

**REVIEW OF CLASSICAL CRYPTOGRAPHY AND PROPOSAL OF A HYBRID  
CIPHER**

**BY**

**UMMAR SHEHU  
M.SC/SCIEN/14458/2010/2011**

**A THESIS SUBMITTED TO THE SCHOOL OF POSTGRADUATE STUDIES,  
AHMADU BELLO UNIVERSITY, ZARIA.**

**IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE AWARD OF  
MASTER DEGREE IN MATHEMATICS**

**DEPARTMENT OF MATHEMATICS  
FACULTY OF SCIENCE  
AHMADU BELLO UNIVERSITY, ZARIA  
NIGERIA**

**AUGUST 2015**

## Declaration

I declare that the work in this Thesis entitled “Review of Classical Cryptography and Proposal of a Hybrid Cipher” has been carried out by me under the supervision of Professor D. Singh and Professor S. B. Junaidu in the department of mathematics. The information derived from the literature has been duly acknowledged in the text and list of references provided. No part of this Thesis was previously presented for another degree or diploma at this or any other Institution.

Ummar Shehu

Name of Student

\_\_\_\_\_  
Signature

\_\_\_\_\_  
Date

## Certification

This thesis entitled REVIEW OF CLASSICAL CRYPTOGRAPHY AND PROPOSAL OF A HYBRID CIPHER by Ummar SHEHU meets the regulations governing the award of the degree of M.Sc. Mathematics of the Ahmadu Bello University, and is approved for its contribution to knowledge and literary presentation.

---

Prof. D. Singh

---

Date

Chairman Supervisory Committee

---

Prof. S. B. Junaidu

---

Date

Member Supervisory Committee

---

Dr. Babangida Sani

---

Date

Head of Department

---

Prof. M. Y. Bello

---

Date

External Examiner

---

Prof. Z. Hassan

---

Date

Dean Postgraduate School

## **Acknowledgement**

First and foremost I acknowledge for life and the ability to do this work given to me by the possessor of all things-Allah (SWT).

I profoundly acknowledge the support and contribution given to me by my supervisor, Prof. D. Singh, despite inhibiting challenges. He made this work to be completed.

I also appreciate the encouragement and support extended to me by my second supervisor, Prof. S. B. Junaidu, towards completion of this work.

Moreover, I very much acknowledge the moral support and contribution given to me by Dr. H. G. Dikko.

Finally, I highly appreciate all those that have contributed to the success of this work in one way or the other. These include my family-parents, siblings, wife and my lovely daughter. Also included are departmental members of staff, specially the head of department.

The backbone support for this research work, despite being carried out in a difficult circumstance, was Allah (SWT), through my supervisors and some people close to me and those around me.

## **Abstract**

This work is all about the design of a new cipher called a *hybrid cipher*. The work started with an overview of Cryptography thereafter the design of hybrid cipher, which combines stream and block cipher encryption techniques was achieved. The hybrid cipher is simple but yet apparently a strong cipher that can be used for cryptographic encryption.

## Table of Contents

Title Page.....	i
Declaration.....	ii
Certification.....	iii
Acknowledgement.....	iv
Abstract.....	v
Table of Contents.....	vi
Chapter One.....	1
General Introduction.....	1
1. 1 Introduction.....	1
1. 2 Statement of the Problem.....	3
1. 3 Justification.....	3
1. 4 Aim and Objectives.....	3
1. 5 Methodology.....	4
1. 6 Organization of thesis.....	4
Chapter Two.....	5
Literature Review.....	5
2.1 Introduction.....	5
2.2 Classical Block and Stream Cipher.....	5
2.2 Modern Symmetric-Key Cryptography.....	6
2.2 Different Contexts of Hybrid Ciphers.....	8
Chapter Three.....	10
An Overview of Cryptography.....	10
3.1 Introduction.....	10
3.1 Cryptographic goals.....	11
3.2 One-Way Function and Trapdoor One-Way Function.....	11
3.3 Basic terminology and concepts.....	12
3.3.1 For Encryption Domains and Codomains.....	12
3.3.2 For Encryption and Decryption Transformations.....	13
3.3.3 Achieving confidentiality.....	13
3.3.4 For Communication participants.....	14

3.3.5	Security .....	14
3.3.6	Cryptology .....	15
3.3.7	Symmetric-key encryption.....	16
3.3.8	Substitution ciphers and transposition ciphers .....	16
3.3.9	Transposition ciphers.....	18
3.3.10	Product ciphers .....	18
3.3.11	Stream ciphers .....	18
3.3.12	The Vernam cipher .....	19
3.3.13	The Key Space.....	20
3.3.14	Digital Signature .....	20
3.3.15	Authentication and identification .....	21
3.3.16	Public-key cryptography .....	21
3.3.17	Public-key Encryption .....	21
3.3.18	The need for authentication in Public-key cryptosystems .....	23
3.3.19	Digital signature arising from public-key encryption.....	24
3.3.20	Construction needed for a digital signature scheme.....	24
3.3.21	Comparison between Symmetric-key Cryptography and Public-key Cryptography ...	27
3.3.22	Hash Functions .....	31
Chapter Four .....		33
Classical Ciphers.....		33
4.1	Examples of Mono-alphabetic substitution .....	33
	Example 4.1.1 (Shift Cipher) .....	33
	Example 4.1.2 (Affine Cipher).....	34
4.2	Examples of Polygram substitution.....	34
	Example 4.2.1 (Playfair Cipher).....	35
	Example 4.2.2 (Hill Cipher).....	35
4.3	Examples of Polyalphabetic substitution.....	35
	Example 4.3.1 (Simple Vigenere Cipher).....	36
	Example 4.3.2 (Beaufort variant of Vigenere Cipher) .....	36
	Example 4.3.3 (Compound Vigenere Cipher) .....	36
	Example 4.3.4 (Running-key Vigenere Cipher).....	36
	Example 4.3.5 (Auto-key Vigenere Cipher) .....	37
	Example 4.3.6 (Vigenere Cipher viewed as Vernam Cipher).....	37
Chapter Five .....		38

Design of Hybrid Cipher .....	38
5.1 Description of Hybrid Cipher .....	38
5.2 Encryption in Hybrid Cipher .....	39
5.3 The Cipher Key.....	39
5.4 Decryption of Hybrid Cipher.....	39
5.5 Implementation Issues .....	40
Chapter Six .....	41
Summary, Conclusion and Recommendations .....	41
6.1 Summary .....	41
6.2 Conclusion.....	41
6.3 Recommendations .....	41
References .....	43



# Chapter One

## General Introduction

### 1.1 Introduction

Cryptography is all about sending messages, such that only the targeted recipient can read the message. The idea of cryptography is essentially secret communication between two or more entities (called legitimate sender(s) and legitimate receiver(s)) and an intruder who tries to know part of the information or all of the information being communicated. Hence, cryptography is about communication in the presence of an intruder. So, in cryptography messages are disguised by the legitimate sender so that the intruder, after intercepting the disguised message, cannot recover the original message; but the legitimate receiver can remove the disguise and recover the original message. Cryptography can be traced to the Romans, when Caesar used it to disguise communicated messages between himself and his generals at battlefield. In disguising a message and removing the disguise mathematical techniques are usually employed. According to Mollin (2007), *Cryptography* is the study of methods for sending messages in *secret* (namely, in *enciphered* or *disguised* form) so that only the intended recipient can remove the disguise and read the message (or *decipher* it).

Similarly, Menezes *et al.* (1996) defined *Cryptography* as the study of mathematical techniques related to aspects of information security such as confidentiality, data integrity, entity authentication, and data origin authentication.

These two definitions are equivalent. Since in trying to ensure only the intended recipient of the message reads it, concepts like: data integrity, authentication, etc. surfaced.

Another concept, closely related to Cryptography, is *steganography*, which is “covert secret writing”, where the message being sent is totally hidden from any intruder. Example of *steganography* is *invisible*. Mollin (2007)

The origin of the word Cryptography is from the Greek word *kryptos*, meaning *hidden*, and *graphein*, meaning *to write*, Mollin (2007). Cryptography was started by the Egyptians about 4000 years, where it was used in limited form.

Before proliferation of computers and communications systems, cryptography was an art predominantly practiced by those associated with the military, the diplomatic service and government in general, where it was used to protect national secrets and strategies. However, the proliferation of computers and communications systems in the 1960s brought with it a demand from the private sector for means to protect information in digital form and to provide security services.

In 1977 a U.S. Federal Information Processing Standard (FIPS) for encrypting unclassified information, the Data Encryption Standard (DES), was adopted. DES was later considered insecure, and it has been replaced by the Advanced Encryption Standard (AES), in the year 2001 as the U.S. Federal Information Processing Standard for encrypting unclassified information. AES remains the standard means for securing electronic commerce for many financial institutions around the world. Menezes *et al.* (1996).

In recent years, various issues related to cryptography, like using biologically-inspired computational optimization techniques for cryptanalysis on one hand; and encryption of images on the other hand, surfaced (see Uddin and Youssef (2006), Ragheb and

Subbanagouder (2008), and Chandran and Rajesh (2013)).

## **1.2 Statement of the Problem**

To design a hybrid cipher which would be free from the weakness inherent with the polyalphabetic substitution.

## **1.3 Justification**

Currently, among the existing models of ciphers, it is known that monoalphabetic substitution is easily cryptanalysed because of the redundancy of natural language, and a Polyalphabetic substitution which is apparently more secure than monoalphabetic substitution, can also be cryptanalysed without much difficulty. In fact, once the block length in Polyalphabetic substitution is found, by methods such as that of Kasiski's (see cf. Menezes *et al.* 1996), the individual blocks are cryptanalysed just like in monoalphabetic substitution. In view of these flaws construction of some new designs for a cipher that would be resistant against successful attacks on polyalpabetic substitution could be an interesting piece of research.

## **1.4 Aim and Objectives**

The aim of this work is to give an overview of cryptography with emphasis on classical cryptography and a description of a new cipher which will be called *hybrid cipher*. The following are the objectives of this research:

- i. Carrying out a systematic study of classical cryptography and identify deficiencies in classical ciphers
- ii. Defining a new cipher which would be stronger vis-à-vis classical ciphers

## **1.5 Methodology**

A critical review of classical cryptography will be undertaken, the deficiencies of polyalphabetic substitution cipher will be identified, and finally a new cipher will be outlined which would be free from such deficiencies.

## **1.6 Organization of thesis**

Besides chapter one, this thesis consists of remaining five chapters. Chapter two contains the literature review, chapter three consists of an overview of cryptography, chapter four contains the presentation of some classical ciphers, chapter five contains the design of the proposed hybrid cipher, and finally chapter six has the summary, conclusion and recommendations.

## Chapter Two

### Literature Review

#### 2.1 Introduction

Cryptography can be regarded to be as old as when man began to communicate in secrecy. As it may be a difficult task to review in detail all the available literature on Cryptography, only a brief review is presented in the thesis.

Cryptography can be divided as either public-key or symmetric-key. Since this work focuses on symmetric-key cryptography, the literature review is largely on symmetric-key cryptography.

#### 2.2 Classical Block and Stream Cipher

First we review literatures on classical ciphers. Kahn (1967) gave a thorough but the non-technical account of the classical ciphers and cryptography in general. He traces cryptography from its initial and limited use by the Egyptians some 4000 years ago, to the twentieth century where it determined the outcomes of both world wars. Shannon (1949) authored an extremely clear paper on classical ciphers. Some standard references for classical cryptanalysis are Friedman (1944), Gaines (1956), and Sinkov (2009). More recently we have books providing expository material on classical ciphers, machines, and cryptanalytic examples. These include Beker and Piper (1982), Meyer and Matyas (1982), Denning (1983), and Davies and Price (1989), among many others.

After cryptanalysis of the classical ciphers, a lot of researches began the automation of the cryptanalysis of these classical ciphers. This led to bringing in of biologically-inspired computational optimization techniques in cryptography. Jokobsen (1995) used a fast

algorithm for cryptanalysis of both mono- and polyalphabetic where an initial key is guessed and refined through a number of iterations. Jokobsen also discovered that the only information necessary to break the cipher is the knowledge of the digram distribution of the ciphertext and the expected digram distribution of the plaintext.

Uddin and Youssef (2006) explored the use of Particle Swarm Optimization (PSO) in automated cryptanalysis of classical simple substitution ciphers. Their experimental results showed that PSO-based attacks are very effective on various sets of encoding keys.

Ragheb and Subbanagouder (2008) presented the Cryptanalysis of polyalphabetic cipher by applying Genetic Algorithms (GA). They used GA for searching the key space of encryption scheme. Frequency analysis was used as an essential factor in the objective function of the GA and the GA was then used to find the key size in Vigenere cipher.

## **2.2 Modern Symmetric-Key Cryptography**

In recent years, cryptographic encryption is used to encrypt images. Chandran and Rajesh (2013) increased the transmission speed and provided a more secure communication when an image was transmitted through a network. For some application scenarios, the network operator providing the resource for the transmission channel, needs to be prevented from having access to the transmitted data between the sender and the receiver.

That is, the sender encrypts the original data and the network provider compresses the encrypted data without any knowledge of the cryptographic key and the original data. At the receiver's side, decryption functions will be applied to the decompressed data to

reconstruct the original image. Encryption gives secure communication through the network while compression reduces the size of the image and therefore increases the transmission speed. They proposed a chaos based encryption. Chaotic systems are sensitive to initial conditions and mixing property. They used two chaotic maps, Quadratic and tent maps to produce the chaotic sequence used to control the encryption process. The bitwise XOR operation was used to create the encrypted image.

In wireless sensor network, the issue of constraint memory is a big challenge. When wireless medical sensor network used, the sensitive patient data is transmitted through the open air. It is more vulnerable to eavesdropping and other attacks, compared with the wired network. The work that has been done so far can protect the patient data during transmission, but cannot stop the inside attack where the administrator of the patient database can see the sensitive patient data. Xun *et al.* (2013) proposed a practical approach to prevent the inside attack by using *sharemind system*, developed by Cybernetica to perform computations on input data without compromising its privacy. Their work has two main contributions. One contribution is proposing a lightweight encryption algorithm to protect the communication between the sensor node and the sharemind system. Another contribution is employing sharemind system to protect patient data privacy and support medical research.

Similarly, Mobile ad hoc networks (MANETs) are memory constraint. MANETs are now very attractive area of research and security is the most challenging point that they undergo. Cryptography is used to provide security within MANETs. However, storing all keys in every node, even if practically possible, is inefficient for large scale MANETs due to memory or processing capability limitations. Gharib *et al.* (2013) presented several

different practical scenarios for key selection, and the impact of each scenario on the performance and security metrics is analyzed. Their results showed that the proposed scenarios can reduce the path length in addition to keeping the network highly connected.

Based on the aforesaid review, we in this thesis, proposed to design a hybrid cipher which would be free from the weakness inherent with the polyalphabetic substitution.

## **2.2 Different Contexts of Hybrid Ciphers**

The most common Hybrid Cipher is what is popularly known as Hybrid Cryptosystem. Hybrid Cryptosystem is common in most fundamental literatures of cryptography like Menezes *et al.* (1996) and Mollin (2007) among others. Hybrid Cryptosystem combines the advantages of both symmetric-key and public-key encryptions. This scheme uses the receiver's public key to encrypt the symmetric key used in encrypting the message. Both the encrypted message and the encrypted symmetric key are thereafter transmitted to the receiver. Upon receiving the message, the receiver first decrypts the encrypted symmetric key and subsequently uses the recovered key to get back the message.

Subasree and Sakhtive (2010) use hybrid cryptosystem to improve communication in computer network. Unlike the hybrid system described above where the target is confidentiality, Subasree and Sakhtive considered data integrity and authentication in addition to confidentiality. They used Elliptic Curve Cryptography for confidentiality, Dual-RSA for authentication, and Message Digest (MD-5) for integrity.

Aghajanzadeh *et al* (2013) developed a hybrid cipher by combining the features of AES, RC4, and SERPENT. Their work is aimed at bringing together the advantages of the three ciphers into a single cipher; the advantages of AES and SERPENT are security

while the advantage of RC4 is its speed. So, the hybrid cipher they developed is faster than AES and resistant to many attacks.

## Chapter Three

### An Overview of Cryptography

#### 3.1 Introduction

Cryptography is all about having secure communications in the presence of an intruder.

Before the advent of computers, cryptography was practiced only by the military, diplomats and the secret services. When computers became ubiquitous in the 1970s, the need to protect information in digital form significantly widened the application of cryptography.

By 1977, National Institute of Standards and Tehnology (NIST) of the United States adopted a standard for encrypting unclassified information. This standard which is called Data Encryption Standard (DES) was designed by IBM. DES is a Feistel cipher which is a symmetric-key block cipher having a block length and key length of 64 bits. DES uses a set of permutation depending on a key to encrypt plaintext and to decipher ciphertext (NIST 1977). Although DES was in use for about a quarter of a century, improvements on mathematical algorithm and development of faster computers made DES unable withstand the continued cryptanalytic onslaught. Therefore, it became weak and hence its replacement was very much needed.

NIST called for proposal for Advanced Encryption Standard (AES) to replace DES and finally *Rijndael*, named after its designers Vincent **Rijmen** Joan **Daemen**, was selected as replacement of DES in 2001 (Mollin 2007).

AES process data as 128-bit block using key length of 128, 192, and 256 bits. The data and the key are manipulated as matrices of byte.

### 3.1 Cryptographic goals

The following, mostly abstracted from Menezes *et al.* (1996), are considered the most important information security objectives are: (1) privacy/confidentiality; (2) data integrity; (3) authentication; and (4) non-repudiation

1. *Privacy or confidentiality* aims at ensuring that only those entities that are legitimately allowed to have access to information do have such access.
2. *Data integrity* is a service that prevents the illegitimate modification of a communicated message.
3. *Authentication* and identification are concepts that go together in cryptography. In a communication, the identity of the communication participants needs to be confirmed, and the communicated message needs also to be authentic-that is the message received by the receiver has not been altered while in transit.
4. *Non-repudiation* is a service that applies to the sender of a message. This service ensures that the sender of a message cannot deny sending the (sent) message.

A primary aim in cryptography is to sufficiently address the aforementioned goals.

Tools or primitives employed to ensure achievement of cryptographic goals include, but not limited to the following, encryption schemes, digital signature schemes, and hash functions.

### 3.2 One-Way Function and Trapdoor One-Way Function

Trapdoor one-way function is very crucial in, specifically, public-key cryptography. However, it is necessary to understand one-way function before discussing trapdoor one-way function.

### **Definition 3.1 (One-way function)**

A function is called a *one-way function* if it is easy to compute the image of every element in domain of the function but computationally infeasible to invert almost every element in the range of the function.

### **Definition 3.2 (Trapdoor One-way function)**

A *one-way function*, whereby given special information (called trapdoor) it becomes computationally feasible to invert every element in the range of the function, is called a *trapdoor one-way function*.

## **3.3 Basic terminology and concepts**

The following are some basic concepts and terminologies used in cryptography (mostly abstracted from Menezes *et al.* 1996):

### **3.3.1 For Encryption Domains and Codomains**

1. *Alphabet of Definition*: A finite set, such that every element or character of a message is represented by the elements of this set, is called the alphabet of definition. The alphabet of definition is usually denoted by  $A$ . An example is the binary alphabet whereby every ASCII character can be represented by strings of zeros and ones.
2. *Message space*: This is the set of all possible strings of symbol from  $A$ , that is, the collection of all possible messages called plaintext. The message space is usually denoted by  $M$ .
3. *Ciphertext space*: The *ciphertext space*, denoted by  $C$ , consists of strings of symbols or messages. Like the message space, ciphertext space is also from an

underlying alphabet of definition, which may or may not be the same as the alphabet of definition of the message space. The ciphertext space consists of elements called *ciphertexts* or encrypted messages.

### 3.3.2 For Encryption and Decryption Transformations

1. The key space is the collection of all elements called keys. It is usually denoted by  $K$ .
2. An *encryption transformation* is a bijective map from  $M$  to  $C$ . There are several such encryptions and each is determined by a key  $e$  in  $K$  and denoted by  $E_e$ .
3. The corresponding inverse map of any encryption transformation  $E_e$  is called a *decryption transformation*. The decryption transformation is usually denoted by  $D_d$ , where  $d$  is in  $K$
4. An *encryption scheme* is the set of encryption transformations  $\{E_e : e \in K\}$  together with a parallel set of decryption transformations  $\{D_d : d \in K\}$  such that for each encryption transformation there exists a corresponding decryption transformation (which is the inverse of the encryption transformation).

### 3.3.3 Achieving confidentiality

To have a secure communication, an encryption scheme is used in the following manner:

The sender (referred to as Alice) and the receiver (referred to as Bob) initially and confidentially obtained encryption and decryption keys. Thereafter, Alice sends an encrypted message (ciphertext), encrypted by the encryption transformation determined by the encryption key, to Bob. Upon receiving the ciphertext, Bob decrypts the ciphertext with the decryption transformation determined the decryption key to get back the plaintext.

Encryption and decryption functions are not simply use without keys in an encryption scheme

because, if any encryption/decryption key pair is compromised, it is the key pair that will be changed and not the whole encryption scheme.

### 3.3.4 For Communication participants

1. In a communication, the legitimate transmitter of a message, the legitimate receiver of the transmitted message and the intruder are called parties or entities.
2. The actual originator of a message is called a *sender* (referred to as Alice).
3. The targeted recipient of a message is called the *receiver* (referred to as Bob).
4. The illegitimate party that tries to participate in a communication is called an *adversary*. (Since the sender and receiver are referred to Alice and Bob, the adversary is also referred to as Eve).
5. A *channel* is the link connecting communication participants by which information is transmitted among the participants.
6. A channel that is physically protected against the intruder is called a *secure channel* or *physically secure channel*.
7. If an intruder can manipulate information in a channel, then that channel is called an *unsecured channel*.
8. A *secured channel* is a channel whereby the intruder cannot manipulate the information being transmitted in the channel.

A secured channel can either be secured physically or by cryptographic means.

### 3.3.5 Security

In cryptography, the sets  $M$ ,  $C$ ,  $K$ ,  $\{E_e : e \in K\}$ , and  $\{D_d : d \in K\}$ , defined above, are usually not kept secret. However, the encryption and/or the decryption key(s) must be kept secret.

Although additional security can be obtained by keeping the sets of encryption and decryption transformation secret, the security of the encryption scheme should not be based on the secrecy of these sets of transformations, as such a security can easily be defeated.

**Definition 3.3 (Breakable Encryption Scheme)**

If in an encryption scheme, the plaintext can be retrieved from only the ciphertext in a reasonable amount of time, then such an encryption scheme is called *breakable*.

A “reasonable amount time” means the time needed for the information to remain secret. While paying for goods requires the payment information to remain secret until the transaction is completed, state secrets may on the other hand be needed to remain secret indefinitely.

**3.3.6 Cryptology**

1. The study of breaking encryption schemes and other cryptographic services is called *Cryptanalysis*
2. A *cryptanalyst* is a practitioner of cryptanalysis.
3. Combining the study of both cryptography and cryptanalysis is termed as *Cryptology*.
4. A *cryptosystem* is a collection of cryptographic algorithms used mostly for achieving confidentially or any information security service.

Cryptographic techniques are divided into two main types: *symmetric-key* and *public-key*. Encryption methods of these types will be discussed separately below.

### 3.3.7 Symmetric-key encryption

Encryption schemes (both public and symmetric key schemes) are divided into block and stream ciphers.

#### Definition 3.4 (Symmetric-key Encryption)

An encryption scheme that uses essentially the same key for encryption and decryption is called *symmetric-key* encryption.

In some cases the encryption and decryption are equal; while in other instances it is easy to compute the encryption key from the decryption key or vice versa.

#### Definition 3.5 (Block Cipher)

A *block cipher* is an encryption scheme that encrypts each fixed-length block of the plaintext with a constant encryption transformation.

Examples of standard block ciphers include DES and AES. Block ciphers are usually divided into two important classes: *substitution ciphers* and *transposition ciphers*.

### 3.3.8 Substitution ciphers and transposition ciphers

Substitution ciphers are block ciphers which acts by replacing symbols or set of symbols by other symbols or set of symbols employing a permutation defined on the alphabet of definition.

#### Definition 3.6 (Simple Substitution Ciphers)

A simple substitution cipher substitutes symbols or set of symbols usually with different symbols or set of symbols employing a permutation defined on the alphabet of definition.

**Definition 3.7 (Polyalphabetic substitution ciphers)**

A *polyalphabetic substitution cipher*, unlike a simple substitution cipher, encrypts characters of plaintext using different permutations that are defined on the alphabet of definition.

**Example 3.4 (Vigenère Cipher)**

Vigenère Cipher is a famous example of polyalphabetic substitution cipher. The cipher is defined thus: Suppose we have three permutations (defined on the English alphabet);  $f_1, f_2$  and  $f_3$ ; defined as follows;  $f_1$  replaces each letter of the plaintext with a letter that is two positions ahead of the plaintext letter,  $f_2$  replaces each letter of the plaintext with a letter that is five positions ahead of the plaintext letter, and  $f_3$  replaces each letter of the plaintext with a letter that is six positions ahead of the plaintext letter.

Therefore, the following plaintext:

$m =$  THE TER ROR IST WIL LAT TAC KAT NIG HT

is encrypted as:

$c = E_e(m) =$  VMK VJX PTX KXZ YNR NFZ VFI MFZ PNM JY

Polyalphabetic ciphers, although not too difficult to cryptanalyse, are more difficult to cryptanalyze than monoalphabetic substitution, because, unlike monoalphabetic substitution, Polyalphabetic ciphers do not preserve the ciphertext character of a particular plaintext character, if the plaintext character is encrypted more than once.

### 3.3.9 Transposition ciphers

Transposition ciphers can be regarded as simplified version of substitution ciphers, in that they may maintain the set of characters in the plaintext block.

#### Definition 3.8 (Transposition ciphers)

A transposition cipher is a block cipher whereby each block of the ciphertext is a permutation of the same characters in the corresponding plaintext block.

Therefore, transposition cipher is, in fact, easier to cryptanalyze than substitution cipher.

### 3.3.10 Product ciphers

Because simple substitution and transposition are on their own not very difficult to cryptanalyze, they are usually combined to form strong ciphers called product ciphers.

The number of times substitution and transposition ciphers are combined is called a *round*. A good example of product cipher is DES.

#### Remark 3.1 (*confusion and diffusion*)

In a round, substitution is regarded as contributing *confusion* while transposition is regarded as contributing *diffusion* to the encryption process. Confusion is aimed at hiding any connection between the key and the ciphertext. On the other hand, diffusion is aimed at not carrying over any redundancy present in the plaintext to the ciphertext.

### 3.3.11 Stream ciphers

Encryption schemes are divided into block and stream ciphers. The above discussion focuses on block symmetric-key encryption, and the focus will now be shifted to stream symmetric-key encryption. In contrast to block cipher, where blocks of the plaintext are the entities

processed during encryption; stream ciphers encrypt one character of the plaintext at a time.

**Definition 3.9 (Key Stream)**

A collection of keys from the key space used in the encryption of a plaintext message is called a *keystream*.

**Definition 3.10 (Stream Cipher)**

Suppose  $m_1 m_2 m_3 \dots$ ,  $e_1 e_2 e_3 \dots$ , and  $E e_1 E e_2 E e_3 \dots$  be respectively the plaintext, a keystream, and the respective encryption transformations determined by the keystream. A *stream cipher* encrypts each of the plaintext characters with corresponding encryption transformation to get the ciphertext characters. That is  $E_{e_j}(m_j) = c_j$ , for  $j=1,2,3\dots$

**3.3.12 The Vernam cipher**

A well known, simple stream cipher is the vernam cipher.

**Definition 3.11 (Vernam Cipher)**

Let  $m_1 m_2 \dots m_n$  be a binary plaintext message and  $k_1 k_2 \dots k_n$  be a binary key, then the *Vernam Cipher* is defined as  $m_i \oplus k_i = c_i$ , for  $i = 1,2,3,\dots,n$ , and the  $c_i$ 's are the resulting ciphertext bits and  $\oplus$  is the XOR operation between bits.

The *one-time pad* which is the only proven unbreakable cipher is built upon the one-time pad.

A Vernam cipher in which the key is selected randomly and used only once becomes a one-time pad.

### **3.3.13 The Key Space**

The number of all possible encryption or decryption keys in an encryption scheme is termed the key space.

For any secure encryption system, the key space must be very massive so as it becomes infeasible to exhaustively search it.

### **3.3.14 Digital Signature**

Traditional signature is a piece of writing used by any individual to attach his/her identity to a document. Therefore, the individual can authorize any party to perform some function, and the authorized party can verify (authenticate) the signature, and also the individual giving the authority cannot at a later time deny issuing such an authority (non-repudiation), assuming the signature was not forged.

Similarly, a digital signature is aimed at performing the function(s) of traditional signature in digital sense. Hence a digital signature is a mathematical procedure by which a party attaches its identity to a piece of information originating from that party; so that authorization, authentication, and non-repudiation can be achieved.

Just like in traditional signature scheme, an individual tries to make his/her signature difficult to forge; also a digital signature ought to be computationally infeasible to forge.

However, no digital signature scheme has been proved to be computational infeasible to forge.

### **3.3.15 Authentication and identification**

Authentication implies a means by which entities can be verified whether they are whom they claimed to be, or that a piece of information has not been altered with by unauthorized entities. Authentication is among the most important security objectives. Although authentication and confidentiality are apparently connected, they are indeed two separate security objectives. Authentication is divided into types, namely, *entity authentication*, which is also called *identification*, and *data origin authentication*.

#### **Definition 3.11 (Entity Authentication or Identification)**

Entity authentication or identification is a procedure, in a two-party communication, by which one of the communicating parties confirms that the other party is actually the legitimate communication participant.

In a communication system, Identification could either be unilateral or mutual.

#### **Definition 3.12 (Data Origin Authentication or Message Authentication)**

Data origin authentication or message authentication is a procedure by which the receiver of a message confirms the identity of the originator of the message.

### **3.3.16 Public-key cryptography**

Public-key cryptography is a revolutionary concept that changed the way cryptography is done.

### **3.3.17 Public-key Encryption**

Unlike in symmetric-key encryption, where the encryption key equals the decryption key or it is easy to obtain the decryption key from the encryption key; in public-key

cryptography it is computationally infeasible to obtain the decryption key from the encryption key.

In symmetric-key cryptography, Alice and Bob need to know the encryption and decryption keys that are needed to be kept secret, and the keys must be exchanged between Alice and Bob through some secure means like over a secure channel.

By contrast, in public-key cryptography, Bob decides on encryption and decryption keys and send the encryption key (public key) over an unsecure channel while keeping the decryption key (private key) secret. Thereafter, whenever Alice wants to send a message to Bob, she encrypts the message using Bob's public key. Upon receiving the encrypted message, Bob uses the private key known only by him to decrypt the message.

An analogy to public key cryptography is a box with number lock, where the combination of the numbers is secretly generated by Bob. Whenever the box is open, any individual wanting to place anything in the box can do so; but once the box is closed using the number lock, only Bob can then open the box to retrieve items in the box.

**Definition 3.13 (Public-Key Encryption)**

Given an encryption scheme with a class of encryption functions,  $\{E_e: e \in K\}$ , a class of decryption functions,  $\{D_d: d \in K\}$ , and a key pair,  $(e, d)$ ; then such an encryption scheme is called a public key encryption, if the public key,  $e$ , is publicly available; and the private key,  $d$ , is kept secret and moreover it is computationally infeasible to obtain  $d$  from  $e$ .

In public key cryptography,  $e$  is termed a private key and in symmetric-key

cryptography it is termed a secret key. This is because privacy involves only a single entity, whereas secrecy is shared by two or more entities.

The security of public key cryptosystems is based on the difficulty of some mathematical problems. However, apart from the one-time pad, there is no cryptosystem, symmetric-key or public-key, which has been proven to be unconditionally secure.

### **3.3.18 The need for authentication in Public-key cryptosystems**

Public-key cryptography is apparently appealing, in that Alice only need to encrypt a message with Bob's publicly available public key and send the encrypted message to Bob, who decrypt the ciphertext with his private key. A closer examination reveals a loop hole, if public-key cryptography is used as described above. This is because since the communication channel is unsecured, an adversary can claim to be Bob and therefore transmits his (adversary's) public key to Alice, and Alice will mistakenly think the received public key is Bob's. When Alice send a message with this key, the adversary intercepts the message, decrypt the message with his private key, and thereafter re-encrypt the message with Bob's public key before sending the message to Bob.

In this scenario, Alice thought she was communicating with Bob, and similarly, Bob assumed he was communicating with Alice; whereas, both Alice and Bob actually communicated with the adversary.

This situation is called an *impersonation attack* resulting in *protocol failure*. In order to remedy impersonation attack, keys must be authenticated so as to know whom they originated from.

### 3.3.19 Digital signature arising from public-key encryption

There is a special set of digital signature schemes that arise from public-key cryptography. This type of digital signature schemes make use of the parameters of the public-key encryption, and the only new parameter introduced in these schemes is the verification transformation. For these digital signatures, the set of messages to be signed, the plaintext space and the ciphertext space are all equal. Therefore, any plaintext message is also contained in the ciphertext space, so the result of first applying the encryption function and then applying the decryption function is the same as that of first applying decryption function and thereafter applying the encryption function.

### 3.3.20 Construction needed for a digital signature scheme

1. Let  $M$  and  $C$  be respectively the message space and the ciphertext space for the public-key encryption. Let  $S$  be the signature space. Then  $S = C = M$ .
2. Suppose  $e$  is any key which determines the encryption transformation,  $E_e$ , and  $d$  be the corresponding decryption key which determines the decryption transformation,  $D_d$ . Suppose also that  $S_A$  is the signing transformation for any entity  $A$ . Then  $S_A = D_d$  (since for digital signature  $S_A$  needs to be kept secret and similarly for public-key encryption  $D_d$  needs also to be kept secret). Therefore,  $D_d(m) = S_A(m) = s$ , where  $m$  is in  $M$  and  $s$  is a signature on  $m$ .
3. If  $S_A(m) = s$ , that is  $D_d(m) = s$ , then  $E_e(D_d(m)) = E_e(s)$ . But  $E_e(D_d(m)) = m$ . Hence  $E_e(s) = m$ . Therefore, the verification transformation for  $A$ ,  $V_A$ , is defined as

$$V_A(m, s) = \text{true, if } E_e(s) = m \text{ and } V_A(m, s) = \text{false, otherwise.}$$

A simplification is usually further used when signing; that is the signer does not sign all messages but rather only sign those messages with particular features that are publicly known.

The digital signature described above is termed *digital signatures with message recovery*, and signing only messages with particular features is called selecting messages with *redundancy*.

Selecting messages with redundancy, apart from being a simplification, also has security advantage, because it minimises the chance of forging *A*'s signature.

Digital signatures arising from reversible public-key encryption are appealing, but they have a major drawback of requiring an encryption method as a primitive. However, in some instances, a digital signature scheme devoid of encryption is required.

### **Digital Signatures used in practice**

To make digital signatures practically useful, additional properties are required. These properties are:

1. Ease in computing the signature.
2. Ease in verification of any signature on a message.
3. Computationally forgery-proof for the time necessary for the signature to be secure.

### **Disputes Resolution**

Traditional signatures help in resolving disputes between individuals. An individual may deny signing a particular document, or that another individual may falsely claim someone has signed a document. To resolve such a dispute, there is need to confirm whether the disputed signature in each of the above cases is authentic.

A similar scenario also occurs with digital signature. In fact, the aim of digital signature is to serve the purpose of traditional signature.

When there is dispute between  $A$  and  $B$ , either when  $A$  falsely claims not to have sign a message he already signs, or that  $B$  falsely claims that a message has been signed by  $A$ . To resolve any of these disputes, the service of what is called *trusted third party* (TTP) is required. The TTP as his/her name implies is trusted by both  $A$  and  $B$ . If  $A$  falsely claims not to have sign a message she already signs, the TTP will confirm that the signature on the signed message is  $A$ 's using the verification function. Similarly, if  $B$  falsely claims that a message has been signed by  $A$ , the TTP will also confirm that signature on the message is not  $A$ 's using the verification function.

If  $B$  is sure the verification used by the TTP is the same at that of  $A$ , he accepts the TTP's decision. Also,  $A$  agrees with the TTP's resolution if she is certain the TTP used her verification function and moreover her signing function has not been compromised.

Hence, in order to have fair resolution, the following conditions should be satisfied:

### **Requirements for resolving disputes involving digital signatures**

1. The signature on any message should be computationally infeasible to forge.
2. The TTP must use the authentic verification function.
3. The signing function must remain securely undisclosed to other entities.

Although these properties are required for digital signature schemes to be secure, they may not be achievable in practice. One reason is that, for some signature schemes forging signatures is always possible. In addition,  $A$  may not be sincere about the signing function

being compromised. Overcoming these issues involves the use of a time frame for which A must not deny that the verification has been compromised. A similitude to this situation is the act of credit card revocation, where as long as the card has not been reported missing to the bank by the owner of the card, he/she is considered responsible for any transaction performed with the card.

### **3.3.21 Comparison between Symmetric-key Cryptography and Public-key Cryptography**

Both symmetric-key and public-key cryptography have their advantages and disadvantages. The following is a list of some of these advantages and disadvantages.

#### **Advantages of Symmetric-key Cryptography**

- i. Fast. Symmetry-key encryptions are faster than public-key encryptions.
- ii. Short key length. Symmetry-key cryptography uses shorter key length than the corresponding public-key cryptography
- iii. Several uses. Symmetric-key ciphers have several uses, like in construction of hash functions, digital signatures and so on.
- iv. Make use of simple components. Strong and complex symmetric-key ciphers like AES are constructed form simple and weak substitution and transposition ciphers.
- v. Long history. Being the first known encryption schemes, symmetric-key encryptions have an extensive history, both before and after the advent of the digital computer.

## **Disadvantages of Symmetric-key Cryptography**

- i. Both the encryption and decryption keys must be kept secret.
- ii. There is large number of keys to be managed. The size of the keys grows with the growth of the communication network. As such an unconditional TTP is required for good key management.
- iii. Frequent changing of key. It is considered a good cryptographic practice to frequently change keys in symmetric-key cryptography.
- iv. Additional requirement needed in digital signature usage. Digital signature schemes created with symmetric-key encryptions either require the use of a TTP or public verification function with large keys.

## **Advantages of Public-key Cryptography**

- i. Only one key need to be kept secret. The private key is the only key that must be kept secret, while the encryption (public) key is publicly available, but this public key must be authenticated.
- ii. Administration of keys requires only a functionally TTP. Unlike when using symmetric-key encryption on a network, where the administration of keys requires the service of an unconditional TTP, public-key cryptography needs just the presence of a functionally TTP.
- iii. No need to frequently change keys. In public-key cryptography, the encryption/decryption key pair can be used for a substantial period of time as opposed to symmetric-key encryption where it is important to change keys often.

- iv. Give rise to efficient digital signatures. A number of public-key schemes yield comparatively, computationally efficient digital signatures schemes having verification function with shorter key length than for symmetric-key encryption.
- v. Less difficult key management problem. The problem of key management is relatively easier, as there are fewer keys to manage than in symmetric-key encryptions.

### **Disadvantages of Public-key Cryptography**

- i. Slow. No public-key encryption is as fast as symmetric-key encryption.
- ii. Long key length. Significantly large key sizes are needed for public encryptions in comparison of the key sizes required for symmetric-key encryptions.
- iii. Not proven to be absolutely secure. Unlike symmetric-key encryption, where the one-time pad has been proven to be unconditionally secure, no public-key encryption has been proven to be secure. The security of the most efficient public-key encryptions is based on the difficulty of some mathematical problems
- iv. Does not have a long history. Public-key cryptography does not have an extensive history like symmetric-key cryptography. Public key cryptography actually started in the 70's.

### **Remark 3.2 (Conclusion of Comparison between Public-key and Symmetric-key Cryptography)**

A deep analysis of the advantages and disadvantages of public-key and symmetric-key

encryptions shows that the two encryptions methods have complementary advantages. As a result, current cryptographic systems employ the advantages of both worlds.

In this current cryptographic technique, public-key encryption is used to exchange the key between the entities *A* and *B*. Thereafter, the exchanged key will be used in the symmetric-key communication between *A* and *B*.

Since the public/private key pair in public-key encryption can be used for relatively long time without the need of changing the keys, and symmetric-key encryption is computationally efficient, then current cryptographic setting seize these advantages. In addition, the key exchange carried out using the public-key encryption uses a relatively small amount of time compared to the amount of time required for the data encryption carried out using symmetric-key encryption.

In conclusion, the following are what to take note of in practical cryptography:

Public-key encryption is the most appropriate for digital signature and key management applications; on the other hand, symmetric-key encryption is best for data encryption and data integrity applications.

Finally, regarding the comparison of public-key and symmetric-key is a note on the sizes of the keys. Why are larger keys required in public-key encryptions? This is because, while the efficient attacks on symmetric-key encryption are brute-force, public-key encryptions are vulnerable to short-cut attacks. Therefore, in order to have an equivalent level of security in public-key encryption large keys must be used.

### 3.3.22 Hash Functions

Hash Function is one of the most important cryptographic primitives. It is applied to data integrity and digital signatures.

#### Definition 3.14 (Hash Function)

A *hash function* is a map between binary data of arbitrary length to binary data of fixed length.

The images produced by the hash function are termed as hash values. Given an appropriate hash function having a codomain consisting of digital data of fixed length  $n$ , the probability of any input string to be mapped to a particular hash value is  $2^{-n}$ .

The following properties are desirable for any cryptographic hash function:

1. Computationally efficient.
2. Collision-free, that is, it is computationally infeasible for more than one input string to be mapped to the same value.
3. Computationally infeasible to invert any hash value.

Hash functions are applied in digital signatures in the following way:

Instead of signing arbitrarily long messages, only the hash values of the messages are signed using a publicly known hash function.

In digital signatures only the hash value of a long message is usually signed. The message is hashed using a hash function that is publicly available. At the receiving party's end, a received message is hashed and compared with the hash-value whether they are the same.

The process of signing only the hash value of a message saves time and space, because if

the actual message was to be signed, then the message must be broken into appropriate components and thereafter signing each of these components.

It is important for a hash function to be collision-free, because if it were not, the signer could sign a particular message and later claimed to have signed a different message.

For data integrity, hash function is used in the following way: At a particular point in time, the integrity of a computed hash value of a specific input is kept unaltered. Subsequently, to confirm that the input has not been tampered with, the hash value corresponding to that input is recomputed and verified that it is equal to the initial hash value.

When hash functions are used in data integrity as described above, they are called *Modification Detection Codes* (MDCs).

## Chapter Four

### Classical Ciphers

Cryptosystems that were invented before 1950 are usually called *classical ciphers*. Most classical ciphers are either substitution or transposition ciphers. These ciphers have come to be known insecure, however, usually studied for pedagogical reasons (Menezes *et al.* 1996).

Transposition and substitution ciphers have been discussed in the previous chapters. In fact, every cryptosystem can be regarded as a substitution cipher, including transposition ciphers. This is because, in every cryptosystem, the end result is replacement/substitution of any particular symbol with another symbol. The following are examples of some classical ciphers abstracted from Menezes *et al.* (1996) and Mollin (2007).

#### 4.1 Examples of Mono-alphabetic substitution

In a mono-alphabetic or simple substitution cipher, the plaintext space and the ciphertext space are the same and only one single encryption (a permutation) is used for every character of the plaintext.

##### Example 4.1.1 (Shift Cipher)

Let  $A$  be the alphabet of definition. Suppose that we associate all the characters of  $A$  with the numbers 0 to  $(|A| - 1)$ ; that is the first letter of  $A$  is represented by 0, the second by 1, ..., the last one by  $(|A| - 1)$ .

The encryption  $e$  (a permutation), determined by the key,  $k$ , which is an alphabetic shift through  $k$  characters of  $A$ , is defined, for any alphabet,  $m$ , of  $A$ , by

$$e(m) = c = m + k \pmod{|A|}.$$

Decryption is achieved by using  $d(c) = c - k \pmod{|A|} = m$ .

If  $|A| = 26$  and  $k = 3$ , then the shift cipher becomes what is popularly known as the *Caesar's cipher*.

#### **Example 4.1.2 (Affine Cipher)**

Suppose, as in the previous example,  $A$  is an alphabet of definition, which we associate the numbers  $0$  to  $(|A| - 1)$  for each alphabet of  $A$ . Then the affine cipher is defined for each  $m$  in  $A$ , by  $e(m) = c = am + k \pmod{|A|}$ , where  $0 \leq a, k \leq (|A| - 1)$ .

Decryption of the affine cipher is achieved by,  $d(c) = a^{-1}(c - k) \pmod{|A|}$ . It is worthwhile to note that  $a^{-1}$  exists if and only if the greatest common divisor (gcd) of  $a$  and  $|A|$  is  $1$ , that is  $a$  and  $|A|$  are relatively prime.

It can be observed that the shift cipher is a special case of the affine cipher, with a shift cipher being an affine cipher with  $a = 1$ . Therefore, the affine cipher is not more significantly difficult to cryptanalyse than the shift cipher.

## **4.2 Examples of Polygram substitution**

Unlike simple substitution ciphers which substitute a single plaintext letter with another single plaintext letter, polygram substitution replaces groups of characters of plaintext with other groups of characters. For example, sequences of two plaintext characters (*digrams*) may be replaced by other digrams, sequences of three plaintext characters (*trigrams*) are replaced by other trigrams and, in general, sequences of  $n$ -grams plaintext characters are replaced by other  $n$ -grams.

### **Example 4.2.1 (Playfair Cipher)**

A playfair cipher is digram substitution cipher. It is defined by a 25-letter alphabet (with letter *I* and *J* taken as one) arranged in a 5x5 matrix, *K*. From any given plaintext, adjacent plaintext characters are paired. Any particular pair,  $(p_1, p_2)$ , is replaced by  $(c_3, c_4)$  in the following manner. When  $p_1$  and  $p_2$  are neither in the same row nor column, then they define the corners of a submatrix (could be *K* itself), the remaining corners being  $c_3$  and  $c_4$ , with  $c_3$  being in the same column as  $p_1$ . If  $p_1$  and  $p_2$  are in the same row, then  $c_3$  is defined as the character immediately (circularly) to the right of  $p_1$  and  $c_4$  is defined as the character immediately (circularly) to the right of  $p_2$ . However, if  $p_1$  and  $p_2$  are in the same column, then  $c_3$  is defined as the character immediately (circularly) below  $p_1$  and  $c_4$  is defined as the character immediately (circularly) below  $p_2$ . Finally, if  $p_1 = p_2$ , a not commonly used character like *X* is inserted between them and the entire plaintext is then rearranged.

### **Example 4.2.2 (Hill Cipher)**

Hill cipher is an n-gram substitution cipher mapping n-character plaintext to n-character ciphertext using an  $n \times n$  invertible matrix, regarded as the key, *K*. As usual any alphabet of definition, *A*, is associated with the numbers 0 to 25. Therefore, the elements of the ciphertext and plaintext and entries of the key matrices are all numbers between 0 and 25. Any n-character plaintext, *m*, is encrypted thus:  $e(m) = mK = c$ . Decryption is achieved by  $d(c) = cK^{-1} = m$ .

## **4.3 Examples of Polyalphabetic substitution**

In contrast to simple substitution ciphers, polyalphabetic ciphers use different substitutions on different portions of the plaintext. The implication of this is that same characters in the

plaintext will get mapped to different ciphertext characters and thus precluding frequency analysis.

**Example 4.3.1 (Simple Vigenère Cipher)**

A simple Vigenère cipher having a period  $t$ , over an alphabet of  $s$  characters has  $t$ -character key  $k_1 k_2 k_3 \dots k_t$ . A plaintext  $m = m_1 m_2 m_3 \dots$  is mapped to a ciphertext  $c = c_1 c_2 c_3 \dots$ . The map is defined on each character by  $c_i = m_i + k_i \pmod s$ , and the subscript  $i$  in  $k_i$  is taken modulo  $t$ . That is the map is actually defined as  $c_i = (m_i + k_{i \pmod t}) \pmod s$ .

Observe that the shift cipher can be viewed as simple Vigenère cipher with period,  $t = 1$ .

**Example 4.3.2 (Beaufort variant of Vigenère Cipher)**

Beaufort cipher having a period  $t$ , over an alphabet of  $s$  characters has  $t$ -character key  $k_1 k_2 k_3 \dots k_t$ . A plaintext  $m = m_1 m_2 m_3 \dots$  is mapped to a ciphertext  $c = c_1 c_2 c_3 \dots$ . The map is defined on each character by  $c_i = k_i - m_i \pmod s$ , and the subscript  $i$  in  $k_i$  is taken modulo  $t$ . That is the map is actually defined as  $c_i = (k_i - m_{i \pmod t}) \pmod s$ .

Note that this cipher is its own inverse.

**Example 4.3.3 (Compound Vigenère Cipher)**

The compound Vigenère cipher is defined by the map  $c_i = m_i + (k_i^1 + k_i^2 + k_i^3 + \dots + k_i^r) \pmod s$ , where  $m_i$ ,  $c_i$ , and  $k_i^j$  as usual represent the plaintext, ciphertext and key characters respectively. In general each of the keys  $k^j$ ,  $1 \leq j \leq r$ , has distinct period modulo  $t_j$  and the subscript  $i$  in  $k_i^j$  denotes the  $i$ th character of  $k^j$ .

**Example 4.3.4 (Running-key Vigenère Cipher)**

A running-key Vigenère cipher is a Vigenère cipher where the key is as long as the plaintext.

**Example 4.3.5 (Auto-key Vigenère Cipher)**

An Auto-key cipher is a cipher in which the plaintext acts the key after the use of an initial key called *priming key*.

Define a priming key  $k = k_1 k_2 k_3 \dots k_t$  for a running-key cipher over  $s$ -character alphabet. The plaintext characters,  $m_i$ , are encrypted as  $c_i = k_i + m_i \bmod s$  where  $1 \leq i \leq t$ . The simplest case is when  $t = 1$ . If  $i > t$ ,  $c_i = (m_i + m_{i-t}) \bmod s$

**Example 4.3.6 (Vigenère Cipher viewed as Vernam Cipher)**

In a simple Vigenère cipher, if the keystream is randomly chosen, and the plaintext and the key never repeat, then the Vigenère cipher produces an unconditionally secure Vernam cipher, which is a generalisation from binary alphabet to any alphabet.

## Chapter Five

### Design of Hybrid Cipher

As mentioned earlier, Polyalphabetic substitution cipher is a classical cipher that has been found more secure than simple substitution block and simple substitution stream ciphers. In fact, polyalphabetic substitution cipher was thought to be unbreakable for almost 300 years! However, polyalphabetic substitution cipher is now considered insecure and usually presented only for pedagogical reasons (Mollin 2007).

In the following, we intend to propose a cipher that combines features of both polyalphabetic block and stream ciphers and name it *hybrid cipher*.

Essentially, the design of the proposed hybrid cipher involves incorporating different permutations into polyalphabetic substitution cipher.

#### 5.1 Description of Hybrid Cipher

Let  $A$  be the alphabet of definition, for example  $A$  could be the English alphabet.

Suppose we have a plaintext  $m_1, m_2, m_3, \dots, m_n$ . Suppose also the plaintext is divided into blocks each of length  $l$ . The set of all possible permutations on  $A$  is  $|A|!$  and the set of all possible permutations of length  $l$  from  $|A|!$  is  ${}^{|A|!}P_l$  (the set of permutations of  $|A|!$  taken  $l$  at a time). This set is considered as the key space,  $K$ . By the division algorithm (Rosen 1992),  $n = lq + r$ , where  $q$  and  $r$  are unique integers and  $0 \leq r < l$ , the cipher randomly selects  $(q + 1)$  items from  $K$  to encrypt the message.

## 5.2 Encryption in Hybrid Cipher

To encrypt a plaintext  $m_1, m_2, m_3, \dots, m_n$ , we select  $(q + 1)$  items from  $K$  (observe that each of the  $(q + 1)$  items consists of  $l$  distinct permutations) and apply them on the plaintext in the following way: the first item of  $(q + 1)$  items is applied to the first block (first  $l$  elements of the plaintext), the next item from  $(q + 1)$  items is applied to the next block (second  $l$  elements of the plaintext), and so on until  $q$ th item is applied to the  $q$ th block. If  $r = 0$ , then the encryption process is over. However, if  $r \neq 0$ , then we select the  $(q + 1)$ th and from the  $l$  permutations of the  $(q + 1)$ th item, the first  $r$  permutations are selected and used to encrypt the remaining  $r$  elements of the plaintext.

## 5.3 The Cipher Key

The key is  $q$  or  $(q + 1)$  items, each of which consists of  $l$  (block length) permutations, randomly selected from the key space and used for enciphering the plaintext. If the message is a multiple of  $l$ , the key consists of  $q$  items. But if the message is not a multiple of  $l$ , the key consists of  $(q + 1)$  items but only the first  $r$  permutations in the  $(q + 1)$ th item are used.

## 5.4 Decryption of Hybrid Cipher

Decryption is achieved by applying the inverse of the set of permutations on the ciphertext to get back the plaintext.

### Example 5.1 (Hybrid Cipher)

We are going to consider a simple example so as to have a good perspective of the hybrid cipher. Suppose we use  $A$ , the alphabet of definition, as the English alphabet and we have the following plaintext: *Attack at midnight*. Then the length of the message is  $n = 16$  (disregarding the gap between the words of the plaintext). Now suppose, for the sake of

simplicity, we choose a block length,  $l = 3$ . Then the above plaintext splits as:  
*Att|ack|atm|idn|igh|t.*

Thus, by the division algorithm,  $16 = (3)(5) + 1$ , and therefore there will be 5 complete blocks and the sixth block will be incomplete. The key space will consist of  ${}^{126}P_3$ . Since the length of the message (16) is not a multiple of the block length (3), we select 4 sets of 3 permutations from the key space. Suppose under this example, we use the key: (1,2,3; 4,5,6; 7,8,9; 10,11,12; 13,14,15,16; 17). That is, for the first block, the position of first letter  $a$ , which is 0, is added to 1 (mod 26), and the outcome is 1, which is the letter  $B$ ; the second letter,  $t$ , which is 19, is added to 2 (mod 26), and the outcome is 21, which is the letter  $V$ ; and the third letter also is  $t$ , which is also 19, is added to 3 (mod 26), and the outcome is 22, which is the letter  $W$ . A similar process is applied to the second, third, fourth, and the partial fifth block. Therefore, the plaintext *Att|ack|atm|idn|igh|t* becomes the ciphertext *BVW|EHQ|HBV|SOZ|VUW|J.*

To decrypt the ciphertext *BVW|EHQ|HBV|SOZ|VUW|J*, we only need to subtract the elements of the key from the position of each letter of the ciphertext and thereby recover the plaintext.

### 5.5 Implementation Issues

Unlike modern block ciphers such as AES, the hybrid cipher security is not reduced by smaller block length. In fact, the hybrid cipher is expected to be secure with a block length,  $l \geq 2$ . However, the key of the hybrid cipher is as long as that of the plaintext. Moreover, if the length of the plaintext is shorter than the block length, then hybrid cipher becomes a stream cipher.

## Chapter Six

### Summary, Conclusion and Recommendations

#### 6.1 Summary

In this research work a cipher called *hybrid cipher* has been proposed after critically studying the cryptanalysis of polyalphabetic substitution cipher. The weakness of polyalphabetic substitution was rectified in the design of the hybrid cipher. This weakness mentioned was with regard to use of the same set of permutations for encryption of each block. Therefore, once the block length is discovered, by a method like that of Kasiski (see cf. Menezes *et al.* 1996), each block is thereafter cryptanalysed like a monoalphabetic substitution cipher. However, cryptanalysis of Monoalphabetic substitution is possible because of the redundancy of natural language (Menezes *et al.* 1996).

Although the hybrid cipher uses a key that is as long as the plaintext, it has the advantage of being simple compared to other modern block ciphers. Hybrid cipher simply combines stream and block cipher encryption techniques, hence the name.

#### 6.2 Conclusion

In this research work, hybrid cipher was designed. The mathematical formalization of the hybrid cipher was the most challenging part of this research work.

#### 6.3 Recommendations

Hybrid cipher needs to be improved upon and perfected. Having an understanding beyond classical cryptography requires a very broad knowledge cutting across Computer Science,

Mathematics and in some cases Electrical Engineering. The following are some future research directions to be investigated in order to improve the performance of hybrid cipher:

1. Construction of a fast implementation algorithm

If a fast implementation algorithm is constructed, it will allow efficient implementation of the cipher.

2. Reduction of the key length

Generally, a small key length improves the efficiency of a cipher and hence its practicality. Therefore, reducing the length of the key of the hybrid cipher will increase its performance.

## References

- Aghajanzadeh, N., Aghajanzadeh, F., Kargar, H. R. (2013). Developing a new Hybrid Cipher using AES, RC4 and SERPENT for Encryption and Decryption. . *International Journal of Computer Applications*, Vol. 69, No. 8, pp. 53-62.
- Beker, H. and Piper, F. (1982). *Cipher Systems: The Protection of Communications*, John Wiley & Sons, New York
- Chandran, S. and Rajesh, T. (2013). Multi chaos-based image encryption and lossy compression. *International Journal of Management, IT and Engineering*, Vol. 3, No.9, pp. 11-22.
- Daemen, J. and Rijmen, V. (1999). *AES Proposal: Rijndael*, AES Algorithm Submission. Retrieved from <http://www.nist.gov/CryptoToolkit>
- Davies, D.W. and Price, W.L. (1989) *Security for Computer Networks*, JohnWiley&Sons,New York, second Edition.
- Denning, D.E. (1983) *Cryptography and Data Security*, Addison-Wesley, Reading, Massachusetts [Reprinted with corrections].

Friedman, W. (1944) *Military Cryptanalysis*, U.S. Government Printing Office, Washington DC, Volume I – Monoalphabetic substitution systems. Volume II – Simpler varieties of polyalphabetic substitution systems. Volume III Aperiodic substitutions. Volume IV– Transposition systems.

Gaines, H. (1956) *Cryptanalysis: A Study of Ciphers and their Solutions*, Dover Publications, New York.

Gharib, M., Minaei, M., Golkari, M., Movaghar, A, (2013). *Expert key selection impact on the MANETs' performance using probabilistic key management algorithm*. Proceedings of the 6th International Conference on Security of Information and Networks, New York, NY, USA, pp. 347-351.

Jakobsen, T. (1995). *A Fast Method for Cryptanalysis of Substitution Ciphers*. *Cryptologia*, Vol. 19, No.3, pp. 265-274.

Kahn, D. (1967). *The Codebreakers*, Macmillan Publishing Company, New York.

Menezes, A., van Oorschot, P., and Vanstone, S. (1996). *Handbook of Applied Cryptography*, CRC Press, New York.

Meyer, C.H. and Matyas, S.M. (1982) *Cryptography: A New Dimension in Computer Data Security*, JohnWiley&Sons, New York.

Mollin, R. A. (2007). *An Introduction to Cryptography*, Chapman& Hall/CRC, Taylor & Francis Group.

National Institute of Standards and Technology (NIST), Federal Information Processing Standards (FIPS) Publication (2001). *Advanced Encryption Standard (AES)*. Retrieved from <http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf>

National Institute of Standards and Technology (NIST), Federal Information Processing Standards (FIPS) Publication (1977). *Data Encryption Standard (DES)*. Retrieved from <http://csrc.nist.gov/publications/fips/fips46-3/fips46-3.pdf>

Ragheb, T. and Subbanagouder, A. (2008). Applying Genetic Algorithms for Searching Key Space of Polyalphabetic Substitution Ciphers. *International Arab Journal of Information Technology*, Vol. 5, No.1, pp. 87-91.

Rosen, K.H. (1992). *Elementary Number Theory and its Applications*, Addison-Wesley, Reading, Massachusetts, Third Edition.

Shannon, C.E. (1949). Communication theory of secrecy systems. *Bell System Technical Journal*, 28, pp. 656-715.

Sinkov, A. (2009). *Elementary Cryptanalysis: A Mathematical Approach*,  
Mathematical Association of America, USA, Second Edition.

Subasree, S., Sakthive, K. (2010). Design of a New Security Protocol using Hybrid  
Cryptography Algorithms. *International Journal of Research and Reviews in Applied Sciences*, Vol. 2, No. 2, pp. 95-103.

Uddin, M. F. and Youssef A. M. (2006). *Cryptanalysis of Simple Substitution Ciphers  
Using Particle Swarm Optimization*. Evolutionary Computation, 2006. CEC.  
IEEE Congress, Vancouver, pp. 1-29.

Xun, Y., Willemsen, J., Nait-Abdelsselem, F.(2013). *Privacy-Preserving Wireless  
Medical Sensor Network*, [Trust, Security and Privacy in Computing and  
Communications \(TrustCom\), 12th IEEE International Conference](#), Melbourne,  
pp. 118-125.