Data Security in Cloud Computing Based on Advanced Secret Sharing Key Management Scheme

Aastha Mishra

Department of Computer Science and Engineering
National Institute of Technology Rourkela
Rourkela-769 008, Odisha, India

Data Security in Cloud Computing Based on Advanced Secret Sharing Key Management System

Thesis submitted in partial fulfilment of the requirements for the degree of

Master of Technology

in

Computer Science and Engineering

(Specialization: Information Security)

by

Aastha Mishra

(Roll No. 212CS2110)

under the supervision of

Prof. P.M Khilar

Department of Computer Science and Engineering
National Institute of Technology Rourkela
Rourkela, Odisha, 769 008, India

June 2014

National Institute of Technology Rourkela
Department of Computer Science and Engineering
Rourkela-769 008, Odisha, India.

Certificate

This is to certify that the work in the thesis entitled ***Data Security in Cloud Computing Based on Advanced Secret Sharing Key Management Scheme*** by ***Aastha Mishra*** is a record of an original research work carried out by her under my supervision and guidance in partial fulfillment of the requirements for the award of the degree of Master of Technology with the specialization of Information Security in the department of Computer Science and Engineering, National Institute of Technology Rourkela. Neither this thesis nor any part of it has been submitted for any degree or academic award elsewhere.

Place: NIT Rourkela
Date: June 2, 2014

Dr. P.M Khilar
Asst. Prof.
CSE Department
NIT Rourkela, Odisha

Acknowledgment

I am grateful to numerous local and global peers who have contributed towards shaping this thesis. At the outset, I would like to express my sincere thanks to Prof. P.M Khilar for his advice during my thesis work. As my supervisor, he has constantly encouraged me to remain focused on achieving my goal. His observations and comments helped me to establish the overall direction of the research and to move forward with investigation in depth. He has helped me greatly and been a source of knowledge.

I am really thankful to my all friends and to everyone who has provided me with kind words, a welcome ear, new ideas, useful criticism, or their invaluable time, I am truly indebted.

I would also like to thank my parents and sister for standing besides me all the time and support me morally and ethically.

I must acknowledge the academic resources that I have got from NIT Rourkela. I would like to thank administrative and technical staff members of the Department who have been kind enough to advise and help in their respective roles.

Last, but not the least, I would like to dedicate this thesis to my family, for their love, patience, and understanding.

Aastha Mishra
Roll: 212CS2110

Author's Declaration

I hereby declare that all work contained in this report is my own work unless otherwise acknowledged. Also, all of my work has not been submitted for any academic degree. All sources of quoted information has been acknowledged by means of appropriate reference.

Aastha Mishra
Roll: 212CS2110
Department of Computer Science

Abstract

Cloud computing is a globalised concept and there are no borders within the cloud. Computers used to process and store user data can be located anywhere on the globe, depending on where the capacities that are required are available in the global computer networks used for cloud computing. Because of the attaractive features of cloud computing many organizations are using cloud storage for storing their critical information. The data can be stored remotely in the cloud by the users and can be accessed using thin clients as and when required.One of the major issue in cloud today is data security in cloud computing.Storage of data in the cloud can be risky because of use of Internet by cloud based services which means less control over the stored data. One of the major concern in cloud is how do we grab all the benefits of the cloud while maintaining security controls over the organizations assets.

Our aim is to propose a more reliable, decentralized light weight key management technique for cloud systems which provides more effecient data security and key management in cloud systems. Our proposed technique provides better security against byzantine failure, server colluding and data modification attacks.
Keywords: *Cloud security; key management; server colluding attacks; Byzantine failure;*

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Chapter 1

Introduction

1.1 Introduction

According to NIST [1] Cloud computing is a model for 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.

In cloud computing data and applications are maintained with the use of central remote server and internet and allows consumers to use the applications without installating and also with the help of internet cloud computing allows customers to access their personal files which are stored in some other computer. Yahoo email, Gmail, or Hotmail etc are examples of cloud computing. The email management software and the server are fully managed and controlled by the CSP Google, Yahoo etc and are all on the cloud (internet) .

The chapter 1 is organized as follows. In section 1.2 classifications of cloud computing is discussed. Section 1.3 discusses the architecture of cloud computing. In section 1.4 characteristics of cloud computing is discussed. Section 1.5 covers the security issues and risks in cloud computing. In section 1.6 the motivation of this work is discussed and section 1.7 covers the problem statement. In section 1.8 the outline of thesis is discussed.

1.2 Classifications of Cloud Computing

It can be classified in following two ways [1]:

1. Deployment Model
2. System Model

1.2.1 Deployment models

Based on the deployment the cloud can be of following types.Different types are clouds are described below:

1. **Public cloud:** The cloud vendor hosts the computing infrastructure at its own premises and the customer dont have any control on where the computing infrastructure is hosted. The computing infrastructure is open for use by general public or different organizations share this computing infrastructure.

2. **Private cloud:** Computing infrastructure is not shared between the organizations. It is dedicated to a particular organization and is therefore more secure but expensive than public clouds. The two types of private clouds are described below.
   Externally hosted private clouds are hosted by some third party that specializes in cloud infrastructure and are also exclusively used by one organization.
   On-premise private clouds are hosted by the enterprise itself and are more expensive as compared to externally hosted private clouds.

3. **Hybrid cloud:** The cloud infrastructure in hybrid cloud is a synthesis of two or more unique cloud infrastructures that are bounded together by some standardized technology that empowers application and data portablility.

4. **Community cloud:** The cloud infrastructure is solely utilized by consumers from organization that have related or imparted concerns. Community cloud may be overseen, managed and worked by a third party or one or a greater amount of organizations in the community or some consolidation of them.

1.2.2 Service models

Different service delievery models in Cloud Computing are shown in figure 1.1.

1. **Software as a Service(SaaS):**

   In SaaS model consumer has the proficience to utilize applications of supplier that are running on a cloud infrastructure. A program interface can be used to access the applications through various client devices. Operating systems, storage, servers, network or other underlying cloud infrastructure are not overseen by the consumer.

2. **Platform as a Service (PaaS):** Consumer is given the capability to deploy onto the cloud infrastructure created by the consumer or applications acquired that are created using services, libraries, programming languages and tools supported by the provider.The underlying cloud infrastructure including storage, operating systems, servers, network controlled by the consumer

3. **Infrastructure as a Service (IaaS):** Consumer has the capability to provision computing resources like networks, storage, processing etc where the consumer can deploy and run arbitrary software, which can include applications and operating systems. The underlying cloud base is not overseen or controlled by the customer however the consumer has control over deployed applications,storage and operating systems.

1.3 Cloud computing architecture

The architecture comprises of many loosely coupled cloud components. Cloud can be broadly classified into two parts front end and the back end [3]. Client part of the cloud computing system is referred to as front end which consists of the applications and interfaces that are needed for accessing cloud computing platforms, e.g., Web Browser. Back end alludes to cloud itself and comprises of every last one of resources that are obliged to give cloud computing services.

It comprises of virtual machines, huge data storage, virtual machines, security mechanism, services, servers ,deployment models and so forth. These ends are typically connected through a network, normally connected by means of Internet. Back end is responsible to provide protocols, traffic control and built-in security mechanism. The server employs certain protocols, known as middleware, that helps the devices that are connected to communicate and correspond with each other.

1.4 Characteristics of Cloud Computing

1. **Broad network access:** Various client platforms like laptops,tablets,mobile phones can be used to access these capabilities that are available over the network.

2. **On-demand self-service:** Without the human interaction with each service provider a consumer can provision computing capabilities automatically as and when required, for example server time and network storage.

3. **Location Independence Customer:** dont have any control or knowledge of where the resources are located exactly but may have some control at a higher level of abstraction Consumer can control on which country or state the data should be stored.

4. **Resource pooling:** Multiple consumers are served with the providers pooled computing resources using a model, with different virtual and physical resources dynamically assigned and reassigned depending on the demand of consumer.

1.5 Security Issues And Risks In Cloud Computing

Gartner in 2008 recognized seven security issues [4] that need to be tended to before organizations switch completely to the cloud computing model.

1. **Data location:** While storing data in cloud some clients might not know where their data is actually located.

2. **Regulatory compliance:** Customers can choose between providers that permit to be examined by third party organizations that check levels of security provided by cloud service providers.

3. **Data segregation:** Since the data in encrypted form from different organizations may be stored in the same place, so a system is required that separates data from different organizations and it should be provided by the cloud service provider.

4. **Long-term viability:** It alludes to the capability to withdraw an agreement and all information if the current supplier is bought out by another firm.

5. **Investigative support:** In case a customer suspects defective movement from the supplier, he might not have numerous legitimate ways seek after an investigation.

6. **Recovery:** Each supplier ought to have a disaster recovery convention to ensure client data is protected in case of a disaster also.

7. **Privileged user access:** Data transmitted from the customer through the Internet represents a certain level of risk, in view of issues of information possession; ventures ought to invest time getting to know their suppliers and their regulations however much as could be expected before allotting some trivial applications.

Risks In Cloud Computing

The six specific areas of cloud computing where substantial security attention is required is are as foolows

1. Security of data in transit.

2. Security of data at rest.

3. Cloud legal and regulatory issues.

4. Robust separation between data belonging to different customers.

5. Authentication of users/applications/processes.

6. Incedent response.

1.6 Motivation

The information and data that is stored on the Cloud is important to people with noxious intent so security is very important in cloud environment. A considerable measure of conceivably secure information and particular data is put away on Pcs, and this basic data is currently being put away and exchanged to Cloud. So understanding the security measures that the Cloud provider uses is very important. The principal thing that must be dealt with is the efforts to establish safety that the cloud supplier recently has set up. These efforts to establish safety that cloud service provider give change from supplier to supplier and around the different types of Clouds. Some of the important issues [6] are:

1. What are the encryption methods that providers are using?

2. How is the actual hardware where data wil be stored is protected?

3. Is the backup provided for data that is stored in cloud?

4. Are the firewalls set up?

5. In case of a community Cloud, how is the information from one company separated from other company?

Standard terms and conditions as characterized by the cloud suppliers may address these inquiries. The home client for the most part have exceptionally little negotiation room to talk about the terms and condition in their Cloud contract while a small business client generally have slightly more room to discuss terms and conditions and large have marginally more room to examine the terms of their

agreement with the supplier and will have the capacity to pose these questions throughout that time. By subscribing to the Cloud the control of data is given to some outside source. This separation between cloud user and the physical location of information stored in cloud makes a hindrance and in the meantime more space is made for an outsider to get to critical data. Therefore, currently security is one of the biggest challanges in cloud computing. However, to take the advantage of the benifits provided by the cloud it is required to transfer the direct control of users data in the cloud.

1.7 Problem statement

Objective to improve the key management and data security in cloud computing based on advanced secret sharing key management algorithm. Our proposed method helps to give better fault tolerance against Byzantine attacks,server colluding and data modification attacks.

1.8 Organisation of thesis

The thesis is organized in the following way chapter 2 describes the existing architecture and literature review that was done for this thesis.In chapter 3 the proposed model and how the methods is applied to make system more secure is discussed. In Chapter 4 the implementationand results are discussed. Finally chapter 5 concludes with the summary of work done.

1.9 Conclusion

In this chapter overview of cloud computing is given which includes types of clouds, characteristics of cloud, architecture of cloud, security and risk issues, motivation and objective of our work.

Chapter 2

Chapter 2

Literature Review

2.1 Introduction

One of the most challenging problems of cloud service solicitation is to persuade users to trust the security of cloud service and upload their sensitive data. Although cloud service providers can claim that their services are well-protected by elaborate encryption mechanisms, traditional cloud systems still cannot persuade the users that even if the cloud servers are compromised, the data are still securely protected [17]. Key management is the toughest part to manage in cryptosystems.In order to manage the encryption keys securely, enterprises need to employ encryption in their cloud environment, while maintaining secure off-site storage of their encryption keys.Keys should never be stored in the same place as encrypted data. The keys used for encrypting sensitive customer data should be managed effectively by periodic key rotation, and re-encryption of data with new keys. Employees should be not be given more access than what is needed to complete their tasks.
Morsy et al, 2010 investigated cloud computing problems from the cloud architecture [13], cloud offered characteristics, cloud stakeholders, and cloud service delivery models perspectives.

This chapter is organized as follows:Section 2.2 discusses the key management challanges in cloud. Section 2.3 and section 2.4 introduces the types of key management schemes and issues in cloud systems. In section 2.5 solution to these security issues is discussed. Section 2.6 and 2.7 discusses the need of secret shar-

ing schemes and different types of secret sharing schemes respectively. In section 2.8 required background concepts are discussed. Section 2.9 discusses the problems with existing secret sharing and section 2.10 concludes the chapter.

2.2 Cryptographic Key Management Challenges in the Cloud

One of the critical aspect of cloud computing is the secure management of the resources that are associated with cloud services. One of the main tasks of secure management is cryptographic operations. Hence, while self configurable resources, elastic capablities and ubiquitous computing isprovided by cloud services at a lower cost, they also entail performing several cryptographic operations for the following:

- To provide secure storage of data that is processed by those services.
- To provide secure interaction of the cloud consumer with various services.

The above functions [12] can increase the complexity of the key management system (KMS) required to support the cryptographic operations for these functions for the above because differences in control and ownership of underlying infrastructures on which the resources and KMS are located.

2.3 Types of key management scheme

Selective distribution of keys and encryption that is utilized for security of critical information is a essential system for restricting access to data. Data information is given to encryption algorithm and a few transformations are performed on it utilizing a cryptographic key. A figured content is produced in this procedure. There is no simple approach to recover the original message from the figured content other than by knowing the right decryption key [14].

There are a few diverse methodologies to group key management and these could be partitioned into three fundamental classes:

1. **Centralized group key management protocols:** In this protocol single entity controls the whole group, hence this protocol minimizes the bandwidth

utilization computational power, storage requirements, on both server and client sides.

2. **Decentralized architectures:** In decentralized architectures a large group is separated into small subgroups and these subgroups are overseen separate subgroup administrators which minimizes the issue of concentrating the work load in a solitary spot.

3. **Distributed key management protocols:** In this architecture the key management task is performed by the members themselves and no explicit KDC is required[8].

2.4 Issues with key management in cloud Systems

Key management is the toughest part to manage in cryptosystems. Some of the issues [15] with encryption key management in the cloud.

- In the cloud platform, there is always a possibility of insider attack. Keys can be accessed or stolen by employees without the knowledge of end users.
- The keys for all accounts need to be managed properly. The challenge is to index proper accounts with their respective keys, quickly and effectively.
- Another issue with key management is availability. If a system goes offline, then how the keys will be accessed? There needs to be key cache in order to retrieve keys, even in the event that a system goes offline.
- Byzantine failure [17] is very common fault in cloud servers . A storage server can fail in arbitrary ways when this kind of fault occurs.
- Server colluding and data modification attacks are also very common in clouds in which the storage servers can be compromised by the adversary, as a result of which files can be modified .

2.5 Solution to the security issues

- In order to manage the encryption keys securely, enterprises need to employ encryption in their cloud environment, while maintaining secure off-site storage of their encryption keys.
- Encryption keys should never be stored in the same place as encrypted data. The keys used for encrypting sensitive customer data should be managed effectively by periodic key rotation, and re-encryption of data with new keys.
- Employees should be not be given more access than what is needed to complete their tasks.

Byzantine failure is very common fault in cloud servers,in which a storage server can fail in arbitrary ways. On occurance of a byzantine failure system responds in an unpredictable way[19]. At the point when a Byzantine failure has happened, the framework may react in any erratic way, unless it is intended to have Byzantine fault tolerance. The cloud is also inclined to data modification and server colluding attack in which the storage servers can be compromised by the adversary, as a result of which dat files can be modified as long as they are internally consistent. For providing secure storage of data in cloud storage server , the data should be encrypted.

2.6 Need of Secret Sharing schemes

In 1968, Liu [8] considered the following problem: Eleven scientists are working on a secret project. They wish to lock up the documents in a cabinet so that the cabinet can be opened, if and only if, six or more of the scientists are present. What is the smallest number of locks needed? What is th smallest number of keys to the locks each scientist must carry ?

If five scientists are considered together and there is a specific lock which they cannot open. If we consider a particular scientist, he must have the keys of those

locks which cannot be opened by any five scientists from among the other ten scientists. Five scientists among 11 scientists can be selected in 11C5=462 ways, and among ten scientists, five scientists can be selected in 10C5 = 252 ways. So, the minimal solution requires 462 locks and 252 keys per 14 scientist. This is clearly impractical, and if number of scientists increases it can become exponentially worse.

2.7 Secret Sharing Schemes

Different secret sharing schemes are discussed below:

- **Simple secret sharing**

  1. **Additive Secret Sharing**

     In additive secret sharing for a given secret SϵF,n 1 random integers R =r_1, r_2..... r_{n-1} are selected by the dealer uniformly from F. X then computes

     $S_n = S - \Sigma_{i=0}^{n-1} r_i mod F$

     X sends the share $s_i = r_i$ to each player P_j $1 \leq j \leq n-1$,and the share s_n is sent to P_m The secret S reconstruction is trivial and it can be generated simply by adding all the shares together.

     $S = \Sigma_{i=0}^{n} s_i mod F$ In this scheme contribution of shares from all the participants is required in order to reconstruct the secret. The original share cannot be recovered in case if one or more of the participants are missing, A scheme in which contribution of all the participants is required is known as a perfect secret sharing scheme.

  2. **Threshold secret sharing scheme**

     In 1979 Shamir [7] and Blakley [16] introduced the concept in order to make the message more secure. In this scheme, the message MG is divided into n pieces MG_1;MG_2;MG_3; : : : ;MG_n, with or witout ransformation of the message, in such a way that, for a specifed k,

($2 \leq$ k $\leq$ n), 1. knowledge of any k or more pieces MG_i makes MG computable; 2. knowledge of any k -1 or fewer MG_i pieces leaves MG completely undetermined.

This scheme is called a (k; n)-threshold scheme. The parameter k in (k,n) is called the threshold value.

3. **The Asmuth and Bloom Secret Sharing Scheme** Asmuth and Bloom proposed a scheme[11] in which modular arithmetic is used to share the secret between parties and Chinese remainder theorem isused for the reconstruction of the original secret. Keys are also known as shadows.

2.8 Background concepts

2.8.1 Lagrange polynomial

For polynomial interpolation lagrange polynomials are used in numerical analysis. When a set od distinct points x_j and numbers y_j are given, the Lagrange polynomial is the polynomial of the least degree that at each point x_j assumes the corresponding value y_j

When a set of p+1 data points are given

(x_0, y_0),...,(x_j, y_j),...,(x_p, y_p) where no two x_j are the same, the interpolation polynomial in the Lagrange form is a linear combination

L(x) := $\sum_{j=0}^{p} y_j \ell_j(x)$ of Lagrange basis polynomials

$\ell_j(x) := \prod_{\substack{0 \leq m \leq p \\ m \neq j}} \frac{x - x_m}{x_j - x_m} = \frac{(x-x_0)}{(x_j-x_0)} \cdots \frac{(x-x_{j-1})}{(x_j-x_{j-1})} \frac{(x-x_{j+1})}{(x_j-x_{j+1})} \cdots \frac{(x-x_p)}{(x_j-x_p)}$,

where 0$\leq$j$\leq$ k.

2.8.2 Shamir Secret Sharing Scheme

Shamir's threshold sharing scheme is based on theidea that in order to define a line two points are sufficient, in order to define a parabola three points are suffcient, 4 points to define a cubic curve and so forth.So in order to define a polynomial of degree k-1, k points are sufficient.

Suppose we want to use a (k,n) threshold scheme to share our secret S, without loss of generality assumed to be an element in a finite field F of size P. Choose at random k-1 positive integers $a_1, \cdots, a_{k-1}$ with a_i, and let a_0=S. Build the polynomial f(x)=$a_0 + a_1x + a_2x^2 + a_3x^3 + \cdots + a_{k-1}x^{k-1}$. Let us construct any n points out of it, for instance set $i = 1, \cdots, n$ to retrieve (i,f(i)). Every participant is given a point (an integer input to the polynomial, and the corresponding integer output). Given any subset of k of these pairs, we can find the coefficients of the polynomial using interpolation. The secret is the constant term a_0.

2.9 Problems with existing secret sharing scheme

The concept of secret sharing scheme was developed by Shamir [7] and Blakley [16] in 1979 in order to keep the secret efficiently and safely. Lagrange interpolating polynomial is the basis of shamir secret sharing , while blakely secret sharing is based on the Linear projective geometry. Some of the drawbacks in both these secret sharing schemes [7] and [16] are as follows: 1. A fake shadow may be distributed to a certain participant by a dishonest dealer and then the true secret would never be obtained by that participant.

2. A fake share may be provided by a malicious participant to other participants, and so the troe secret can only be reconstructed by the malicious participant.

2.10 Conclusion

If a single KDC is used it may become a point of failure. In addition to it because of the large number of users it becomes difficult to maintain a single key distribution center .Therefore in in this proposed work we try to emphasize on decentralized approach in clouds while distributing secret keys and attribute to users.

Chapter 3

Chapter 3

Proposed Technique

3.1 Introduction

This chapter is organized as follows: Section 3.2 introduces the main entities in the proposed model. Section 3.3 describes the cloud manager and the submodules in it. In section 3.4 the share renewal phase is discussed. Section 3.5 describes the voting technique used to ensure integrity of the shares. In section 3.6 the working of proposed model is described in detail. In section 3.7 the advantages of proposed system over existing systems is described. Section 3.8 concludes the chapter.

3.2 Main entities in proposed model

The main entities in the proposed algorithm are cloud users, cloud storage server, cloud manager, keysplitter servers, share holder servers, security servers, log editor which are defined in detail as follows:

1. **User:** The user can create, update and delete his/her profile, store and retrieve the data.

File_Info	UsersShare(xu)	UsersShare(f(xu))

2. **Cloud Storage Server:** It is a model of data storage on virtualized storage pools or servers located remotely. Cloud storage can be used by users to store their data. Users can buy storage capacity from the cloud hosting companies. The main responsibilities of cloud storage server are storing the encrypted

document, storing the splitted encryption key values for the purpose of key management .

3. **Key Management Server:** Key splitter server splits the encryption keys into different shares and store the splitted keys in different share holder servers.

CloudUser_Id	File_Info	ShareHldrServer_Id

4. **Share Holder Server:** These servers stores the shares for the different keys for different users. Share holders can be of two types. Primary share holder directly receive the shares from the cloud manager. Secondary share holders are the share holders at the leaf level and these share holders receive their shares through primary share holders.

CloudUser_Id	File_Info	xi	f(xi)

5. **Log editor:**

   It checks the share holder servers timely to see if the shares are getting modified.

ShareHldServer_Id	CloudUser_Id	Value(t-60)	Value(t)	Value_Status

6. **Security server:** It has the encryption decryption algorithm.

   Encryption process

   Step 1- Split the letter of modified plaintext.

   Step 2- Assign the position(i) of the letter.

   Step 3- Generate the ASCII value of plaintext letter.

   Step 4- E=(p+k+i)

   p-plaintext, k-shared key, i-position

   Step 5- Generate the ASCII character of the corresponding decimal value in the result from the above given formula.This would be the cipher text.

   Decryption process

   Step 1- Generate the ASCII value of the cipher text character.

Step 2- Same encryption key is used.

Step 3- Assign the position i of the cipher text.

Step 4- D=((c-k-i)+256)

p-plaintext, k-shared key, i-position.

Step 5 Generate the ASCII character of the corresponding decimal value in the result from the above given formula.This would be the original plain text.

Log editor checks in period of 60 sec if any share gets updated. If no key updates are performed the primary share holders are used in decryption process. If a primary share holder gets updated other primary share holders are also checked. If more than half are unchanged the unchanged values are used. Table 3.1 shows the actions that should be performed for different values of primary and secondary share holders.

Table 3.1: Actions to be performed for different values of primary and secondary share holders.

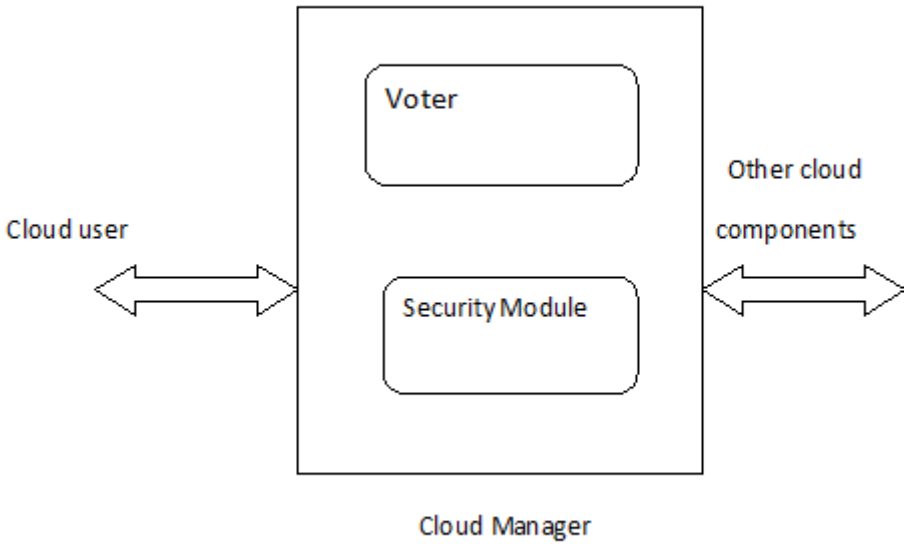
PrimaryShareHolder	SecondaryShareHolder	Action
Unchanged	Unchanged	Trusted
Changed	Changed	Use another share holder
Unchanged	Changed	Trusted
Changed	Unchanged	Regenerate primary from secondary

3.3 Cloud Manager

It consistes of a voter module and a security server module. Security server has the encryption decryption algorithm and the voting module in cloud manager performs the voting to check whether the share key holder is authentic or not. System reliability is increased by using the voting technique. It is assumed that the communication channel between the client and cloud manager is secure. Figure 3.1 shows the different modules in the cloud manager.

The proposed technique is suggested for the cloud systems using symmetric encryption.

Figure 3.1: Cloud manager



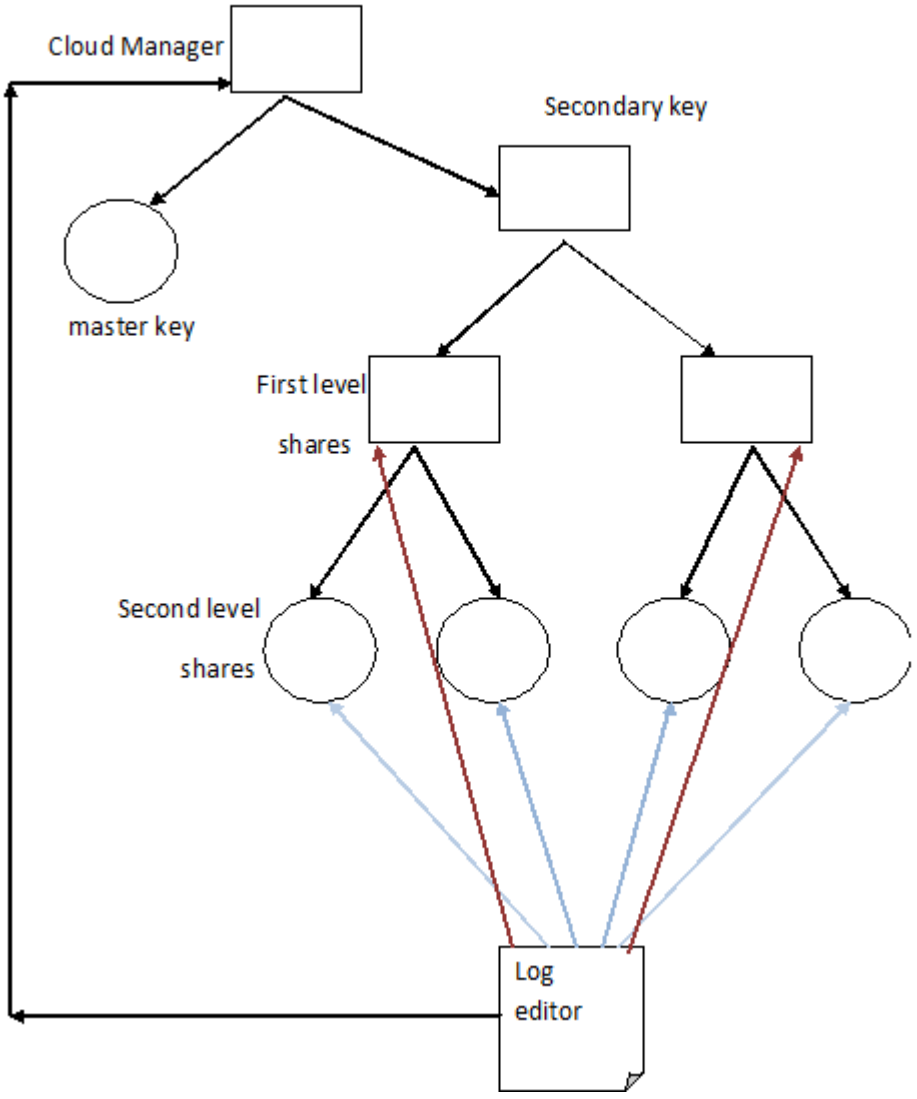
3.4 Share Renewal Phase

The keys are assumed to be stored in a hierarchical way. The secondary key manager distributes the shares to the primary key holders. Primary key holders in turn distribute the secret to the secondary key holders. The shared keys will have a crypto_period. When this period is about to expire all the shares will be renewed. The values of shares that the share holders are having should also be monitored from time to time since there may be a possibility of some of the shares being modified by the attackers. Figure 3.2 shows how the primary and secondary share holders are monitored by the log editor.

3.5 Overview of voting technique

Security and privacy of users data is preserved in the proposed technique by the replication of key share among several clouds, bu the use of the secret sharing approach, and using a voting method to check the integrity of shares. Figure 3.3 shows how the voting is performed by the voter module in cloud manager.

Figure 3.2: Proposed key management scheme in cloud

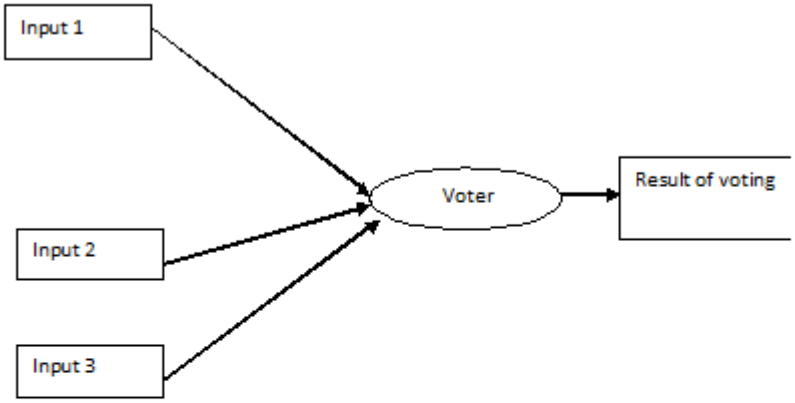


3.6 Working of proposed system

3.6.1 File Upload

When the cloud user wants to submit a file to a cloud first the file is forwarded to the cloud manager. Security module in cloud manager generates the key and encrypts the file using the encryption algorithm as shown and then forwards the key to key management module. Encrypted file is forwarded to the cloud data storage center. Key management module divides the key into number of shares. Sends a master key to the cloud user and distributes all the remaining keys to the

Figure 3.3: Voter in the cloud manager module



ShareHolderServers. All the primary share holders and secondary share holders are monitored from time to time to ensure that their values arenot modified by attacker.

Figure 3.4 shows the data communication in file upload process. The file upload consists of following communications:

1 User requests to upload a file in cloud system.

2(a) Encrypts the file. Forwards the shares to the secondary key manager.

2(b) Forwards the master key to the cloud user.

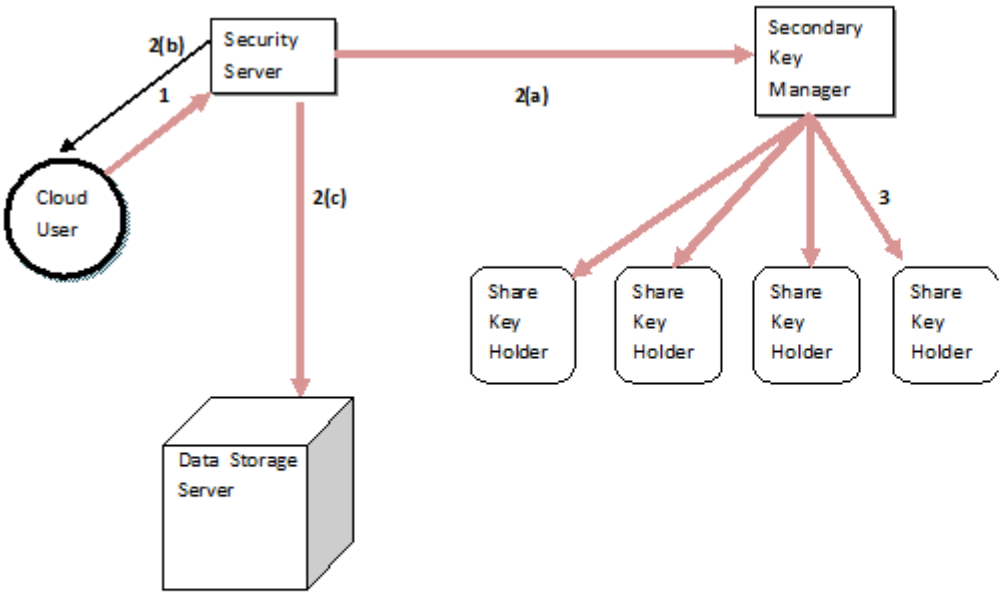
2(c) Forwads the encrypted file to the data storage server.

3 Key manager distributes the shares among the share key holders.

3.6.2 File Download

When the cloud user wants to download a file that is stored in cloud file name and shared master key are entered by cloud user. Download request is forwarded to key management server. Key management server requests all the ShareHolderServers

Figure 3.4: Communication in file Upload



to forward their part of keys that correspond to the file name required to it.Key management server combines all the shares to generate the 2nd level keys and forwards the key to the security server Security server combines the master key with other secondary key to generate the main key. The file is decrypted and is sent to the cloud user.

Figure 3.5 shows the process of file downloading. It consists of the following data communication:

1(a) Request to the security server for file downloading.

1(b) Request to cloud user to provide master key that was given in the encryption process.

1(c) Master Key provided by cloud user.

2(a) Request to data storage server to send the encrypted file as requested by user.

2(b) Data storage server sends encrypted file to cloud storage server.

3(a) Security server enquires the secondary key from the key manager.

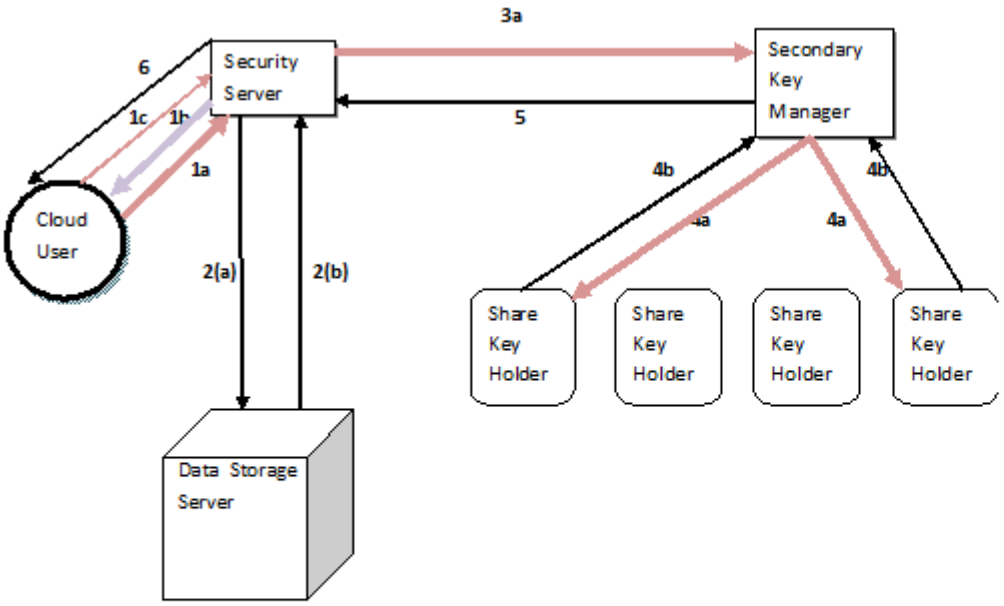


Figure 3.5: Communication in file download

4(a) Key Manager requests the share from the different share holders.

4(b) Share holder servers forward their shares corresponding to the user to the key manager.

5 Key manager combines all the shares and sends it to security server.

6 Security server combines the master key provided by the user and the secondary key provided by key management server and generates the actual key. Then,it decrypts the user file stored in cloud and send it to the user.

3.7 Advantages of the proposed technique over existing techniques

1. Existing techniques are centralized in nature . We try to provide to provide a distributed approach for key management.

2. Our technique provides fault tolerance to byzantine attacks, data modification and server colluding attacks.

3. Reliabily of the system is increased by using the voting technique to ensure that the shares does not get modified by the attacker.

4. After a pre decided crypto_time, the shares are renewed in order to ensure the security of user data if in case some of the shares get compromised.

3.8 Conclusion

In this chapter we have proposed a system which removes the drawbacks of some of the existing cloud systems.New modules are added to the existing system to improve the reliability and security of the existing cloud systems that use symmetric key encryption tecniques to ensure data security.

Chapter 4

Chapter 4

Implementation and results

4.1 Simulation Environment

The implementation is done using the following tool and techniques:

- CloudSim
- Amazon Web Services
- Jswings

Cloud Sim

- CloudSim is a web app that runs in avirtual machineon theAmazon Web Services (AWS)cloud.
- It allows users to launch, terminate and monitor virtual machines in the AWS cloud.
- Different configurations can be launched, depending on the requirements, and available machines on the cloud.
- Each CloudSim configuration maps to a constellation, which are collections of multiple virtual machines running together.

4.2 Results

Proposed technique is implemented with different file size ranges from 100KB to 50MB and we try to find out performance comparison between a existing technique

and the proposed technique.The key provided from AWS is the token that allow CloudSim to access AWS on behalf of the AWS user. Figure 4.1 and Figure 4.2 shows the snapshots of how the key is divided into several shares and is again combined to generate the oroginal key. For our implementation 128 bit key is used and n=128.

Figure 4.1: Snapshot of key generation

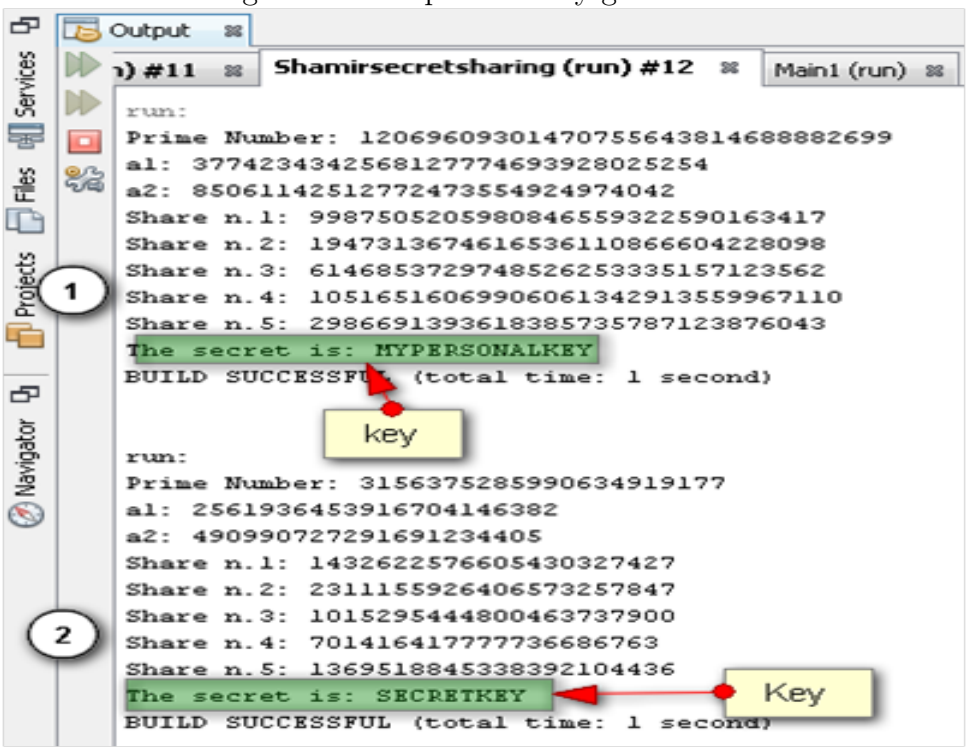


Table 4.1 and figure 4.3 represents the time taken to split a 128 bit key into 10 pieces for different threshold values.

Table 4.1: Key dispersal for a key into 10 shares

Threshold(k)	Time(ms)
5	4.56
6	4.08
7	3.39
8	2.50
9	3.41
10	2.26

Table 4.2 and figure 4.4 represents the time taken to combine the shares for different threshold values.

Figure 4.2: Snapshot of key splitting

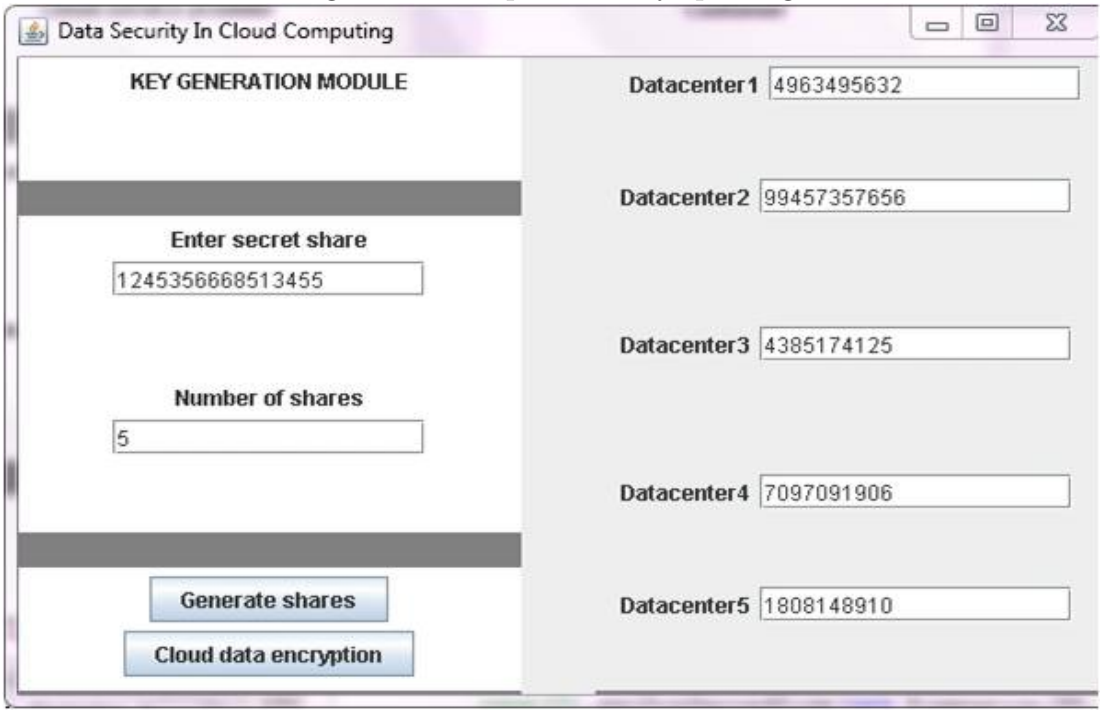
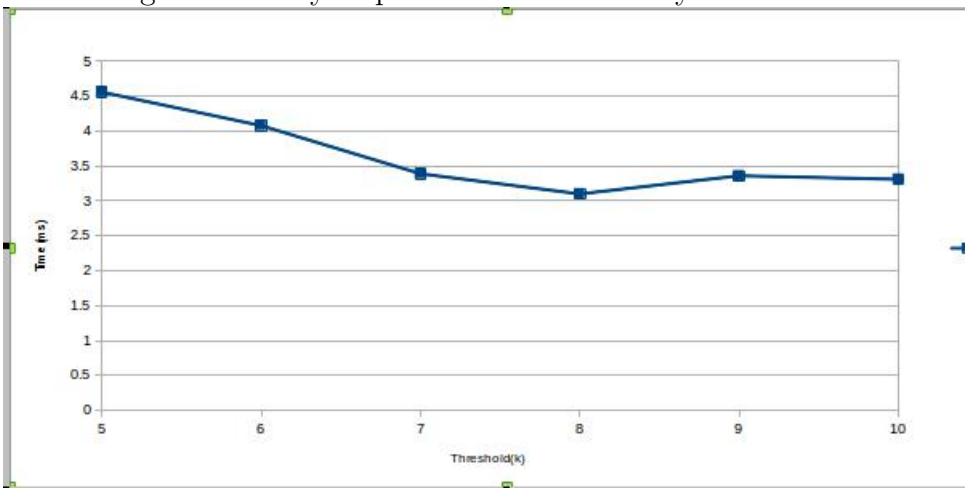


Figure 4.3: Key dispersal for a 128 bit key with n=10



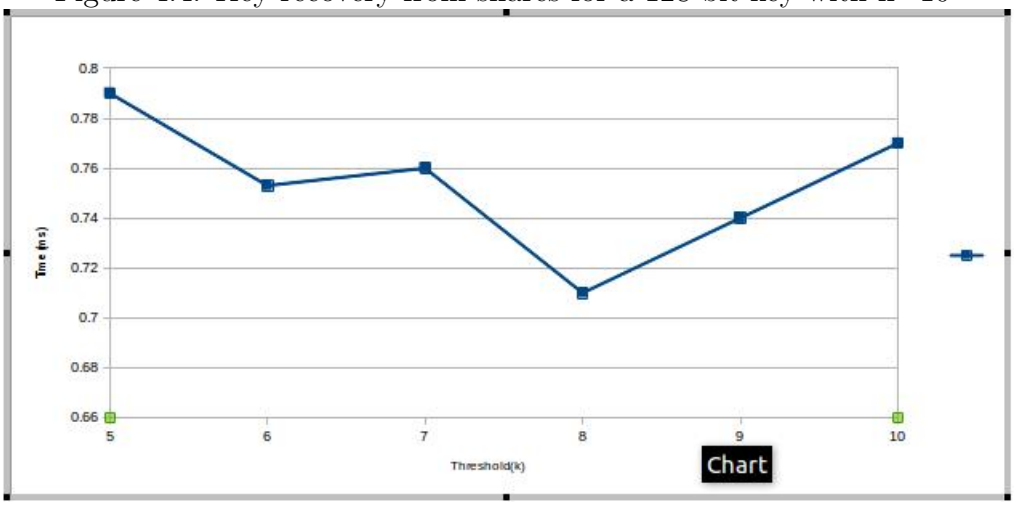
File Upload Time-It includes the time to encrypt the file as requested by the client.It is the time between the points when user requests the cloud system to upload the file and the time when the tasks of encryption and generating key shares actually finishes and the encrypted file is actually stored in the cloud data storage. Figure 4.5 shows the time taken for file uploading for different file sizes.

File download time- It includes time to collect the shares, generate the

Table 4.2: Key recovery time from shares

Threshold(k)	Time(ms)
5	.79
6	.75
7	.76
8	.073
9	.74
10	.77

Figure 4.4: Key recovery from shares for a 128 bit key with n=10



secondary key, merge the master key and the secondary key and time to decrypt the input file. It is the time between the two points when the user makes a request to download a file and user actually receives his file. Figure 4.6 shows tthe time taken for file downloading for different file sizes.

Figure 4.5: Size vs Time for file uploading

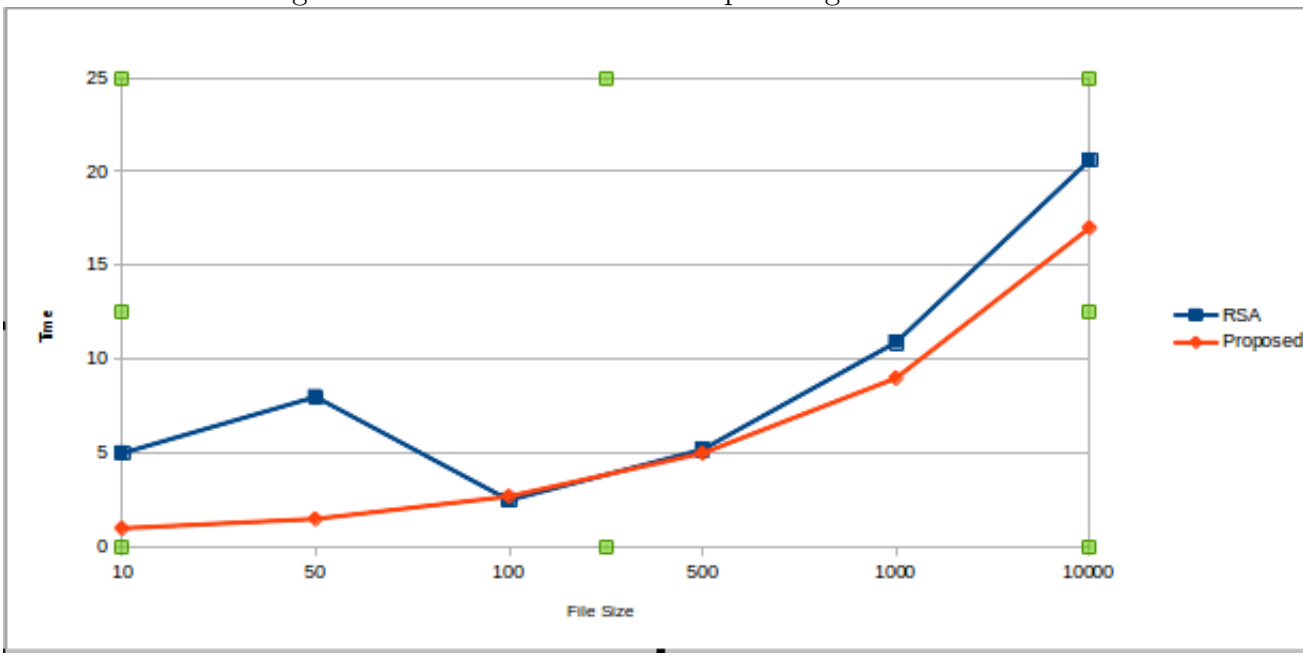
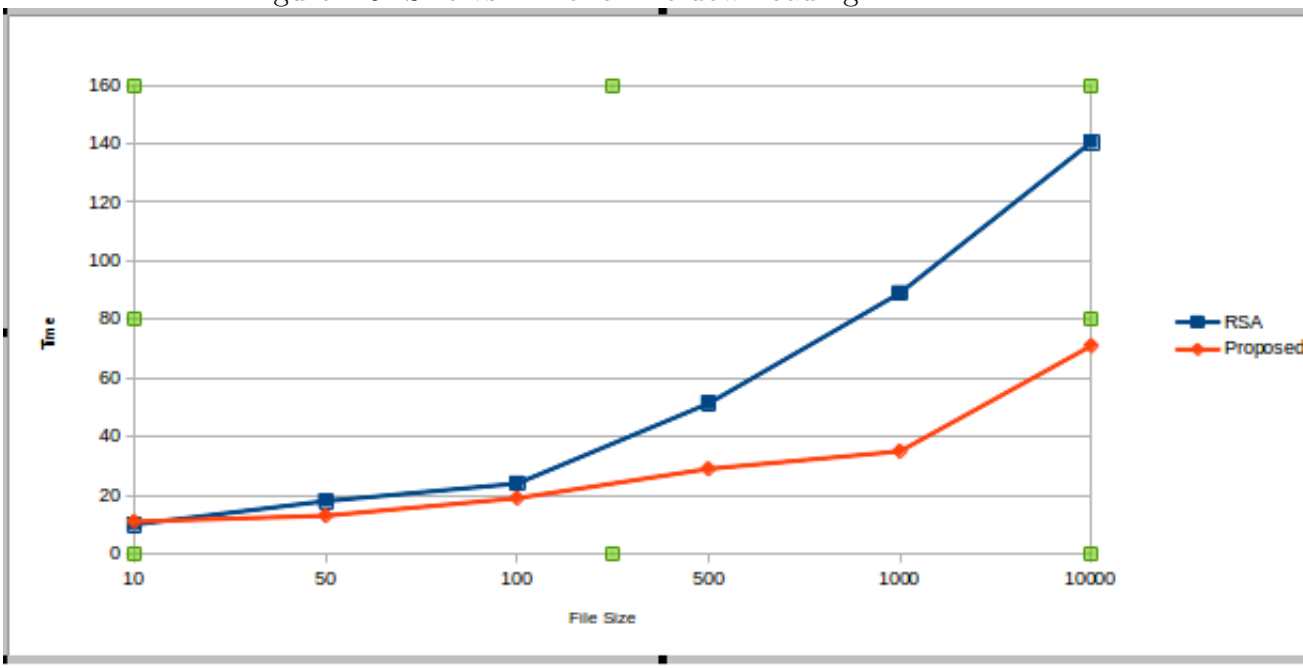


Figure 4.6: Size vs Time for file downloading



Chapter 5

Chapter 5

Conclusion

5.1 Conclusion

Key management is the toughest part to manage in cryptosystems. In the cloud platform, there is always a possibility of insider attack or outsider attack. Keys can be accessed or stolen by employees without the knowledge of end users. Our aim is to provide secrecy to the data as well as keys that are stored in cloud systems. Our proposed technique provides better data security and key management in cloud systems. This technique also provides better security against byzantine failure, server coluding and data modification attacks.

5.2 Future work

In future, this work can be extended to use some other secret sharing schemes which are more effecient so that the performance of proposed system can be further improved. In addition to this, the proposed technique can be extended to work with asymmetric encryption algorithms.

Bibliography

[1] Mell, Peter, and Timothy Grance. "The NIST definition of cloud computing (draft)." NIST special publication 800.145 (2011): 7.

[2] Jaydeep. "Security and Security and Privacy Privacy Privacy Issues in C loud Computing." http://arxiv.org/.

[3] "Cloud Computing Architecture".http://communication.howstuffworks.com/cloud-computingl.htm.

[4] Brodkin, Jon. "Gartner: Seven cloud-computing security risks." Infoworld (2008): 1-3.

[5] Calheiros, Rodrigo N., et al. "Cloudsim: A novel framework for modeling and simulation of cloud computing infrastructures and services." arXiv preprint arXiv:0903.2525 (2009).

[6] Ogbu, Richard Chukwu, and Ifeanyi Ugbaga Nkole. "Cloud Computing: A review."

[7] Shamir, Adi. "How to share a secret." Communications of the ACM 22.11 (1979): 612-613.

[8] Liu, C.L. Introduction to Combinatorial Mathematics. McGraw- Hill, New York, 1968.

[9] Damgrd, Ivan, et al. "Secure key management in the cloud." Cryptography and Coding. Springer Berlin Heidelberg, 2013. 270-289.

[10] Mazieres, David, et al. "Separating key management from file system security." ACM SIGOPS Operating Systems Review 33.5 (1999): 124-139.

[11] Asmuth, Charles, and John Bloom. "A modular approach to key safeguarding." IEEE transactions on information theory 30.2 (1983): 208-210.

[12] Chandramouli, Ramaswamy, Michaela Iorga, and Santosh Chokhani. Cryptographic Key Management Issues and Challenges in Cloud Services. Springer New York, 2014.

[13] Almorsy, Mohamed, John Grundy, and Ingo Mller. "An analysis of the cloud computing security problem." the proc. of the 2010 Asia Pacific Cloud Workshop, Colocated with APSEC2010, Australia. 2010.

[14] Rafaeli, Sandro, and David Hutchison. "A survey of key management for secure group communication." ACM Computing Surveys (CSUR) 35.3 (2003): 309-329.

[15] Kalyani M. ."Cloud Security: Efficient and Reliable Encryption Key Management Crucial for Data Protection". https://spideroak.com/privacypost/cloud-security/secure-encryption-key-management-in-the-cloud/

[16] Blakley, George Robert. "Safeguarding cryptographic keys." Managing Requirements Knowledge, International Workshop on. IEEE Computer Society, 1899.

[17] Vukolic, Marko. "The Byzantine empire in the intercloud." ACM SIGACT News 41.3 (2010): 105-111.

[18] Yung-Wei Kao, Kuan-Ying Huang, Hui-Zhen Gu, Shyan-Ming Yuan: uCloud: a user-centric key management scheme for cloud data protection. IET Information Security 7(2) (2013)

[19] Agbaria, Adnan, and Roy Friedman. "Overcoming Byzantine Failures Using Checkpointing." University of Illinois at Urbana-Champaign Coordinated Science Laboratory technical report no. UILU-ENG-03-2228 (CRHC-03-14) (2003).