model with the decision l-Expanded BDHI assumption. The comprehensive analysis indicates that the proposed KP-TSABE scheme is superior to other existing schemes.

13. REFERENCES

[1]   Security Analysis of Attribute Revocation in Multi-Authority Data Access Control for Cloud Storage System, Jianan Hong, Kaiping Xue, Member, IEEE, and Wei Li**,** 2015

[2]   Hybrid Attribute- and Re-Encryption-Based Key Management for Secure and Scalable Mobile Applications in Clouds,Piotr K. Tysowski and M. Anwarul Hasan, Senior Member, IEEE, 2013.

[3]   A secure data self-destructing scheme in cloud computing, Jinbo Xiong, Ximeng Liu, Zhiqiang Yao, Jianfeng Ma, Qi Li, Kui Geng, and Patrick S. Chen,2014.

[4]   Oruta: Privacy-Preserving Public Auditing for Shared Data in the Cloud, Boyang Wang, Baochun Li,2014.

[5]   Mona: Secure Multi-Owner Data Sharing for Dynamic Groups in the Cloud, Xuefeng Liu, Yuqing Zhang, Member, IEEE, Boyang Wang, and Jingbo Yan,2014.