2017

Confidentiality protection model for securing data in cloud computing

James Mwasela Mwanyika
Faculty of Information Technology (FIT)
Strathmore University

CONFIDENTIALITY PROTECTION MODEL FOR SECURING DATA IN CLOUD COMPUTING

Mwanyika James Mwasela

089671

Submitted in partial Fulfillment of the requirements for The Degree Of Master of Science in Information Technology at Strathmore University

Faculty of Information Technology
Strathmore University
Nairobi, Kenya.

June 2017

Declaration

I declare that this work has not been previously submitted and approved for the award of a degree by this or any other University. To the best of my knowledge and belief, the thesis contains no material previously published or written by another person except where due reference is made in the thesis itself.

Mwanyika, James Mwasela

9th June 2017

Approval

The Thesis of Mwanyika, James Mwasela was reviewed and approved by:

Dr. Vitalis Ozianyi
Senior Lecturer, Faculty of Information Technology
Strathmore University.

Dr, Joseph Orero
Dean, Faculty of Information Technology
Strathmore University.

Proffesor Ruth Kiraka,
Dean, School of Graduate Studies
Strathmore University.

Abstract

Cloud storage providers store the data in multiple servers maintained by hosting companies. This increases the risk of unauthorized access to the private data. Even though the cloud continues to gain popularity in usability and attraction, the problems lies with data confidentiality, loss of control, lack of trust, data theft and the fact that user data is stored in unencrypted format such as in the case of amazon 3 cloud storage servers. This research focuses on internal threats presented by cloud service providers. Using encryption techniques, the risk of unauthorized access can be controlled. In the proposed methodology, a user encrypts files with secret keys before uploading them into the cloud. Once encrypted, the file is stored in an encrypted format in the cloud. For a user to download files form the cloud, the file owner first accepts a request by an authorized user, and an application server provides an Access key. Using an access key, a user downloads data and uses a secret key to convert cipher text into a plain text. This technique ensures end-to-end encryption and completely hides the data from cloud service providers hence maintain confidentiality. Implementation involved building an encryption application algorithm, for deployment on the user computer. The algorithm consists of a single encryption and hybrid encryption modules. A user selects either a single or hybrid encryption module from the application based on security level requirements of data to be uploaded to the cloud. The model consists of registration/login module, encryption module, uploading module, downloading module and decryption module. This research contributes to providing security to the data stored in the cloud, by encrypting the data before uploading it into the cloud. Data owner controls key management where generation, storage and distribution remain in his control. Data owners lack the courage to strategically outsource data storage to the cloud. However, once the trust issues between data owners and cloud service providers are addressed through the deployment of this model, there shall be some attitude change on the side of data owners towards the adoption of cloud storage usage and therefore bridging the trust issues existing between data owners and cloud service providers.

Table of Contents

List of Tables

List of Figures

Acknowledgements

To my priceless family, you have been extraordinary and understanding. You understood the consequences of a journey to a Masters Degree. It gave me strength when I realised that we were together in this lonely journey. You made it happen in a special way. I owe you. Thank you so much.

To my friends at the University, your team dynamics was key and encouraging. It was our main strength. It has been great working with you all.

To all my lecturers, and entire Strathmore University Community, it has been an excellent experience. I enjoyed every bit of the interaction. Thank you all.

To my supervisor, Dr. Vitalis Ozianyi thank you very much for your believe, understanding, trust and guidance throughout the journey to this thesis.

Dedication

To my late Father, you believed I could make it happen. You knew it. This is for you.

Chapter 1: Introduction

1.1 Background to the Study

Cloud computing is an upcoming paradigm that offers tremendous advantages. It has become an emerging computing infrastructure for organizations throughout the world. The cloud computing uses specialized connections with a network of servers gathered substantially for data processing across them. Cloud computing make use of computer virtualization to optimize the use of computers (Mell, 2012). Through the use of virtualization, it reduces the need of purchasing, maintaining and updating applications and hardware platforms.

Due to security vulnerabilities in cloud computing, data owners are very cautious to store their data outside their own control limits and securing data in remote places have become important researchers' big headache. This research focuses on data confidentiality in the cloud repository system. Cloud users know very little or nothing about security policies implemented by the cloud vendors because of the nature of cloud computing. Almost every cloud service provider do not provide adequate data security measures or none at all to ensure data owners confidentiality. This is the main reason why potential cloud users are reluctant adopting cloud computing technology.

Cloud computing paradigm provides scalable computing function with resources being shared among tenants. Users access the service through Internet connectivity. It provides for sharing of computing resources thereby reducing costs of ownership to enterprises (Ikechukwu & Ugochukwu, 2013). Significant amount of data has traversed the Internet infrastructure thanks to technology such as cloud computing (Abduljabbar et al., 2014). Rapid growth of Internet has triggered fast growth of cloud computing technology since it's the main medium of connectivity between cloud users and the cloud service providers. Cloud computing allows computing resources provision on need basis, provide virtual storages and encourages resource sharing (Wang et al., 2013). Internet infrastructure remains the only medium that enables access to cloud computing services (Arijit et al., 2013)

Services mainly provided by the cloud are Software-as-a-Service (SaaS), Infrastructure as a Service (IaaS) and Platform as a Service (PaaS). The primary use of cloud computing services is data storage (Kumaraswamy et al., 2009). Cloud computing storage is designed for virtualized computer environment. The main objective of a business entity is to make

more profit with less investment. Cloud computing enables enterprises raise their income with less investment on information technology infrastructure. Industries no longer need to invest on high powerful computers. A simple less powerful device such as a smartphone enables a user access the cloud systems. The client is no longer constrained by its limited computing resources. It is widely known that information security; expertise, scalability, cost, and convenience are the major deciding factors for enterprises to outsource their data to cloud computing.

Xiaojun and Qiaoyan (2010) observed that despite numerous advantages of cloud computing such as, lower costs, ubiquity, location independency, availability, redundancy, performance, and reliability, cloud computing still lacks credentials for trust SMEs. Data Security is still a major challenge in cloud system due to its architectural nature. Unless robust security scheme is implemented, cloud system will continue to be vulnerable and risk for adoption by SMEs. According to Monikandan et al., (2014), data security is ensured by a number of known security parameters such as Authentication, Authorization, Confidentiality, Integrity and Availability. In information security confidentiality, integrity and authentication are the main security services.

Cloud repository may be attacked in two different ways: the attacks from outside and within the cloud infrastructure (Arockiam & Monikandan, 2013). Outside attackers are hackers who attack data from outside Cloud Service Providers (CSP) domain. Inside attackers are cloud service administrators who have the administrative privileges over computing resources. Data Security remains unresolved threat in cloud computing innovation considering the design, architecture and the deployment mechanisms (Xiaojun & Qiaoyan, 2010).

In cloud computing, users have no idea where their data is stored due to the distrusted nature of cloud computing. Since user's data protection mechanism in the cloud platforms has been geared towards controlling threats from external threats, it is now upon users to make their own security arrangements while in the tenancy of cloud service providers. Therefore there is need for researchers and academicians to come up with models for confidentiality protection of user data to enhance and provide data security (Sudha & Monica, 2012). This research investigates and proposes a confidentiality protection model for SMEs to ensure confidentiality of data in cloud repository systems.

1.2 Research Problem Statement

Cloud attracts commercial entities and individual customers by its fascinating characteristics and strategic commercial benefits. The main service provided by cloud computing is data storage. Despite cloud service providers claiming to offer security to data they host, there have been cases of security breach. There are cases where information theft and data integrity violations have been experienced to suit certain interests. A recent research by Mulazzani et al., (2011) revealed serious security flaws in dropbox platform. Dropbox management later admitted allowing government agents accessing user sensitive data through opened backdoors. This shows that user sensitive data is never safe while in the custody of cloud service providers.

Moreover, third party cloud service administrators have all access privileges to manage storage cloud servers. Malicious insiders can attack the computing infrastructure with relatively easy and less knowledge of hacking, since they have a detailed description of the underlying infrastructure and high-level access privileges. Without using a complete trustworthy solution for defending against insider attacks, malicious insiders can easily obtain the passwords, cryptographic keys, files and gain access to clients' records. This calls for the need to develop and implement data confidentiality protection model for data security in the cloud, (Sudha & Monica, 2012).

1.3 Research Objectives

i. To determine existing confidentiality protection techniques for securing data in cloud computing storage.
ii. To asses challenges of the current confidentiality protection techniques used for securing data in cloud computing.
iii. To develop confidentiality protection model for securing data in cloud computing.
iv. To validate and Analyses the model.

1.4 Research Questions

i. What are the current confidentiality protection techniques for securing data in cloud?
ii. What are the weaknesses that exist in the current confidentiality protection techniques preventing cloud customers from adopting cloud storage as a service?
iii. How will the proposed model be developed and implemented?
iv. How will the model be validated and analyzed?

1.5 Justification

Commercial Enterprises and individual customers have shown their readiness to adopting cloud-computing storage by outsourcing their Information Technology storage infrastructure. However, due to the issues related to security vulnerabilities in the cloud, these customers are hesitant to outsource computing data storage function to cloud service providers. Data is stored in an unencrypted format in servers hosted and managed by cloud providers. Any encryption provided is managed and operated by the cloud providers. The cloud service providers does the encryption if any, does key generation distribution and storage. This amounts to security vulnerability as malicious cloud service providers can have unauthorized access to users data without their knowledge. The key security issue here is that the data owner has no control of where and how the data is stored. If customers want to take advantages of cloud computing innovation, they must ensure data confidentiality mechanisms are in place. Sensitive data must remain confidential even to the cloud service providers. To achieve this, a confidentiality protection model must be developed and implemented.

1.6 Scope

Cloud computing has a vast scope of study. From infrastructure, platform and software all provided on pay per use arrangements (Mell & Grance, 2011). This thesis focuses on confidentiality of data stored in the cloud repository system. We take note that any security flaw on users data stored in the cloud would affect client trust in using cloud repository system. In order to win users trust, cloud service providers must ensure zero security flaws. Our focus will be to research on ways to provide confidentiality protection of user's data from cloud service providers. Assumption is made that cloud service providers can never be trusted on user sensitive data and that all users data is confidential.

1.7 Limitations

The research is limited by time constraints. There has been very limited time to juggle between my professional day-to-day work obligations and academic duty. This is overcome by limiting the scope of the research to one component of cloud computing security; data confidentiality, and to specifically center the research on how to provide user data confidentiality residing in cloud repository systems.

Chapter 2: Literature Review

2.1 Introduction

Ensuring confidentiality of client's data in cloud is the main research problem around the cloud computing. Cloud storage providers store users critical data; it needs to be secured. Cloud computing has a recent success in Information Technology (IT) and will dominate the IT industries in the coming years. Cloud computing also faces the overwhelming challenges. To ensure the proper physical, logical and personnel security controls, especially in cloud data storage are more significant. Moreover, while moving such large volumes of data, the management of the data may not be fully trustworthy. This section describes the research works, which are related to problem domain on ensuring the confidentiality of data in cloud storage.

2.2 Understanding Cloud Computing

Cloud models include private, public and hybrid providing services such as platform, software and infrastructure (Rabi et al., 2011). Cloud computing optimizes the concept of resource sharing. The main features of cloud are flexibility, ease of expansion, reduced cost of ownership, redundancy, speedy processing, reliability and location independency. According to JorgSehwenk et al., (2009) some of the major security flaws in cloud computing include unauthorized access, confidentiality, lack of control, recovery and responsibility.

2.3 Security in Cloud Computing

According to Chen and Zhao (2012) when it comes to cloud computing security, people must focus their minds on data security itself and how confidentiality of user information is protected. Security threats are the major factors that make key strategic managers reluctant from making strategic decisions on whether to outsource big data storage to third part service based systems. Such fear by SMEs executives have really contributed to slow pace of acceptability of cloud computing technology in the market. According to Akhil and Kanika (2012) a significant level of trust must be established and maintained before cloud providers considering selling the concept of cloud computing to consumers.

In cloud computing there several risks that are faced by both the cloud users and cloud provider. Risks such as loss of control lack of standards especially when it comes to information systems auditing, resource sharing, location independency laws and regulations.

According to Mariana et al., (2011) the main risk people are concerned about in the cloud technology is security of user data in the cloud repository system. Computer security is concerned about data confidentiality, integrity and availability. These are the main parameters when it comes to data security. This research focussed on confidentiality parameter.

2.4 Data Confidentiality in Cloud Computing

Cryptography is so far the main technology that provides confidentiality of information. It is used to protect data in transit and in store. Several encryption algorithms are in use in the industry today to ensure data confidentiality in different sectors of the economy. Technically, the algorithms are either symmetric or asymmetric in the way encryption keys are generated and used (Chandra, 2012). Symmetric algorithm uses a single key for both encryption and decryption while asymmetric algorithm uses private and public key to perform encryption and decryption respectively. In terms of response times, symmetric encryption is superior over asymmetric encryption (Nigoti et al, 2013). Despite cloud servers hosting very sensitive user data in their midst, Yau and Ho (2010) submit that cloud service administrators who manage cloud servers cannot be trusted with user data residing in the cloud repository systems.

2.4.1 Cross-VM attack via Side Channels

According to Aviram et al., (2010), resource sharing by multiple cloud users and nature of parallelism architecture in cloud computing is the biggest threat to backdoor unauthorized access that happens between cloud tenants. Malicious customers may steal other customer's data leaving no trail behind.

2.4.2 Malicious System Administrator

Not all system administrators are trustworthy to organizational information. The same applies to cloud system administrators. Malicious cloud provider system administrator with extensive privileges for accessing cloud computing platform information technology infrastructure may use his position for financial gain. A Tenant competitor for example, may bribe a cloud system administrator to get other business rival company secrets for business or sensitive innovation due to business rivalry.

2.5 Confidentiality Protection in Cloud Computing

Most of data confidentiality techniques existing were designed for protecting data from external threats and data on transit. Consideration was not given to confidentiality protection

of user data due to threats emanating from cloud service providers themselves. Cloud service providers are a new category of threats to user data confidentiality. A different perspective of how we view data protection must change. This means that researchers must go back to the drawing board and come up with a model that can safeguard data confidentiality from cloud service providers.

2.5.1 Encryption using Hybrid Algorithm and Secured Endpoints

Rani et al., (2012) proposed a technique of protecting user data in cloud servers whereby data is subjected to three layers of encryption sequentially. This technique uses ceaser, Rivest Shamir and Adleman (RSA) that is then enhanced by monoalphabetic substitution method. This technique provides an enhanced data protection. In this technique, users access the encryption keys from the cloud servers. It is worth noting that this technique provides great security and it may not be easy for an adversary to be able to break it through brute force.

However this technique has some drawbacks. The cloud service provider does generation distribution, and storage of cryptographic keys. This further exposes some weaknesses. The main threat on this technique is that cloud service provider may use the decryption keys he's keeping to still decrypt and access user's data. Secondly, the response time is higher than a single encryption due to overhead brought about by multiple encryptions and compression.

2.5.2 Confidentiality Protection Through Service Separations

According to Yau and An (2010) when you separate cloud services, hide data and enforce data obfuscation one may achieve data confidentiality. The authors observed that cloud service providers have the privilege to know where users data is located, the format used on data and administrative rights to access that data. The two further argued that if a model can prevent cloud provider from these privileges and prevent cloud providers from understanding the meaning of data then data confidentiality might be achieved in cloud computing repository systems. I agreed with his argument because technically, once you prevent these privileges bestowed on the providers, data confidentiality may be achieved.

However, according to Mell and Grance (2011) on cloud computing architecture, they submitted that cloud computing system consists of infrastructure module, software module and software development platform module all of them being provided as a package of a complete cloud computing system. This means that every subscriber is virtually forced to use all services provided by one cloud-computing provider. This allows cloud service providers to retain all the privileges on users data. Moreover the user is forced to use all interfaces

configured by the cloud provider with data in a format dictated by the cloud provider. Finally the user will again understand the meaning of users data. Therefore this technique is not technically possible due to the nature of cloud computing architecture.

2.5.3 Confidentiality In Public Cloud Database

This technique involves securing data and providing confidentiality of database queries at runtime in cloud databases. The databases provide users with ability to perform the action of creating, storing, modifying and retrieving of data from anywhere through Internet infrastructure. Luca et al., (2013) presented an idea of encryption architecture on Structured Query Language (SQL) operations. With this model, data confidentiality would be provided on any type of SQL operations at run time. Popa et al., (2011) made his contribution on this model by conducting research that came up with a recommendation for an adaptive encryption technique. The technique guarantees and provides data confidentiality for SQL operations at runtime. It allows cloud database servers to carryout vast number of SQL operations on encrypted data. The technique uses different algorithms each supporting specific SQL operators.

It is prudent to acknowledge that this scheme provides some level of confidentiality on user data in cloud databases. The scheme provides improved database confidentiality because of the use of an improved encryption schemes as compared to homomorphic encryption technique, which uses a simple encryption algorithm. The scheme also provided confidentiality for different SQL operations at run time. The scheme also enables some form of encryption where SQL operations can be performed on encrypted data. It does not require any proxy to manage any encryption functions. Moreover, low Internet and Local Area Network Latency mask its drawbacks such as response times. Despite these advantages, the scheme has some drawbacks. It has very high computational costs where there is double encryption of each plain column hence increasing database sizes. The other drawback is response time's overhead where the scheme has to encrypt all parameters and decrypt all results of each SQL operation on layers of all encryption schemes.

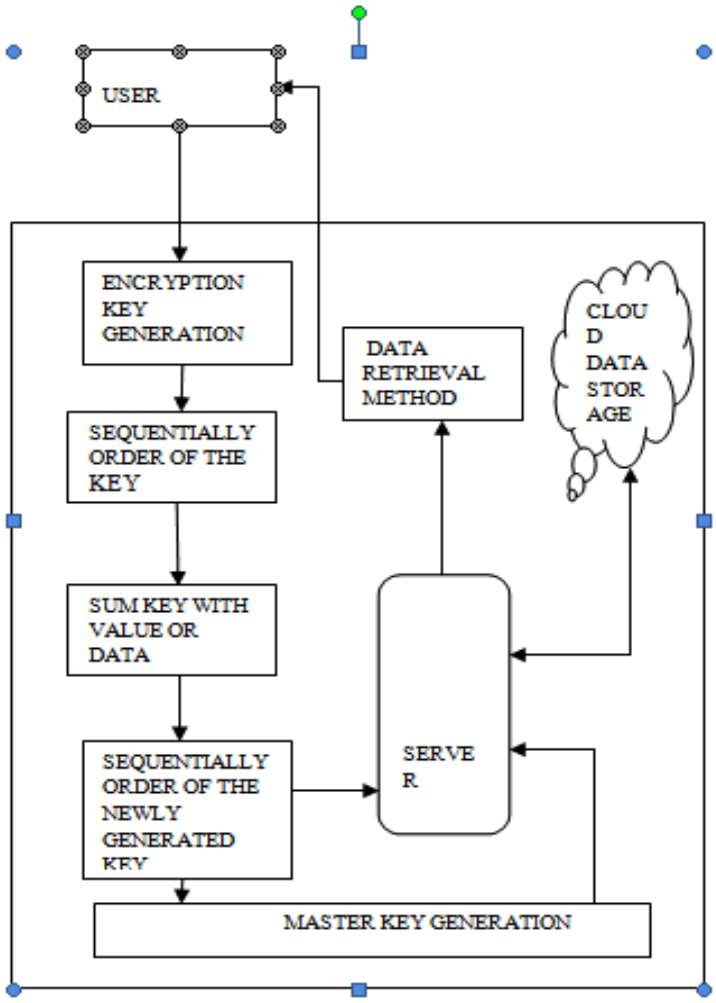

Figure 2.1:Cloud Database

Adapted from Bhuvaneswari and Pradeepa, (2015)

2.5.4 Enhancing Cloud Security Using Multi-cloud Architecture

Rutuja and Basha (2015) proposed a scheme that tackles three objects: Setting up and configuring multiple cloud storage servers, developing encryption techniques like Advanced Encryption System (AES) for file encryption before storing it in the cloud; develop file management classes and a web interface for file upload and download in cloud storage. In this scheme, data is stored in multiple clouds rather than on a single cloud. Further, the data security at each Cloud Service Providers is enhanced using encryption methods and device based data access. The model presents a multi-cloud architecture.

The approach proposes breaking application logic in parts, executing and stores each part over multiple clouds to preserve confidentiality. It allows data to be broken into parts and executed over various clouds that helps to protect the data from malicious Cloud Service

Providers. In this scheme, the file is encrypted using the encryption algorithm to guarantee confidentiality with access permission bound to the file as well as to the identity of the authorized access device. The file is then decrypted when requested by the authenticated user.

The approach provides redundancy apart from providing confidentiality. Data is also hidden from cloud service providers and from external hackers threat. However, the scheme has some drawbacks. There is an expected higher costs on cloud subscription to multiple cloud due to the nature of the model. There is also overhead on central processing unit (CPU) on data splitting and merging cycles; the scheme also has performance issues related to response times on data splitting and merging processes. The architecture only allows specific authorized devices to access stored data. Finally cloud service provider does encryption where the responsibility of key management is under his control.

2.5.5 Anonymous Service Usage Model

This model was proposed by Huang (2010) in his contribution to provide user data confidentiality in the cloud. He introduced the idea of using trusted entities to preserve users identities. The main idea is to hide user information to prevent identity of cloud users. In Figure 2.2, a user can access, use and pay cloud service providers through proxy thereby hiding his identity and trail. Any attempt by a user to perform unorthodox activities, the scheme will have his identity revealed.

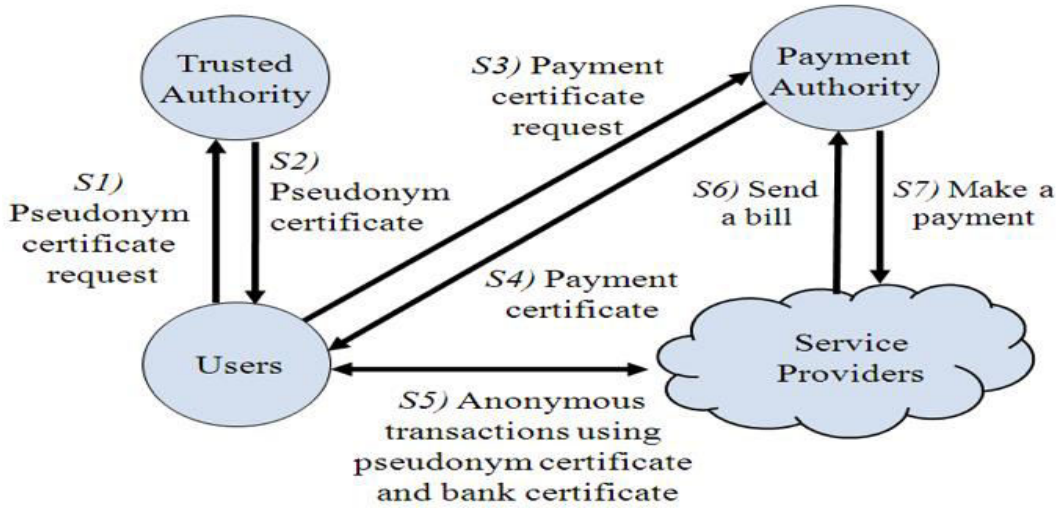

Figure 2.2: Anonymes Service Usage and Payment

Adapted from Sanket Salvi et al. (2015)

From figure 2.2, the technique has five modules. Trusted authority that issues pseudonym certificate on request, payment certificate issued by payment authority and make payment to the cloud providers based on the cloud usage by the user. The user does not deal directly with the cloud service provider but through proxies as shown in the figure 2.2. The approach provides confidentiality in different forms. It is based on a strategy of hiding the user identity. Since cloud service provider is considered the threat in this study, the technique ensures that user identity is hidden. Cloud service providers cannot therefore know the type of users subscribed to his services.

The technique however has security flaws. A Cloud provider may steal either Pseudonym certificate or payment certificate of the user and masquerade as the actual user. We may have fake bills being submitted for payment. Also, a cloud provider by way of invoking the security flaws may repeatedly charge users on same service rendered multiple times without the knowledge of the user. It is also possible that an attacker can easily steal users Pseudonym certificate or payment certificate and make illegal transactions. Finally it's easy for a user to deliberately refuse to pay his bill since his identity is unknown.

2.5.6 Encryption, Compression and Key (ECK) Edu-Cloud Architecture Model

The technique to safeguard data confidentiality was proposed by Salvi et al., (2015) on Edu cloud. It was meant to promote use of cloud computing technology among researchers. The author concept involved encrypting and compressing researchers work in a virtual machine space. He enhanced it further by adopting a privacy protection by use of a key. A user has to be a registered user in the Edu Cloud. In Edu architecture, each user encrypts his data, which is placed in the security layer where it is compressed. Subscriber's password is used as a key for encryption as shown in Figure 2.3 below. This scheme uses a variant of data security as presented by (Eman et al., 2012).

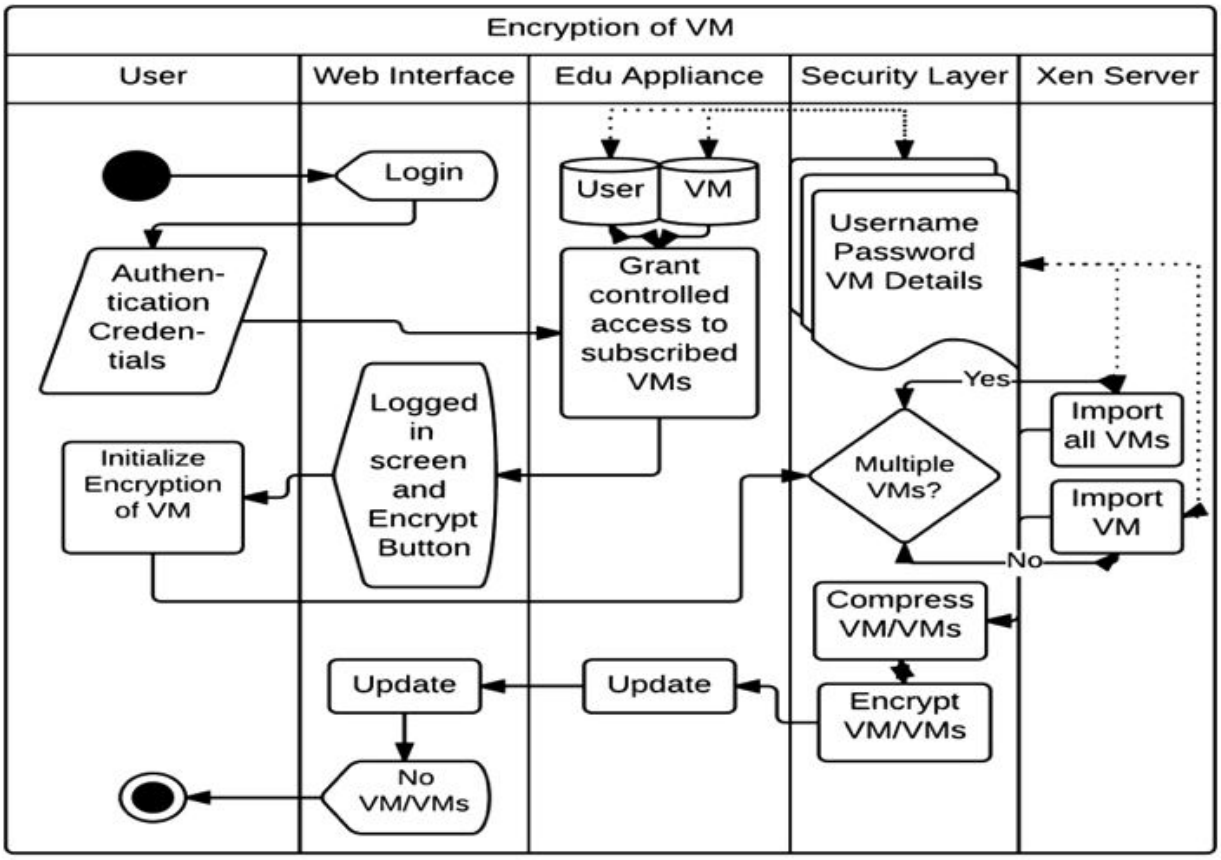

Figure 2.3: Edu Cloud Encryption

Adapted from Sanket Salvi et al. (2015)

When a user logs and accesses the Edu Cloud through the web interface, he is authenticated and allowed to decrypts his encrypted data stored in the Edu appliance platform of the architecture. The security layer will decrypt the encrypted data and decompress the encrypted data based on logged user credentials and exported back to user platform.

The Figure 2.4 describes in detail the decryption process once a user is correctly authenticated to access the cloud.

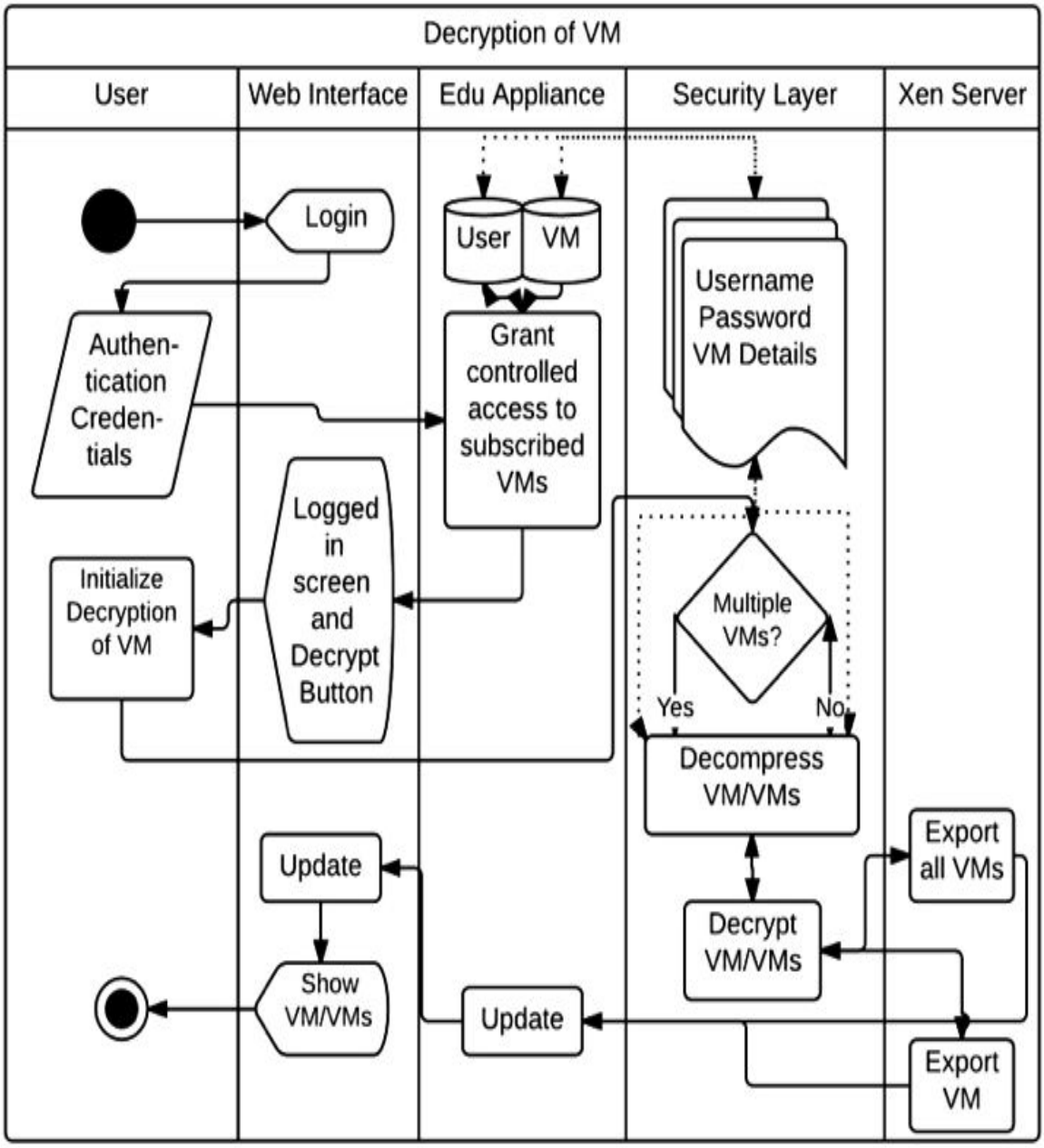

Figure 2.4: Decryption of VM

Adapted from Sanket Salvi et al. (2015)

Figure 2.5 shows a complete and detailed integrated Edu-Cloud architecture used to by researchers to secure stored data.

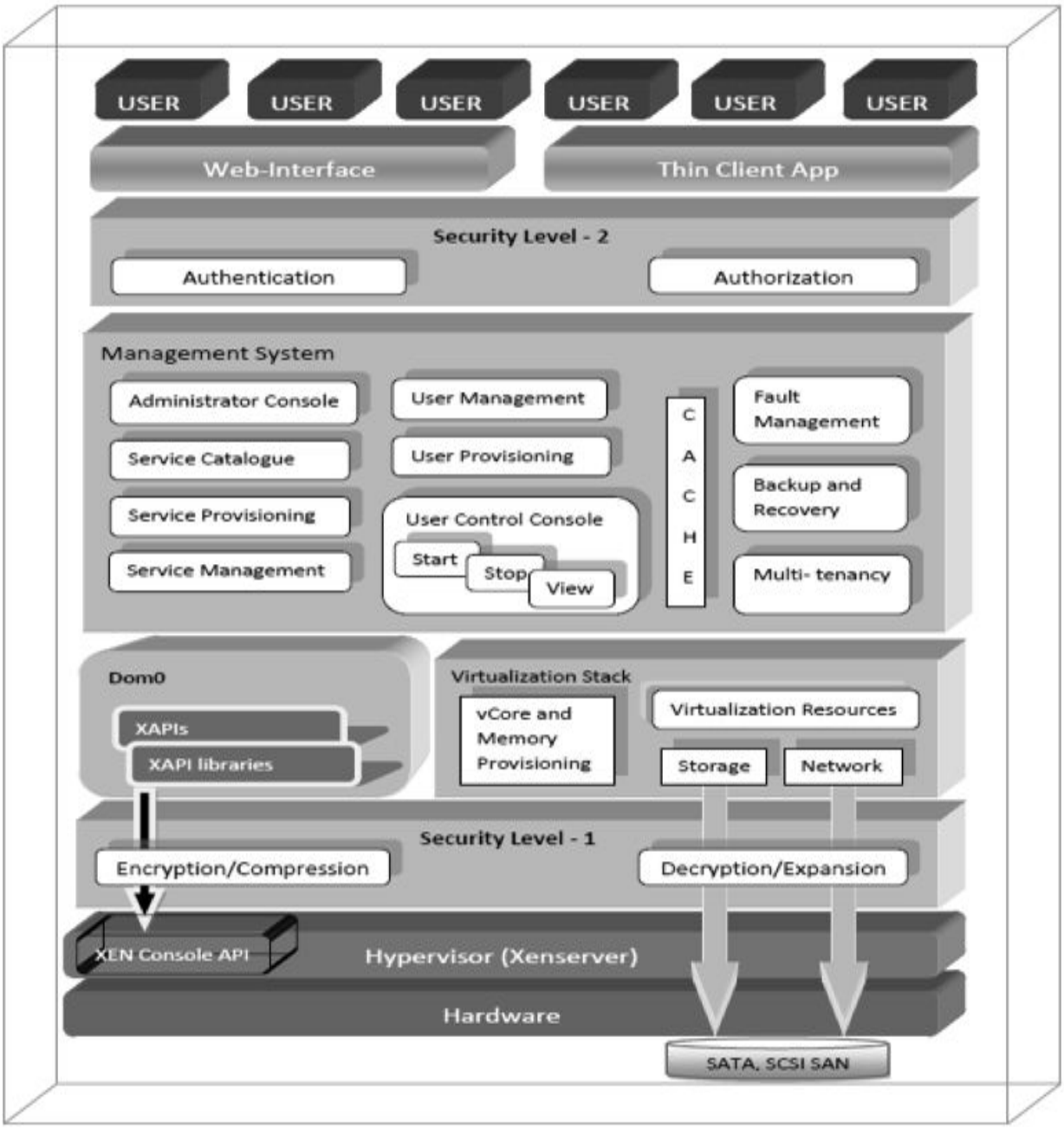

Figure 2.5: Edu-Cloud Framework

Adapted from Sanket Salvi et al. (2015)

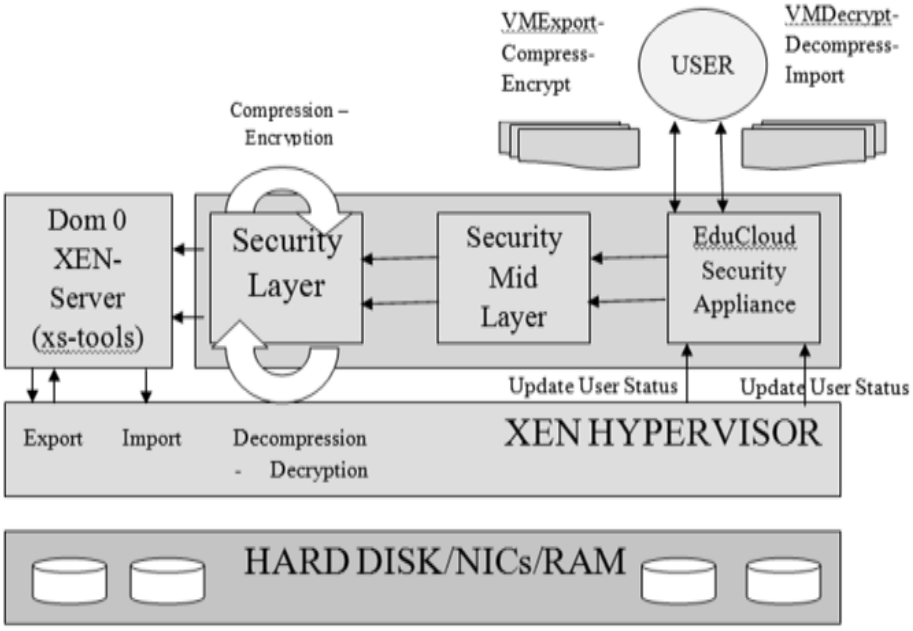

Figure 2.6: Security Framework in Edu – Cloud

Adopted From Sanket Salvi et al. (2015)

According to Wen and Xiang (2011) designing and implementing security in a virtualized environment has vast number of advantages such as flexibility, resource utilization, mobility and control. In addition, a vast contribution on virtual computing by Chunxiao et al., (2010) on Xen virtualization in cloud security and expansive research on disk encryption in virtual computing by (Liang & Chang, 2011) provide deep understanding and need our appreciation on the advantages of implementing full virtual machine encryption as adopted by Edu-Cloud Architecture.

Despite the scheme providing data confidentiality, the scheme has some drawbacks. The cloud providers do encryption and decryption at the cloud side, which opens up window of unauthorized access of user data by cloud providers. User credentials including user names and passwords are stored within cloud servers. This does not guarantee user data confidentiality as the cloud administrators can access the credentials and access user data. There is also additional overhead cost of encryption and compression of user data.

2.6 Conceptual Framework

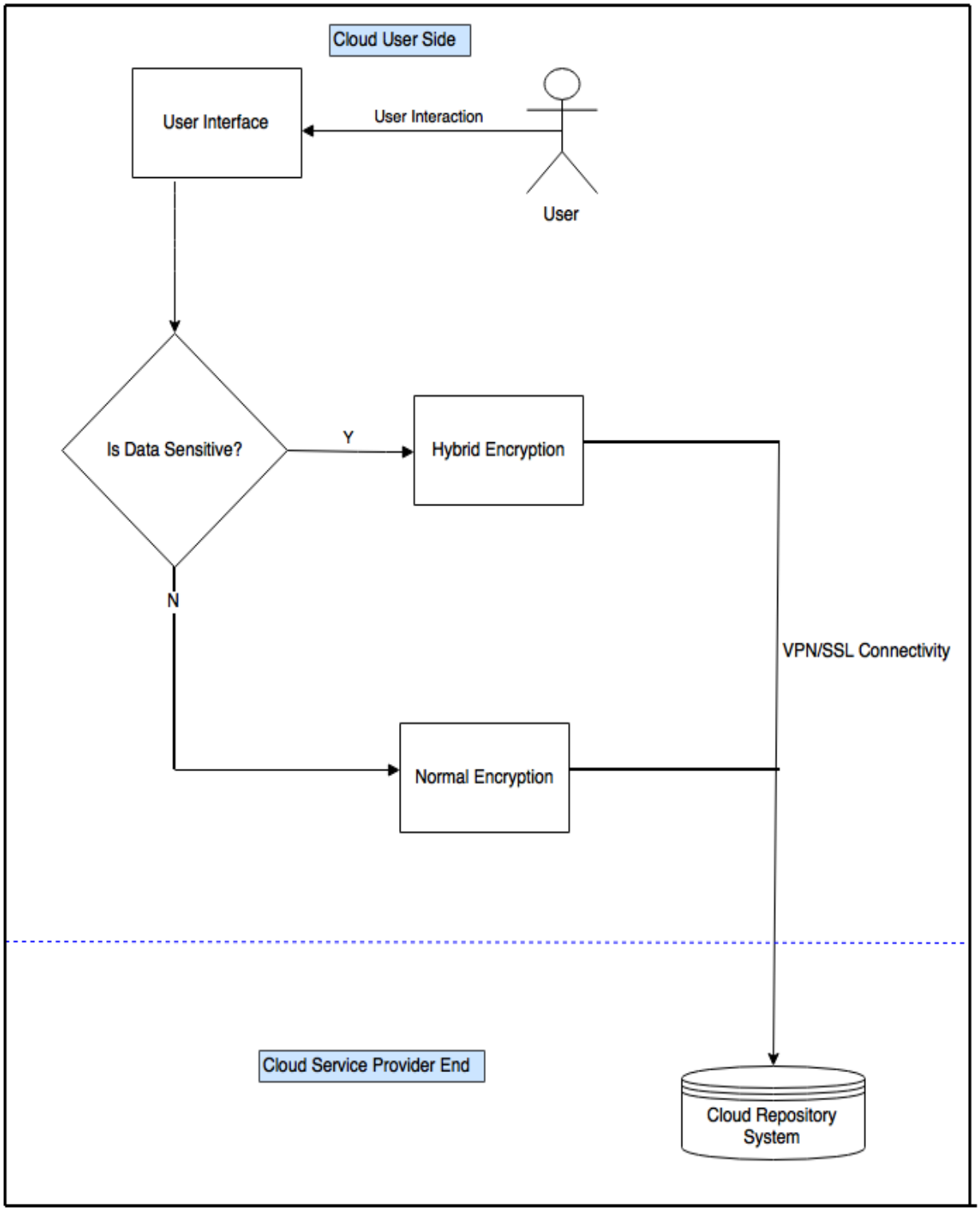

Figure 2.7: Conceptual Framework

The Conceptual Framework consists of cloud user side and Cloud service provider side. User side performs the function of registering user; user login into the office domain, selection of file tile to upload to the cloud storage; secret key generation and file encryption. Username and password credentials are used to login. The user also performs Request/ retrieve credential; Uploads and verify stored files; View/ manage rights of own files and Stores important data into Cloud Database. All the data are encrypted before being uploaded to the cloud database. Encryption is a process that converts plain text into unreadable format called cipher text. Hybrid encryption uses the combination of two encryption algorithms such as Triple Data Encryption Standard Algorithm (3DES) and Rivest Cipher 4 (RC4) to enhance security level of user data. Single encryption uses any encryption algorithms.

In this model, Encryption is based on sensitivity of data: Data owner may divide data in three categories based on its sensitivity less sensitive, Adequate sensitive and highly sensitive. Based on the sensitivity of the data, user makes a decision on either to apply single encryption or Hybrid encryption. Decision Module is used to find out the security level of data, which is ready to be uploaded and store in the cloud storage. Based on the security level of data, single encryption or Hybrid is used to encrypt the data before forwarded to the cloud. If the data has a higher security level, Hybrid encryption is applied, if the data is has a lower security level single encryption is applied on the data. This module calls for a corresponding algorithm based on the level of security.

Module for single/normal encryption uses symmetric encryption algorithm to encrypt data using an encryption algorithm implemented at the user computer before upload. This algorithm is used for both numeric and alphanumeric data type. It is used to encrypt data with a lower security level as determined by data owner. Hybrid encryption module is used to encrypt data that has been determined to be of higher security level. Two symmetric encryption algorithms 3DES and RC4 are used to enhance data security. This module sacrifices performance at the expense of achieving higher security level of data.

Once data is encrypted, it is uploaded into cloud repository system through a secured Internet channel. The cloud repository system allocates storage space; Store user files, Calculate on users Demand, Store versions of files and Provides Group sharing.

Chapter 3: Research Methodology

3.1 Introduction

The main objective of this chapter is to present some form of techniques that were used towards achieving the objectives and answer the research questions. The methodology adopted will be used to present a confidentiality protection model that will guarantee user data in cloud computing repository systems.

According to Kothari (1990) research methodology is an approach used to finding a solution to a research problem of a topic under study. Some may understood it as a science of trying to understand how a research is done scientifically. Research methodology involves the study of steps to be generally adapted to by a researcher in studying his research problem towards achieving objectives. Therefor it is prudent for a researcher to both know research methods as well as methodology necessary to carrying out a research.

3.2 Research Design

The research design helps ensure that the research process is as efficient as possible to be able to provide enough information. The research design helped in the collection of important and relevant information with less consumption of time, effort and money. The research purpose adopted in this study was descriptive and partly experimentation.

In research design, a model is presented to show how a research was conducted. Research design provides an actual framework for the design; build tests and analysis of test results, which were used to answer research questions of the topic under study. In this research, we adopted mixed methods consisting of descriptive and experimentation research design.

In this study, preliminary designs were developed based on the conceptual framework. Final detailed designs for the model under study were then developed together with functional and non-functional requirements. Then architecture of how the final model would look like was developed. It gave a layout of how the final model would look like and also provided reference during development. Next was the implementation of the designs. Implementations consisted of developing multiple modules making up the model. Each module had different independent functions. The modules formed part of the building blocks to the final product. It made development easy as each module was separately verified for compliance to requirements. The modules were then integrated together to form a complete product. The end product was then tested.

3.2.1 Descriptive Research

This design category provides some concrete answers to the questions related to who, where, when what and how that is linked to a particular research problem. It is often used to obtain accurate information regarding current status of an occurrence and also it describes what exists with respect to some variables or conditions. The type of descriptive study used in this research study involved review of the existing records and literature materials. It involved qualitative and quantitative aspect of research.

3.2.2 Experimental Research

Experimental research allows an investigator to manipulate conditions for the purpose of determining their effects and behaviour. It describes the process a researcher undertakes for the purpose of controlling some variables and manipulating others. This technique is always randomized to ensure that there will be no bias or any error to compromise results of the research.

In this research an experiment was carried to determine performance of an encryption algorithm suitable for adoption. A performance analysis of data encryption algorithms experiment involved a fixed text of data on three (3) encryption algorithms. Since we know security features of each algorithm in use and their strengths against cryptographic attacks, we focussed on performance. This experiment was carried out to determine encryption time for three encryption algorithms. Response time analysis was carried out using a performance analysis tool. The results were compared with results obtained form previous similar experiments on performance and security of encryption algorithms.

3.3 System Design and Architecture

An important component of the design phase is the architecture design, which describes the system's hardware, software, and network environment. The architecture design flows primarily from the nonfunctional requirements, such as operational, performance, security, cultural, and political requirements. In this research, the main deliverables from architecture design were the architecture design and the hardware and software specification. It helped in providing the layout of the model showing how modules interacts, how data flows and hardware and software modules. It also describes all components and their communication between them.

3.4 Development Methodology

Deveoplemt of a prototype model for this research was critical for the purpose of validation of the concept. The development methodology adopted was rapid application development. Rapid application development is a software development methodology that uses minimal planning in favor of rapid prototyping. A software prototype was developed to validate a working model that is functionally equivalent to a component of the product. The functional modules were developed in parallel as prototypes and were integrated to make the complete product for faster product delivery. Since there was no detailed preplanning, it made it easier to incorporate the changes within the development process.

3.5 Sampling

Sampling involves selection of a fraction of data from a given population to help draw a conclusion about the whole population. The main objective of this stage was to present a plan on how a sample can be selected and of what size such a sample could be. The parameter of interest was Performance. A sample data was used to test performance of encryption algorithm. Approximate value was obtained and used to classify algorithms based on response times.

3.6 Testing and Analysis

In a research work, testing has to be performed and results analysed as per the procedure laid down at the times of developing research plan. Testing means investigating the performance of behaviour of a system to establish whether the research questions have been substantially answered. Analysis is a process of calculating certain measure and searching for certain patterns of a given relationship existing among data groups. In this study, testing was performed on all the modules constituting the building block of the model. Finally a final set up of the model was tested with cloud user end connecting with a local user end through Internet infrastructure. Analyses were done from observations and test results.

3.6.1 Data Testing

Several test scenarios were conducted on application modules. Testing involved testing of different modules and the final integrated model set up tests. Test one consisted of a login and registering module. In this test case, authentication was tested in registration page and login page. For a user to successfully log into the system, his log in credentials must match his profile stored in a database else login fails. On other tests multiple functionalities were tested. This included, encryption of data, downloading of data to a user computer and finally

using the symmetric secret key to decrypt downloaded data from the cloud repository system.

A final test was conducted at the local user in the cloud architecture set up. This was a form of integrated testing where a complete model was configured and testing done from authentication, encryption, uploading/downloading and decryption. Platform independency and ubiquity testing was also performed. A smart phone running on android and operated away from the office was used to access the encrypted file from the cloud repository. This test was meant to demonstrate how user friendly and interoperability the proposed model is.

3.6.2 Data Analysis

Analysis was done on experimental results related to performance levels of encryption algorithms and on test scenarios results.

3.7 Research Reliability and Viability

In order for research data to be of value and of use, they must be both reliable and valid. Reliability refers to the repeatability of findings. The experimental results compares with similar experimental results obtained before by different scholars specifically on values of security and performance levels of encryption algorithms. This shows that if the study were to be done a second time, it would yield similar results. Therefore in this research, the data obtained from the experiments are reliable. End to end encryption was tested severally on a fixed text of data where text files were encrypted, uploaded to the cloud, downloaded and finally decrypted. Data integrity and size were maintained. The performance levels of the encryption algorithms were severally tested to give credible results the average of which compares with descriptive results obtained by other similar experiments carried out before.

Validity refers to the credibility or believability of the research. This research had some degree of validity. The research is credible. The research answered all the research questions, which were subject of the study. The model was validated and results found credible as they compared favorably with other results obtained by other scholars and met expectations.

3.8 Research Quality

Ethical issues can arise at some stage or in every stage of a research study. To ensure quality of a research, the following standards were maintained throughout the study.

i. The problem should be well formulated, and the purpose of the study should be clear.
ii. The study approach should be well designed and executed.

iii. The study should demonstrate understanding of related studies
iv. The data and information should be the best available.
v. Assumptions should be explicit and justified.
vi. The findings should advance knowledge and bear on important policy issues.
vii. The implications and recommendations should be logical, warranted by the findings, and explained thoroughly, with appropriate caveats.
viii. The documentation should be accurate, understandable, clearly structured, and temperate in tone.
ix. The study should be compelling, useful, and relevant to stakeholders and decision makers.
x. The study should be objective, independent, and balanced.

To ensure quality in this research, we upheld with humility, the above quality research standards and used them as a guide throughout the research period.

3.1 Ethical Considerations

Throughout the period of this thesis, it has been my pleasure engaging in ethical practices while anticipating what ethical issues would likely to come up. This chapter presents some outlines for the overall ethical issues that were considered as the thesis report was being prepared. As we are aware theses can be considered incomplete if this part alone is not included in the report. Ethical consideration has been one of the most key elements throughout the period of this research study work. The following ethical considerations were adopted throughout the study. The following ethical guidelines were put into place throughout the research period:

i. Respect for the dignity of research participants should be prioritized.
ii. Respect for research sites so that they are left undisturbed after a research study.
iii. Providing accurate account of information during interpretation of data.
iv. Desisting from suppressing, falsifying, or inventing findings to meet a researcher's or an audience's needs.
v. Desisting from engaging in duplicate or redundant publication in which authors publish papers that present exactly the same data, discussions, and conclusions.
vi. Present an introduction that identifies a significant problem or issue to study and presents a rationale for its importance.
vii. Both the researcher and the audience should understand the objectives of the study.

viii. Any type of communication in relation to the research should be done with honesty and transparency

Chapter Four: System Architecture and Design

4.1 Introduction

The architecture and detail designs for the model are presented in this chapter. The designs present technical details of our secure cloud storage model to aid implementation, installation, testing and maintenance. The proposed model bridges the trust gap between the cloud service providers and the users. It ensures security of user data is assured. Due to vulnerabilities observed in chapter two by letting the cloud service provider manage encryption of data, this model devolves the function of encryption and key management to user side thus enhancing data confidentiality.

This model shall remove the trust issue of key management by the cloud service providers by having it maintained by data owners. End users have the power to control all access controls which otherwise was left under the care of the cloud service providers. Here the user first encrypts the file before uploading to the cloud repository system. Secret keys for encryption and decryption is stored in local end user database. Data is stored and downloaded in an encrypted format.

4.2 Architecture

The System architecture for this model presents a conceptual model while defining views, structure and behaviour. In this research, the architecture presents, layout of the model showing how modules interacts, how data flows and hardware and software building blocks composition. Model architecture enables the implementation, understanding, maintenance, repair and further development of the model under study. The System Architecture of the cloud repository system shown in Figure 4.1 describes various components and communication between those components. A user as depicted in the system architecture, shall be authorized to login to the local user computer domain.

The key generation and encryption module is where a selected encryption algorithm generates secret key. This module converts a plain text into cipher text of a file that is be uploaded to the cloud storage. This process automatically generates a secret key. This module resides at the client side of the model. The encrypted file is then uploaded to the cloud repository system residing at the cloud side of the model.

Cloud repository system is a cloud storage infrastructure. It hosts cloud storage servers with the backup infrastructure to ensure continuity. A cloud user rents this space from cloud

service provider and pay per use for the storage of uploaded files. This model requires that the uploaded cipher text is stored in an encrypted format. The download module enables an authorized user download an encrypted cipher text from the cloud repository system. Only users with access key can access this module to be able to download files. Access key is controlled and distributed by a system administrator. Decryption module is located at the user end. It is used to decrypt the downloaded cipher text form the user machine. A user must have a secret key to decrypt a cipher text.

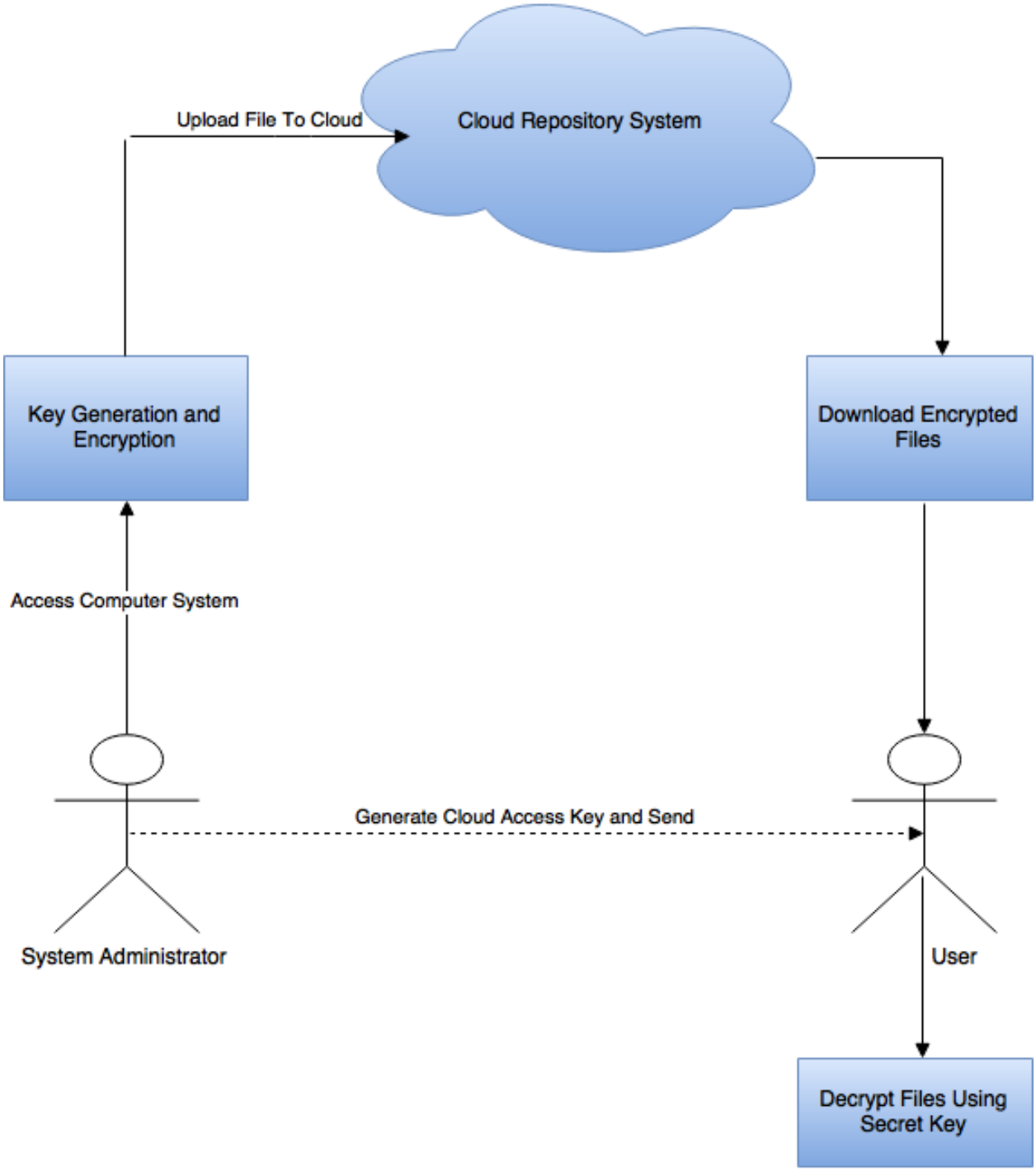

Figure 4.1: Proposed Model Architecture

4.3 Design

Here the model layout is presented depicting how the final integrated assembly will look like. It shows how modules used to build the final model look like, how they interact, the specifications and functional and non-functional requirements. It is a blue print of the final product expected and the design provides a reference to developers. Designs assist the audience and users understand how the final product will look like. Moreover, design specifications make it easier for building and implementation of a conceptual framework.

Figure 4.2 shows the Model design of the confidentiality protection model. It uses cloud to store encrypted information uploaded by the users, download encrypt data from the cloud, access key distribution to users and decryption of downloaded data using the secret key. The login validations checks the username and password entered with the username and password in the local database and either accept or reject a user. Upon confirmation, the application server will establish a connection with the model. This model allows the user to store or retrieve data from cloud database in encrypted format.

4.3.1 Design Concept

At the user side, a user has to register with office computer system first before log in. Local database store profiles of all users registered with the office domain. A user may either want to retrieve encrypted data or upload an encrypted data. For a user to access the cloud depository system to download encrypted data, he needs an access key. This key will enable a user to download the data for decryption at the local user workstation. An access key may either be issued or rejected depending on enterprise security policy.

The key used to encrypt the file is stored in the local database and the encrypted data is stored in the cloud. Whenever a user tries to retrieve the data the encrypted file can be downloaded directly whereas to decrypt the downloaded file, a user needs to request for a secret key. Using this key, user can decrypt the downloaded file.

4.3.2 Proposed Model Design

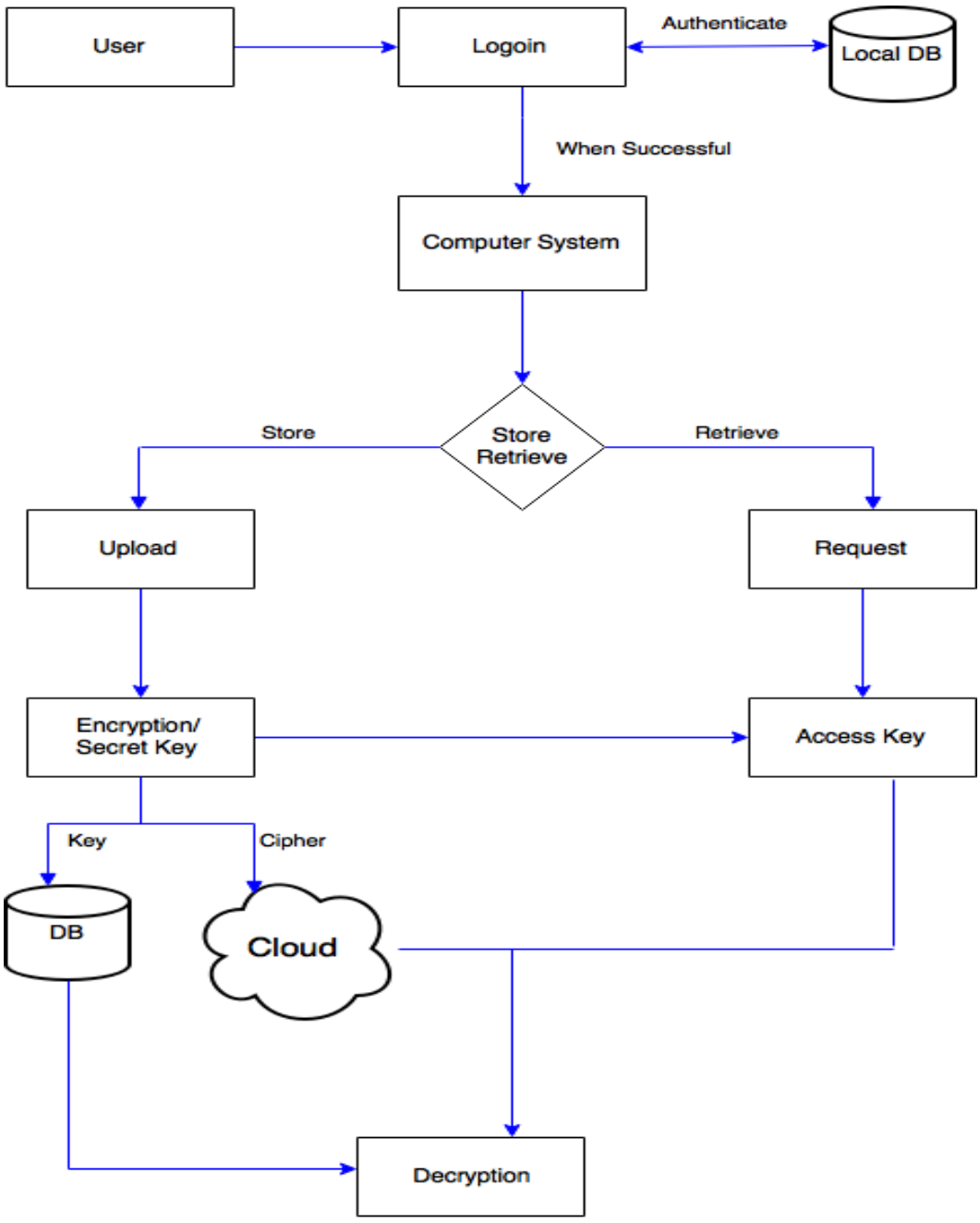

Figure 4.2 Proposed Model Design

4.4 Data Flow Diagrams

Other books refer to Data Flow Diagram (DFD) as process Model. It's a form of an analysis used to depict the flow of inputs through a system or a group of interrelated processes to their respective outputs. In DFD, we have mainly four types of symbols constituting the process. These include the process, data flow, external entity and data store where some authors refer

to it as simply database. DFDs represent what a system does. System analysts generate DFD after discussion with users to verify that their requirements are well captured and interpreted. It is used to model an existing or proposed system.

Data Flow Diagram

Figure 4.3 shows Data Flow Diagram for the model showing the flow of process between the components. A file is encrypted at the user computer before upload. Whenever the user uploads a file, the encryption algorithm will generate a secret key. A secret key is generated during encryption of a file. The key is used for both encryption and decryption. A user needs an access key to access a file from the cloud. The system administrator controls this key. Secret key is stored in the local database and the encrypted content is stored in the cloud servers.

When user downloads the data, a flow process would start from downloading the encrypted file. The encryption algorithm used here is symmetric. The symmetric decryption process is used to recover the file.

.

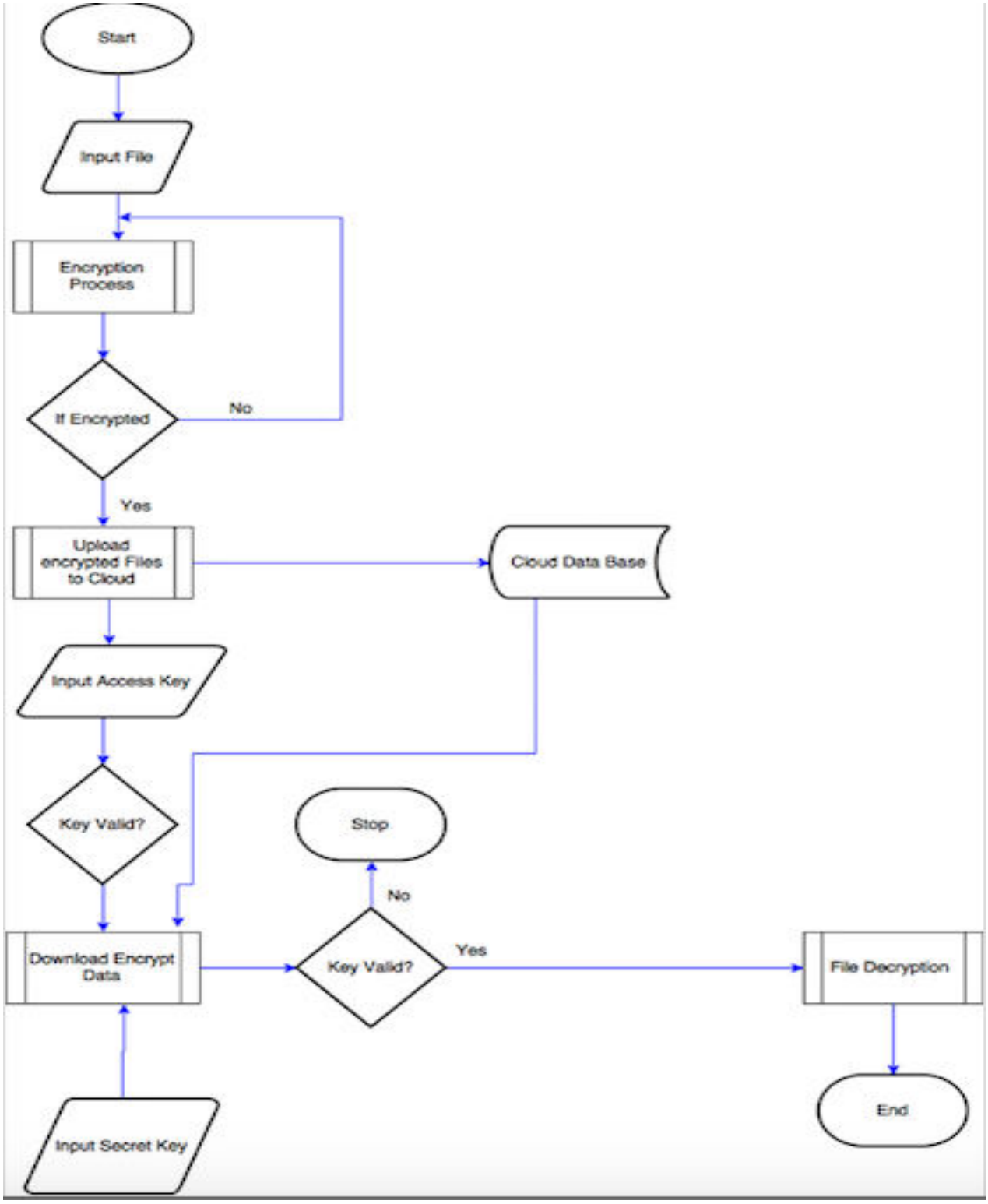

Figure 4.3: Model Data Flow Diagram

4.5 Unified Modeling Language Diagrams

This topic presents Unified Modeling Language (UML) diagrams for the logical design to represent the proposed model under study. The diagrams provide a standard technique to build the model blueprints while linking the concept and programming language that will be written in PHP, MY SQL database schemas, and other components required in building and deployment of the final product.

4.5.1 Use Case Diagram

In system modelling, it depicts the behaviour of the system as it interacts with a user. Use cases represent the activities or the functionalities. The use case diagram in Figure 4.6 describes the various actors and processes that are involved in the system. It represents the actions that are performed by one or more actors in the pursuit of a particular goal. The main actors are the users and the components are the functions performed by the users in authentication, encryption, upload, and store, download encrypted data and finally decrypts.

Register

Before a user is granted access to the computer system, he will be required to register his profiles such as your names, user ID, Email, Mobile Number, Date Of Birth, Gender user name and password. The profile authenticates the user.

Log In

On successful registration, a log in form is displayed. This allows the user to authenticate into the enterprise user domain. This procedure configuration depends in organizations security policy as set out.

Encrypt Files

This is where an authenticated user is allowed to encrypt the file for upload to the cloud. Encryption is done at the user side in a user computer. The choice of an encryption algorithm depends on the sensitivity and performance requirements of data. The user decides these two parameters.

Key Generation

A secret Key management takes place at the local user machines. This includes key generation, distribution and storage.

Access Key

Access key is the key required by a user to access files in the cloud repository system. Only authenticated users are issued the access key.

Decrypt

Files are downloaded from the cloud storage in an encrypted format. A user will need a secret key to decrypt the file. Only authenticated users are allowed access to the key. Apply correct secret key on downloaded encrypted file.

Cloud

The cloud represents outsourced repository system. This is where user data is uploaded and stored in an encrypted format.

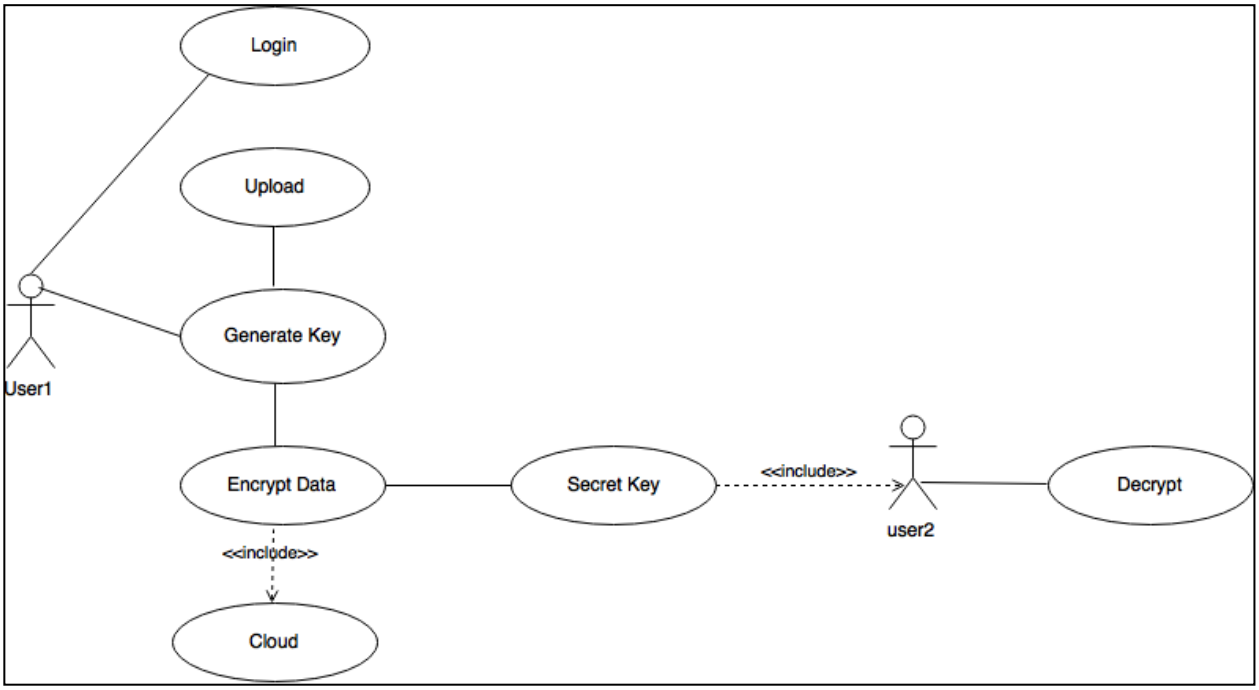

Figure 4.4: Model Use Case Diagram

4.5.2 Sequence Diagram

A sequence diagram shows different parts of a system working in a sequential manner to get something done. Sequence Diagrams (SD) is used to show objects interactions and flow messages. For this design, six objects appear along the top margin. The objects include; user interface, server, key generation, encryption, decryption and cloud. SDs are used to model interactions between objects in a single use case illustrating how different parts of a

system interacts with one another to execute a particular function. It also shows the order that interaction happens when a particular case is implemented. SD represents a timeline that begins at the top descending downwards to mark the sequence of interactions. Arrows represent message exchange between objects while lifeline rerepresent the different objects or parts that interact with each other in the system during the sequence. The activation bars are placed on the lifeline representing that the object is active.

Figure 4.5 shows the sequence diagram for the activities of this model. The SD in this model consists of six objects with both forward and return arrows. They include the user interface which provides graphic user interface which, interacts directly with the user, local application and database server which host local application programs and data store, the key, encryption, decryption and cloud which is located at the cloud provider side and where encrypted data is uploaded and stored.

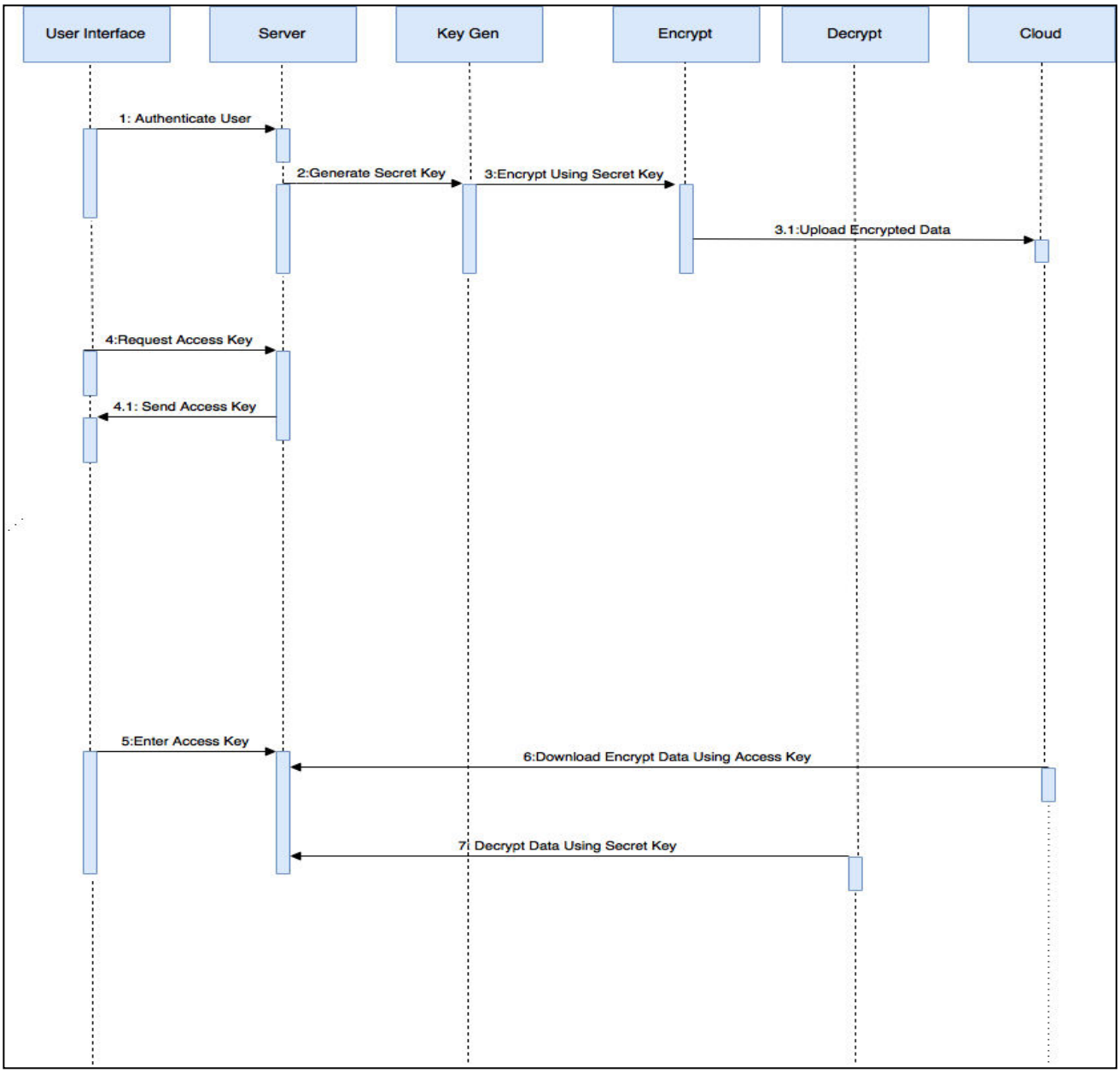

Figure 4.5: Sequence Diagram

4.5.3 Activity Diagram

Activity diagrams are used for modeling behavioral logic like business processes or work-flow. They describe a dynamic aspect of a system. They represent the flow of data from one particular operation of a system to another. It consists of a control flow, which is always drawn from one operation to another. The model has two activity diagrams, one for upload and the other for download of encrypted data.

Figure 4.6 shows an activity diagram for the uploading of a cipher text to cloud repository system. In this figure, registration module allows a user to register in the system computer domain. Login module allows a user to log in. If a user is authenticated through a username

and password, he proceeds and encrypts the file and finally uploads it to the cloud servers, else the activity ends.

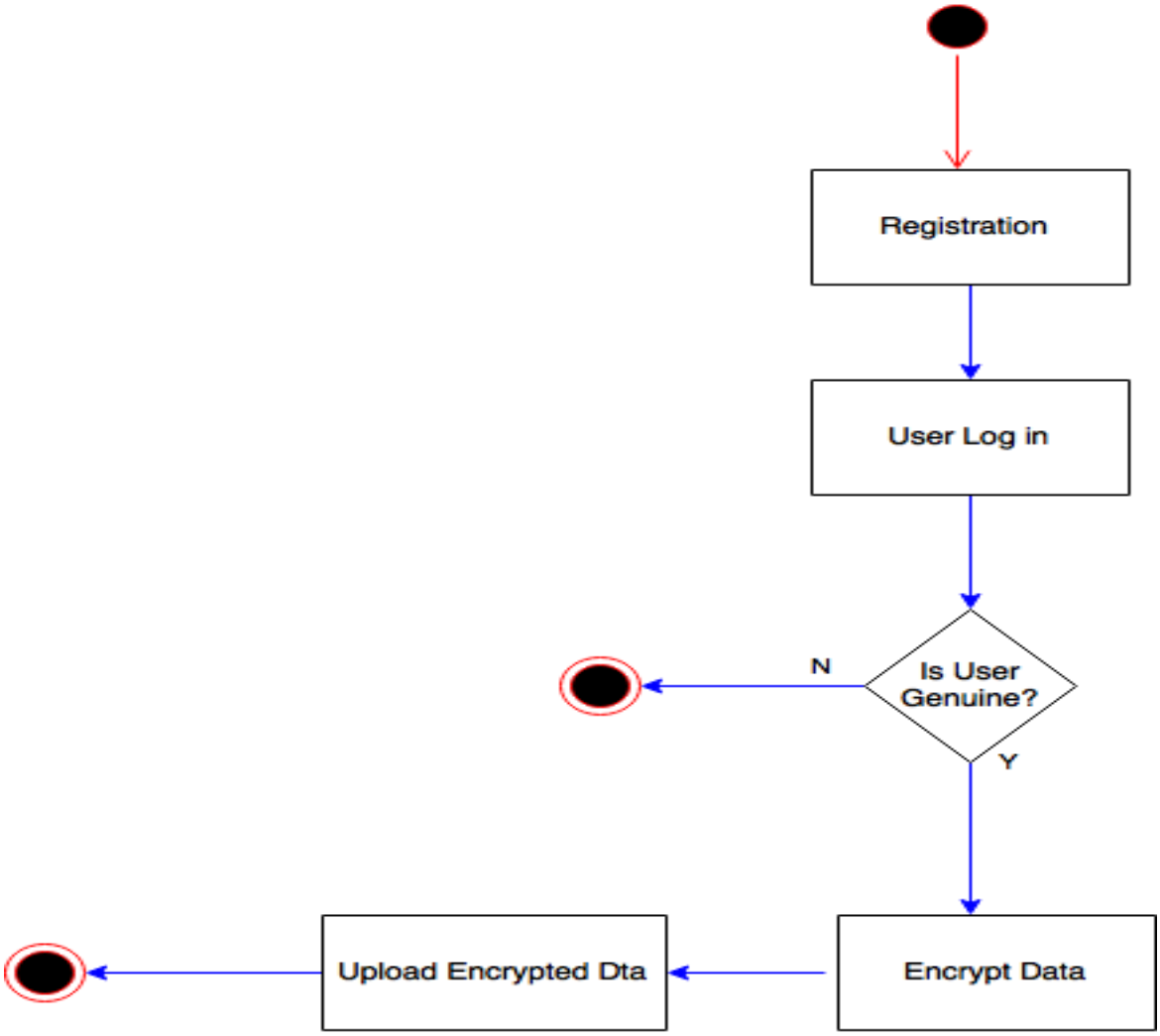

Figure 4.6: Upload Activity Diagram

Figure 4.9 shows an activity diagram for downloading cipher text from the cloud servers. Registration module allows a user to register with system domain. The registered user then logs in through user module. Once a user is authenticated, he receives an access key to enable user access the cloud storage servers. If user receives an access key, he accesses the storage servers and download the cipher text, else activity diagram ends.

Decryption module is implemented at the user computer domain. The module decrypts the downloaded cipher text using the secret key.

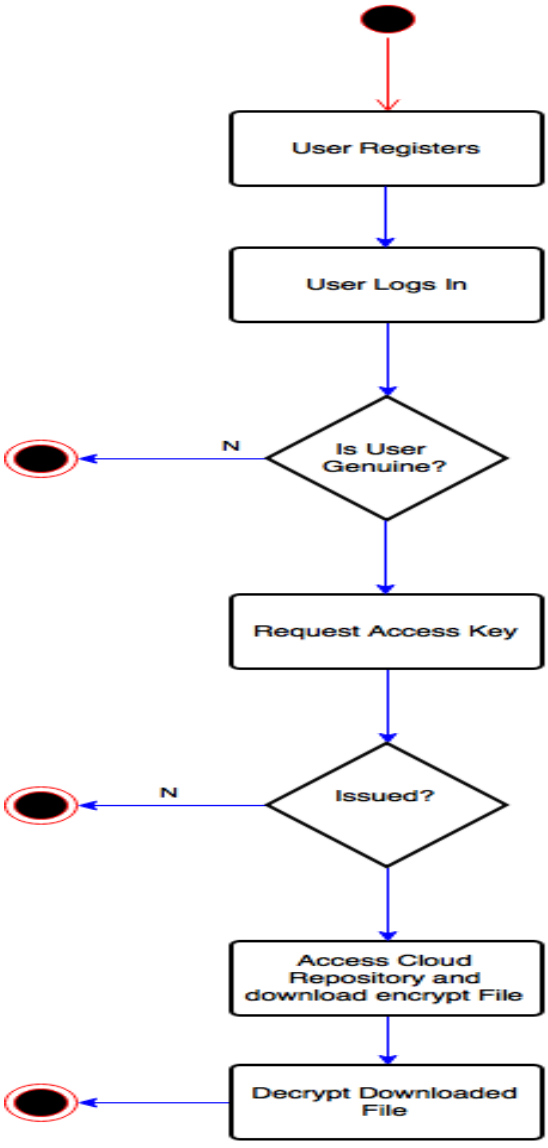

Figure 4.7: File Download Activity Diagram

4.6 Evaluation of Symmetric Key Algorithms

In this model, an analysis tool determines whether an algorithm is suitable for a particular class of data. It measures the performance of an encryption algorithm. The tool is capable of analysing plaintext inputs. More importantly this tool can be extended to evaluate combined algorithms as well as new symmetric key algorithms. The other idea for incorporating this tool was to give a numerical output to depict the response times for a fixed text sample rather than simulating some known attacks. This reduces the complexity of the tool and makes it more users friendly. If someone introduces a new symmetric key algorithm the tool can be used to analyse its secrecy and performance, (Weerasinghe, 2014).

4.6.1 Analysis of the Tool

The choice of an encryption algorithm is very critical in the design of this model. Performance and security level are key parameters that determine the choice of encryption algorithm to use. To assist the user selecting a particular encryption algorithm to balance between performance and security, it was prudent to incorporate a tool for evaluating the performance level of existing encryption algorithms. The tool was developed to assist in identifying the best performing algorithm for use in this model based on user requirements.

Average performance analysis was performed using the tool. Several numbers of experiments/tests were considered to calculate the average values and obtain reasonable data. The objectives of the experiments were to test the performance and security levels. The results were compared to some data set obtained by various researchers who carried out experiments of similar nature. The results obtained had reasonable level of reliability with respect to performance in the context of this study.

4.6.2 Scalability of the Tool

The tool has the ability to enhance its scope by adding new functionality at minimal effort. This tool can have its scope extended if one wants to analyse a newly implemented symmetric key algorithm in Java. Whenever a new algorithm that is not under the scope of this tool is developed, this tool can be improved to enable it perform similar tests on the algorithm. This tool will be helpful to the users to evaluate the secrecy and throughput of encryption algorithms. It will help the users to obtain numerical values for the secrecy of ciphers.

4.7 Functional Requirements

This refers to what the model can do in fulfilling the user objectives with an improved performance, affordable cost without compromising quality. The main functionalities of this model are listed below

i. User Authentication
ii. Upload encrypt data
iii. Providing confidentiality to the data stored in the cloud
iv. Restricting access control levels
v. Requesting access for encrypt data
vi. Reducing the cloud security as a service cost
vii. Maintaining logs of downloaded files
viii. Non-Functional Requirements

The model is designed with flexibility to enable it being responsive and adaptive to change of user requirements and environment. Scalability; the model provide for future expansion as level of security threats changes. It enables SMEs to change encryption algorithms based on threats and performance requirements. On usability, the model is easy to use, as SMEs only need to install a hybrid encryption tool on their user workstation. The security function of the model is improved in the sense that all encryption and decryption is undertaken at the user side. Key generation and storage is at the user side. Security control is under the data owner. In terms of securing users data stored in the cloud repository system, this model provides a reasonable data confidentiality protection.

Chapter Five: Implementation, Testing And Analysis

5.1 Introduction

This chapter shows how the model was implemented and validated. Moreover, it provides the analysis of the test scenario results. Its worth noting that the chapter captured some results obtained by other researchers showing results obtained from evaluation of encryption algorithms on the basis of response times and security level. Security and performance levels of algorithms were key parameters in choosing an algorithm to use.

5.2 Application Development

The scripting language used was java while the database management software used was MySQL. All source codes were built on Netbeans version 8.2 platform. APACHE was used as an actual web server application with the function of processing and delivering web content. It was preferred mainly because its easy to use and an open source web server. Hypertext Pre-processor (PHP) was used as a server side programming language. PHP was used because it's a free source and its platform independence features.

5.3 Application Modules

The Application Modules for the model are as follows:

Registration and login

This module allows first time users to register and log in to the model for authentication to use the model. In the registration page as shown in appendix, a form will be displayed to the user where valid information needs to be filled in the provided fields with a generated unique user id. If the registration is successful, the user is redirected to the login page prompting successful registration.

Encryption Files

The evaluation analysis application was used to encrypt a sample text using triple Data Encryption System (3DES), Rivest Cipher 4 (RC4) and hybrid (3DES+RC4). The file was uploaded to the cloud storage. The application allows a selction of an ecryption algorithm based on security level of a file. This application was also used to test performance of an algorithm on a given fixed file size (.txt file).

Uploading Files

This module allows the user to either upload text files to the cloud repository system. The

text files are first encrypted before being uploaded and stored in the cloud servers. The encryption algorithm selected is based on a balanced response time and level of security determined the encryption application. While uploading, the user needs to assign the file name and upload it. When a user clicks on encrypt button a secret key will be generated. This key is used for converting plain text into cipher text.

Downloading Files

A user can download his/her files directly from the "FILES" page and the requested files can be downloaded in a client computer, laptop, IPad or a smart phone. From the cloud side menu, a user can download or share a file. In order to download files, a user need to navigate to the "FILES SCREEN" and select the encrypted file. User will then click the download button from the cloud menu. The file will be downloaded to the user computer in an encrypted format. Once downloaded, a secret key is used to decrypt the file and recover plain text. This module allows a user to download encrypted files when he has an access key issued by the system administrator.

Decryption Module

This module allows a user decrypt the downloaded file. This happens in the user computer. A secret key is used to decrypt the file downloaded from the cloud.

5.4 Testing and Evaluation

Functionalities of the model were tested to verify whether the objectives were met. Test scenarios were carried out to evaluate the model. The purpose of this test was to evaluate the implementation of the model to establish whether all the objectives have been achieved. All the modules constituting the system were separately tested. Final test was performed in an integrated model. The integrated testing involved setting up a user cloud environment.

Test Plan

Unit testing, integration testing and complete set up testing were carried out on the proposed model. Each module was tested separately to validate quality and compliance to objectives. The modules were then tested in an integrated set up. This has been geared towards convincing the researcher and users that the model is good enough for operational use. It was also intended to build confidence in the model.

Test Scenario One

The objective of this test was to evaluate performance of existing algorithms. The encrypter application source code was written in java programming language. It was used to test performance of 3DES, RC4, and Hybrid (RC4+3DES). The experiment was repeated several times to obtain consistent results. The security levels of major encryption algorithms are already known having been carried out by previous researchers with interest in cryptography. The main difference between this tool and other primitive tools is that the tool gives a numerical output to show the response time of each cipher rather than simulating some known attacks when testing security level. The main source packages include the following:

i. Java programming language
ii. Netbeans IDE version 8.2
iii. Xampp
iv. Encrypter
v. File Read
vi. Login
vii. Registration
viii. Get Current Date
ix. Hybrid encrypter
x. Triple DES
xi. RC4
xii. Hybrid

The system requirement for developing and building this tool includes;

i. Intel® CoreTM i3 CPU
ii. Processor M370 @ 2.40 GHz
iii. 2GB usable RAM
iv. Microsoft Windows 7 Home Basic (32 bit) Operating System

The results of the experiment for a fixed input txt file were recorded in Table 5.1 and Table 5.2. The results were compared with other datasets obtained by experiments performed by other researchers on evaluation of performance and security level of encryption algorithms.

Table 5.1: Algorithm Vs Performance

ENCRYPTION	RESPONSE TIME										
Algorithm	**Avg**	**T1**	**T2**	**T3**	**T4**	**T5**	**T6**	**T7**	**T8**	**T9**	**T10**
3DES	**21.4**	21	23	20	23	20	24	20	22	21	20
RC4	**8.9**	9	8	7	9	10	11	9	8	9	9
3DES+RC4	**24**	22	24	24	25	25	23	25	23	24	25

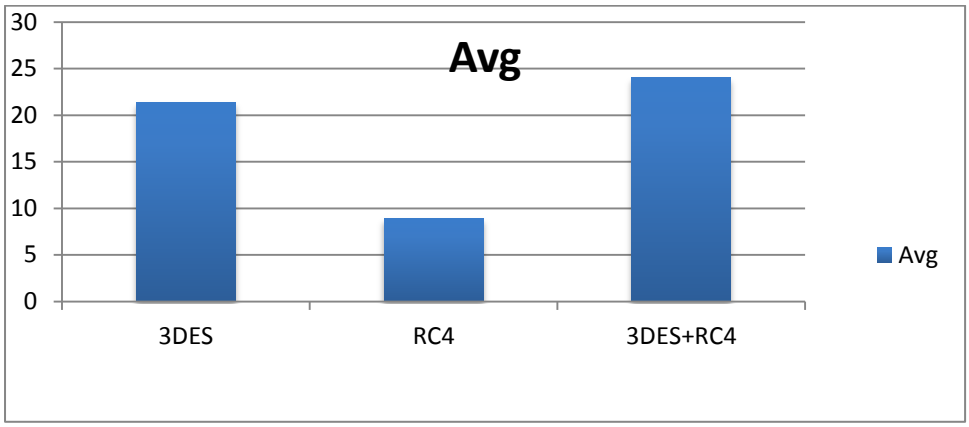

Figure 5.3: Algorithm Vs Performance

Table 5.2: Encryption Time Comparison

Adapted from Arockiam et al. (2015)

Size	Encryption Techniques			
	DES	**3DES**	**BLOWFISH**	**AROcrypt**
1MB	502	618	397	282
2MB	967	1078	602	468
3MB	1302	1422	891	656
4MB	1701	1847	1073	889
5MB	2108	2236	1207	1102
10MB	4282	4404	2421	2253
15MB	6331	6597	3642	3388

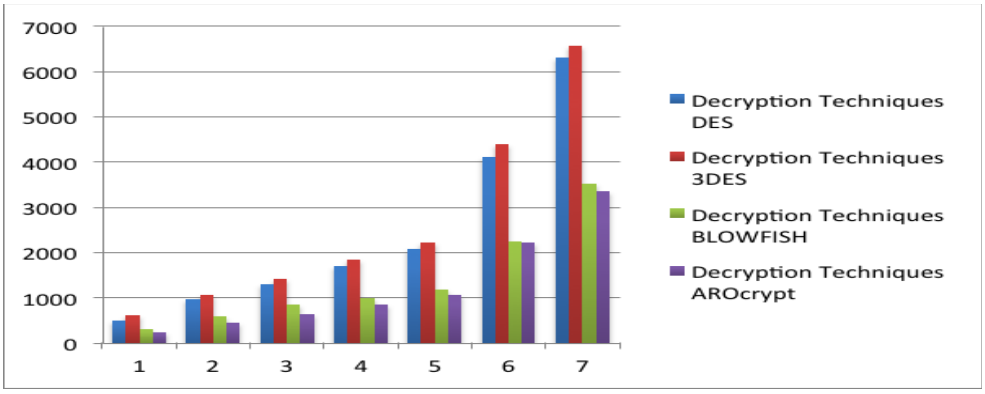

Figure 5.4:Performance Comparaison based on Encryption

Table 5.3: Decryption Time Comparison

Adapted from Arockiam et al. (2015)

Size	Decryption Techniques			
	DES	3DES	BLOWFISH	AROcrypt
1MB	497	607	312	235
2MB	958	1062	592	438
3MB	1289	1403	837	627
4MB	1689	1832	996	843
5MB	2084	2219	1170	1069
10MB	4116	4391	2231	2216
15MB	6298	6578	3512	3341

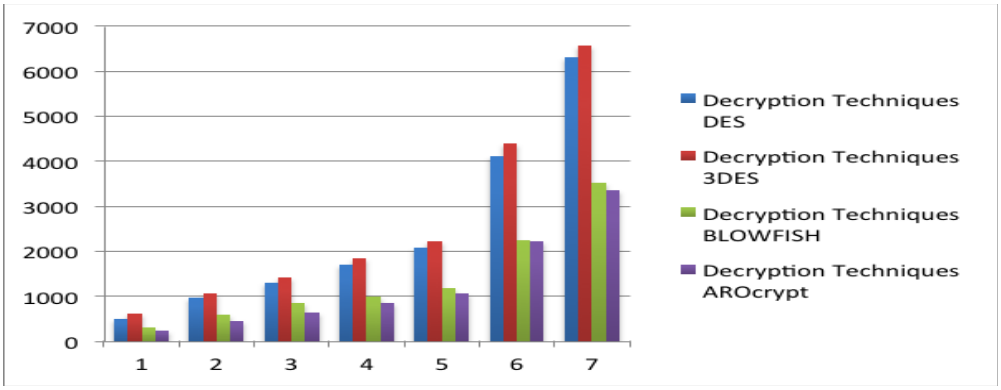

Figure 5.5: Performance Comparison Based on Decryption

Table 5.4:Security Level

Adapted from Arockiam et al. (2015)

No.	Algorithm	Security level (%)
1.	Blow fish	78
2.	3DES	82
3.	DES	74
4.	AROcrypt	89

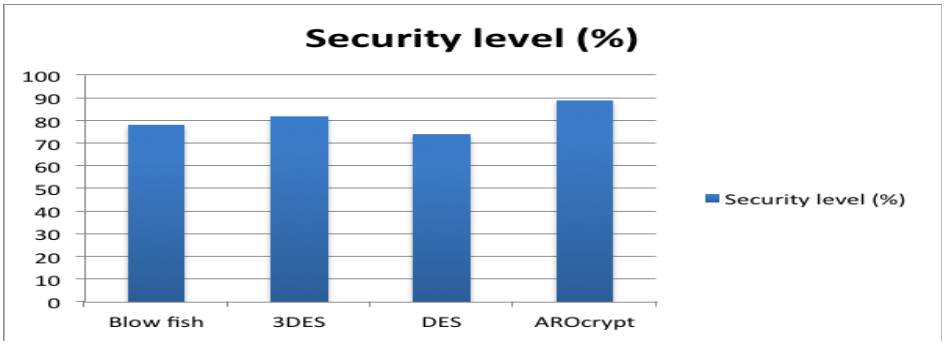

Figure 5.6:Comparison of Security Level

Test Scenario Two

In this test scenario, user authentication was tested in registration page and login page. Test involved validation for user credentials. In addition, validations were performed for empty fields for login page and registration page. Validations for appropriate information were verified in the registration page where error message was displayed when authentication failed.

Test Scenario Three

Cloud repository system client configuration was used. This test scenario was conducted on a complete integrated model with all modules integrated. User computer was configured with a cloud repository system to test the end-to-end encryption. We used Google's One drive cloud platform. In this test, a user logged into the local machine and connected to the Google's One drive cloud. Using the developed encryption application, a file was encrypted from a user local computer and secret key generated. The encrypted file was then uploaded to Google's One drive cloud platform and stored in an encrypted format. The file was then downloaded to

the user machine and decrypted using the secret key. This test was aimed at validating end-to-end encryption.

Test Scenario Four

This scenario was used to test the ubiquity, location independency and interoperability. In this experiment, a phone accessed the One Drive cloud repository system to access an encrypted file initially uploaded. The main objective was to test platform independency of the model and ubiquity. The cloud server in this case uses windows operating system while the smartphone used android operating system.

Test Scenario Five

In this test case, multiple functionalities were tested i.e. requesting access to the cloud repository system, downloading encrypted files and testing access controls. The main objectives of this test were to evaluate the model security.

Chapter Six: Discussion

This topic involves the interpretation and explanations of the results from the entire research. It answers the research questions as well as discussion in details metrics such as critique of the key concept and justification of the ideas mentioned. The discussion follows from the results and also relates back to the literature review. The five different scenarios concerning the style of interaction (between the service user and CSP) presented were analyzed in comparison to current user cloud interactions. The topic also discusses the experiment carried out on performance and security evaluation of encryption techniques.

6.1 Confidentiality

Unlike other models proposed earlier, this model presents enhanced data confidentiality in the cloud. In this model, secret key development, encryption and decryption takes place at the user machine end. End to end encryption is maintained in this model thereby improving user trust on the use of cloud repository system. Compared to other cloud computing confidentiality protection models, key management is controlled by the data owner and is done at the local user domain. The cloud service providers are not afforded complete trust in the encryption key management. Scenario One allows the data owner to choose the type of encryption to use depending on degree of security or performance response.

6.2 User Empowerment

In each of the scenarios, service users have control over who precisely can encrypt and decrypt their data, and by extension how it can be used. They have become empowered. The user can create numerous level of access as a form of control to further enhance data confidentiality. Security is not provided as a service in cloud repository systems relieving the cloud provider autonomy of encryption of user data. Finally, test scenario one offers the user complete freedom over the composition of the data attribute and moreover, the service user is also in control of who is able to obtain decryption secret keys for the downloaded data.

6.3 User Responsibilities

With test scenarios, the cloud provider is not afforded complete trust in their abilities to secure user's data. In all scenarios, user data has no trust. The data owner is responsible for all matters, excluding storage, arising from the use of the proposed data confidentiality model. All Scenarios offer a solution that allows the service user to have some trust in the cloud service provider. The cloud provider responsibility is just to provide repository system for users encrypted data.

6.4 Cloud Provider Service Level agreements.

Most Cloud providers provide service level agreement (SLA) is mainly based on availability of the cloud to users needs. There is no SLA that explicitly touches on data confidentiality. This means that data is actually stored in unencrypted format as mentioned in chapter one of this thesis. This model provides end-to-end encryption that ensures that data is first encrypted then uploaded to the cloud, stored and downloaded to user machine in its encrypted format. This way, data confidentiality is assured. Cloud security mechanisms if any, will form another security layer for data protection.

6.5 Flexibility

The proposed model provide for flexibility on the use of an encryption algorithm that suits the performance and security requirements of a user. If security is provided as a service by a cloud vendor, the cloud owners themselves without any user input decides on the encryption algorithm to use it because they are only guided by an SLA. Any change on an encryption algorithm has to be negotiated to be included in the SLA. Moreover the user may not know whether the right encryption algorithm is applied. Also in this proposed model, user does not have to encrypt all his data as some uncontrolled data can be uploaded as plaintext.

6.6 Secret Key Encryption

This model uses same key to encrypt and decrypt user files. Secret key symmetric algorithm is one of the most secured cryptographic models. In this proposed model, key management is in control of the user. He preforms key generation, key storage, key generation and key distribution to legitimate users. Current cloud computing providing security as a service takes responsibility of secret key management, meaning that they generate, distribute and store encryption keys. With this kind of key management, cloud service providers can easily have unauthorized access to user data without the knowledge of the user.

6.7 Server Authentication

This model has several layers of security. Local application and database servers are used to authenticate users before they register into the office domain. Once a user is allowed log in, he will require an access key from the database to be able to access the cloud repository system. This is a second security layer. Again once a user downloads an encrypted data, he will need a secret key to decrypt the downloaded file. This forms the third level of a security wall.

6.8 Data confidentiality

According Xiaojun, and Qiaoyan (2010), data security is all about confidentiality, integrity, authentication and availability. In security terms, confidentiality of data is a situation where the right persons only access data after successful authentication. From the testing and analysis of the proposed model, it's very true this model provides confidentiality protection of user data in cloud repository systems by incorporating an end-to-end encryption algorithm. There is a built in trust between data owner and the cloud-computing provider.

With all Scenarios secret key is tied to user. The user handles encryption and decryption operations. The service user, prior to uploading his data into the cloud, he will have to encrypt the data. The cloud service provider cannot decrypt data once it has entered the cloud. Finally, service users can decrypt data once it has been successfully downloaded. In this way, the proposed confidentiality protection model has an assured End-to-end confidentiality of the data.

6.9 Usability, Accountability

It's an easy model to implement, deploy, manage, use and maintain. From the architecture of this model, a user only needs to register into the computer system domain for him to be authenticated to use the model. Upon registering, a user is issued with user name and password to be used to login. Log in and registering module provide authentication. No additional user training is required to use this model. It is a plug and play implementation.

6.10 Ubiquitous Computing

Ubiquity computing is the ability to access computing resources of enterprises from any geographical location as long as there exist Internet connectivity. The model allows the use of the cloud repository system from anywhere and any time. It is device independency meaning that a user can use the model to access cloud platform from any device such as smartphone, IPad or Laptop that has Internet reach. The model does not limit its use to only office staff but also those staff on the move.

6.11 Experimental Results

The analysis involved data obtained from the experiment for the testing of performance and security levels of an encryption algorithm. The main objective was to obtain results to enable select the right encryption algorithm. The tool allowed testing through experiments on established encryption algorithms. The results were compared with previous similar experiments performed by other researchers as shown in Tables 5.3 and 5.4.

Each experimental result was compared with similar experiments done before by other researchers. The experiments were carried-out for all known and established block and stream ciphers. It's worth noting that a hybrid encryption has higher response time. This is simply because it's a combination of two different algorithm meaning that it averages the response times of two algorithms. However the security level of a hybrid encryption algorithm is higher as shown in Table 5.5. Theses results show that there must a balance between performance and security capability of an algorithm. Users will have to compromise one parameter either performance or security depending on requirements to achieve an objective. If the security is the priority, then performance has to be compromised else security is compromised.

6.12 Simulation Results

Arockiam and Monikandan (2015) performed performance and evaluation of encryption algorithms and proposed AROcrypt technique and key generation, which they implemented in JAVA. The tests were carried out in Amazon Elastic Cloud (Amazon EC2), which was connected to a user machine running on windows operating system. Response times for encryption and decryption were recorded as shown in Table 5.2 and Figure 5.3. Different block sizes of data were used. Time taken for encryption and decryption were taken. From results analysis, it was found that AROcrypt algorithm had best performance.

The two researchers also performed additional experiments to test security levels of different existing encryption algorithms. The results were presented as shown in Table 5.4 and showed that AROcrypt technique encryption algorithm had highest security level as compared to the rest. In both instances, performance and security experiments, AROcrypt technique came out the best above the rest

In this experiment Performance and security level of proposed AROcrypt technique was compared with existing encryption techniques. Performance was based on response times for both encryption and decryption. Security level was analyzed through the use of a security tool to test the security levels of each encryption algorithm. Technically, the tool was implemented in the Amazon server. The tool used different network attacks such as dictionary attack and use of brute force to try recover plain text stored in the cloud repository servers. After the attack, tests were carried out to establish the percentage of original plaintext retrieved. Similar tests were carried out on some existing encryption algorithms as shown in Table 5.4.

Chapter Seven: Conclusion and Recommendations

7.1 Conclusion

Cloud Storage is a cost-effective Information Technology service to the general user or enterprise customer. Most organizations and individual consumers do not have the infrastructure to keep their data safe. Cloud storage provides plenty of storage capability with nominal cost. Data Owners are interested in outsourcing their sensitive data to the cloud storage. However, there exist data confidentiality flaws within cloud computing storage servers. Due to this, data owners lack the courage to strategically use the cloud computing storage as a service. Once the trust issues are addressed through the deployment of this model where data confidentiality protection starts at local user machine, there shall be some attitude change on the side of data owners towards the need to adopt cloud computing because the trust gap between user and service provider will be minimized. Managers will consider making strategic decisions on the need to adopt cloud computing repository systems while saving costs on acquisition, ownership and maintenance of information technology storage infrastructure.

In chapter two of the literature review, we identified major drawbacks arising from using current data protection techniques in the cloud repository system. To mitigate threats to confidentiality in the cloud, this thesis proposes a data confidentiality protection model to address these drawbacks. In this model, end-to-end encryption of user data is maintained. Data is encrypted at the user local machine before it is uploaded into the cloud. Once encrypted, it is uploaded to the cloud repository system. While in the cloud repository, data remains in an encrypted format. When a user wants to access the cloud to download encrypted data, he shall require an access key. If the access key is the right one, user will access the cloud repository and download the encrypted data to the local client computer stationed at the user environment. The user will require a secret key to decrypt this data. This secret key is stored at local database and its distribution is controlled.

The proposed confidentiality protection model provides confidentiality of user data stored in the cloud, by encrypting user data before uploading into the cloud. As encryption consumes more processing overhead, many cloud service providers will have basic encryption applied only on few data fields. If cloud service providers are allowed to encrypt user data, then they can be able to decrypt and thereby steal information. To keep the cost low and maintain high sensitive data, it is recommended that users encrypt data at the user end before uploading to

cloud data storage servers. This model enables secret keys to be retained by the user. User takes the responsibility of encryption, key management and key storage.

7.2 Recommendations

Cloud Storage is a cost-effective IT service to the general user or enterprise customer. Most of SMEs do not have adequate budget to invest on the infrastructure required to keep their data safe. Cloud computing provide plenty of storage capability with nominal cost. The main reason why SMEs were reluctant to adopt cloud-computing platforms was fear that their data will land into the hands of unauthorized persons. This research has presented a model that will ensure user data confidentiality is maintained. The model provides end-to-end encryption. In this model Encrypted data is stored in cloud storage servers while data owner keeps secret keys. Access to cloud repository system is granted when a user is issued with an access key. Secret key is generated and kept by the end user unlike current scenario where the cloud service provider does the key management function. Secret key is not shared with third party service providers and data is available in encrypted format in cloud servers. This enhances data confidentiality. This model is highly recommended for use by SMEs and by extension cloud service providers as a marketing tool to encourage cloud tenants to use their storage as a service facility. Cloud providers will do this by cooperating with SMEs by opening up interfaces that will enhance this model functionalities and even help with future works on how to improve and enhance the model to gain customer trust.

7.3 Contribution of the Study

Since the objectives of this research study have been successfully accomplished, development of this model can be considered a great contribution in the field of computer security. This is because the model has overcome the existing data confidentiality threats in cloud computing by providing most secure and trusted cloud storages as a service to SMEs clients and other cloud consumers. The most notably contribution of this research is that it is going to benefit the cloud consumers and cloud service providers as well. Users will endeavour to adopt this cost effective and easy to use model to accrue advantages of cloud computing such as higher availability, redundancy, high confidentiality and disaster recovery. Moreover, cloud service providers will adopt this model to win back the customer trust. This research highly expects that the adoption of this model will encourage more and more SMEs to consider adopting cloud storage as a service to solve their investments from acquisition of

powerful computing infrastructure and spare their energy for use in their core business models.

The research thesis can be thought as one that adopted software engineering approach by designing and deployment of a secure cloud storage product. That is to say that this research has immensely contributed in software engineering by assuming the roles of software engineers for developing of a trusted cloud storage service. To overcome the research problem stated in chapter one, this research has managed to provide an improved and enhanced confidentiality protection model by designing as well as developing a secured cloud storage infrastructure for adoption by SMEs at a lower cost with a quick return to their investments.

7.4 Suggestions for Future Research

In future research, researchers should come up with a convenient and secured way to distribute secret keys to users. Further research should be conducted on how users can be able to work on the files in encrypted format in the cloud repository without the need to first download to user computers as this may reduce the overhead associated with Internet latency. Consideration should be given to develop new encryption algorithms with improved security levels and acceptable performance levels respectively as attackers keep on improving attacks such as dictionary attacks and brute force. It is also proposed that future work should focus on how we can have access key used for encryption, decryption and access for encrypted data in the cloud. This will reduce maintenance of secret and private keys.

.

References

Abduljabbar, Z., Jin, H., Zou, D., Yassin, A., Hussein, Z. & Hussein, M. (2014). An Efficient and Robust One-Time Message Authentication Code Scheme Using Feature Extraction of Iris in Cloud Computing. *Proceedings of the IEEE International Conference on Cloud Computing and Internet of Things (CCIOT), Changchun, China, pp. 22-25.*

Arijit, U., Debasish, J. & Ajanta, D.S. (2013*).* A security framework in cloud computing infrastructure*. International Journal of Network Security & Its Applications (IJNSA), Vol. 5, pp. 11-24.*

Aviram, A. S. Hu, Ford, B., & Gummadi, R., (2010). Determinating timing channels in compute clouds. *In Proc. ACM workshop on Cloud computing security workshop. ACM,* Arockiam, & L. Monikandan S. (2015). A confidentiality Technique for securing Enterprise's data in cloud. *International Journal of Engineering and Technology (IJET)*

Arockiam, L., & Monikandan, S. (2013). Data security and privacy in cloud storage using hybrid symmetric encryption algorithm. *International Journal of Advanced Research in Computer and Communication Engineering, Vol. 2, pp. 3064-3070.*

Basha, G., W., & Basha, V. (2015). Enhancing Cloud Security Using Multicloud Architecture and Device Based Identity. *Seventh International Conference on Emerging Trends in Engineering & Technology New York, NY, USA, 103-108.*

Chu, C. K., Chow, S. S. M., Tzeng, W.G., Zhou, J. & Deng, R. H. (2014). Key-Aggregate Cryptosystem for Scalable Data Sharing in Cloud Storage.

Chandra M.P. (2012). Evaluation of performance of the Symmetric Key Algorithms: DES, 3DES, AES and Blowfish. *Journal of Global Research in Computer Science, Volume 3, No. 8, pp. 67– 70.*

Chen, D., & Zhao, H. (2012). Data Security and Privacy Protection Issues in Cloud Computing. *International Conference on Computer Science and Electronics Engineering. IEEE*

Chunxiao, Li., Raghunathan A., & Niraj K. (2010). Secure Virtual Machine Execution under an Untrusted Management OS. *Third International Conference on Cloud Computing, IEEE.*

Eman, M.M., Hatem, S. A., & EI-Etriby, S., (2002). Enhanced Data Security Model for Cloud Computing. *The 8th International Conference on INFOrmatics and Systems (INFOS2012)*

Huang, D., (2010). Anonymous Certification Services. *Proceedings of GLOBECOM pp.1-6.*

Ikechukwu, V.U, & Ugochukwu, E.O. (2013). Building Trust and Confidentiality in Cloud computing Distributed Data Storage. *West African Journal of Industrial & Academic Research, Vol. 6 No.1, pp.78-83.*

Kothari C.R. (1990). *Research methodology, methods and techniques.* (Second revised edition)

Luca, F., Fabio, P., Michele, C., & Mirco, M. (2013). Security and confidentiality solutions for public cloud database services. *The Seventh International Conference on Emerging Security Information, Systems and Technologies, SECURWARE.*

Liang, M., & Chang, C., (2011). Full Disk Encryption based on Virtual Machine and Key Recovery Scheme. *Journal of Information and Computing Science Vol. 6, No. 3, 2011, 163-172.*

Mather, T., Kumaraswamy, S., & Shahed, L. (2009*). Cloud security and privacy. Chapter 4, O'Reilly Media, Inc., pp. 61-71.*

Mell, G.T. (2012). The NIST definition of cloud computing. *National Institute of Standards and Technology, U.S. Department of Commerce.*

Meiko, J., & JorgSehwenk (2009). On Technical Security Issues in Cloud Computing. *IEEE International Conference on Cloud Computing, pp. 109-116.*

Mariana, C., Alta van der M., Paula K. (2011). Secure Cloud Computing. *Benefits, Risks and Controls. IEEE.*

Mell, P., & Grance, T. (2011). The NIST Definition of Cloud Compu*ting. NIST Special Publication 800-145.*

Mulazzani, M., Schrittwieser, S., Leithner, M., Huber, M. & Weippl, E. (2011). Dark clouds on the horizon: Using clouds storage as attack vector and online slack space. *In Proceeding of the 20th USENIX Conference on Security.*

Nigoti, R., Jhuria, M., & Singh, S. (2013. A Survey of Cryptographic Algorithms for Cloud Computing. *International Journal of Emerging Technologies in Computational and Applied Sciences, Vol. 4, pp.141-146.*

Newton, D. (2011), *Dropbox authentication: insecure by design,* Available: http://dereknewton.com/2011/04/dropbox-authentication-static-host-ids/.

Popa, R. A., Redfield, C. M. S., Zeldovich, N., & Balakrishnan H. (2011*).* Crypt DB: protecting confidentiality with encrypted query processing. *23rd ACM Symposium on Operating Systems Principles, pp. 85–100.*

Rani, S., Gangal A., (2012). Cloud Security with Encryption using Hybrid Algorithm and Secured Endpoints. *International Journal of Computer Science and Information Technologies, Vol. 3, pp. 4302–4304.*

Rauber, K. (2013). Cloud Cryptography. *International Journal of Pure and Applied Mathematics, Vol. 85, pp. 1-11.*

Rabi, P., P., ManasRanjanPatra & Suresh C., S. (2011). Cloud Computing*: Security Issues and Research Challenges. IJCSITS Vol. 1.*

Sudha, M., & Monica, M. (2012). Enhanced Security Framework to Ensure Data Security in Cloud Computing Using Cryptography. *Advances in Computer Science and its Applications, pp. 32-37.*

Salvi, S., Sanjay, H.A., Deepika, K.M., & Rangavittala S.R. (2015). An Encryption, Compression and Key (ECK) Management based Data Security Framework for Infrastructure as a Service in Cloud. *IEEE International Advance Computing Conference (IACC).*

Wang, H., Li, Zhao, Qi, Q. He, Q & Sun, J. (2013). A scheme to ensure data security of cloud storage*. Proceedings of the IEEE International Conference on Service Operations and Logistics and Informatics (SOLI), Dongguan, China, pp. 79-82.*

Wen Fu, & Xiang Li. (2011). *The Study on Data Security in Cloud Computing based on Virtualization. ISSN 978-1-61284-704-7 IEEE.*

Weerasinghe, T.D.B (2014). International Journal of Information & Network Security (IJINS) Vol.3, No.1, pp. 26~32.

Xiaojun, Y. & Qiaoyan W. (2010). A View about Cloud Data Security from Data Life Cycle. *International Conference on Computational Intelligence and Software Engineering (CiSE), IEEE, pp. 1-4.*

Xiao Jun, Y., & Qiaoyan, W. (2010). A view about cloud data security from data life cycle. *IEEE International Conference on Computational Intelligence and Software Engineering (CiSE), pp. 1-4.*

Yau, S.S., & Ho G. (2010). Confidentiality Protection in Cloud Computing Systems. *Int J Software Informatics, Vol.4, No.4, pp. 35-136.*

Yau, S.S., & An, H.G., (2010). Confidentiality protection in cloud computing systems. *International Journal Software Informatics, Vol. 4, pp. 351- 365.*

Zhang, Qi, Cheng, Lu & RaoufBoutaba (2010). Cloud Computing: *State-of-the-art and research challenges. J Internet ServAppl.*

Appendices

Appendix A: Sample Code Segment

```
package encrypter;
class MyCustomFilter extends javax.swing.filechooser.FileFilter {
public boolean accept(File file) {
return file.isDirectory() || file.getAbsolutePath().endsWith(".txt");
        }
public String getDescription() {
return "Text documents (*.txt)";
        }
    }
public class Encrypter_ extends javax.swing.JFrame {
    public static String input, textInput, textKey;
public Encrypter_() {
        initComponents();
    }
public static void writeKey(String key){
        PrintWriter out;
        try{
            String fileName = JOptionPane.showInputDialog(null, "Enter name of file to save key
...");
```

```
            File file = new File("//Users//Admin//Desktop//My Project//Key// " + fileName+
".txt"
            out = new PrintWriter(file);
            out.println(key);
            out.close();
        }
        catch(FileNotFoundException ex)
        {
            Logger.getLogger(Encrypter_.class.getName()).log(Level.SEVERE, null, ex);
        }}
```

Appendix B: Screen Shots

Figure B.1: User Login Validation Validation

In the login page, a form will be displayed to the user as shown in Figure B.1. The user enters his credentials provided during registration User must be registered to log in. Credentials such as user names and password are used to authenticate an authorized user. Validations will be performed on the values entered in the fields. When a user clicks the login button a digest for the password entered is compared with the digest stored in the database server. The user can login if the username and password entered matches with the records in the database server or else error message will be displayed. If login is successful, the user can start the process of encryption plain text, uploading to the cloud, downloading and decryption.

Figure B.2: User Registration Validation

In registration module, the first time login user registers with the user computer system domain to use the model. The form enables creation of user profile. The profile is associated with the user during authentication. A form is displayed to the user where valid information is filled in the respective fields.

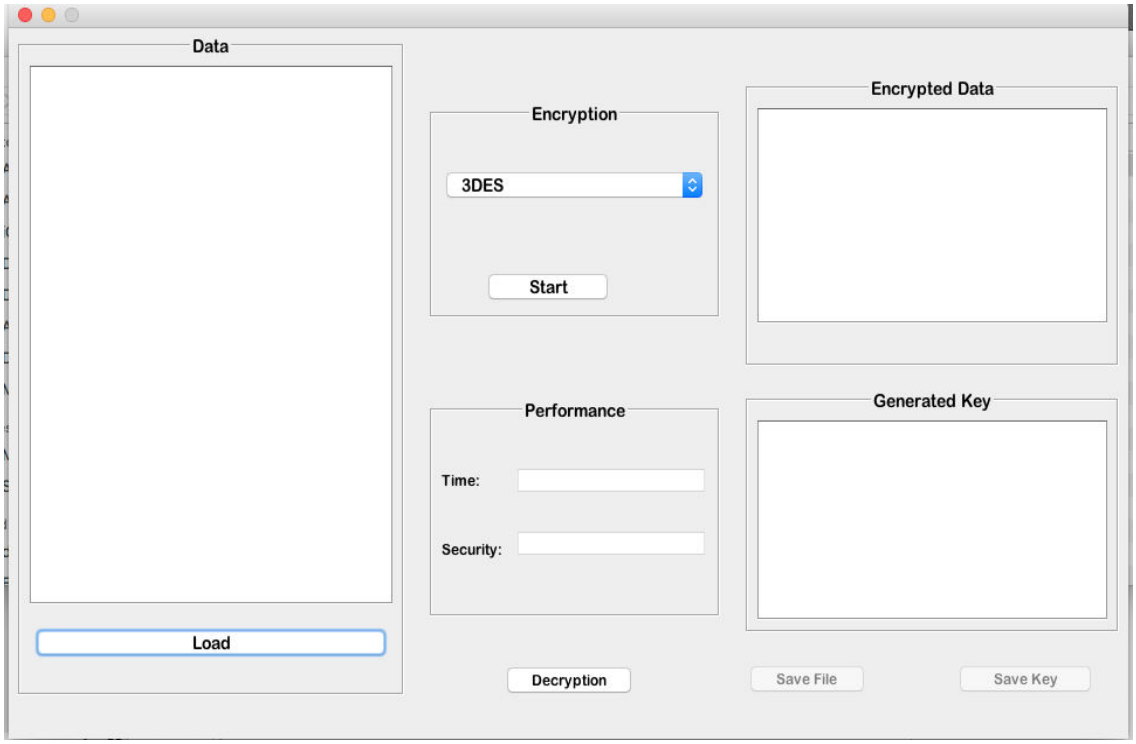

Figure B.3: Encrypter Performance Validation

This module is installed at the user end computer. The module demonstrates how the model works. Fields such as cipher text data, loading button, encryption algorithms selection, encryption performance, key generation and encrypted data are displayed. For validation purposes, a user loads a file to be encrypted by hitting load button. User then selects the encryption algorithm to use. The start button allows the selected encryption algorithm to encrypt the file. Encrypted file is displayed in the encrypted data window. During encryption process, a secret key is generated which the user saves. At the same time, the performance of the algorithm is determined in terms of response time. The form also, provides a button for saving an encrypted data before being uploaded to the cloud.

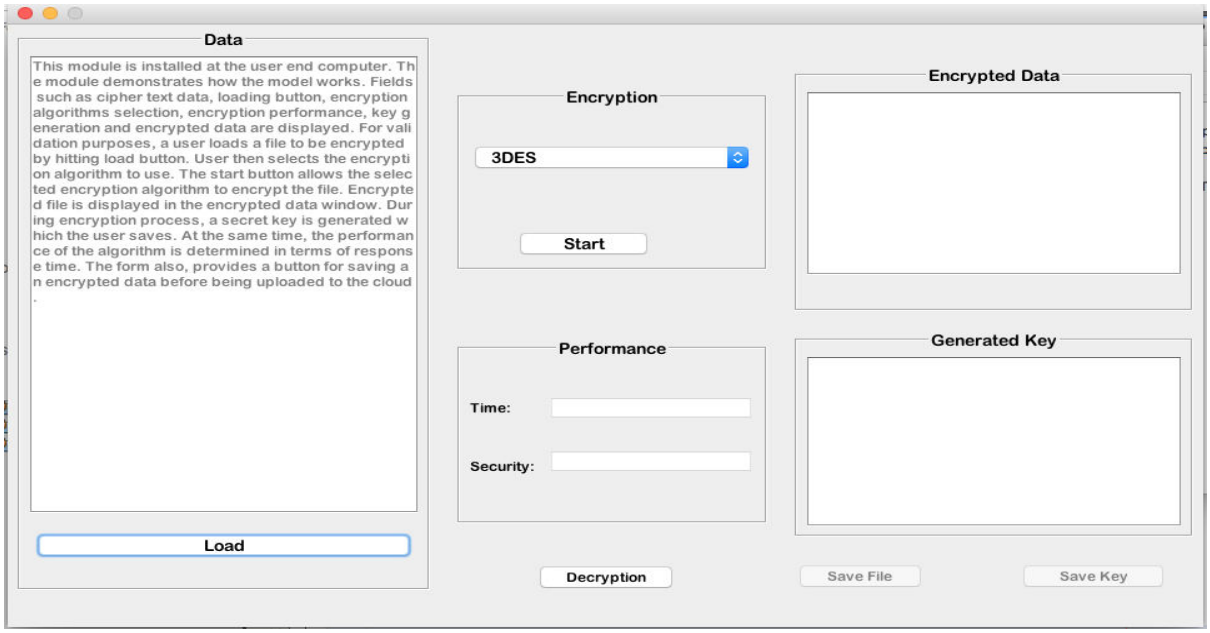

Figure B.4: Plain Text to Cipher Validation

This module shows the plain text file selected for encryption with a particular algorithm. The text is converted into cipher text by invoking the start button. Converted text is displayed in the encrypted data window.

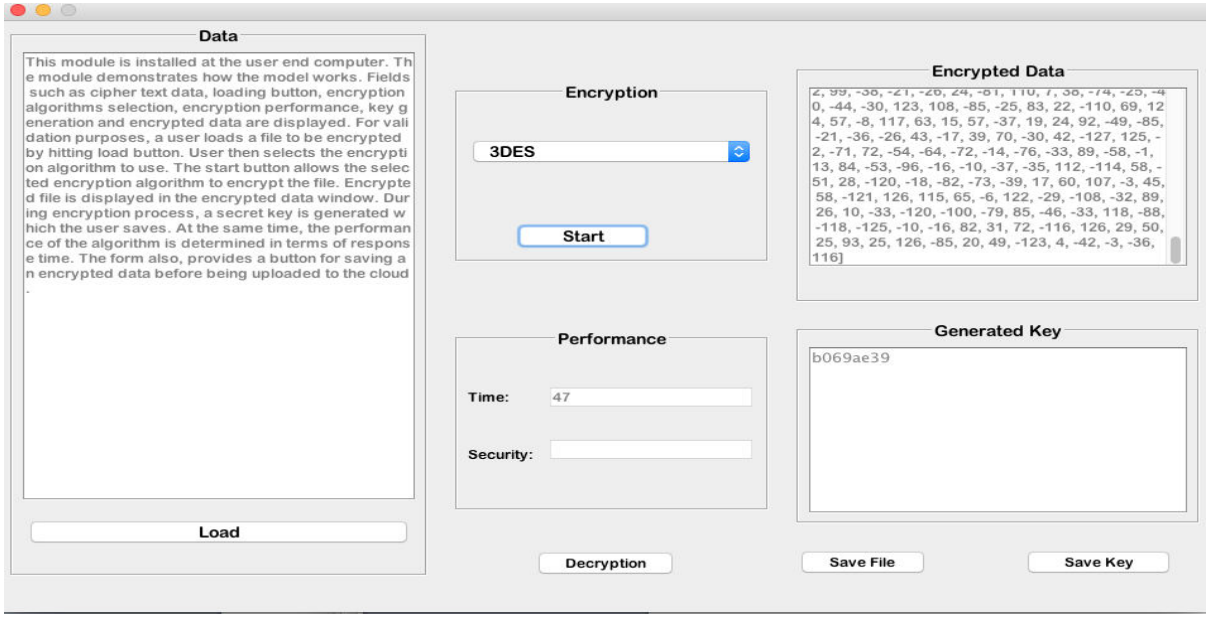

Figure B.5: Converting Plain Text into Ciphertext

The module displays a plain text file converted to a cipher text file with a corresponding key being generated. It also displays the response times for the 3DES algorithm used to encrypt the file. Pressing save file and save key buttons respectively saves the encrypted and secret keys.

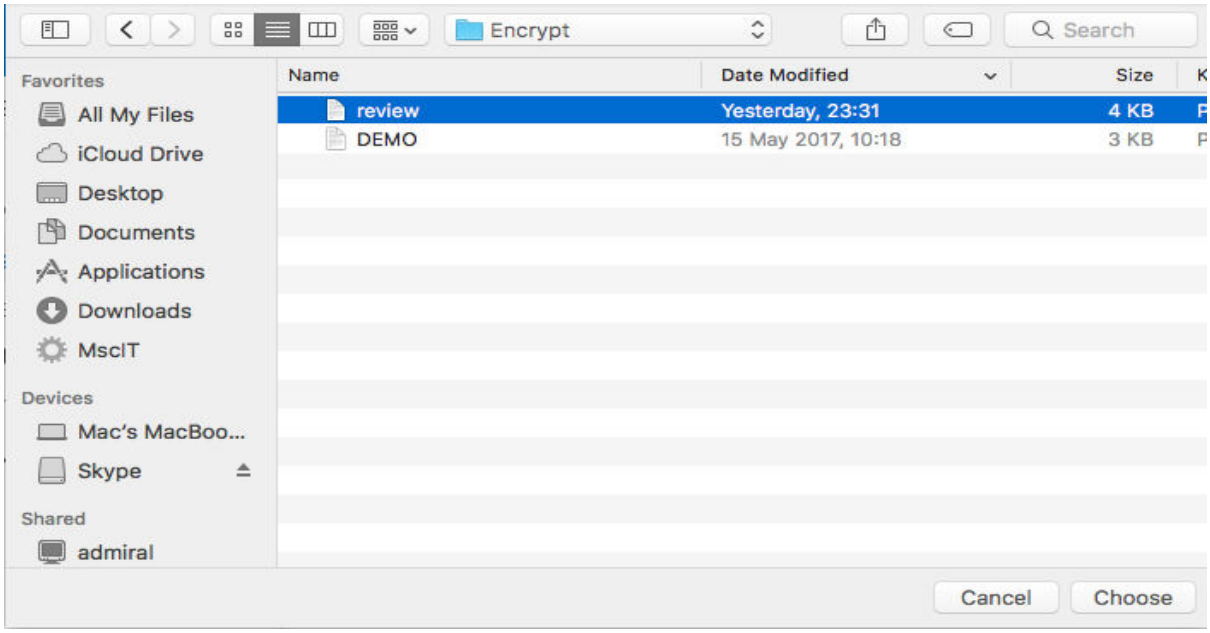

Figure B.6: Uploading File to the Cloud Validation

The module shows the encrypted file called review being uploaded to the cloud from a folder named encrpyt. In this validation, microsoft drive one cloud was used. The file is uploaded and stored in an encrypted format. The files are encrypted using a selected encryption algorithm.

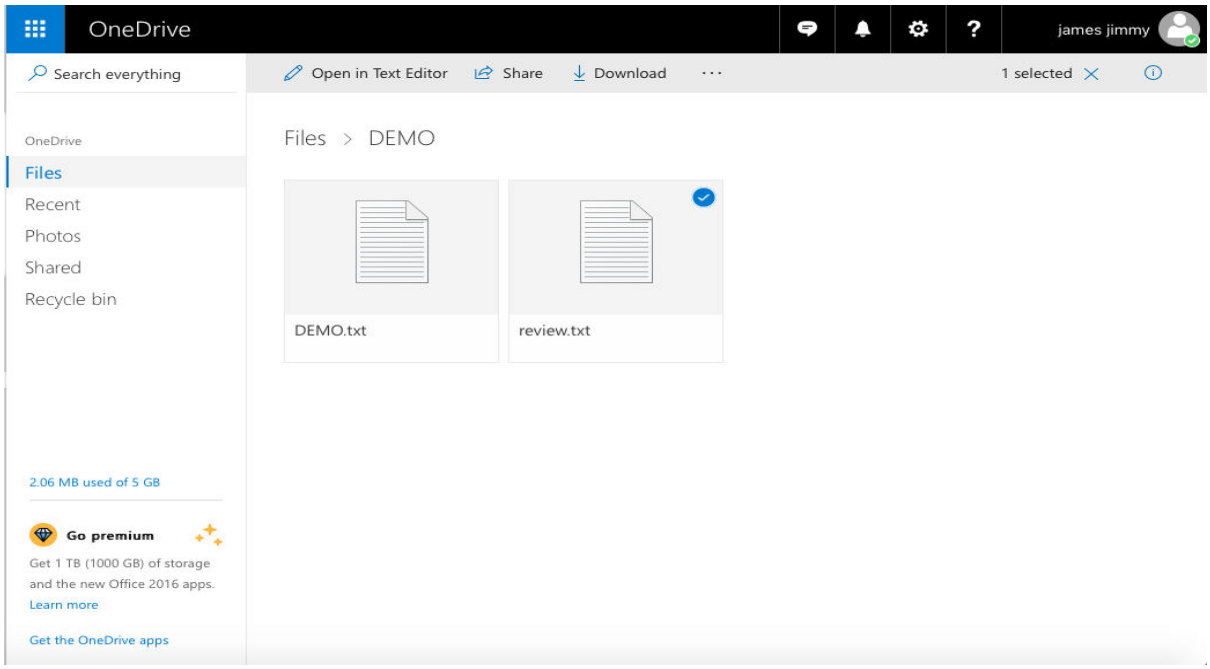

Figure B.7: File Storage in the Cloud Validation

The module shows uploaded encrypted file named review.txt stored in the cloud repository system under the folder FILES/DEMO. The file is stored in an encrypt format.

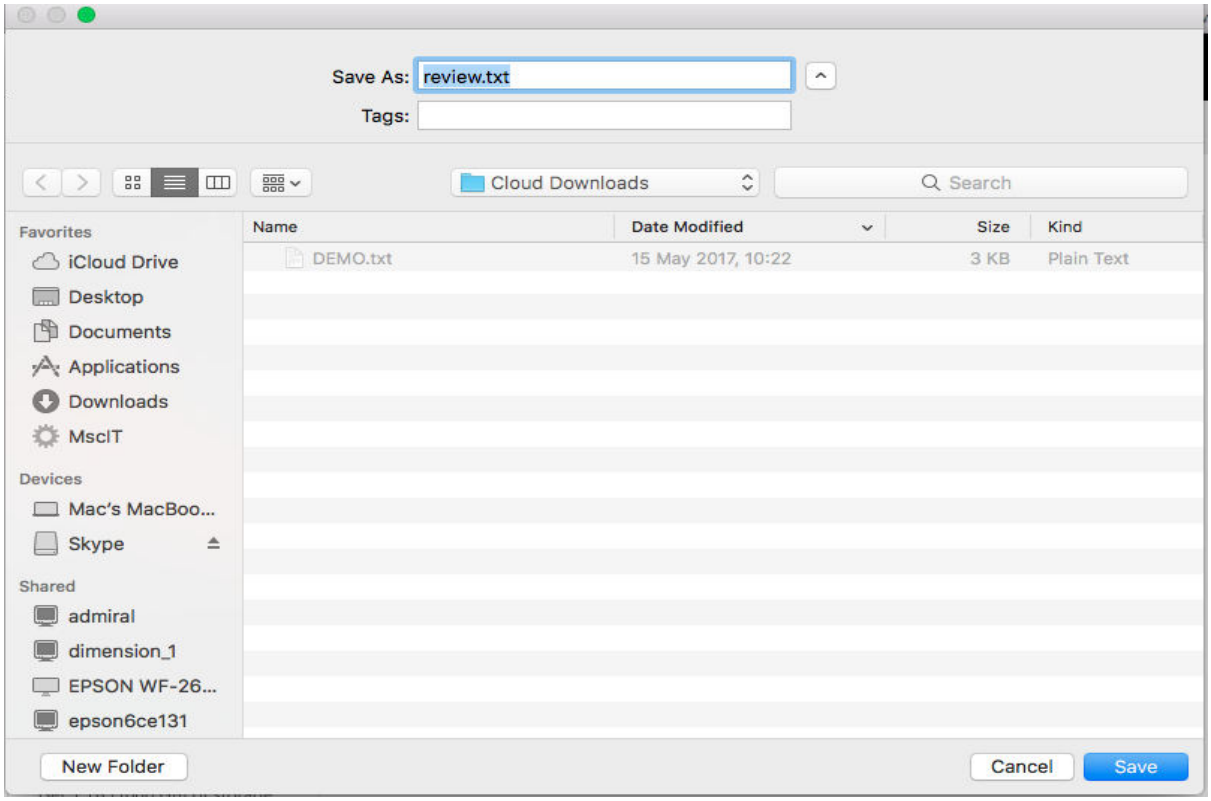

Figure B.8: Downloading File Validation

The module validates file downloading from the cloud to a folder named DOWNLOADS. In order to download the encrypted files from the cloud, the user needs to request an access and secret keys, from the data owner.

Appendix C: TurnitIn Report