ACCESS CONTROL BASED DATA SECURITY IN CLOUD COMPUTING

A DISSERTATION

Submitted in partial fulfillment of the requirements for the award of the degree of

MASTER OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

By

SONAM CHUGH

DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY ROORKEE
ROORKEE - 247 667 (INDIA)
MAY, 2012

CANDIDATE'S DECLARATION

I hereby declare that the work, which is being presented in the dissertation entitled "**ACCESS CONTROL BASED DATA SECURITY IN CLOUD COMPUTING**" towards the partial fulfillment of the requirement for the award of the degree of **Master of Technology** in **Computer Science and Engineering** submitted in the Department of Electronics and Computer Engineering, Indian Institute of Technology Roorkee, Roorkee, India is an authentic record of my own work carried out during the period from July 2011 to June 2012, under the joint guidance of **Dr. P. Sateesh Kumar,** Department of Electronics and Computer Engineering, IIT Roorkee.

The matter presented in this dissertation has not been submitted by me for the award of any other degree of this or any other Institute.

Date: 23.05.12

Sonam

Place: Roorkee **(SONAM CHUGH)**

CERTIFICATE

This is to certify that the above statement made by the candidate is correct to the best of my knowledge and belief.

Date: 23.05.12

Place: Roorkee **(Dr. P. Sateesh Kumar)**

(Supervisor)

Dr. P. SATEESH KUMAR
Assistant Professor
Deptt. of Electronics & Computer Engg.
Indian Institute of Technology Roorkee
ROORKEE-247 667 (U.K.) INDIA

ACKNOLEDGEMENTS

First and foremost, I would like to extend my sincere gratitude to my guide and mentor Dr. P. Sateesh Kumar, Department of Electronics and Computer Engineering, Indian Institute of Technology Roorkee, for his invaluable advices, guidance, support, and for sharing his broad knowledge. It was an opportunity for me to learn more and to increase my knowledge in the field.

I am greatly indebted to all my friends, who have graciously applied themselves to the task of helping me with ample moral supports and valuable suggestions.

Last but not the least; I would like to thank my parents, my family members for supporting me in every endeavor I have engaged in and to everyone else who has knowingly or unknowingly helped me carry out this work.

Sonam

SONAM CHUGH

ABSTRACT

Cloud Computing provides on demand IT resources, computation and services on pay-per-use basis. In Cloud Computing, storage is also offered as a service, where data owner can store their data in the cloud. Data is biggest asset to an organization & how could confidentiality, authentication and access control be outsourced. There is a threat to data owner that if CSP (Cloud Service Provider) is malicious or CSP has some vulnerability. Hence, data owner must have some way of ensuring the data is confidential from CSP.

In this dissertation entitled "Access Control Based Data Security in Cloud Computing", a framework is proposed which secures the text files using hybrid cloud infrastructure with which the data security threat in Cloud technology can be solved. This model also provides an access control technique in which access to data based on user privileges. Encryption and authentication are two security measures you can use to keep your data safe on a cloud storage provider. Encryption maintains confidentiality, while authentication ensures only legitimate user accesses the data. Authentic users first encrypt their data and then stores into the cloud. Typically file encryption mechanism includes metadata attached to the protected object that contains information about how to decrypt the protected object. This metadata is part of the encrypted file header and is always inserted at the beginning of the file. This metadata allows individual users to access the file.

One of the enhancements in this framework is to use concept of group key. In a typical organization users are grouped into groups. A Group Key is a key that is used by all users in a group. Main purpose of this application is to use a new type of key called "Group Key" that represents a group of users. File Encryption scheme will be enhanced to use group keys. With Group Key, it will be possible to modify the members of the group without affecting the Encrypted File Meta data associated with protected files. Data is one of the biggest assets for an organization, & my proposed framework is extensible enough that it allows an organization to decrypt the encrypted file.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS

CSP	Cloud Service Provider
FEK	File Encryption Key
DB	Database
MAC	Message Authentication Code
LDAP	Lightweight Directory Access Protocol
DO	Data Owner

Chapter 1

Introduction

1.1 Introduction

The cloud is not simply the latest fashionable term for the Internet. Though the Internet is a necessary foundation for the cloud, the cloud is something more than the Internet. The cloud is where you go to use technology when you need it, for as long as you need it, and not a minute more. You do not install anything on your desktop, and you do not pay for the technology when you are not using it. Cloud computing, where applications and files are hosted on a "cloud" consisting of thousands of computers and servers, all linked together and accessible via the Internet. Hence, you can access all your programs and documents from any computer that's connected to the Internet. It enables cloud customers to remotely store their data into the cloud so as to enjoy the on-demand high quality applications and services from a shared pool of configurable computing resources [1].

In cloud computing, end user can get service from provider via internet pay-as-per-use basis. The service may be any of the three types: Software-as-a-Service, Platform-as-a-Service or Infrastructure-as-a-Service. It also provides Storage-as-a-Service, where data owner can store their data in the cloud. Some of the examples of Cloud Storage Service Providers are Dropbox [2],Amazon's EC2 and S3 [3], iCloud [4] and Nirvanix [5] which provide data storage service in the pay-as-you-use fashion at relatively low prices. For example, Amazon's S3 data storage service just charges $0.12 to $0.15 per gigabyte month. As compared to building their own infrastructures, users are able to save their investments significantly by migrating businesses into the cloud.

The benefits brought by this new computing model include but are not limited to: relief of the burden for storage management, universal data access with independent geographical locations, and avoidance of capital expenditure on hardware, software, and personnel maintenances [6]. With the prevalence of cloud services, more and more sensitive information are being centralized into the cloud servers, such as emails, private videos and photos, company finance data, government documents, etc [7].

On the surface, cloud storage has several advantages over traditional data storage. For example, if you store your data on a cloud storage system, you'll be able to get to that data from any location that has Internet access. You wouldn't need to carry around a physical storage device or use the same computer to save and retrieve your information.

Some examples of Cloud Storage:

- **Dropbox** is a free cloud service that lets you bring your photos, docs, and videos anywhere and share them easily. Users can store documents into the shared folder so that other people (with whom the folder has been shared) can read them or even make edits.
- Web e-mail providers like Gmail, Hotmail and Yahoo! Mail store e-mail messages on their own servers. Users can access their e-mail from computers and other devices connected to the Internet.
- Sites like Flickr and Picasa host millions of digital photographs. Their users create online photo albums by uploading pictures directly to the services' servers.
- YouTube hosts millions of user-uploaded video files.

1.2Motivation

Cloud computing is a new paradigm in which computing resources such as processor, memory, software applications and storage are not physically present at the user's location. Instead, a service provider owns and manages these resources, and users access them via the Internet [6]. Cloud Storage provides infrastructure to users for storing their data. Also, data owners may share their outsourced data with a large number of users. Cloud storage services can offer lower storage rates because they more efficiently use the server space they have; space gets reassigned to users almost instantly, on an as-needed basis. It's a lot cheaper to use excess space in the cloud than it is to purchase a new server or hard disk drive. Although cloud computing provides a number of advantages that include economies of scale, dynamic provisioning, increased flexibility and low capital expenditures, it also introduces a range of new security risks

Data is one of the most valuable assets to an organization and how confidentiality, authentication and access control can be outsourced. The security of data present on cloud depends on CSP. If security of CSP breached / by passed / compromised, CSP malicious and shares data with competitors whole organization is at risk. There is a threat to data owner that if CSP is malicious or has some vulnerability. Hence, there is a need by which data owner can put the data on cloud yet maintaining the confidentiality from CSP. Also each organization audits their employee, hence keeping the data on cloud yet organization what to keep track of what their employees are doing. Hence, Data Security is the number one issue when it comes to cloud computing. Since a third party stores your data, you don't know what's going on with it. It's easy to worry about the security risks of a cloud solution.

1.3 Statement of the Problem

The main objective of this thesis is to design a hybrid cloud infrastructure scheme using cryptographic algorithms in order to provide a secure and robust solution that allows enterprises to audit the data on Cloud yet maintaining the confidentiality of data from CSP.

This problem can be divided into following phases:

- ❖ *To develop a framework for organization / enterprise using concepts of hybrid cloud that enables secure access (only legitimate person can decrypt the file) to files/data present on cloud. The framework should be extensible enough to work with any CSP. The framework should also give access of files to users based access control list.*
- ❖ *To develop an application which should have minimum file metadata so storage cost of encrypted file on cloud should not increase significantly.*
- ❖ *To develop an application where organization / enterprise can track what data is on cloud and if any need comes organization should be able to decrypt the encrypted files.*

1.4 Organization of the Thesis

This dissertation report comprises of six chapters including this chapter that introduces the topic and states the problem. The rest of the report is organized as follows:

Chapter 2 gives the background of Cloud Computing, brief literature review of related work including research gaps.

Chapter 3 describes the framework designed for access control based data security, the major components, their functionality and role of each in the system.

Chapter 4 gives the implementation details of the proposed framework.

Chapter 5 discusses the results with the snapshots of the application.

Chapter 6 concludes the dissertation work and gives suggestions for future work.

Chapter 2

Background and Literature Review

2.1 Cloud Computing

2.1.1 Basic Overview

Cloud computing can be defined as “a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g. networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction [1].” It relies heavily on virtualization, resilient computing, and low-cost or open source software, and it’s usually geographically dispersed and focused more on services than systems. Let’s starts with the revolution of cloud computing. The revelation of the cloud computing is started from early beginning of computer. Figure 2.1, adapted from [8], shows six phases of computing paradigms, given as below:

In phase 1 (mainframe system), many users shared powerful mainframes using dummy terminals. In phase 2 (personal computer), stand-alone PCs became powerful enough to meet the majority of users’ needs. In phase 3 (network computing), PCs, laptops, & servers were connected together through local networks to share resources and increase performance. In phase 4 (Internet), local networks were connected to other networks forming a global network such as the Internet to utilize remote applications and resources. In phase 5, grid computing provided shared computing power and storage through a distributed computing system. Finally, in phase 6, cloud computing further provides shared resources on the Internet in a scalable and simple way based on utility computing.

Comparing these six computing paradigms, it looks like that cloud computing is a return to the original mainframe computing paradigm. However, these two paradigms have several important differences, mainframe computing offers finite computing power, while cloud computing provides almost infinite power and capacity. In addition, in mainframe computing dummy terminals acted as user interface devices, while in cloud computing powerful PCs can provide local computing power and cashing support.

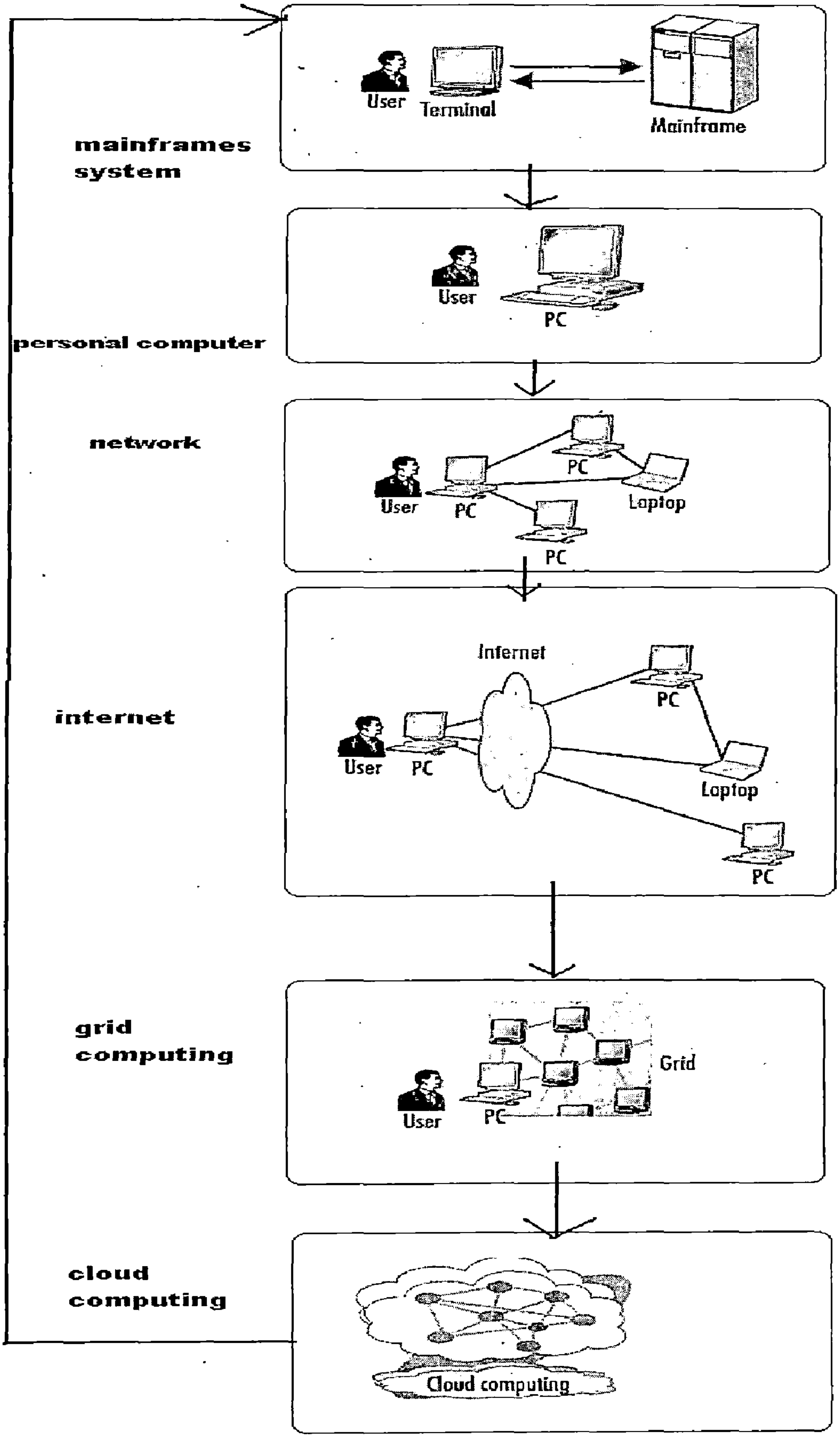


Figure 2-1 Cloud Computing Revolution [8]

Some important characteristics of cloud computing, which distinct it from any other distributed computing paradigm [9], is as follows:

- Pay as per as use: It is the most important characteristic of cloud computing. It means that end clients only pay for the compute resources they use, they will pay more if they use more or pay less for little use.
- On-demand service: This characteristic of cloud computing indicates that the load generation in cloud computing is dynamic rather than static i.e. user may not know that some application may need in feature, but if it is needed then it must be fulfilled by on demand service.
- Multitenency: Unlike previous computing models, which assumed dedicated resources (i.e. computing facilities dedicated to a single user or owner), cloud computing is based on a business model in which resources are shared (i.e., multiple users use the same resource, through virtualization technique) at the network level, host level, and the application level.
- Resiliency: Any service in this world can fail, nothing is 100% perfect. Cloud service may fail sometime. But failure of cloud server or resource must be isolated from cloud customer. The resiliency property achieves it.
- Rapid elasticity: allows quick scalability or downsizing of resources depending on demand i.e. a client may need 10 GB space in morning, 100 GB space after 2 hrs, again after using them he may release it, thus consumer can increase or decrease capacity at will.

After having a brief overview about the cloud and its main characteristics, let's have a look on cloud eco-system.

Cloud system is made up of several elements: clients, the datacenter, and distributed servers [9]. Cloud client are three types: thick client, thin client and mobile client. Thick client are generally PC having its own hard disk and memory. Thin client are light weighted computer having no hard disk and only capable to connect internet. The examples of mobile client are mobile hand set and PDA. Datacenter is a collection of servers where the user data or user subscribed application are placed.

Data center may not be centralized. It may be geographically distributed, called distributed servers.

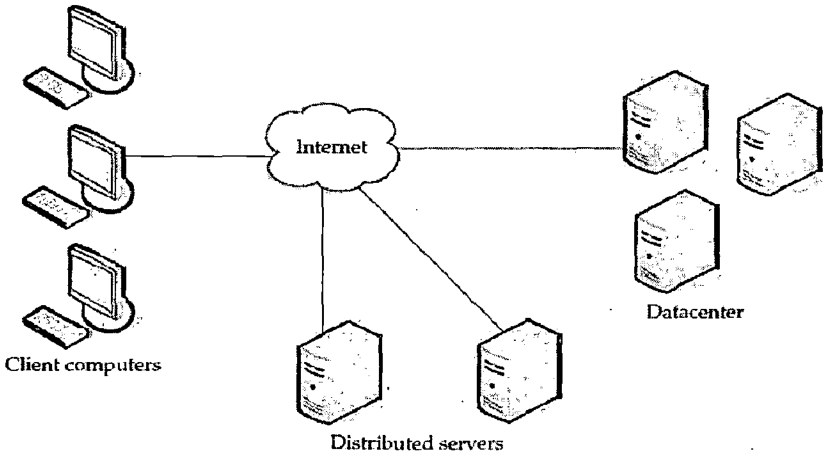


Figure 2-2 Cloud Components [9]

One of the most important ideas behind cloud computing is scalability, and the key technology that makes that possible is virtualization.

Virtualization is a technique which emulates one of more workstations/servers within a single physical computer [10]. This allows a single computer to take on the role of multiple computers. It provides necessary abstraction, encapsulation of different services and fault tolerance. Generally virtualization can be achieved by two ways: software virtualization and hardware virtualization [11]. In first case software image management and software code management can be used and in later case hardware can be used as plugged and play manner. There may also exist of different degree of virtualization: Full virtualization and Para virtualization [10].

Full virtualization is a technique in which a complete installation of one machine is run on another. Para virtualization allows multiple operating systems to run on a single hardware device at the same time by more efficiently using system resources, like processors and memory. We can use some virtualization software to create virtualization on a single physical machine, called hypervisor [11]. Hypervisor support multiple virtual systems, managing scheduling and accessing I/O device and process migration.

One can use some virtualization software to create virtualization on a single physical machine, called hypervisor [11]. Hypervisor support multiple virtual systems, managing scheduling and accessing I/O device and process migration.

Now, it's time to discuss about deployment and service model of cloud computing.

a) Deployment model of Cloud Computing

Deployment model of cloud computing [9] deals with the considerations when a cloud computing architects want to move from a standard enterprise application deployment model to one, based on cloud computing. There are three deployment models for Cloud computing:

i) **Private Cloud:** Private cloud (or Internal Cloud) refers to cloud computing on private networks. Private clouds may be owned by and managed by the organization or the designated service provider. Private clouds are built for the exclusive use of one client, providing full control over data, security, & quality of service.

ii) **Public Cloud:** Public clouds (or External Cloud) are run by third parties, and applications from different customers are likely to be mixed together on the cloud's servers, storage systems, and networks. A public cloud provides services to multiple customers.

iii) **Hybrid Cloud:** This model of Cloud computing is a composition of two or more Clouds (public or private) that remain unique entities but are bound together by standardized or proprietary technology that enables data and application portability.

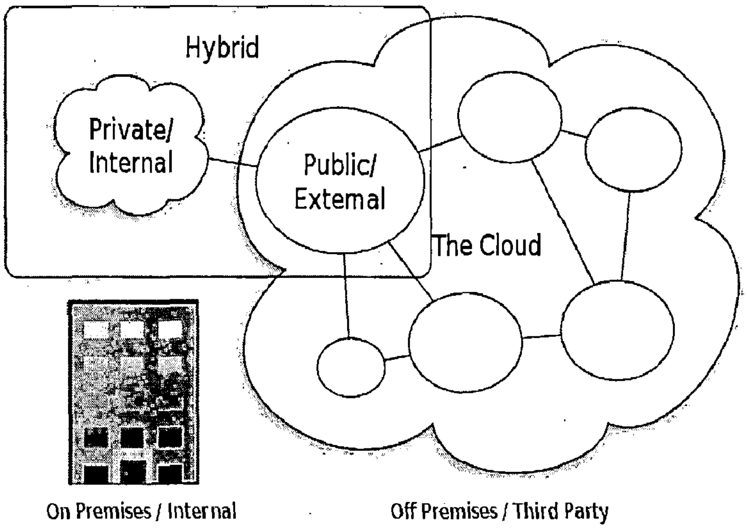


Figure 2-3 Deployment model of Cloud Computing [9]

b) Service model of Cloud Computing

The term services in cloud computing is the concept of being able to use “reusable, fine-grained, components” across a vendor’s network. This is widely known as “as a service”. There are three models by which Cloud computing services are delivered: Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS), IaaS is the foundation of all Cloud services (i.e. the bottom layer) and is overlaid with PaaS (the middle layer) and SaaS (the top layer), respectively.

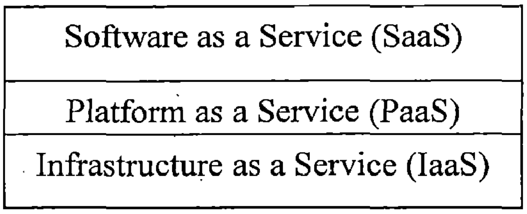

Software as a Service (SaaS)
Platform as a Service (PaaS)
Infrastructure as a Service (IaaS)

Figure 2-4 Service Model of Cloud Computing [9]

i) **Software as a Service (SaaS)** [12]: This is the most popular form of cloud services. Software or applications are provided to customers as services on demand. The services become accessible through web based user interfaces or available as web services. The clients (end users) however, cannot change or modify the software. One of the biggest benefits of SaaS is, of course, costing less money than buying the whole application. You just have to pay a monthly or annual fee to use the service. Some other benefits include reduce overhead of installation and maintenance of software etc. Google Apps, such as Googlemail, Googledoc, and Salesforce are prominent examples of SaaS.

ii) **Platform as a Service (PaaS)** [13]: Offers a platform to clients for different purposes. For example, the Windows Azure offers a platform to developers to build, test, and host applications that can be accessed by the end users. The end users may or may not know that the application is hosted on the cloud. As with the SaaS, you do not need to build the platform. You just pay a nominal fee for using the service. An example is Google’s App Engine which enables user to build web application.

iii) **Infrastructure as a Service (IaaS)** [13]: It is also known as a hardware as a service (HaaS), which mainly offers hardware so that your organization can put whatever they want onto it. HaaS allows us to “rent” such resources as Server space, Network equipment, Memory and Storage space. This infrastructure can be dynamically scaled up or down, based on the application resource needs. When you opt for IaaS, you save a lot on expenses, space, and personnel required to set up and maintain the infrastructure. The cloud service provider takes care of setting up and maintaining the infrastructure. You just pay a fee to use it per your requirements. Examples are Dropbox, Amazon S3, iCloud etc.

2.2 Literature Review

As cloud computing is a relatively new field, very less research has been done for tackling the issue of data security. Some proposed solutions are as follows:

Rajarajan et al [13] present a capability based access control technique that ensures only valid users will access the outsourced data. The authentic users get the data file by the Cloud Service Provider that is stored on the cloud in a confidential manner, capability based access control technique is used by the owner such that by which users he/she wants to share its’ data. Users cannot access other’s data files as there will be no capability granted by Data Owner for these users. Communication between CSP and user is made secure using Diffie-Hellman Key Exchange, as it generates a shared session key which is used for the encryption and decryption of data file.

Ateniese et al [14] proposed a secure distributed storage scheme based on proxy re-encryption. The data owner encrypts blocks of content with symmetric content keys. The content keys are all encrypted with a master public key. The data owner uses his master private key and user’s public key to generate proxy re-encryption keys, using which the semi trusted server can then convert the cipher text into plaintext for a specific user.

Miklau et al [15] present a framework for access control on published XML documents by using different cryptographic keys over different portions of XML tree. It also introduces special metadata nodes in the structure to enforce access control.

Yu et al [16] proposed a scheme to achieve fine-grained, secure, and scalable access control in cloud computing by combining techniques of attribute-based encryption (ABE), proxy re encryption, and lazy re-encryption. A set of attributes are associated to a file that are meaningful in the context of interest. The access structure of each user is defined as a logical expression over these attributes, which reflects the scope of data file that the user is allowed to access. A public key component is defined for each attribute. Data files are encrypted using the public keys corresponding to their attributes. User secret keys are defined matching their access structures so that a user is able to decrypt a cipher text if and only if the data file attributes satisfy his access structure.

2.3 Research Gaps and scope of work

After analyzing the previous work, a number of research gaps have been found which are discussed below:

- Untrusted Cloud Service Provider can attack the unencrypted data and leak the sensitive data.
- Using a single key for encryption is more dangerous, as if key is leaked then it effect whole data stored in the cloud by the data owner.
- If file encrypted to large audience, the size of meta-data increases drastically, this increases the cost of hosting data on cloud.
- Key storage at user side results in burden for user. And security threat as well since keys can be compromised easily in that case. In cloud scenario we don’t want to restrict user to use a particular computer in which its keys has been stored.
- Collaboration between a malicious server and any single malicious user takes place, they would expose decryption keys of all the encrypted data and compromise data security of the system. Also, Computational overhead on Cloud Service Provider as all work has been done by CSP.
- Cloud servers store a vast amount of data, deriving a unique logical expression for every user using the attributes of every file will become computationally complex.

Chapter 3

Proposed Framework for Access Control Based Data Security

Storing data on cloud can lead to leakage of sensitive information due to application and cloud vulnerabilities. In the given proposed work, the concept of hybrid cloud infrastructure (i.e. combination of public and private cloud) has been used for securing the text files on cloud yet maintains the confidentiality of data from CSP and auditing the employees. (The implementation is limited by text file since needed viewer that shows decrypted file, I implemented viewer for text file but in market viewers reading files of multiple format are available. These can be used.) Confidential files can be encrypted and uploaded on cloud from managed client. A managed client is a computer or a device that has software required to encrypt the file and communicate with internal cloud. This application would provide an Encryption solution to protect files stored on cloud and allow users to view/edit Encrypted files stored on cloud.

3.1 Components of Proposed Framework

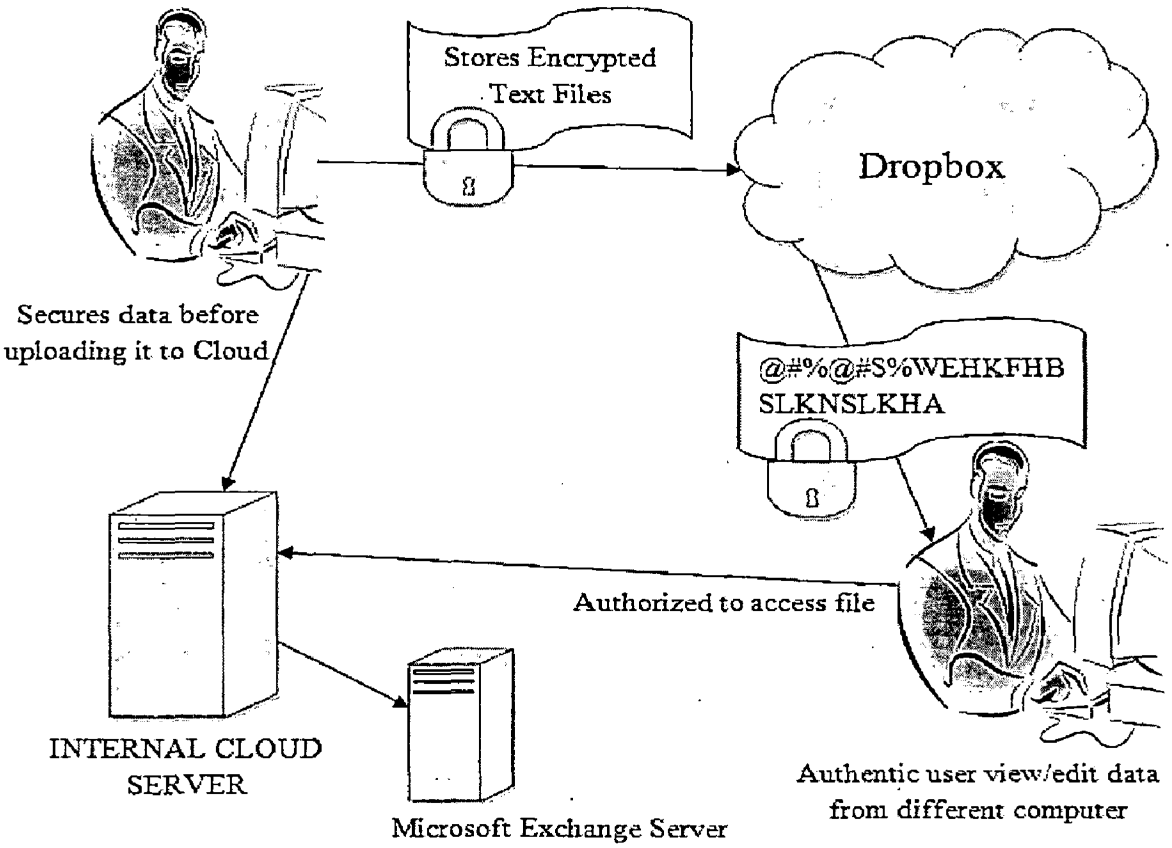


Figure 3-1 Design of Proposed Framework

The various components used in figure 3-1 are explained as follows:

3.1.1 Clients

- *Data Owner*: Owner is the part of the architecture; he can upload a file to the cloud after encryption. After successful login, owner selects the file, the file get encrypted locally and store it into cloud. It is important to note that, the data is actually encrypted in user side, so it avoids the possible attack of stilling data from cloud.
- *Data Users:* Users with whom the stored data is going to be shared. Here also, after successful login, user selects encrypted file from cloud, he uses the key to decrypt the data. Again the decryption is done on the client side to avoid possible data stilling attack in cloud.

3.1.2 External Cloud (Dropbox): External Cloud (also known as public cloud) will be only hosting the documents. It will only acts as Cloud as a Storage. The cloud provider provides storage service to store encrypted data. (Of course, as cloud storage cannot be trusted as it is provided by third party. We generally perform all encryption and decryption locally and only the encrypted user data is stored in cloud.) Here, Dropbox has been used for this purpose (one can also use any other cloud service that offers Storage as a service). Dropbox provides a cloud based service to enable users to store and share files and folders with others across the Internet using file synchronization [2]. Each user stores their files in the dropbox (into the shared folder).

3.1.3 Internal Cloud Server: Private Cloud (Internal Cloud) will be used for the authentication, key management, access control and auditing the users' activity.

- Authentication [17]: The purpose of authentication is to prevent unauthorized access to user data. Both the client users and cloud service providers must be authenticated before using the data; this requirement will surely reduce the risk of information leakage. You can use various technologies to authentication, such as passwords, certificates, and biometrics and so on.
- Key Management [17]: The activities such as data encryption store have many key used; therefore, the key determines the security of data

indirectly. The key management can used to solute the related problem of key, which include key generation, release, storage, use, share, update, archive and destruction. Key management will be one of the most complex problems in data protection and in the proposed framework is done by Internal Cloud.

- Access Control: It allows the user access to various resources based on the user's identity. It provides different privileges to different users (assign by owner of the data).
- Audit the Data: An organization / enterprise can audit what their employees are putting on cloud. It can be done by internal cloud. All encrypted files can be decrypted by internal cloud as all the keys are stored in its database.

Internal Cloud Server is used to maintain database of information about registered groups and users, storing keys of users / groups, message authentication code of the files & the usernames and group names along with their different privileges. Hibernate [18] and Postgresql [19] has been used for this purpose (one can also use any other RDBMS). *Postgresql* is an object-relational database management system.

Hibernate is an object relational mapping (ORM) library for the Java language, providing a framework for mapping an object-oriented domain model to a traditional relational database. Hibernate solves object-relational impedance mismatch problems by replacing direct persistence-related database accesses with high-level object handling functions.

Features of Hibernate:

- Mapping from Java classes to database tables (and from Java data types to SQL data types).
- Provides data query and retrieval facilities.
- Generates the SQL calls and attempts to relieve the developer from manual result set handling and object conversion and keep the application portable to all supported SQL databases with little performance overhead.

Benefits of Using Hibernate:

- It provides the facility of saving objects in the database and instead of string it returns the object at the time of retrieval.
- It is Database Independent. As in our model, we are using postgresql, one can also use oracle or any other database available in market.

3.1.4 Microsoft Exchange Server: My proposed framework leverages on existing resources for authentication purposes. Every organization maintains information regarding their users in various forms. One of the widely used repository for maintain this information is Microsoft Active Directory [20]. *Active Directory* is a directory service created by Microsoft for Windows domain networks. It is based on the LDAP protocol for authentication purposes. It is a database that keeps track of all the user accounts and passwords in your organization. It also provides the facility of groups. Active Directory Groups contain users who are called members of the group. All permissions, authorizations and restrictions placed on the group apply to all the members of the group. For authentication purpose, a user credentials (i.e. username and password) has been used.

3.2 Brief Description of Proposed Model

One can store their data in the cloud but introduces the threat that CSP might leak the confidential data of organization. Encryption and authentication are two security measures you can use to keep your data safe on a cloud storage provider. Encryption maintains confidentiality, while authentication ensures only legitimate user accesses the data. Authentic users first encrypt their data and then stores into the cloud. Encrypting your data before it is sent to the service provider ensures that if the provider's security measures are breached, your data is still secure. If someone does get your data, they need the proper credentials for viewing the files or all they get is gibberish.

Initially, Users will enroll by using their existing LDAP login credentials. After a server has been specified and authenticated, users can enroll by using their existing LDAP login credentials to download their keys from the Internal Cloud Server. The end-user will be presented with a login screen where they can enter their LDAP login credentials. Upon authentication, the user's keys will be downloaded from the server and onto the device. Users can then begin to encrypt/decrypt content.

Typically file encryption mechanism includes metadata attached to the protected object that contains information about how to decrypt the protected object. This metadata is part of the encrypted file header and is always inserted at the beginning of the file. This metadata allows individual users to access the file.

Two keys used for encrypting a file:

1. File Data encrypted to unique File Encryption Key (FEK). This is also referred as Symmetric Key.
2. FEK encrypted to each user's public key. This is also referred as Encrypted Symmetric Keys (ESK).

If numbers of users who access encrypted file grow, associated metadata grows as well. This increases the encryption overhead significantly if original files size is small. Additionally, as new users joins and leaves the group, the management of metadata associate with each file causes significant maintenance issues with no easy way to manage it. And the only way to manage it is re-encrypt the file which is costlier operation.

A group is basically a distribution list whose members will be resolved first in order to retrieve the individual keys for every group member. File Encrypting application then uses the member keys to protect the encrypted object. However, changes to the group membership require changes to the protected objects. On the other hand, this group feature does not require an infrastructure for Re-Encryption and thus is immune to all sorts of Group Membership changes.

In a typical organization users are grouped into groups. A Group Key is a key that is used by all users in a group. Main purpose of this application is to use a new type of key called "Group Key" that represents a group of users. File Encryption scheme will be enhanced to use group keys. With Group Key, it will be possible to modify the members of the group without affecting the Encrypted File metadata associated with protected files. Also, organization / enterprise can audit what their employees are putting on cloud. Moreover if an organization wants to decrypt the file, it can be done by internal cloud as it has all the user keys.

Chapter 4
Implementation of Proposed Framework

The project has been implemented in Java programming language using the Eclipse 3.7.2 Integrated Development Environment (IDE). The Eclipse IDE is written in Java and runs everywhere where a JVM (Java Virtual Machine) is installed, including Windows, Mac OS, Linux, and Solaris. A JDK (Java Development Kit) is required for Java development functionality. The Eclipse Platform allows applications to be developed from a set of modular software components called modules. The IDE is available as a free download from internet. It also supports multithreading, all type of encryption and decryption scheme, database access, which are required for the implementation.

This Java based File Encrypting application has been implemented on Windows XP Operating System. It consists of two important modules which are described in the succeeding subsections and the functions that are used to implement it. Basic Modules in the framework are:

- ❖ Implementation of Security Module
- ❖ Implementation of Sub-Modules
    - o Graphical User Interface (GUI)
    - o Client Server Communication Interface
    - o Database Interface
    - o LDAP Interface

4.1 Modules in proposed framework

4.1.1 Implementation of Security Module

To implementation of all security algorithms of File Encryption Scheme (FES), we used an external library of Java called Crypto [21]. The toolkit Crypto can help us speed up development when cryptography is involved. It provides lot of inbuilt function to implement all security aspects like Hash functions, Encryption algorithms and Public-key algorithms.

4.1.1.1 Encryption / Decryption Algorithm

In our FES, we have used both symmetric key and asymmetric / public key cryptography.

a) Symmetric key encryption / decryption is used to encrypt/decrypt the file. Before storing the file into the cloud, we encrypt the data of the file to maintain the confidentiality and integrity of the data present in file. When shared user wants to view/edit the file, the file will be decrypted by using the same symmetric key.

The sample code for encryption and decryption is given as below:

```
1.  import javax.crypto
2.  keygen = KeyGenerator.getInstance(“AES”)
    Cipher cipher = Cipher.getInstance("AES")
3.  text of file= 'original data'
4.  encryptFile(file1, file2, key)
            cipher.init(Cipher.ENCRYPT_MODE, key)
5.  cipher_text present in file
6.  '\xec\xc2\x9e\xd9] a\xd0'
7.  decryptFile(file2, file3, key)
            cipher.init(Cipher.DECRYPT_MODE, key)

8.  'original data'
```

This sample code shows how we can perform AES using Java. Let’s look at one of the block cipher: AES. The key size used by this cipher is 8 bytes and the block of data it works with is 8 bytes long. The simplest mode for this block cipher is the electronic code book mode where each block is encrypted independently to form the encrypted text.

b) Asymmetric key encryption/decryption is used for the encryption/decryption of the File Encryption Key (FEK, key using which the data of the file has been encrypted). FEK encrypted to each user’s or groups' public key and decrypted by using users' or groups' private key. This is also referred as Encrypted Symmetric Keys (ESK). The sample code is given below:

```
1.  import javax.crypto
2.  random_generator = Random.new().read
3.  key = RSA.generate(1024, random_generator)
4.  public_key = key.publickey()
    private_key = key.privatekey()
5.  Cipher cipher = Cipher.getInstance("RSA/ECB/PKCS1Padding");
    cipher.init(Cipher.ENCRYPT_MODE, public_key);
6.  Cipher cipher = Cipher.getInstance("RSA/ECB/PKCS1Padding");
    cipher.init(Cipher.DECRYPT_MODE, private_key);
```

The above code illustrates the RSA algorithm. It is easy to generate a private/public key pair with "crypto". We need to specify the size of the key in bits: we picked 1024 bits. Larger the number of bits more is the security but more time is also required in encryption and decryption. We also need to specify a random number generator function; we use the Random module of crypto for that. Now that we have our key pair, we can encrypt/decrypt some data.

4.1.1.2 Hash Algorithm

The MD5 Message-Digest Algorithm is a widely used cryptographic hash function that produces a 128-bit (16-byte) hash value. MD5 is used to check data integrity. An MD5 hash is typically expressed as a 32-digit hexadecimal number. To implement MD5, we used an external library of Java called Security [22].

The sample code is given as below:

```
1.  import java.security.
2.  MessageDigest md
3.  md = MessageDigest.getInstance("MD5")
```

4.1.2 Implementation of Sub-Modules

4.1.2.1 Graphical User Interface

The GUI is provided only for client part of application, using which a user can request to encrypt/decrypt the file and connect to Internal Cloud. To implement GUI, we used an external library of Java called Swing [23]. Swing is the primary Java GUI widget toolkit. It is part of Oracle's Java Foundation Classes (JFC) — an API for providing a graphical user interface (GUI) for Java programs. Swing package called javax.swing contains a lot of classes that will create GUI components for us. Swing components are not implemented by platform-specific code. Instead they are written entirely in Java and therefore are platform-independent. The term "lightweight" is used to describe such an element.

4.1.2.2 Client Server Communication Interface

Sockets are used for client server communication. It is one end-point of a two-way communication link between two programs running on the network. Socket classes are used to represent the connection between a client program and a server program. The java.net package provides two classes--Socket and ServerSocket--that implement the client side of the connection and the server side of the connection, respectively. To implement sockets, java package java.net [24] has been used. It provides the classes for implementing networking applications. Since our server needs to serve multiple clients simultaneously, multithreading has been used. Java provides built-in support for multithreaded programming. A multithreaded program contains two or more parts that can run concurrently. Each part of such a program is called a thread, and each thread defines a separate path of execution. In the multithreading concept, several multiple lightweight processes are run in a single process/task or program by a single processor.

4.1.2.3 Database Interface

This module is responsible for storing the information about registered users & groups, what privilege each user and group has and message authentication code of the content of the file.

Table 4-1 List of Functions used for Internal Cloud Database

1	createUser()	Inserts user information like user name, password and its public & private key.
2	createGroup()	Inserts group information like group name, password and its public & private key.
3	setMAC()	Stores Message Authentication Code corresponding to its file id and owner id.
4	setUser_write_access()	Stores user id that has write access permission with file id.
5	setGroup_write_access()	Stores group id that has write access permission with file id.
6	setUser_read_access()	Stores user id who has read access permission with file id.
7	setGroup_read_access()	Stores. group id that has write access permission with file id.

Internal Cloud maintains Database, whose DB Schema is shown as below:

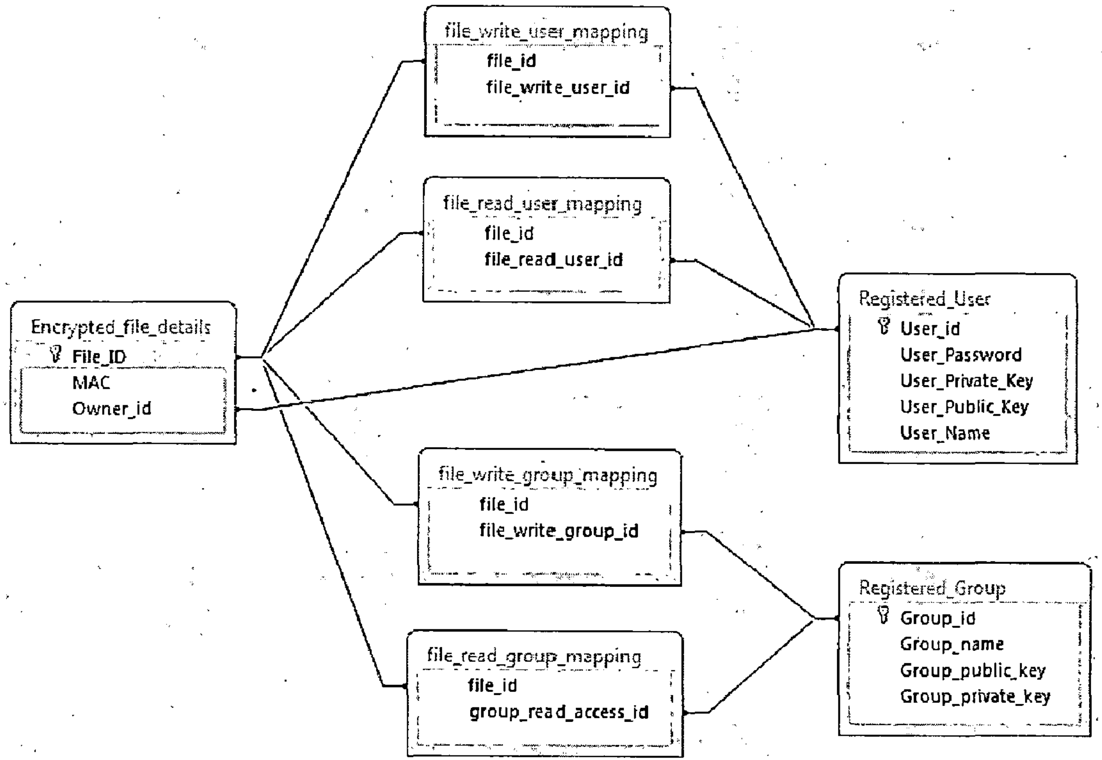


Figure 4-1Schema of the Database maintained by Internal Cloud

4.1.2.4 LDAP Interface

To query Active Directory LDAP call are used. We used an external library of Java called Naming [25] for LDAP call. The Java Naming and Directory Interface (JNDI) is a Java API for a directory service that allows Java software clients to discover and look up data and objects via a name. Like all Java APIs that interface with host systems, JNDI is independent of the underlying implementation. Additionally, it specifies a service provider interface (SPI) that allows directory service implementations to be plugged into the framework.

4.2 Details of Implementation of Proposed Work

Initially, user logins to the cloud by entering the credentials to the UI screen. User writes its credentials to the UI Screen and presses the Submit button. The application passes the username and password to the server to check whether the login user is authenticated or not. Three cases can arise at the time of login:

- Username and password, both present in the Internal Cloud Server Database (DB).
- Only username presents in the DB.
- Both username & password, not present in the DB.

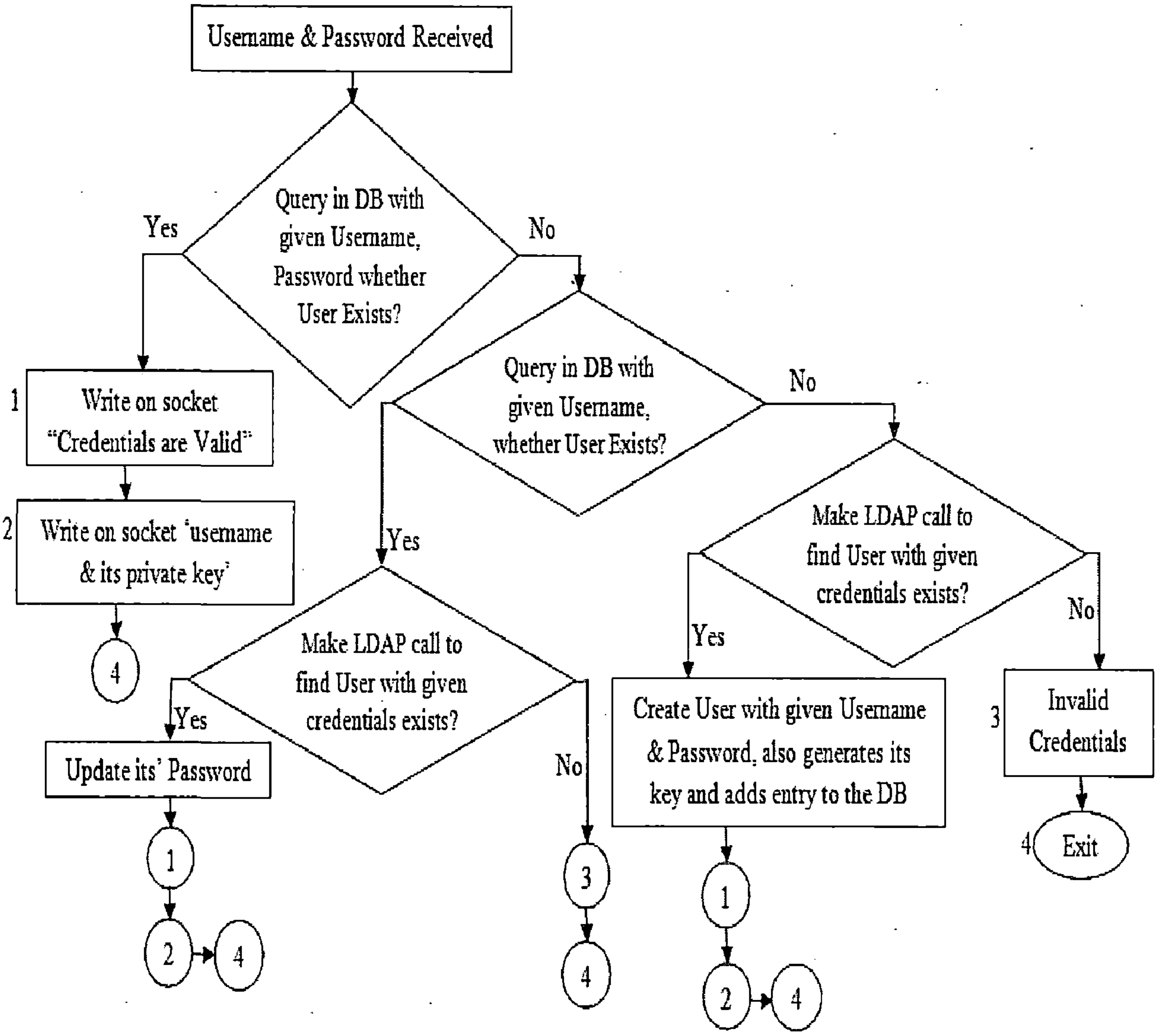
Explanation of the above mentioned cases are given as below:

- *Username and password, both present in the DB*

  This case arises when user who has already logged in to Internal Cloud logs in again. After receiving the username and password, the internal cloud server first queries in its database with given username and password, whether the credentials are correct or not, if it is, then it replies back to the client that credentials are valid and send its' private key. If it's not, then the next case occurs.

- *Only username presents in the DB*

  This case arises when a user has a file encrypted to him but not logged in on server yet. If a user has a file encrypted to him, then a user is created in DB with dummy password but valid keys. Upon receiving user credentials, server makes LDAP call to Microsoft Exchange Server to check in its Active Directory whether the given credentials are valid or not, if it is then, service provider updates the credentials (password) in its' database and also replies back to the client that credentials are valid and send its' private key. If the credentials are not present in Active Directory, then server simply returns "Invalid Credentials".



Username & Password Received
Query in DB with given Username, Password whether User Exists?
Yes
No
1
Write on socket "Credentials are Valid"
2
Write on socket 'username & its private key'
4
Query in DB with given Username, whether User Exists?
No
Yes
Make LDAP call to find User with given credentials exists?
Yes
No
Update its' Password
1
2
4
3
4
Make LDAP call to find User with given credentials exists?
Yes
No
Create User with given Username & Password, also generates its key and adds entry to the DB
1
2
4
3
Invalid Credentials
4
Exit

Figure 4-2 Diagram represents the Login Structure

- *Both username & password, not present in the DB*

This case arises when neither file encrypted to user nor logged in yet. The internal cloud server makes LDAP call to Microsoft Exchange Server to check in its Active Directory whether the credentials are valid or not, if it is, then service provider adds the credentials in its' database, also generate users' keys and replies back to the client that credentials are valid and send its' private key. If it is not valid (hence the user does not exist within the organization), then the server simply returns "Invalid Credentials".

4.2.1 Steps performed at Owners' Side / Steps of Encryption

User logins into the cloud and loaded with its' private key. Now, if user is part of some group, then the user has been notified by the server to different groups to which it belongs.

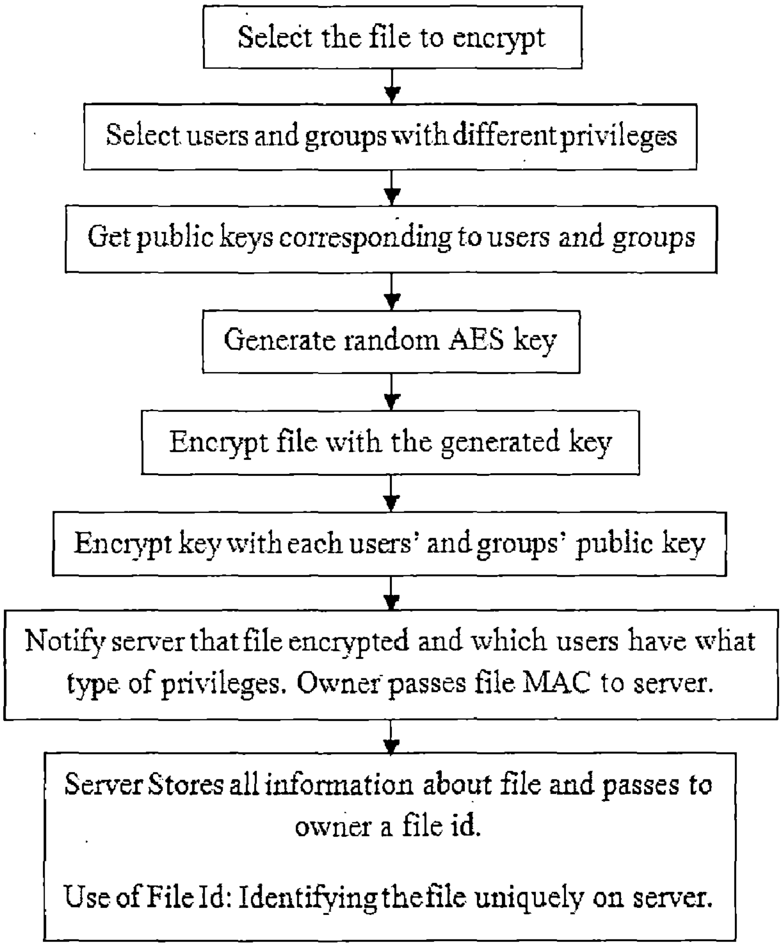

Select the file to encrypt
Select users and groups with different privileges
Get public keys corresponding to users and groups
Generate random AES key
Encrypt file with the generated key
Encrypt key with each users' and groups' public key
Notify server that file encrypted and which users have what type of privileges. Owner passes file MAC to server.
Server Stores all information about file and passes to owner a file id.
Use of File Id: Identifying the file uniquely on server.

Figure 4-3 Steps of Encryption

Step 1: After successful login, user clicks the browse button (present in front of Source File label in the UI Screen) and selects the file that he/she wants to store/upload into the cloud.

Step 2: After selecting the file, user enters the name of the users and groups (with whom owner wants to share its file with different access permissions) having read or write privilege and press 'Validate' button to verify whether the entered users and groups are valid or not. This validate request has been sent to the Internal Cloud Server, cloud service provider checks whether the entered users and groups are present in its database or not; if it is not, then the server makes LDAP call to the MS Exchange Server to check in the Active Directory whether they are valid or not. If they are not valid, then the server returns a message "Entered usernames / groups are invalid" otherwise "entered names are valid".

> *Note 1:* Multiple names must be separated by comma.
>
> *Note 2:* If owner wants to encrypt the file only for itself, then space for entering the names of read or write users or groups must be empty. (Refer snapshot)

Step 3: Once user selects a file to encrypt along with the users and groups with different privileges (read / write) with whom user (owner of the file) wants to share its' file; the user will get the public keys corresponding to selected users and groups from the internal cloud server.

Step 4: The application running at clients' side generates a random number using AES algorithm to encrypt the file (individual file has been encrypted by individual random number). The generated random number has been known as unique File Encryption Key (FEK). This is also referred as Symmetric Key. The message digest (also referred as message authentication code, MAC) of the content of the file has been computed and after that the file has been encrypted by session key.

Step 5: Symmetric key (random generated key) used for encrypting the original file encrypts with each users' and groups' public key (only the authorized ones who have access to the file) and the resulted encrypted keys have been appended to the header of the encrypted file with their usernames and group names. The file header contains the file id (generated by the server) , an integer value that represents the number of users and groups with whom the file has been shared, usernames & group names and their corresponding encrypted keys (key used for the decryption of the encrypted file).

Step 6: Once the file is encrypted, owner notifies the server and also which users and groups have what kind of privileges (read / write). DO also send the MAC of the content of the file to the Internal Cloud Server. The MAC and privileges has been stored by the server in its database that has been further used to check the confidentiality and integrity of the data present in the file. Then, the server returns a file id to the file owner that he/she adds it to the file header. The use of File Id is to identify the file uniquely on server.

4.2.2 Steps performed at Clients' (File Users') Side / Steps of Decryption

User logins into the cloud and loaded with its' private key. Then, the user has been notified by server to different groups to which users belong.

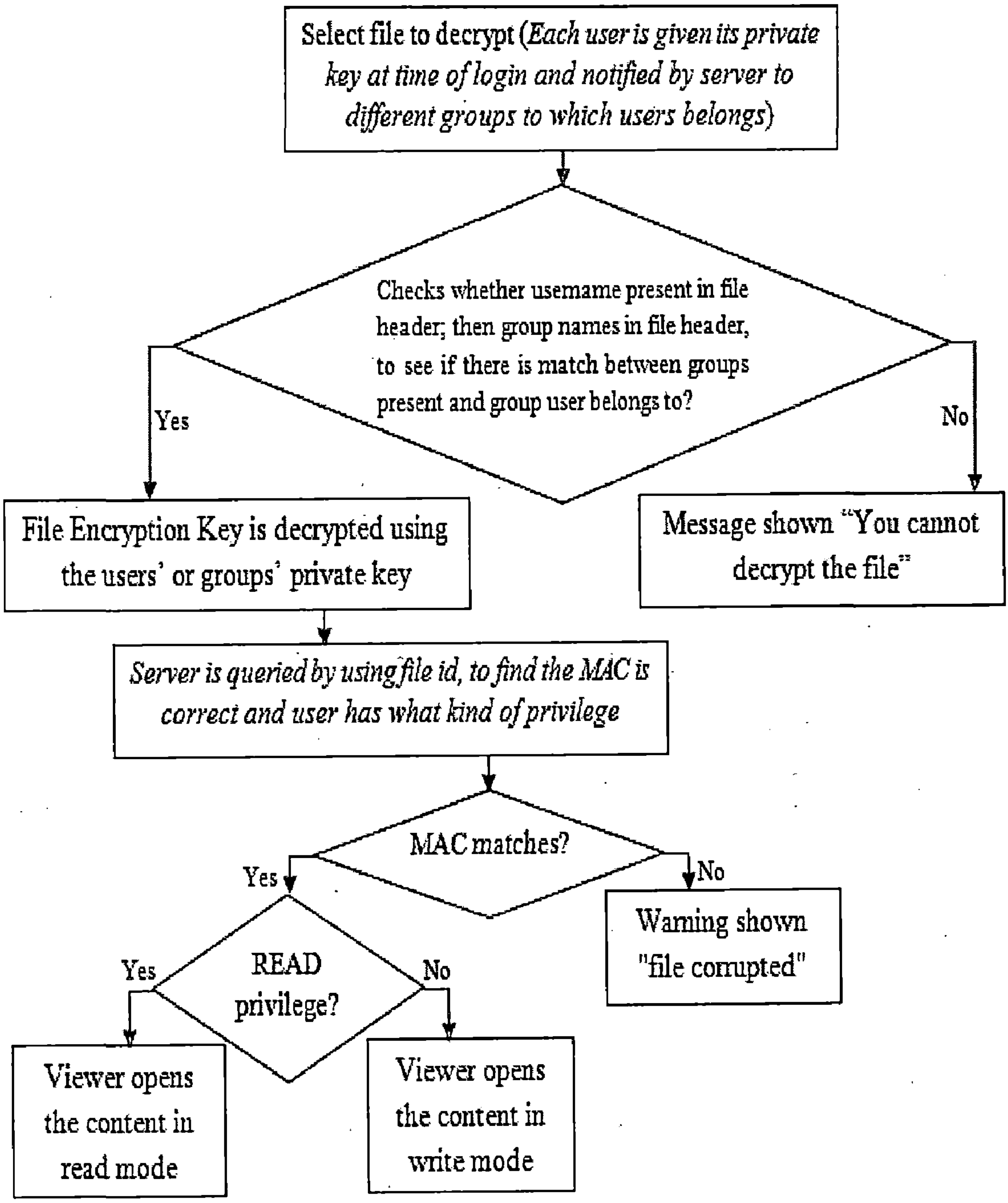

Select file to decrypt (Each user is given its private key at time of login and notified by server to different groups to which users belongs)
Checks whether username present in file header; then group names in file header, to see if there is match between groups present and group user belongs to?
Yes
No
File Encryption Key is decrypted using the users' or groups' private key
Message shown "You cannot decrypt the file"
Server is queried by using file id, to find the MAC is correct and user has what kind of privilege
MAC matches?
Yes
No
Warning shown "file corrupted"
READ privilege?
Yes
No
Viewer opens the content in read mode
Viewer opens the content in write mode

Figure - Steps of Decryption

Step 1: User selects a file to read or edit from the shared folder of its dropbox account. Now, the application running at clients' side tries to decrypt it. The running application first attempts to check whether the login username present in encrypted file header or not, then group names in file header, to see if there is a match between groups present and group user belongs to.

> ***Note:*** If login user is part of some group and there is a match between groups present and group user belongs to, then user will get the private key of that group at the time of its use and the key will automatically flush by the running application after its use.

Four cases can be possible:

- Both username and group to which user belongs present in file header.
- Only username is present in file header.
- Only group to which user belongs is present.
- Both are not a part of the metadata present in header.

Explanation of the above mentioned cases are given as below:

❖ *Both username and group to which user belongs present in file header*

a) User has read privilege and group has write privilege.
b) User has write privilege and group has read privilege.
c) Both have read privilege.
d) Both have write privilege.

According to (a), user has an access permission of read only and the same user is part of some group that has write access permission on the same file. Finally, the user has write access permission and the symmetric key (also known as File Encrypted Key, FEK) will be decrypted by using groups' private key.

Now in (b) it is given that user has write access permission and the same user is part of some group that has an access permission of read only on the same file. Finally, the user has write access permission and the symmetric key will be decrypted by using users' private key.

As mentioned in (c) & (d), both are having read or write privilege, hence ESK can be decrypted by either users' private key or groups' private key.

- *Only username is present in file header*

  If only username is present in the file header then FEK will be decrypted by users' private key.

- *Only group to which user belongs is present*

  If only group to which user belongs is present in the file header then FEK will be decrypted by groups' private key.

- *Both are not a part of the Meta data present in header*

  If both are not present in the file header then the application simply returns a message "You cannot decrypt the file".

Step 3: The symmetric key used for the encryption of the file has been decrypted by the application. Now the encrypted data present in the file will be decrypted by the symmetric key.

Step 4: As the application running at clients' side gets the content of the required file, now it generates the MAC of the content and then makes a query to server using file id (present in file header, generates by the server), to check whether the generated MAC and MAC stored in the servers' database for the given file id is same or not, if MAC doesn't match then the "file corrupted" warning will be shown. If MAC matches, then the server checks in its' database that the login user has what kind of privilege and returns it to client.

Step 5: Now, the application checks if user has read privilege then viewer opens the data in read mode otherwise in write mode. If the viewer opens in read mode then the login user will not be able to make changes (neither add nor delete the content) to the data. But in write mode, user can make changes (must have to press the save icon) and these changes will be reflect in future. The updated file has been again encrypted by the running application.

4.2.3 Steps at Internal Clouds' Side: (Key management, Access control, Auditing Support)

- When user log for first time on internal cloud, RSA key pair is generated, its private key passed, server checks to which all groups the given user belongs and tells to user all groups to which user has membership.
- When owner requests for public key, if user is present then its public key passed else ensured user exist in LDAP, if exists, user with random password and new RSA key is created and its public key passes.
- As mentioned above, when user logs server tells to user all groups to which user has membership (using LDAP), if the return group name / names is / are not present in the database then Group Keys are generated and new entries will be added in the DB.
- Access control server keeps track of which users have write permission on file and only users with write permissions are allowed to change the MAC for given file ID. Server passes MAC for file ID only if user has access to file.
- For supporting auditing each and every action at server is logged. All the keys are present on Internal Cloud Server so loss / hiding of data due to encryption can be prevented as server can decrypt any file using the appropriate keys.

Chapter 5
Results and Discussion

The implementation of the access control based data security application is done in java.

Table 5-1: Table showing the relationship between users and groups

User name	*Belongs to group*
User1, user2	Group1
User3, user4	Group2

Consider that users belong to the groups as mentioned in table 5-1. Now, the DO specifies the access permissions to the file mentioned as below.

File access:

Read users- user1, user3

Read Groups- group1

Write users- user2, user4

Write group- group2

Table 5-2: Table Showing the Users' Access Permission on File

Username	*User access permissions on file*	*Reason*
User1	Read	User1 and Group1 both have Read permission.
User2	Write	User2 has Write permission and Group1 has Read permission.
User3	Write	User3 has Read permission and Group2 has Write permission.
User4	Write	User4 and Group2 both have Write permission.
User5	No access	User5 has no right to access, also doesn't belong to any group.

The table 5-2 shows the users' access permission on file.

The table 5-3 briefly explains the observed test results in different scenarios.

Table 5-3 Test results observed in different scenario

Test Results	*Result Observed*	*Result Description*
When user with access privilege tries to access the file.	User has access to the file. Read user has no right to make changes (neither add nor delete the content) to the data. But write user can make changes and these changes will reflect in future.	User has corresponding decryption key (symmetric key, decrypted by user's private key only if user is authorized) to decrypt the content of the file.
When a read user tries to write the file using some other editor.	The file will be detected as corrupted when used by other users in future.	The updated file can be uploaded by user but user has no right to update the MAC stored into the internal cloud. Now, when file will be used by other users then MAC varies hence file will be resulted as corrupted.
When illegitimate user tries to access the file.	User has no right to access the file. The file will not be decrypted by the user.	User has no corresponding decryption key as he doesn't have the private key to decrypt the symmetric key.
When illegitimate user tries to create a new file.	User cannot upload the MAC corresponding to the file id on internal cloud.	User can upload the file on cloud using malicious CSP but he is not authenticated to the internal cloud as he doesn't have the active directory credentials.

5.1 Performance based on Encryption time

The time observed does not take into account the time required to fetch each public key as fetching of keys can be made parallel. So, it doesn't matter whether we are fetching public key for many users or public key for one group. In extreme scenario difference may be observed as when internal user is in heavy load the fetching of one user key will be faster than fetching each users public key. Key Size is fixed. AES is used for symmetric encryption. Symmetric Encryption Key Size is 8 bytes. RSA is used for asymmetric encryption. Symmetric key is encrypted to public key of size 1024 bits.

Table 5-4 Encryption time observed without using the concept of groups

File Size	Number of users	Time to encrypt (milli second)
5 KB	10	36
5 KB	100	39
5 KB	1000	89
10 KB	10	71
10 KB	100	75
10 KB	1000	118

Table 5-5 Encryption time observed with using the concept of groups

File Size	Number of Groups	Number of users in each group	Time to encrypt (milli second)
5KB	1	10	35
5KB	1	100	35
5KB	1	1000	35
10KB	1	10	69
10KB	1	100	69
10KB	1	1000	69

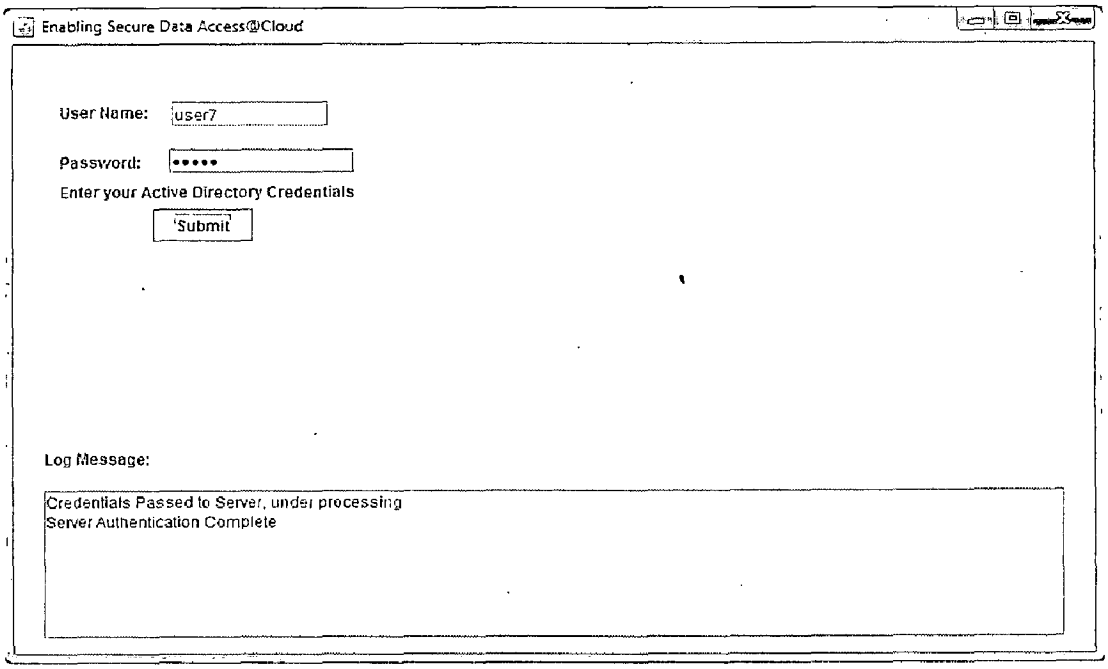
The table 5-4 and 5-5 shows the time taken to encrypt the file with / without the concept of groups. Hence, table 5-5 shows the reduction in time with same number of users but with the concept of groups (considering users belongs to same group).

5.2 Snapshots

The snapshots of each of the phase are shown as:

5.2.1 Snapshots of Login

Snapshot no. 1



User Name: user7

Password: •••••

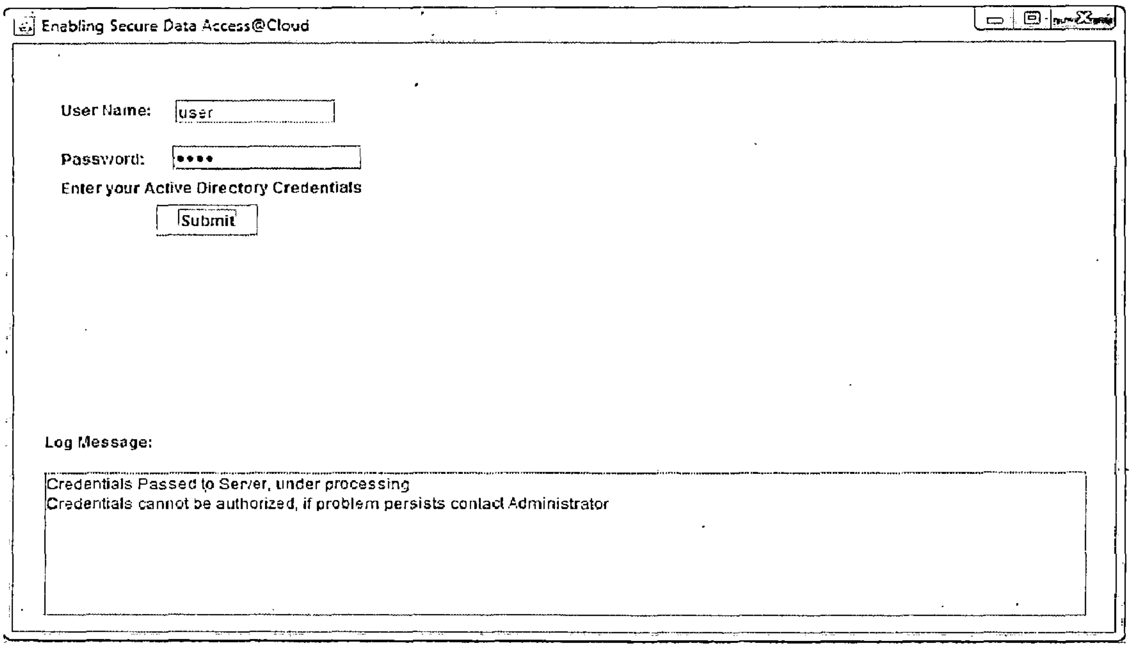
Enter your Active Directory Credentials

Log Message:

```
Credentials Passed to Server, under processing
Server Authentication Complete
```

Above snapshot shows the login user has been authenticated by the server.

Snapshot no. 2



User Name: user

Password: ••••

Enter your Active Directory Credentials

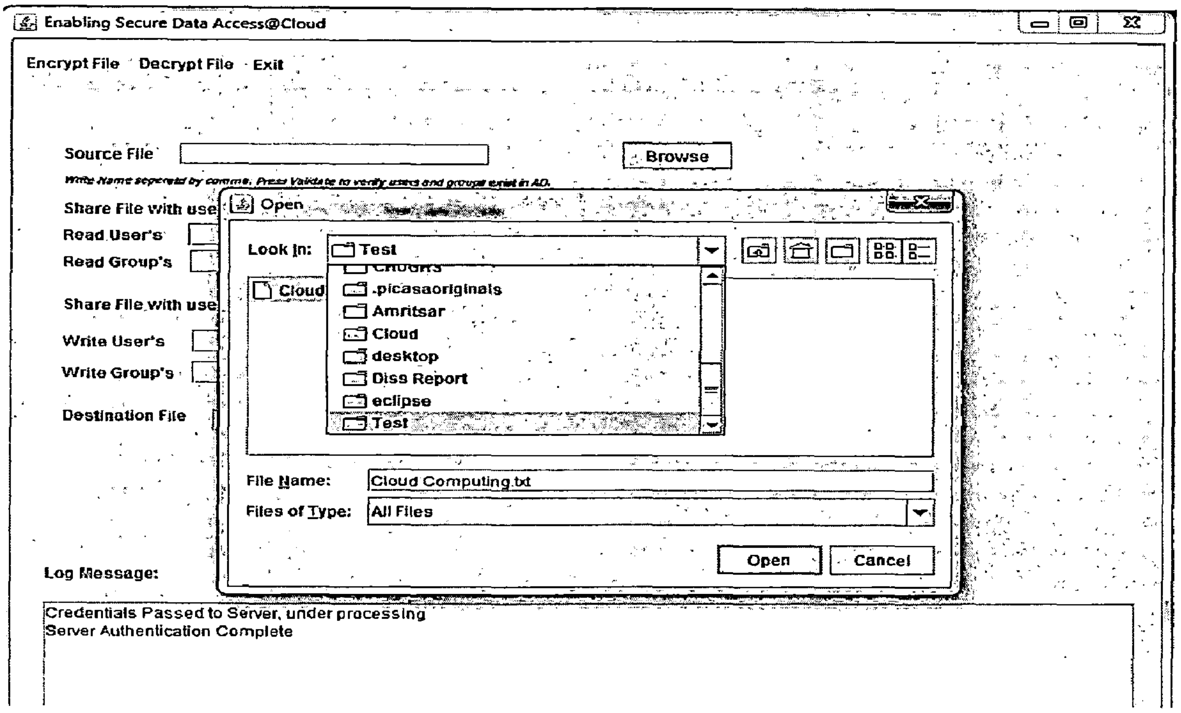
Log Message:

```
Credentials Passed to Server, under processing
Credentials cannot be authorized, if problem persists contact Administrator
```

Above snapshot shows that user credentials are not valid. Hence, login user is not authenticated.

5.2.2 Snapshots of Encryption

Snapshot no. 1



Look In: Test

File Name: Cloud Computing.txt

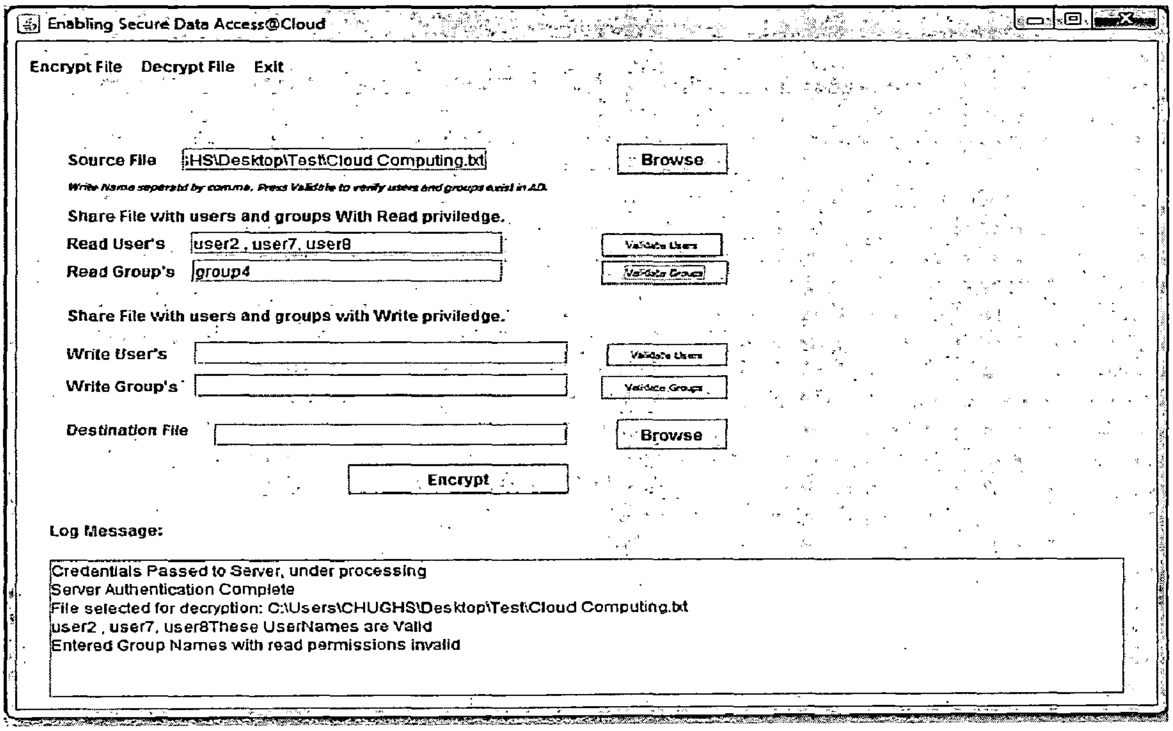
Files of Type: All Files

Write Name seperated by comma. Press Validate to verify users and groups exist in AD.

```
Credentials Passed to Server, under processing
Server Authentication Complete
```

After successful login, this screen pops up, now DO selects a file that is going to be encrypted and then stored into the cloud.

Snapshot no. 2



Source File: ;HS\Desktop\Test\Cloud Computing.txt

Write Name seperatd by comma. Press Validate to verify users and groups exist in AD.

Share File with users and groups With Read priviledge.

Read User's: user2 , user7, user8

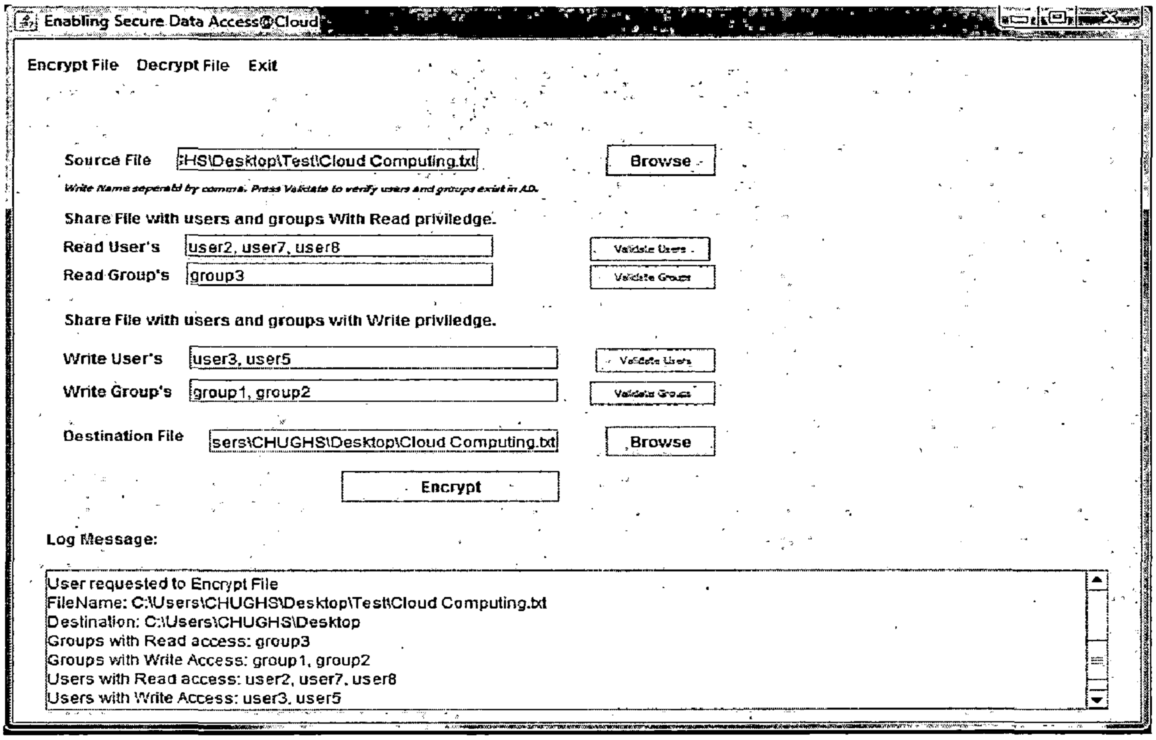
Read Group's: group4

Share File with users and groups with Write priviledge.

```
Credentials Passed to Server, under processing
Server Authentication Complete
File selected for decryption: C:\Users\CHUGHS\Desktop\Test\Cloud Computing.txt
user2 , user7, user8These UserNames are Valid
Entered Group Names with read permissions invalid
```

The entered usernames with read privilege are valid but the entered group is invalid that means it doesn't present in the Active Directory.

Snapshot no. 4



Source File: ;HS\Desktop\Test\Cloud Computing.txt

Write Name seperatd by comma. Press Validate to verify users and groups exist in AD.

Share File with users and groups With Read priviledge.

Read User's: user2, user7, user8

Read Group's: group3

Share File with users and groups with Write priviledge.

Write User's: user3, user5

Write Group's: group1, group2

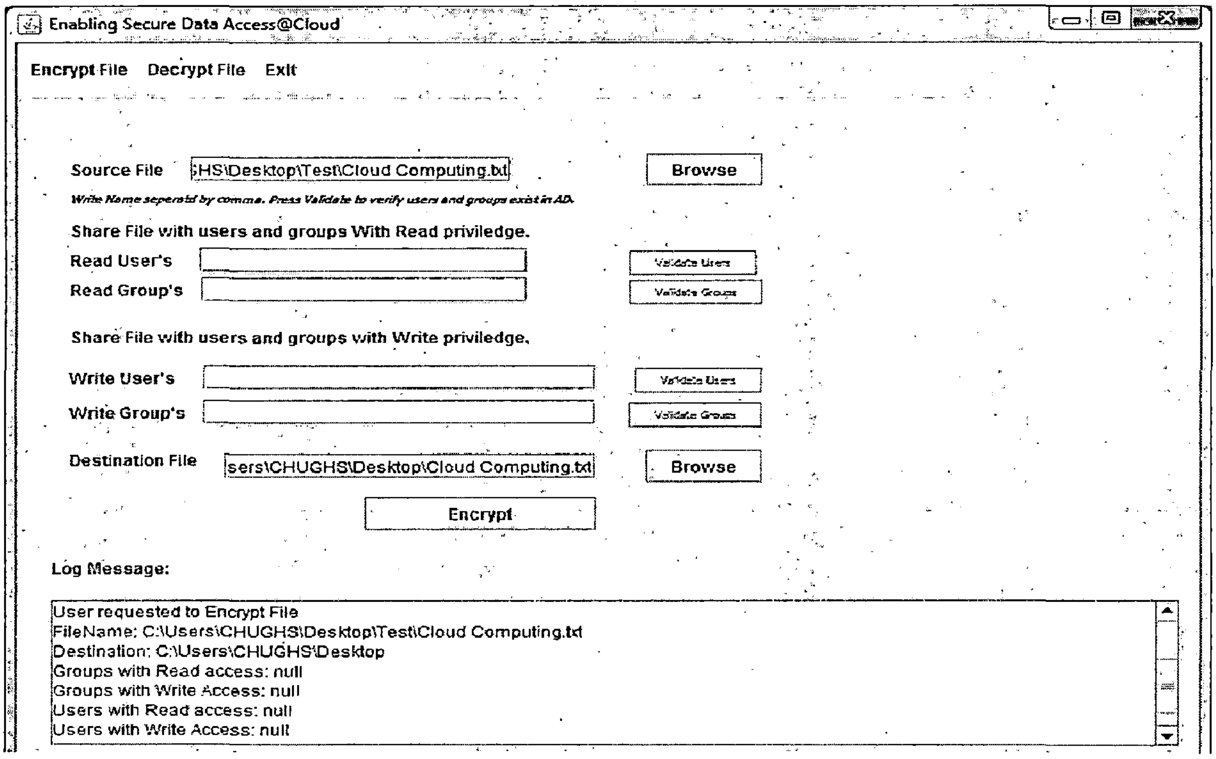
Destination File: sers\CHUGHS\Desktop\Cloud Computing.txt

Log Message:

```
User requested to Encrypt File
FileName: C:\Users\CHUGHS\Desktop\Test\Cloud Computing.txt
Destination: C:\Users\CHUGHS\Desktop
Groups with Read access: group3
Groups with Write Access: group1, group2
Users with Read access: user2, user7, user8
Users with Write Access: user3, user5
```

Now, the information (usernames and group names with access permissions) given in log message is going to be sent to the internal cloud server for storing it in the database.

Snapshot no. 5



Source File: ;HS\Desktop\Test\Cloud Computing.txt

Write Name seperatd by comma. Press Validate to verify users and groups exist in AD.

Share File with users and groups With Read priviledge.

Read User's:

Read Group's:

Share File with users and groups with Write priviledge.

Write User's:

Write Group's:

Destination File: sers\CHUGHS\Desktop\Cloud Computing.txt

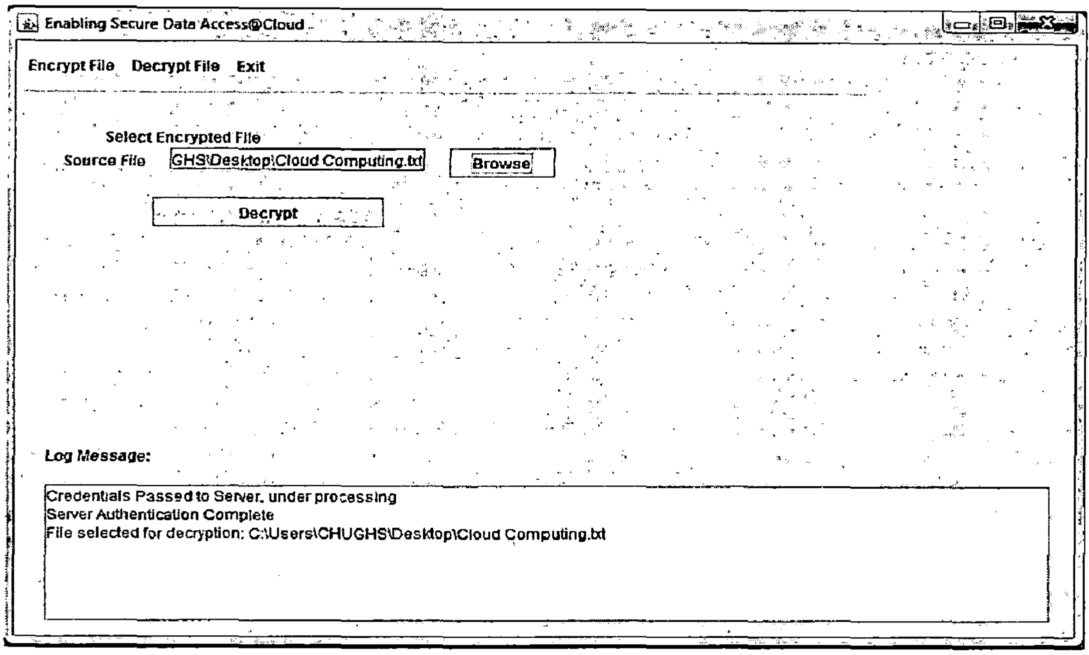
Log Message:

```
User requested to Encrypt File
FileName: C:\Users\CHUGHS\Desktop\Test\Cloud Computing.txt
Destination: C:\Users\CHUGHS\Desktop
Groups with Read access: null
Groups with Write Access: null
Users with Read access: null
Users with Write Access: null
```

If user wants to encrypt a file only for itself (not shared with others), then all fields must be empty.

5.2.3 Snapshots of Decryption

Snapshot no. 1

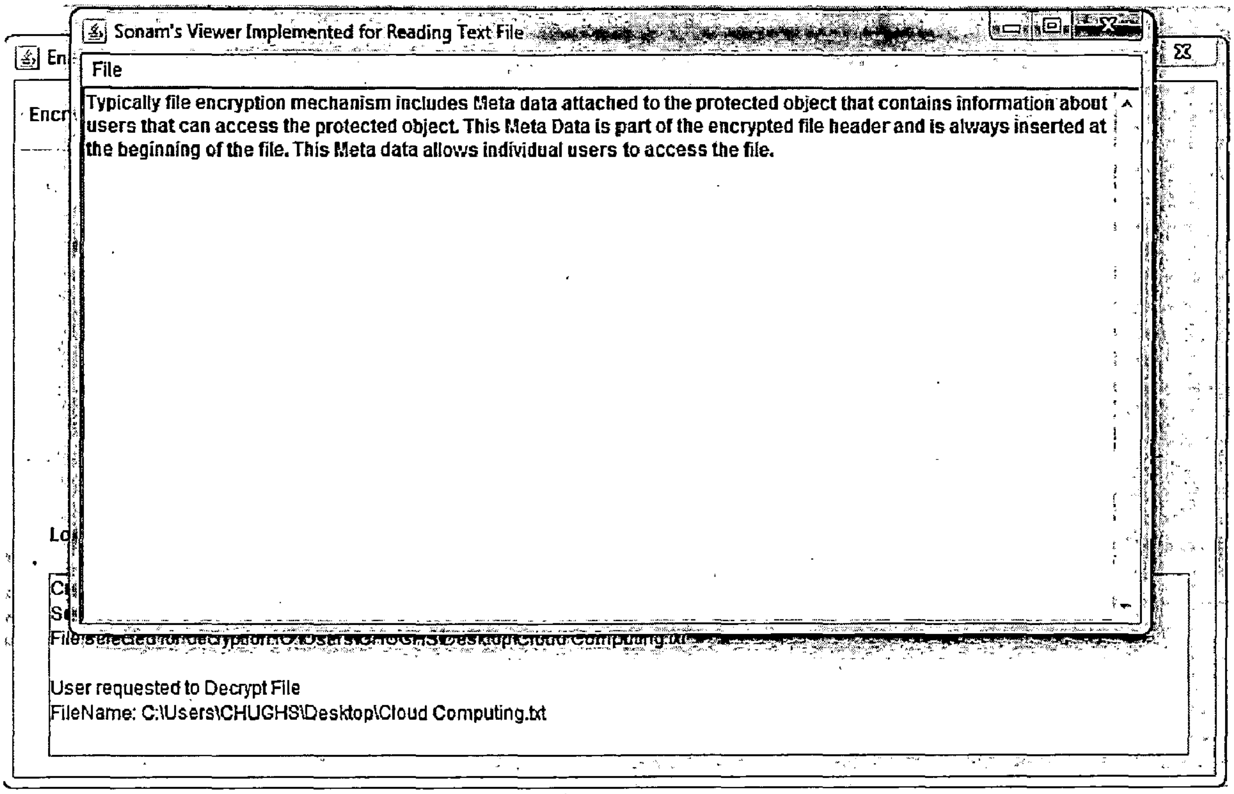


Source File GHS\Desktop\Cloud Computing.txt

```
Credentials Passed to Server. under processing
Server Authentication Complete
File selected for decryption: C:\Users\CHUGHS\Desktop\Cloud Computing.txt
```

It shows that encrypted file has been selected by the shared user for decryption. The text box contains the path of the selected file.

Snapshot no. 2

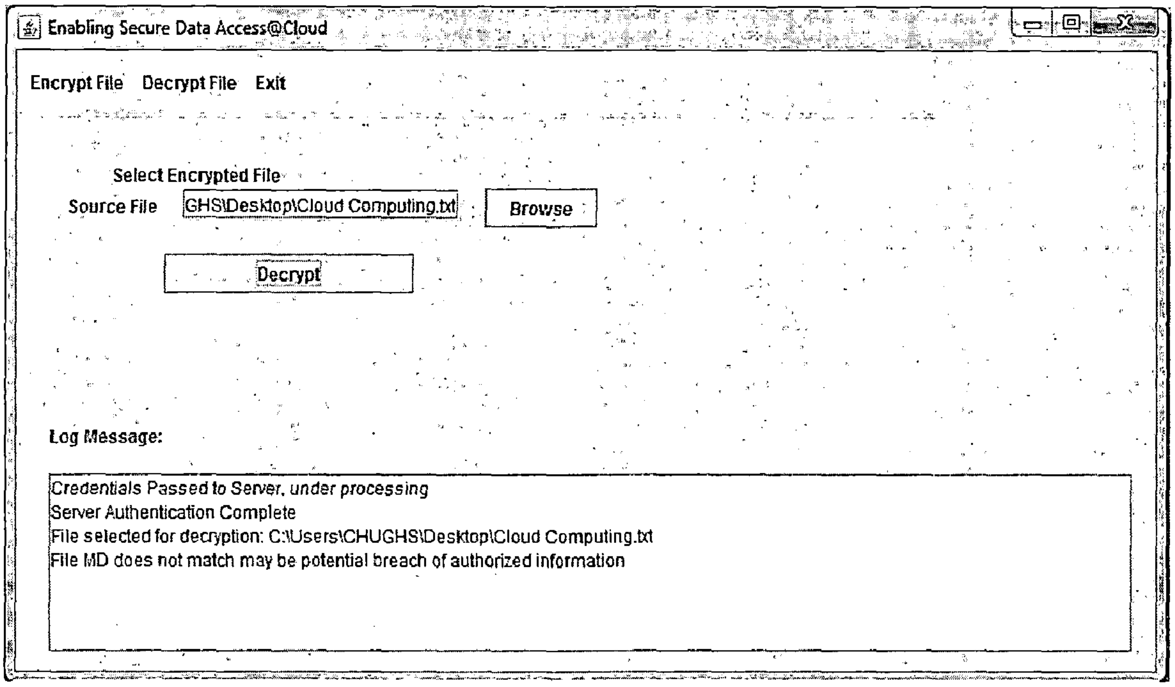


Typically file encryption mechanism includes Meta data attached to the protected object that contains information about users that can access the protected object. This Meta Data is part of the encrypted file header and is always inserted at the beginning of the file. This Meta data allows individual users to access the file.

```
User requested to Decrypt File
FileName: C:\Users\CHUGHS\Desktop\Cloud Computing.txt
```

Viewer showing decrypted file

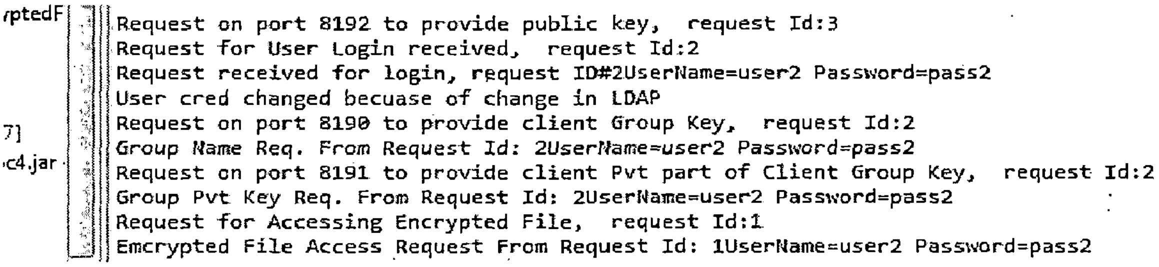
Snapshot no. 3



```
Credentials Passed to Server, under processing
Server Authentication Complete
File selected for decryption: C:\Users\CHUGHS\Desktop\Cloud Computing.txt
File MD does not match may be potential breach of authorized information
```

Unauthorized user tries to access the stored encrypted file.

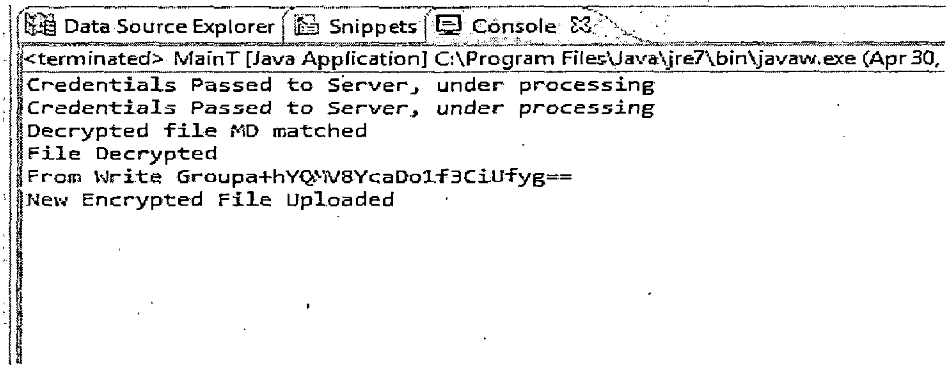
Snapshot no. 4



```
Request on port 8192 to provide public key,  request Id:3
Request for User Login received,  request Id:2
Request received for login, request ID#2UserName=user2 Password=pass2
User cred changed becuase of change in LDAP
Request on port 8190 to provide client Group Key,  request Id:2
Group Name Req. From Request Id: 2UserName=user2 Password=pass2
Request on port 8191 to provide client Pvt part of Client Group Key,  request Id:2
Group Pvt Key Req. From Request Id: 2UserName=user2 Password=pass2
Request for Accessing Encrypted File,  request Id:1
Emcrypted File Access Request From Request Id: 1UserName=user2 Password=pass2
```

Operations performed at Servers' side when shared user logins and access an encrypted file.

Snapshot no. 5



```
<terminated> MainT [Java Application] C:\Program Files\Java\jre7\bin\javaw.exe (Apr 30,
Credentials Passed to Server, under processing
Credentials Passed to Server, under processing
Decrypted file MD matched
File Decrypted
From Write Groupa+hYQ\V8YcaDo1f3CiUfyg==
New Encrypted File Uploaded
```

Clients' (shared user having write permission) console window, after changes MAC has been updated and new encrypted file uploaded.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

At present, there has been a lot of attention on cloud data security. This dissertation shows how authorized user can securely access the data stored by data owner. The combined approach of access control and cryptography is used to protect the data that now resides at cloud.

The conclusions drawn from the present work can be summarized as follows:

- A framework for organization / enterprise has been developed that has enabled Access Control based Data Security in cloud computing.
- In the proposed framework concept of group key has been introduced which is a step towards reducing the file metadata & it solves the problem of re-encrypting the file which arises in case group membership of user changes. But there is a lot of computation overhead to cloud servers hence some extensions can be used to reduce the overhead.
- The concept of groups reduces file size hence also reduces cost of uploading the encrypted file (which is the main feature of cloud computing).
- The proposed framework is extensible enough to support auditing of data on Cloud Service provider and it enables an organization capability to decrypt the encrypted file just in case a situation arises.

6.2 Suggestions for the Future Work

Some directions for further research work are as follows:

- In the existing implementation of the framework only text files have been encrypted. This limitation is since once the file has been decrypted it is necessary to show file in a viewer on which we have control programmatically just in case user wants to save the file, file need to be saved in encrypted manner. Hence

Viewer can be enhanced to decrypt file in different formats. I have implemented only for text file due to time constraint.

- In the existing implementation only the files have been encrypted similarly folders can also be encrypted where each file in a folder is encrypted with a symmetric key(FEK), file symmetric key is encrypted by one more symmetric key that is at folder level and folder level symmetric key is encrypted by users public key (ESK).

REFERENCES

[1] P. Mell and T. Grance, "The NIST definition of cloud computing," *National Institute of Standards and Technology*, vol. 53, no. 6, 2009.

[2] *Dropbox, Online at* https://www.dropbox.com/.

[3] *Amazon Web Services (AWS), Online at* http://aws.amazon.com.

[4] *iCloud, Online at* https://www.icloud.com/.

[5] *Nirvanix, Online at* http://www.nirvanix.com/.

[6] A. Fox and R. Griffith, "Above the clouds: A Berkeley view of cloud computing," *Dept. Electrical Eng. and Comput. Sciences, University of California, Berkeley, Tech. Rep. UCB/EECS*, vol. 28, Feb 2009.

[7] S. Kamara and K. Lauter, "Cryptographic cloud storage," in Proceedings of Financial Cryptography: Workshop on Real-Life Cryptographic Protocols and Standardization 2010, January 2010, pp. 136-149.

[8] B. Furht and A. Escalante, *Handbook of Cloud Computing*: Springer-Verlag New York Inc, 2010.

[9] Anthony T. Velte, Toby J. Velte, Robert Elsenpeter, "Cloud computing: a practical approach," pp. 3-34, McGraw-Hill Companies, 2010.

[10] I. Menken and G. Blokdijk, "Cloud Computing Virtualization Specialist Complete Certification Kit-Study Guide Book and Online Course," 2009.

[11] B. P. Rimal, E. Choi, and I. Lumb, "A taxonomy and survey of cloud computing systems," in *2009 Fifth International Joint Conference on INC, IMS and IDC*, 2009, pp. 44-51.

[12] T. Sridhar, "Cloud Computing—A Primer Part 1: Models and Technologies," *The Internet Protocol Journal*, Volume 12, No. 3, September 2009, pp. 1-25.

[13] S. Sanka, C. Hota, and M. Rajarajan, "Secure data access in cloud computing," in Internet Multimedia Services Architecture and Application(IMSAA), 2010 IEEE 4th International Conference on, 2010, pp. 1-6.

[14] G. Ateniese, K. Fu, M. Green, and S. Hohenberger, "Improved proxy re-encryption schemes with applications to secure distributed storage," ACM Transactions on Information and System Security (TISSEC), vol. 9, no. 1, 2006, pp. 1-30.

[15] G. Miklau and D. Suciu, "Controlling access to published data using cryptography," in Proc. of 29th VLDB, Germany, Sept 2003, pp. 898-909.

[16] S. Yu, C. Wang, K. Ren, and W. Lou, "Achieving secure, scalable, and fine-grained data access control in cloud computing," in Proc. of IEEE INFOCOM 2010, 2010, pp. 1-9.

[17] Y. Xiaojun and W. Qiaoyan, "A View about Cloud Data Security from Data Life Cycle," in Computational Intelligence and Software Engineering (CiSE), 2010 International Conference on, 2010, pp. 1-4.

[18] *Hibernate, Online at* http://www.hibernate.org/.

[19] *PostgreSQL, Online at* http://www.postgresql.org/.

[20] *Active Directory, Online at* http://en.wikipedia.org/wiki/Active_Directory

[21] Available:http://docs.oracle.com/javase/1.4.2/docs/api/javax/crypto/package-summary.html.

[22] http://docs.oracle.com/javase/1.4.2/docs/api/java/security/MessageDigest.html.

[23] Available:http://docs.oracle.com/javase/1.4.2/docs/api/javax/swing/package-frame.html.

[24] Available: http://docs.oracle.com/javase/1.4.2/docs/api/java/net/Socket.html

[25] Available: http://docs.oracle.com/javase/jndi/tutorial/ldap/security/ldap.html.

PUBLICATION

Sonam Chugh and Sateesh Kumar Peddoju, "Access Control Based Data Security in Cloud Computing", *International Journal of Engineering Research and Applications (IJERA)* (to appear in the issue of May 2012)