B.Sc. in Computer Science and Engineering Thesis

Recapitulating the Development Initiatives of Symmetric Key Encryption Algorithm and Prototyping LWE for Small Scale Data Cryptanalysis

Submitted by

Soumen Roy
ID-201114017

Md Abu Sohaib
ID-201114037

Md Khaled Kamal
ID-201114045

Supervised by

Jahidul Arafat

Lecturer

Department of Computer Science and Engineering(CSE)

Military Institute of Science and Technology (MIST) , Dhaka-1216, Bangladesh.

Department of Computer Science and Engineering

Military Institute of Science and Technology

December 2014

CERTIFICATION

This thesis paper titled **"Recapitulating the Development Initiatives of Symmetric Key Encryption Algorithm and Prototyping LWE for Small Scale Data Cryptanalysis"**, submitted by the group as mentioned below has been accepted as satisfactory in partial fulfillment of the requirements for the degree B.Sc. in Computer Science and Engineering in December 2014.

Group Members:

Soumen Roy

Md Abu Sohaib

Md Khaled Kamal

Supervisor:

Jahidul Arafat
Lecturer
Department of Computer Science and Engineering(CSE)
Military Institute of Science and Technology (MIST) , Dhaka-1216,
Bangladesh.

CANDIDATES' DECLARATION

This is to certify that the work presented in this thesis paper, titled, "**Recapitulating the Development Initiatives of Symmetric Key Encryption Algorithm and Prototyping LWE for Small Scale Data Cryptanalysis**", is the outcome of the investigation and research carried out by the following students under the supervision of Jahidul Arafat, Lecturer, Department of Computer Science and Engineering(CSE), Military Institute of Science and Technology (MIST) , Dhaka-1216, Bangladesh..

It is also declared that neither this thesis paper nor any part thereof has been submitted anywhere else for the award of any degree, diploma or other qualifications.

Soumen Roy
ID-201114017

Md Abu Sohaib
ID-201114037

Md Khaled Kamal
ID-201114045

ACKNOWLEDGEMENT

We are thankful to Almighty Allah for his blessings for the successful completion of our thesis. Our heartiest gratitude, profound indebtedness and deep respect go to our supervisor, Jahidul Arafat, Lecturer, Department of Computer Science and Engineering(CSE), Military Institute of Science and Technology (MIST) , Dhaka-1216, Bangladesh., for his constant supervision, affectionate guidance and great encouragement and motivation. His keen interest on the topic and valuable advices throughout the study was of great help in completing thesis.

We are especially grateful to the Department of Computer Science and Engineering (CSE) of Military Institute of Science and Technology (MIST) for providing their all out support during the thesis work.

Finally, we would like to thank our families and our course mates for their appreciable assistance, patience and suggestions during the course of our thesis.

Dhaka
December 2014

Soumen Roy

Md Abu Sohaib

Md Khaled Kamal

ABSTRACT

In the digital world, which is currently evolving and changing at such a rapid pace, the security of digital information has become increasingly more important. Due to great development of secret contact and communication in the world, the necessity of communication security is becoming more and more significant. Cryptography has specific role to protect secret files like secret documents from unauthorized access. Cryptography algorithms are classified into two types, Public-key producing and symmetric-key producing algorithms. In this paper, we suggested a new robust and lightweight cryptography algorithm named as LWE algorithm to increase security at a cheaper cost in the Symmetric-key producing algorithm. For writing this paper, we followed the deductive approach that is first we studied the existing symmetric key algorithms, formed a hypotheses, developed this new algorithm, performed experiments on data, analyzed the results and then finally came to the conclusion. Our algorithm has better performance and lower error rate than most other lightweight encryption algorithms and is reliable and valid as the efficiency has been tested on a large scale of data. Our only limitation is that this algorithm is effective for small scale data only. However, from the experiments done and analysis performed, it can safely be said that this LWE algorithm will be greatly helpful for individual users and small companies who want to protect their data efficiently at a minimum cost.

TABLE OF CONTENT

LIST OF FIGURES

LIST OF TABLES

LIST OF ALGORITHMS

LIST OF ABBREVIATION

LWE : Lightweight Encryption
R-LWE: Robust Lightweight Encryption
IT : Information Technology
PIN : Personal Identification Number
ATM : Automated Teller Machine
MAC : Message Authentication Code
InfoSec: Information Security
XOR : Exclusive-Or
AES : Advanced Encryption Standard
odt : ODF Text Document
fodt : Open Document Text (Flat XML)
uot : Unified Office Text
NIST : National Institute of Standards and Technology
DES : Data Encryption Standard
IBM : International Business Machines
NBS : National Bureau of Standards
NSA : National Security Agency
FIPS : Federal Information Processing Standard
PHT : Pseudo-Hadamard Transform
SAFER: Secure and Fast Encryption Routine
CAST : Carlisle Adams and Stafford Tavares
GPG : GNU Privacy Guard
PGP : Pretty Good Privacy
RC4 : Rivest Cipher 4
ARC4 : Alleged Rivest Cipher 4
TLS : Transport Layer Security
3DES : Triple Data Encryption Algorithm
EPKE : Enveloped Public Key Encryption
FPKE : Forward Public Key Encryption
IPKE : Inverse Public Key Encryption
DH : DiffieHellman Key Exchange
RSA : Ron Rivest, Adi Shamir and Leonard Adleman
DSA : Digital Signature Algorithm
DSS : Digital Signature Standard

LWC : Lightweight Cryptography
RFID : Radio-Frequency Identification
ISO : International Organization for Standardization
IEC : International Electrotechnical Commission
JTC : Joint Technical Committee
RAM : Random-Access Memory
NCSA : National Center for Supercomputing

CHAPTER 1
INTRODUCTION

Information technology security is a rapidly changing field of research with emerging application domains and ever-improving hardware and software. With development of Information and Communication Technology, data transmission becomes more critical day by day. Higher security for transmitting data is especially required. Cryptography is thereby referred as a term to study the information secrets and secure those. It is probably most important aspect of communication security and is becoming increasingly important as a basic building block for information security. Cryptography has a specific role to protect secret communication from unauthorized access and to prevent such attacks encryption technique is the best way. Since twentieth century, cryptography has become one of the fields of modern science to protect secret communication. Due to the importance of cryptography in protecting secret communication, security of information has become a major issue during the last decade. Thereby this chapter has elaborated on the information security concerns and assessed the background of symmetric key encryption and decryption on small scale data in section 1.1. It also has defined the scopes and objectives of the research in section 1.4 and 1.5. Finally the chapter ends with a summarized description on individual chapters of this research in section 1.8.

1.1 Background of the Research

Information security plays a pivotal role during internet communication in todays era of technology. It is tremendously important for people committing e-transactions. For naive people it may seem to be not that necessary or increased security may provide comfort to paranoid people but the truth is that it is absolutely essential when communication is carried between tens of millions of people daily [1]. There are various cryptography methods that provide a means for secure commerce and payment to private communications and protecting passwords [2]. Cryptography is necessary for secure communications; it is not by itself sufficient. Cryptographic algorithms are widely used in the financial sector to ensure security in various financial transactions. For instance, the algorithms are used to ensure confidentiality and integrity of data such as a personal identification number (PIN) and an account number in automated teller machine (ATM) transactions. The application of various

cryptography algorithms in ATM machines have been discussed in detail by Cheng and Kurita [1,2]. The algorithms are also used to authenticate counterparties of the transactions in Internet banking services. The use of cryptography algorithms in e-banking system is one of the common topics of modern IT researchers. Some of the remarkable works done on this topic are the studies done by Srinivasan and Hiltgen [3,4]. Generally, the algorithms used for cryptography applications are classified into two types, Asymmetric methods or public key cryptography and Symmetric methods or Symmetric key cryptography [5].

Symmetric ciphers are adopted to ensure confidentiality. To ensure integrity and authenticity, a message authentication code (MAC) based on symmetric ciphers or a digital signature based on asymmetric ciphers is adopted. This kind of protection is needed in every sphere of IT sector now-a-days. Individual persons, small industries and companies also need security, privacy and protection for data communication. For these areas, existing algorithms would be bulk some and redundant. But as we studied, we observed that there are no lightweight cryptography algorithms for these purposes. The study of Cunsolo & Distefano further justified it [5]. Lightweight algorithms which can operate on small size text and doc files with considerable reliability and validity would be enough for them. This happened due to the fact that cryptography algorithms are thought to be applied on large scale data. But the fact is that privacy, security and integrity as well as minimizing the time complexities is more important for small scale data than that is for large scale data when it requires fast cryptanalysis. This thereby signifies the importance of this present research and defines a logistic background to carry on it.

1.2 Statement of the Problem

This study has found that the lack of robust lightweight cryptography algorithms often encounter problems to secure small scale data of both higher education institutions and organizations. It has worked on recapitulating the development initiatives of symmetric key encryption. It also has focused to prototype a small scale algorithm for lightweight cryptanalysis on 1-512 KB data of cross platforms i.e., Linux, Windows and Mac OS. Thereby the study believes to develop a robust and lightweight symmetric key encryption algorithm that could enable the organizations to secure their small scale data and to avoid penetrations while reducing the time complexities.

1.3 Significance of the Research

The significance of a research depends on its present utility in the field. This study on symmetric key encryption and prototyping a lightweight algorithm for small scale data crypt-

analysis signify the todays day network and security system on several reasons. Hence a small number of algorithms exist in literatures those could fit best for small scale data encryption and decryption on cross OS platform, this research makes this process of crypt-analysis easy and affordable it ensures small scale data stability, integration and privacy. Manifavas mentioned that, it is not feasible for small industries to spend large amount of money to purchase costly antivirus software and other security measures, development of a light weight encryption affords thereby will emerge in the upcoming years which will be easy, reliable, fast and affordable [6] . Engels & Smith in their research on Hummingbird algorithm justified it further [7]. They said that a lightweight symmetric key encryption prototype will not only enable an organization to ensure fast data cryptanalysis but will also save the time and reduce the data complexities. This study thereby focused on to in-depth in this effort to provide the organizations with a cheap and effective crypto-alternative which will work effectively on cross OS platform. This algorithm believes to work on various categories of input data (.doc, .docx, .txt etc) of size 1-512KB and believes to act as a valid tool for cipher text analysis. This also has ensured time effectiveness in the field of symmetric key cryptanalysis and avoids complexities, but has ensured reliabilities. Thereby this study bears enough significance to contribute in this contemporary field of data security with a light weight symmetric key support.

1.4 Scope of the Research

This study has been experimental in nature. It has addressed the different aspects of existing symmetric key encryption algorithms including time and space complexities, strengths and weakness and rounds of cryptanalysis. It has fetched the best features from them; which could best fit the development of a new Light Weight Encryption (LWE) algorithm on small scale data. All these defined the basic structure of the present study. The scope was further delimits within the arena of testing the small scale data (size 1 to 512 KB) of diverse formats (.doc, .docx, .txt and equivalent formats) on three different OS platforms namely Windows, Linux and Mac OS. Thus the study believes to have profound impacts on the small scale data networks which remained to be the most ignored part of research in previous studies with less complexity and more time efficiencies.

1.5 Research Context

1.5.1 Purpose of the Research

The purpose of the research was to study and analyze the features of existing symmetric key encryption algorithms and develop a prototype lightweight symmetric key encryption algorithm. This could encrypt and decrypt small scale data on satisfactory efficiency level with low time and space complexity and high performance level.

1.5.2 Research Aim

The aim of this study is to recapitulate the development initiatives of symmetric key encryption and develop a LWE algorithm for small scale data (of format .doc, .docx , txt and equivalent) cryptanalysis of size 1 to 512KB for cross OS platform (i.e., Windows OS, Linux OS and Mac OS).

1.5.3 Research Objectives

Every research has some predefined clear objectives which it tries to achieve. The research objectives can be divided into following two categories:

General Objective: General objective of this research is to analyze the existing symmetric key encryption algorithms and develop a lightweight encryption (LWE) algorithm for small scale data cryptanalysis.

Specific Objective: The specific objectives of this research are-

- To study the different features of existing cryptography algorithms
- To design and develop a Light Weight Encryption (LWE) algorithm
- To reduce the time and space complexity of small scale data cryptanalysis with the newly developed LWE algorithm

1.5.4 Research Questions

- What are the important features of symmetric key encryption a LWE should address?

- How to design and implement a LWE algorithm for small scale data encryption with reduced time and space complexities?
- How to increase efficiency of the new LWE algorithm over the existing algorithms?

1.6 Delimitation of the Study

The study only deals with the development of a lightweight symmetric key cryptography algorithm and finding its efficiency and percentage of error. The algorithm neither discussed about large scale and asymmetric key algorithms and nor compared the new algorithm with these two kinds of algorithms. It also has not covered the detailed comparison of all the features of the new algorithm with the existing algorithms. It only deals with English characters and ignores other language characters.

1.7 Definition of the used terms

Cryptography: It is the practice and study of techniques for secure communication in the presence of third parties (called adversaries). More generally, it is about constructing and analyzing protocols that block adversaries; various aspects in information security such as data confidentiality, data integrity, authentication, and non-repudiation are central to modern cryptography [8] .

Cryptanalysis: Cryptanalysis (from the Greek krypts, "*hidden* ", and analein,"*to loosen* "or "*to untie* ") is the study of methods for obtaining the meaning of encrypted information, without access to the secret information that is normally required to do so. Typically, this involves knowing how the system works and finding a secret key. It is also used to refer to any attempt to circumvent the security of other types of cryptographic algorithms and protocols in general, and not just encryption [15].

Decryption: In cryptography, decryption is the process of decoding messages or information. It is the opposite of encryption [9].

Encryption: In cryptography, encryption is the process of encoding messages or information in such a way that only authorized parties can read it. Encryption does not of itself prevent interception, but denies the message content to the interceptor [9].

Lightweight Encryption (LWE) Algorithm: Lightweight encryption (LWE) algorithms are those algorithms which can work efficiently at a fast rate on smaller and powerful devices. Lightweight algorithms are designed by keeping in mind the restriction of small devices i.e., computational power, Memory, Storage space, Available energy while maintaining adequate performance. Lightweight does not imply less secure [11]. The goal is to have lightweight encryptions as secure as heavyweight encryptions. For this some Compromises may be needed. Some examples of LWE are- *PRESENT, Humming bird-2, HIGHT, DESL* etc [10].

Symmetric key algorithm: Symmetric-key algorithms are a class of algorithms for cryptography that use the same cryptographic keys for both encryption of plaintext and decryption of ciphertext. The keys may be identical or there may be a simple transformation to go between the two keys. The keys, in practice, represent a shared secret between two or more parties that can be used to maintain a private information link. This requirement that both parties have access to the secret key is one of the main drawbacks of symmetric key encryption, in comparison to public-key encryption [10].

1.8 Structure of the Dissertation

Chapter 1: Introduction- This chapter gives the outlook of the research subject and discusses the aims and objectives of the research and clarifies in what context the research originated and what significance does this research carry.

Chapter 2: Literature Review- The second chapter deals with Literature Review which is the systemic study of the existing works those has done on the research topic and gives the structure and theoretical framework of references. It also provides the current image of the Educational and Organizational Information Security industry and what impact lightweight sym-metric key algorithm has on the security system trends. The chapter also contains current scale, scope and diversity of the Light Wight and Symmetric Key Data Encryption. In short, it gives a brief and concise account of existing lightweight symmetric key encryption methods.

Chapter 3: Research Methodology- The third chapter is the Research Methodology which explains how the research was designed and what is the nature of it. It also provides the philosophy of the research and what strategies were implemented to do the research.

This study also has described each phase of the algorithm development and how data was collected and analyzed in this chapter. Finally, the limitations of the research are also highlighted here.

Chapter 4: Experimental Result, Analysis and Discussion- This chapter includes the result of the research and analyzes it. It calculates the File type vs. Average execution time. Then a quantitative analysis has been done and a comprehensive discussion has made on the analyzed data and outcomes.

Chapter 5: Conclusion and Recommendation- This chapter gives the concluding remark about the research and recommends what future work can be done on the topic. It contains some general and critical observations about the research and discusses how the present study can be improved.

CHAPTER 2
LITERATURE REVIEW

A literature review is a text of a scholarly paper, which includes the current knowledge including substantive findings, as well as theoretical and methodological contributions to a particular topic. Literature reviews use secondary sources, and do not report new or original experimental work [11]. This chapter is of extreme importance for the literature review has to be complete and not to take any sides. Also, this chapter paves light on various researches that are conducted on the topics of similar nature.

Literature review is crucial for doing a good research because generating and implementing a new idea is possible only when a thorough study of existing materials has been done. This chapter will highlight on studies done so far on symmetric key cryptography algorithms. It is a record of what we studied about our research subject.

2.1 Structure of the literature review

The main attribute of a good literature review is that it is well structured. At first, we created a mind map of the concepts that we were going to discuss in our review, including key words and synonyms. As we reviewed books and journals, we wrote down the topic words that we had selected. We created a mind map of all of the terms that applied to our topic before conducting a literature search. This mind map was then used to guide our literature search as well as making sure that we discussed pertinent concepts in the review itself. This mind map and its sections can also be the subsections that we used for storing the results of our research. First there is theoretical framework of references which contains the framework according to which the research is carried out. Within the theoretical framework, information security and encryption are defined. After that, different encryption and decryption techniques are discussed in details. Then the reader is informed about small scale data and the current structure of information security and various development initiatives taken in the field of lightweight symmetric key encryption. The current scale, scope and diversity of this algorithm are also discussed in details. Recent developments done on this topic are also discussed. The next important topic discussed in literature review is assessing the role of lightweight encryption in educational and organizational field. The literature review also discusses the dilemma faced by these above mentioned fields towards adoption of large scale

security system and the impact of the lightweight encryption on the current information security trend. Next topic is the critical assessment of the literature studied and knowledge gaps that have been identified while studying the existing literature about encryption algorithms. Then the original research questions and objectives are analyzed with respect to the theoretical frame of reference. Then a summary of what have been studied is given.

2.2 Theoretical framework of references

According to Nieswiadomy the word theory is derived from "theoria ", a Greek Word. Theoria means a beholding or speculation [12]. Theories are speculations. Theories are NEVER PROVED. Theories are used to describe, predict, explain, and control phenomena. Now we describe what a theoretical framework is. "*A theoretical framework is a frame of reference that is a basis for observations, definitions of concepts, research designs, interpretations, and generalizations, much as the frame that rests on a foundation defines the overall design of a house* " [13]. So by theoretical framework of research we mean a specific reference frame according to which the research is carried out. Every research has its own framework of references. A theoretical framework consists of concepts and, together with their definitions and reference to relevant scholarly literature, existing theory that is used for your particular study. The theoretical framework must demonstrate an understanding of theories and concepts that are relevant to the topic of your research paper and that relate to the broader areas of knowledge being considered. The theoretical framework is most often not something readily found within the literature. One must review course readings and pertinent research studies for theories and analytic models that are relevant to the research problem you are investigating. The selection of a theory should depend on its appropriateness, ease of application, and explanatory power.
The theoretical framework strengthens the study in the following ways:

- An explicit statement of theoretical assumptions permits the reader to evaluate them critically.
- The theoretical framework connects the researcher to existing knowledge. Guided by a relevant theory, you are given a basis for your hypotheses and choice of research methods.
- Articulating the theoretical assumptions of a research study forces you to address questions of why and how. It permits you to intellectually transition from simply describing a phenomenon you have observed to generalizing about various aspects of that phenomenon.

- Having a theory helps you identify the limits to those generalizations. A theoretical framework specifies which key variables influence a phenomenon of interest and highlights the need to examine how those key variables might differ and under what circumstances.

2.2.1 Information Security

Information security, sometimes shortened to InfoSec, is the practice of defending information from unauthorized access, use, disclosure, disruption, modification, perusal, inspection, recording or destruction. It is a general term that can be used regardless of the form the data may take (e.g., electronic, physical). Sometimes referred to as computer security, Information Technology security is information security applied to technology (most often some form of computer system). It is worthwhile to note that a computer does not necessarily mean a home desktop. A computer is any device with a processor and some memory. Such devices can range from non-networked standalone devices as simple as calculators, to networked mobile computing devices such as smartphones and tablet computers. IT security specialists are almost always found in any major enterprise/establishment due to the nature and value of the data within larger businesses. They are responsible for keeping all of the technology within the company secure from malicious cyber attacks that often attempt to breach into critical private information or gain control of the internal systems.

Information assurance is the act of ensuring that data is not lost when critical issues arise. These issues include but are not limited to: natural disasters, computer/server malfunction, physical theft, or any other instance where data has the potential of being lost. Since most information is stored on computers in our modern era, information assurance is typically dealt with by IT security specialists. One of the most common methods of providing information assurance is to have an off-site backup of the data in case one of the mentioned issues arises.

2.2.2 Encryption

In cryptography, encryption is the process of encoding messages or information in such a way that only authorized parties can read it. Encryption does not of itself prevent interception, but denies the message content to the interceptor. In an encryption scheme, the message or information, referred to as plaintext, is encrypted using an encryption algorithm, generating cipher text that can only be read if decrypted. For technical reasons, an encryption scheme usually uses a pseudo-random encryption key generated by an algorithm. It is in principle possible to decrypt the message without possessing the key, but, for a well-designed encryption scheme, large computational resources and skill are required. An

authorized recipient can easily decrypt the message with the key provided by the originator to recipients, but not to unauthorized interceptors.

2.2.3 Different Techniques of Information Encryption and Decryption

Different types of encryption and decryption techniques are described in the following paragraphs:
In cryptography, a **block cipher** is a deterministic algorithm operating on fixed-length groups of bits, called *blocks*, with an unvarying transformation that is specified by asymmetric key. Block ciphers are important elementary components in the design of many cryptographic protocols, and are widely used to implement encryption of bulk data. The modern design of block ciphers is based on the concept of an iterated product cipher. Product ciphers were suggested and analyzed according the publication *Communication Theory of Secrecy Systems* as a means to effectively improve security by combining simple operations such as substitutions and permutations [14] . A **stream cipher** is a symmetric key cipher where plaintext digits are combined with a pseudorandom cipher digit stream (keystream). In a stream cipher each plaintext digit is encrypted one at a time with the corresponding digit of the keystream, to give a digit of the ciphertext stream. An alternative name is a state cipher, as the encryption of each digit is dependent on the current state. In practice, a digit is typically a bit and the combining operation an exclusive-or (xor). **The Advanced Encryption Standard (AES)** is a specification for the encryption of electronic data established by the U.S.National Institute of Standards and Technology (NIST) in 2001. AES is based on the Rijndael cipher developed by two Belgian cryptographers, Joan Daemen and Vincent Rijmen, who submitted a proposal to NIST during the AES selection process. Rijndael is a family of ciphers with different key and block sizes. For AES, NIST selected three members of the Rijndael family, each with a block size of 128 bits, but three different key lengths: 128, 192 and 256 bits.

The Data Encryption Standard (DES) was once a predominant symmetric-key algorithm for the encryption of electronic data. It was highly influential in the advancement of modern cryptography in the academic world. Developed in the early 1970s at IBM and based on an earlier design by Horst Feistel, the algorithm was submitted to the National Bureau of Standards (NBS) following the agency's invitation to propose a candidate for the protection of sensitive, unclassified electronic government data. In 1976, after consultation with the National Security Agency (NSA), the NBS eventually selected a slightly modified version (strengthened against differential cryptanalysis, but weakened against brute force attacks), which was published as an official Federal Information Processing Standard (FIPS) for the United States in 1977. In cryptography, **Twofish** is a symmetric key block cipher with a

block size of 128 bits and key sizes up to 256 bits. It was one of the five finalists of the Advanced Encryption Standard contest, but it was not selected for standardization. Twofish is related to the earlier block cipher Blowfish. Twofish's distinctive features are the use of pre-computed key-dependent S-boxes, and a relatively complex key schedule. One half of an n-bit key is used as the actual encryption key and the other half of the n-bit key is used to modify the encryption algorithm (key-dependent S-boxes). Twofish borrows some elements from other designs; for example, thepseudo-Hadamard transform (PHT) from the SAFER family of ciphers. Twofish has a Feistel structure like DES. Blowfish is a symmetric-key block cipher, designed in 1993 by Bruce Schneier and included in a large number of cipher suites and encryption products. Blowfish provides a good encryption rate in software and no effective cryptanalysis of it has been found to date. However, the Advanced Encryption Standard (AES) now receives more attention.

Serpent is a symmetric key block cipher that was a finalist in the Advanced Encryption Standard (AES) contest, where it was ranked second to Rijndael. Serpent was designed by Ross Anderson, Eli Biham, and Lars Knudsen. Like other AES submissions, Serpent has a block size of 128 bits and supports a key size of 128, 192 or 256 bits. The cipher is a 32-round substitution-permutation network operating on a block of four 32-bit words. Each round applies one of eight 4-bit to 4-bit S-boxes 32 times in parallel. Serpent was designed so that all operations can be executed in parallel, using 32 1-bit slices. This maximizes parallelism, but also allows use of the extensive cryptanalysis work performed on DES. In cryptography, **CAST-128** (alternatively **CAST5**) is a symmetric-key block cipher used in a number of products, notably as the default cipher in some versions of GPG and PGP. It has also been approved for Canadian government use by the Communications Security Establishment. The algorithm was created in 1996 by Carlisle Adams and Stafford Tavares using the CAST design procedure. Another member of the CAST family of ciphers, CAST-256 (a former AES candidate) was derived from CAST-128. According to some sources, the CAST name is based on the initials of its inventors, though Bruce Schneier reports the authors' claim that "*the name should conjure up images of randomness* " [15].

In cryptography, RC4 (also known as ARC4 or ARCFOUR meaning Alleged RC4, see below) is the most widely used software stream cipher and is used in popular protocols such as Transport Layer Security (TLS) (to protect Internet traffic) and WEP (to secure wireless networks). While remarkable for its simplicity and speed in software, RC4 has weaknesses that argue against its use in new systems. It is especially vulnerable when the beginning of the output keystream is not discarded, or when nonrandom or related keys are used; some ways of using RC4 can lead to very insecure cryptosystems such as WEP. In cryptography, **Triple DES (3DES)** is the common name for the **Triple Data Encryption Algorithm (TDEA**

or Triple DEA) symmetric-key block cipher, which applies the Data Encryption Standard (DES) cipher algorithm three times to each data block. The original DES cipher's key size of 56 bits was generally sufficient when that algorithm was designed, but the availability of increasing computational power made brute-force attacks feasible. Triple DES provides a relatively simple method of increasing the key size of DES to protect against such attacks, without the need to design a completely new block cipher algorithm. **Enveloped Public Key Encryption (EPKE)** is the method of applying public-key cryptography and ensuring that an electronic communication is transmitted confidentially, has the contents of the communication protected against being modified (communication integrity) and cannot be denied from having been sent (non-repudiation). This is often the method used when securing communication on an open networked environment such by making use of the Transport Layer Security (TLS) or Secure Sockets Layer (SSL) protocols. EPKE consists of a two-stage process that includes both Forward Public Key Encryption (FPKE) also known as symmetric encryption and Inverse Public Key Encryption (IPKE). Both Forward Public Key Encryption and Inverse Public Key encryption make up the foundation of Enveloped Public Key Encryption.

DiffieHellman key exchange (D-H) is a specific method of securely exchanging cryptographic keys over a public channel and was the first specific example of public-key cryptography as originally conceptualized by Ralph Merkle. DH is one of the earliest practical examples of public key exchange implemented within the field of cryptography. The DiffieHellman key exchange method allows two parties that have no prior knowledge of each other to jointly establish a shared secret key over an insecure communications channel. This key can then be used to encrypt subsequent communications using a symmetric key cipher. The scheme was first published by Whitfield Diffie and Martin Hellman in 1976. By 1975, James H. Ellis, Clifford Cocks and Malcolm J. Williamson within GCHQ, the British signals intelligence agency, had also shown how public-key cryptography could be achieved; although, their work was kept secret until 1997. **RSA** is one of the first practicable public-key cryptosystems and is widely used for secure data transmission. In such a cryptosystem, the encryption key is public and differs from the decryption key which is kept secret. In RSA, this asymmetry is based on the practical difficulty of factoring the product of two large prime numbers, the factoring problem. RSA stands for Ron Rivest, Adi Shamir and Leonard Adleman, who first publicly described the algorithm in 1977. Clifford Cocks, an English mathematician, had developed an equivalent system in 1973, but it was not declassified until 1997. **The Digital Signature Algorithm (DSA)** is a Federal Information Processing Standard for digital signatures. It was proposed by the National Institute of Standards and Technology (NIST) in August 1991 for use in their Digital Signature Standard (DSS) and adopted as FIPS 186 in 1993. Four revisions to the initial specification have

been released: FIPS 186-1 in 1996, FIPS 186-2 in 2000, FIPS 186-3 in 2009, and FIPS 186-4 in 2013. DSA is covered by U.S. Patent 5,231,668, filed July 26, 1991 and attributed to David W. Kravitz, a former NSA employee. This patent was given to "The United States of America as represented by the Secretary of Commerce, Washington, D.C.", and NIST has made this patent available worldwide royalty-free. Claus P. Schnorr claims that his U.S. Patent 4,995,082 (expired) covered DSA; this claim is disputed. DSA is a variant of the ElGamal Signature Scheme.

2.2.4 Small Scale data

Small scale data are those data which are small in size and collected from a small range of population or events. It can be stated more easily and conveniently as the amount of data one can conveniently store and process on a single machine, and in particular, a high-end laptop or server. The interesting and new thing right now is the democratization of data and the associated possibility of large-scale distributed community of data wranglers working collaboratively. A key point is that the dramatic advances in computing, storage and bandwidth have far bigger implications for "*small data*"than for "*big data*". The recent advances have increased the realm of small data, the kind of data that an individual can handle on their own hardware, far more relatively than they have increased the realm of big data. Suddenly working with significant datasets that is datasets containing tens of thousands, hundreds of thousands or millions of rows can be a mass-participation activity. Small and big are relative terms that change as technology advances for example, in 1994 a terabyte of storage cost several hundred thousand dollars, today its under a hundred. This also means todays big is tomorrows small.

2.2.5 Current structure of the Information Security

Information security has become a vital entity to most organizations today due to current trends in information transfer through a borderless and vulnerable world. The concern and interest in information security is mainly due to the fact that information security risk assessment (ISRA) is a vital method to not only to identify and prioritize information assets but also to identify and monitor the specific threats that an organization induces; especially the chances of these threats occurring and their impact on the respective businesses. However, organizations wanting to conduct risk assessment may face problems in selecting suitable methods that would augur well in meeting their needs. This is due to the existence of numerous methodologies that are readily available. However, there is a lack in agreed reference benchmarking as well as in the comparative framework for evaluating these ISRA methods to access the information security risk.

Generally, organizations will choose the most appropriate ISRA method by carrying out a comparative study between the available methodologies in detail before a suitable method is selected to conduct the risk assessment. This paper suggests a conceptual framework of info-structure for ISRA that was developed by comparing and analyzing six methodologies which are currently available. The info-structure for ISRA aims to assist organizations in getting a general view of ISRA flow, gathering information on the requirements to be met before risk assessment can be conducted successfully. This info-structure can be conveniently used by organizations to complete all the required planning as well as the selection of suitable methods to complete the ISRA.

2.2.6 Development Initiatives of Light Weight Symmetric Key Encryption

Recent advancement of communication and computing technologies introduces different types of portable devices that populate in our day to day life. These devices have limited battery power, restricted storage and low computation power to bring the device in affordable cost and portable size. Information security is of primary concern for all users irrespective of the computing device being used. Among the different approaches for achieving information security, the present work concerns with symmetric key cryptography. Varieties of encryption algorithms are available to encrypt the data. Execution of the traditional encryption algorithms consumes time, space and energy. Moreover, side channel attacks are based on time and power that can be applied to the block ciphers implemented on smart card technology. Also protecting implementation against these kinds of attacks is usually difficult and expensive. Application of cryptographically strong algorithm such as AES-Rijndael leads to significant transmission delay, and require high computations as well as large storage capacity. It leads to infeasible to incorporate the strong cryptographic algorithms in the resource constrained devices [16]. For these reasons, researchers are becoming more interested to lightweight cryptography day by day. Many development initiatives have been taken on lightweight symmetric key algorithms to make security process simpler, more integrated and less time and resource consuming. Lightweight cryptographies have been implemented successfully in several fields to ensure security. For example- Kumar & Aggarwal have developed a lightweight cryptography solution for resource constraint mobile ad-hoc networks (MANET), Barbero & Ytrehus have developed lightweight cryptography for RFID devices [17, 18]. Lightweight cryptography scheme has also been used in neural network [19].

On the other hand, encryption algorithm ICEBERG proposed for its implementation with special emphasis to the reconfiguration hardware. However, the software implementation is not suitable i.e., not cost effective with respect to storage requirement and/ or speed. Data dependent permutation (DDP)-based fast encryption algorithms are appeared to be faster and

efficient for high speed networks. Recently, Cobra-H64 and Cobra-H128 were proposed in use switchable operations to prevent the weak keys identified against the earlier DDP-based encryption techniques. These ciphers are specially emphasized for high speed performance hardware implementation but require more hardware resources [20].

2.2.7 Current scale, scope and diversity of the Light Wight and Symmetric Key Data Encryption

Database encryption has the potential to secure data at rest by providing data encryption, especially for sensitive data, avoiding the risks such as misuse of the data. In order to achieve a high level of security, the complexity of encryption algorithms should be increased with minimal damage to database efficiency, ensuring performance is not affected. Presently, lightweight and Symmetric Key Data Encryption is very popular. It has been proved by several researches that lightweight encryption can considerably prevent data loss and hacking and can provide data integration and security. So a significant number of researches are done on the subject. Current scale of lightweight cryptography is not very large because it is not used in large industries but its importance is increasing day by day.
Symmetric key encryptions are very common and useful for personal users and small companies. So the scale, scope and diversity of these researches are increasing day by day. Many users now prefer to have a lightweight encryption system which will be less time consuming and more efficient for their data security and integrity.

2.2.8 Recent development of Symmetric key encryption affecting Data Security Architecture

The tremendous development of technology and data storage leads organizations to depend on database systems. Organizations store huge amounts of data in secured databases in order to retrieve them in a fast and secure way. Some of the stored data is considered sensitive and has to be protected. In the presence of security threats, database security is becoming one of the most urgent challenges because much damage to data can happen if it suffers from attacks and unauthorized access. With databases in complex, multi-tiered applications, attackers may reach the information inside the database. Damage and misuse of sensitive data that is stored in a database does not only affect a single user; but possibly an entire organization.

So now the current trend is to create a simple and single-tiered architecture which supports symmetric key encryption. This saves much time and energy and is helpful for the users. "*Simple architecture for better performance* "is the motto of the present age .

2.2.9 Major Focus of Light Weight Symmetric Key Encryption

According to Mike Vegara, director of product management at RSA, "*Symmetric key encryption is always faster than asymmetric, so what you do is encrypt a piece of data using a symmetric key and then encrypts the key using the RSA algorithm* " [21]. As symmetric key encryptions are faster by nature, lightweight symmetric key encryption makes them more time saving and simple. So the major focus of Light Weight Symmetric Key Encryption is to make the encryption process faster and easy to understand. Its target is to create an encryption system specialized for small scale data analysis.

2.2.10 Propositional structure of Light Weight Encryption on Small Scale data

Proposition is the act of offering, suggesting something to be accepted or done. It is also defined as a scheme or plan to be proposed. So propositional structure is the structure in which an offer is presented. Symmetric key encryption has same key for both encryption and decryption which makes it faster to encrypt as there is no requirement of processing two keys. It is mostly used for normal user level encryptions. Cryptographic technologies are advancing: new techniques on attack, design and implementation are extensively studied. One of the state-of-the-art techniques is "*Lightweight Cryptography (LWC)* ". Lightweight cryptography is a cryptographic algorithm or protocol tailored for implementation in constrained environments including RFID tags, sensors, contactless smart cards, health-care devices and so on. The properties of lightweight cryptography have already been discussed in ISO/IEC 29192 in ISO/IEC JTC 1/SC 27. ISO/IEC 29192 is a new standardization project of lightweight cryptography, and the project is in process of standardization [22] .

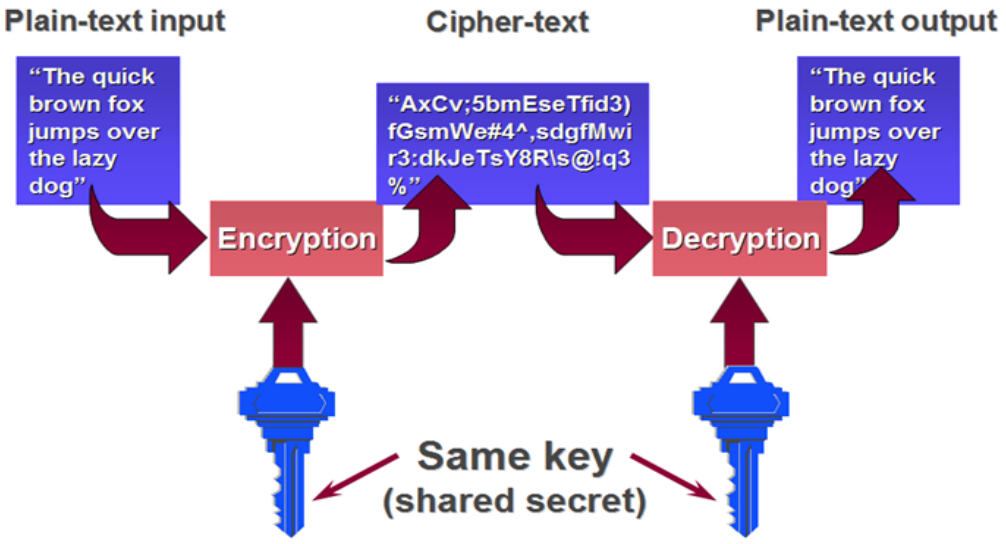

Figure 2.1: Symmetric Key Encryption

In ISO/IEC 29192, lightweight properties are described based on target platforms. In hardware implementations, chip size and/or energy consumption are the important measures to evaluate the lightweight properties. In software implementations, the smaller code and/or RAM size are preferable for the lightweight applications. From the view of the implementation properties, the lightweight primitives are superior to conventional cryptographic ones, which are currently used in the Internet security protocols, e.g., IPsec, TLS. Lightweight cryptography also delivers adequate security. Lightweight cryptography does not always exploit the security-efficiency trade-offs. We report recent technologies of lightweight cryptographic primitives.

2.2.11 Current image of the Educational and Organizational Information Security industry

Currently educational and organizational information security has become a tremendous industry. Education sector needs heavy information security because many transactions and researches are done in internet. If proper security is not assured, valuable research papers may be moved or modified from databases which can do lifelong damage to the person involved. Almost all organizations have confidential reports which have to be kept secret from outsiders and insiders also. Cryptography and encryption play a great role in doing so. Now-a-days, Educational and Organization sectors use heavyweight cryptography to guard their data from being misused, modified or accesses without proper authentication. They spend huge amount of money, time and human resources on this subject. This has both positive and negative aspects. It ensures security but costs much and requires huge resources.

2.2.12 Assess the role of Light Weight Encryption for securing organizational and educational data

The upcoming era of pervasive computing will be characterized by many smart devices thatbecause of the tight cost constraints inherent in mass deploymentshave very limited resources in terms of memory, computing power, and battery supply. Here, its necessary to interpret Moores law differently: Rather than a doubling of performance, we see a halving of the price for constant computing power every 18 months. Because many foreseen applications have extremely tight cost constraintsfor example, RFID in tetra packsover time, Moores law will increasingly enable such applications. Many applications will process sensitive health-monitoring or biometric data, so the demand for cryptographic components that can be efficiently implemented is strong and growing. For such implementations, as well as for ciphers that are particularly suited for this purpose, we have used the generic

term lightweight cryptography in this study. Every designer of lightweight cryptography must cope with the trade-offs between security, cost, and performance. Its generally easy to optimize any two of the three design goalssecurity and cost, security and performance, or cost and performance; however, it is very difficult to optimize all three design goals at once. For example, a secure and high-performance hardware implementation can be achieved by a pipelined, side-channel-resistant architecture, resulting in a high area requirement, and thus high costs. On the other hand, its possible to design a secure, low-cost hardware implementation with the drawback of limited performance. If the above design conditions are fulfilled, Lightweight encryption can play a great role in securing organizational and educational data. It can save much resources and time as it is faster and simpler [23].

2.2.13 Economic crisis of Education and Organizational Sector towards adoption of Large Scale information security architectures

Despite sophisticated monitoring tools for runtime detection of intruders and techniques designed to protect computing systems from a wide range of attacks, attackers continually penetrate even well-protected systems. Attack data from real, large-scale production environments (National Center for Supercomputing Applications (NCSA) at Illinois, in this work) are used as a basis for characterizing and modeling attacker behavior and for uncovering deficiencies of the monitoring infrastructure. Increased understanding of attacks arising from these analyses and modeling activities significantly contributes to improvements in secure systems analysis and design. The analyses uncover new and realistic attack scenarios that can guide the design of enhancements to improve system protection against malicious activities at every level. Understanding real attack patterns and classes through detailed forensics pinpoints the open holes in a network/system and characterizes attacker behavior. In-depth study of the data allows investigating actions and intentions of the attacker, and creates a foundation for the design of an automated tool to assist in data collection, analysis, and response. The size and variety of the data enable a flexible framework to be developed that can incorporate insights gained from attacks yet unseen. So it is seen that even large scale information security architectures can be breached and additionally they cost much money and time. Thus companies face a dilemma between large scale information security systems and lightweight encryption system as the latter needs less resources and money.

2.2.14 Impact analysis of Light Weight Symmetric key for the predicted Information Security Trends and Developments in Education and Organizational Sector

Light Weight Symmetric key has great impact on Information Security Trends and Developments. Large scale security systems are not giving required throughput and performance

as compared to the money and human resources invested to it. If performance, cost and efficiency-these three factors can be combined optimally in a lightweight symmetric key encryption system, then there is no need of a large scale security system. This can change the trend of current security industry. Individual researchers and companies are now more interested in lightweight encryption system. Many researches are currently being done on the subject and new and promising outcomes are being produced. In short, Lightweight symmetric key encryption is setting the new trend of security industry.

2.3 Critical Assessment

As the scale of the data used and time and space complexities of lightweight encryption are small, it cant be used for large levels of data. But surely it can fasten the secure information transaction process in the educational and organizational sector. Here the main challenge is increasing the efficiency and lowering the error percentage. So we recommend that researchers focus on this topic and try to make that optimum balance among cost, efficiency and performance.

2.4 Identified Knowledge Gaps

So we see that work done on lightweight and symmetric key cryptography has still many things to achieve. Researches done on this subject are not adequate. The number of lightweight symmetric key encryptions is not much and they are not efficient enough and performance level is not also up to the expected level. So we targeted to build an encryption system which will work on small scale data but will be considerably efficient and rate of error will be minimal. We hope that our research will be a significant contribution towards the shift from large scale encryption to small scale symmetric key encryption.

2.5 Analyzing the original research questions and research objectives with respect of Theoretical framework of references

To achieve the research objectives, we have chosen a specific theoretical framework of reference which will build the concept, definitions and relevance of our study. Our research objectives were to study the different features of existing cryptography algorithms, to design and develop a new algorithm, to reduce the time and space complexity of small scale data with the newly developed algorithm and to minimize the percentage of errors in LWE process. So we have chosen deductive approach by which first we will build a hypothesis

on the basis of existing theories and then test the hypothesis on data and then analyze the results.

2.6 Summary

If we summarize the whole literature review, there are three points to observe. First is, there are many types of encryption systems but symmetric ones are faster and mostly used. Second is, though security is the topmost priority for every individual and organization now-a-days, lightweight encryption system which is very suitable and reliable for small scale use is not given that much importance. People spend huge money and resources on large scale security systems which dont give expected output and are still vulnerable whereas many researchers now-a-days say that lightweight symmetric key encryption can do a better job at a low cost. Third is lightweight encryption systems are going to be the new trend of the future and most probably they will replace the large scale security systems.

CHAPTER 3
RESEARCH METHODOLOGY

Research in common parlance refers to a search for knowledge. One can also define research as a scientific and systematic search for pertinent information on a specific topic. In fact, research is an art of scientific investigation. Dictionary definition of research is a careful investigation or inquiry especially through search for new facts in any branch of knowledge. Methodology is the systematic, theoretical analysis of the methods applied to a field of study. It comprises the theoretical analysis of the body of methods and principles associated with a branch of knowledge. Typically, it encompasses concepts such as paradigm, theoretical model, phases and quantitative or qualitative techniques [24]. A **methodology** does not set out to provide solutions - it is, therefore, not the same thing as a method. Instead, it offers the theoretical underpinning for understanding which method, set of methods or so called "best practices "can be applied to specific case, for example, to calculate a specific result.

3.1 Research Design

A **research design** is a systematic plan to study a scientific problem. The design of a study defines the study type (descriptive, correlation, semi-experimental, experimental, review, meta-analytic) and sub-type (e.g., descriptive-longitudinal case study), research question, hypotheses, independent and dependent variables, experimental design, and, if applicable, data collection methods and a statistical analysis plan. Research design is the framework that has been created to seek answers to research questions. Our research design is experimental and exploratory in nature. The word **experimental research** has a range of definitions. In the strict sense, experimental research is what we call a true experiment. This is an experiment where the researchers manipulates one variable, and control/randomizes the rest of the variables. It has a control group, the subjects have been randomly assigned between the groups, and the researcher only tests one effect at a time. It is also important to know what variable(s) you want to test and measure. A very wide definition of experimental research, or a quasi experiment, is research where the scientist actively influences something to observe the consequences. Most experiments tend to fall in between the strict and the wide definition. Exploratory research is research conducted for a problem that has not been clearly defined. It often occurs before we know enough to make conceptual distinctions or

posit an explanatory relationship.**Exploratory research** helps determine the best research design, data collection method and selection of subjects. It should draw definitive conclusions only with extreme caution. Given its fundamental nature, exploratory research often concludes that a perceived problem does not actually exist. Exploratory research often relies on secondary research such as reviewing available literature and/or data, or qualitative approaches such as informal discussions with consumers, employees, management or competitors, and more formal approaches through in-depth interviews, focus groups, projective methods, case studies or pilot studies. As our research requires much experiment to come to a definite conclusion and explores new ideas about lightweight symmetric key encryption, so we have chosen experimental and exploratory design method for our research.

3.2 Research Paradigms

Webster Dictionary defines paradigm as "*an example or pattern: small, self-contained, simplified examples that we use to illustrate procedures, processes, and theoretical points* ". The most quoted definition of paradigm is "*paradigm as the underlying assumptions and intellectual structure upon which research and development in a field of inquiry is based* " [25]. The other definitions in the research literature include "*A paradigm is a world view, a general perspective, a way of breaking down the complexity of the real world*" [26]. Available research paradigm names are qualitative, quantitative, scientific, inductive and deductive. Our research paradigm is deductive in nature. First we give the definition of deductive research and then explain why our research is deductive. A very famous definition is "*A deductive approach is concerned with developing a hypothesis (or hypotheses) based on existing theory, and then designing a research strategy to test the hypothesis* " [27] . In our research, first we have studied the theory of encryption and then taken a hypothesis to build a lightweight symmetric key encryption which will operate efficiently and accurately on small scale data on all platforms. Then we have tested the hypothesis on data, analyzed it and finally have come to conclusion. The sequences can be understood better with the help of a diagram:

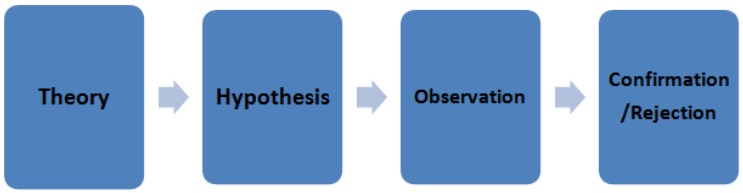

Figure 3.1: Deductive Research Approach

3.3 Research Strategy

Research strategy is a methodology that helps the researcher to investigate the research issue. Research strategy is a general plan that helps researcher in answering the research questions in a systematic way. An effective research strategy contains the clear objectives, research questions, data collection resources and various constraints that affects the research in different ways such as access limitations, time limitations, location and money limitations, ethical issue constraints etc. An effective research strategy helps the researcher to define that why researcher employing a particular research strategy to conduct the research study in an effective manner. Research strategy is also helpful for the researcher to use specific data collection methods to support the arguments [28]. In an effective research strategy, researcher collects the background information and analyzes the data to reach a specific conclusion. Some important research strategy includes the analysis of literature review, case study analysis, interview, observation, experiments, survey etc. In order to accomplish the research aims and objectives, the researcher used survey and analysis of academic articles strategies. Survey strategy helps the researcher to collect qualitative data and information [29]. With the help of survey strategy, researcher is able to collect general views of people that are related to the topic of role of marketing strategies to development of UK tourism industry. Researcher also uses academic journal articles analysis in order to achieve research aims and objectives. It helps researcher to collect relevant information and data that is related to the research issues. Both of these research strategies are helpful for the researcher to collect the valid and reliable data and information to achieve research aims and objectives [28]. We have divided the research work into three main divisions which are discussed in the next three subtopics.

3.3.1 Random Key Assignment

First, a four digit random key like abcd, efgh, ijkl etc is taken. Four digit key is chosen because it is neither too short nor too long. Too short key length would be easy to crack and too long key length will lower the speed of the algorithm. Another reason for choosing four letter key is that most PIN numbers, security codes are four digits. The key is changed during every iteration and new key is generated. This makes the algorithm hard to crack. The number of iterations of the algorithm is equal to the number of letters in the sentence including space and special characters.

3.3.2 Key Encryption and New Key generation

First the ASCII values of the key letters are calculated and then summed up. Then the mod of the sum by 100 is calculated. The result is converted into 8-bit binary number system and divided into four blocks that is each block contains two bits. The two bits of each block are concatenated with three binary bits ranging from 000 to 111 from both directions i.e. forward and backward. In first iteration, the concatenation is done with 000, in the second iteration it is done with 001 and so on. After eight iterations, it again comes back to 000. So now each block contains eight digits. The decimal values of these new blocks are found out and these are the ASCII values of new key. Thus new key is generated from previous one. This procedure continues up to the last iteration.

3.3.3 Small Scale Data Encryption and Decryption

In the encryption part, first the ASCII value of the plaintext is found out. Then, its corresponding binary value is calculated. Then, we take a random number 298 which is fixed for this algorithm. The mod of 298 is done by the value which was found when the sum of the ASCII values of the key letters was mod by 100. The result is then converted into binary and then XOR-ed with the binary value of the ASCII value of the plaintext. The result of the XOR is again converted to decimal and this decimal value is the ASCII value of the ciphertext. The character corresponding to that ASCII value is found out and this is the ciphertext of the plaintext. Thus encryption is done. Similarly, in the decryption part, first the ASCII

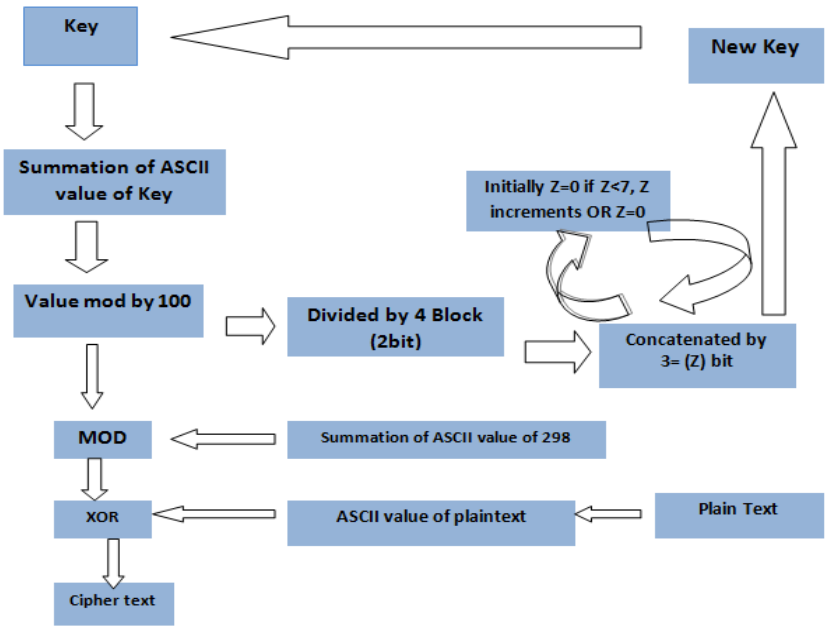

Figure 3.2: Encryption Flowchart

value of the ciphertext is found out and then its corresponding binary value is found out. Again, the mod of random number 298 is done by the value which was found when the sum of the ASCII values of the key letters was mod by 100. This value was transmitted to the decryption part. This new result is converted into binary and XOR-ed with the binary value of the ASCII value of the ciphertext. The result of the XOR is again converted to decimal and this decimal value is the ASCII value of the plaintext. The character corresponding to that ASCII value is found out and this is the plaintext of the ciphertext. In this way, decryption is done. To make the whole process more clear and visible to the reader, an example is taken.

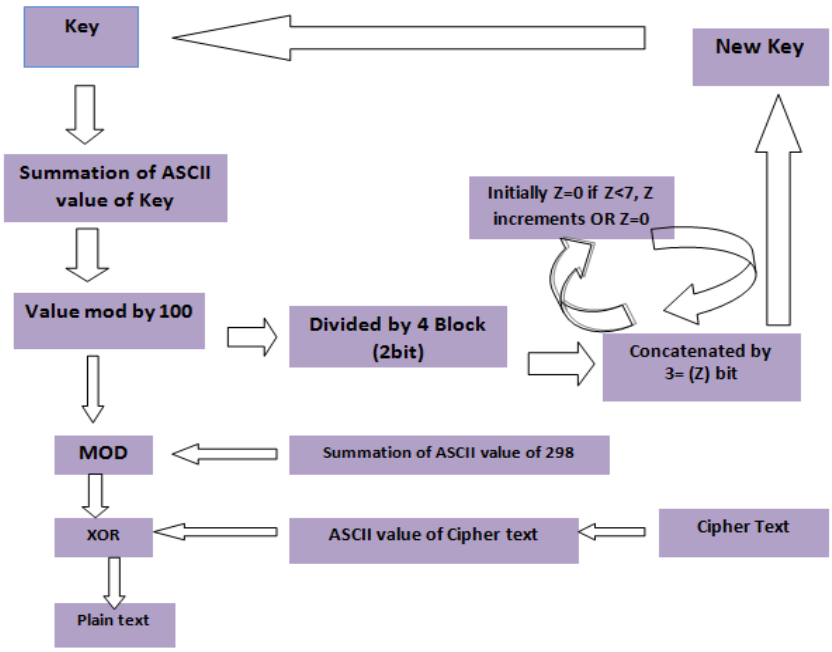

Figure 3.3: Decryption Flowchart

The sentence “MY MIST ”is taken as the plaintext. As it is a symmetric key algorithm, the key for encryption and decryption will be same. The string “abcd ”is taken as the random key. The letter by letter encryption and decryption of the sentence is shown below:

Encryption part:

Plain text: MY MIST
Random key: abcd
1st iteration:
Key generation:

Key:	a	b	c	d
ASCII Value:	97	98	99	100

Summation of ASCII Value: 394
Mod by 100(m1): 94
Binary Value: 01011110
Divided By 4 Block: 01 01 11 10
Concatenated By 000: 000 01 000 000 01 000 000 11 000 000 10 000
ASCII value of New Key: 8 8 24 16

Encryption:

Plain Text: M
ASCII Value: 77
Binary Value: 01001101
Random Number: 298
Mod of R.N. by m1: 298 % 94 =16= $(10000)_2$
XOR with Plain Text: 77 XOR 16= $(01011101)_2$
ASCII Value of Cipher Text: 93
Cipher Text:]

2nd iteration:

Key generation:

ASCII Value: 8 8 24 16
Summation of ASCII Value: 56
Mod by 100(m2): 56
Binary Value: 00111000
Divided By 4 Block: 00 11 10 00
Concatenated By 001: 001 00 001 001 11 001 001 10 001 001 00 001
ASCII value of New Key: 33 57 49 33

Encryption:

Plain Text: Y
ASCII Value: 89
Binary Value: 1011001
Random Number: 298
Mod of R.N. By m2: 298 % 56 =18= $(10010)_2$
XOR with Plain Text: 89 XOR 18= $(1001011)_2$
ASCII Value of Cipher Text: 75
Cipher Text: K

3rd iteration:

Key generation:

ASCII Value: 33 57 49 33

Summation of ASCII Value: 172

Mod by 100(m2): 72

Binary Value: 01001000

Divided By 4 Block: 01 00 10 00

Concatenated By 010: 010 01 010 010 00 010 010 10 010 010 00 010

ASCII value of New Key: 74 66 82 66

Encryption:

Plain Text: Space

ASCII Value: 32

Binary Value: 100000

Random Number: 298

Mod of R.N. By m3: 298 % 72 =10= $(1010)_2$

XOR with Plain Text: 32 XOR 10= $(101010)_2$

ASCII Value of Cipher Text: 42

Cipher Text: *

4th iteration:

Key generation:

ASCII Value: 74 66 82 66

Summation of ASCII Value: 288

Mod by 100(m2): 88

Binary Value: 01011000

Divided By 4 Block: 01 01 10 00

Concatenated By 011: 011 01 011 011 01 011 011 10 011 011 00 011

ASCII value of New Key: 107 107 115 99

Encryption:

Plain Text: M

ASCII Value: 77

Binary Value: 01001101

Random Number: 298

Mod of R.N. by m4: 298 % 88 =34 = $(100010)_2$

XOR with Plain Text: 77 XOR 34= $(1101111)_2$

ASCII Value of Cipher Text: 111

Cipher Text: o

5th iteration:

Key generation:

ASCII Value: 107 107 115 99

Summation of ASCII Value: 428

Mod by 100(m5): 28

Binary Value: 00011100

Divided By 4 Block: 00 01 11 00

Concatenated By 100: 100 00 100 100 01 100 100 11 100 100 00 100

ASCII value of New Key: 132 140 156 132

Encryption:

Plain Text: I

ASCII Value: 73

Binary Value: 1001001

Random Number: 298

Mod of R.N. By m5: 298 % 28 =18= $(10010)_2$

XOR with Plain Text: 73 XOR 18= $(1011011)_2$

ASCII Value of Cipher Text: 91

Cipher Text: [

6th iteration:

Key generation:

ASCII Value: 132 140 156 132

Summation of ASCII Value: 560

Mod by 100(m6): 60

Binary Value: 00111100

Divided By 4 Block: 00 11 11 00

Concatenated By 101: 101 00 101 101 11 101 101 11 101 101 00 101

ASCII value of New Key: 165 189 189 165

Encryption:

Plain Text: S

ASCII Value: 83

Binary Value: 1010011

Random Number: 298

Mod of R.N. by Mod of Key: 298 % 60 =58 = $(111010)_2$

XOR with Plain Text: 83 XOR 58= $(1101001)_2$

ASCII Value of Cipher Text: 105
Cipher Text: i

7th iteration:
Key generation:
ASCII Value: 165 189 189 165
Summation of ASCII Value: 708
Mod by 100(m7): 8
Binary Value: 00001000
Divided By 4 Block: 00 00 10 00
Concatenated By 110: 110 00 110 110 00 110 110 10 110 110 00 110
ASCII value of New Key: 198 198 214 198
Encryption:
Plain Text: T
ASCII Value: 84
Binary Value: 1010100
Random Number: 298
Mod of R.N. By m7: 298 % 8 =2 = $(10)_2$
XOR with Plain Text: 84 XOR 2= $(1010110)_2$
ASCII Value of Cipher Text: 86
Cipher Text: V

Ciphertext:]K*o[iV

Decryption part:

Ciphertext:]K*o[iV
Key: abcd
1st iteration:
Key generation:

Key:	a	b	c	d
ASCII Value:	97	98	99	100
Summation of ASCII Value:	394			
Mod by 100(n1):	94			
Binary Value:	01011110			
Divided By 4 Block:	01	01	11	10
Concatenated By 000:	000 01 000	000 01 000	000 11 000	000 10 000
ASCII value of New Key:	8	8	24	16

Decryption:
Ciphertext:]
ASCII Value: 93
Binary Value: 01011101
Random Number: 298
Mod of R.N. By n1: 298 % 94 =16= $(10000)_2$
XOR With ciphertext: 93 XOR 16= $(01001101)_2$
ASCII Value of plaintext: 77
Plaintext: M

2nd iteration:
Key generation:

ASCII Value:	8	8	24	16
Summation of ASCII Value:	56			
Mod by 100(n2):	56			
Binary Value:	00111000			
Divided By 4 Block:	00	11	10	00
Concatenated By 001:	001 00 001	001 11 001	001 10 001	001 00 001
ASCII value of New Key:	33	57	49	33

Decryption:
Ciphertext: K

ASCII Value: 75
Binary Value: 1001011
Random Number: 298
Mod of R.N. by n2: 298 % 56 =18= $(10010)_2$
XOR with ciphertext: 75 XOR 18= $(1011001)_2$
ASCII Value of plaintext: 89
Plaintext: Y

3rd iteration:
Key generation:

ASCII Value:	33	57	49	33
Summation of ASCII Value:	172			
Mod by 100(n3):	72			
Binary Value:	01001000			
Divided By 4 Block:	01	00	10	00
Concatenated By 010:	010 01 010	010 00 010	010 10 010	010 00 010
ASCII value of New Key:	74	66	82	66

Decryption:
Ciphertext: *
ASCII Value: 42
Binary Value: 101010
Random Number: 298
Mod of R.N. By n3: 298 % 72 =10 $(1010)_2$
XOR with ciphertext: 42 XOR 10= $(100000)_2$
ASCII Value of plaintext: 32
Plaintext: Space

4th iteration:

Key generation:

ASCII Value: 74 66 82 66

Summation of ASCII Value: 288

Mod by 100(n4): 88

Binary Value: 01011000

Divided By 4 Block: 01 01 10 00

Concatenated By 011: 011 01 011 011 01 011 011 10 011 011 00 011

ASCII value of New Key: 107 107 115 99

Decryption:

Ciphertext: o

ASCII Value: 111

Binary Value: 1101111

Random Number: 298

Mod of R.N. by n4: 298 % 88 =34= $(100010)_2$

XOR with ciphertext: 111 XOR 34= $(01001101)_2$

ASCII Value of plaintext: 77

Plaintext: M

5th iteration:

Key generation:

ASCII Value: 107 107 115 99

Summation of ASCII Value: 428

Mod by 100(n5): 28

Binary Value: 00011100

Divided By 4 Block: 00 01 11 00

Concatenated By 100: 100 00 100 100 01 100 100 11 100 100 00 100

ASCII value of New Key: 132 140 156 132

Decryption:

Ciphertext: [

ASCII Value: 91

Binary Value: 1011011

Random Number: 298

Mod of R.N. by n5: 298 % 28 =18= $(10010)_2$

XOR with ciphertext: 91 XOR 18= $(1001001)_2$

ASCII Value of plaintext: 73

Plaintext: I

6th iteration:

Key generation:

ASCII Value: 132 140 156 132

Summation of ASCII Value: 560

Mod by 100(n6): 60

Binary Value: 00111100

Divided By 4 Block: 00 11 11 00

Concatenated By 101: 101 00 101 101 11 101 101 11 101 101 00 101

ASCII value of New Key: 165 189 189 165

Decryption:

CiphertText: i

ASCII Value: 105

Binary Value: 1101001

Random Number: 298

Mod of R.N. By n6: 298 % 60 =58= $(111010)_2$

XOR with ciphertext: 105 XOR 58= $(1010011)_2$

ASCII Value of plaintext: 83

Plaintext: S

7th iteration:

Key generation:

ASCII Value: 165 189 189 165

Summation of ASCII Value: 708

Mod by 100(n7): 8

Binary Value: 00001000

Divided By 4 Block: 00 00 10 00

Concatenated By 110: 110 00 110 110 00 110 110 10 110 110 00 110

ASCII value of New Key: 198 198 214 198

Decryption:

Ciphertext: V

ASCII Value: 86

Binary Value: 1010110

Random Number: 298

Mod of R.N. By n7: 298 % 8 =2= $(10)_2$

XOR with ciphertext: 86 XOR 2= $(1010100)_2$

ASCII Value of plaintext: 84

Plaintext: T

Plaintext: MY MIST

3.3.4 The LWE Algorithm

Encryption Algorithm

In Algorithm 1 we show how to encrypt a plaintext data to cipher text data .

Algorithm 1 Plaintext (P) To Cipher text (C)

Require: $L.Key == 4 \vee L.P \neq 0$

Ensure:

$S \leftarrow Key(1) + Key(2) + Key(3) + Key(4)$
$i \leftarrow 1$
$conV \leftarrow 0$
while $i \neg L.P$ **do**
  $S \leftarrow S$ mod 100
  $Y \leftarrow 298$ mod S
  $F \leftarrow Y$ XOR $P(i)$
  $C(i) \leftarrow F$
  $Z \leftarrow Bin(S, 8)$
  $B1 \leftarrow$ CONCATE [Z(1),Z(2)]
  $B2 \leftarrow$ CONCATE [Z(3),Z(4)]
  $B3 \leftarrow$ CONCATE [Z(5),Z(6)]
  $B4 \leftarrow$ CONCATE [Z(7),Z(8)]
  $conB \leftarrow Bin(conV, 3)$
  $B1 \leftarrow$CONCATE[conB,B1,conB]
  $B2 \leftarrow$CONCATE[conB,B2,conB]
  $B3 \leftarrow$CONCATE[conB,B3,conB]
  $B4 \leftarrow$CONCATE[conB,B4,conB]
  if $conV >= 7$ **then**
    $conV \leftarrow 0$
  else
    $conV \leftarrow conV + 1$
  end if
end while

Decryption Algorithm

In Algorithm 2 we show how to decrypt a cipher text data to plain text data.

Algorithm 2 Cipher text (C) To Plaintext (P)

Require: $L.Key == 4 \vee L.C \neq 0$

Ensure:

$S \leftarrow Key(1) + Key(2) + Key(3) + Key(4)$
$i \leftarrow 1$
$conV \leftarrow 0$
while $i \neg L.C$ **do**
  $S \leftarrow S$ mod 100
  $Y \leftarrow 298$ mod S
  $F \leftarrow Y$ XOR $C(i)$
  $P(i) \leftarrow F$
  $Z \leftarrow Bin(S, 8)$
  $B1 \leftarrow$ CONCATE [Z(1),Z(2)]
  $B2 \leftarrow$ CONCATE [Z(3),Z(4)]
  $B3 \leftarrow$ CONCATE [Z(5),Z(6)]
  $B4 \leftarrow$ CONCATE [Z(7),Z(8)]
  $conB \leftarrow Bin(conV, 3)$
  $B1 \leftarrow$CONCATE[conB,B1,conB]
  $B2 \leftarrow$CONCATE[conB,B2,conB]
  $B3 \leftarrow$CONCATE[conB,B3,conB]
  $B4 \leftarrow$CONCATE[conB,B4,conB]
  if $conV >= 7$ **then**
    $conV \leftarrow 0$
  else
    $conV \leftarrow conV + 1$
  end if
end while

3.3.5 Sampling Strategy

We have categorized the data into seven main categories: only characters, only numbers, only special characters, char-special char, char-number, num-special character, char-num-special character. We have taken 300 files of each category. So there are total 2800 (300*7) test files. The files are 1-512 KB in size as our algorithm will deal only with small scale data. The file types are .txt and .doc. we have used random sampling in data. Random samples are used in population sampling situations when reviewing historical or batch data.

The key to random sampling is that each unit in the population has an equal probability of being selected in the sample. Using random sampling protects against bias being introduced in the sampling process, and hence, it helps in obtaining a representative sample. In general, random samples are taken by assigning a number to each unit in the population and using a random number table or Minitab to generate the sample list. Absent knowledge about the factors for stratification for a population, a random sample is a useful first step in obtaining samples. For example, an improvement team in a human resources department wanted an accurate estimate of what proportion of employees had completed a personal development plan and reviewed it with their managers. The team used its database to obtain a list of all associates. Each associate on the list was assigned a number. Statistical software was used to generate a list of numbers to be sampled, and an estimate was made from the sample. We have run the files on different platforms like windows, Linux, Mac. So our research and analysis is platform independent.

3.3.6 Data Collection

Data collection is the process of gathering and measuring information on variables of interest, in an established systematic fashion that enables one to answer stated research questions, test hypotheses, and evaluate outcomes. The data collection component of research is common to all fields of study including physical and social sciences, humanities, business, etc. While methods vary by discipline, the emphasis on ensuring accurate and honest collection remains the same. The goal for all data collection is to capture quality evidence that then translates to rich data analysis and allows the building of a convincing and credible answer to questions that have been posed. Regardless of the field of study or preference for defining data (quantitative, qualitative), accurate data collection is essential to maintaining the integrity of research. Both the selection of appropriate data collection instruments (existing, modified, or newly developed) and clearly delineated instructions for their correct use reduce the likelihood of errors occurring. A formal data collection process is necessary as it ensures that data gathered are both defined and accurate and that subsequent decisions based on arguments embodied in the findings are valid. The process provides both a baseline from which to measure and in certain cases a target on what to improve.
Generally there are four types of data collection and they are:

- **Surveys:** Standardized paper-and-pencil or phone questionnaires that ask predetermined questions.
- **Interviews:** Structured or unstructured one-on-one directed conversations with key individuals or leaders in a community.

- **Focus groups:** Structured interviews with small groups of like individuals using standardized questions, follow-up questions, and exploration of other topics that arise to better understand participants.
- **Experiments:** Experiments are done on test subjects to collect data.

As our research is experimental in nature, we have used experiments and diagnosis as basis of our data collection. All the test files were run by the encryption algorithm and their corresponding values were obtained. Great caution was taken during data collection because any mistake in data collection would lead to faulty research conclusions.

3.3.7 Data Analysis

Data analysis is the most important phase of Experimental research because by the analysis of collected data, conclusion is drawn. Analysis of data is a process of inspecting, cleaning, transforming, and modeling data with the goal of discovering useful information, suggesting conclusions, and supporting decision-making. Data analysis has multiple facets and approaches, encompassing diverse techniques under a variety of names, in different business, science, and social science domains. The data necessary as inputs to the analysis are specified based upon the requirements of those directing the analysis or customers who will use the finished product of the analysis. The general type of entity upon which the data will be collected is referred to as an experimental unit (e.g., a person or population of people). Specific variables regarding a population (e.g., age and income) may be specified and obtained. Data may be numerical or categorical (i.e., a text label for numbers). Analysts may apply a variety of techniques referred to as exploratory data analysis to begin understanding the messages contained in the data. The process of exploration may result in additional data cleaning or additional requests for data, so these activities may be iterative in nature. We have used generalized samples as our samples contain files of various extensions, data categories and are run on different platforms. The sample is also large as there are 2800 test files.

3.3.8 Limitation

Our research focuses on small scale data security. It does not deal with large scale data. Our test files taken are 1-512 KB in size. So our algorithm will not be efficient for large data and it is not applicable for encrypting files other than .txt and .doc extension. Individual users and small companies will be benefitted from our algorithm but it will be misleading and ineffective for large companies and organizations which needs extreme privacy, security and deal with huge amounts of data at a time.

3.3.9 Reliability and Validity

The idea behind reliability is that any significant results must be more than a one-off finding and be inherently repeatable. Other researchers must be able to perform exactly the same experiment, under the same conditions and generate the same results. This will reinforce the findings and ensure that the wider scientific community will accept the hypothesis. Without this replication of statistically significant results, the experiment and research have not fulfilled all of the requirements of testability. This prerequisite is essential to a hypothesis establishing itself as an accepted scientific truth. For example, if we are performing a time critical experiment, we should be using same type of stopwatch. Generally, it is reasonable to assume that the instruments are reliable and will keep true and accurate time. However, diligent scientists take measurements many times, to minimize the chances of malfunction and maintain validity and reliability.

Validity encompasses the entire experimental concept and establishes whether the results obtained meet all of the requirements of the scientific research method. For example, there must have been randomization of the sample groups and appropriate care and diligence shown in the allocation of controls. Internal validity dictates how an experimental design is structured and encompasses all of the steps of the scientific research method. Even if the results are great, sloppy and inconsistent design will compromise integrity in the eyes of the scientific community. Internal validity and reliability are at the core of any experimental design. External validity is the process of examining the results and questioning whether there are any other possible causal relationships. Control groups and randomization will lessen external validity problems but no method can be completely successful. This is why the statistical proofs of a hypothesis called significant, not absolute truth. Any scientific research design only puts forward a possible cause for the studied effect. There is always the chance that another unknown factor contributed to the results and findings. This extraneous causal relationship may become more apparent, as techniques are refined and honed. From the above discussion, we come to the decision that our study is reliable and valid to a large extent because the test files were run iteratively under same conditions and linear results were found. Again as we used random data for test files, the results were valid and there were no other causal relationships among the results.

3.3.10 Ethical Consideration

There are a number of ethical principles that should be taken into account when performing undergraduate and master's level dissertation research. At the core, these ethical principles stress the need to (1) do good (known as beneficence) and (2) do no harm (known as non-malfeasance).

In practice, these ethical principles mean that as a researcher, we need to:(a) obtain informed consent from potential research participants; (b) minimize the risk of harm to participants; (c) protect their anonymity and confidentiality; (d) avoid using deceptive practices; and (e) give participants the right to withdraw from our research. This section discusses these five ethical principles and their practical implications with respect to our research work.
The data we collected and analyzed are from sources which dont have a copyright issue and are open for all to use. We will use the data only for this research purpose and there is no other intention about it. We will not be engaged into any financial or commercial activity by using these data. As our test subjects are not human, there is no scope of taking consent or giving the right to withdraw from our research. We have not used any deceptive means to gather data and information about this research and were free and frank about our research output to our supervisor and honorable authority. Above all, we dont mean to cause anyone slightest harm by our research and are committed to do good to our users by helping them to protect their data efficiently and cheaply.

3.3.11 Summary

The above discussion shows that our research was done following scientific and accepted methodology, design and strategy. Data collection was done using well defined methods and data reliability and validity was ensured. Ethical considerations were taken into account and the research was done fulfilling all the standard conditions. So this research will hopefully be accepted and valued by the scientific community and will be useful for the target people.

CHAPTER 4
EXPERIMENTAL RESULT, ANALYSIS AND DISCUSSION

Experimental result, analysis and discussion of the proposed and developed light weight symmetric key encryption and decryption algorithm have been discussed on this chapter. The overall study not only deals with the development of a lightweight symmetric key cryptography algorithm but also finds it's efficiency and percentage of error. The algorithm does not discuss about heavy scale and asymmetric key algorithms and never compare the new algorithm with these two kinds of algorithms. It also does not cover the detailed comparison of all the features of the new algorithm with the existing algorithms. It only deals with English characters and ignores other language characters. As our proposed algorithm (LWE) is a light weight symmetric key cryptography algorithm, it only deals with small scale text files around 1 K.B to 512 K.B and doc type files around 1 K.B to 66.5 K.B. As a result, the encryption and decryption process of the algorithm is less time consuming, less complex as well as little percentage of error. It is easy to implement and also provides better security during data transmission. This lightweight symmetric key cryptography algorithm deals with text file which contents character, number, special character etc and also with their combination.

4.1 Experimental Results

For the simulation of lightweight symmetric key cryptography algorithm Mathlab had been used. For result analysis, 1 K.B to 512 K.B range text file and 1 K.B to 66.5 K.B doc type files had been encrypted and decrypted and its execution time had been collected (see Appendix). There were 7 categories of text files around 1 K.B to 512 K.B and doc type files around 1 K.B to 66.5 K.B had been executed on three different platforms like Windows, Linux and Mac OS and the categories of files were

1. Plain character data text file and doc type file.
2. Plain number data text file and doc type file.

3. Plain special character data text file and doc type file.

4. Plain character data and number data text file and doc type file.

5. Plain special character data and number data text file and doc type file.

6. Plain character data and special character data text file and doc type file.

7. Plain character data, special character data and number data text file and doc type file.

4.1.1 Analysis on Category 1 Only plain Character data

In this category, the text file and doc type file contained only plain character data where the sizes of the text files were in 1 K.B to 512 K.B and doc type files were 1 K.B to 66.5 K.B (See Appendix B). For the analysis of category 1 in result analysis file type vs average execution time had been calculated.

Table 4.1: File type versus average execution time calculation(category-1)

File category	File Type	Used Platform	Used Key length	Average iteration	Per iteration Execution time	Total Execution Time(sec)
Category -1 (150 K.B for txt 20K.B for doc type)	.txt	Windows	4	164,410	1232	133.393
	.doc/.docx	Windows	4	164,410	1215	135.252
	.txt	Linux	4	164,410	1213	135.459
	.odt/.fodt/.uot	Linux	4	164,410	1213	135.323
	.txt	Mac OS	4	164,410	1204	136.489
	.page	Mac OS	4	164,410	1206	134.256
Txt file equivalent type .txt and doc file equivalent type .doc, .docx, .odt, .fodt and .uot						

The table shows lightweight symmetric key cryptography algorithm simulation on different platform (Windows, Linux, Mac OS) on 150 K.B text file and 20 K.B doc type file. The table contains **file category, file type, used platform, used key length, average iteration, per iteration/execution time and total execution time**. File category means one of the seven categories of text files and doc type files which are discussed above. Used platform means which platform has been used during the simulation. Used key length of the proposed lightweight symmetric key cryptography algorithm is fixed by default but user can use any combination of 4-key length. Average iteration of proposed lightweight symmetric key cryptography algorithm depends on the total number of character data in the text file and doc type file. Per iteration/execution time explains the total number of average iteration have been completed in one second. Total execution time explains the total time required for completing the average iteration in the given text file and doc type file.

4.1.2 Analysis on Category 2 Only plain number data

In category-2, the text file and doc type file contained only plain number data where the sizes of the text files were 1 K.B to 512 K.B and doc type files were 1 K.B to 66.5 K.B . For the analysis of category 2 in result analysis file type vs. average execution time had been calculated.

Table 4.2: File type versus average execution time calculation(category-2)

File category	File Type	Used Platform	Used Key length	Average iteration	Per iteration Execution time	Total Execution Time(sec)
Category -2 (150 K.B for txt 20K.B for doc type)	.txt	Windows	4	115393	902	127.876386
	.doc/.docx	Windows	4	115393	902	127.1125
	.txt	Linux	4	115393	887	129.97856
	.odt/.fodt/.uot	Linux	4	115393	886	128.555
	.txt	Mac OS	4	115393	891	129.48796
	.page	Mac OS	4	115393	890	128.1235
Txt file equivalent type .txt and doc file equivalent type .doc, .docx, .odt, .fodt and .uot						

File type vs. average execution time calculation table contains file category, file type, used platform, used key length, average iteration, per iteration/execution time and total execution time where the file size is 150 K.B for txt type file and 20 K.B for doc type file. File category means one of the seven categories where this table shows details of category 2. Used platform means which platform has been used during the simulation where the platforms can be windows, Linux, Mac OS. Used key length of the proposed lightweight symmetric key cryptography algorithm is fixed by default but user can use any combination of 4-key length. Average iteration of proposed lightweight symmetric key cryptography algorithm depends on the total number of number data in the text file and doc type file. Per iteration/execution time explains the total number of iterations have been completed in one second. Total execution time explains the total time required for completing the average iteration in the given text file and doc type file.

4.1.3 Analysis on Category 3 only plain special character data

Category-3 type, the text file and doc type file contained only plain special character data where the sizes of the text files were in between 1 K.B to 512 K.B and doc type files were in between 1 K.B to 66.5 K.B . For the analysis of category 3 in result analysis file type vs average execution time had been calculated.

Table 4.3: File type versus average execution time calculation(category-3)

File category	File Type	Used Platform	Used Key length	Average iteration	Per iteration Execution time	Total Execution Time(sec)
Category -3 (150 K.B for txt 20K.B for doc type)	.txt	Windows	4	115393	902	127
	.doc/.docx	Windows	4	115393	902	127.19755
	.txt	Linux	4	115393	887	129.125
	.odt/.fodt/.uot	Linux	4	115393	886	128.578
	.txt	Mac OS	4	115393	891	129.90
	.page	Mac OS	4	115393	890	128.8934
Txt file equivalent type .txt and doc file equivalent type .doc, .docx, .odt, .fodt and .uot						

The table shows file category, file type, used platform, used key length, average iteration, per iteration/execution time and total execution time in seconds. The table contains lightweight symmetric key cryptography algorithm simulation on different platform which are Windows, Linux, Mac OS on 150 K.B text file and 20 K.B doc type file. File category means one of the seven categories of text file as well as doc type files discussed above. User can use any combination of 4-key length for used key length for the lightweight symmetric key cryptography algorithm simulation but the length of key is fixed. Used platform means which platform has been used during the simulation. Average iteration of proposed lightweight symmetric key cryptography algorithm is the total number of special character data in the text file and doc type file. Per iteration/execution time is the total number of iterations that have been completed in one second. Total execution time is the total time required for completing the average iteration in the given text file.

4.1.4 Analysis on Category 4 plain character and number data

This category text files and doc type file contained combination of plain character data and number data where the sizes of the text files were in between 1 K.B to 512 K.B and doc type files were in between 1 K.B to 66.5 K.B . For the analysis of category 4 in result analysis file type vs average execution time had been calculated.

File type vs. average execution time calculation table shows lightweight symmetric key cryptography algorithm simulation on different platforms (Windows, Linux, Mac OS) on 150 K.B text file as well as 20 K.B doc type file. File category means one of the seven categories of text files and doc type files where this table discusses category type 4. Used platform is the platform that has been used during the simulation of lightweight symmetric key cryptography algorithm. Used key length of the proposed lightweight symmetric key cryptography algorithm is fixed but user can use any combination of 4-key length. Average iteration of proposed lightweight symmetric key cryptography algorithm is the total number of character data and number data in the text files and doc type files. Per iteration/execution

Table 4.4: File type versus average execution time calculation(category-4)

File category	File Type	Used Platform	Used Key length	Average iteration	Per iteration Execution time	Total Execution Time(sec)
Category -4 (150 K.B for txt 20K.B for doc type)	.txt	Windows	4	116504	945	127
	.doc/.docx	Windows	4	116504	941	131.955
	.txt	Linux	4	116504	911	130.0
	.odt/.fodt/.uot	Linux	4	116504	910	128.157
	.txt	Mac OS	4	116504	909	129.90
	.page	Mac OS	4	116504	909	128.9634
Txt file equivalent type .txt and doc file equivalent type .doc, .docx, .odt, .fodt and .uot						

time is the total number of iterations had been completed per second. Total execution time is the total time required for completing the average iteration in the given text file as well as the doc type files.

4.1.5 Analysis on Category 5 plain character and special character data

In this case, the text file and doc type file contained combination of plain character data and special character data where the sizes of the text files were in between 1 K.B to 512 K.B and doc type files were in between 1 K.B to 66.5 K.B . For the analysis of category 5 in result analysis file type vs average execution time had been calculated.
This table shows lightweight symmetric key cryptography algorithm simulation on different

Table 4.5: File type versus average execution time calculation(category-5)

File category	File Type	Used Platform	Used Key length	Average iteration	Per iteration Execution time	Total Execution Time(sec)
Category -5 (150 K.B for txt 20K.B for doc type)	.txt	Windows	4	116504	980	127
	.doc/.docx	Windows	4	109405	972	133.0
	.txt	Linux	4	109405	969	132.90
	.odt/.fodt/.uot	Linux	4	109405	969	131.157
	.txt	Mac OS	4	109405	964	131.90
	.page	Mac OS	4	109405	963	130.34
Txt file equivalent type .txt and doc file equivalent type .doc, .docx, .odt, .fodt and .uot						

platforms (Windows, Linux, Mac OS) for 20 K.B doc type file and for 150 K.B text file and also shows the file category, file type, used platform, used key length, average iteration, per iteration/execution time and total execution time. File category means one of the seven categories of text files and doc type files where this table discusses category type 5. Used platform is the platform that has been used during the simulation. Used key length of the

proposed lightweight symmetric key cryptography algorithm is fixed but user could use any combination of 4-key length. Average iteration of proposed lightweight symmetric key cryptography algorithm depends on the total number of character data and special character data in the text file as well as in the doc type files. Per iteration/execution time explains the total number of iterations per second. Total execution time shows the total time required for completing the average iteration in the given text file as well as doc type file in seconds.

4.1.6 Analysis on Category 6 plain number and special character data

Category 6 type text files as well as doc type files contained the combination of plain number data and special character data where the sizes of the text files were 1 K.B to 512 K.B and doc type file were 1 K.B to 66.5 K.B . For the analysis of category 6 in result analysis file type vs average execution time had been calculated.

Table 4.6: File type versus average execution time calculation(category-6)

File category	File Type	Used Platform	Used Key length	Average iteration	Per iteration Execution time	Total Execution Time(sec)
Category -6 (150 K.B for txt 20K.B for doc type)	.txt	Windows	4	164,410	1232	133.393
	.doc/.docx	Windows	4	164,410	1215	135.252
	.txt	Linux	4	164,410	1213	135.459
	.odt/.fodt/.uot	Linux	4	164,410	1213	135.323
	.txt	Mac OS	4	164,410	1204	136.489
	.page	Mac OS	4	164,410	1206	134.256
Txt file equivalent type .txt and doc file equivalent type .doc, .docx, .odt, .fodt and .uot						

The file type vs. average execution time calculation table shows file category, file type, used platform, used key length, average iteration, per iteration/execution time and total execution time for lightweight symmetric key cryptography algorithm simulation on different platforms (Windows, Linux, Mac OS) for 150 K.B text file and for 20 K.B doc type files. File category means one of the seven categories of text files and doc type files. Used key length is fixed by default but user can use any combination of 4-key length if wishes. Used platform discusses which platform has been used during the simulation. Average iteration of proposed lightweight symmetric key cryptography algorithm is the total number of character in the text. Per iteration/execution time explains the total number of iteration per second. Total execution time explains the total time required for completing the average iteration in the given text file and doc files.

4.1.7 Analysis on Category 7 plain character, number and special character data

Category 7 type text files as well as doc type files contained the combination of plain number data, special character data and character data where the sizes of the text files were 1 K.B to 512 K.B and doc type file were 1 K.B to 66.5 K.B . For the analysis of category 7 in result analysis file type vs average execution time had been calculated.

This table shows file category, file type, used platform, used key length, average iteration,

Table 4.7: File type versus average execution time calculation(category-7)

File category	File Type	Used Platform	Used Key length	Average iteration	Per iteration Execution time	Total Execution Time(sec)
Category -7 (150 K.B for txt 20K.B for doc type)	.txt	Windows	4	151,912	1201	130.9
	.doc/.docx	Windows	4	151,912	1196	130.0
	.txt	Linux	4	151,912	1195	129.8
	.odt/.fodt/.uot	Linux	4	151,912	1193	128.0
	.txt	Mac OS	4	151,912	1194	128.1
	.page	Mac OS	4	151,912	1193	127.5
Txt file equivalent type .txt and doc file equivalent type .doc, .docx, .odt, .fodt and .uot						

per iteration/execution time and total execution time. File category means one of the seven categories of text files and doc type files where this table analysis category 7. Used platform means which platform has been used during the simulation where the used platforms are Windows, Linux, Mac OS. The combination of key is any which can be decided by the user but key length is fixed and cannot be changed. Average iteration depends on the total number of combination of character data, special character data and number data in the text files and doc type files. Per iteration/execution time is the total number of iterations completed per second. Total execution time is the total time required for the completion the average iteration in the given text file as well as doc type files.

4.2 Quantitative Analysis

Quantitative analysis is a business or financial analysis technique that seeks to understand the behavior by using complex mathematical and statistical modeling, measurement and research. By assigning a numerical value to variables, quantitative analysts try to replicate reality mathematically. Quantitative analysis can be done for a number of reasons such as measurement, performance evaluation or valuation of a financial instrument. It can also be used to predict real world events such as changes in a share price. Quantitative analyses have been successfully applied on various fields like for quantifying the side-channel leakage in cryptographic algorithms, for capturing the loss of privacy in statistical data analysis, and for

quantifying security in anonymity networks. In the lightweight symmetric key cryptography algorithm, the need for quantitative analyses has been recognized. Performing quantitative analysis on a lightweight symmetric key cryptography algorithm is a challenging problem due to the complexity of modern software. It is mandatory to provide developers with tool support for this task. One goal of the quantitative analysis on lightweight symmetric key cryptography algorithm is to explain the importance of this algorithm in security, their connection to programming languages and verification techniques, and the limitations of the lightweight symmetric key cryptography algorithm. For quantitative analysis on lightweight symmetric key cryptography algorithm 500 text file as well as doc type files had been executed and their average iteration, per iteration/execution time, total execution time in seconds had been calculated in order to find out the lightweight symmetric key cryptography algorithms efficiency, the problem and the effectiveness of the algorithm on security. From the analysis of our proposed lightweight symmetric key cryptography algorithm it was seen that the proposed lightweight symmetric key cryptography algorithm was quite efficient, effective and less problem less. It was only worked with English characters and ignored other language characters. Since our proposed cryptography algorithm was a light weight symmetric key cryptography algorithm, it encrypted and decrypted a small scale txt file as well as doc type files around 1 K.B to 512 K.B. Because of this reason the encryption and decryption process of the algorithm was less time consuming, less complex as well as little percentage of error. It was easy to implement and also provided better security during data transmission. This proposed lightweight symmetric key cryptography algorithm encrypted and decrypted text file containing character, number, special character etc and also with their combination in the text file as well as the doc type files. The text files and the doc type files that had been used on lightweight symmetric key cryptography algorithm were Plain character data text file and doc type file, Plain number data text file and doc type file, Plain special character data text file and doc type file, Plain character data and number data text file and doc type file, Plain special character data and number data text file and doc type file, Plain character data and special character data text file and doc type file, Plain character data, special character data and number data text file and doc type file.

4.3 Discussion

Our proposed lightweight symmetric key cryptography algorithm (LWE) was used on 500 text file as well as doc type files where sizes of the files were 1 K.B to 512 K.B. There were three different types of platforms used for this purpose. The platform that had been used for the proposed lightweight symmetric key cryptography algorithm were Linux, widows, Mac OS in order to find out the efficiency of the algorithm in different platform and to define its platform independent so that any user of any platform could be benefitted using

this algorithm. Their average iteration, per iteration/execution time, total execution time had been calculated in order to find out efficiency, the problem and the effectiveness of the algorithm on security. This proposed lightweight symmetric key cryptography algorithm provided security on seven types of text files as well as the doc types files. They were Plain character data text file and doc type file, Plain number data text file and doc type file, Plain special character data text file and doc type file, Plain character data and number data text file and doc type file, Plain special character data and number data text file and doc type file, Plain character data and special character data text file and doc type file, Plain character data, special character data and number data text file and doc type file where the sizes of text files were around 1 K.B to 512 K.B as well as the doc type files were around 1 K.B to 66.5 K.B. As a result, the encryption and decryption process of the algorithm was less time consuming, less complex as well as little percentage of error. Because of this, LWE (lightweight encryption) algorithm was easy to implement and also provide better security during data transmission. LWE (lightweight encryption) algorithm ensures data stability, integration and privacy. It is not feasible for small industries to spend large amount of money for buying costly antivirus soft-wares and other security measures. Our algorithm provides a cheap and effective alternative for them. The algorithm had been tested on various platforms and with various categories of input data which makes it very reliable and valid. So the user could have valid output at a low price. In result analysis, the lightweight symmetric key cryptography algorithm was executed on different platforms (Windows, Linux, Mac OS) for 150 K.B text file and for 20 K.B for type files showed and discussed using table. It also discussed about file category, file type, used platform, used key length, average iteration, per iteration/Execution time and total Execution time on seven category in three different platforms.

4.4 Summary

Many company and organization need to transfer data which is not big in size but they need better security for the transmission of data. Our LWE (lightweight encryption) algorithm ensures data stability, integration and privacy for those data files which are in size around 1 K.B to 512 K.B. It is not reasonable for small organization to spend large amount of money for small scale of data transmission and other security measures. Our algorithm provides a cheap and effective alternative for them. The algorithm can be used on various platforms and with various categories of input data which makes it very reliable and valid. So the user can easily transfer data with better security at a low price rate. This lightweight symmetric key cryptography algorithm is only valid for English characters, numbers and special character and invalid for other languages. Because our algorithm was a light weight symmetric key cryptography algorithm, it only worked on small scale txt file and doc type files around 1

K.B to 512 K.B. As a result the encryption and decryption process of the algorithm was less time consuming, less complex as well as little percentage of error. It was easy to implement and also provide better security during data transmission. This lightweight symmetric key cryptography algorithm worked with text files and doc type files which contained character, number, special character etc and also with their combination in text file as well as doc type files on different platform which were Windows, Linux, Mac OS.

CHAPTER 5
CONCLUSION AND RECOMMENDATION

This is the last chapter of our research. This chapter restates our beliefs and findings and reaffirms why the topic is important and should be studied. It also recommends what future works can be done on the topic and gives a direction to the interested people. This helps other researchers to understand how far the present study goes and what the limitations of the study are. They can then start from where we left.

5.1 Conclusion

The conclusion of a research thesis reaffirms the thesis statement, discusses the issues, and reaches a final judgment. The conclusion is not a summary; it is a belief based on our reasoning and on the evidence we have accumulated. This is the place to share with readers the conclusions that have been reached because of our research. The conclusion attempts to carry the examiner or reader to a new level of perception about the thesis. A summary of what have been said in the thesis is not satisfactory. After all, the reader will hardly need reminding of things just read. The nature of the study can dictate overall content of the conclusion. However, it should particularly reaffirm the thesis statement and seek to offer answers to the questions raised in the research and justification for the approach used by the study as well as pathways forward. The purpose of a conclusion is to tie together, or integrate the various issues, research, etc., covered in the body of the thesis, and to make comments upon the meaning of all of it. This includes noting any implications resulting from our discussion of the topic, as well as recommendations, forecasting future trends, and the need for further research. The conclusion chapter or section seeks to tie together, integrate and synthesize the various issues raised in the discussion sections, whilst reflecting the introductory thesis statement (s) or objectives, provide answers to the thesis research question (s), identify the theoretical and policy implications of the study with respect to the overall study area, highlights the study limitations, provide direction and areas for future research.

The main background or context of this study was that lightweight encryptions are becoming the trend of the age and are giving reasonable performance within low cost as compared to

large scale security systems. So our aim was to develop a robust and lightweight symmetric key encryption algorithm which will efficiently work on small scale data with minimal costing. The LWE algorithm fulfills those needs and is user friendly. The deductive paradigm was taken that is first we proposed an idea of the LWE algorithm by studying the existing symmetric key algorithms and then developed the algorithm by following proper method and then tested the algorithm against data, analyzed it and finally found the result of the experiment. As we did experiments to establish our hypothesis and then explored new facts, so our research is experimental and exploratory in design. Again since we used mathematical models, theories and hypotheses to construct our hypotheses and measured the performance and error rate to analyze the data, so our research is a quantitative analysis.

The data we collected was chosen randomly and run on various platforms. So it is reliable and valid as similar and linear results were found in the testing phase. As we took a large number of test files, the data is generalized also that is it can be said to be applicable for a large number of samples. We followed a clear and scientific method for collecting and analyzing data and no one was harmed during the process nor do we intend to harm anyone in the future by this research . So our research is ethical and lawful. Thus our research reaffirms the hypotheses that LWE is cheaper and efficient than large scale encryption systems and operates accurately on small scale data.

5.2 Recommendation

Feasibility study is an analysis of the viability of an idea. The feasibility study focuses on helping answer the essential question of "*should we proceed with the proposed project idea?* "All activities of the study are directed toward helping answer this question. Feasibility studies aim to objectively and rationally uncover the strengths and weaknesses of an existing business or proposed venture, opportunities and threats present in the environment, the resources required to carry through, and ultimately the prospects for success. In its simplest terms, the two criteria to judge feasibility are cost required and value to be attained. Our study will replace large scale encryption systems from small companies and individual users which cause much money and memory waste and are redundant for them. This LWE algorithm will provide an easy alternative for them to protect their data efficiently at a cheap rate. So the feasibility of our research is beyond doubt and is very practical and easy to implement.

The algorithm will be implemented in three main stages- Random Key Assignment, Key Encryption and New Key generation and finally Small Scale Data Encryption and Decryp-

tion. First, random four letters key will be given as input, then encryption will be done using that key and then new keys will be generated from old ones and this will continue up to the number of words in the sentence. The performance of encryption has been evaluated and it is reasonably good compared to the resources needed. So users will be able to get better security, privacy and data integration.

5.3 Future Study

This study only deals with .txt , .doc equivalant file types of three different OS platforms. It is not applicable for other types of files. It could correctly encrypt and decrypt files of size 1-512 K.B of .txt and equivalent format and 1-66.5 K.B of .doc and equivalent format. Future works can be done on applying LWE algorithm on all types of files including .pdf, media files and .db files etc. Besides, our algorithm encrypts only English characters, special characters, numbers and their combinations. It is not applicable for other languages. Future studies may include other languages also by modifying the algorithm for UNICODE characters. As our time was limited, we couldn't assess the robustness of the LWE algorithm by comparing it with other existing encryption algorithms. Future research can be done by assessing the robustness of this algorithm by comparing it with other existing lightweight encryption algorithms. The size of the text files may also vary from 512 KB for image, .pdf, media and .db files. To encrypt image files with this LWE algorithm, first the image has to be converted into matrix. Then a pattern has to be generated within the matrix. To generate the pattern, we need a key. So for the modified algorithm for image encryption and decryption, two different keys are needed. One key is for pattern generation and another for encryption and decryption. Our research can act as a guideline for further researches which will enhance the lightweight encryption system in a far greater way and increase the performance and scope manifolds.

REFERENCES

[1] W. Wanping, J. Jin, and J. Cheng, "The research and design of ATM PIN pad based on triple DES," in *IEEE International Conference on Information and Automation (ICIA)*, (Shenzhen), pp. 443 – 447, IEEE, 2011.

[2] S. Kurita and K. Komoriya, "Privacy protection on transfer system of Automated Teller Machine from brute force attack," in *International Conference on Advanced Information Networking and Applications Workshops (WAINA)*, (Fukuoka), pp. 72 – 77, IEEE, 2012.

[3] O. Dandash, P. D. Le, and B. Srinivasan, "Security analysis for internet banking models," in *Eighth ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing*, (Qingdao), pp. 1141 – 1146, IEEE, 2007.

[4] A. Hiltgen, T. Kramp, and T. Weigold, "Secure internet banking authentication," *Security & Privacy, IEEE*, pp. 21 – 29, 2006.

[5] V. Cunsolo and S. Distefano, "Achieving information security in network computing systems," in *Eighth IEEE International Conference on Dependable, Autonomic and Secure Computing*, (Chengdu), pp. 71 – 77, IEEE, 2009.

[6] C. Manifavas, G. Hatzivasilis, K. Fysarakis, and K. Rantos, "Lightweight cryptography for embedded systems a comparative analysis," (Egham), 6th International Workshop on Autonomous and Spontaneous Security, 2013.

[7] D. Engels, X. Fan, G. Gong, H. Hu, and E. M. Smith, "Hummingbird: ultra-lightweight cryptography for resource-constrained devices," in *FC'10 Proceedings of the 14th international conference on Financial cryptograpy and data security*, (Heidelberg), pp. 3–18, Springer-Verlag Berlin, 2010.

[8] J. V. Leeuwen, *Handbook of Theoretical Computer Science*. Massachusetts: The MIT Press, 1994.

[9] O. Goldreich, *Foundations of Cryptography*. Cambridge: Cambridge university press, 2004.

[10] H. Delfs and H. Knebl, *Introduction to cryptography: principles and applications*. Nurnberg: Springer Publications, 2007.

[11] L. A. Baglione, "Writing a research paper in political science: A practical guide to inquiry, structure, and methods," Thousand Oaks: CQ Press, 2nd edition ed., 2012.

[12] R. M. Nieswiadomy, *Foundations of Nursing Research*. Michigan: Appleton & Lange, 1998.

[13] J. H. Geri LoBiondo-Wood, *Nursing Research: Methods and Critical Appraisal for Evidence-Based Practice, (Nursing Research: Methods, Critical Appraisal & Utilization)*. Sydney: Libby Houston and Melinda McEvoy, 8 th ed., 2013.

[14] C. Shannon, "Communication theory of secrecy systems," 1949.

[15] B. Schneier, "Applied cryptography: Protocols, algorithms, and source code in c," New Jersey: John Wiley & Sons, 1996.

[16] S. Knapskog, "New cryptographic primitives," in *Computer Information Systems and Industrial Management Applications (CISIM) Conference*, (Ostrava), pp. 3 – 7, IEEE, 2008.

[17] A. Kumar, K. Gopal, and A. Aggarwal, "A complete, efficient and lightweight cryptography solution for resource contrainst mobile ad-hoc networks," in *2nd IEEE International Conference on Parallel Distributed and Grid Computing (PDGC)*, (Solan), pp. 854 – 860, IEEE, 2012.

[18] A. Barbero, G. Horler, A. Kholosha, and O. Ytrehus, "Lightweight cryptography for rfid devices," in *IET International Conference on Wireless, Mobile and Multimedia Networks*, (Beijing), pp. 294 – 297, IET, 2008.

[19] M. Stottinger, S. Huss, S. Muhlbach, and A. Koch, "Side-channel resistance evaluation of a neural network based lightweight cryptography scheme," in *IEEE/IFIP 8th International Conference on Embedded and Ubiquitous Computing (EUC)*, (Hong Kong), pp. 603 – 608, IEEE, 2010.

[20] S. Tripathy and S. Nandi, "Lcase: Lightweight cellular automata-based symmetric-key encryption," *International Journal of Network Security*, p. 243, 2009.

[21] D. Bradbury, "feature: Encryption-the-key-to-secure-data: computerweekly.," April 2005. Retrieved December 20, 2014, from computerweekly: http://www.computerweekly.com/feature/Encryption-the-key-to-secure-data.

[22] M. Katagi and S. Moriai, "Lightweight cryptography for the internet of things," Tokyo: Sony Corporation, 2010.

[23] S. Matsuda and S. Moriai, "Lightweight cryptography for the cloud: exploit the power of bitslice implementation," in *Proceedings of the 14th international conference on Cryptographic Hardware and Embedded Systems*, (Heidelberg), pp. 408–425, Springer-Verlag Berlin, 2012.

[24] S. Irny and A. Rose, "Designing a strategic information systems planning methodology for malaysian institutes of higher learning," *Issues in Information System*, 2005.

[25] T. S. Kuhn, *The Structure of Scientific Revolutions*. Chicago: University of Chicago Press, 1962.

[26] M. Q. Patton, *Qualitative Evaluation and Research Methods*. London: SAGE Publications, 1990.

[27] J. Wilson, *Essentials of Business Research: A Guide to Doing Your Research Project*. SAGE Publications, 2010.

[28] M. Saunders, *Research Methods for Business Students*. 2003: Pearson Education India, 2003.

[29] U. Flick, *An Introduction to Qualitative Research*. Thousand Oaks: SAGE Publications Ltd, 2009.

APPENDIX A
CODES

A.1 Encryption Code

Encryption code is given below.....

```
tic;
fp=fopen('C:\Users\tonmoy\Documents\MATLAB\Thesis\PlainText.txt','r');
x=fscanf(fp,'%c');
fclose(fp);
l=length(x);
conValue=0;
key=input('Please give your key input:','s');
sum=key(1)+key(2)+key(3)+key(4);
fp=fopen('C:\Users\tonmoy\Documents\MATLAB\Thesis\CipherText.txt','w');

for i=1:l

        sum=mod(sum,100);
        yy=mod(298,sum);
        decvalue=x(i)/1;
        xorfinal=bitxor(decvalue,yy);


        %xorfinal
        xorfinal=char(xorfinal);
        fprintf(fp,'%c',xorfinal);

        y = dec2bin(sum,8);
        block1=strcat(y(1),y(2));
        block2=strcat(y(3),y(4));
        block3=strcat(y(5),y(6));
        block4=strcat(y(7),y(8));
```

```
28
29          conValueBin=dec2bin(conValue,3);
30          block1 = strcat(conValueBin,block1,conValueBin);
31          block2 = strcat(conValueBin,block2,conValueBin);
32          block3 = strcat(conValueBin,block3,conValueBin);
33          block4 = strcat(conValueBin,block4,conValueBin);
34
35          if(conValue>=7)
36                  conValue=0;
37          else
38                  conValue=conValue+1;
39          end
40
41          sum=bin2dec(block1)+bin2dec(block2)+bin2dec(block3)+bin2dec(blo
42
43 end
44 fclose(fp);
45 toc;
```

A.2 Decryption Code

Decryption code is given below.....

```
1 tic;
2 fp=fopen('C:\Users\tonmoy\Documents\MATLAB\Thesis\CipherText.txt','r');
3 x=fscanf(fp,'%c');
4 fclose(fp);
5 l=length(x);
6 conValue=0;
7 key=input('Please␣give␣your␣key␣input:','s');
8 sum=key(1)+key(2)+key(3)+key(4);
9 fp=fopen('C:\Users\tonmoy\Documents\MATLAB\Thesis\PlainText_from_cipher
10
11 for i=1:l
12
13          sum=mod(sum,100);
14          yy=mod(298,sum);
15          decvalue=x(i)/1;
16          xorfinal=bitxor(decvalue,yy);
```

```
17
18
19          %xorfinal
20          xorfinal=char(xorfinal);
21          fprintf(fp,'%c',xorfinal);
22
23          y = dec2bin(sum,8);
24          block1=strcat(y(1),y(2));
25          block2=strcat(y(3),y(4));
26          block3=strcat(y(5),y(6));
27          block4=strcat(y(7),y(8));
28
29          conValueBin=dec2bin(conValue,3);
30          block1 = strcat(conValueBin,block1,conValueBin);
31          block2 = strcat(conValueBin,block2,conValueBin);
32          block3 = strcat(conValueBin,block3,conValueBin);
33          block4 = strcat(conValueBin,block4,conValueBin);
34
35          if(conValue>=7)
36                  conValue=0;
37          else
38                  conValue=conValue+1;
39          end
40
41          sum=bin2dec(block1)+bin2dec(block2)+bin2dec(block3)+bin2dec(blo
42
43 end
44 fclose(fp);
45 toc;
```

APPENDIX B
EXECUTION TIME CALCULATION STATISTICS