Project Report

on

"Secure Data Storage On Multi-Cloud Using DNA Based Cryptography"

Submitted by :

Sagar Vithal Dhuri

Prakash Kondibhau Naikade

Nilesh Balakrishna Gade

Abhijeet Bhagwan Teke

Under the Guidance of:

Mrs. D. S. Zingade

Bachelor of Computer Engineering At

All India Shri Shivaji Memorial Society's

Institute Of Information Technology

Shivaji Nagar, Pune - 01

Academic Year 2014 - 2015

Affiliated to

Savitribai Phule Pune University

AISSMS's Institute Of Information Technology,

Shivaji Nagar, Pune- 01

CERTIFICATE

This is to certify that the Project Report entitled ***"Secure Data Storage On Multi-Cloud Using DNA Based Cryptography"***

submitted by

Sagar Vithal Dhuri

Prakash Kondibhau Naikade

Nilesh Balakrishna Gade

Abhijeet Bhagwan Teke

is the record of bonafide work carried out by them in the partial fulfillment of requirement of award of Degree of **Bachelor of Engineering** (Computer Engineering) as prescribed by the University of Pune in the academic year 2014-15.

Date: .....

.....

Project Guide [Mrs. D. S. Zingade]

.....

HOD,Computer Engineering[Mrs. S.N.Zaware]

.....

Principal [Dr. P. B. Mane]

ACKNOWLEDGMENT

Apart from our own, the success of this report depends largely on the encouragement and guidelines of many others. We are especially grateful to our guide **Mrs. D. S. Zingade** who has provided guidance, expertise and encouragement.

We express our heartfelt gratefulness to **Mrs. D. S. Zingade** and **Mrs. S.N.Zaware**, Head of Computer Engineering Department, AISSMS's IOIT, for their stimulating supervision whenever required during our project work. We are also thankful to the staff of **Computer Engineering Department** for their cooperation and support.

We would like to put forward our heartfelt acknowledgement to all our classmates, friends and all those who have directly or indirectly provided their overwhelming support during our project work and the development of this report.

Sagar Vithal Dhuri
Prakash Kondibhau Naikade
Nilesh Balakrishna Gade
Abhijeet Bhagwan Teke

ABSTRACT

Cloud computing has a great capability to boost productivity and minimize costs, hence many users are embracing it, but at the same time it constitutes many security risks and challenges.

Due to possibilities of multiple risks such as service outage, theft of data, data leakage and the chances of malicious insider attack, using single cloud is becoming obnoxious by many users and new notion of Multi-Clouds usage is becoming perceptible to cope with these security issues.

We aim to demonstrate powerful security strategy of using DNA based cryptography which ensures secure data storage on multi-cloud.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Overview

Cloud computing offers great potential, but at the same time it presents many security risks and challenges. Using single cloud provider is becoming less popular due to service availability failure risk and the possibility of malicious insiders in the single cloud. Solution that comes up recently is multi-clouds, or in other words, inter-clouds or cloud-of-clouds.

Better security and data availability can be achieved by breaking down the users critical data block into parts and dispersing them among the available Cloud Service Providers (CSP). Each divided part of data can be further protected by utilizing some interesting features of DNA sequences and data hiding.

This project aims to review DNA Encryption as a possible solution for security and privacy concerns of cloud.

1.2 Brief Description

1.2.1 DNA Encryption

Deoxyribonucleic acid [DNA] is a long polymer made from repeating units called nucleotides. DNA was first identified and isolated by Friedrich Miescher and the double helix structure of DNA was first discovered by James Watson and Francis Crick, using experimental data collected by Rosalind Franklin and Maurice Wilkins. The structure of DNA of all species comprises two helical chains.

Watson-Crick Base Pairing Rules

In order to convert binary data into amino acids as a DNA sequence, the base pairing rules must be used.

Synthesizing nucleotides in real environment (biology) is done in constant rules:

- Purine Adenine (A) always pairs with the pyrimidine Thymine (T).
- Pyrimidine Cytosine (C) always pairs with the purine Guanine (G).

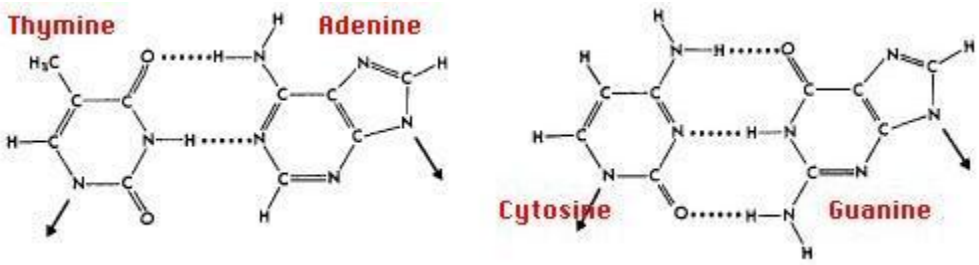

Figure 1.1: Natural Pairing Of Nucleotides

DNA In Computing Area

In order to gain more complexity and to make it hard for intrusion by attacker, we will change these universal rules. For example, in biology A is synthesized to T while we can assume A to C, T or G, anything and so on, as we decide.

1.2.2 Multi-cloud

Data availability, security, privacy, and integrity are the most critical issues to solve in cloud computing. Even though the cloud service providers have powerful infrastructure along with standard regulations to provide a better availability and ensure customers data privacy, there still exist many reports of service outage and privacy breach in last few years. Solution is using multiple clouds for storing critical data.

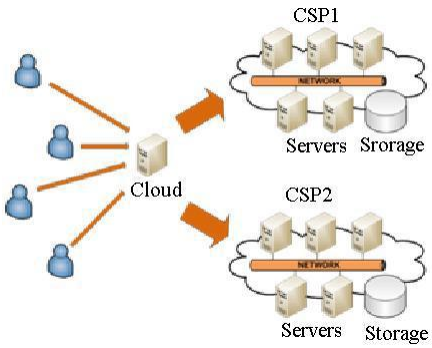

Figure 1.2: Multi-cloud Architecture

Chapter 2

Literature Survey

2.1 DNA Encryption

Secret Data Writing Using DNA Sequences

Deepak Kumar and Shailendra Singh ***978-1-4577-0240-2/11/&26.002011 IEEE***

Over last 50 years, DNA, the magic code of life, has been known for genetic material of living systems. In past few years, much research work has been carried out in DNA based cryptography field . Density of information and vast parallelism to DNA in addition to results of Adlemans experiments encouraged many researchers to exploit bio-computing area to solve the hard problem of computer sciences fields. Lipton extended this approach to solve NP complete problem. DNA computing provides a firenew data structure and calculating methods for parallel processing capability with molecules. Boneh et al. demonstrated an approach to break the DES algorithm by using DNA computing methods. DNA cryptography is a new born cryptography field emerged with the research of DNA computing, in which DNA is used as information carrier and modern biological technology is used as implementation tool.

DNA cryptography researches are in its primitive stage, and there are many problems to be solved. DNA cryptography is far from both mature theory and realization this is the strong reason that why only few example are proposed. Some key technologies in DNA research, such as DNA synthesis, DNA digital coding and Polymerase Chain Reaction (PCR) have only been proposed.

Automatic DNA Sequence Generation for Secured Effective Multi -Cloud Storage
IOSR Journal of Computer Engineering (IOSR-JCE)

In order to convert binary data into amino acids as a DNA sequence, the base pairing rules must be used. Synthesizing nucleotides in real environment (biology) is done in constant rules:

- Purine Adenine (A) always pairs with the pyrimidine Thymine (T).
- Pyrimidine Cytosine (C) always pairs with the purine Guanine (G).

Always, those rules are done naturally because the opportunities to synthesize hydrogen bonds between A and T (two bonds), and also between C and G. These concepts are named Watson-Crick base pairing rules.

In binary computing area, it is possible to change the natural rules by own decision. For example, in biology A is synthesized to T while we can assume A to C or A to G, and so on, as we prefer. Increasing the complexity of the algorithm is the main purpose of the changing the rules. A way to increase the complexity is complementary pair rule. Complementary pair rule is a unique equivalent pair which is assigned to every nucleotides base pair. There are four basic alphabets therefore four likelihood of complementary rule for every DNA sequences. So, the final number of possible those rules are 4*3*2*1=24.

2.2 DNA Encryption In Cloud

Enabling Data Hiding for Resource Sharing in Cloud Computing Environments Based on DNA Sequences
2011 IEEE World Congress on Services

One of the famous ways to protect data through the Internet is data hiding. Because of the increasing number of Internet users, utilizing data hiding or Steganographic techniques is inevitable. Eliminating the role of the intruder and authorizing the clients are eventual goals of these techniques.

Therefore, the role of data hiding has become more eminent nowadays. Before employing biological properties of DNA sequences, the common way of embedding a secret data into the host images was the traditional way of data hiding . It unfortunately leads to some liabilities. The most important ones was the detection of the distortions of the image when the host image changed to some degrees. That was the best spot to start the wholly detection of the secret data through the image. By advent of biological aspects of DNA sequences to the computing areas, new data hiding methods have been proposed by researchers, based on DNA sequences . The key portion of their work is, utilizing biological characteristics of DNA sequences.

2.3 DNA Encryption In Multi-cloud

Multi Cloud Architecture to Provide Data Security And Integrity

International Journal of Emerging Technology and Advanced Engineering

Cloud computing is a model for enabling convenient, on demand network access to a shared pool of configurable computing resources that can be rapidly provisioned and released with minimal management effort. The SaaS (software as a service) cloud model can be defined as a cloud deployment model in which centrally built software is used for users to access the cloud. Cloud computing offers a data storage system which enables the users to be less dependent on the client system and provides an architecture to upload the data in a cloud that can be shared by multiple users and also provide security through authentication of the user. This architecture introduces a secondary cloud controlled by a single administrator which provides the data backup for primary cloud after undergoing specific segmentation and encryption algorithms to ensure security and integrity of data.

Review on Multi-Cloud DNA Encryption Model for Cloud Security

Richa H. Ranalkar et al. Int. Journal of Engineering Research and Applications : www.ijera.com

Data availability, security, privacy, and integrity are the most critical issues to solve in cloud computing. Even though the cloud service providers have powerful infrastructure along with standard regulations to provide a better availability and ensure customers data privacy, there still exist many reports of service outage and privacy breach in last few years. Solution is using multiple clouds for storing critical data.

Better security can be obtained by dispersing the user data over multiple cloud service providers (CSP) in such a way that, none of the CSP can successfully retrieve meaningful information from the data pieces allocated at their servers. Also, because of redundancy in data distribution, user is assured of data availability. If a service provider goes bankrupt or suffers service outage, the user still can access his critical data from other CSPs. Thus advantages of using Multi-cloud storage are data availability, avoid vendor lock-in, business continuity and disaster recovery.

Given p number of cloud service providers, User will divide his data into N data pieces where at-least k data pieces out of N data pieces are necessary to recover any meaningful information of the data; as data redundancy is used. This (k, N) is the first threshold. Second threshold is of (q, p); which implies, at least q out of p number of CSPs must be a part of retrieval process for successful data retrieval.

2.4 Advantage Of DNA Encryption

An Efficient Data Security Model for Cloud Environment Using DNA Cryptography Technique

2014 INTERNATIONAL CONFERENCE ON COMPUTATION OF POWER, ENERGY, INFORMATION AND COMMUNICATION (ICCPEIC)

Table 2.1: Comparison of various performance parameters for different encryption algorithms

Types Of Algorithm	**Computation Time (seconds) for Databits (bits)**						**Average Time**	**Throughput**	**Storage (Mb)**
	8	*10*	*32*	*64*	*128*	*256*			
RSA	50	52	54	56	58	60	55.00	9.163	82.26
AES	48	49	50	50	53	56	51.00	9.881	82.15
DES	49	49	51	51	52	55	51.16	9.853	82.35
DNA	50	50	51	51	52	53	50.26	10.02	82.03

Table 2.2: Comparison of various performance parameters for different Decryption algorithms

Types Of Algorithm	**Computation Time (seconds) for Databits (bits)**						**Average Time**	**Throughput**	**Storage (Mb)**
	8	*10*	*32*	*64*	*128*	*256*			
RSA	58	60	61	63	65	68	62.50	8.064	136.48
AES	56	58	59	59	60	61	58.83	8.566	136.37
DES	59	60	61	61	63	64	61.33	8.217	136.17
DNA	53	54	55	55	56	58	55.10	9.137	136.02

2.5 Summary

Cloud computing is restructuring how IT resources and services to be used and managed, but major problem in cloud implementation is security challenges. By dividing users data and applying data hiding using DNA encryption, and then storing it on multiple clouds; this model has shown its ability of providing a cloud customer with a more secured storage. The proposed concepts discussed here will help to build strong security architecture in cloud computing. This will also improve customer satisfaction and will attract more investors for industrial as well as future research farms.

Chapter 3

Software Requirement Specification

3.1 Introduction

As, the use of cloud computing is increasing rapidly, because these services provide fast access to their applications and reduce their infrastructure costs. Cloud providers should address privacy and security issues as a matter of high and urgent priority. Our system focuses on the issues related to the data security aspect of cloud computing as follows

- Data Integrity :-

  The data stored in the cloud may suffer from damage during transition operations from or to the cloud storage provider. Example of a risk to data integrity recently occurred in Amazon S3 where users suffered from data corruption. Our system ensures this issue with the help of multiple cloud and DNA encryption algorithm which encrypts the files before sending it to the providers and at the providers end. This provides high data integrity to clients

- Data Intrusion :-

  As data and information will be shared with a third party, cloud computing users want to avoid an untrusted cloud provider. If third person gains access to the clients account password, he would be able to access all the account resources. Protecting private and important information and files from attackers or malicious insiders is of critical importance. This can be achieved by the use of secret sharing technique to perform the file operations. The client needs the secret keys to upload and download the files which ensure the security against the data intrusion.

- Data availability :-

  Another major concern in cloud services is data availability. In the single cloud environment it is pos-

sible that the service might be unavailable from time to time. Our system ensures the data availability by replicating the clients data to multiple providers. As we are using multiple providers our system ensures the data availability. If in case, one provider fails clients data will still be available to them.

3.1.1 Project Scope

Cloud computing offers great potential, but at the same time it presents many security risks and challenges. Using single cloud provider is becoming less popular due to service availability failure risk and the possibility of malicious insiders in the single cloud. Solution that comes up recently is multi-clouds, or in other words, inter-clouds or cloud-of-clouds.
Better security and data availability can be achieved by breaking down the users critical data block into parts and dispersing them among the available Cloud Service Providers (CSP). Each divided part of data can be further protected by utilizing some interesting features of DNA sequences and data hiding.

3.1.2 User Classes and Characteristics

- Client :-
  Client can register and log in. Client is be able to upload and download file. Also he can update profile. He holds his own private key. So he uploads file in Encrypted format.

- Cloud Service Provider :-
  Cloud service provider is the admin of the cloud storage who can store file sent to it. It is the database administrator over the cloud storage space.

- Application Provider :-
  Application provider is the bridge between client and cloud service provider which performs various methodology to store data modules from client over multiple clouds.

3.1.3 Operating Environment

The Operating Environment for the software part of the product consists of a PC or Laptop with Microsoft Windows operating environment. The hardware platform must be compatible with Microsoft Windows Operating Systems XP, Vista or Windows 7 as these are the most common operating system.

H/W System Configuration:-

- Processor - Intel Core 2 Duo/i3/ i5/i7

- Speed - 1.9/2.1 GHz

- RAM - 1024 MB (min)
- Hard Disk - 20 GB (min)
- Server/client computers

S/W System Configuration:-

- Operating System - Windows XP/7
- Application Server - Tomcat5.0/6.X
- Front End - HTML, Java, Jsp
- Client Side Scripts - JavaScript.
- Server side Script - Java Server Pages.
- Database Connectivity - Mysql/Oracle/SQL Server

3.1.4 Design and Implementation Constraints

- Design Constraints :-
  - Error Recognition: Error should be easily recognized and get solved out.
  - Exception: All kind of exception should be handling properly.
- Design Constraints :-
  - This software would be designed to work only on Linux and Windows platform using MySQL database.

3.1.5 Assumptions and Dependencies

Assumptions

- Users own the computer and know how to operate it.
- We further assume that the hardware required by the system is already present.
- We assume that all the components are compatible.
- A further assumption is that the computer being used can understand and interpret the programming of the software.

Dependencies

- This system highly depends on internet service.
- Secondly it require computer to carry out the operations.

3.2 System features

Our system focuses on the issues related to the data security aspect of cloud computing. As data and information will be shared with a third party i.e. cloud, Users want to avoid an untrusted cloud provider for Protecting private and important information.

Security Risks in Cloud Computing and proposed work

We have mentioned security issues in detail in our introduction. Now, we focus on solutions related to these problems through our system features.

- Data Integrity
  - We can counter data integrity using cryptography.
  - We can encrypt the data stored on cloud so that malicious insiders cant access our data.
- Data Intrusion :-
  Security risk that may occur with a cloud provider, such as the Amazon cloud service, is a hacked password or data intrusion. If someone gains access to an Amazon account password, they will be able to access all of the accounts instances and resources. Thus the stolen password allows the hacker to erase all the information inside any virtual machine instance for the stolen user account, modify it, or even disable its services. Furthermore, there is a possibility for the users email (Amazon user name) to be hacked, and since Amazon allows a lost password to be reset by email, the hacker may still be able to log in to the account after receiving the new reset password.
- Service availability :-
  Solution for availability can be DepSky Architecture The DepSky architecture consists of four clouds and each cloud uses its own particular interface. The DepSky algorithm exists in the clients machines as a software library to communicate with each cloud. These four clouds are storage clouds, so there are no codes to be executed. In our system, we need to incorporate all these modules with user uploading and downloading the data on cloud.

3.3 External Interface Requirements

3.3.1 User Interfaces

- Provision for user login or registration.
- Provision for owner login.
- User friendly interface.

3.3.2 Hardware Interfaces

- 1024 MB RAM Memory
- Network cards

3.3.3 Software Interfaces

- Oracle 10g
- Net beans 6.8
- Eclipse
- JDK7
- Visual Paradigm for UML Modeling
- File System Database.

3.4 Non-Functional Requirements

3.4.1 Performance Requirements

- System should be high speed.
- System should be reliable.
- System should be efficient.
- User interface must be very friendly self descriptive.
- Client-server response should be as fast as possible.
- Application must consider the case of Network Failure.

3.4.2 Safety Requirements

- The data safety must be ensured by arranging for a secure and reliable transmission media.
- Cloud Storage should be reliable.
- Dealing with single cloud providers is predicted to become less popular with customers due to risks of service availability failure and the possibility of malicious insiders in the single cloud.

3.4.3 Software Quality Assurance

Software should have following quality attributes:

Quality of service attributes:

- **Availability** : Data should be Available all the time
- **Response time** : It should be as fast as possible
- **Scalability** : Automatic
- **Security Privacy** : Data storage and transfer should be Secure
- **Costs** : Cost should be Low for the users

Subjective wishes needs:

- **Sustainability** : Efficient
- **Cost effectiveness** : Should be good enough.
- **Usability** : It should be easy to use

3.5 Database Requirements

- **Availability** : Database should be available all the time
- **Manageability** : It should be easily manageable both for routine tasks and monitoring
- **Cost** : Costs should be as lower as possible.

3.6 System Design Diagrams

3.6.1 Use Case Diagrams

It shows the set of use cases and actors (special kind of the class and their relationship).Use case diagram addresses the static use case of the system. These diagrams are especially important in organizing and modeling the behavior of a system.

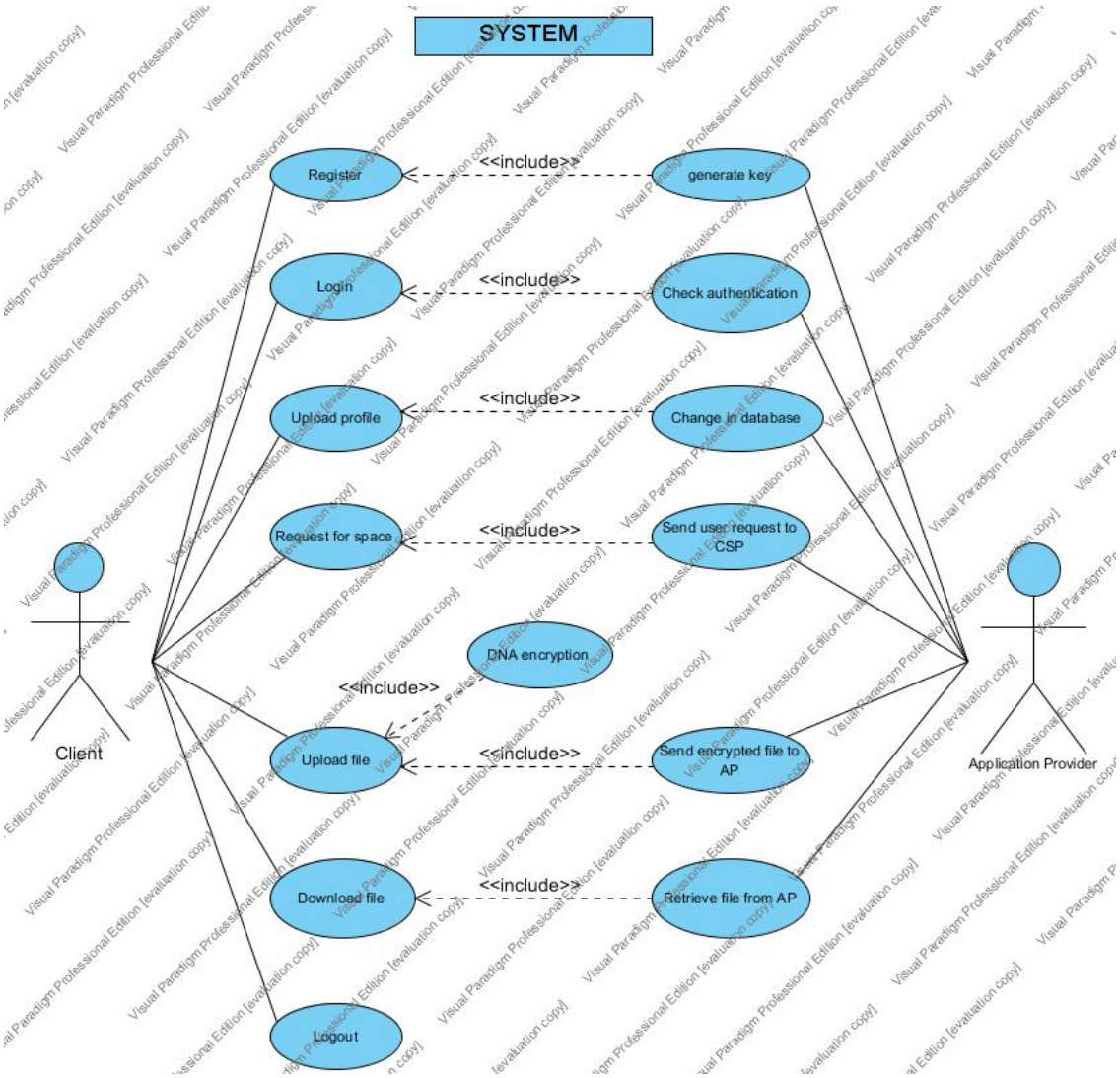

Figure 3.1: Use Case For Client And Application Provider

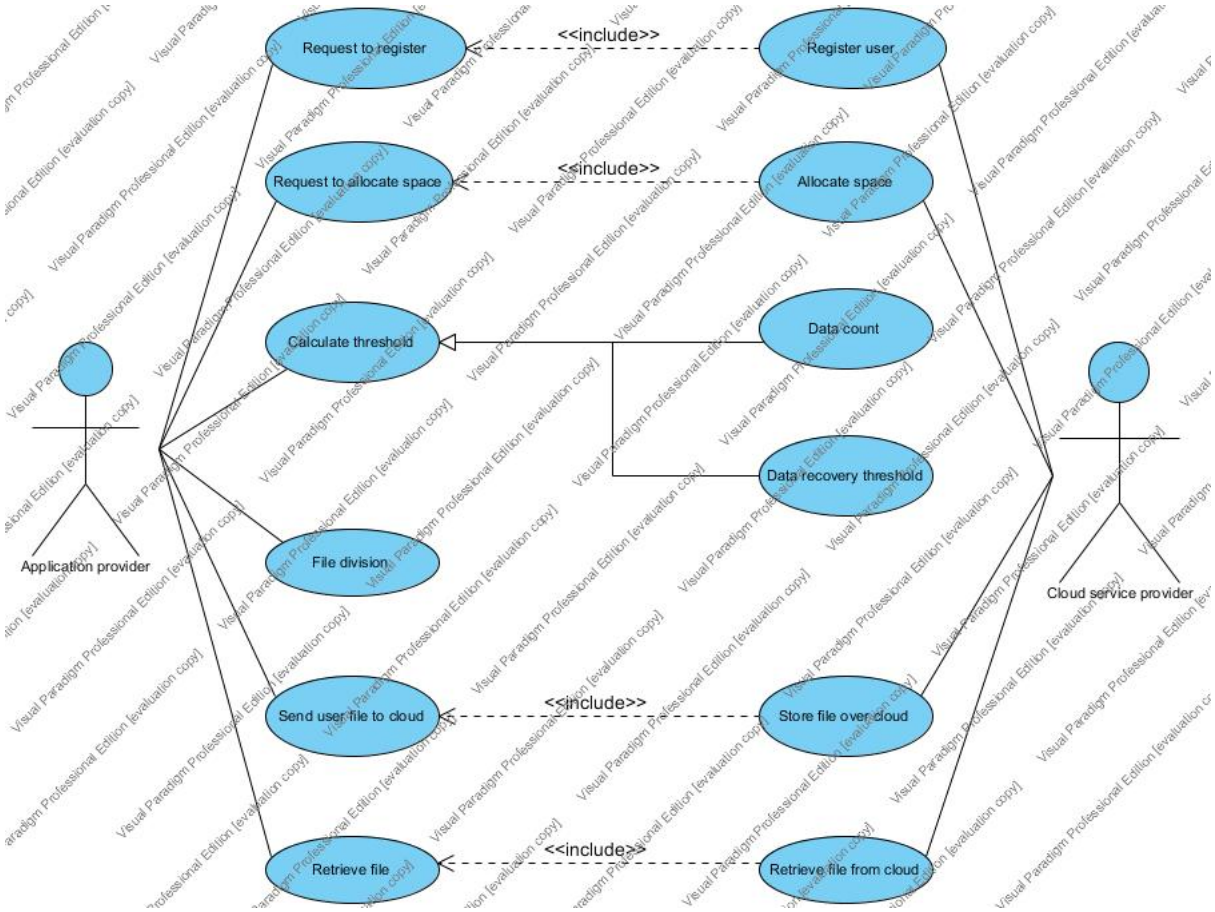

Figure 3.2: Use Case For Application Provider And Cloud Service Provider

3.6.2 Class Diagram

A class diagram is a type of static structure diagram that describes the structure of a system by showing the systems classes, their attributes, operations or methods and the relationships between the classes.

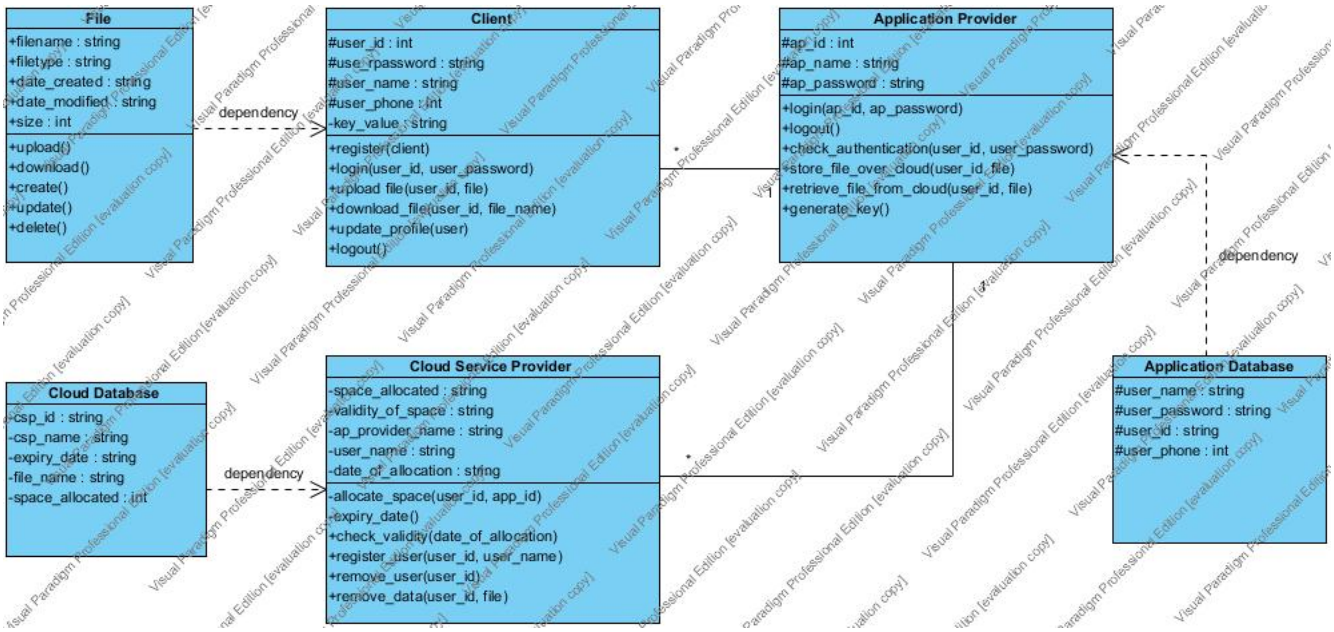

Figure 3.3: System Class Diagram

3.6.3 Sequence Diagram

A sequence diagram shows the time-ordering of message. It actually gives you the step by step flow of the system. It is dependent on time.

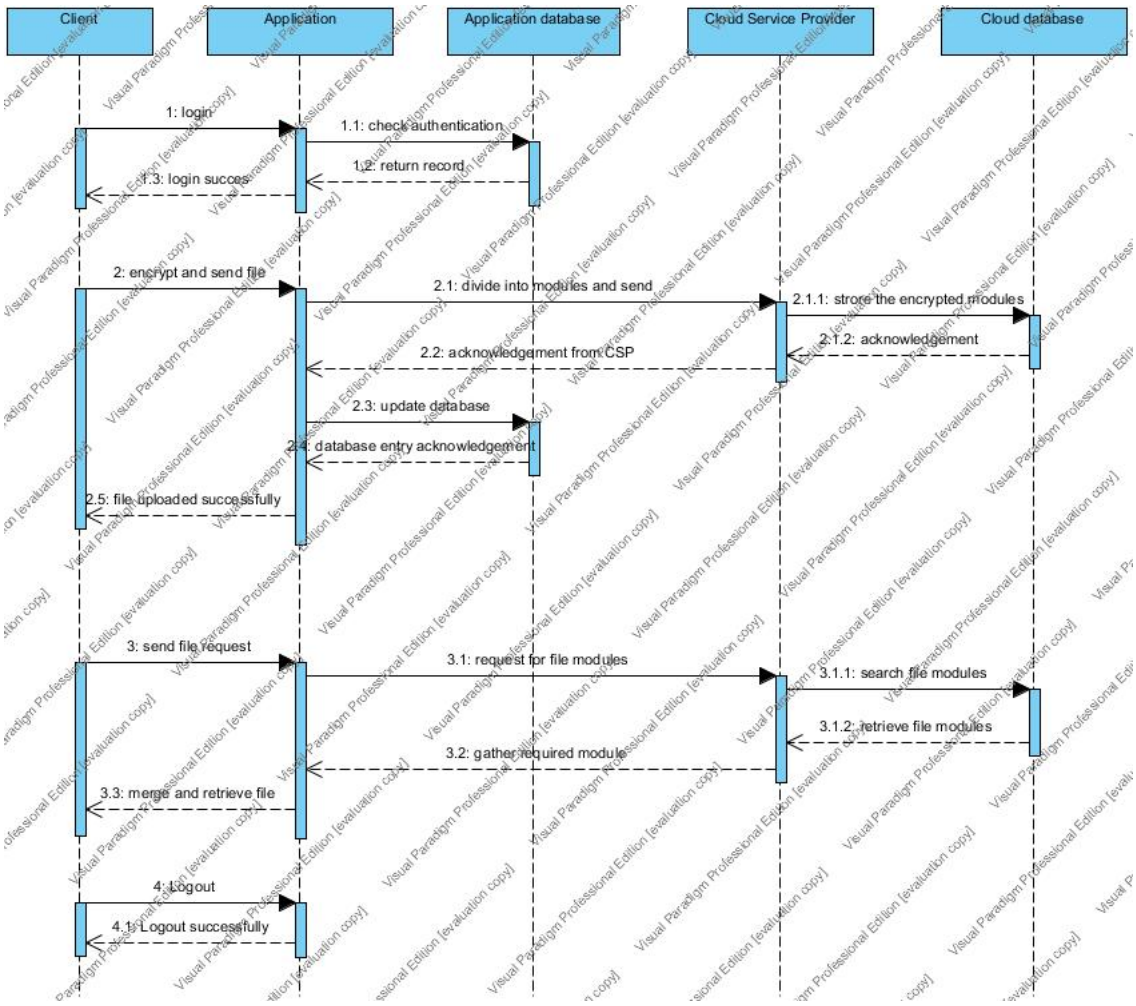

Figure 3.4: System Sequence Diagram

3.6.4 Collaboration Diagram

A collaboration diagram shows the communication of messages within the system. It also gives the step by step flow of the system. It is dependent on time.

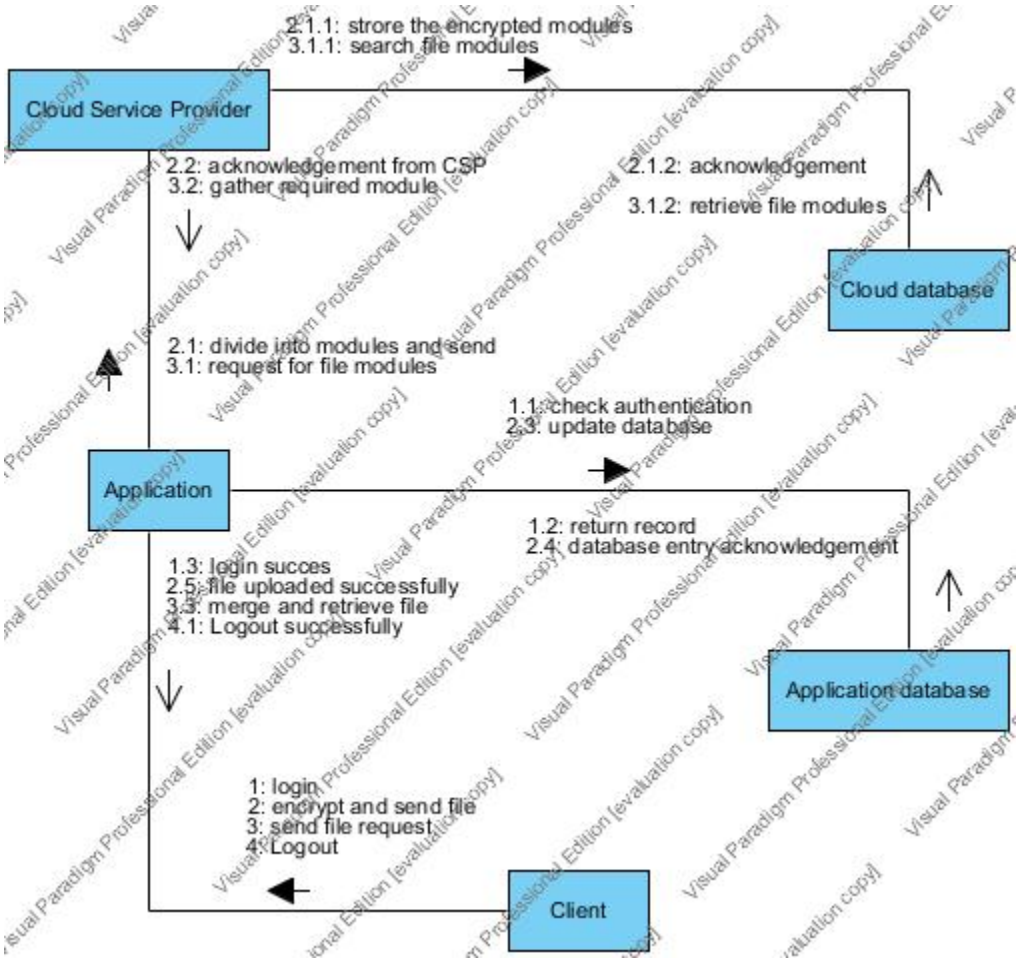

Figure 3.5: Collaboration Diagram

3.6.5 State Transition Diagram

A State machine diagram consists of states, transitions events and activities. It addresses the dynamic view of the system.

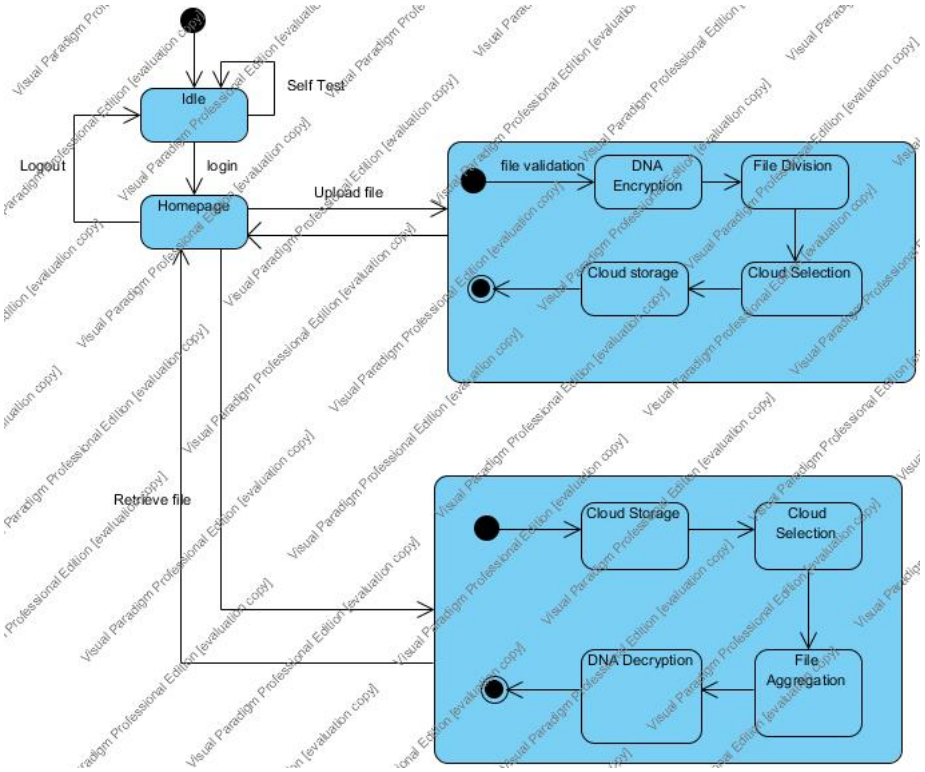

Figure 3.6: State Transition Diagram

3.6.6 Deployment Diagram

A deployment diagram in the Unified Modeling Language models the physical deployment of artifacts on nodes.

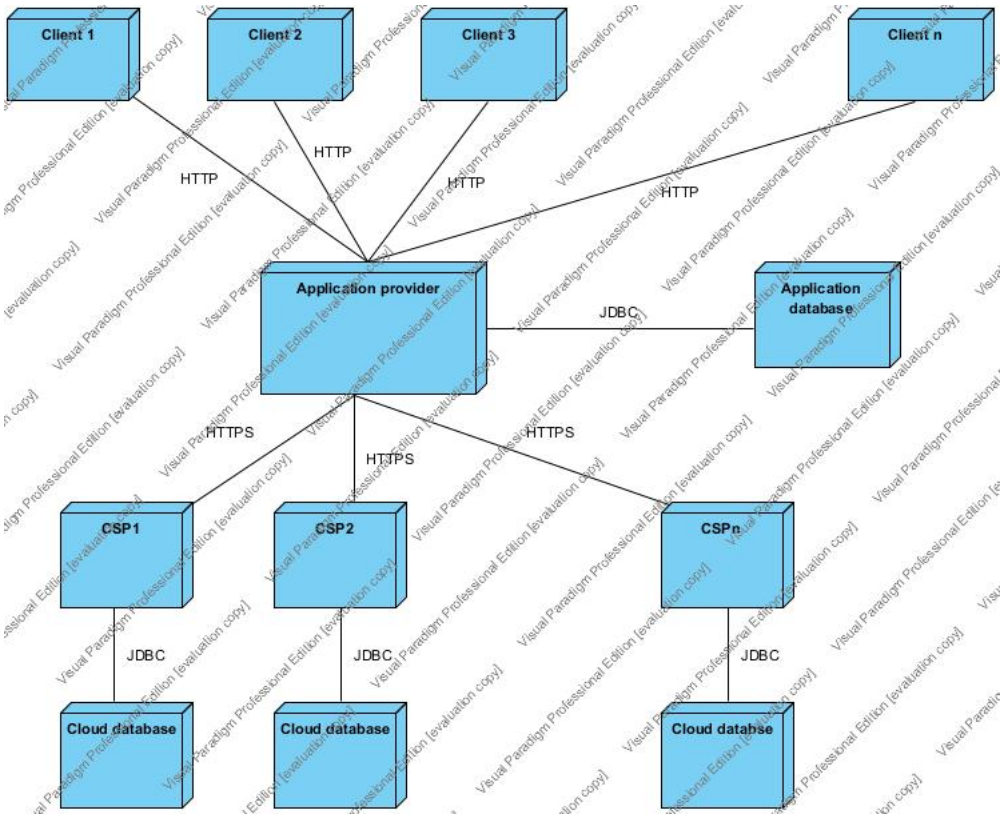

Figure 3.7: Deployment Diagram

3.6.7 Component Diagram

A deployment diagram in the Unified Modeling Language models the physical deployment of artifacts on nodes.

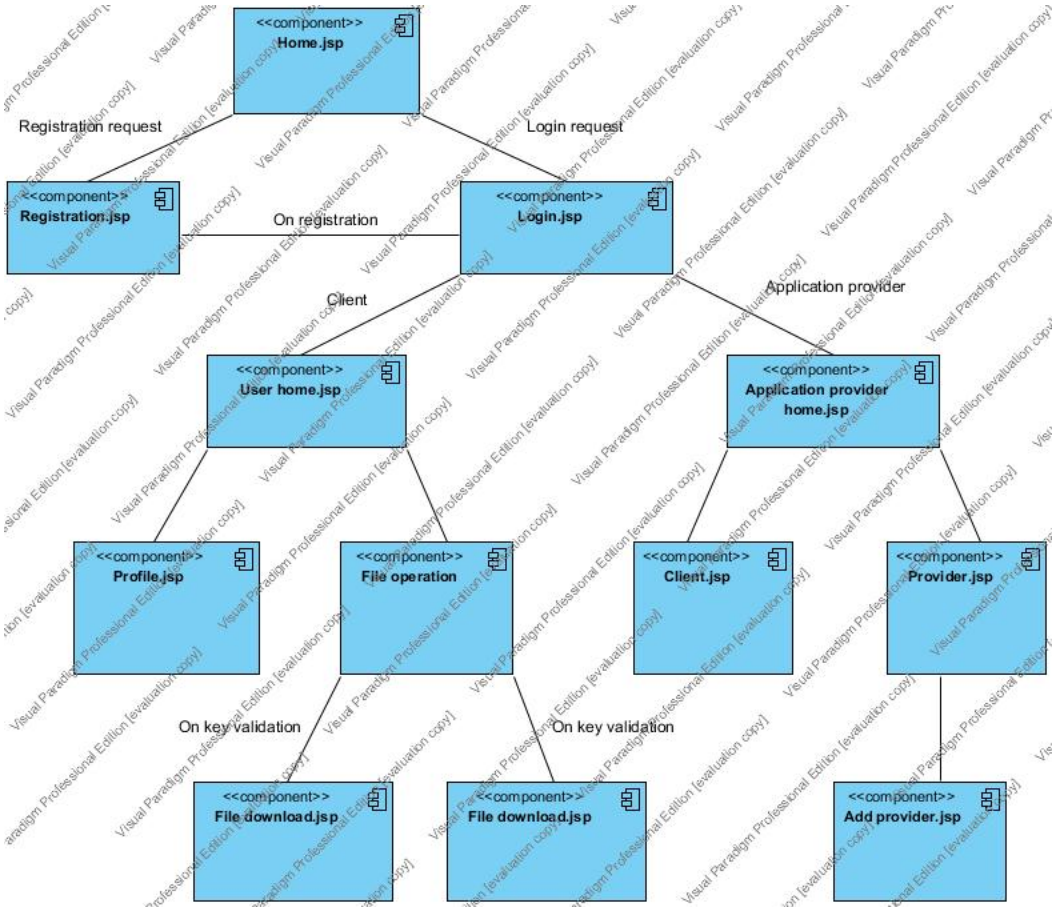

Figure 3.8: Component Diagram

Chapter 4

Advantages, Disadvantages, Future scope

4.1 Advantages

- Data Integrity
- Service Availability.
- The user runs custom applications using the service providers resources.
- Highly Secure.

4.2 Disadvantages

- Time Consuming.

4.3 Future Scope

- Many applications are moving to the cloud, so, it is possible to think of new applications that would use the storage cloud as a back-end storage layer.
- Storing and sharing the data on cloud is much easy and secure with our system.
- Data availability issue will be solved completely as we are addressing.

Chapter 5

Conclusion

Cloud computing is restructuring how IT resources and services to be used and managed, but major problem in cloud implementation is security challenges. By dividing users data and applying data hiding using DNA encryption, and then storing it on multiple clouds; this model has shown its ability of providing a cloud customer with a more secured storage. The proposed concepts discussed here will help to build strong security architecture in cloud computing. This will also improve customer satisfaction and will attract more investors for industrial as well as future research farms.

Chapter 6

Annexure A

Laboratory Assignments on Project Analysis of Algorithmic Design

Mathematical Model

6.1 Embedding Data

In order to explain embedding phase, separating the phases into some successive and vivid sub-phases, is the best way of proposing current method. In below, sub-phases have been shown, respectively.

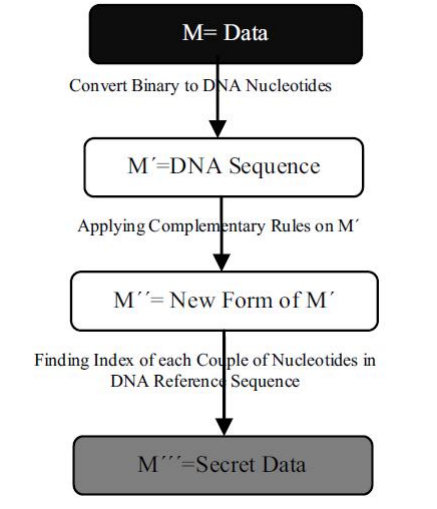

Figure 6.1: Embedding secret data

The first sub-phase is, converting by DNA base pairing rules. The product is M. M contains nu-

cleotides sequences. **The next (second) sub-phase is**, applying complementary rules. Increasing the complexity is the real and exact purpose of this step. By applying the complementary rules, the new form of the M which is M emerges.

When all the indexes have been extracted, M has been made, properly. M is precisely the secret data with some changes through the embedding phase.

- DNA Reference Sequence:
  $\mathrm{AT}_1CG_2AA_3TT_4CG_5CG_6CT_7GA_8GT_9CA_{10}CA_{11}AT_{12}TC_{13}GC_{14}GC_{15}TG_{16}AG_{17}TG_{18}AA_{19}CC_{20}$

- Let The Message Be M.
  M = 100111000011

- Sub-phase1$_{(A=00,T=01,C=10,G=11)}$:
  $M' = CTGAAG$
  $M' = \Sigma_{start}^{eof}(encode1(d))$;
  where eof $\rightarrow$ $End\ Of\ File$
  $encode1 \rightarrow function\ returning\ DNA\ nucleotide\ for\ set\ d$
  $d = (d1, d2) \mid d1\ and\ d2\ are\ two\ consecutive\ digits\ in\ the\ file$
  $M'\ is\ phase\ 1 encryption\ message.$

- Sub-phashe2$_{((AC)(CG)(GT)(TA))}$:
  $M'' = GATCCT$
  $M'' = \Sigma_{start}^{eof}(encode2(x))$;
  $encode2 \rightarrow function\ returning\ DNA\ nucleotide\ compliment\ for\ set\ 'x'$
  $x = \{A, T, C, G\}$
  $M',\ is\ phase\ 2\ encryption\ message.$

- Sub-phase3 (Indexes):
  M"' = 8137
  M"' = $\Sigma_{start}^{eof}(encode3(g))$;
  $encode3 \rightarrow function\ returning\ indexes\ for\ DNA\ pair$
  $g = (g1, g2) \mid DNA\ Pair\ from\ M''$
  $M'''\ is\ phase\ 3\ encryption\ message.$

Now, embedding phase is finally completed. Then, sender sends 8,13,7 to the cloud. In the next section, the client 2 will apply the extracting phase for extracting the original data by using three consecutive phases.

6.2 Extracting Original data

the secret data in form of some numbers. For the purpose of extracting the original data from DNA reference sequence, phase two with its sub phases will extract the original data, correctly.

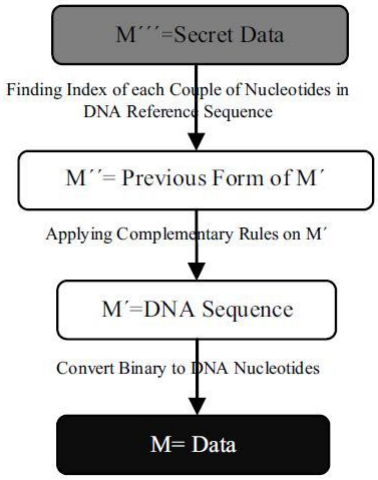

Figure 6.2: Extracting original data

- DNA Reference Sequence:
  $\mathrm{AT}_1CG_2AA_3TT_4CG_5CG_6CT_7GA_8GT_9CA_{10}CA_{11}AT_{12}TC_{13}GC_{14}GC_{15}TG_{16}AG_{17}TG_{18}AA_{19}CC_{20}$

- Sub-phase 1 (Indexes):
  M = 8137
  M" $= \Sigma_{start}^{eof}(decode3(g))$;
  $decode3 \rightarrow function\ returning\ DNA\ pair\ for\ indexes$
  $g = index\ from\ M'''$
  $M''\ is\ phase\ 31\ extracted\ message.$

- Sub-phashe $2_{((AC)(CG)(GT)(TA))}$:
  $M = GATCCT$
  $M'' = \Sigma_{start}^{eof}(decode2(x))$;
  $decode2 \rightarrow function\ returning\ DNA\ nucleotide\ compliment\ for\ set\ 'x'$
  $x = \{A, T, C, G\}$
  $M',\ is\ phase\ 2\ extracted\ message.Sub-phase3_{(A=00,T=01,C=10,G=11)}$:
  $M = CTGAAG$

$M' = \Sigma_{start}^{eof}(decode1(d))$;

$decode1 \rightarrow function\ returning\ binar\ digits\ for\ DNA\ nucleotided = \ DNA\ Nucleotide\ From\ M''M'\ is\ original\ message$

- M = 100111000011

6.3 Throughput

$\mu = (\Sigma_{i=0}^{n} D_n)/T$

where ,

$\mu \ \rightarrow \ Throughput.$

D_n *is data upto n bits.*

T *is Average time consumed.*

6.4 Time Complexity Conclusion (NP Hard/Complete, P)

The Mathematical model obtained gives the solvable output for proposed system. But the time depends on the positions of the base pairs in the reference string. So we conclude that our system lies in NP-complete.

Chapter 7

Annexure B

7.1 Introduction

Testing Strategies: Software Testing Strategies should follow generic characteristics:

- Testing begins at module level and works outward toward the integration of the entire computer base system.
- Different testing techniques are applied at each point.
- Testing is conducted by developer of software.

Testing challenges the assumptions, risks and uncertainty inherent the work of the other disciplines, addressing those concerns by concrete demonstration and impartial evaluation. First testing software is enormously difficult. The different ways a given program can behave are unquantifiable. Second, testing is typically done without a clear methodology so results vary from project to project. A strategy for software testing begins with Unit Testing.

Testing Objectives:

The purpose of testing is to uncover the errors in the software. The objective behind testing is to correct and rectify the errors that are present in the current work product. There are certain rules of testing that can serve as testing objectives:

- Testing is the process of executing the program with the intent of finding errors.
- A good test is the one which has higher probability of finding an error.

- A successful test is the one which uncovers a yet undetected error.

Unit Testing:

It concentrates on each unit of software as implemented in source code. Unit testing focuses verification effort on the smallest unit of software design the software component or module. Using the component-level designed description as a guide, important control path are tested to uncover error is limited by constrained for unit testing. The unit testing is white-box oriented, and the step can be conducted in parallel for multiple components.

Unit testing is normally considered as an adjunct to the coding step. After source level code has been developed, reviewed, and verified for correspondence to component-level designed, unit rest design begins. Each test case should be coupled with a set to expected results.

The Testing process progresses by moving to Integration testing, where the focus is on the design and construction of software design. Next Validation Testing is encountered, where requirements established as part of project requirement analysis are validated against the project that has been constructed. Finally we arrived at System Testing, where project and other system elements are tested as a whole.

System Testing: System testing is actually a series of different tests whose purpose is to fully exercise the computer based system. In system testing, the platform must be as close to production used in the customers environment, we can more accurately test softer system features (performance, security and fault tolerance).

White Box Testing:

White box approach focuses on the inner structure of the software to be tested. Tester must have knowledge of the structure. Using White box testing methods, the software engineer can derive test cases that:

- Guarantee that all independent paths within a module have been exercised at least once.
- Exercise all logical decisions.
- Execute all loops at their boundaries and in their operational bounds
- Exercise internal data structures to maintain their validity.

Black Box Testing:

Black Box Testing focuses on the functional Requirements of the software. It enables the software engineer to derive sets of input conditions that will fully exercise all functional requirements for a program. Using Black box approach, a tester considers the software-under-test to be an opaque box. There is no knowledge of its inner structure. The tester only has knowledge of what it does. The size of the software-under-test using this approach can vary from a simple module, member function, or subsystem to a complete system.

After a system has been successfully built, it becomes necessary to test it prior to deployment. Informally, Software Testing can be defined as the process of exercising a program with the specific intent of finding errors prior to delivery to the end user.
The main purpose of Testing is to:

- Debug and rectify errors in the product.
- See that the product built conforms to the requirements specified.
- Ensure that the product meets certain performance requirements.
- It is also an indication of quality of the product.

7.2 Test Item (Function)

For the basic implementation of the system we will use the Incremental model and we will do the unit testing for testing each module. In our system the module are calculation of the ASP, calculation of the power saved by using proposed system, No Data Loss Check, Trafic Analysis, Implementation of wake up packet.

7.3 Feature to be tested

S. No.	Feature to be tested(Use-Cases)	Level of testing required /priority
1	Input Validation	H
2	File Validation	H
3	Space Allocation	H
4	File Fragmentation	H
5	Fragment Encryption	H
6	Module Recombination	H
7	Constraint Satisfaction	H

Table 7.1: Features to be tested

Chapter 8

Annexure C

Project Planner And Progressin Report

8.0.1 Project Planning

Name	Duration	Begin date	End date
Group Selection	5	6/23/14	6/26/14
Searching Current Trend	6	6/27/14	7/4/14
Domain Selection	7	7/7/14	7/15/14
Domain Project Idea	4	7/8/14	7/11/14
Searching IEEE Papers	5	7/14/14	7/18/14
Project Topic Finalization	2	7/21/14	7/22/14
Survey Regarding Related Work	3	7/23/14	7/25/14
Requirement Gathering	5	7/23/14	7/28/14
Find Possible Techniques Topics	3	7/28/14	7/30/14
Selecting Techniques	4	8/1/14	8/4/14
Hardware & Software System	2	8/5/14	8/6/14
Submit Synopsis	5	8/7/14	8/11/14
Searching and selecting algorithms	4	8/12/14	8/15/14
Literature Survey	3	8/18/14	8/18/14
Preparing Mathematical Model	6	8/19/14	8/25/14
All UML Diagram	3	8/26/14	8/28/14
Designing module on paper	4	8/29/14	9/1/14
SRS creation	4	9/2/14	9/4/14
SRS modification and submission	3	9/5/14	9/5/14
Planning for requirements testing	8	9/8/14	9/15/14
Prepare requirement testing	8	9/16/14	9/23/14
Submission of report	9	9/24/14	10/2/14

Figure 8.1: Project Planning 1

Learn Basics of Java	17	10/3/14	10/20/14
Learn java scripts	7	10/21/14	10/28/14
Development of GUI components	10	10/28/14	11/11/14
Create sample apllication	20	11/12/14	12/8/14
Creation of database of client server	15	12/9/14	12/26/14
Development of login module	10	12/29/14	1/8/15
Development of server module	20	1/9/15	2/5/15
Integration of all module	20	2/6/15	3/5/15
Testing	15	3/6/15	3/26/15
project Demo	4	3/27/15	3/30/15
Final Report Submission	1	3/31/15	3/31/15

Figure 8.2: Project Planning 2

8.0.2 Progression Chart

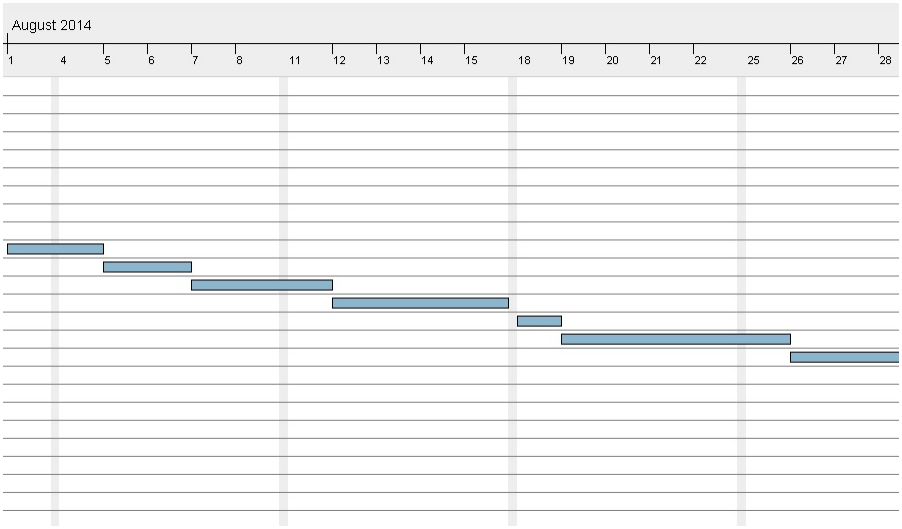

Figure 8.3: Progression Chart

Bibliography

[1] "Enabling Data Hiding for Resource Sharing in Cloud" , *2011 IEEE World Congress on Services*, by Mohammad Reza Abbasy, Bharanidharan Shanmugam

[2] "Secret Data Writing Using DNA Sequences", *978-1-4577-0240-2/11/$26.00 2011IEEE*, by Deepak Kumar, Shailendra Singh

[3] Information Hiding Based on DNA Steganography , *978-1-4673-5000-6/13/$31.00 2013 IEEE* , by Zicheng Wang, Xiaohang Zhao, Hong Wang and Guangzhao Cui

[4] Message Encryption Using DNA Sequences *2014 World Congress on Computing and Communication Technologies*, by K. Menaka

[5] Automatic DNA Sequence Generation for Secured Effective Multi-Cloud Storage, *IOSR Journal of Computer Engineering (IOSR-JCE)e-ISSN: 2278-0661, p- ISSN: 2278-8727Volume 15, Issue 2 (Nov. - Dec. 2013), PP 86-94,www.iosrjournals.org*, by D.Sureshraj, Dr.V.Murali Bhaskaran

[6] Review on Multi-Cloud DNA Encryption Model for Cloud Security ,*Int. Journal of Engineering Research and Applications ISSN : 2248-9622, Vol. 3, Issue 6, Nov-Dec 2013, pp.1625-1628, www.ijera.com* ,by Richa H. Ranalkar , Prof. B.D. Phulpagar

[7] Workflow-based Automation of Next-Generation Sequencing Data Analysis using Cloud Computing Resources , *Journal of Next Generation Information Technology(JNIT) Volume 4, Number 9, November 2013*, by Youngmahn Han, Hyunsik Kim

[8] DNA based Cryptography in Multi-Cloud: Security Strategy and Analysis, *International Journal of Emerging Trends Technology in Computer Science (IJETTCS)*

[9] Multi Cloud Architecture to Provide Data Security And Integrity, *International Journal of Emerging Technology and Advanced Engineering Website: www.ijetae.com (ISSN 2250-2459, ISO 01:2008 Certified Journal, Volume 3, Issue 10, October 2013)*, by Nikhil Dutta, Himanshu Bakshi, Mujammill Mulla.