KWARA STATE UNIVERSITY, MALETE

The University for Community Development

Faculty of Information and Communication Technology

DESIGN AND IMPLEMENTATION OF DATA CRYPTOGRAPHY SYSTEM

BY

OLANREWAJU TOYYIB OLAITAN

16/47CS/629

SEPTEMBER 2020

DESIGN AND IMPLEMENTATION OF DATA CRYPTOGRAPHY SYSTEM

BY

OLANREWAJU TOYYIB OLAITAN

16/47CS/629

A RESEARCH PROJECT SUBMITTED TO THE DEPARTMENT OF COMPUTER SCIENCE, FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY, KWARA STATE UNIVERSITY, MALETE, IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE AWARD OF BACHELOR OF SCIENCE (B.Sc.) DEGREE IN COMPUTER SCIENCE.

SEPTEMBER 2020

DECLARATION

I hereby declare that this research work titled **"Design and Implementation of Data Cryptography System"** is my own work and has not been submitted by any other person for any degree or qualification at any higher institution. I also declare that the information provided therein are mine and those that are not mine are properly acknowledged.

Olanrewaju Toyyib Olaitan	
Name of student	Signature and Date

CERTIFICATION

This is to certify that the research project titled **"Design and Implementation of Data Cryptography System"** was carried out by **"Olanrewaju Toyyib Olaitan"**. The project has been read and approved as meeting the requirements for the award of Bachelor of Science (B.Sc.) Degree in Computer Science in the Department of Computer Science, Faculty of Information and Communication Technology, Kwara State University, Malete.

_____	_____
Dr. R. M. Isiaka	Signature/Date
Supervisor	
_____	_____
Prof. K. A. Gbolagade	Signature/Date
Head of Department	
_____	_____
External Examiner	Signature/Date

DEDICATION

This Project is dedicated to Almighty Allah, the beginning and the end who has been with me since my birth till the moment and to Prophet Muhammad (SAW). Also, to my parents (Alhaji Olanrewaju Umar and Alhaja Sijuola Umar), my guardians, supervisor and my boss at Plat Technologies and Iqraa Books (Mr. Abdulkareem Taofik and Mr. Abdul-lateef Bolaji) for their supports, guidance and prayers.

ACKNOWLEDGEMENT

All praise and adoration belong to Almighty Allah to his mercy and protection over me throughout my program in the university.

I acknowledge the efforts of my parents Alhaji Olanrewaju Umar, Alhaja Sijuola Umar, may almighty Allah spare their le to reap the soul of their labour (Amin). My sincere appreciation also goes to my love and caring brothers and sisters starting from Balikis Umar for her leadership role and Muhammed Jamiu Olanrewaju for his courageous words towards the success of this program, and thanks to entire family and it community in general. May Allah reward them all abundantly, Furthermore, I acknowledge the support of my friends from Kubiat Kelvin, Matthew Ayodeji and Ishola Taofiq, Lastly Nasri Shittu, and Olarewaju Ridwan. May Almighty God be with them and crown their efforts with success.

I appreciate my colleagues in the university, Taofiq Looro, Habeeb Mashod and my entire classmates. May He answer our prayers and crown all our efforts with success. The school authority is also inclusive, for creating an opportunity and avenue for us to be exposed to the outside world.

My profound gratitude goes to my supervisor, Dr. R.M. Isiaka, who did all he could to make this report a successful one. My appreciation also goes to all lecturers in the department.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF APPENDICES

ABSTRACT

This project application is well featured and provides the two operations of cryptography the encryption/decryption that can protect message from unauthorized access and disclosure over networks. To send message, a recipient or registered user types and encrypts a text message using keyword mono-alphabetic substitution algorithm with a key, selected from key list. The encrypted message is stored in the on the user root storage. The receiver, after log into his/her own account, selects the key value and then decrypts the encrypted message with the key to see the original message. With compared to other messaging systems. It use the advance encryption standard (AES) algorithm ad implemented using web programming tool, it store users activities and provide a clue of what ca do next, The end results is to perform the encryption/decryption operations on the best algorithm for cryptography operations and effectively manages and stores the end operations with time constraints and reduces the level of manual way of storing a classified / sensitive documents.

CHAPTER ONE

INTRODUCTION

1.1 Background of the Study

Cryptography has been used for long recent years. The earliest known cryptography was found engraved on Egyptian. Hebrew scholars used simple alphabetic substitutions, wherein one letter of the alphabet stands for another letter, with no letters standing for more than one and no letters being omitted. The ancient Greeks used cryptographic operation to transmit military messages. Encryption was used heavily to obscure radio transmissions and telegrams. Data cryptographic refers to mathematical calculations and algorithmic schemes that transform plaintext into cipher text, a form that is non readable to unauthorized parties. The recipient of an encrypted message uses a key which triggers the algorithm mechanism to decrypt the data, transforming it to the original plaintext version. The key pair might encrypt, decrypt or perform both functions. In public key, the key can be freely shared or given to anyone because its only job is to encrypt, while the private key is not shared and required to decrypt anything that has been encrypted by the public key. (Kayne, 2003-2012).

In the past, security was simply a matter of locking the door or storing files in a locked filing cabinet or safe. Today, paper is no longer the only medium of choice for housing information. Files are stored in computer databases as well as file cabinets. Hard drives

and floppy disks hold many of our secret information. In the physical world, security is a fairly simple concept. If the locks on your house's doors and windows are so strong that a thief cannot break in to steal your belongings, the house is secure. For further protection against intruders breaking through the locks, you might have security alarms. Similarly, if someone tries to fraudulently withdraw money from owner bank account but the teller asks for identification and does not trust the thief's story, your money is secure. When you sign a contract with another person, the signatures are the legal driving force that impels both parties to honour their word.

According to Pandya, Dwiti & Khushboo, Ram & Narayan, & Thakkar, Sneha & Madhekar, Tanvi & Thakare, B. (2015). Brief History of Encryption. International Journal of Computer Applications. Secure communication has been required several of years. This led to the invention of cryptography. In ancient world, primitive methods were adopted for passing messages secretly. But with the invention of internet and World Wide Web, which is used for communicating via mail, messages, online shopping, online banking, etc., increased the need of information security.

Thus, a proper understanding of various methods of cryptography and its implementation can fulfil the requirements of securing valuable and sensitive information. This research takes us through the various methods of cryptography adopted in the ancient period, medieval period and the modern era. In the digital world, security works in a similar way. One concept is privacy, meaning that no one can break into files to read your sensitive data

such as medical records or steal money. For example, obtaining credit card numbers or online brokerage accounts information. Privacy is the lock on the door. Another concept, data integrity, refers to operation that tells us when something has been altered. That's the alarm. By applying the practice of authentication, we can verify identities. That's comparable to the identification required to withdraw money from a bank account or conduct a transaction with an online broker. And finally, non-repudiation is a legal driving force that impels people to honour their word.

Another issue is that the cryptography does not solve access control problems. Most organizations need to limit data access to those who have a need to know. This type of security policy limiting data access to those with a need to see it is typically addressed by access control. The Oracle database has provided strong, independently evaluated access control for many years. It enables access control enforcement to an extremely fine level of access, through its Virtual Private Database capability. Cryptography would therefore not provide any additional security in the sense of better access control, and the encryption might enhance the proper or efficient functioning of the application (Oracle, 2014).

For instance, an employee, his manager, and a human resources clerk may all need to access the employee's record. If all employee data is encrypted, then all three must be able to access the data in decrypted form. Therefore, this research focus and maintain the level which unauthorized user get into confidential, classified and valuables information and to rewrite the information into cryptographic techniques only for designated person, and will

be given with an encryption and decryption key intended for a particular encrypted file to be decrypted.

1.2 Statement of the problem

Lately to the most recent hackers and spy attackers have been successfully gotten access to almost average individual privacies in no limitations. Often used the aforementioned ways in introduction chapter to be threating the lives of averagely and richly people among the society for collection of money in regards of not exposing their secrecy to general public view. The problem is security. The password method used in almost all commercial operating systems is probably not very strong against a sophisticated or unsophisticated attackers.

This project addresses the security problem of warranting data and information using cryptographic operations for the implementation of a secure data and information file household. A data and information that can provide strong security guarantees for users and its partners, even in post-modern era. To this aim, new cryptographic need to be researched, developed and implemented. This includes the choice of adequate parameters and protection of information leakage through side channels.

To improve the system the choice of data cryptography comes next in minds who wants to reduce of level of unauthorized access on confidential files or data and fixing a weakness at the theoretical or the implementation level can be extremely help to improve data

security and on the other hand it's enhance and modify the effectiveness of information security standard.

1.3 Aim and objectives

The aim is to implement a system application controlling the way messages are securely pass to the receiver and provide information security to classified document and files.

The objectives are to:

i. To design and implement a web-based information security using the two cryptographic algorithm operations.
ii. Utilizing the knowledge of programming tool to acquire the execution of the software application system
iii. To evaluate the system with the time effectiveness and responsiveness.

1.4 Scope and limitations of the study

The scope of the research is the modification of existing cryptography algorithms applications that will come up with a unique combination of two encryption algorithms in encrypting and decrypting an original data. In this study, I would like to emphasize the use of web-based in achieving the implementation to make it accessible worldwide if this research permit hosting of the software and the use of the same file system in transferring and copying encrypted data. Moreover, this research study will cover all data and information extension into cryptographic algorithm and its operation. The encryption/decryption key characters should be an alphabets, special characters or numbers and can be of repeating letters any other sequence of computation might fail.

This research will not cover the following concepts:

i. Transferring and copying encrypted data onto a different file system.
ii. This algorithm is tested only on files and no other computational extensions derived were tested upon. Folders will serve as containers for the multiple encrypted files.

1.5 Significant of the study

This research is conducted to develop data and information security that could output a new combination of encrypting and decrypting method that can assure the reliability of the security of multiple data. With this research, the following will be possible:

i. Encrypting and decrypting multiple data with a reliable algorithm.
ii. Creating a backup of the original files to be encrypted and transferred in case files will be damaged during encryption or decryption.
iii. Moving and copying encrypted multiple data into the same file system.
iv. To facilitate the use of more sophisticated tool against hacking, cracking, bugging of a system.
v. To provide a means of safeguarding data in a system and to enhance the integrity of data and extract existing data from historical set.

1.6 Project layout

This shows project layout (or sheets) show the alignment and plan or plan- profile sheet sequence and numbering for the project. This is an option to be included in the plans set at the discretion of the district. The project layout sheet can prove to be of great advantages for large or complicated projects involving large interchanges with a number of diverging routes.

Chapter one: it's emphases clearly the mean purpose of this project from the definition of project title, to the introduction, the background of the study, the statement of the problem, the aim and objectives, the scope of the project and the significant of the project.

Chapter two: this chapter discuss and show vividly the literature review, scholars that has research on way or the other by reviewing they work to advance the work and criticizes the existing once for the welcome development of new work.

Chapter three, system analysis and design the ideal to analysis and design the new system, in methodology approach, is the section in the project that's describe the ideal adopt to new ideal to use in proper documentation they are always in form of visual aid or using system design and analysis approach such Data Flow Diagram (DFD) Use-Case and lot more.

Chapter four: testing the implementation of the project, the gathering of fact to the audience and publishing our work.

Chapter five: This will be remarking all the work done, from chapter one to five the summary, the recommendation, the conclusion and the referencing or appendix

1.7 Definition of Terms

AUTHENTICATION: Prove or show to be authentic (genuine)

AUTHORIZATION: To give official permission for or approval to access information.

CLEAR-TEXT/PLAINTEXT: Original data/information before it is enciphered/scrambled.

CIPHERED TEXT/CRYPTOGRAM: Information that has been converted into other symbols or form.

CIPHERS: The secret code used to convert plain text message.

HACKERS: People who steal information without permission.

CRYPTOGRAPHY: This is process of hiding writing information.

CRYPTOGRAPHERS: Those that hide writing information.

CRYPTOLOGIST: Scientist who study different ways to protect information or data.

CONFIDENTIALITY: Information intended to be kept secret i.e. entrusted with private information.

CRYPTANALYSIS: Is the art of breaking of ciphers, some from program codes.

CUSTODIANS: Persons who are responsible for protecting or guiding something (record keeper).

DATA: A raw fact making the basis of reasoning; it is unprocessed information.

DECRYPTION: A process of converting data/information (multimedia) back to its original form.

ENCRYPTION: A process of converting data/information into another form.

INFORMATION: A processed data that is conveyed or presented by a particular sequence of symbols, impulses etc.

INTEGRITY: Consistency or lack of corruption in electronic data, the quality of information principles.

PASSWORD: A secret word or phrase used to gain admission or access to something (information).

CHAPTER TWO

LITERATURE REVIEW

2.1 Related concepts

The primary use for cryptosystems is to enable two people, Yusuf and Malik, to communicate over an insecure channel in a manner that prevents an opponent, Rahman, from being able to understand the conversation. The scenario is as shown in (Figure 2.0).

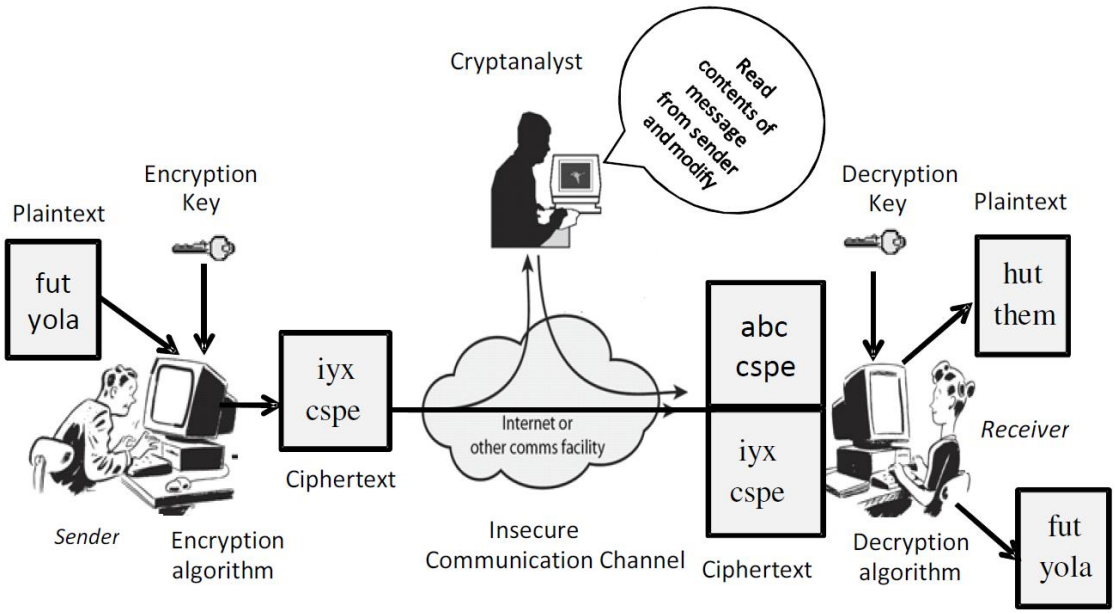


Figure 2.0*: Cryptographic Operations System*

To achieve this, Yusouf would convert his original message, the plaintext, into a message which is only intelligible by Malik, the cipher text. This process is known as encryption. When Malik receives the message, she will decrypt the cipher text to reveal the original message that was sent. Because only Yusouf and Malik have the key to the encryption algorithm, Rahman will be unable to reconstruct the plaintext, even if he intercepts the cipher text. In addition, the encryption and decryption algorithms are used to convert between the plaintext and cipher text and vice versa. A clear set of steps are needed to define how the data is transferred between Yusouf and Malik. This is called the protocol. The protocols used in cryptosystems are implemented to ensure that the participants achieve the desired goal of communication within the constraints of the environment, whilst adhering to the assumptions made during the construction of the components of the system.

Once the logic of the cryptosystem has been designed, the system needs to be implemented. In most cases, this would mean that computers need to be programmed to carry out the encryption and decryption, and that the computers need to be instructed to communicate in strict accordance with the protocols adopted. Thus, a cryptosystem consists of cryptographic algorithms, protocols and an implementation.

If a hacker (Rahman) wishes to break a cryptosystem, that is to affect the security of the communication in an adverse way, then he could attack any combination of the cryptographic algorithms, the protocol or the implementation. Cryptanalysis is the study of the techniques used to break information security systems.

The attacks include attempting to extract the plaintext from a cipher text message without having access to the encryption key or attempting to recover the encryption key when only the cipher text is known. Cryptanalysis can have different modes of attacks like cipher text only attack, known plain text attack, chosen plain text attack and chosen cipher text attack [Douglas R S 2013].

2.1.1 History of cryptography

Cryptography has a long and fascinating history. The most comprehensive non-technical account of the subject is Kahn's The Code Breakers [Kahn D 2000], which traces cryptography from its initial and limited use by the Egyptians some years ago, to the twentieth century, where it played a crucial role in the outcome of both world wars. Published in 2001, Kahn's book covers those aspects of the history which were most significant (up to that time) to the development of the subject. Cryptography is the science of secret writing an ancient art; the first documented use of cryptography in writing dates back years ago. When an Egyptian scribe used non-standard hieroglyphs in an inscription. Some experts argue that cryptography appeared spontaneously sometime after writing was invented, with applications ranging from diplomatic missives to war-time battle plans.

It is no surprise, then, that new forms of cryptography came soon after the widespread development of computer communications. In data and telecommunications, cryptography is necessary when communicating over any untrusted medium, which includes just about *any* network, particularly the Internet.

It is difficult to pinpoint the exact beginning of cryptography. However, the inscriptions carved into the walls of the main chamber of the tomb of the nobleman Khnumhotep II, provide the first example of deliberate transformation of writing. The tomb was found in the town of Menet Khufu bordering on the Nile in Egypt, and the inscriptions date to approximately lately 1900's BC. The intention of the transformations performed by the scribe was not that of concealment, but most likely to impart dignity and authority.

Since cryptography is used to protect a secret, it is to be expected that unintended recipients would attempt to decipher the meaning of the encrypted message. The first record of active cryptanalysis comes from the Arabs during the 700s. Formal techniques, such as letter frequency analysis, came into being only during the past few hundred years.

The information for the cryptography section was mostly attributed to Ibn-ad-Duraihim, who lived and held various teaching and official posts in Syria and Egypt Stewart (Gebbie 2002).

Early cryptosystems usually relied on transformations of the plaintext message, being performed by the person composing the message. However, as the complexity of the methods increased, it became desirable to create tools/machines that would perform the cryptographic tasks.

2.1.2 Applications of cryptography

Cryptography in Everyday Life Authentication/Digital Signatures is authentication, and digital signatures are a very important application of public-key cryptography. For example, if you receive a message from me that I have encrypted with my private key and you are able to decrypt it using my public key, you should feel reasonably certain that the message did in fact come from me. If I think it necessary to keep the message secret, I may encrypt the message with my private key and then with your public key, that way only you can read the message, and you will know that the message came from me. The only requirement is that public keys are associated with their users by a trusted manner, for example a trusted directory. To address this weakness, the standards community has invented an object called a certificate. A certificate contains, the certificate issuer's name, the name of the subject for whom the certificate is being issued, the public key of the subject, and some time stamps. You know the public key is good, because the certificate issuer has a certificate too.

Time stamping is a technique that can certify that a certain electronic document or communication existed or was delivered at a certain time. Time stamping uses an encryption model called a blind signature scheme. Blind signature schemes allow the sender to get a message receipted by another party without revealing any information about the message to the other party.

Time stamping is very similar to sending a registered letter through the U.S. mail but provides an additional level of proof. It can prove that a recipient received a specific document. Possible applications include patent applications, copyright archives, and contracts. Time stamping is a critical application that will help make the transition to electronic legal documents possible.

Electronic Money the definition of electronic money also called electronic cash, or digital cash is a term that is still evolving. It includes transactions carried out electronically with a net transfer of funds from one party to another, which may be either debit or credit and can be either anonymous or identified. There are both hardware and software implementations.

Anonymous Communications applications do not reveal the identity of the customer and are based on blind signature schemes. Digicash's Ecash Identified spending schemes reveal the identity of the customer and are based on more general forms of signature schemes. Anonymous schemes are the electronic analog of cash, while identified schemes are the electronic analog of a debit or credit card. There are also some hybrid approaches where payments can be anonymous with respect to the merchant but not the bank CyberCash credit card transactions or anonymous to everyone, but traceable a sequence of purchases can be related, but not linked directly to the spender's identity.

Encryption is used in electronic money schemes to protect conventional transaction data like account numbers and transaction amounts, digital signatures can replace handwritten signatures or a credit-card authorizations, and public-key encryption can provide confidentiality. There are several systems that cover this range of applications, from transactions mimicking conventional paper transactions with values of several dollars and up, to various micropayment schemes that batch extremely low-cost transactions into amounts that will bear the overhead of encryption and clearing the bank.

Secure Network Communications, Secure Socket Layer (SSL), Netscape has developed a public-key protocol called Secure Socket Layer (SSL) for providing data security layered between TCP/IP (the foundation of Internet-based communications) and application protocols (such as HTTP, Telnet, NNTP, or FTP). SSL supports data encryption, server authentication, message integrity, and client authentication for TCP/IP connections. The SSL Handshake Protocol authenticates each end of the connection (server and client), with the second or client authentication being optional. In phase 1, the client requests the server's certificate and its cipher preferences. When the client receives this information, it generates a master key and encrypts it with the server's public key, then sends the encrypted master key to the server. The server decrypts the master key with its private key, then authenticates itself to the client by returning a message encrypted with the master key. Following data is encrypted with keys derived from the master key. Phase 2, client authentication, is optional.

The server challenges the client, and the client responds by returning the client's digital signature on the challenge with its public-key certificate. SSL uses the RSA public-key cryptosystem for the authentication steps. After the exchange of keys, a number of different cryptosystems are used, including RC2, RC4, IDEA, DES and triple-DES.

2.1.3 Goals of cryptograph

Cryptography is the study of mathematical techniques related to aspects of information security such as confidentiality, data integrity, entity authentication and data origin authentication. So, the main goals of cryptography are privacy or confidentiality, data integrity, authentication and non-repudiation. Privacy or confidentiality is the service used to keep the content of information secret from all but those authorized one to have it. Secrecy, confidentiality and privacy are synonymous terms. There are number of approaches to providing confidentiality through mathematical algorithms which render data unintelligible [Santhosh Kumar 2010].

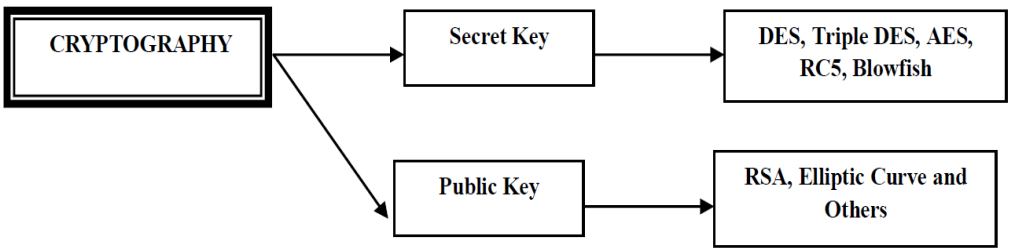


Figure 2.1: *Classification of cryptographic operations*

There are five primary functions of cryptography:

1. Privacy/confidentiality: Ensuring that no one can read the message except the intended receiver.
2. Authentication: The process of proving one's identity.
3. Integrity: Assuring the receiver that the received message has not been altered in any way from the original.
4. Non-repudiation: A mechanism to prove that the sender really sent this message.
5. Key exchange: The method by which crypto keys are shared between sender and receiver.

In cryptography, we start with the unencrypted data, referred to as *plaintext*. Plaintext is *encrypted* into *ciphertext*, which will in turn (usually) be *decrypted* back into usable plaintext. The encryption and decryption is based upon the type of cryptography scheme being employed and some form of key. For those who like formulas, this process is sometimes written as:

$$C = E_k(P)$$

$$P = D_k(C)$$

Where **P** = plaintext, **C** = ciphertext, **E** = the encryption method, **D** = the decryption method, and **k** = the key. Given this, there are other functions that might be supported by crypto and other terms that one might hear:

i. Forward Secrecy (aka Perfect Forward Secrecy): This feature protects past encrypted sessions from compromise even if the server holding the messages is compromised. This is accomplished by creating a different key for every session so that compromise of a single key does not threaten the entirely of the communications.

ii. Perfect Security: A system that is unbreakable and where the cipher text conveys no information about the plaintext or the key. To achieve perfect security, the key has to be at least as long as the plaintext, making analysis and even brute-force attacks impossible. One-time pads are an example of such a system.

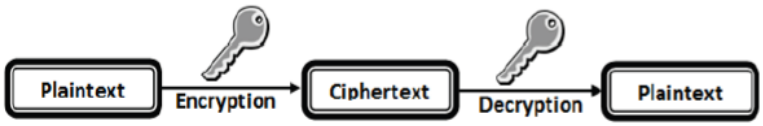
iii. Deniable Authentication (aka Message Repudiation): A method whereby participants in an exchange of messages can be assured in the authenticity of the messages but in such a way that senders can later plausibly deny their participation to a third-party.

In many of the descriptions below, two communicating parties will be referred to as Alice and Bob; this is the common nomenclature in the crypto field and literature to make it easier to identify the communicating parties. If there is a third and fourth party to the communication, they will be referred to as Carol and Dave, respectively. A malicious party is referred to as Mallory, an eavesdropper as Eve, and a trusted third party as Trent.

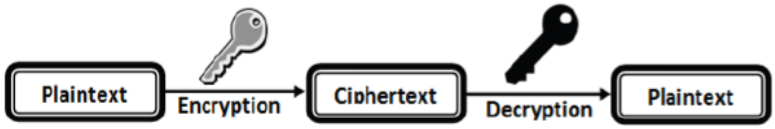
Finally, cryptography is most closely associated with the development and creation of the mathematical algorithms used to encrypt and decrypt messages, whereas cryptanalysis is the science of analysing and breaking encryption schemes. Cryptology is the term referring to the broad study of secret writing and encompasses both cryptography and cryptanalysis.

2.1.4 Types of cryptographic algorithms

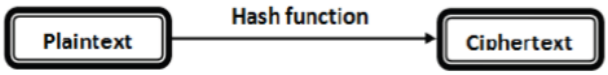
There are several ways of classifying cryptographic algorithms. For purposes of this paper, they will be categorized based on the number of keys that are employed for encryption and decryption and further defined by their application and use. The three types of algorithms that will be discussed are (Figure2.2):



(a) *Symmetric-key Cryptography: A single key for both encryption and decryption.*



(b) *Public-key Cryptography: A pair of keys, one for encryption and the other for decryption.*



(c) *Hash function (one-way cryptography): Hash functions have no key since the plaintext is not recoverable from the ciphertext.*

Figure 2.2: *Three types of cryptography: Symmetric-key, public-key, and hash function*

A comparison of popular encryption algorithms based on block size, key size, number of rounds and attacks if occurred is shown on (Table 2.0). It clearly shows the supremacy of Blowfish algorithm over DES, AES and Triple DES on the basis of key size and security. DES and other algorithms are vulnerable to possible attacks, but Blowfish algorithm has not been cracked till date.

	Symmetric Encryption Algorithms			
	DES	*TDES*	*AES*	*BLOWFISH*
Block Size	64 bit	64 bit	128 bit	64 bit
Key size	56 bit	168 bit	128,192, 256 bit	32-448 bit
Created By	IBM in 1975	IBM in 1978	Joan Daeman in 1998	Bruce Schneier in 1998
Algorithm Structure	Fiestel Network	Fiestel Network	Substitution Permutation Network	Fiestel Network
Rounds	16	48	9,11,13	16
Attacks	Brute Force Attack	Theoretically possible	Side Channel Attacks	Not Yet

Table 2.0: *Comparison of DES, Triple DES, and AES and Blow Fish algorithm*

i. Secret Key Cryptography (SKC): Uses a single key for both encryption and decryption; also called symmetric encryption. Primarily used for privacy and confidentiality.

ii. Public Key Cryptography (PKC): Uses one key for encryption and another for decryption; also called asymmetric encryption. Primarily used for authentication, non-repudiation, and key exchange.

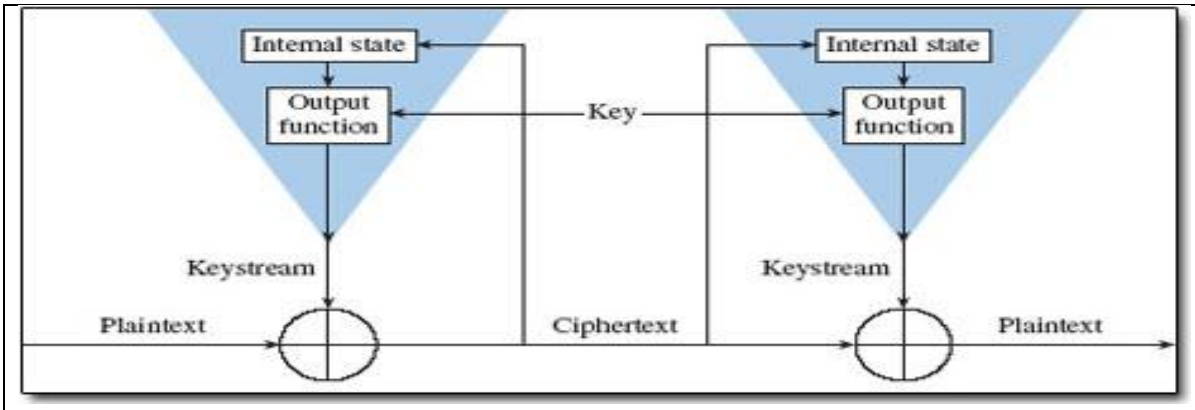
iii. Hash Functions: Uses a mathematical transformation to irreversibly "encrypt" information, providing a digital fingerprint. Primarily used for message integrity.

2.1.5 Secret key cryptography

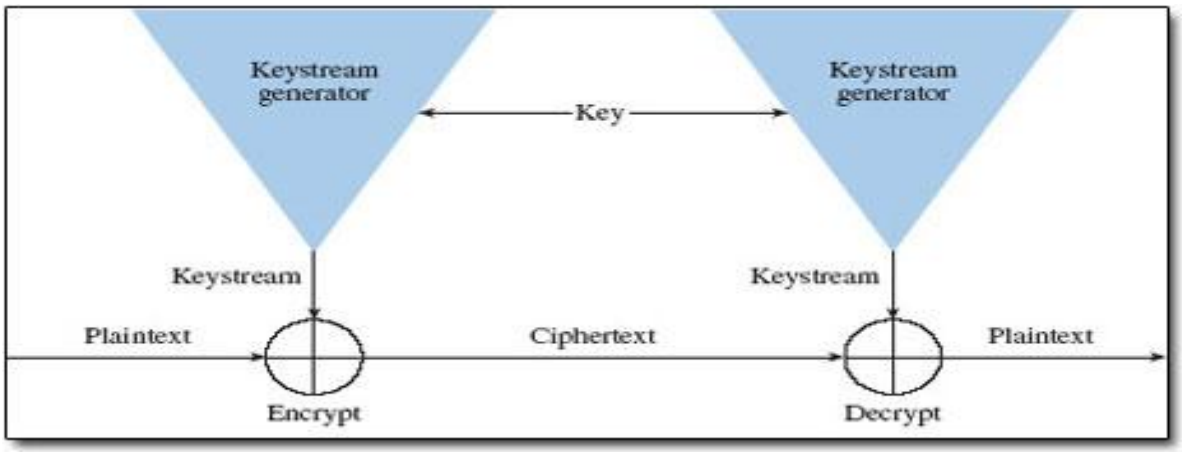
Secret key cryptography methods employ a single key for both encryption and decryption. As shown in Figure 1A, the sender uses the key to encrypt the plaintext and sends the ciphertext to the receiver. The receiver applies the same key to decrypt the message and recover the plaintext. Because a single key is used for both functions, secret key cryptography is also called symmetric encryption.

With this form of cryptography, it is obvious that the key must be known to both the sender and the receiver; that, in fact, is the secret. The biggest difficulty with this approach, of course, is the distribution of the key (more on that later in the discussion of public key cryptography).

Secret key cryptography schemes are generally categorized as being either stream ciphers or block ciphers.



A) Self-synchronizing stream cipher. *(From* Schneider*, 2010)*



B) Synchronous stream cipher. *(From* Schneider*, 2010)*

Figure 2.3: *Types of stream ciphers*

Stream ciphers operate on a single bit (byte or computer word) at a time and implement some form of feedback mechanism so that the key is constantly changing. Stream ciphers come in several flavours but two are worth mentioning here (Figure 2.4).

Self-synchronizing stream ciphers calculate each bit in the keystream as a function of the previous n bits in the keystream. It is termed "self-synchronizing" because the decryption process can stay synchronized with the encryption process merely by knowing how far into the n-bit keystream it is. One problem is error propagation; a garbled bit in transmission will result in n garbled bits at the receiving side. Synchronous stream ciphers generate the keystream in a fashion independent of the message stream but by using the same keystream generation function at sender and receiver. While stream ciphers do not propagate transmission errors, they are, by their nature, periodic so that the keystream will eventually repeat.

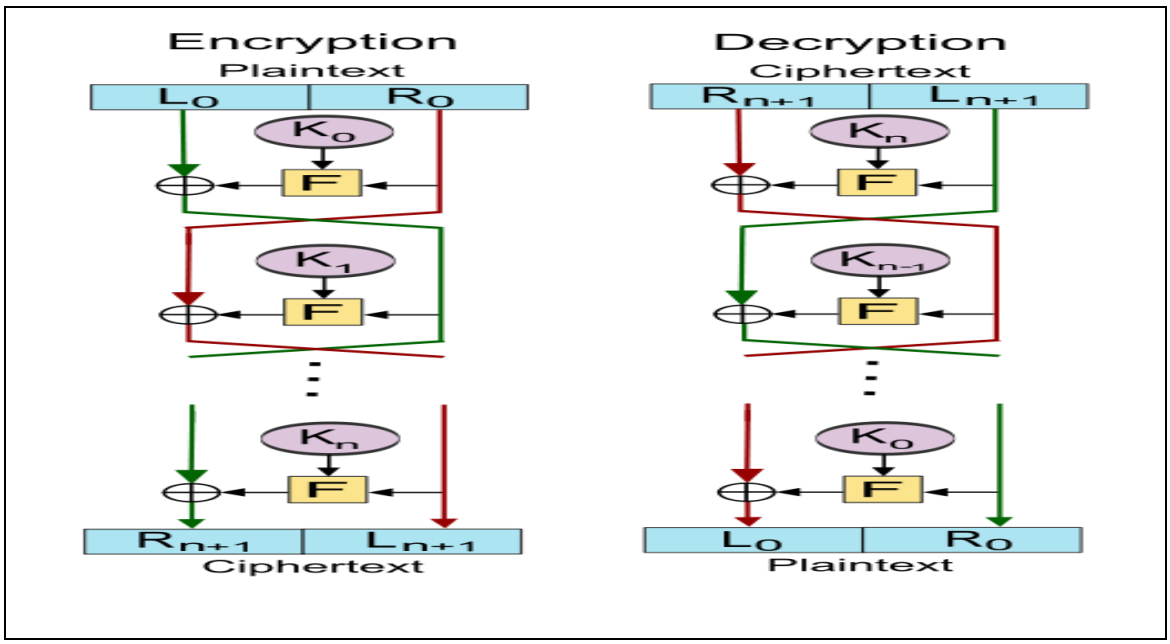


Figure 2.4: *Feistel cipher.* "Source: Wikimedia Commons"

2.1.6 Symmetric key or secret key algorithm

Algorithms	**Key Length**	**Block Size**
DES	56 BITS	64 bits
3DES	56, 112, or 168 bits	64 bits
AES	128, 12, or 256 bits	128 bits
IDEA	128 bits	64 bits
RC4	40 to 256 bits	Stream cipher
RC5	0 to 2040 bits (128 recommended)	32, 64 or 128 bits (64 recommended)

Table: 2.1 *Secret Key Cryptography table*

2.1.7 Public key cryptography

Public key cryptography has been said to be the most significant new development in cryptography in the last 300-400 years. Modern PKC was first described publicly by Stanford University professor Martin Hellman and graduate student Whitfield Diffie in 1976. Their paper described a two-key crypto system in which two parties could engage in a secure communication over a non-secure communications channel without having to share a secret key.

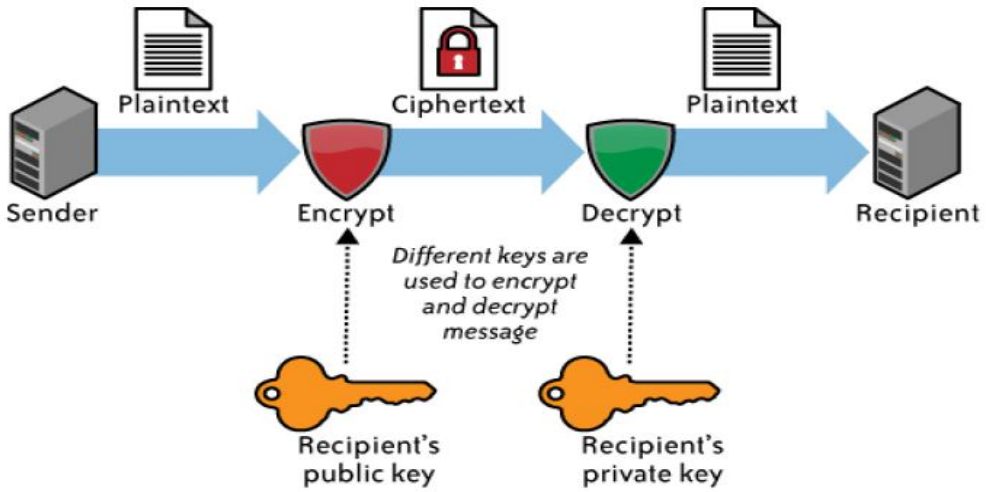


Figure: 2.5 *Secret Key Cryptography "image was downloaded from* www.ijcsmc.com*"*

Block Size ***2n***	**Key Size** ***mn***	**Word Size** ***n***	**Key Words** ***m***	**Rounds** ***T***
32	64	16	4	32
48	72 96	24	3 4	36 36
64	96 128	32	3 4	42 44
96	96 144	48	2 3	52 54
128	128 192 256	64	2 3 4	68 69 72

Table 2.2*: public key cryptography*

Public Key Cryptography depends upon the existence of so-called *one-way functions*, or mathematical functions that are easy to compute whereas their inverse function is relatively difficult to compute. Let me give you two simple examples:

i. Multiplication vs. factorization: Suppose you have two prime numbers, 3 and 7, and you need to calculate the product; it should take almost no time to calculate that value, which is 21. Now suppose, instead, that you have a number that is a product of two primes, 21, and you need to determine those prime factors. You will eventually come up with the solution but whereas calculating the product took milliseconds, factoring will take longer. The problem becomes much harder if we start with primes that have, say, 400 digits or so, because the product will have 800 digits.

ii. Exponentiation vs. logarithms: Suppose you take the number 3 to the 6th power; again, it is relatively easy to calculate $3^6 = 729$. But if you start with the number 729 and need to determine the two integers, x and y so that $\log_x 729 = y$, it will take longer to find the two values.

While the examples above are trivial, they do represent two of the functional pairs that are used with Public Key Cryptography; namely, the ease of multiplication and exponentiation versus the relative difficulty of factoring and calculating logarithms, respectively. The mathematical "trick" in Public Key Cryptography is to find a trap door in the one-way function so that the inverse calculation becomes easy given knowledge of some item of information.

Generic Public Key Cryptography employs two keys that are mathematically related although knowledge of one key does not allow someone to easily determine the other key. One key is used to encrypt the plaintext, and the other key is used to decrypt the cipher text. The important point here is that it does not matter which key is applied first, but that both keys are required for the process to work (Figure 1B). Because a pair of keys are required, this approach is also called asymmetric cryptography.

In Public Key Cryptography, one of the keys is designated the public key and may be advertised as widely as the owner wants. The other key is designated the private key and is never revealed to another party. It is straight-forward to send messages under this scheme. Suppose Alice wants to send Bob a message. Alice encrypts some information using Bob's public key; Bob decrypts the cipher text using his private key. This method could be also used to prove who sent a message; Alice, for example, could encrypt some plaintext with her private key; when Bob decrypts using Alice's public key, he knows that Alice sent the message (authentication) and Alice cannot deny having sent the message (non-repudiation).

2.1.8 Hash functions

Hash functions, also called message digests and one-way encryption, and are algorithms that, in essence, use no key (Figure 1C). Instead, a fixed-length hash value is computed based upon the plaintext that makes it impossible for either the contents or length of the

plaintext to be recovered. Hash algorithms are typically used to provide a digital fingerprint of a file's contents, often used to ensure that the file has not been altered by an intruder or virus. Hash functions are also commonly employed by many operating systems to encrypt passwords. Hash functions, then, provide a mechanism to ensure the integrity of a file.

Let me reiterate that hashes are one-way encryption. You cannot take a hash and "decrypt" it to find the original string that created it, despite the many web sites that claim or suggest otherwise, such as Crack Station, HashKiller.co.uk, MD5 Online, md5thiscracker, Online Hash Crack, and Rainbow Crack.

2.1.9 Why three encryption techniques

Secret key cryptography, on the other hand, is ideally suited to encrypting messages, thus providing privacy and confidentiality. The sender can generate a *session key* on a per-message basis to encrypt the message; the receiver, of course, needs the same session key in order to decrypt the message.

Key exchange, of course, is a key application of public key cryptography (no pun intended). Asymmetric schemes can also be used for non-repudiation and user authentication; if the receiver can obtain the session key encrypted with the sender's private key, then only this sender could have sent the message.

Public key cryptography could, theoretically, also be used to encrypt messages although this is rarely done because secret key cryptography values can generally be computed about 1000 times faster than public key cryptography values.

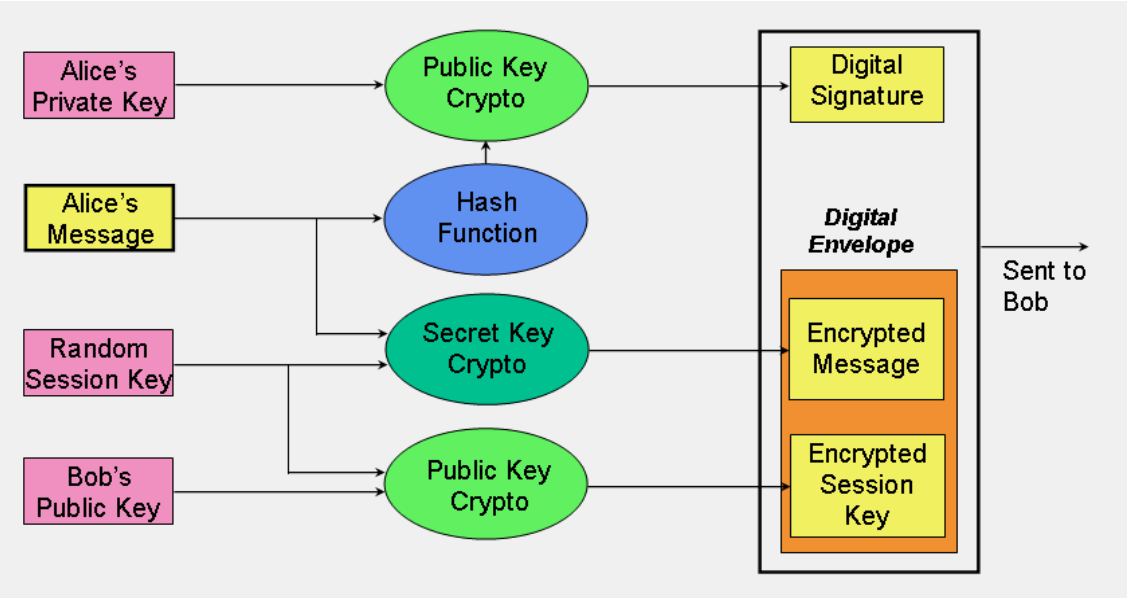


Figure 2.6: *Use of the three cryptographic techniques for secure communication*

2.1.10 The significance of key length

In a 1998 article in the industry literature, a writer made the claim that 56-bit keys did not provide as adequate protection for DES at that time as they did in 1975 because computers were 1000 times faster in 1998 than in 1975. Therefore, the writer went on, we needed 56,000-bit keys in 1998 instead of 56-bit keys to provide adequate protection.

The conclusion was then drawn that because 56,000-bit keys are infeasible (*true*), we should accept the fact that we have to live with weak cryptography (*false!*). The major error here is that the writer did not take into account that the number of possible key values double whenever a single bit is added to the key length; thus, a 57-bit key has twice as many values as a 56-bit key (because 2^{57} is two times 2^{56}). In fact, a 66-bit key would have 1024 times more values than a 56-bit key.

But this does bring up the question "What is the significance of key length as it affects the level of protection."

In cryptography, size matters. The larger the key, the harder it is to crack a block of encrypted data. The reason that large keys offer more protection is almost obvious; computers have made it easier to attack cipher text by using brute force methods rather than by attacking the mathematics (which are generally well-known anyway). With a brute force attack, the attacker merely generates every possible key and applies it to the cipher text. Any resulting plaintext that makes sense offers a candidate for a legitimate key. This was the basis, of course, of the EFF's attack on DES.

2.1.11 Forms of security

Security is "the quality or state of being secure and to be free from danger. "In other words, protection against adversaries from those who would do harm, intentionally or otherwise is the objective.

A successful organization should have the following multiple layers of security in place to protect its operations:

i. **Physical security**, to protect physical items, objects, or areas from unauthorized access and misuse
ii. **Personnel security**, to protect the individual or group of individuals who are authorized to access the organization and its operations
iii. **Operations security**, to protect the details of a particular operation or series of activities
iv. **Communications security**, to protect communications media, technology, and content
v. **Network security**, to protect networking components, connections, and contents
vi. **Information security**, to protect confidentiality, integrity and availability of information assets, whether in storage, processing, or transmission. It is achieved via the application of policy, education, training and awareness, and technology.

2.1.12 Key information security concepts

This book uses a few terms and concepts that are essential to any discussion of information security.

i. **Access**: Authorized users have legal access to a system, whereas hackers have illegal access to a system. Access controls regulate this ability.

ii. **Asset**: An asset can be logical, such as a Web site, information, or data; or an asset can be physical, such as a person, computer system, or other tangible object.

iii. **Attack**: An intentional or unintentional act that can cause damage to or otherwise compromise information and/or the systems that support it. Attacks can be active or passive, intentional or unintentional, and direct or indirect.

iv. **Exploit**: A technique used to compromise a system. Threat agents may attempt to exploit a system or other information assets by using it illegally for their personal gain.

v. **Exposure**: A condition or state of being exposed. In information security, exposure exists when a vulnerability known to an attacker is present.

vi. **Loss**: A single instance of an information asset suffering damage or unintended or unauthorized modification or disclosure. When an organization's information is stolen, it has suffered a loss.

vii. **Risk**: The probability that something unwanted will happen.

viii. **Threat**: A category of objects, persons, or other entities that presents a danger to an asset. Threats are always present and can be purposeful or undirected.

2.1.13 Password protection

Nearly all modern multiuser computer and network operating systems employ passwords at the very least to protect and authenticate users accessing computer and/or network resources. But passwords are *not* typically kept on a host or server in plaintext but are generally encrypted using some sort of hash scheme.

A) /etc/passwd file

root: Jbw6BwE4XoUHo: 0:0: root:/root:/bin/bash

carol:FM5ikbQt1K052:502:100: Carol Monaghan:/home/carol:/bin/bash

alex: LqAi7Mdyg/HcQ: 503:100: Alex Insley:/home/alex:/bin/bash

B.1) /etc/passwd file (with shadow passwords)

root:x: 0:0: root:/root:/bin/bash

carol:x: 502:100: Carol Monaghan:/home/carol:/bin/bash

alex:x: 503:100: Alex Insley:/home/alex:/bin/bash

B.2) /etc/shadow file

root: AGFw1P4u/uhLK$l2.HP35rlu65WlfCzq:11449:0:99999: 7:::

carol: kjHaN%35a8xMM8a/0kMl1? fwtLAM.K&kw.:11449:0:99999: 7:::

alex:1$1KKmfTy0a7#3. LL9a8H71lkwn/. hH22a: 11449:0:99999: 7:::

Figure 2.7*: Sample entries in Unix/Linux password files.*

Passwords are not saved in plaintext on computer systems precisely so they cannot be easily compromised. For similar reasons, we don't want passwords sent in plaintext across a network. But for remote logon applications, how does a client system identify itself or a user to the server? One mechanism, of course, is to send the password as a hash value and that, indeed, may be done. A weakness of that approach, however, is that an intruder can grab the password off of the network and use an off-line attack (such as a *dictionary attack* where an attacker takes every known word and encrypts it with the network's encryption algorithm, hoping eventually to find a match with a purloined password hash). In some situations, an attacker only has to copy the hashed password value and use it later on to gain unauthorized entry without ever learning the actual password.

2.1.14 Encryption/ decryption algorithms

2.1.14.1 Encryption algorithms

```
KeywordEncryption (DATA TRANSFER, CIPHER, INFORMATION, MEDIA)
Enter the message, MESSAGE of length N characters.
Reseat steps (3) and (4) for i = 0 to N-1
If MESSAGE [i] is blanks space, then continue;
Repeat step for j = 0 to M-1 (M is the length of LETTER)
        If MESSAGE [i] is equal to LETTER [j], Then:
        CIPHER[i] = MEDIA[j] and break.
Return CIPHER;
```

Table: 2.3 *Cryptography Encryption Algorithm table*

2.1.14.2 Decryption algorithms

```
KeywordDecryption (DATA TRANSFER, CIPHER, INFORMATION, MEDIA)
Enter the ciphertext,ciphertext, CIPHER of length N characters.
Reseat steps (3) and (4) for i = 0 to N-1
If CIPHER [i] is blanks space, then continue.
Repeat step for j = 0 to M-1 (length of MEDIA)
        If CIPHER [i] is equal to MEDIA [j], Then:
        DATA TRANSFER[i] = INFORMATION[j] and break.
Return DATA TRANSFER;
```

Table: 2.4 *Cryptography Decryption Algorithm table*

2.2 Related works

Peter B. and Gregory W. (2020). Cryptographic Algorithms for Secure Data Communication. Personal privacy is of utmost importance in the global networked world. One of the best tools to help people safeguard their personal information is the use of cryptography. In this paper we present new cryptographic algorithms that employ the use of asymmetric keys. The proposed algorithms encipher message into nonlinear equations using public key and decipher by the intended party using private key. If a third party intercepted the message, it will be difficult to decipher it due to the multilevel ciphers of the proposed application.

Some vital information that are disseminated within an office, across offices, between branches, of an organization and other external bodies and establishments at times get into the hands of the unauthorized persons who may tamper with the contents of the information. And if no security measures are taken, there is no doubt that such data and other sensitive information will be exposed to threats such as impersonation, in secrecy, corruption, repudiation, break-in or denial of services that may cause serious danger on the individual or organization. A secure system should maintain the integrity, availability, and privacy of data. Data integrity usually means protection from unauthorized modification, resistance to penetration and protection from undetected modification. Therefore, algorithms which help prevent interception, modification, penetration, disclosure and enhance data/information security are now of primary importance. This paper suggests new methods for secured means of communication over unsecure channel. This is to ensure that the intruders do not have access to the plaintext without a secret key.

Sheenal M, and Sourabh D. (2018). Secure Data Sharing Scheme using Cryptographic Algorithm for Cloud Storage. The basic service provided by the Cloud is Data Storage. However, it is a difficult task for sharing data in multi owner manner where group admin and all group members can store and modify data while preserving data and identity privacy from an untrusted cloud server, due to the frequent change of the membership. Many of the public cloud computing services have appeared for data storage in group applications.

Two important problems that arise when sharing group data in public cloud are the privacy and security of group member's data. Cloud service providers are separate administrative entities and users don't have access to the cloud internal operational details. Because of the semi trust nature of cloud service provider, the traditional security technologies cannot be directly applied to the public cloud-based group data sharing applications. So that, Data sharing is increasingly important for many users and sometimes an essential requirement, especially for industries and societies used to gain proceeds. Sharing group resource among cloud users is a major problem, still the data privacy leak. Most of the traditional systems are used Group Key Management method for sharing Key Generation and distribution in the group member or users. Sometimes change to user one group to another group, the group key to enable authenticated users to access the files securely and efficiently is still a challenging problem. Here we are proposing a security framework for group data sharing that make accessible to data file in secure manner in public cloud environment. There are a number of different approaches that exist for securing the data on network & among them the cryptography is a widespread and classical approach to secured data. Furthermore, the key reason behind use of cryptography for security is their low cost implementation and freedom and springiness to change the security according to needs. Therefore, in this paper key area of work is investigated and design of a secure data sharing techniques for cloud storage.

Rahman M., Akter. U, and Rahman A. (2018) Development of Cryptography-Based Secure Messaging System. Today data communication is a modern technology that contains a powerful computer processor to exchange information. But brute force attacks are made to break the encryption techniques, and these attacks are the main drawbacks of older algorithms. This paper is concerned with the development of a secure messaging system based on cryptographic algorithms that is which is faster, better immune to attacks, more complex, easy to encrypt and many more advanced security feature included. This project work is designed and developed for a secure messaging both in web and android platforms. The application is well featured and provides encryption/decryption that can protect message from unauthorized access and disclosure over networks. To send message, a recipient or registered user types and encrypts a text message using keyword mono-alphabetic substitution algorithm with a key, selected from key list. The encrypted message is stored in the database and receiver's inbox with serial number of key (not the value). The receiver, after log into his/her own account, selects the key value and then decrypts the encrypted message with the key to see the original message. With compared to other messaging systems, the proposed secure messaging system can be used for chat, messaging, video conferencing and real time file sharing in both web and android platforms.

Hussain A. and Manish M. (2018) Development of encryption and decryption technique to secure the confidential data. Cryptography is the science of converting one form of data into another form, to provide security. Converting plain text to cipher text is encryption and converting cipher text to plain text is decryption. In the today's leading world huge amount of data are stored and transferred every second. But the issue is to provide the security to that data. For this encryption and decryption is considered as one of the better techniques. Two types of approach are used for encryption and decryption that is symmetric key cryptography and asymmetric key cryptography. Rsa is one of the algorithms used in asymmetric key cryptography. Rsa algorithm is better in terms of security, but this algorithm reduces the speed of mutual authentication. In this paper, we have developed a modified rsa approach where the speed of mutual authentication is enhanced with the help of modified rsa algorithm and a strong key is used to provide better security.

Anshu C., Goswami N., and Rakesh S. (2017) Privacy Algorithms to Improve the Secure Framework for Cloud Computing Environment. Cloud computing is a different type of computing platform which are sharing computing resources and handles all applications. The cloud computing environment with the service node to control all users request could provide maximum service to all users. Cloud computing is internet-based computing, a growing latest trend in the information technology (it) world.

The internet is being frequently represented as a cloud and virtualized hence the term “cloud computing”. Cloud computing is a collection of new and old concepts in many research areas, such as distributed computing, grid computing, utility computing and service-oriented architectures. In brief, cloud computing is the dynamic provisioning of it capabilities (such as hardware, software deployment and services) from over the network. Out of various techniques of cloud computing such as distributed, parallel, grid, utility and service oriented, the technique of cloud computing is the most vital one due to its many services on pay per use basis. Privacy issues in traditional web applications are still valid in the cloud computing environment. Cloud computing is a new concept of the modern world. Cloud computing combines all the services models and technologies together to deliver IT enterprise. The objective of this paper is to provide the security to end user to protect files or data from the unauthorized user. Privacy is an important issue for any technology through which unauthorized user can't access your file or data in a cloud. The main aim of this paper is to design and propose an architecture that can help to encrypt and decrypt algorithm. In this paper, we are presenting an encryption algorithm to deal with the privacy problems in cloud computing and protect the data stored in the cloud.

Naveen K. (2017). Implementation of Secure Payment Transaction using AES encryption with extended Visual Cryptography. Web application security has become an important concern for every user. This project implements a secure transaction platform that helps users to make secure bank transactions.

In the communication between bank and merchant, every time merchant must be varied to prevent fraudulent transactions. In this project, when a user requests a transaction from a bank, the bank generated one-time pass code (OTP) will be converted into a quick response (QR) code, a matrix barcode that allows quick access to URLs, among other uses. The QR code image is converted into two deferent shares using (two, two) scheme of visual cryptography. (Two, two) scheme of visual cryptography is a technique in which two image shares are created out of one image. Image shares are components of the original image so that any one share (on its own) does not reveal anything about the original image. Only if you have all of the shares can you determine the secret image. One share is given to the merchant server with an advanced encryption standard (AES) encrypted image. Only the legitimate merchant server can access the share because the AES key is required to decrypt the share. Another share will be sent to the users registered email address. Both shares are overlaid to obtain the original QR code. QR code is scanned by the user to obtain the one-time pass code. OTP is entered in the merchant server web page to complete the transaction. This project is an implementation of online fraud prevention using secure authentication process, visual cryptography, and encryption.

Arati P., and Sunita S. (2016). Data Security Using Cryptography and Steganography Techniques. International Journal of Advanced Computer Science and Applications. As use of computer networks and internet is growing very fast and admiring day by day, information security is become a major concern in computer networks.

There is always risk in violation of network security which leads a need of an efficient and simple way of securing the electronic documents from being read or used by people other than who are authorized to do it. Encryption is one of the security techniques widely used to ensure secrecy. Encryption is an entirely mathematical process that takes in data, performs some predefined mathematical operations on the data, and then outputs the result. Blowfish is one of the superlative encryption algorithms because it requires less execution time, memory and has high throughput. However, if any eavesdropper detects the presence of encrypted data he or she can try several attacks in order to get the original data. So there is a need to provide a two-layer approach for better security. That's why this work presents a security system using combination of cryptography and steganography to enhance the security.

Marwa E., Abdelmgeid A., and Fatma A. (2016). Data Security Using Cryptography and Steganography Techniques. Although cryptography and steganography could be used to provide data security, each of them has a problem. Cryptography problem is that the cipher text looks meaningless, so the attacker will interrupt the transmission or make more careful checks on the data from the sender to the receiver. Steganography problem is that once the presence of hidden information is revealed or even suspected, the message is become known. According to the work in this paper, a merged technique for data security has been proposed using Cryptography and Steganography techniques to improve the security of the information. Firstly, the Advanced Encryption Standard (AES) algorithm has been modified and used to encrypt the secret message. Secondly, the encrypted message has

been hidden. Therefore, two levels of security have been provided using the proposed hybrid technique. In addition, the proposed technique provides high embedding capacity and high-quality steganography images and plain text.

2.2.1 Challenges observed in literature review

Few challenges or issues that were identified during reading and analyzing the research papers have been outlined below;

i. Some of the research papers focused their implementation on platform as a service and Software.
ii. Other papers also concentrated on data Confidentiality without taking into account Integrity, non-repudiation and authenticity.
iii. Few of the papers were theoretical based, meaning actual practical implementation was not done.
iv. In other papers, though the technique proposed seams reliable, it looks weird, complicated and cumbersome to implement.
v. Some proposed techniques were also not experimentally validated like the model seem not in existence.

2.2.2 Conclusion

Security of data becomes more important when we transfer data over insecure communication medium. Data transfer refers to moving data from source location to destination location. To have a secure data transfer, few methods can be applied, and one of them is encryption of data, prepare it to be transferred in encrypted way and decrypted when the data want to be used. In this paper we provide reviews on various encryption techniques on images in the literature.

Authors find it's difficult to suggest for the best-fit algorithms to use, in suggestion of mathematical process to make nth an infinite over power dependent value, it causes mathematical error. It can be resolved by using developed functions that already designed for the suitable problems.

In general, any channel which can carry information from a secure area to the outside should be studied as a potential risk. Implementation-specific timing characteristics provide one such channel and can sometimes be used to compromise secret keys. Vulnerable algorithms, and systems need to be revised to incorporate measures to resist timing cryptanalysis and related attacks.

CHAPTER THREE

SYSTEM ANALYSIS AND DESIGN

3.1 Data acquisition and planning

To identify all the information and requirement such as hardware and software, planning must be done in the proper manner. The planning phase has two main elements namely data collection and the requirements of hardware and software

3.1.1 Data collection

Data collection is a stage in any area of study. At this stage I planned about the project's resources and requirements, literature studies and schedule to get more information in this study. All the materials are collected from journal, texts book and research papers gathered from libraries and Internet. Within the data collection period I have found information security has to do with our life and property to make our belongings save and secure that lead to information security development.

Information security has a lot of way that many as discovered to save us from attack of malicious tracker, but this project is considering for cryptographic techniques. The algorithms to be use is the encryption and decryption techniques.

3.1.2 Hardware and software requirement

Hardware Requirement

Below is the list of the tool components and the other material that will support to complete this project.

1. Microsoft system (DOS)
2. Visual display console
3. Operating System: Windows 7/8/10
4. RAM: 1GB
5. Memory: 500GB

Software Requirement

1. Programming languages: PHP 7.2.18, Java script (JS)
2. Tools: Sublime text editor 3 IDE stable version,
3. Database: MySQL 5.7.26
4. Web server: Apache 2.4.39
5. Web Browser: chrome, fire-fox, Microsoft edge or any others

The software is an open source that can be download from internet wamp server (http local server) it combines the php, database, and apache and while sublime can be download as a standalone software applications.

3.2 Explanation of the proposed analysis and design

This chapter is all about explanation of methodology that is being used to make this project complete and working well. Many methodology or findings are from this field mainly generated from journals for others to take advantages and improve as upcoming studies. The method is used to achieve the objective of the project that will accomplish a perfect result. In order to evaluate this project, this project methodology will be using the System Development Life Cycle (SDLC) to aid a clearer explanations and illustrations, generally three major steps, which is planning, implementing and analysis.

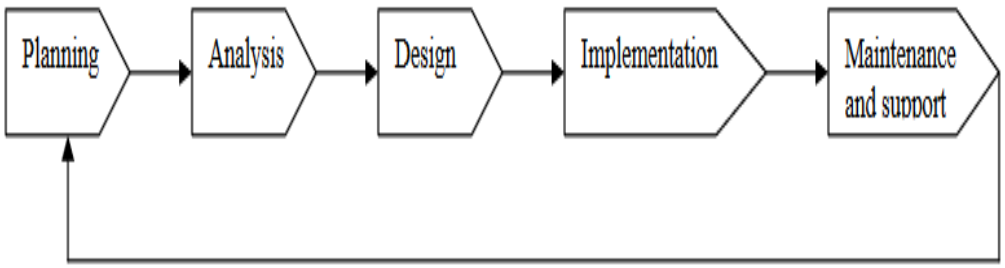


Figure: 3.0 *System Development Life Cycle (SDLC) "image was downloaded from* http://dspace.unimap.edu.my*"*

This project used three major steps to implement project starting from planning, implementing and testing. All the methods used for finding and analysing data regarding the project related.

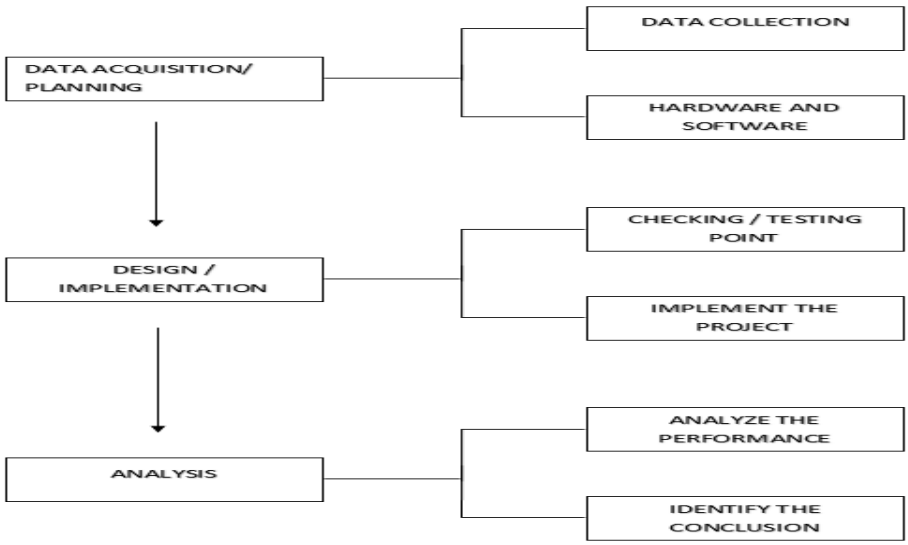


Figure: 3.1 *Steps of Methodology*

3.3 System design

In this action my project and research has been clearly stated the benefits to information security the advantages of this project, also the limitation can be furtherly state as technologies grows and more development of information security using one of the most commonly known techniques is cryptography techniques.

The project design and implantation will be using the approaches adopted from system design techniques and also this often used by scholars and experts on system design. These are the following design tools Flowchart, Waterfall Design, Data Flow Diagram, Use-Case, Sketch/Drawing and others.

3.3.1 Designing and implementing this project

In making the project will simulate and model the real goal for protecting information, the tools that are in modules and when those modules are embedded and merged together and make the complete of the project the needful tools are already state in the above in hardware and software requirement.

Data Information Security Using Data Cryptographic Techniques: Full System Analysis, The use-case design is showing the activities that can be done on the implementation of this application software, in this project work assured data security by cipher ideology (change the nature of the data) after that the software can also responsible of backing up whatever user does over the system and as well to feedback on any related issues that may arise.

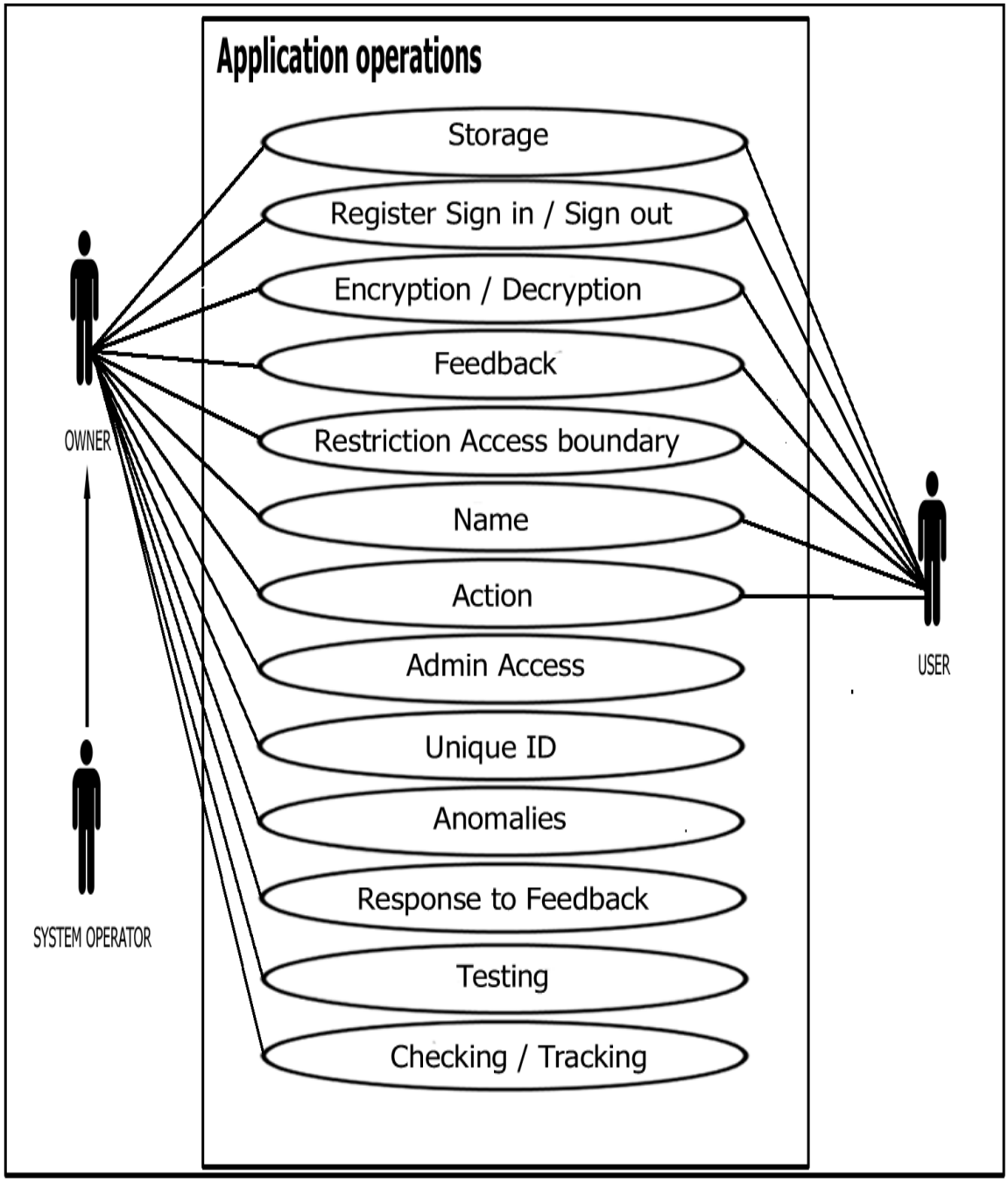


Figure: 3.2 *Full system Design using use-case*

Data Information Security Using Data Cryptographic Techniques: Owner Operations Analysis, At the owner section, the owner has the full action on the project. The owner can tell what the system will do and how it will work, they can perform every action and processes on the project, the project uses cryptography to secure user information or raw and processed data.

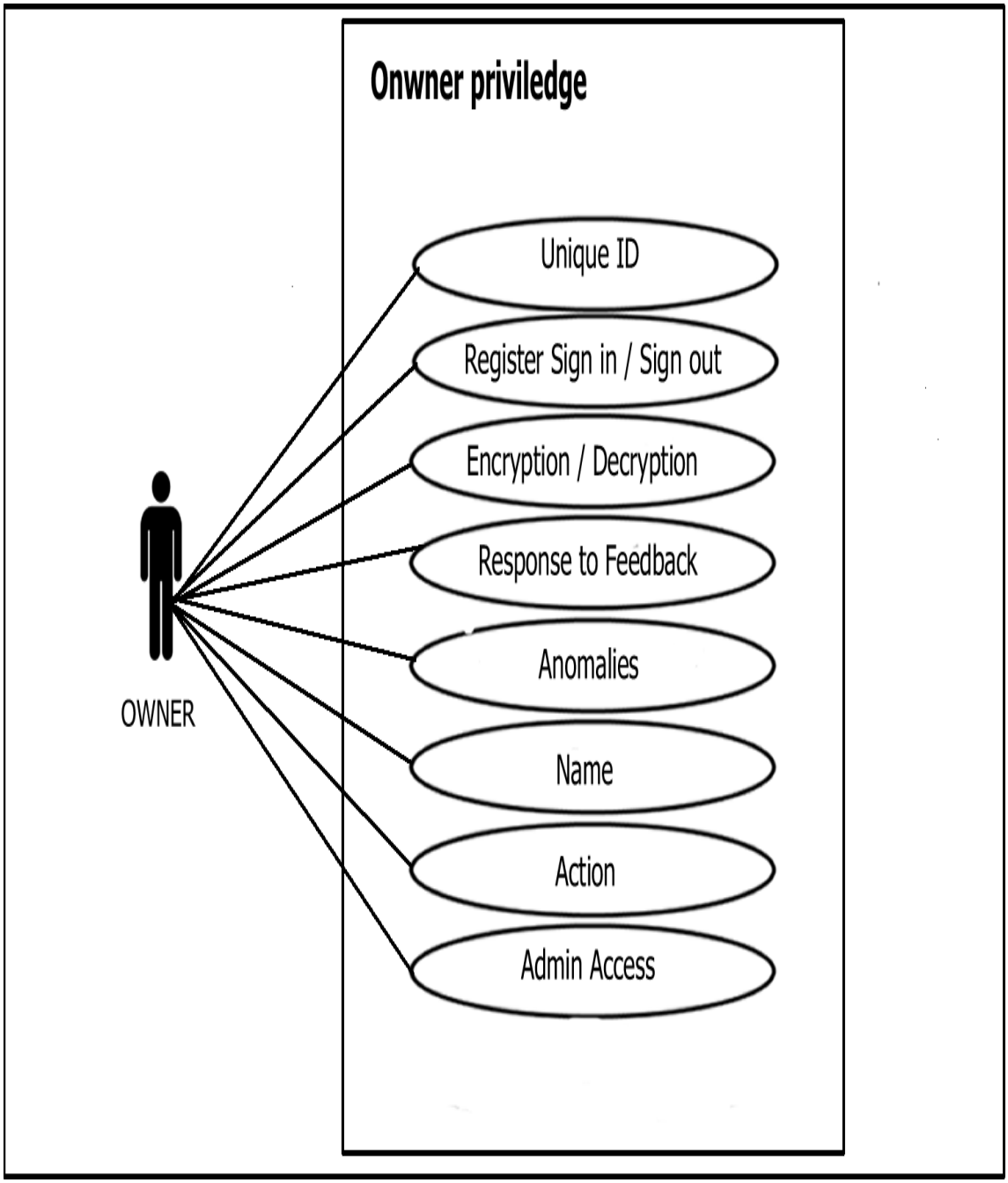


Figure: 3.3 *Owner at operation using use-case*

Data Information Security Using Data Cryptographic Techniques: User Analysis, At the user section, the user will have a limitation over the application software, a user can register under the note of the software record of another time use, the user can also perform the mean goals of this project i.e. information security. And finally, thee user can ask question and suggest for any update for the system.

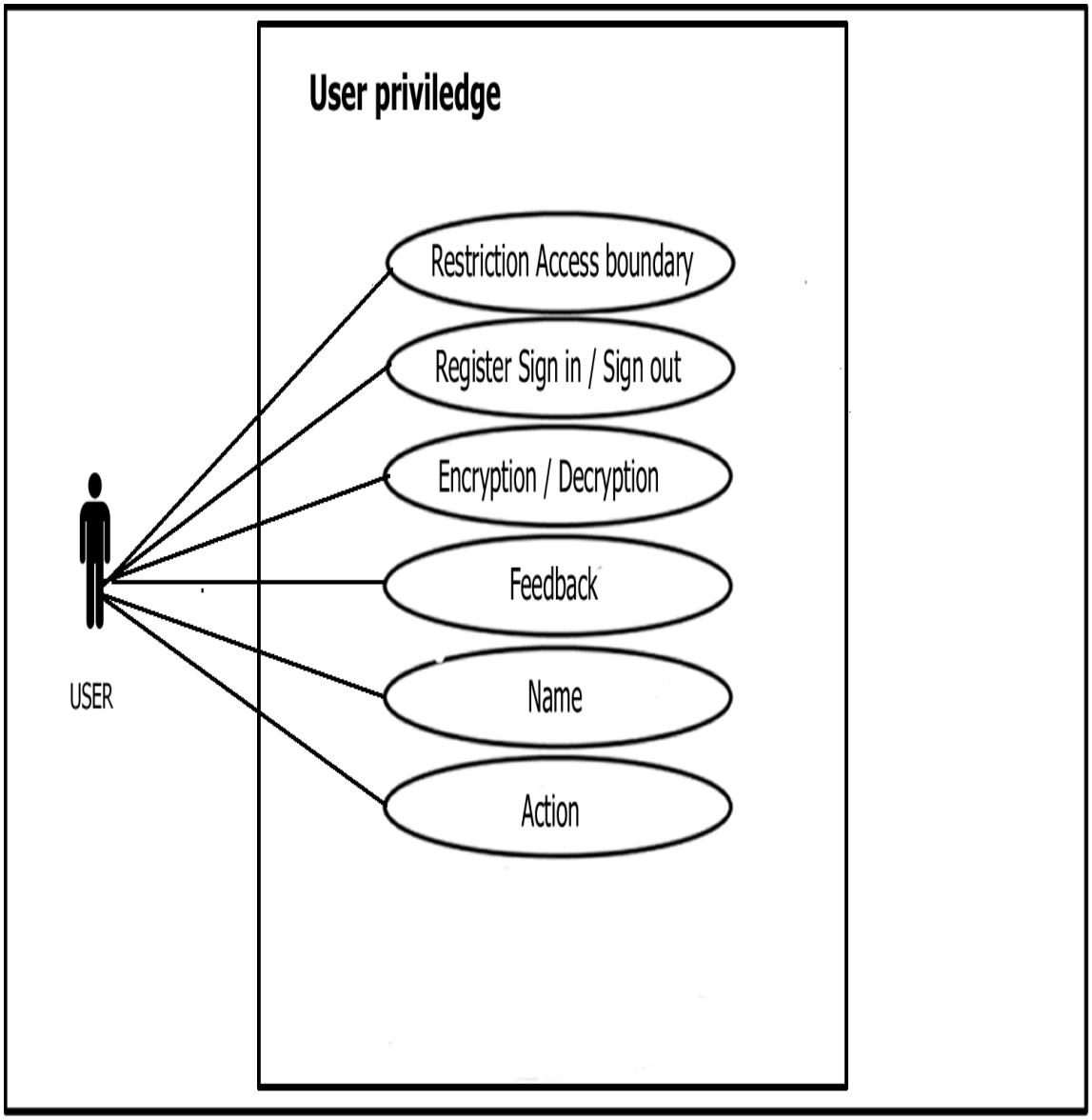


Figure: 3.4 *User at operation using use-case*

Data Information Security Using Data Cryptographic Techniques: Operator Analysis, In the operator section, the operator will be grant to the full system because the operator is going to be the administrator who's going to be at the system for any update. The operator isn't full time operator, which the system will be review may be once in a week. And also the operator the analysist submit remarks on any of processes that are granted.

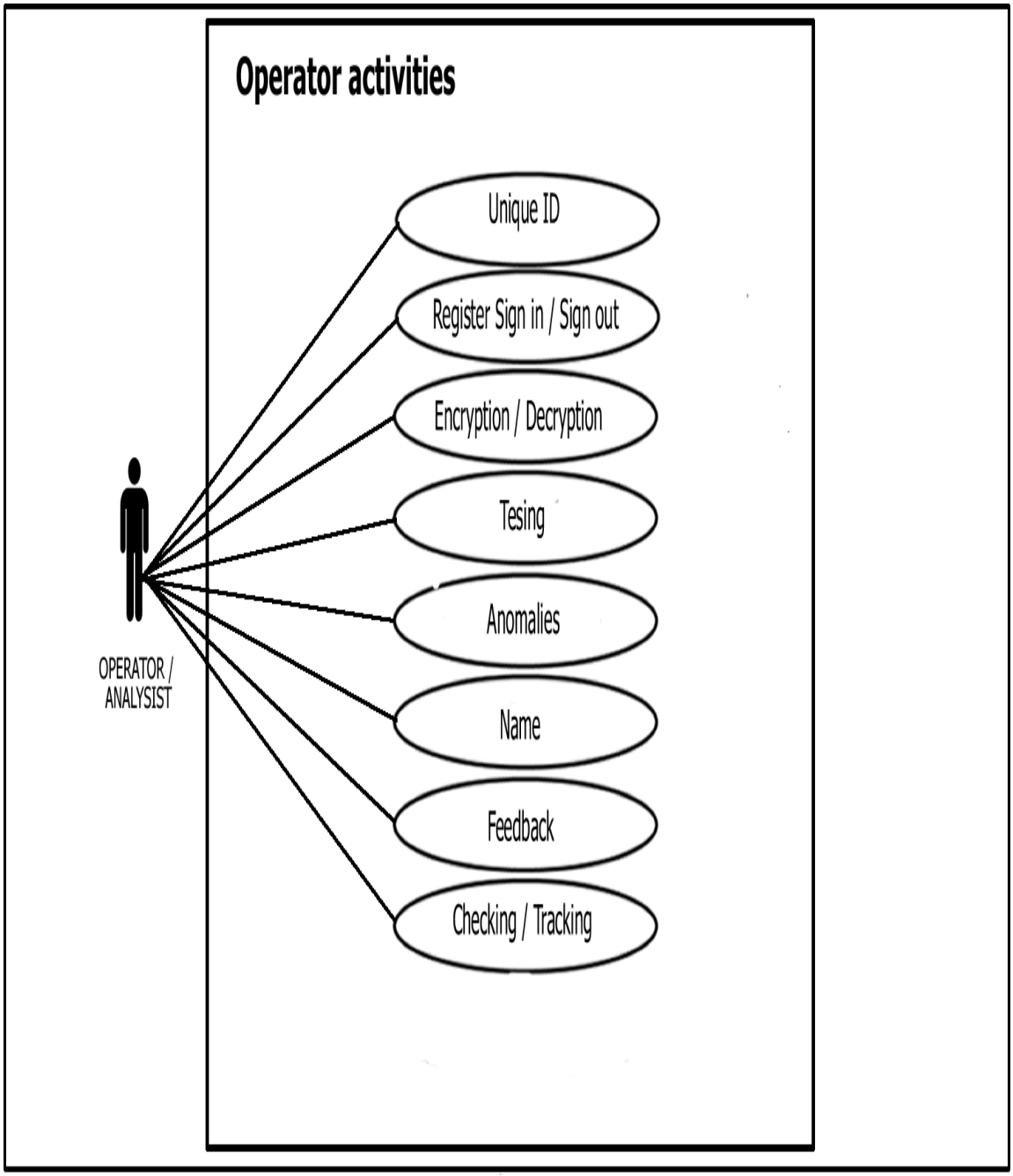


Figure: 3.5 *Operator/Analysist at operation using use-case*

Data Information Security Using Data Cryptographic Techniques: Backend Analysis, This data flow diagram (DFD) is explaining the logical and computational of the project the shown below figure explain how we couple modules together be achieve my aim and objective, the testing section explain in detail of how all these modules works.

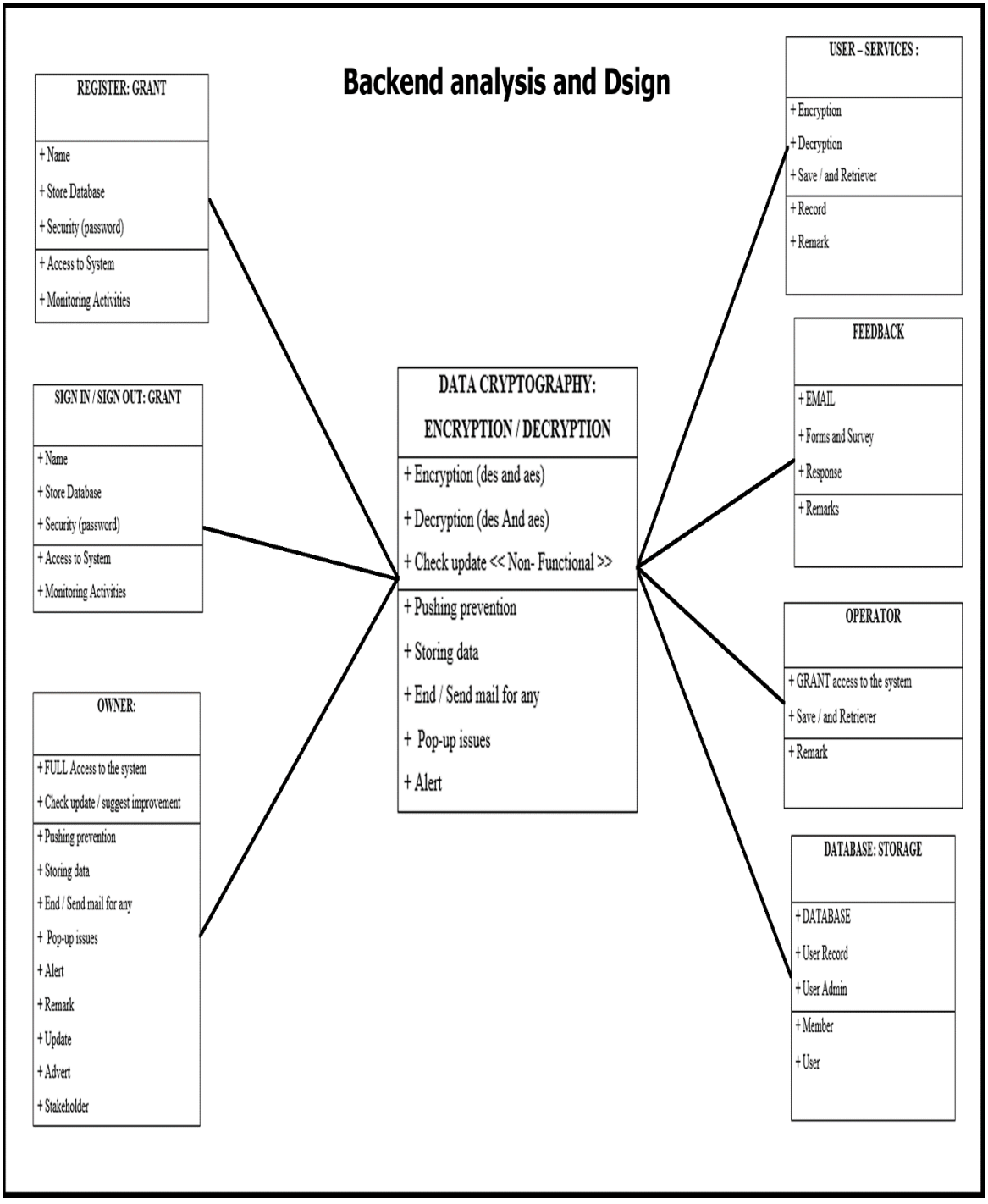


Figure: 3.6 *Backend Processes*

Data Information Security Using Data Cryptographic Techniques: Database Analysis, this project capture some of user record to put some computation secure also to make them work perfectly.

Meet us, this's a section design so that the user can ask for the contact of the project designer, by doing so the program force to their record such unique ID, each user Name, Email is a place I return the message as feedback and the body of the composed message. And the database did the same thing with other Entities.

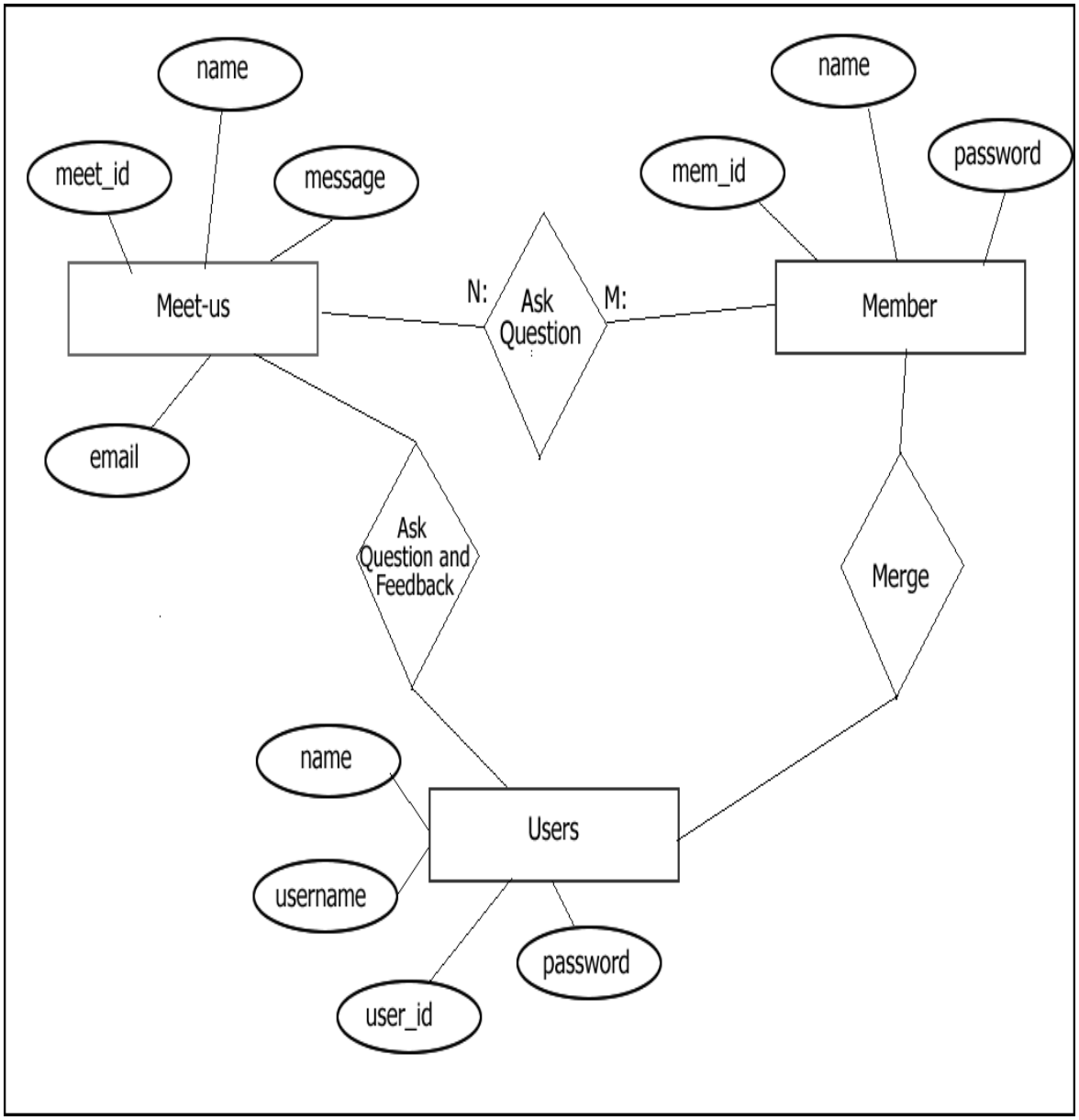


Figure: 3.7 *Entity Relational Model showing database adopted in the development of this project*

3.4 Algorithm

3.4.1 Encryption algorithm (DES)

```
Cipher (plainBlock[64], RoundKeys[16, 48], cipherBlock[64]) {
        permute (64, 64, plainBlock, inBlock, InitialPermutationTable)
        split (64, 32, inBlock, leftBlock, rightBlock)
        for (round = 1 to 16)
          {
        mixer (leftBlock, rightBlock, RoundKeys[round])
        if    (round!=16) swapper (leftBlock, rightBlock)  }
        combine (32, 64, leftBlock, rightBlock, outBlock)
          permute (64, 64, outBlock, cipherBlock, FinalPermutationTable)
          }
        mixer (leftBlock[48], rightBlock[48], RoundKey[48]) { copy (32, rightBlock,
T1) function (T1, RoundKey,T2) exclusiveOr (32, leftBlock, T2, T3) copy (32, T3,
rightBlock)
          }
        swapper (leftBlock[32], rigthBlock[32])  function (inBlock[32],
RoundKey[48], outBlock[32])
        {
        permute (32, 48, inBlock, T1, ExpansionPermutationTable)
        exclusiveOr (48, T1, RoundKey,T2)
        substitute (T2, T3, SubstituteTables) permute (32, 32, T3, outBlock,
StraightPermutationTable)}
        substitute (inBlock[32], outBlock[48], SubstitutionTables[8, 4, 16]) {
        for (i = 1 to 8)  {
        row  <- 2 * inBlock[i * 6 + 1] + inBlock [i * 6 + 6]
        col  <-8 * inBlock[i * 6 + 2] + 4 * inBlock[i * 6 + 3] +  2 * inBlock[i * 6 + 4] +
inBlock[i * 6 + 5]
          value    =    SubstitutionTables [i][row][col]
        outBlock[[i    * 4 + 1]  <- value / 8;    value  <- value mod 8
        outBlock[[i    * 4 + 2]  <- value / 4;    value  <- value mod 4
        outBlock[[i    * 4 + 3]  <- value / 2;    value  <- value mod 2
        outBlock[[i    * 4 + 4]  ¨value
        } }
```

Table: 3.0 *Cryptography Encryption Algorithm (DES) table*

```
DesAlgorithm desAlgorithm = new DesAlgorithmImpl();
public String encrypt(String dataToEncrypt, SecretKey key, String algo) throws public
String      decrypt(String dataToDecrypt, SecretKey key, String algo)
public String decrypt(String dataToDecrypt, byte[] key, String algo)
throws NoSuchAlgorithmException, InvalidKeyException, BadPaddingException,
NoSuchPaddingException, IllegalBlockSizeException,
InvalidAlgorithmParameterException { Security.addProvider(new
BouncyCastleProvider());SecretKey secretKey =
getSecretKey(Algorithm.DES.getValue());
      print("Secret Key : " + DatatypeConverter.printHexBinary
(secretKey.getEncoded()));
      DesApp desApp = new DesApp(); print("Enter your data to encrypt :");
      Scanner scanner = new Scanner(System.in); String dataToEncrypt =
scanner.next(); print("Original data : " +
DatatypeConverter.printHexBinary(dataToEncrypt.getBytes()));
      String encryptedData = desApp.encrypt(dataToEncrypt, DesConstants.baseKey,
getAlgo(Algorithm.DES.getValue(),ChipperMode.CBC.getValue(),DesPaddingMode.
PKCS5_PADDING.getValue()));
      print("Encrypted Data : " + encryptedData);
      String decryptedData = desApp.decrypt(encryptedData, DesConstants.baseKey,
getAlgo(Algorithm.DES.getValue(),ChipperMode.CBC.getValue(),DesPaddingMode.
PKCS5_PADDING.getValue()));  print("Encrypted Data : " + encryptedData);
decryptedData = desApp.decrypt(encryptedData, secretKey,
getAlgo(Algorithm.DES.getValue(),ChipperMode.CBC.getValue(),DesPaddingMode.
PKCS5_PADDING.getValue()));
      print("DecryptedData : " + decryptedData); }
   private String getAlgo(String algo,String cipherMode, String paddingMode) {return
algo + "/" + cipherMode + "/" + paddingMode ; }
   private SecretKey getSecretKey(String algo) throws NoSuchAlgorithmException
{KeyGenerator keyGenerator = KeyGenerator.getInstance(algo); SecureRandom
secureRandom = new SecureRandom();
  } }
```

Table: 3.1 *Cryptography Decryption Algorithm (DES) table*

3.4.3 Encryption algorithm (AES)

```
AES Structure
– Acts on the complete block doesn't split the block into halves, (or fractions of any
sort) for different treatment Key Sizes accommodated are 128, 192 or 256 bit.
(128 likely to be the most common implementation)
– We assume a key size of 128 bits as input. This is expanded into 44 32 – bit words,
w[i].
– 4 words used at a time (128 bit), used once only at
   initial 'Add round key'
    in each of the 10 rounds
 Block size = 128 bit
KeyExpansion(byte key[16], word w[44])
{
word temp
for(i = 0; i < 4; i + +)
w[i] = (key[4*i], key[4*i + 1], key[4*i + 2], key[4*i + 3]);
for(i = 4; i < 44; i + +)
{
temp = w[i – 1];
if ( i mod 4 = 0)
temp = SubWord(RotWord(temp)) + Rcon[i/4];
w[i] = w[i-4] + temp
```

Table: 3.2 *Cryptography Encryption Algorithm (AES) table*

3.4.4 Decryption algorithm (AES)

```
Inverse Final Round { AddRoundKey, ShiftRows , SubBytes }
Inverse Main Round { AddRoundKey, MixColumns, ShiftRows, SubBytes }
Inverse Initial Round
AddRoundKey
```

Table: 3.3 *Cryptography Decryption Algorithm (AES) table*

3.5 Process flow

The flow this research project is the step by step the operation goes at the implementation level; the following are the guide way through the using of this software application

1. Start: Get the system under your noise
2. Register under the system / sign in
3. Read and learn the interface
4. Perform your Cryptographic Operations: Encryption, Decryption
5. Ask Question, Suggest an Update for the system
6. Close / sign out / Out of the application

CHAPTER FOUR

IMPLEMENTATION

4.1 Screenshots, Explanation of table and values obtained during simulation and Chart

The Application Interface

The application interface works with the concept of web and uses web-browsers as a system tool to access the software application. The user interface is friendly, in recent designs of applications, developers design to make the uses ease. The interface contains the functional requirement to satisfy the goals of this project.

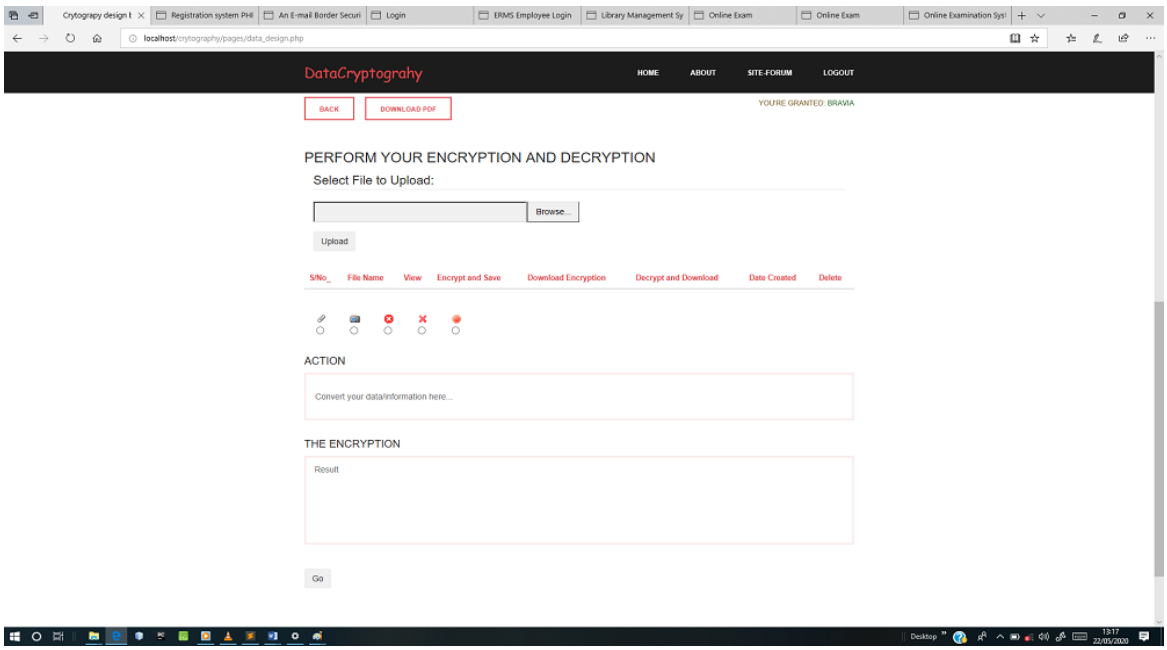


Figure 4.0: *Application Interface structure*

Data Encryption and Decryption. This is where the work lies the user can do multiple of activities such that the client can view, download and encrypt, encrypt and save, decrypt and download and delete.

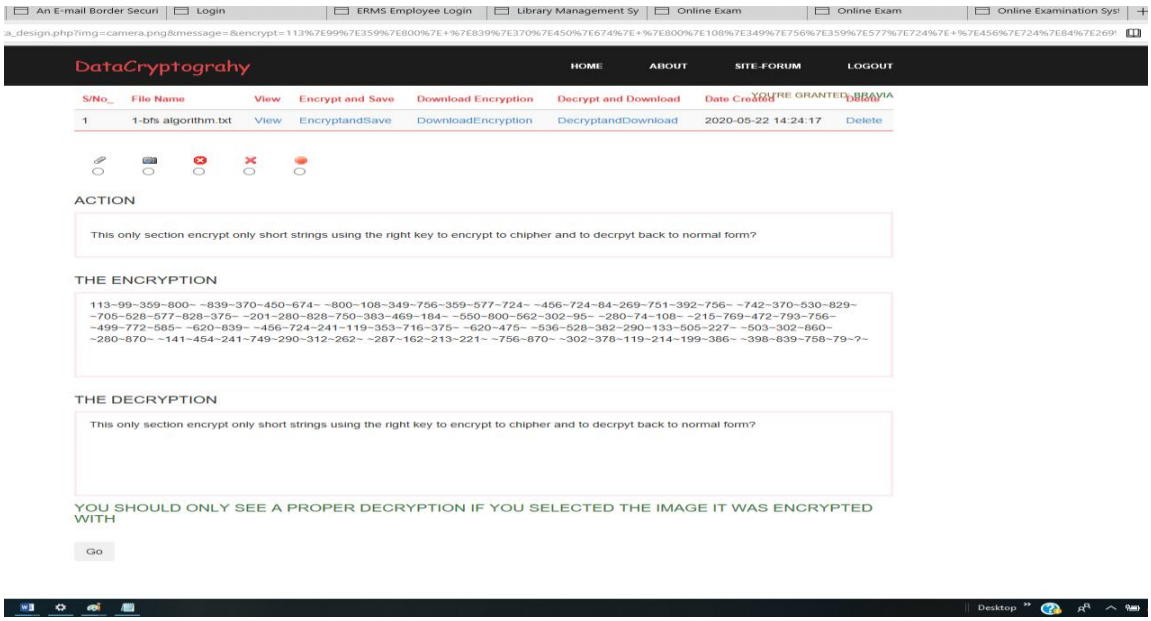


Figure 4.1: *Data Encryption and decryption Process*

To view once the work is uploaded there is already a dynamic button which automatically check for update and make it a hypertext link. Whenever clicked it will pop-up the origination file such to be sure of the file working with on the application.

The Download and Encrypt button it's the first action to be done if user want to convert the original to cipher form. User can click the button to encrypt the file and when the application pops a message on the means the file is encrypted.

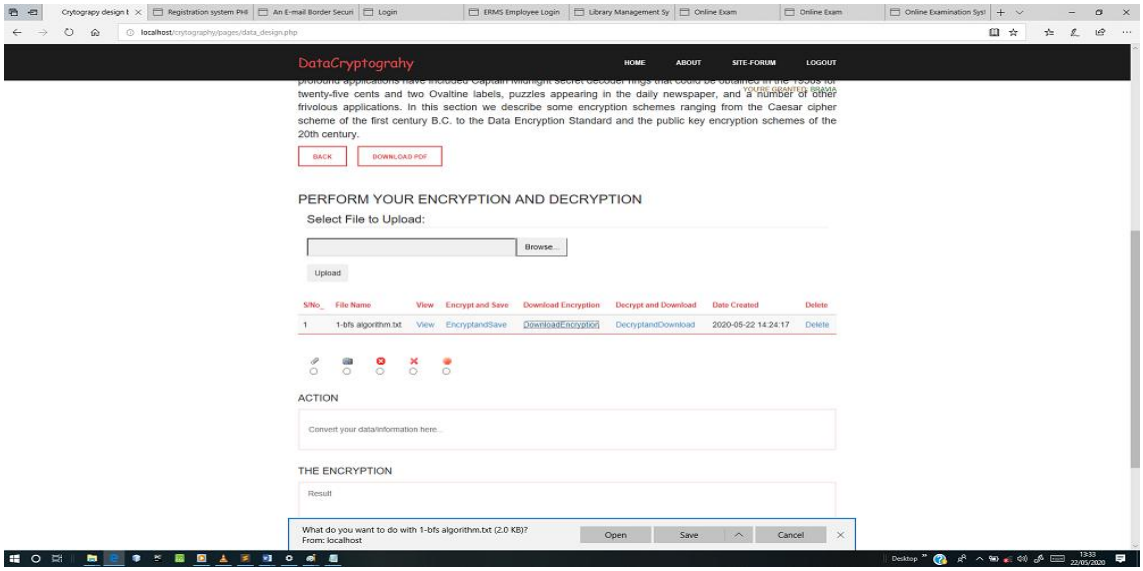


Figure 4.2: *Data Encryption and Decryption Download Process*

The Encrypt is stored on a local disc which local server was installed. Encryption of any file can be done the system, and the application as well save the encryption to where user can easily get that by downloading the work which also easy too. As the user interface is easy to user.

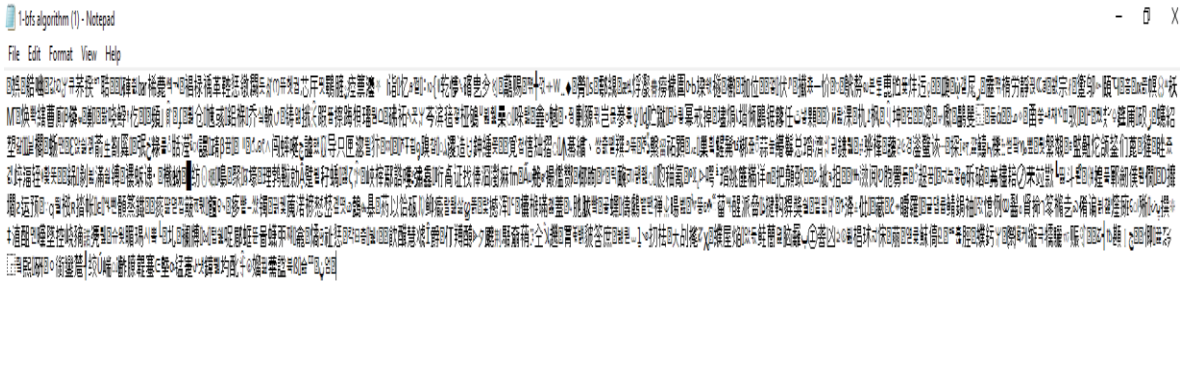


Figure 4.3: *The Cipher Result Encryption Output*

Decrypt and Download this works for already encrypted files which needs to be convert back to original file and its original contents, the application made it avail to download the original file/data to any directory user chooses on another system without any crash.

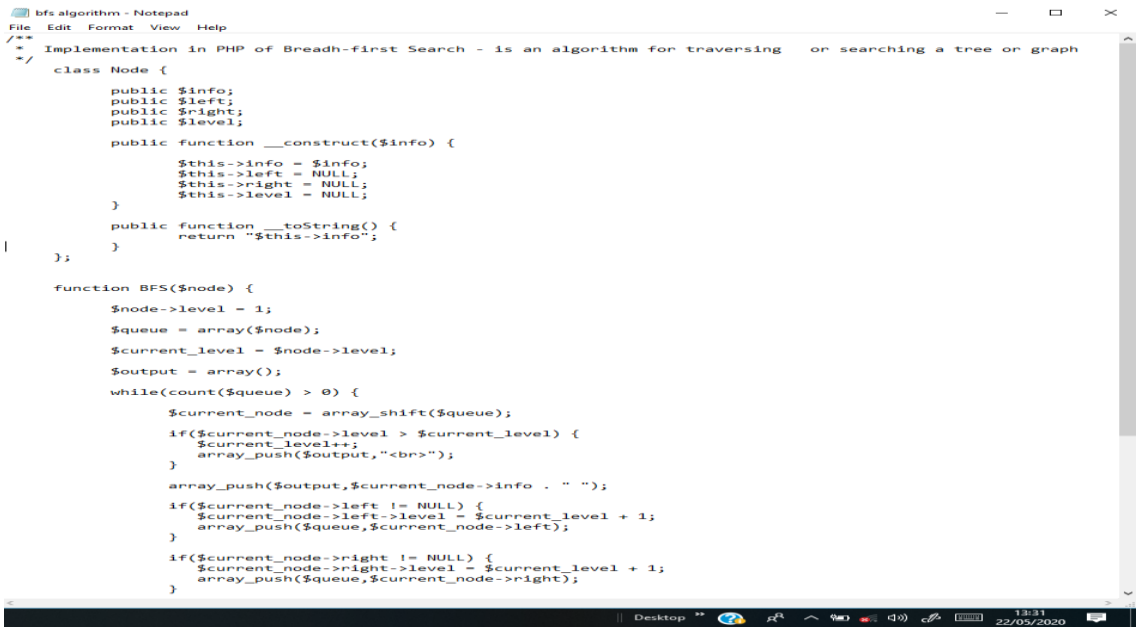


Figure4.4: *The Conversion of Encrypted File to Original File*

Finally Delete. Any work on the system can be deleted at any point in time the purpose for this is that user may want to delete confidential file immediately also to provide security to the file. System admin will also check for the files that has lasted long for days if not week to clean up storage and to make more security to data of people stored on our database.

Delete also make the users knows there is nothing to do with any organization confidential file or data and it's also stores the record of easy users and whatever transaction performed on the application.

CHAPTER FIVE

SUMMARY, CONCLUSION, RECOMMENDATION

This chapter presents the summary of the whole research study and the conclusion to the research. This section also presents related topics or areas for future research.

5.1 Summary

This project is Data/Information security using cryptographic operations. Used the techniques operations to build a software applications which can execute the two basics of cryptography i.e. the encryption and decryption. Also, this project specifically targeted the two most use algorithms which they advanced encryption/decryption standard (AES) and data encryption/decryption standard (DES).

5.2 Conclusion

Generally, this project is a successful one the application can be put in use in any organizations who's in needing to secure and protect their softcopy document as a Data/Information integrity. The application closely works like a human manual thinking; it can check for the users in use of the cryptographic operations and multi users can use the sever at a time. Finally remarking this project firstly this application is time friendly. It works superfast [Nano seconds]. The interface makes an ease understand to use; it's mostly looks like working on a gallery i.e. is explanatory and easy enough to use without any other expert and operator guardians. And secondly the application can be put in use in any

organizations who's in needing to secure and protect their softcopy document as a Data/Information integrity. The encouraging part is that the application was design using worldwide web integration which moveable and can be use anywhere, anytime, in the world.

5.3 Recommendation

In future works, this application could be extended by implementing dynamic cryptography techniques. In addition, developing machine learning features to the system and determining a way to select optimum points to hide data and information will further improve the system. Currently, when shares are merged, there is no authentication testing mechanism, Artificial intelligence testing would improve the quality of cryptographic operations overlapping, which can be achieved through correlation algorithms. This recommends the study entitled data/information security using cryptography, to the future researchers who wants to upgrade this study. The advantage of this study is that it can determine whether the given decryption key is correct or not, that makes the files more secure because if the decryption key inputted is incorrect, it will not decrypt the encrypted files, but it cannot detect whether a file is already encrypted or not.

This study is limited to text files and other multimedia files only so I recommend to the future researcher that they can improve this study using different file format and extend the files size to fit as appropriate format.

REFERENCE

Dhanalakshmi, D., & Ravichandran, T., (2013). *A New Level of Image Processing Technique Using Cryptography and Steganography*. http://ijsetr.org/wp-content/uploads/2013/07/IJSETR-VOL-2-ISSUE-3-659-665.pdf

E-copy, M. (2015). *Tackling Insurgency Using Cryptography*. https://iproject.com.ng/computer-science/tackling-insurgency-using-cryptography/index.html

Gulbarga, & Karnataka (2015). *FPGA Based Cryptography for Internet Security*. https://www.ijcsmc.com/docs/papers/October2015/V4I10201501.pdf

Herder, C., Md, Yu., Koushanfar, F., & Devadas, S. (2014) *Physical Unclonable Functions and Applications*: A Tutorial. Proc. IEEE, 102, 1126–1141. [Google Scholar] [CrossRef].

Hussain, A., & Manish, T., (2015). *Development of Encryption and Decryption Technique to Secure the Confidential Data.*

Joseph C. & Richard, K. (2016) *Energy Consumption Cost Analysis of Mobile Data Encryption and Decryption*. IEEE International Conference on Mobile Services.

Mattkings, B. W. (2019). *Design and implementation of network security using cryptography.* https://www.classgist.com/projectdetails.aspx?id=1841

Mijanur, R. (2016). *Development of Cryptography-Based Secure Messaging System.* https://www.researchgate.net/publication/312047086_Development_of_Cryptography-Based_Secure_Messaging_System

Melisa, S. (2019). *Expert System for Computer Security: Data Encryption, Decryption and Key Hash Algorithms.* https://www.projecttopics.org/expert-system-for-computer-security-data-encryption-decryption-and-key-hash-algorithms.html

Michael, B., & Govindan, P. (2016) *RSA Based Biometric Encryption System Using FPGA for Increased Security (IEEE)*.

Murali, B., Khan, H., & Madhumati, G. (2017). *Reconfigurable pseudo biotic key encryption mechanism for cryptography applications*. International Journal of Engineering & Technology, 7(1.5), 62-70. http://dx.doi.org/10.14419/ijet.v7i1.5.9124

Naveen, K. K. (2017). *Implementation of Secure Payment Transaction using AES encryption with extended Visual Cryptography.* http://sci.tamucc.edu/~cams/projects/519.pdf

Rahman, M., Akter, T., & Rahman, A. (2016). *Development of Cryptography-Based Secure Messaging System*. J Telecommunication System Manage 5: https://www.omicsonline.org/open-access/development-of-cryptographybased-secure-messaging-system-2167-0919-1000142.php?aid=80928

Singh, P., & Kumar, S. (2017). *Study & analysis of cryptography algorithms: RSA, AES, DES, T-DES, blowfish*. International Journal of Engineering & Technology*, 7(1.5), 221-225*. doi:http://dx.doi.org/10.14419/ijet.v7i1.5.9150

Santhosh, R., & Bharanidharan, R. (2017). *Neighbour discovery-based security enhancement using threshold cryptography for IP address assigning in network*. International Journal of Engineering & Technology, 7(1.1), 439-443. http://dx.doi.org/10.14419/ijet.v7i1.1.11243

APPENDIX

Appendix A Project Libraries Codes

Advance Encryption Standard (AES) and Data Encryption Standard (DES)

```
<?php
 //echo "<h1>ENCRYPTION AND DECRYPTION</h1>";
 $filename = "../assets/video/Matrix.mp4";
 //encrypt file
 encrypt_file($filename, "encrypted/".$filename, 'secret-password');
 //decrypt file
 $decrypted = decrypt_file('encrypted/'.$filename, 'secret-password');
 //header('Content-type:application/txt');
 if (file_exists($filename)) {
   header('Content-Description: File Transfer');
   header('Content-Type: application/octet-stream');
   header('Content-Disposition: attachment; filename="'.basename($filename).'"');
   header('Expires: 0');
   header('Cache-Control: must-revalidate');
   header('Pragma: public');
   header('Content-Length: ' . filesize($filename));
   readfile($filename);
   exit;
}
 fpassthru($decrypted);
 function encrypt_file($file, $destination, $passphrase) {
```

```
        $handle = fopen($file, "rb") or die ("could not open the file");
        $contents = fread($handle,filesize($file));
        fclose($handle);
        $iv = substr(md5("\x18\x3C\x58".$passphrase, true),0,8);
        $key = substr(md5("\x2D\xFC\xD8".$passphrase, true).md5("\x2D\xFC\xD8".$passphrase, true),0,24);
        $opts = array('iv'=>$iv, 'key'=>$key);
        $fp = fopen($destination, 'wb') or die ("Could not open file for writting");
        stream_filter_append($fp, 'mcrypt.tripledes', STREAM_FILTER_WRITE, $opts);
        fwrite($fp, $contents) or die ('Could not write to file');
        fclose($fp);
}
function decrypt_file($file, $passphrase) {
        $iv = substr(md5("\x18\x3C\x58".$passphrase, true),0,8);
        $key = substr(md5("\x2D\xFC\xD8".$passphrase, true).md5("\x2D\xFC\xD8".$passphrase, true),0,24);
        $opts = array('iv'=>$iv, 'key'=>$key);
        $fp = fopen($file, 'rb');
        stream_filter_append($fp, 'mdecrypt.tripledes', STREAM_FILTER_READ, $opts);
        return $fp;
}
?>
```

Appendix B Project Libraries Codes

Data Encryption Standard

```
<?php
class imageKey {
```

```
    //The default path to the image key file
    public $imgPath = '';
    //The encrypted character delimeter
    public $msgDelim = '~';
    //Holds the message key from image key file
    public $msgKey;
    //Holds any error message
    public $errorText;
    public function __construct( $file='', $path='', $curl=false, $verify=false ){
        if( !empty($file) ){
            $this->createKey($file,$path,$curl,$verify);
        }
}
    public function createKey( $file, $path='', $curl=false, $verify=false ){
        $path = ( empty($path) ) ? $this->imgPath : $path;
        if( $curl ){
            $ch = curl_init();
            curl_setopt($ch, CURLOPT_URL, $path.$file);
            curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);
            curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, $verify);
            $this->msgKey = curl_exec($ch);
            if( curl_error($ch) ){
                $this->errorText = 'curl error: '.curl_error($ch);
            }else{
                $this->msgKey = base64_encode($this->msgKey);
            }
```

```
        curl_close($ch);
    }else{
        if( $this->msgKey = @file_get_contents($path.$file) ){
            $this->msgKey = base64_encode($this->msgKey);
        }else{
            $error = error_get_last();
            $this->errorText = 'error: '.$error['message'];
        } }}
public function encryptMsg( $message ){
    $message = str_replace($this->msgDelim,'',$message);
    $msgArray = str_split($message);
    $msgCrypt = '';
    foreach( $msgArray as $msgChar ){
        $msgPattern = ( $msgChar == '^' ) ? '\^' : $msgChar;
        $msgPattern = ( $msgChar == '$' ) ? '\$' : $msgPattern;
        $msgPattern = ( $msgChar == '.' ) ? '\.' : $msgPattern;
        $msgPattern = ( $msgChar == '|' ) ? '\|' : $msgPattern;
        $msgPattern = ( $msgChar == '?' ) ? '\?' : $msgPattern;
        $msgPattern = ( $msgChar == '*' ) ? '\*' : $msgPattern;
        $msgPattern = ( $msgChar == '+' ) ? '\+' : $msgPattern;
        $msgPattern = ( $msgChar == ')' ) ? '\)' : $msgPattern;
        $msgPattern = ( $msgChar == '(' ) ? '\(' : $msgPattern;
        $msgPattern = ( $msgChar == '[' ) ? '\[' : $msgPattern;
        $msgPattern = ( $msgChar == '{' ) ? '\{' : $msgPattern;
        $pattern = '/$msgPattern/';
        preg_match_all($pattern,$this->msgKey,$matches,PREG_OFFSET_CAPTURE);
```

```
        if( !empty($matches[0]) ){
            $pos = array_column($matches[0],1);
            shuffle($pos);
            $msgCrypt .= $pos[0].$this->msgDelim;
        }else{
            $msgCrypt .= $msgChar.$this->msgDelim;
        }}
      return $msgCrypt; }
   public function decryptMsg( $msgCrypt ){
$msgArray = explode($this->msgDelim,$msgCrypt);
$message = ''; foreach( $msgArray as $msgPos ){ @$message .= ( is_numeric($msgPos) ) ? $this->msgKey[$msgPos] : $msgPos; }
     return $message; }
}
?>
```

Appendix C Cryptographic Techniques

Appendix D Classifications of Cryptography

Appendix E Synchronization of Cryptographic Operations