A Mechanism to ensure secure sharing and accessing the private data files in a cloud shared environment.

Submitted by: Jenefer Selvakumar

Student No: 10121460

A project presented in Partial Fulfillment of the requirements for the degree in

Masters of Science (MSc)

At

Dublin Business School

Under the supervision of

Mr. Michael Gleeson

Word count: 9309(Excluding Bibliography and Appendix)

MSc.Information Systems with Computing **August 2015**

Declaration:

Declaration: I, **Jenefer Selvakumar**, declare that this project is my original work and that it has never been presented to any institution or university for the award of Degree or Diploma. In addition, I have referenced correctly all literature and sources used in this work and this work is fully compliant with the Dublin Business School's academic honesty policy.

Signed: _____

Date: _____

Acknowledgements

First and foremost I would like to thank my very supportive supervisor Mr. Michael Gleeson for his immense support, advice and guidance throughout the project. His advice, help and feedback throughout the project had been a great support from the start till the end of my project. Also he helped me in the right track throughout the project completion.

I want to thank all the lectures, staff and library staff at Dublin Business School who have contributed to finish my master's project.

Also, I would like to extend my thanks to all the participants of my survey who contributed largely in conducting my research.

Finally, the biggest acknowledge to my parents and friends for their continuous encouragement and support during the entire course of the project.

ABSTRACT

Cloud Computing has been envisioned as the next-generation architecture of IT Enterprise. It moves the application software and databases to the centralized large data centers, where the management of the data and services may not be fully trustworthy. This work studies the problem of ensuring the integrity of data storage in Cloud Computing. With cloud storage services, it is common place for data to be not only stored in the cloud, but also shared across multiple users. However, preserving and0 assuring security to data such as the public and private files stored in the cloud remains to be an open challenge. The security of data in cloud storage, however, is subject to malicious threats, as data stored in the cloud can easily be lost or corrupted due to the inevitable hardware/software failures and human errors.

The purpose of this research is to analyze the major security threats and find out better way to ensure security of data stored in the cloud shared environment. To identify any algorithm or technique that can be implemented to enhance the security measures. A primary research will be carried out in order to find the different ideas and suggestions in order to secure the files that are stored in the cloud shared environment and additional security compliance policies that can be followed. The findings from the research will be summarized and used to build and artifact to provide a working model of the identified solution.

Table of contents

LIST OF FIGURES

1. Introduction:

Cloud Computing is a computer paradigm where data and services reside in massively scalable data centers in the cloud and can be accessed from any connected devices over the internet. It moves the application software and databases to the centralized large data centers, where managing data and its services cannot be trusted completely. Cloud Computing is also called as "Internet based Computing". The Internet is usually considered as clouds, hence the term "Cloud Computing" for computation done through the Internet *(M. Abirami et al., 2013)*.In the simplest terms, Cloud Computing means storing and accessing data and programs over the Internet instead of our computer's hard drive. Cloud Computing is rapidly gaining popularity because Cloud service providers offer users efficient and scalable data storage services with a much lower marginal cost than traditional approaches. It has the ability to make use of computing resources with minimal costs and at high speed. Also cloud users are not required to invest more on infrastructure, hardware and software license. There are also other benefits such as scalability, reliability and efficiency. The scalability factor provides unlimited use of its storage capacity and also it's processing. Cloud Computing is efficient in the way it provides the users with free resources to be used online for development of their business. It is said to be reliable because it maintains data integrity and secure the information stored in the cloud by implementing various security mechanisms. The benefits of Cloud Computing are not limited to scalability, efficiency and reliability it also provides other benefits such as availability and flexibility.

The Cloud Computing offers its benefits through three types of services or delivery models namely Software as a Service (SaaS), Platform as a service (PaaS), and Infrastructure as a service (IaaS).*(M. Abirami et al., 2013)*

Infrastructure as a service is otherwise referred as HaaS or hardware as a service. Infrastructure as a service includes different components such as storage, hardware, servers and network. IaaS provides user computing resources and storage comprised with many servers as an on demand use service. PaaS model provides a platform for creating applications. PaaS solutions are the development platforms for which the development tool itself is hosted in the Cloud and accessed through a browser .With PaaS, developers can build Web applications without installing any tools on their computers. PaaS includes all stack components such as hardware, the infrastructure and storage together with the database, security, user interface, and other tools that allow users to create business applications, web sites, and mobile apps. In the SaaS model, a client server approach is used where the application software is installed in the cloud server and cloud users access the software from client machines. SaaS can be defined through five key

ideas such as services are fully managed and hosted, have regular recurring payments, allow for anytime and anywhere access, have multiple tenants on servers, it does not require installation of specialized software *(M. Abirami et al., 2013).*

In this project the SaaS model will be used so that a cloud server is used to store data files.

Cloud also provides services through four different deployment models such as public cloud, private cloud, community cloud and hybrid cloud.

Public Cloud Computing environment are open for use to anyone who wants to sign up and use them. A private cloud considered when an organization needs more control over their data than they can get by using a third party vendor hosting the cloud service. A hybrid cloud combines both public and private cloud models *(M. Abirami et al., 2013)* .

1.1 Problem Area:

Although Cloud service providers offer users efficient and scalable data storage services with a much lower marginal cost than traditional approaches, cloud is common place for data to be not only stored in the cloud, but also shared across multiple users. However, preserving and assuring security to data files stored in the cloud remains to be an open challenge.

1.2 Aim of the Project:

The aim of the project is ensure secure access and integrity of the private data files in a cloud shared environment.

The objectives of the project are,

- To create a front end application using HTML code where data files can be browsed, uploaded and downloaded.
- To build a working artifact using JAVA, JSP where an uploaded file (Single file) can be split into three different parts using a secure erasure code technique. The Split files are then encrypted/decrypted using DES algorithm.
- To use online cloud environments to host the three different split files that are encrypted and decrypt the file to original plain text

By providing enhanced security to files stored in cloud share environment the users of cloud can rely on cloud server to store their confidential and private data files.

1.3 Purpose of research:

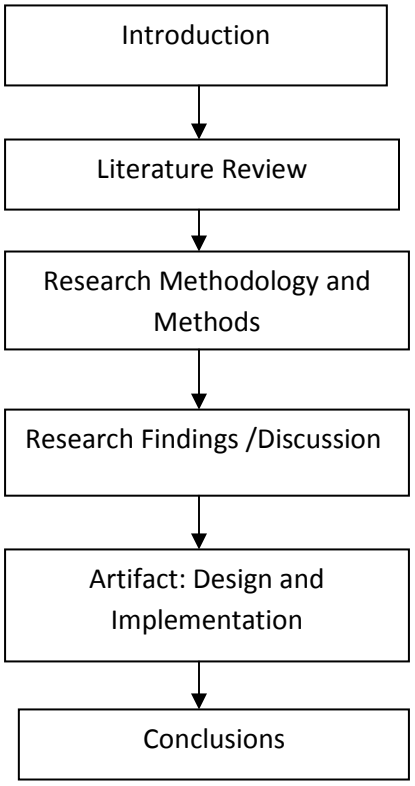
The purposes of this research are as follows,

1. To analyze the user's point of view about their opinion about choosing cloud services based on the different characteristics such as space, integrity, security and cost.

2. To analyze about the security threats to data stored in cloud which is considered to be most important from user's point of view.

3. To analyze the user's opinion to provide better security measures to ensure secure file storage in the cloud shared environment.

1.4 Project Layout:



The introduction presents detailed explanation about the area of Cloud Computing and its services, problem area, purpose of the research and aim of the project.

The literature review section provides a review of the research literature. The section focuses on the area of Cloud Computing explaining the delivery models, the security issues in cloud, the multi cloud approach, use of an encryption mechanism and secure erasure technique to overcome the security threats for the files stored in cloud.

Research methodology section explains and justifies the selected research approach, research strategy and data analysis approaches for the research carried out. This also explains about data collection and analysis, the online survey conducted and the results interpreted through charts and numbers. The inference from the survey results is also discussed.

Research findings and conclusion section is to discuss about key findings and analysis of data collected from the online survey.

The Artifact design and implementation section is to explain in detail about the requirements identified to build the Artifact such as functional and non functional requirements. The design phase explains and represents the Artifact through different UML diagrams. The development phase outlines the important parts of code and its working. The details about the testing carried out on the built Artifact are also explained. The implementation section explains about how the code was executed and implemented in the cloud environment.

The conclusion section explains the summary of the research analysis, Artifact built and research findings. This section also provides recommendations to the future research.

2. Literature Review:

In this section the detailed review of literature is carried out explaining the Cloud Computing and its services, the security issues of data files stored in cloud are discussed. The multi cloud approach, the different security attacks to cloud server, encryption mechanism and secure erasure code technique are discussed in detailed in order to overcome the security threats identified.

2.1 Cloud Computing:

Cloud Computing is a general term is also called as network based computing that takes place over the Internet. The most basic and important service offered by cloud is data storage. With cloud storage services, it is common place for data to be not only stored in the cloud, butdata can also be shared with multiple users in the cloud environment. However, preserving and assuring security to data stored in the cloud remains to be an open challenge. More research has been conducted in order to secure data and files

stored in the cloud. Multi cloud architecture one of the approach that is adopted in order to enhance the security features for the files and data stored on the cloud *(Bohli et al., 2013).*

The Cloud Computing offers its benefits through three types of services or delivery models namely Software as a Service (SaaS), Platform as a service (PaaS), and Infrastructure as a service (IaaS).

Infrastructure as a service is otherwise referred as HaaS or hardware as a service. Infrastructure as a service includes different components such as storage, hardware, servers and network. IaaS provides user computing resources and storage comprised with many servers as an on demand use service. PaaS model provides a platform for creating applications. PaaS solutions are the development platforms for which the development tool itself is hosted in the Cloud and accessed through a browser .With PaaS, developers can build Web applications without installing any tools on their computers. PaaS includes all stack components such as hardware, the infrastructure and storage together with the database, security, user interface, and other tools that allow users to create business applications, web sites, and mobile apps. In the SaaS model, a client server approach is used where the application software is installed in the cloud server and cloud users access the software from client machines *(Bohli et al., 2013).*

In this project the SaaS model will be used so that a cloud server is used to store data such as the private and public files. These files will be hosted to the cloud server environment from a front end application.

The cloud shared environment will contain both private and public files. The public files are the files that are open to all the users sharing the same cloud environment. Private files are the files that can be shared only to specific users in the cloud whom the owner of the files wishes to share. The Erasure correcting code will be applied to these private files

Also the integrity of data in cloud storage, however, is always a question as data stored in the cloud can easily be lost or corrupted due to the hardware/software failures and human errors

2.2 Cloud Security issues:

Cloud Computing has huge number of security threats. According to the *IEEE transaction Dependable and Secure Computing*, a major incident in the SaaS cloud has happened in 2009 with the Google docs. Google docs generally allow the users to edit the documents. But a problem occurred in this system, if a document was shared with anyone, it was accessible by everyone whom the document owner had never shared the files previously. This leads to the unauthorized access to the confidential data *(Bohli et al., 2013).*

2.2.1 Privacy and security:

The fundamental factor of defining the success of the computing technology like cloud relies on how much secure and safe it is to store data. There is a question whether storing data in our local system hard drive is safe or storing it in a cloud is safe. Data stored in hard drives can be accessed whenever we wish to but data stored in cloud servers could potentially reside anywhere in the world and any kind of security threats like an internet breakdown might affect accessing the data. The cloud service providers usually say that their servers and the data stored in their servers is sufficiently protected from any sort of intruding and data loss. However, there are instances where the security is been compromised. It is also a part of cloud architecture, that the client data will be distributed over these individual computers regardless of where the base repository of data is stored. With respect to Cloud Computing environment, is defined as *"the ability of an entity to control what information it reveals about itself to the cloud, and the ability to control who can access that information"*. In case of a public-cloud computing environment, we have multiple security issues that need to be addressed in comparison to a private cloud computing environment *(Bhadauria et al., 2014).*

2.2.2 Different types of security attacks to Cloud:

The primary concern in cloud environments is to provide security and data integrity to the customers who trust cloud environment and store their data files in their cloud servers.

There has been survey works reported that classifies security threats in cloud based on the nature of the service delivery models of a cloud computing system. Service delivery model is one of the many other aspects that need to be considered for a comprehensive survey on cloud security. Security at different levels such as Network level, Host level and Application level is necessary to keep the cloud up and operate continuously. In accordance with these different levels, various types of security breaches may occur. Few of them are outlined below. *(Bhadauria et al., 2014).*

SQL injection attacks, are the one in which a malicious code is inserted into a standard SQL code and the attackers gain unauthorized access to a database and gains access to sensitive information.

Cross Site Scripting (XSS) attacks, which inject malicious scripts into web contents have become quite popular since the inception of Web 2.0. It is based on the type of services provided a website can be classified as static or dynamic. Static websites do not suffer from the security threats which the dynamic websites do because of their dynamism in providing different services to the users. As a result, these dynamic websites get compromised by XSS attacks *(Bhadauria et al., 2014).*

Man in the Middle attacks (MITM). This type of attack is quite popular in the SaaS infrastructure. In this attack, an intruder tries to intrude in an ongoing conversation between a sender and a client to inject false information and to have knowledge of the important data transferred between them *(Bhadauria et al., 2014).*

DNS attack .A Domain Name Server (DNS) server performs the translation of a domain name to an IP address. Since the domain names are very easy to remember, the DNS servers are required to translate an IP address to a domain name. But there are cases when having the server is given a name, the user has been routed to some other malicious cloud instead of the one the user requested for and compromising the security of the domain *(Bhadauria et al., 2014).*

Denial of Service attack. A DoS (Denial of Service) attack is preventing the authorized users of the system from using the system that they have privilege to use. In such an attack, the server providing the service is flooded by a large number of requests and hence the service becomes unavailable to the authorized user *(Bhadauria et al., 2014).*

The Cloud Computing and its services contain an implicit threat of working in a compromised cloud environment. If an external hacker or an attacker is able to intrude into the cloud system itself, all data and files including all the other processes of all users operating on that particular cloud environment may become subject to malicious threat. Hence, the Cloud Computing and its services require an in-depth reconsideration on the security measure that can be taken to meet such situations. In the case of a single cloud provider, hosting and processing all of its user's data, an intrusion or a security threat would immediately affect all its security requirements such as accessibility, integrity, and confidentiality of data and processes may become violated, and further malicious threats may be performed on behalf of the cloud user's identity*(Bohli et al.,2013).*

These Cloud Computing services and its security issues and other challenges has lead to many research activities, resulting in a large number of proposals targeting the various cloud security threats. However, with these security issues, the cloud services has evolved with a new set of unique features that creates a path towards novel security approaches, techniques, and architectures. One promising concept makes use of multiple distinct clouds simultaneously *(Bohli et al.,2013).*

2.3 Multi Cloud Approach:

The basic idea is to use multiple different clouds at the same time is to handle the risks of malicious data manipulation threats, disclosure, and process tampering. When multiple clouds is used data is stored

across different cloud environments where it becomes difficult for an external hacker to integrate all data together from different cloud environments and get the valid data *(Bohli et al.,2013).*

This idea of using a multiple cloud environment was proposed by Bernstein and Celesti, but the main concentration was not on security. Later in time different security measures was proposed. These methods use different operating cloud environments service levels and it is integrated with cryptographic algorithms to meet different requirements *(Bohli et al.,2013).*

In a multi cloud approach data is partitioned into different parts and then stored in different cloud storage operating on different cloud environments. If a system application is partitioned into different tiers it allows separating the logic from the data, so it gives protection against data leakage due to several application system flaws. Also when application logic is fragmented into different partitions it allows distributing the application logic to different clouds. This basically has three main benefits one is where the application logic is not known to any of the cloud provider and second is the cloud provider cannot learn anything about the overall result of the application. Third, the cloud provider cannot gain access to data. This provided enhanced data and application confidentiality (*Bohli et al., 2013)*

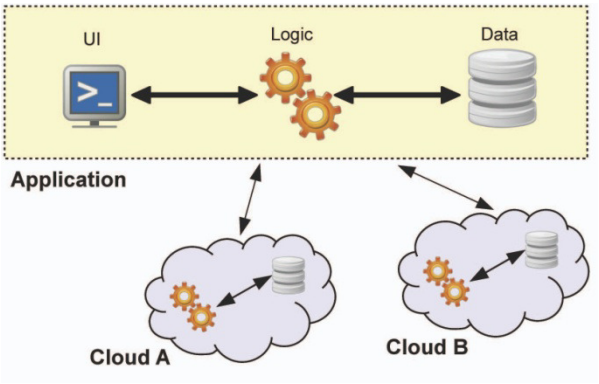


Figure 1.1: Multi cloud architecture*(Bohli et al.,2013).*

2.4 Secure erasure code technique:

A secure Erasure coded data technique is used to fragment the data and store across different locations in the cloud. Using this technique the time and overhead to reconstruct data is reduced. Also this method reduces the redundancy and addresses the space issues in the cloud environment *(Hsiao-Ying Lin et al.,2012).* In addition to this a security mechanism such as encrypting data files using encryption

techniques such as AES/DES will provide more security to access, share and retrieve the private data files across the shared cloud environment. Even though there are many investigation and researches carried out in encrypting data files in the cloud using different encryption mechanisms, in this project a file fragmented using the erasure coded data or erasure correcting code will be encrypted using DES algorithm and a byte value key will be shared to the specific users and that can be used to access, retrieve and modify the private data files from cloud shared environment.

Enhanced Erasure code security Mechanism:

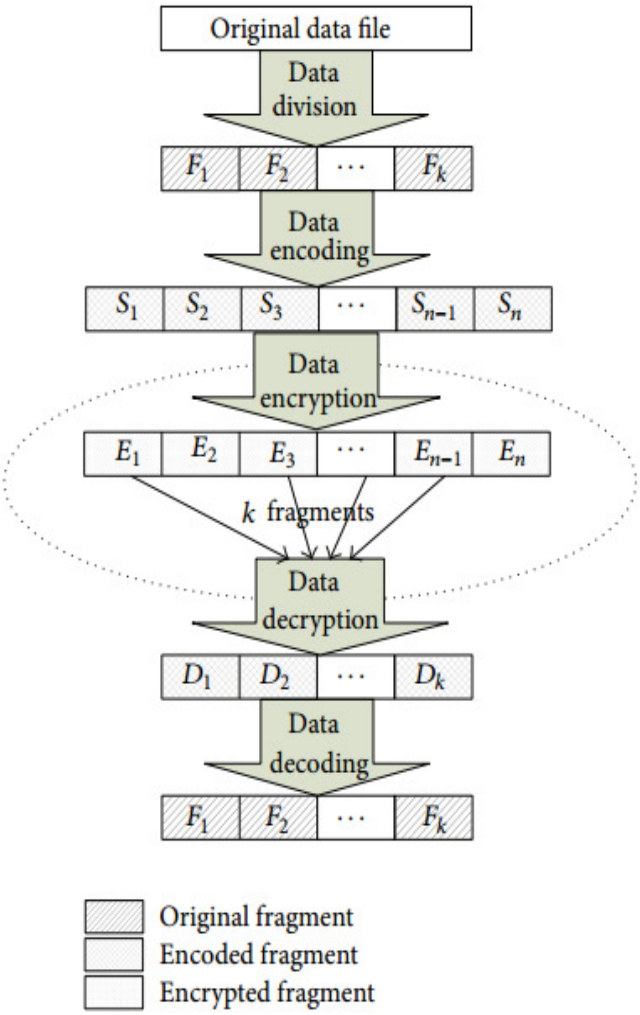


Figure 1.2*(Wenfeng Wang et al.,2014)*

The above design philosophies of the enhanced erasure code-based security mechanism can be described as below,

Step1. Divide the original data file into fragments (1, 2...).

Step2. Encode these fragments into fragments (1, 2... +1,...).

Step3. Encrypt the above fragments into encrypted ones (1, 2,...,, +1,...,) and distribute them to different storage nodes.

Step4. Take any fragments out of the encrypted ones and decrypt them into fragments (1, 2...) in plain text.

Step5. Reconstruct the original data file by using step 3 *(Wenfeng Wang et al.,2014).*

2.5 Encryption:

The DES (Data Encryption Standard) algorithm is most widely used encryption algorithm. The algorithm uses a 56 bit key to encrypt or decrypt a 64-bit block of data. The DES algorithm is a symmetric block cipher which was developed by IBM *(Cryptographyworld.com)*

It is a cryptographic algorithm used to encrypt a plain text.DES is a block cipher which operates on the plain text file with a size of 64 bits and it returns the cipher text of the blocks with the same size.DES works on a permutation and substitution principle *(Page.math.tu-berlin.de)*

DES algorithm uses a private key which is used to encrypt and decrypt a message. Both the sender and the receiver share the same private key. As mentioned previously, the data encryption standard is a block cipher, the algorithm is applied to block of data rather than one bit at a time. Using this algorithm a plain text is encrypted and grouped into 64 bit cipher text by permutation and combination. The process involves 16 rounds totally. Each cipher block is encrypted separately. Decryption process is the inverse of the encryption process that is performed (*Search Security, 2015).*

In this project DES encryption algorithm is used and the algorithm is applied to encrypt the files that are fragmented using the secure erasure code technique. This process will provide enhanced security to the files stored in the cloud environment.

3. Research Methodology:

The main purpose of this research is to analyze and examine the security issue that exists on storing data files in the cloud shared environment and to implement a secure erasure code technique which provides more security to the file and incorporate encryption mechanism in order to enhance the security of files. The research methodology also outlines the survey methods used in data collection process.

3.1 Research Design:

Research design is a method to specify the approach and needs of data to solve the research problem. It is classified as descriptive research, experimental and exploratory research. Descriptive research is used to provide an accurate description of observations. A descriptive research can be done using observation, case study and survey. Exploratory research is a process of performing a literature search by conducting focus group interviews. By gathering more information about the specific areas of study it will provide a better understanding of the research subject. This type of research is mainly used to identify the key variable and functions about the research subject (*Isites.harvard.edu, 2015*).

In this project a descriptive research is carried out to provide accurate description of observations. An online survey is conducted with people from different backgrounds. Conducting survey is the main approach in this project to collect the primary data in quantitative research. Data obtained through the survey contribute to the research. The data provides detailed information about people views on Cloud Computing and its security issues .Also it provides information about their preference of using an encryption mechanism applied to data files stored in the cloud and using a multi cloud approach.

The survey questions are focused on different areas such as,

- The user's preference to use cloud based on different criteria's
- Issues that are identified as major security threats
- User's opinion of using an encryption mechanism and what they think or predict about Cloud Computing as the future of IT industry.

An Exploratory research is carried out in order to perform more literature search about the secure erasure code technique its advantages, disadvantages and its implementation. Also the exploratory research is used to identify and define best methods of encrypting data in order to provide enhanced security to data files store in the cloud environment.

3.2 Research Approach:

There are two main approaches inductive approach and deductive approach. An inductive approach is used to generate a new theory through analyzes and data collection. The new theory will be created through the analysis based on the existing theory. A deductive approach is mainly aimed to test the theory. The existing theory is used to conduct the research and narrow the scope of study.

A deductive approach was followed in this project as this research is based on the existing theory. There are many research conducted in the areas of Cloud Computing. This research mainly focuses on the cloud data, security issues and choosing the best options in order to protect data stored in the cloud from security threats. This approach is used to explore more information about the secure erasure technique which is one of the options identified to be a secure way of saving the files in the cloud.

3.3 Research Strategy:

A Quantitative research is a structured process to get statistical results in order to explain any particular theory or technique and collection of data to support the hypotheses. An experimental design is used to test the attitudes and it is carried out both before and after the experiment is implemented. The data collected on an instrument that measures attitudes and the information is analyzed using statistical procedures and hypothesis testing. Data collected is in the form of numbers. Analysis of data collected can be done in the form of tables or charts and the results are compared. A qualitative research is used to identify and describe the individual attitudes, perceptions, views and beliefs about the subject area. Data collected using qualitative research is in the form of words from documents, observations, and transcripts. The analysis will be done by finding out the themes from the evidence and organizing data to present coherent hypotheses *(Creswell, 2003).*

A Qualitative approach is used in this project in order to collect data in the form of numbers. An online survey is conducted and the responses are interpreted through numbers.

3.4 Research Ethics:

Research ethics is about analysis of ethical issues that comes up when people are involved as participants in research. Research ethics briefs about the guidelines and core principles to be followed during the research .There are three core principles to be followed as part of research ethics. The first principle is to respect human participants. The second objective is to ensure that research is conducted in a way that it serves interests of individuals, groups and society as a whole, to minimize the risk and maximize the benefits of the research study. Finally, the third principle is to examine specific research activities and projects looking at issues such as the management of risk, protection of confidentiality and the process of informed consent. It is up to the individuals or the participants to decide whether to take part in the research or not (*DBS Ethical principles*).

Certain steps are followed in this project as part of research ethics

- Detailed online study was carried out with different research papers in the IEEE forum in order to gain more knowledge about the area of Cloud Computing and its previous research papers. Information used from those papers are addressed and referenced in this project.
- The link to the online survey was sent through email and also through other social media. Also, the questions of the survey were framed in such a way that they do not harm the respondents personally or professionally.

3.5 Data collection and Analysis:

Data collection is important aspect of research which can be used to collect information regarding the research subject and the information gathered can be used to analyze the research problem and provide possible solutions. There are different types of data collection methods such as observation, interviews, focus groups, case studies and surveys.

A survey is a process where sample of respondents is identified from a large population and a standard questionnaire is prepared and the results are gathered. A survey can be conducted online, face to face or through telephone. In this project an online survey is conducted.

3.5.1 Primary Data:

The primary data was collected from the survey. The responses of the survey are recorded and used as primary data.

The responses are interpreted in the form of charts below.

The Survey consists of 10 different questions. The questions were in general related to Cloud Computing, its security threats and overcoming the security threats through encryption mechanism and splitting the files.

The survey was conducted in July 2015.

Charts:

The charts represent the graphical format of the responses from the survey

Chart 1:Cloud Computing and its awareness

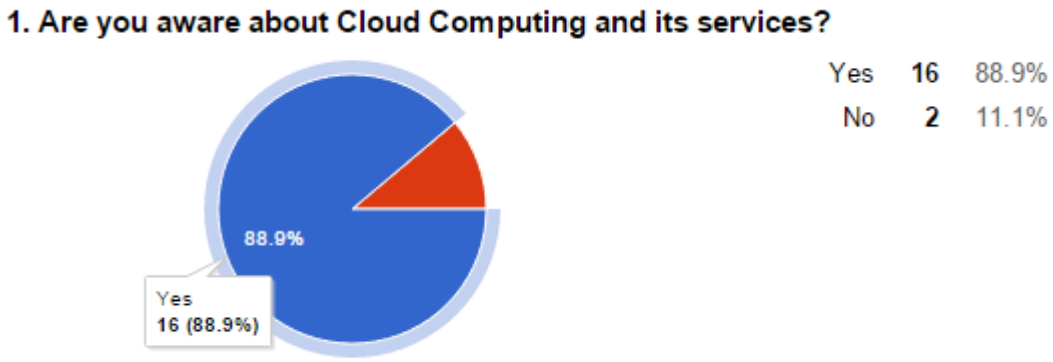


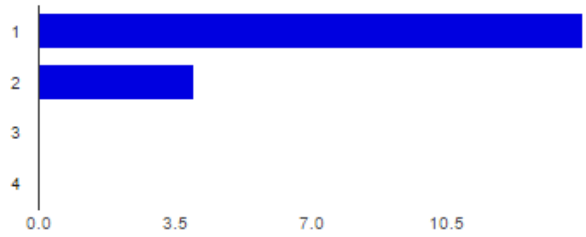
Figure 1.3

Inference:

Major group of people who took this survey were aware about the Cloud Computing and its services. The response from the people who selected that they are not aware about the cloud and its services are considered as invalid data for this project.

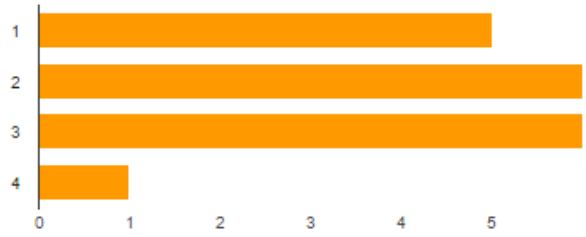
Chart 2: The choice of Cloud Computing based on the order of importance of its major characteristics

Security [2. Rate the following in order of importance based on how you would choose a cloud service.(important.]



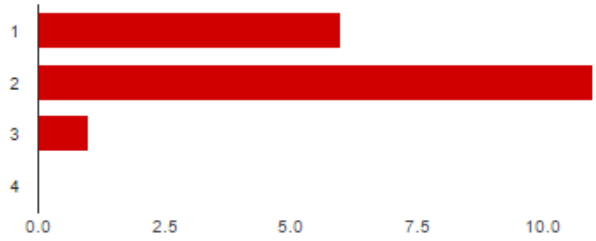
1	14	77.8%
2	4	22.2%
3	0	0%
4	0	0%

Cost [2. Rate the following in order of importance based on how you would choose a cloud service.(On t important.]



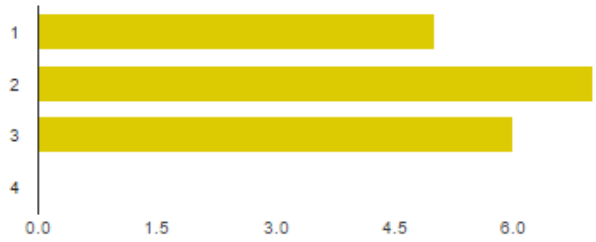
1	5	27.8%
2	6	33.3%
3	6	33.3%
4	1	5.6%

Integrity [2. Rate the following in order of importance based on how you would choose a cloud service important.]



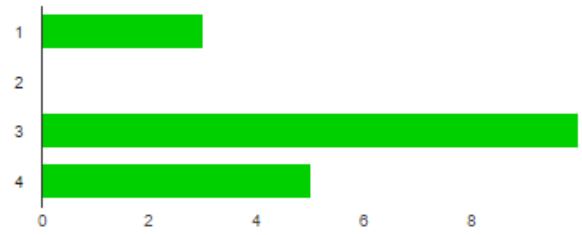
1	6	33.3%
2	11	61.1%
3	1	5.6%
4	0	0%

Space [2. Rate the following in order of importance based on how you would choose a cloud service.(C important.]



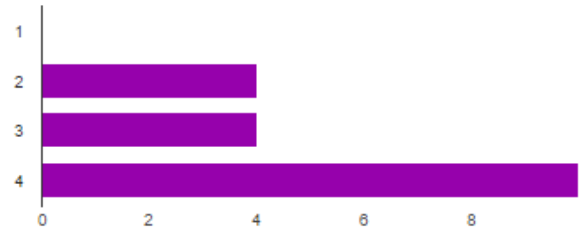
1	5	27.8%
2	7	38.9%
3	6	33.3%
4	0	0%

Accesibility [2. Rate the following in order of importance based on how you would choose a cloud service least important.]



1	3	16.7%
2	0	0%
3	10	55.6%
4	5	27.8%

Flexibility [2. Rate the following in order of importance based on how you would choose a cloud service.(important.]



1	0	0%
2	4	22.2%
3	4	22.2%
4	10	55.6%

Figure 1.4

Inference:

The above rating scale is from 1 to 4 where 1 is considered to be the highest and 4 is the lowest. Majority of the people have considered Security, cost space and integrity as the highest importance whereas accessibility and flexibility are less importance when compared to the above specified characteristics. Security, cost, space and integrity are well known characteristics of the Cloud Computing. The artifact built in this project ensured security of data that are stored in the cloud shared environment.

Chart 3: Preference to Multi Cloud approach:

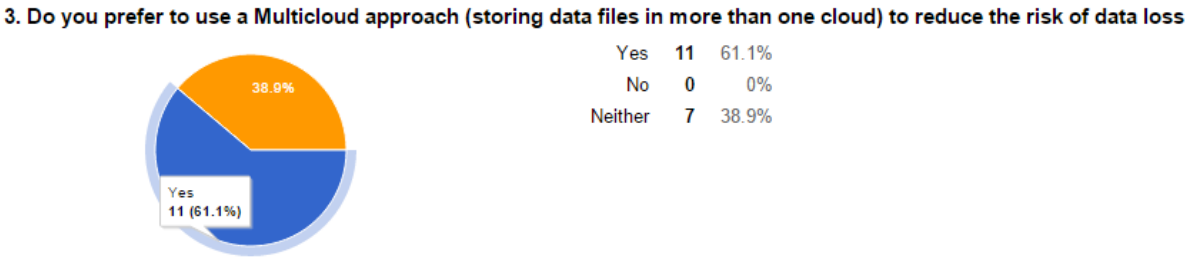


Figure 1.5

Inference:

From the responses most of them have opted to use a multi cloud approach where the single file can split into three different parts and each file can be hosted separately in three different clouds in order to protect the files from data breach. Even if one file is hacked there will be a data dependency to other file which is part of the original file.

Chart 4:Best Security compliance to protect files in the cloud share environment

Figure 1.6

Inference:

From the responses above majority of the people have selected encryption as the best security compliance policy in order to store files and data in the cloud shared environment. The artifact built in this project has an encryption mechanism implemented in it in order to assure security to data that is hosted in the cloud environment.

Chart 5: Choice of using an encryption mechanism

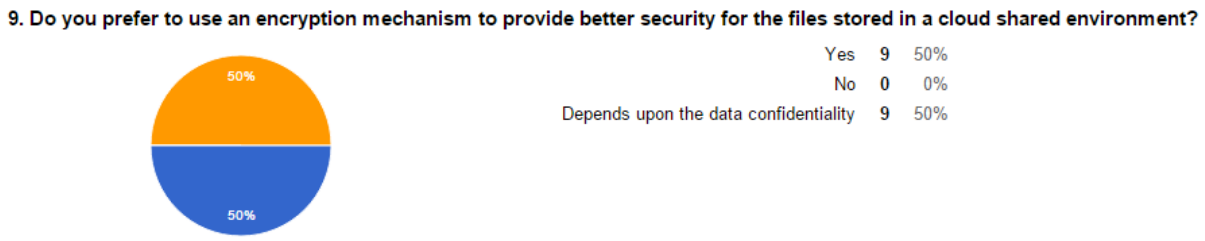


Figure 1.7

Inference:

From the above response equal amount of people have provided yes as they prefer to use an encryption mechanism and other set of people have said that it depends upon data availability whether it is highly confidential or not.

Considering Cloud Computing as future in the IT industry and respondents were asked to provide their choice whether they agree or disagree and justify their choice .Most of them have agreed the reasons are as follows.

- Provides better business services
- I think yes because we have started using cloud services in our day today life to manage our documents and files .Though this is a small level of use definitely this will spread widely and future business will run on cloud storages
- Cloud Computing is been adopted in many start up business reducing their cost of storage, and provides high volume of storage capacity.
- Not necessarily, as in I don't agree that Cloud Computing is the future of IT, provided its security component alone is more robust, various other factors need to be enhanced as well, such as ease of access and data portability.

3.5.2 Secondary Data:

The IEEE papers, documents and the information gathered from research related to Cloud Computing and its security threats are considered to be the secondary data. The data is collected from public external records. The external records include the research papers and journals.

3.6 Research Findings and conclusions:

Based on the primary research results, it is identified that the cloud services requires a secure storage of data as security is considered to be one of the important characteristic of cloud service providers. Along with it the integrity of data needs to be maintained. It is also found that in order to ensure data confidentiality or security a multi cloud approach is preferred much and can be used widely. Data files can be fragmented into different parts and can be stored in different multi cloud environments. So this creates a data dependency to track the original data file even if an external hacker has access to one part of the fragmented data file. A secure erasure code technique can be followed to implement this data fragmentation. To ensure additional security to these files and encryption mechanism is applied to data files that are fragmented. From the primary research most of the respondents has opted for an encryption mechanism in order to provide a better security for data files stored in the cloud environment. Though few of them has responded that it depends on data confidentially, it is acceptable that an encryption mechanism can be applied based on how important and confidential data is used.

Based on the research findings a working artifact can be built, where a data can be fragmented into different parts and then an encryption mechanism is applied to the fragmented files. The encrypted file can be decrypted and the original text file can be viewed. This ensures security which is considered to be one of the most important characteristic to choose a cloud service according to the primary research results. Additionally, the response from the primary data has also marked that an encryption mechanism will be a better security compliance policy to be followed to ensure security to data stored in the cloud. So an encryption mechanism can be applied to the fragmented data files and then stored in the cloud environment.

Finally, Cloud Computing and it services serves the future IT industry is agreed by most of the respondents because of the following reasons,

- Provides better business services
- Cloud Computing is been adopted in many start up business reducing their cost of storage, and provides high volume of storage capacity.

- Yes. Because we have started using cloud services in our day today life to manage our documents and files. Though this is a small level of use definitely this will spread widely and future business will run on cloud storages.

Future research can be done in this area to enhance the current measures that are followed to ensure security and confidentiality to data files stored in the cloud shared environment. Also a distributed cloud approach can be built and implemented where more than one cloud environments can actually be used to store the fragmented data and distinct cloud servers can be used to store data and encrypted for better security.

4. Artifact:

An artifact is built in order to split a file into three different parts using a secure erasure code technique, an encryption mechanism is applied to the files that are split and then the files are stored in the cloud shared environment. This provides better security to protect data files in the cloud.

The artifact has the following functionalities:

1. A home page where a file can be browsed and upload. During the upload process the file is split into three different parts and store in the given directory path.
2. Similarly the uploaded file can be downloaded from the homepage using the file download option available in the home page. During this download process the files that are split is decrypted, consolidated to original text file and can be viewed

4.1 Software Development models:

There are different software development models available according to the requirements of the project that is to be developed. The various models are,

1. Waterfall model
2. V model
3. Incremental model
4. RAD model
5. Agile model
6. Iterative model
7. Spiral model

Each model specified above has advantages and disadvantages to develop a project. Selecting a software model to develop a project depends upon the requirements of the project being developed. If the project has no fixed requirements before starting phase then iterative or incremental model will be an appropriate choice. When the requirements of the project are fixed before the start of the project then waterfall model can be followed as the development model. Nowadays agile methodology is becoming more popular and used in the industry. Agile methodology is similar to the incremental model, if the project requirements are changing then agile methodology is followed as each release is tested. New changes are implemented in every release and tested thoroughly for the quality of the application. The V-model is similar to the above mentioned Waterfall model, the requirements needs to be very clear before starting the project. This is a highly disciplined model where each phase is verified and validated one at a time. We cannot use this model for the main reason it is not flexible to changes and just in case there is a requirement change, which is very common in our system

An iterative approach is followed in this project in order to accommodate new requirements evolving over time of developing the artifact. In iterative development initially the project requirements are not fully specified. Initially a part of the software is developed and then further requirements are identified and developed .The process is repeated producing a new version of the software. A high level design is created initially before starting the project or building an application.

The advantages of using this approach is the defects can be identified and tracked initially which reduces the cost. It helps in building the application step by step.

4.2 Requirement Analysis:

In the requirement analysis phase the functional requirements and non functional requirements of artifact are identified.

The functional requirements are used to define the operation of the artifact on how it should behave. The input and output operational behavior of an application are specified as functional requirements

The non functional is defined as the ability of the system to operate or perform. In general the performance requirements of the artifact is described as non functional requirements

4.2.1 Hardware Requirements:

Processor	:	Minimum -Pentium IV 2.3 GHZ
Hard Disk	:	Minimum -250 GB.

Ram	:	Minimum 1 GB

4.2.2 Software Requirements:

Operating system	:	Windows 7 or Higher
Platform	:	JDK1.6 or above
Languages	:	HTML, JAVA (JSP, Servlet)

4.2.3 Functional Requirement:

1. The home page should contain the button for File Upload and File Download.

2. When File upload button is clicked the page should be navigated to the page where the files can be uploaded

3. The file upload page should contain the browse button and submit button.

4. When the browse button is clicked the menu with list of files should be displayed.

5. When submit button is clicked the file should be selected.

6. A single files when uploaded should be split into three parts and encrypted using an encryption algorithm.

7. The files should be decrypted and received as plain text as the original file when the File download button is clicked in the home page.

4.2.4 Non functional requirements:

1. The navigation from home page to file upload page should be no more than 5 seconds

2. The file upload process should not exceed 2 minutes.

3. The file download process should not exceed more than 4 minutes

4. The data integrity of the file should be maintained.

5. The User Interface should be easily accessible and it should have consistent look and feel.

5. Design:

In the design phase the design of the artifact is prepared from the requirements specification. The design of the system serves as the input for the next phase of the software development life cycle. In the design phase the hardware and the software requirements to build the artifact is identified.

The following are the requirements to build the artifact:

Language: Java, HTML, JSP

IDE:Net beans version 8.0.2

5.1 UML Diagrams:

The UML diagrams are widely used in the design phase of the software development life cycle to represent the model of the artifact and its operation. The unified modeling language is a standard language used to visualize a model in the software development process. In this project this is used to represent the model of the artifact. The UML diagrams used to depict the behavior of the artifact.

5.1.1 Use Case:

The use case diagram are otherwise called as behavioral diagrams which is used to represent the set of actions the system could perform with other components of the system. The use case diagram has the following components such as the actor or the user of the system and the use cases. The use cases are used to define set of action performed by the actor or the user of the system. In this project the use case diagram is used to represent the behavior of the artifact that is going to be designed.

The use case diagram below represents how the artifact will work and also represents the file encryption and decryption process.

The user in the below diagram represents the user of the system and the uses cases represents various actions that takes place between the user and the system. Initially, theuser browses and uploads the plain text file and the files are split into three different partitions using the secure erasure technique. The files that are split into three different parts are then encrypted using an encryption algorithm and a key is generated. The same key is used to decrypt the file and return the original plain text file.

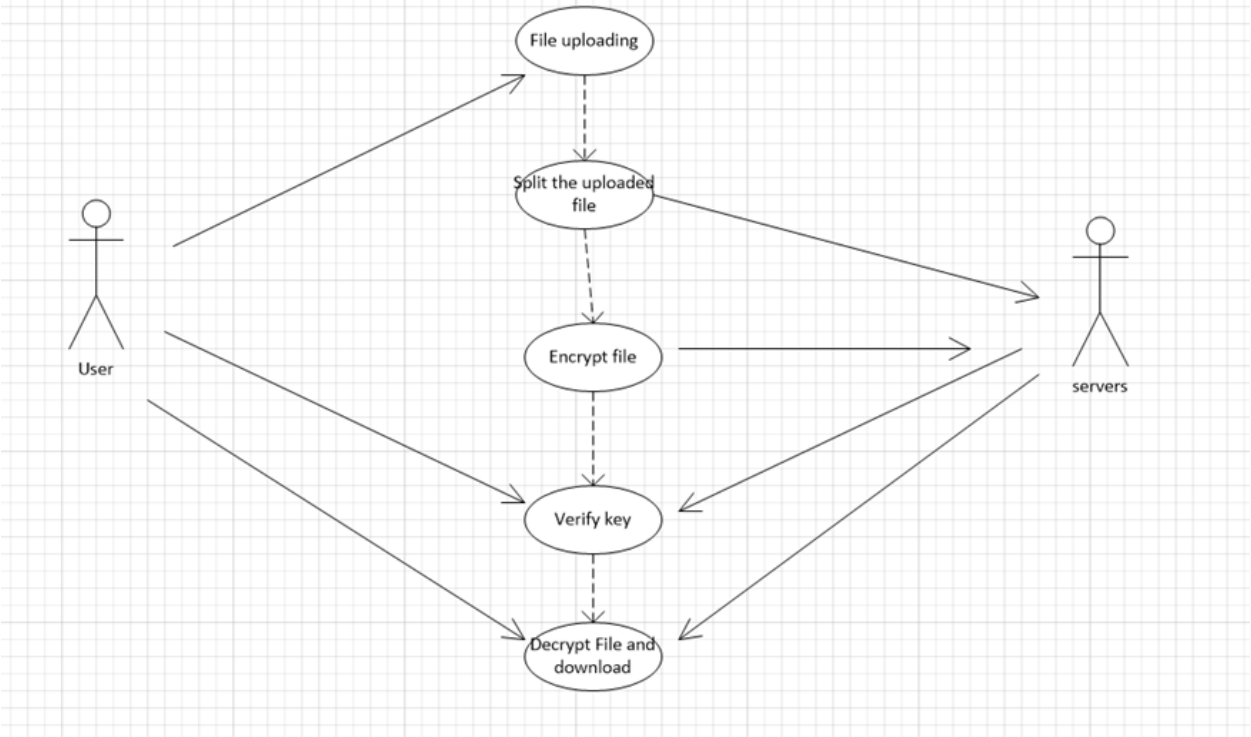


Figure 1.8

5.1.2 Sequence Diagram:

The sequence diagram is also called as an interaction diagram which focuses on the sequence of message interchange between a numbers of lifelines.

The sequence diagram represents the sequence of operations that are carried out in the artifact built in this project.

Initially, the original plain text file is browsed and using the upload option it is uploaded into the system. The file uploaded in the system is first split into three different parts. The second phase is to encrypt the file using an encryption algorithm. Using a DES encryption algorithm the files are encrypted and a secret key is generated. The same key is used to decrypt the file to original text format.

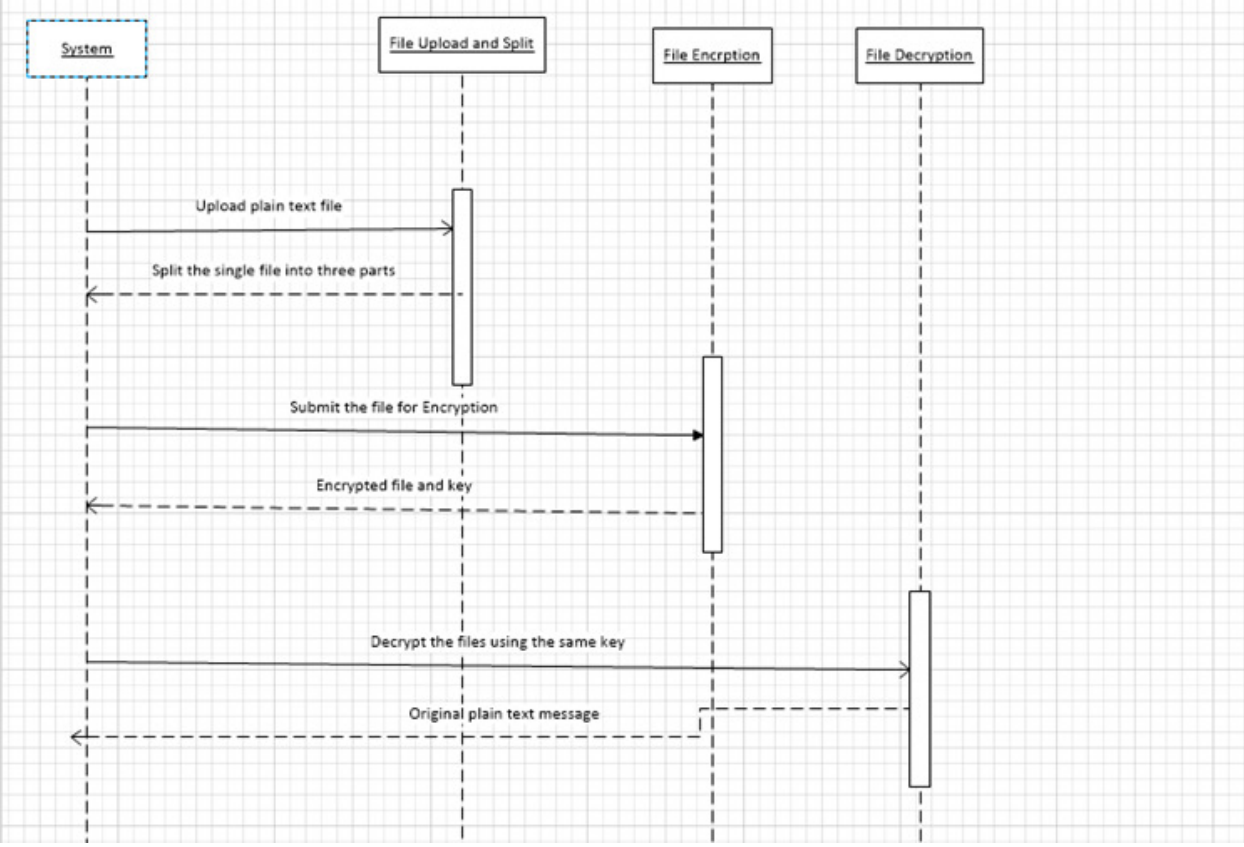


Figure 1.9

5.1.3 CLASS DIAGRAM:

The class diagram shows how the different entities (people, things, and data) relate to each other in the system. In other words, it shows the static structures of the system. A class diagram can be used to display logical classes. Class diagrams can also be used to show implementation classes, which are used to develop the code. A class is depicted on the class diagram as a rectangle with three horizontal sections, as shown in the below figure. The upper section shows the class name, the middle section contains the class's attributes; and the lower section contains the class's operations (or "methods"). The diagram has five main classes which give the attributes and operations used in each class (*Books.google.ie).*

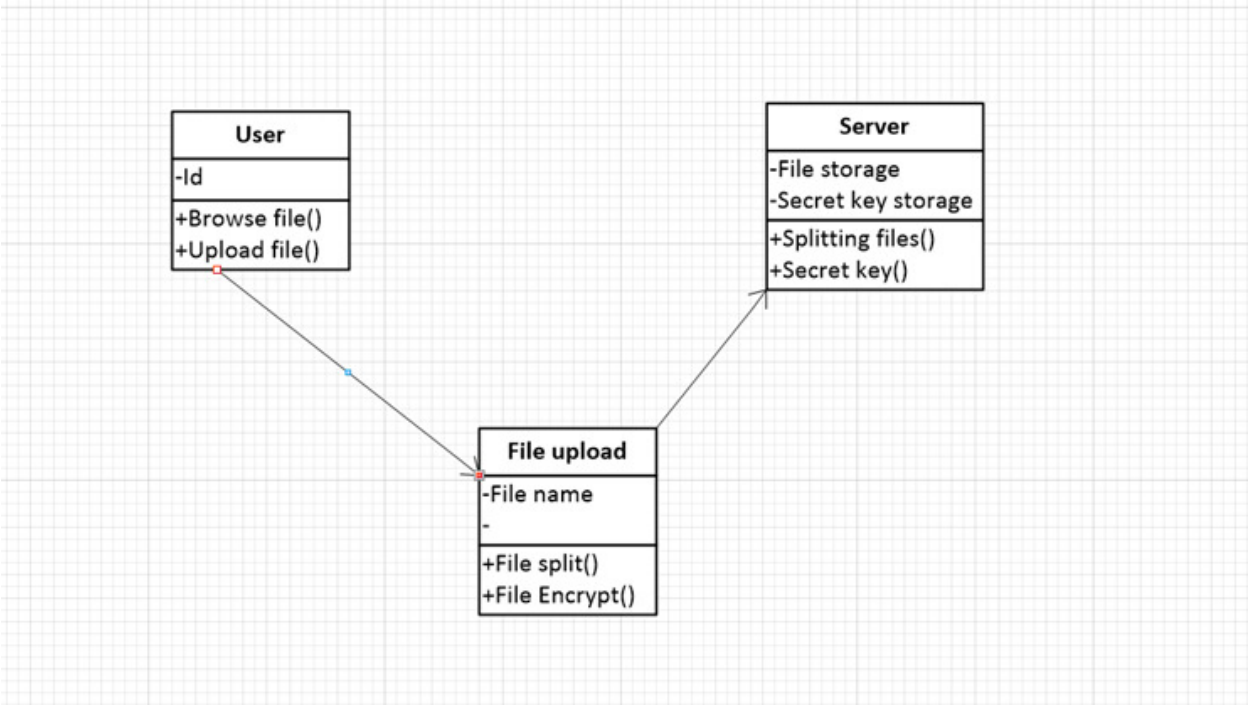


Figure 1.10

5.1.4 COLLABORATION DIAGRAM:

Collaboration diagrams are a technique for defining external object behavior. They include the same information as in sequence diagram (or message trace diagrams) but are better able to show asynchronous message passing. Collaboration diagrams show how objects collaborate by representing objects by icons and their message passing as labeled arrows *Pandey, Anil(2014).*

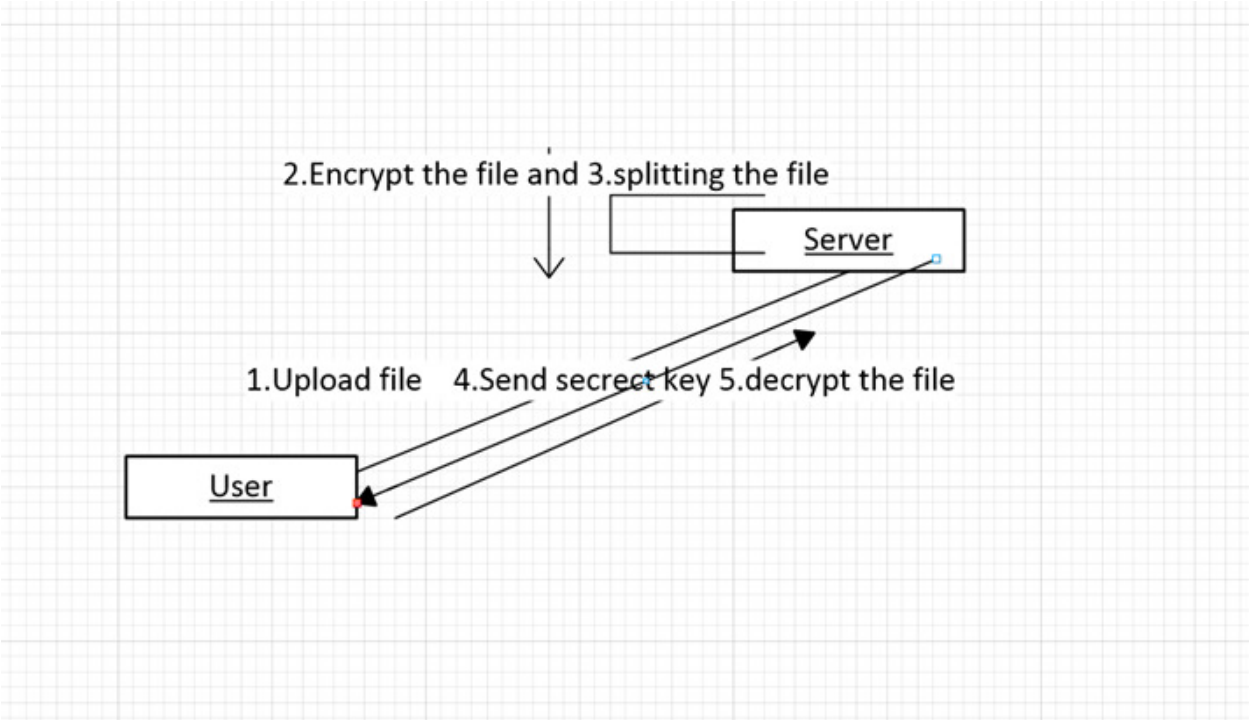


Figure 1.11

6. Development:

After completing the design phase the development phase is started. The requirements are divided into different modules and the coding is done accordingly. The actual coding of the artifact is started. In this project initially the front end page/home page was coded using HTML code. The home page consists of a browse button, upload and download button.

Then the coding was done to split the original plain text file into three different parts using the secure erasure code technique.

Module 1: Splitting the files using secure erasure code technique:

```
char cr[] = new char[si];
                    StringBuffer fid = new StringBuffer();
                //  Random rm = new Random();
                    int fc = 0;
```

```
                    FileInputStream fr = new
FileInputStream(dirName+"/"+flname);



                    DataInputStream dis1 = new DataInputStream(fr);

                    BufferedReader br1 = new BufferedReader(new
InputStreamReader(dis1));

                    for (int j = 0; br1.read(cr) != -1; j++)

                       {

                        StringBuffer sb1 = new StringBuffer();

                        for (int i = 0; i < cr.length; i++) {

                            sb1.append(cr[i]);

                        }

                        if (j == 0)

                           {

                            s1 = fn + (j + 1) + "." + ffmt;

                            File ff1 = new File(dirName1+"/"+fn+(j + 1)+ "."
+ffmt);

                            FileWriter fw = new FileWriter(ff1);

                            BufferedWriter bw = new BufferedWriter(fw);
```

Module 2: Encrypting the split file:

The below lines of code is used to encrypt the split files,

```
FileInputStream fise = new FileInputStream(dirName1+"/"+fn+(j + 1)+ "."
+ffmt);

FileOutputStream fos = new FileOutputStream(dirNameE1+"/"+fn+(j + 1)+ "."
+ffmt);

               InputStream is=fise;

               OutputStream os=fos;
```

```
              String key=”Jenefe123";

 int mode=Cipher.ENCRYPT_MODE; //constant of cipher

               DESKeySpec dks = new DESKeySpec(key.getBytes()); //converting
secret key to byte

            SecretKeyFactory skf = SecretKeyFactory.getInstance("DES");
//creating a new object

            SecretKey desKey = skf.generateSecret(dks); //generating secrete
key Jenefe123 in byteformat

            Cipher cipher = Cipher.getInstance("DES"); //
DES/ECB/PKCS5Padding

            if (mode == Cipher.ENCRYPT_MODE) {

                  cipher.init(Cipher.ENCRYPT_MODE, desKey); //initializing
encryption,deskey(secret key)

                  CipherInputStream cis = new CipherInputStream(fise,
cipher); //actual encryption

                  //doCopy(cis, os);

byte[] bytes = new byte[64];

      int numBytes;

      while ((numBytes = cis.read(bytes)) != -1)

                        {

                  fos.write(bytes, 0, numBytes);

            }

            fos.flush();
```

7. Testing:

The testing phase in the software development life cycle is used to test he developed code against the requirements to make sure that the artifact meets the needs that are addressed during the requirements gathering phase. There are different types of testing such as unit testing, System testing, Integration testing, white box testing, black box testing. The following testing was done in this project,

- Black Box testing
- Unit Testing
- System Testing

7.1 Black Box testing:

Black box testing is also called as functional testing where the test is done against the functional requirements of the artifact specified in the requirements phase is working as expected. The black box testing is done in order to test the below mentioned categories,

- When there is an incorrect functionality or to find a missing functionality
- To find out if there are any interface errors.
- To identify errors in databases or external structures
- To find if there are any initialization and termination errors

Black box testing can be done without knowing the internal structure of the code or component of the system. It has several advantages such as,

- Testing can be done from the end user point of view
- There is no requirement to know the code or the languages used to test the component
- The test cases can be written during the design phase with the gathered requirements.

In this project the artifact is tested against the requirements that are specified under the functional requirements in the design phase.

7.2 Unit Testing:

The unit testing is done to validate small unit of the component is working as expected. *In procedural programming a unit may be an individual program, function and procedure. In object-oriented programming, the smallest unit is a method, which may belong to a base/ super class, abstract class or derived/ child class (Softwaretestingfundamentals.com).*

In this project unit testing is carried with the components such as splitting the files, encryption of file and decryption.

The major advantage of unit testing is the smaller parts of the code is tested earlier before deployment so the defect can be identified for any new changes that are implemented in the code.

7.3 System Testing:

The system testing is use to test the complete integrated software. The system testing is carried out in order to verify that the requirements specified are been satisfied.

The artifact of this project is system tested in order to ensure the complete workflow of the system is working as specified in the requirements.

Sample test cases are provided below.

Sno	Test Case Name	Test case Description	Step ID	Test steps	Expected Result	Actual Result
1	Validate whether the home page consists of upload/download buttons	To Validate that the home page/front end page displays the upload, Download and browse buttons	Step 1	Execute the code through the run button in the Net beans IDE	The code should be executed successfully and the home page should be displayed	The code executed successfully and the home page is displayed
			Step 2	Validate the following buttons are displayed in the home page Upload File Download File Browse	The listed options should be available in the home page	The listed options is displayed in the home page
2	Verify that the file can be split into three different parts and encrypted	To verify that when the upload file button is clicked the original file is split into three different parts and encrypted	Step 1	Browse the original text file	The original text file should be browsed	The original text file is browsed and can be viewed
			Step 2	Click the upload File button	The file should be uploaded and split into three different parts. It should be available in the specified directory	The file is uploaded and split into three different parts and it is available in the specified directory path
			Step 3	Verify that the split files are encrypted	The split files should be encrypted and available in the directory specified	The split files is encrypted and it is available in the directory.

8. Implementation/Deployment:

After the coding and testing phase is completed successfully the product is deployed and implemented for use of the end user.

In this project the code is implemented in a Cloud environment called as Jelastic. Jelastic provides PaaS(platform as a service) and IaaS (Infrastructure as a service) services in a cloud environment. The Jelastic itself supports languages such as JAVA and Python. As the artifact in this project is developed using java code this is most suited to deploy the developed code. The code is hosted with the server provided in this platform.

The code is deployed and tested successfully in the Jelastic cloud environment.

9. Conclusions

The artifact delivered as part of this project, is a secure mechanism where the original files can be split into three parts, encrypted and stored in the cloud environment and the stored file can be decrypted and the original text file can be viewed. This was developed using JAVA, JSP and HTML languages. The project was initially started with a literature review which provided information about the splitting up of files using the secure erasure code technique and how to encrypt/decrypt the files. An online survey was conducted in order to identify the major how the users choose to use Cloud Computing based on the order of importance of the characteristics of cloud such as security, integrity and cost. Also the major security threats to files stored in cloud are identified through data collected from the survey. The user's options to use an encryption mechanism which provides better security to the files stored in the cloud are recorded from the survey results. To enhance this security a secure erasure code technique was used to fragment the original file into three different parts and then encrypt the files that are fragmented. This creates a data dependency when a part of the fragmented file is hacked by an external attacker. In this scenario finding the other content of the original file becomes a hard task for an external attacker, so the files are kept secure.

9.1 Future Enhancements:

Future enhancements can be done to this artifact. The suggested enhancement for this project is that the files which are split into three different parts can be placed in a distributed cloud shared environment which increase data dependency and provides enhanced security.

Also, the artifact can be enhanced by adding an admin page where different users can login. The users can be given different privileges such as access to only private files or having access to both private and public files. A database can be set up at the back end to store the credentials of the users.

9.2 Self Reflection:

This section is to outline the development, abilities and skills that I gained during the course of this project.

Planning things before performing was very much useful from the beginning of the project. Appropriate time and effort was put in choosing the topic. The project plan was proposed as part of RM2 and I almost was sticking to the scheduled timeline. By this I improved my time management skills. The route to the dissertation started by submitting the proposal initially and then a supervisor was assigned. The supervisor was of significant help to me. I got guidance and support from my supervisor and in depth details was provided by the supervisor on how to proceed further with the dissertation. The project layout was provided as I was supposed to build an artifact along with the research.

By writing the literature review I gained good knowledge about the subject. I read various journals articles related to the subject of the research. I conducted an online survey and from the results some idea was implemented in my artifact. In this course of learning, I was able to gain specific skills such as communication, task management and analytical skills. During the development phase of the artifact I faced some minor to major issues due to lack of knowledge, but I managed to fix the bugs through the learning process.

10. Bibliography:

- *M. Abirami, N.M. Nandhitha, S.Emalda Roslin,2013* International Journal of Engineering Trends and Technology*, 2013, ,* DBS Library*,* Vol. 4, Issue 3, p.275.
- *Hsiao-Ying Lin Hsinchu, Taiwan ,Tzeng, W.-G ,2012* Parallel and Distributed Systems, IEEE Transactions on 2012,Volume:23 , Issue: 6
- *Wenfeng Wang, Peiwu L, Longzhe Han, Shuqiang Huang, Kefu Xu, Changgui Yu, Jin'e Lei ,2014,* Mathematical Problems in Engineering.2014,pp.1-8.
- *Creswell, J. (2003).* Research Design: Qualitative, Quantitative, and Mixed Methods Approaches. 4th Ed.
- *SearchSecurity, (2015). What is Advanced Encryption Standard (AES),* [online] Available at: http://searchsecurity.techtarget.com/definition/Advanced-Encryption-Standard.
- *Isites.harvard.edu, (2015*). Harvard University.http://isites.harvard.edu
- *DBS Ethical Principles*. Ethical Guidelines for Research with Human Participants & Procedures for Ethical Approval (http://elearning.dbs.ie).
- *Bohli, J., Gruschka, N., Jensen, M., Iacono, L. and Marnau, N. (2013).* Security and Privacy-Enhancing MulticloudArchitectures.IEEE Trans. Dependable and Secure Computing., 10(4), pp.212-224
- *Pandey, Anil(2014).* 'Test Case Generation Technique By Using Collaboration UML Diagram'.http://www.ijari.org/. 2014.
- *Softwaretestingfundamentals.com*,. 'Black Box Testing | Software Testing Fundamentals', 2010.
- Books.google.ie,. 'Google Books', 2015.
- *Istqbexamcertification.com*,. 'ISTQB Exam Certification – Study Material For Foundation Level, Through ISTQB And ASTQB Exam, Certification Questions, Answers, Tutorials And More'.
- *Cryptographyworld.com*. 'Cryptographic Algorithms: Triple DES'.

- *(Bhadauria, Rohit,Chaki, Rituparna,Chaki, Nabendu,Sanyal, Sugata, 2014).*ActaTechnicaCorvininesis - Bulletin of Engineering. Oct-Dec2014, Vol. 7 Issue 4, p159-177. 19p
- *Page.math.tu-berlin.de*. 'The DES Algorithm Illustrated'.
- *Docs.oracle.com*,. 'Inputstreamreader, 'DataInputStream,Java.io,BufferedReader ,IOException ,PrintWriter, Bufferedwriter, Biginteger, Javax.Crypto.Spec, Java.Util, Multipartrequest, Java.Sql, Message digest, FileInputStream (Java Platform SE 7)'.
- *Stackoverflow.com*,. 'Stack Overflow'.
- *Avajava.com,.* Encrypt And Decrypt Files - Web Tutorials - Avajava.Com'.
- *Docs.oracle.com,.* 'Requestdispatcher', ServletException,annotation.WebServlet, http.HttpServlet , HttpServletRequest; HttpServletResponse; HttpSession; (Java EE 6)'.
- *Avajava.com,.*

11. Appendix:

Index.html

```
<html>
<head>
<title> Secure File Storage </title>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initialscale=
1.0">
</head>
<body>
<table align="center" width="50%">
<tr>
<th colspan="2" height="150"><h1>Secure file
storage</h1></th>
</tr>
<tr>
<th><h2><a href="upload.jsp"> Upload File</a></h2></th>
<th><h2><a href="download.jsp">Download File</a></h2></th>
</tr>
</table>
</body>
</html>
```

Upload.jsp

```
<html>
<head>
<title>CloudSecureErasureCode</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<script>
function val()
{
```

```
if(document.up.fname.value==0||document.up.fil.value==0)
{
alert("fill the all field");
return false;
}
}
function val1(status)
{
if(status)
{
document.up.sharusers.value="";
}
document.up.sharusers.disabled = status;
}
</script>
</head>
<body >
<table width="820" border="0" align="center" cellpadding="0" cellspacing="0"
bgcolor="#FFFFFF" style="margin-bottom:10px;">
<tr>
<td width="70" valign="top"></td>
<td width="750" height="37" valign="top"><table width="690" border="0"
cellpadding="0" cellspacing="0">
<tr valign="top">
<td height="154" colspan="4">
<h1 align="center" class="TitreB style13"> Secure Storage Services In
Cloud Computing</h1>
</td>
</tr>
<tr>
<td width="225" height="55" align="center"></td>
<td width="181" align="center"><a
href="index.html">Home</a></td>
```

```
</tr>
<tr>
<td height="257" colspan="4">
<FORM ENCTYPE="multipart/form-data" name="up" ACTION="upanddownJ.jsp"
METHOD="POST" onSubmit="return val();">
<table align="center">
<tr>
<td colspan="4" height="85" valign="top"
class="Box">
<table width="618">
<tr><th
height="50">FileUploading</th><td> <input type="file" name="file"></td></tr>
<tr><th height="50"><input
name="submit" type="submit" ></th>
</tr>
<tr>
<td colspan="2">
<%
String succ=request.getParameter("msm");
if(succ!=null)
{
%>
<h3 style="color: green">File Uploaded Successfully....</h3>
<%
}
%> </td>
</tr>
</table> </td>
</tr>
</table>
</form>
</td>
</tr>
```

```
</table>
</td>
</tr>
</table>
</body>
</html>
```

UpanddownJ.jsp

```
<%@page import="java.math.BigInteger"%>
<%@page import="java.security.MessageDigest"%>
<%@page import="javax.crypto.*"%>
<%@page import="javax.crypto.spec.*"%>
<%@page import="java.util.*"%>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@page import="com.oreilly.servlet.MultipartRequest"%>
<%@page import="java.io.BufferedWriter"%>
<%@page import="java.io.FileWriter"%>
<%@page import="java.io.File"%>
<%@page import="java.util.Random"%>
<%@page import="java.io.InputStreamReader"%>
<%@page import="java.io.BufferedReader"%>
<%@page import="java.io.DataInputStream"%>
<%@page import="java.io.FileInputStream"%>
<%@page import="java.sql.*"%>
<%@page import="java.io.*"%>
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>JSP Page</title>
</head>
<body>
```

```
<% int a; %>
<%=a=(int) (Math.random() * 1000) %>
<%
try
{
String tkn1="";
String tkn2="";
String tkn3="";
String flname=null;
String paramname=null;
String s1 = null;
String s2 = null;
String s3 = null;
int ii=0;
ArrayList list = new ArrayList();
String fname=null;
int f1=0;
File file1=null;
ServletContext context = getServletContext();
String dirName =context.getRealPath("/OFile/");
session.setAttribute("orginalfiledir", dirName);
String dirName1 =context.getRealPath("/split1/");
String dirName2 =context.getRealPath("/split2/");
String dirName3 =context.getRealPath("/split3/");
String dirNameE1 =context.getRealPath("/splitE1/");
String dirNameE2 =context.getRealPath("/splitE2/");
String dirNameE3 =context.getRealPath("/splitE3/");
MultipartRequest multi = new MultipartRequest(request, dirName,40 * 1024 *
1024); // 10MB
Enumeration params = multi.getParameterNames();
while (params.hasMoreElements())
{
paramname = (String) params.nextElement();
```

```
}
Enumeration files = multi.getFileNames();//reads all fies inside the folder
while (files.hasMoreElements())
{
fname = (String) files.nextElement();
if(fname.equals("d1"))
{
fname = null;
}
if(fname != null)
{
f1 = 1;
flname = multi.getFilesystemName(fname);
out.println("cp1");
String fPath = context.getRealPath("/OFile/"+flname);
out.println("cp2");
file1 = new File(fPath);
out.println("cp3");
FileInputStream fs = new FileInputStream(file1);
System.out.println("FileInputStream fs "+fs);
list.add(fs);
}
}
String ff = request.getParameter("file");
if (flname != null) {
String ss = flname.toString();
System.out.println(flname);
FileInputStream fis = new
FileInputStream(dirName+"/"+flname);
String ffmt = "";
String sss = ss.replace('.', ' ');
String ffs[] = sss.split(" ");
String fn = ffs[0];
```

```
int fln = ffs.length;
if (fln > 0) {
ffmt = ffs[1];
System.out.println(ffmt);
}
DataInputStream dis = new DataInputStream(fis);
BufferedReader br = new BufferedReader(new
InputStreamReader(dis));
String brr;
StringBuilder sb = new StringBuilder();
while ((brr = br.readLine()) != null) {
String[] brl = brr.split("(?!^)");
ii=ii+brl.length;
for (int i = 0; i < brl.length; i++) {
sb.append(brl[i]);
}
ii++;
}
// int si = (sb.length()+366) / 3;
int si=ii/3;
//out.println(si);
System.out.println("SI " + si);
// ------------- split the file.....
char cr[] = new char[si];
StringBuffer fid = new StringBuffer();
// Random rm = new Random();
int fc = 0;
FileInputStream fr = new
FileInputStream(dirName+"/"+flname);
DataInputStream dis1 = new DataInputStream(fr);
BufferedReader br1 = new BufferedReader(new
InputStreamReader(dis1));
for (int j = 0; br1.read(cr) != -1; j++) {
```

```
StringBuffer sb1 = new StringBuffer();
for (int i = 0; i < cr.length; i++) {
sb1.append(cr[i]);
}
if (j == 0) {
s1 = fn + (j + 1) + "." + ffmt;
File ff1 = new File(dirName1+"/"+fn+(j + 1)+ "."
+ffmt);
FileWriter fw = new FileWriter(ff1);
BufferedWriter bw = new BufferedWriter(fw);
bw.write(cr);
bw.flush();
fw.close();
bw.close();
//---------------------- Encrypt------------------------
FileInputStream fise = new FileInputStream(dirName1+"/"+fn+(j + 1)+ "."
+ffmt);
FileOutputStream fos = new FileOutputStream(dirNameE1+"/"+fn+(j +
1)+ "." +ffmt);
// InputStream is=fise;
// OutputStream os=fos;
String key="Jenefe123";
int mode=Cipher.ENCRYPT_MODE; //constant of cipher
DESKeySpec dks = new DESKeySpec(key.getBytes()); //converting secret key to
byte
SecretKeyFactory skf = SecretKeyFactory.getInstance("DES");//creating a new
object
SecretKey desKey = skf.generateSecret(dks); //generating secret key in byte
format
Cipher cipher = Cipher.getInstance("DES");
if (mode == Cipher.ENCRYPT_MODE) {
cipher.init(Cipher.ENCRYPT_MODE, desKey); //initializing encryption,deskey
(secret key)
```

```
CipherInputStream cis = new CipherInputStream(fise,
cipher); //actual encryption
//doCopy(cis, os);
byte[] bytes = new byte[64];
int numBytes;
while ((numBytes = cis.read(bytes)) != -1) {
fos.write(bytes, 0, numBytes);
}
fos.flush();
fos.close();
cis.close();
}
//---------------------- END-Encrypt-----------------------
} else if (j == 1) {
s2 = fn + (j + 1) + "." + ffmt;
File ff1 = new File(dirName2+"/" + fn + (j + 1) +
"." + ffmt);
FileWriter fw = new FileWriter(ff1);
BufferedWriter bw = new BufferedWriter(fw);
bw.write(cr);
bw.flush();
fw.close();
bw.close();
//---------------------- Encrypt-----------------------
FileInputStream fise = new FileInputStream(dirName2+"/"+fn+(j + 1)+ "."
+ffmt);
FileOutputStream fos = new FileOutputStream(dirNameE2+"/"+fn+(j +
1)+ "." +ffmt);
// InputStream is=fise;
// OutputStream os=fos;
String key="Jenefe123";
int mode=Cipher.ENCRYPT_MODE;
DESKeySpec dks = new DESKeySpec(key.getBytes());
```

```
SecretKeyFactory skf = SecretKeyFactory.getInstance("DES");
SecretKey desKey = skf.generateSecret(dks);
Cipher cipher = Cipher.getInstance("DES");
if (mode == Cipher.ENCRYPT_MODE) {
cipher.init(Cipher.ENCRYPT_MODE, desKey);
CipherInputStream cis = new CipherInputStream(fise,
cipher);
byte[] bytes = new byte[64];
int numBytes;
while ((numBytes = cis.read(bytes)) != -1) {
fos.write(bytes, 0, numBytes);
}
fos.flush();
fos.close();
cis.close();
}
//---------------------- END-Encrypt------------------------
} else {
s3 = fn + (j + 1) + "." + ffmt;
File ff1 = new File(dirName3+"/" + fn + (j + 1) +
"." + ffmt);
FileWriter fw = new FileWriter(ff1);
BufferedWriter bw = new BufferedWriter(fw);
bw.write(cr);
bw.flush();
fw.close();
bw.close();
//---------------------- Encrypt------------------------
FileInputStream fise = new FileInputStream(dirName3+"/"+fn+(j + 1)+ "."
+ffmt);
FileOutputStream fos = new FileOutputStream(dirNameE3+"/"+fn+(j +
1)+ "." +ffmt);
// InputStream is=fise;
```

```
// OutputStream os=fos;
String key="Jenefe123";
int mode=Cipher.ENCRYPT_MODE;
DESKeySpec dks = new DESKeySpec(key.getBytes());
SecretKeyFactory skf = SecretKeyFactory.getInstance("DES");
SecretKey desKey = skf.generateSecret(dks);
Cipher cipher = Cipher.getInstance("DES");
if (mode == Cipher.ENCRYPT_MODE) {
cipher.init(Cipher.ENCRYPT_MODE, desKey);
CipherInputStream cis = new CipherInputStream(fise,
cipher);
byte[] bytes = new byte[64];
int numBytes;
while ((numBytes = cis.read(bytes)) != -1) {
fos.write(bytes, 0, numBytes);
}
fos.flush();
fos.close();
cis.close();
}
//---------------------- END-Encrypt------------------------
}
}
}
} catch (Exception e) {
System.out.println(e);
}
response.sendRedirect("upload.jsp?msm=suc"); %>
</body>
</html>
```

Download.jsp:

```
<html>
<head>
<title>CloudSecureErasureCode</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<script>
function val()
{
if(document.dow.fil.value == 0||document.dow.key.value ==
0||document.dow.name.value == 0||document.reg.mail.value ==
0||document.reg.pass.value == 0)
alert("Enter All Field");
return false;
}
</script>
</head>
<body >
<table width="820" border="0" align="center" cellpadding="0" cellspacing="0"
bgcolor="#FFFFFF" style="margin-bottom:10px;">
<tr>
<td width="101" valign="top"></td>
<td width="719" height="37" valign="top"><table width="620" border="0"
cellpadding="0" cellspacing="0">
<tr valign="top">
<td height="154" colspan="4">
<h1 align="center" class="TitreB style13"> Secure Storage Services In
Cloud Computing</h1>
</td>
</tr>
<tr>
<td width="157" align="center"><a
href="upload.jsp">Upload</a></td>
<td width="158" align="center"><a
```

```
href="index.html">Home</a></td>
</tr>
<tr>
<form name="dow" action="filesearch.jsp " onsubmit="return val()">
<td height="89" colspan="4" align="center">
File Name
<input type="text" name="fil"> <input type="submit" value="SEARCH" >
</td></form>
</tr>
</table>
</td>
</tr>
</table>
<br/>
</div>
</body>
</html>
```

Filesearch.jsp:

```
<%@page import="java.io.File"%>
<%@page import="java.sql.*"%>
<html>
<head>
<title>CloudSecureErasureCode</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<script type="text/javascript">
function sendview()
{
var dwnf=document.getElementById("dfln").value;
document.srchfile.action="filetoview.jsp";
document.srchfile.submit();
}
```

```
</script>
</head>
<body >
<table width="820" border="0" align="center" cellpadding="0" cellspacing="0"
bgcolor="#FFFFFF" style="margin-bottom:10px;">
<tr>
<td width="101" valign="top"></td>
<td width="719" height="37" valign="top"><table width="620" border="0"
cellpadding="0" cellspacing="0">
<tr valign="top">
<td height="154" colspan="4">
<h1 align="center" class="TitreB style13"> Secure Storage Services In
Cloud Computing</h1>
</td>
</tr>
<tr>
<td width="157" align="center"><a
href="upload.jsp">Upload</a></td>
<td width="158" align="center"><a
href="download.jsp">Download</a></td>
<td width="145" align="center"><a
href="index.html">Home</a></td>
</tr>
<tr>
<td height="89" colspan="4" align="center">
<%
try {
String m=request.getParameter("fil"); // file name entered in search box
ServletContext context = getServletContext();
String dirName =context.getRealPath("/OFile/");// reads the path of Ofile
boolean filexis=false;
File folder = new File(dirName); // creating one folder object
File[] listOfFiles = folder.listFiles(); //getting all file inside ofile
```

```
folder
for (int i = 0; i < listOfFiles.length; i++) {
System.out.println("File " + listOfFiles[i].getName());
if(listOfFiles[i].getName().equalsIgnoreCase(m))
{
filexis=true;
break; // to check the entered file name  exist in ofile
directory
}
}
if(filexis)
{
%>
<table width="100%">
<tr>
<td width="27%"></td>
</tr>
<tr>
<form name="srchfile" action="down1" method="post">
to down1.java
<td><input type="hidden" name="dwfilename"
value="<%=m%>"/></td>//m = filename entered in search box to be downloaded
<th><%=m%></th>
<td width="40%"><input type="submit"
value="Download"></td>
</form>
</tr>
</table>
<%
}
else
{
out.println("File Not Found");
```

```
}
}
catch(Exception e)
{
out.println(e);
}
%>
</td>
</tr>
</table>
</td>
</tr>
</table>
<br/>
</div>
</body>
</html>
```

Down1.java:

```
import java.io.*;
import java.io.FileInputStream;
import java.io.IOException;
import java.io.PrintWriter;
import java.sql.*;
import java.util.*;
import javax.crypto.*;
import javax.crypto.spec.*;
import javax.servlet.RequestDispatcher;
import javax.servlet.ServletContext;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
```

```
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;
@WebServlet(name="down1", urlPatterns={"/down1"})
public class down1 extends HttpServlet {
protected void processRequest(HttpServletRequest request,
HttpServletResponse response)
throws ServletException, IOException {
response.setContentType("text/html;charset=UTF-8");
PrintWriter out = response.getWriter();
HttpSession session = request.getSession(true);
int counter=0;
//String ofile=null;
try {
ServletContext context = getServletContext();
String dirName =context.getRealPath("/DFile/");
String fln=request.getParameter("dwfilename");
String filename=fln;
String[] filn=filename.replace(".", ",").split(",");
String s1=filn[0]+"1."+filn[1];
String s2=filn[0]+"2."+filn[1];
String s3=filn[0]+"3."+filn[1];
try {
//---------------------------Decrypt-----------------------------
int mode = Cipher.DECRYPT_MODE;
String dirNameE1 =context.getRealPath("/splitE1/");
String dirNameE2 =context.getRealPath("/splitE2/");
String dirNameE3 =context.getRealPath("/splitE3/");
String dirNameDF =context.getRealPath("/DFile/");
String dirNameD1 =context.getRealPath("/splitD1/");
String dirNameD2 =context.getRealPath("/splitD2/");
String dirNameD3 =context.getRealPath("/splitD3/");
String keye="Jenefe123";
```

```
FileInputStream fis1 = new FileInputStream(dirNameE1+"/"+s1);
FileInputStream fis2 = new FileInputStream(dirNameE2+"/"+s2);
FileInputStream fis3 = new FileInputStream(dirNameE3+"/"+s3);
FileOutputStream fos1 = new FileOutputStream(dirNameD1+"/"+s1);
FileOutputStream fos2 = new FileOutputStream(dirNameD2+"/"+s2);
FileOutputStream fos3 = new FileOutputStream(dirNameD3+"/"+s3);
DESKeySpec dks = new DESKeySpec(keye.getBytes());
SecretKeyFactory skf = SecretKeyFactory.getInstance("DES");
SecretKey desKey = skf.generateSecret(dks);
Cipher cipher = Cipher.getInstance("DES");
if (mode == Cipher.DECRYPT_MODE) {
//-----FILE-1-----------
cipher.init(Cipher.DECRYPT_MODE, desKey);
CipherOutputStream cos = new CipherOutputStream(fos1, cipher);
//doCopy(is, cos);
byte[] bytes = new byte[64];
int numBytes;
while ((numBytes = fis1.read(bytes)) != -1) {
cos.write(bytes, 0, numBytes);
}
cos.flush();
cos.close();
fis1.close();
//-----FILE-2-----------
cipher.init(Cipher.DECRYPT_MODE, desKey);
CipherOutputStream cos2 = new CipherOutputStream(fos2, cipher);
//doCopy(is, cos);
byte[] bytes2 = new byte[64];
int numBytes2;
while ((numBytes2 = fis2.read(bytes2)) != -1) {
cos2.write(bytes2, 0, numBytes2);
}
cos2.flush();
```

```
cos2.close();
fis2.close();
//-----FILE-3----------
cipher.init(Cipher.DECRYPT_MODE, desKey);
CipherOutputStream cos3 = new CipherOutputStream(fos3, cipher);
byte[] bytes3 = new byte[64];
int numBytes3;
while ((numBytes3 = fis3.read(bytes3)) != -1) {
cos3.write(bytes3, 0, numBytes3);
}
cos3.flush();
cos3.close();
fis3.close();
}
//----------------------------END Decrypt-----------------------------
//----------------------------File Join-------------------------------
FileInputStream ff1=new FileInputStream (dirNameD1+"/"+s1);
FileInputStream ff2=new FileInputStream (dirNameD2+"/"+s2);
FileInputStream ff3=new FileInputStream (dirNameD3+"/"+s3);
Vector<InputStream> inputStreams = new Vector<InputStream>();
inputStreams.add(ff1);
inputStreams.add(ff2);
inputStreams.add(ff3);
FileWriter fileWriter = new FileWriter(dirNameDF+"/"+filename);
PrintWriter out1 = new PrintWriter(fileWriter);
Enumeration<InputStream> enu = inputStreams.elements();
SequenceInputStream sis = new SequenceInputStream(enu);
int oneByte;
while ((oneByte = sis.read()) != -1) {
// out.println(oneByte);
//als.add(oneByte)));
out1.write(oneByte);
System.out.write(oneByte);
```

```
}
System.out.flush();
out1.flush();
out1.close();
fileWriter.close();
//----------------------------END File Join------------------------------
-
String filepath=dirName;
response.setContentType("APPLICATION/OCTET-STREAM"); //download content type
response.setHeader("Content-Disposition",
"attachment;filename=\""+filename+"\"");
FileInputStream fileInputStream=new FileInputStream(dirName+"/"+filename);
int i;
while((i=fileInputStream.read())!=-1)
{
System.out.println(i);
out.write(i);
}
fileInputStream.close();
out.close();
} catch (Exception e) {
System.out.println(e);
}
out.println("<h2 align=center><a
href=http://"+servnam+":"+locpor+"/CloudSplit/DFile/"+filename+">Click</a></h
2>");
}
catch(Exception e)
{
}
}
@Override
protected void doGet(HttpServletRequest request, HttpServletResponse
```

```
response) //doget,dopost is autogenerated in java
throws ServletException, IOException {
processRequest(request, response);
}
@Override
protected void doPost(HttpServletRequest request, HttpServletResponse
response)
throws ServletException, IOException {
processRequest(request, response);
}
@Override
public String getServletInfo() {
return "Short description";
}
}
```