INTERNAL REVENUE DEPARTMENT (IRD) DATA SYSTEM BY USING BLOWFISH ALGORITHM

PYAE SANDAR WIN

M.C.Sc. **SEPTEMBER 2022**

INTERNAL REVENUE DEPARTMENT (IRD) DATA SYSTEM BY USING BLOWFISH ALGORITHM

By

Pyae Sandar Win

B.C.Sc.

A Dissertation Submitted in Partial Fulfillment of the Requirements for the Degree of

Master of Computer Science

(M.C.Sc.)

University of Computer Studies, Yangon

SEPTEMBER 2022

STATEMENT OF ORIGINALITY

I hereby certify that the work embodied in this thesis is the result of original research and has not been submitted for a higher degree to any other University or Institution.

…… …..…….

Date Pyae Sandar Win

ACKNOWLEDGEMENTS

I would like express my gratitude and thanks to the following persons who supported and contributed directly or indirectly towards the success of the thesis.

Firstly, I would like to express my respectful thanks to **Dr. Mie Mie Khin**, the Rector of the University of Computer Studies, Yangon, for her kind permission to submit this thesis.

I would like to thank course coordinators, **Dr. Si Si Mar Win** and **Dr. Tin Zar Thaw**, Professors, Faculty of Computer Science, the University of Computer Studies, Yangon for their superior suggestions and administrative supports during my academic study.

I am deeply thanks and respect to **Dr. Yu Wai Hlaing**, Lecturer, Faculty of Computer Science, the University of Computer Studies, Yangon, for her valuable advice and inspiring ideas and not only for her supervision of my thesis but also for her understanding, encouragement and above all, moral support that she has given me throughout the thesis.

My gratitude also goes to **Daw Win Lai Lai Bo**, Assistant Lecturer, Department of English, the University of Computer Studies, Yangon, for editing my thesis from the language point of view.

I would like to thank all of the staff, teachers who taught and helped me from University of Computer Studies, Yangon, for their support.

Finally, I would like to offer my heartfelt thanks to my parents who always gave me encourage and emotional support during my thesis.

ABSTRACT

The document storage system has become the most popular for electronic communication in the world. Documents are used in every organization day by day and most of these documents' data need security system. A cryptographic system (or a cipher system) is a method of protecting information and communication through code so only the user for whom the information intended can read it. A cryptographic system consists of keys, algorithms and key management facilities. In this proposed system, Blowfish Algorithm is applied to encrypt and decrypt the document. Blowfish is a variable-length key with sixteenth rounds and with each block sizes of 64 bits. It can encrypt and decrypt the document by selecting the file on symmetric keys. The system is implemented using C# programming language.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

Security is one of the biggest concerns in communications and electronic applications. With the progress in data exchange by electronic system, the need of information security becomes a necessity. Due to growth of multimedia application, security becomes an important issue of communication and storage of data.

There are many reasons why data is important to organizations all over the world. Data security is very important task for the today enterprise. Organizations are legally obliged to protect customer and user's data from being lost or stolen and ending up in the wrong hands.

Nowadays, most of the communication is done by using electronic media. Most of the data are also collected and stored over the server or database across different networks. So there is a need to protect from various access. Data security plays an important role in such communication. One of the most important methods for ensuring data secrecy is cryptography. Cryptography is a method of protecting information and communications through the use of codes, so that only those for whom the information is intended to read and process it. It not only protects data from theft or alteration, but can also be used for user authentication.

1.1 The Function of the Encryption System

Encryption technique uses an encryption algorithm (mathematical process) to encrypt and decrypt the data. Encryption algorithms is a mathematical formula, with the help of a key, changes plaintext into ciphertext. The encryption algorithms are usually divided into two types: Symmetric key encryption (private) and Asymmetric key encryption (public). In Symmetric key encryption or secret key encryption, the same key is used to encrypt (scramble) and decrypt (unscramble) data. In Asymmetric key encryption or public key encryption, two different keys are used, one for encryption and the other for decryption.

1.2 Motivation of the System

Information security is crucial for every organization day by day, big and small. Information security means protecting digital data such as those in a database from destructive forces and from the unwanted actions of unauthorized users such as a cyberattack or a data breach. Maintaining privacy is important in our personal communication. Because everyone desires want to know personal of privacy. Encryption technique was invented for achieve that privacy. Nowadays, the document file is widely used in every organization; its security is important for organization and users. There is a need for the document file encryption in order to provide secure communication.

1.3 Objectives of the System

The objectives of the system are as follows:

- To secure for the important data in IRD system
- To study the data security for a storage system
- To study the security system and programming for security
- To study how the encryption mechanism plays a vital role for organization
- To implement one of the cryptographic technique
- To develop data encryption system by using Blowfish Algorithm

1.4 Overview of the System

There are two main parts of the system: encryption and decryption phase. In encryption phase, the system uses 64 bits keys stream. The system encrypts the selected document file using Blowfish encryption algorithm and cipher file will generate.

In decryption phase, the system calculates the rounds keys upon the input symmetric keys using key expansion algorithm. And then the system decrypts the selected document file that is encrypted using Blowfish decryption algorithm. The encryption and decryption keys must be same because the Blowfish algorithm is a symmetric algorithm.

1.5 Organization of the System

The organization of the thesis is as follows: Chapter 1, introduction, motivation, objectives, overview of the system and thesis organization are described. Chapter 2 presents the background theory that are dealing with the thesis. Chapter 3 discusses system overview and Blowfish Algorithm. Chapter 4 expresses the design and implementation of the proposed system. Finally, Chapter 5 summarizes the conclusion of this thesis.

CHPATER 2

BACKGROUND THEORY

2.1 Cryptography

Cryptography is a method of protecting information and communications through the use of codes, so that the information is only used by this method user. The term is derived from the Greek word kryptós, meaning "hidden or secret" and "graphein", meaning "write or study". So, cryptography is the art of secret writing. In cryptography the techniques which are use to protect information are obtained from mathematical concepts and a set of rule-based calculations known as algorithms to convert data in ways that make it hard to decode it.

A message in its original human readable form is known as plaintext. Plaintext would refer to any message, document and file. Ciphertext is changeable text transformed from plaintext using an algorithm. Ciphertext can't be read until it has been converted into plaintext with a key. The process of conversion of cipher text to plain text is known as decryption.

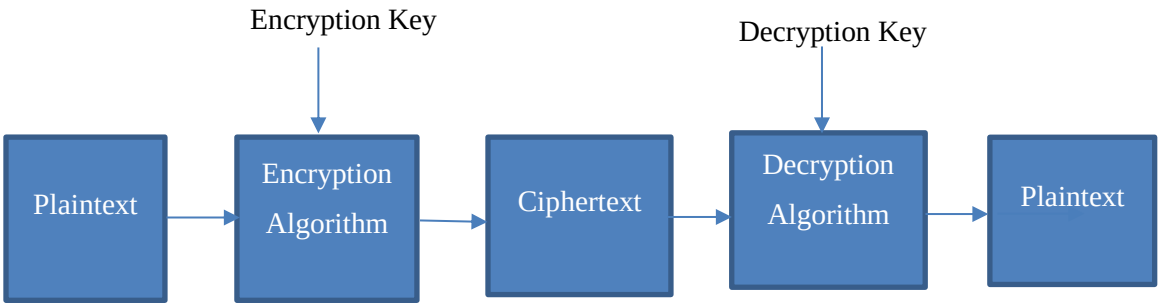


Figure 2.1 Cryptographic Process

Cryptographic systems include both an algorithm and a secret value as shown in Figure 2.1. The secret value is known as the key. The reason for addition a key in cryptography is that it is difficult to scrambling of information.

2.2 Cryptographic Features

Cryptographic systems can cover one or more of the following services:

- **Confidentiality:** Information can only be accessed by the person for whom it is intended and no other person except him can access it.
- **Integrity:** Information cannot be modified in storage or transition between sender and intended receiver without any addition to information being detected.
- **Non-repudiation:** The creator/sender of information cannot deny his intention to send information at later stage.
- **Authentication:** The identities of sender and receiver are confirmed. As well as destination/origin of information is confirmed.

2.3 Cryptographic Algorithms

Cryptographic algorithms are sequences of processes, or rules, used to encipher and decipher messages in a cryptographic system. In simple terms, they're processes that protect data by making sure that unwanted people can't access it. These algorithms have a wide variety of uses, including ensuring secure and authenticated financial transactions.

Most cryptography algorithms involve the use of encryption, which allows two parties to communicate while preventing unauthorized third parties from understanding those communications. In general, there are three types of cryptography:

(1) Symmetric Key Cryptography (SKC)

(2) Asymmetric Cryptography (PKC)

(3) Hash Function

2.3.1 Symmetric Key Cryptography

Symmetric encryption is an old and best-known technique. Symmetric uses single key, which works for both encryption and decryption. Symmetric-key algorithms are those algorithms which uses only one and only key for both. The key is kept as secret. Symmetric algorithms have the advantage of not taking in too much of computation power and it works with very high speed in encryption. Symmetric key cryptography is

simpler and faster but the main drawback is that both sides must agree upon the secret key before the starting of the transmission and they should maintain it secretly.

The sender uses the key to encrypt the plaintext and sends the ciphertext to the receiver. The receiver also uses the same key to decrypt the message and recover the plaintext. Because a single key is used for both functions, symmetric key cryptography is also known as private key or secret key cryptography.

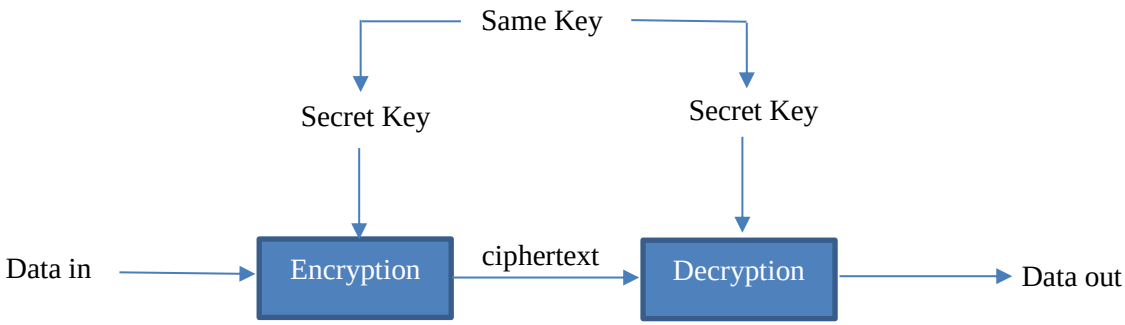


Figure 2.2 Symmetric Encryption Process

2.3.2 Asymmetric Key Cryptography

Asymmetric uses different keys for both encryption and decryption. One key, the public key, is used for encryption and the other, the private key, is for decryption. The public key is distributed freely. The private key is required must be kept secretly. Asymmetric key cryptography is more secure but the main drawback is that the encryption process is slow and resource utilization is high.

The important point here is that both keys are required to process the work. Because a pair of keys is required, this approach is also known as public key or conventional cryptography. Figure 2.3 shows the basic process of this cryptographic function.

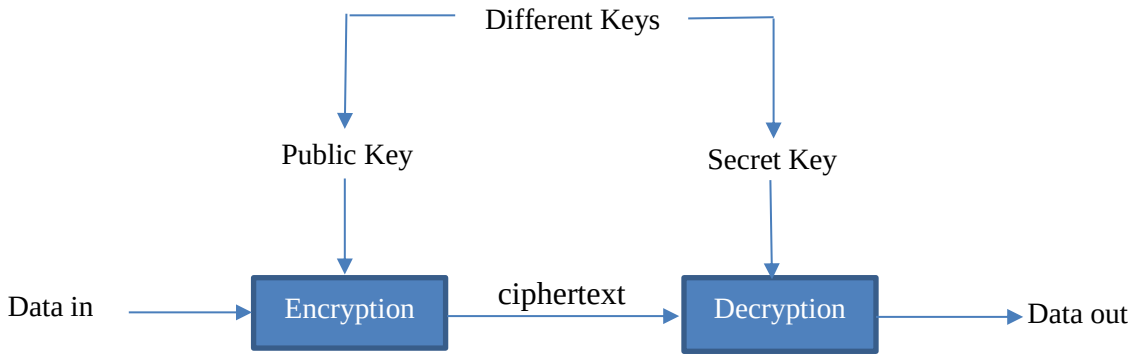


Figure 2.3 Asymmetric Encryption Process

2.3.3 Hash Function

Hash functions takes an arbitrary amount of data input and produces a fixed-size output of enciphered text called a hash value, or just "hash". This method will not need any kind of key as it functions in a one-way scenario. It is also termed as a mathematical equation by taking numerical values as input and produce the hash message. A hash value with fixed length is calculated as per the plain text which makes it impossible for contents of plain text to be recovered. Many operating systems use hash functions to encrypt passwords.

Hash functions is also known as message digest function, one-way encryption and hashing algorithm. Hashing algorithms are typically used in the encryption and decryption of digital signatures.

Hash functions are sometimes misunderstood that no two files the same hash value. That is, in fac, not correct.

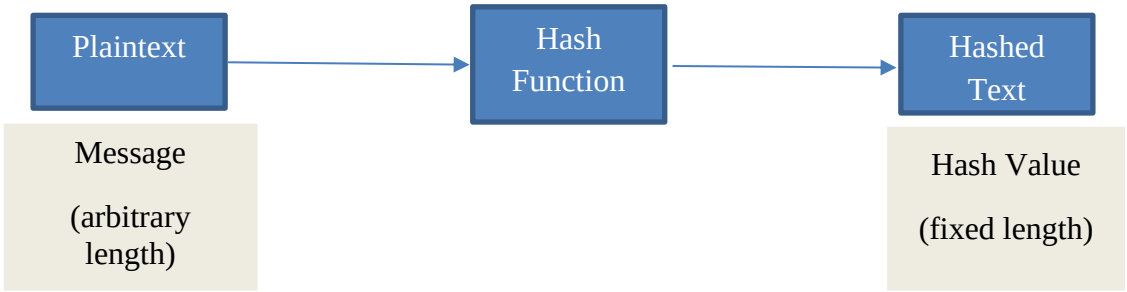


Figure 2.4 Hash Function Process

2.4 Encryption

Encryption is a way of translation data into a secret code. Encryption is one of the most popular and effective data security methods used by organizations. To read an encrypted file, user must have a secret key or password to decrypt it. Encrypted data is commonly referred to as ciphertext, while unencrypted data is called plaintext. Encryption has been a longstanding way for sensitive information to be protected.

There are two main kinds of encryption:

1. Asymmetric Encryption (also called Public Key Encryption) and
2. Symmetric Encryption.

2.4.1 Asymmetric Encryption

A cryptographic system uses two keys: a public key is used for encryption, and a secret or private key is used for decryption. The decryption key is kept private (the "private key"), while the encryption key is shared publicly, for anyone to use (the "public key"). When Alice wants to send a secure message to Jasmine, he uses Jasmine's public key to encrypt the message. Jasmine then uses her private key to decrypt.

An important fact that only the public key can be used to encrypt the messages and only the corresponding private key can be used to decrypt them. The only difficulty of public key system is that user need to know the recipient's public key to encrypt a message for him. Public key cryptography was invented in 1976 by Whitfield Diffie and Martin Hellman. So, it is sometime called Diffie-Hellman encryption. It is also called asymmetric encryption because it uses two keys.

2.4.2 Symmetric Encryption

Symmetric encryption is the process of converting plaintext into ciphertext and vice versa using the same secret key. A secret key that can either be a number, a word or a string of random letters. Secret key is blended to the plain text of a message to change the content in a particular way. Symmetric encryption algorithm requires that both the sender and the recipient known the secret key so they can encrypt and decrypt all the messages.

There are two types of symmetric encryption algorithms: Stream algorithms (Stream ciphers) and Block algorithms (Block ciphers).

2.4.3 Hybrid Encryption

A method of hybrid encryption that combines two or more encryption methods and includes a combination of symmetric and asymmetric encryption to take advantage of strengths of each type of encryption.

2.5 Encryption Algorithms

Encryption algorithm or a cipher is a mathematical function used in the encryption and decryption process. It is the sequences of mathematical transforms plaintext or readable information incomprehensible ciphertext. In simple terms, they're processes that protect data by making sure that unwanted people can't access it. A cryptographic algorithm is a combination with a key (number, word, or string) to encrypt and decrypt data.

The algorithm combines the information to be protected with a supplied key in encryption. And then, the encrypted data is achieved. The algorithm performs a calculation the encrypted data with the supplied key in decryption. The result of this is the decrypted data. The goal of every algorithm is intended to make it difficult to decrypt the generated ciphertext without the corresponding key.

Each algorithm uses a set of mathematical values known as a “key” to perform the calculations. The longer the key, it harder to break the encryption by guessing the key. Some cryptographic method rely on secrecy of the encryption algorithms; such algorithm are adequate for real-world needs.

2.6 Types of Encryption Algorithms

There are two different types of encryption algorithms, symmetric encryption algorithms (secret key algorithms) and asymmetric encryption algorithms (public key algorithms). The basic difference between these two types of encryption is that symmetric encryption algorithms allow encryption and decryption (the decryption key is derived from the encryption key) of the message with the same key. On the other hand, asymmetric

encryption algorithms use a different key for encryption and decryption (the decryption key cannot be derived from the encryption key).

2.6.1 Symmetric Encryption Algorithms

Symmetric encryption algorithms can be divided into block ciphers and stream ciphers. Most encryption algorithms use the block ciphers method, which encrypts the plain text into cipher text by taking a number of bits (typically 64 bits in modern ciphers) known as block at a time. The usual size of each block are 64bits, 128bits and 256bits. So for example, a 64-bit block cipher will take in 64bits of plaintext and encrypts it into 64 bits of ciphertext. Most of the symmetric ciphers used today are actually block ciphers. Some encryption algorithms use the stream ciphers, which encrypts the plain text into cipher text by taking a single bit or byte of plain text at a time.

Some examples of commonly used symmetric key encryption algorithms are as follows:

1. Advanced Encryption Standard (AES)
2. Data Encryption System (DES)
3. Blowfish
4. Triple DES (3DES)
5. Twofish
6. SEED

2.6.1.1 Advanced Encryption Standard (AES)

AES stands for Advanced Encryption Standard. AES, also known as Rijndael, is a type of symmetric block cipher. This block cipher was developed by Joan Daemen and Vincent Rijmen, and was chosen in October 2000 by the National Institute of Standards and Technology to be the U.S.'s new Advanced Encryption Standard. AES algorithm uses three key sizes: a 128-,192-, or 256-bit encryption key. Each encryption key size causes the algorithm to behave slightly differently, so the increasing key sizes not only offer a larger number of bits with which user can scramble the data, but also increase the complexity of the cipher algorithm.

2.6.1.2 Data Encryption System (DES)

The Data Encryption System (DES) is a symmetric block cipher that used 56-bit keys and encrypted block sizes of 64 bits. DES was adopted in the United States as a federal standard in 1977. DES encrypts and decrypts data in 64-bit blocks, using a 56-bit key. It takes a 64-bit block of plaintext as input and outputs a 64-bit block of ciphertext. DES consists of 16 steps, each of which is called a round, meaning the main algorithm is repeated 16 times to produce the ciphertext. Each round performs the steps of substitution and transposition in the algorithm. The DES is a strong algorithm, but today the short key length limits its use.

2.6.1.3 Blowfish

Blowfish is an encryption technique designed by Bruce Schneier in 1993 as an alternative to existing encryption algorithm. Blowfish has a 64-bit block size and a variable key length from 32 bits up to 448 bits. It is a 16-round Feistel cipher and uses large key-dependent S-boxes. This approach has been proven to be highly resistance against many attacks such as differential and linear cryptanalysis.

It has been analyzed considerably, and it is slowly gaining acceptance as a strong encryption algorithm. Blowfish is unpatented and license-free, and is available free for all uses.

2.6.2 Asymmetric Encryption Algorithms

Asymmetric encryption algorithms (public key algorithms) use paired keys (a public and a private key) in performing their function for encryption and decryption. Asymmetric encryption methods are important because they can be used for transmitting encryption keys or other data securely even w when the parties have no to agree on a secret key in private.

Types of asymmetric key encryption algorithm are as follows:

1. Rivest-Shamir-Adleman (RSA)
2. Diffie-Hellman
3. Digital Signature Algorithm (DSA)

4. EIGamal
5. Elliptic Curve Digital Signature Algorithm (ECDSA)
6. XTR

2.6.2.1 Rivest-Shamir-Adleman (RSA)

Rivest-Shamir-Adleman (RSA) is the most commonly used asymmetric algorithm (public key encryption algorithm). It can be widely used both for digital signatures and encryption. The security of RSA is generally considered equivalent to factoring, although this has not been proved.

RSA computation occurs integers modulo n=p * q, for two large secret primes p and q. To encrypt a message m, it is exponentiated with a small public exponent e. For decryption, the recipient of the ciphertext c=me (mod n) computes the multiplicative reverse d=e-1(mod (p-1) * (q-1)) (we require that e is selected suitably for it to exist) and obtains cd=m e * d= m (mod n). The private key consists of n, p, q, e, d (where p and q can be omitted); the public key contains only n and e. The problem for the attacker is that computing the reverse d of e is assumed to be no easier than factorizing 'n'.

The key size should be greater than 1024 bits for a reasonable level of security. Keys of size, says, 2048 bits should allow security for decades. There are actually multiple incarnations of this algorithm; RC5 is one of the most common in use, and RC6 was a finalist algorithm for AES.

2.6.2.2 Diffie – Hellman

Diffie-Hellman is the first public key encryption algorithm invented in 1976 by Whitfield Diffie and Martin Hellman. This algorithm uses the discrete logarithm in a finite field. It allows two users to exchange a secret key over an insecure channel without any prior secrets.

Diffie-Hellman (DH) is a widely used key exchange algorithm. However, it is assumed that they do not initially possess any common secret thus cannot use secret key cryptosystems. The key exchange by Diffie-Hellman protocol remedies this situation by allowing the construction of a common secret key over an unsafe channel. It is based on a

problem related to discrete logarithms, namely the Diffie-Hellman problem. This problem is considered hard, and it is in some instances as hard as the discrete logarithms problem.

The Diffie-Hellman key exchange was one of the most important developments in public-key cryptography and it is still frequently implemented in a range of today's different security protocols.

The Diffie-Hellman protocol is generally considered to be secure when an appropriate mathematical group is used. In particular, the generator element used in the exponentiations should have a large period (i.e. order). Usually, Diffie-Hellman is not implemented on hardware.

2.6.2.3 Digital Signature Algorithm (DSA)

Digital Signature Algorithm (DSA) is a United States Federal Government standard or FIPS for digital signatures. It was proposed by the National Institute of Standards and Technology (NIST) in August 1991 for use in their Digital Signature Algorithm (DSA), specified in FIPS 186 [1], adopted in 1993. A minor revision was issued in 1996 as FIPS 186-1 [2], and the standard was expanded further in 2000 as FIPS 186-2 [3].

Digital Signature Algorithm (DSA) is similar to the one used by EIGamal signature algorithm. It is fairly efficient though not as efficient RSA for signature verification. The standard defines DSS to use the SHA-1 hash function exclusively to compute message digests.

The main problem with DSA is the fixed subgroup size (the order of the generator element), which limits the security to around only 80 bits. Hardware attacks can be menacing to some implementations of DSS. However, it is widely used and accepted as a good algorithm.

2.7 Benefits of Symmetric Encryption Algorithm over Asymmetric Encryption Algorithm

Symmetric encryption is used today because it can encrypt and decrypt large amounts of data quickly, and it's easy to implement. It's simple to use, and its Blowfish iteration is one of the most secure forms of data encryption available. Symmetric encryption has several advantages over asymmetric encryption. Asymmetric encryption is slow compared with symmetric encryption, which means that it is not suitable for decrypting bulk messages. It is also impractical to use them to encrypt large amounts of data.

Its public keys are not authenticated. Basically, no one absolutely knows that a public key belongs to the individual it specifies, which means that users will have to verify that their public keys truly belong to them. A symmetric encryption algorithm uses password authentication to prove the receiver's identity.

Asymmetric encryption risks loss of private key, which may be irreparable. When you lose your private key, your received messages will not be decrypted. Even if the secret key is lost in symmetric encryption, you can ask the sender again.

The main disadvantage of asymmetric encryption is that it's slower than symmetric encryption and resource utilization is high. Also, in theory, public keys can be used to crack private keys. Because, they're also mathematically linked to each other. Asymmetric encryption algorithms require at least a 2048-bit key to achieve the same level of security of a 128-bit symmetric algorithm.

Symmetric encryption algorithms like Blowfish take many years to crack using attacks. Symmetric encryption algorithms like Blowfish have become the gold standard of data encryption because of its security and speed benefits.

CHAPTER 3

SYSTEM OVERVIEW AND BLOWFISH ALGORITHM

3.1 System Overview

The proposed system is implemented for taxpayer side and admin side. The taxpayer side can develop the Blowfish encryption/decryption and admin side can also develop Blowfish encryption/decryption.

In taxpayer side, the user needs to fill taxpayer's data, return file which want to pass and the secret key as in Figure 3.1. In the background of the application, the program takes the information, such as key and return file, work through the operation steps of the Blowfish algorithm. After these steps, the program outputs the encrypted file (ciphertext) and which will upload on a e-filling page.

When the admin downloads the return file from e-filling page, the decryption process takes place and the overview system flow of the Blowfish decryption is shown in Figure 3.2.

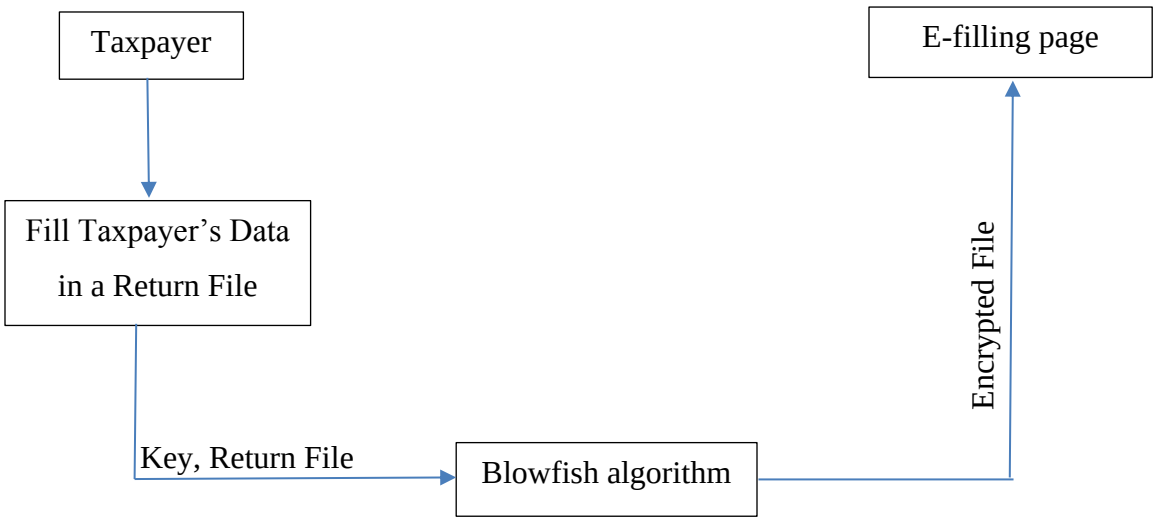


Figure 3.1 Overview of the System Design for Encryption

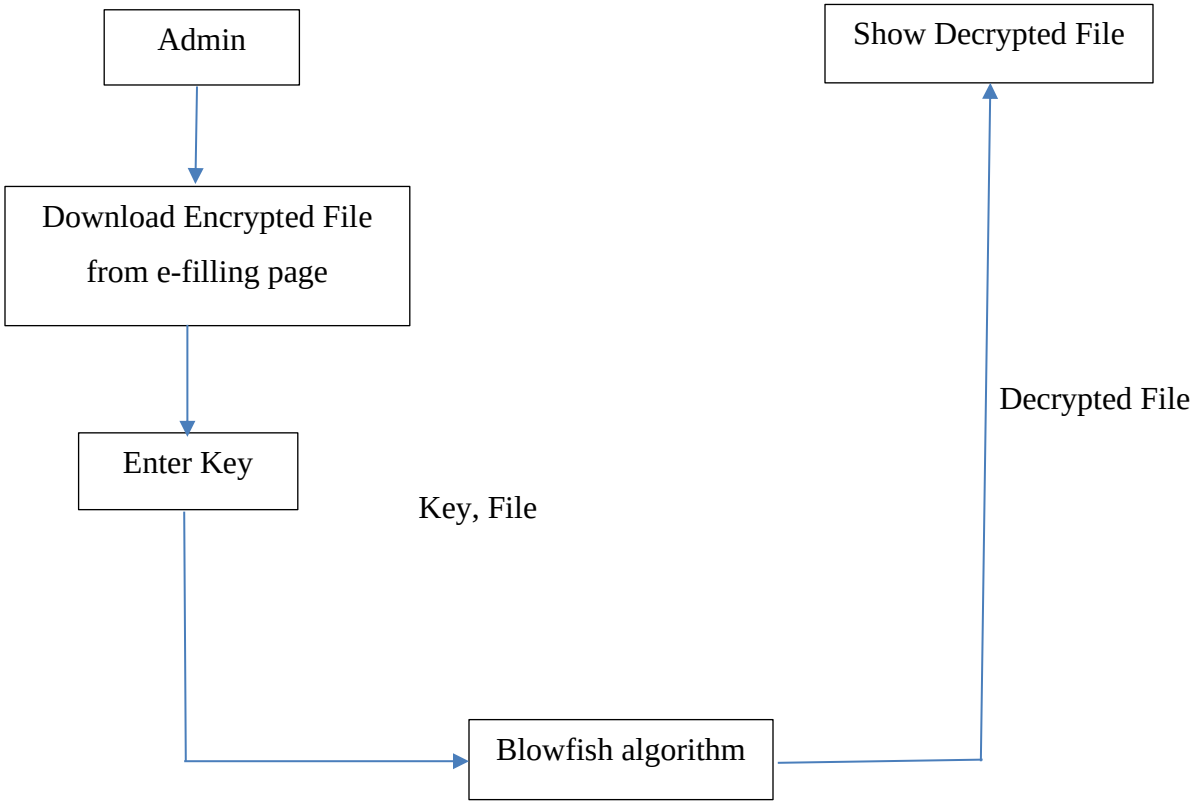


Figure 3.2 Overview of the System Design for Decryption

The decryption stage is identical to encryption, with the exception that the P array is used in reverse order. When admin enters the secret key, the program uses this key to operate the steps of Blowfish decryption. After processing of these steps with required rounds, the program generates the original return file (plaintext).

3.2 Flow Chart

The system flow design of taxpayer is shown in Figure 3.3.

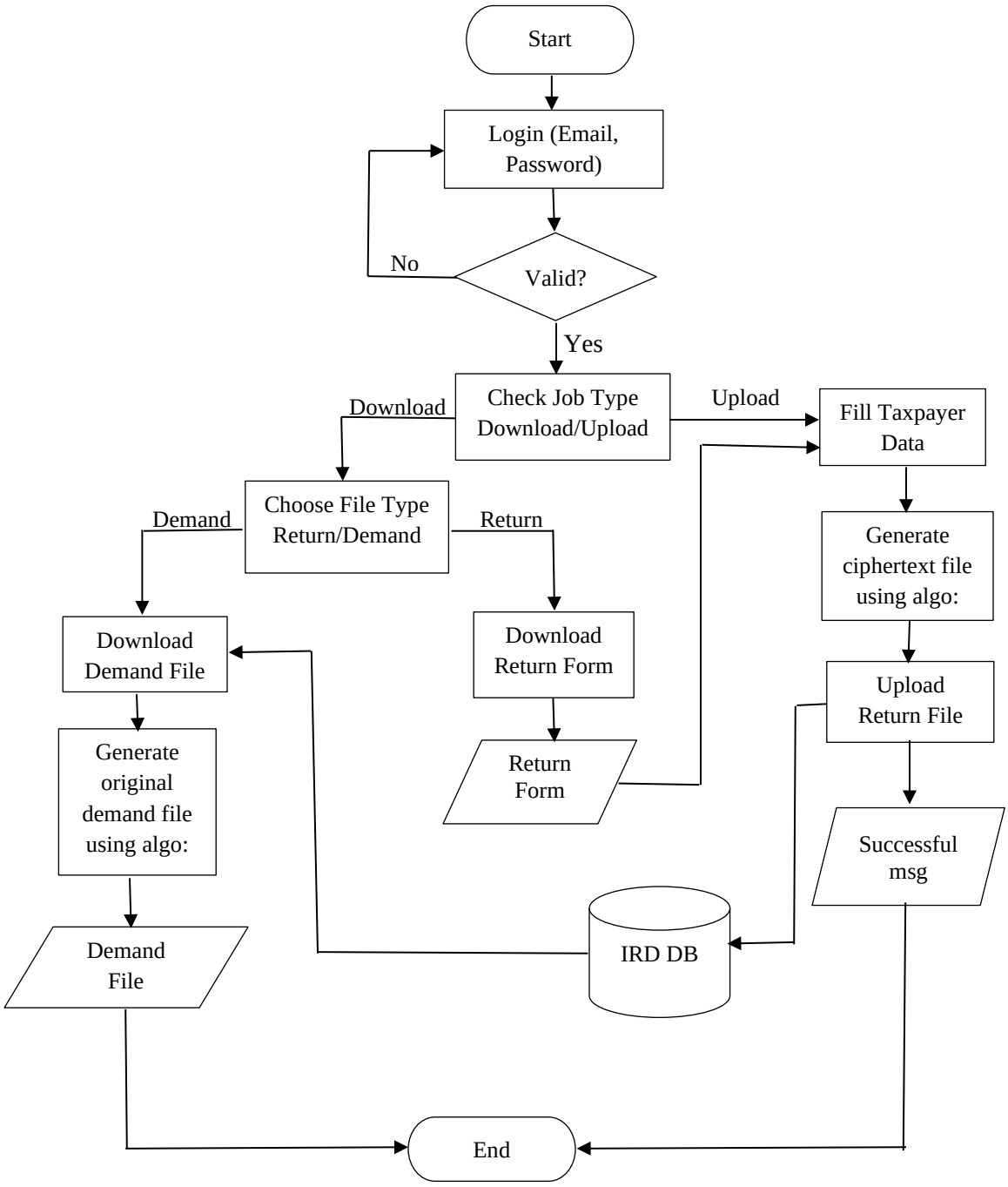


Figure 3.3 System Flow Diagram for Taxpayer

At the taxpayer side, the taxpayer opens the application, he/she will need to login with email and password. When the email and password are correct, he/she can upload and download action. The taxpayer needs to download return form before the return file is uploaded. Then, he/she need to fill the information about the return file and upload to e-filling page. In the background of the application, the program takes the information, such as key and return file, work through the operation steps of the Blowfish algorithm. After these steps, the return file is successfully uploaded and shown on e-filling page.

The taxpayer can download the tax amount file uploaded by the admin on the e-filling page. When he/she downloads the tax amount file, he/she will need to enter the secret key to generate the original tax amount file. In the background of the application, the program takes the secret key, work through the operation steps of the Blowfish algorithm. After these steps, the program outputs the original tax amount file will appear.

When the taxpayer wants to review the return file uploaded by himself/herself, he/she can also download it with corresponding secret key. In figure 3.4, admin's system flow is shown.

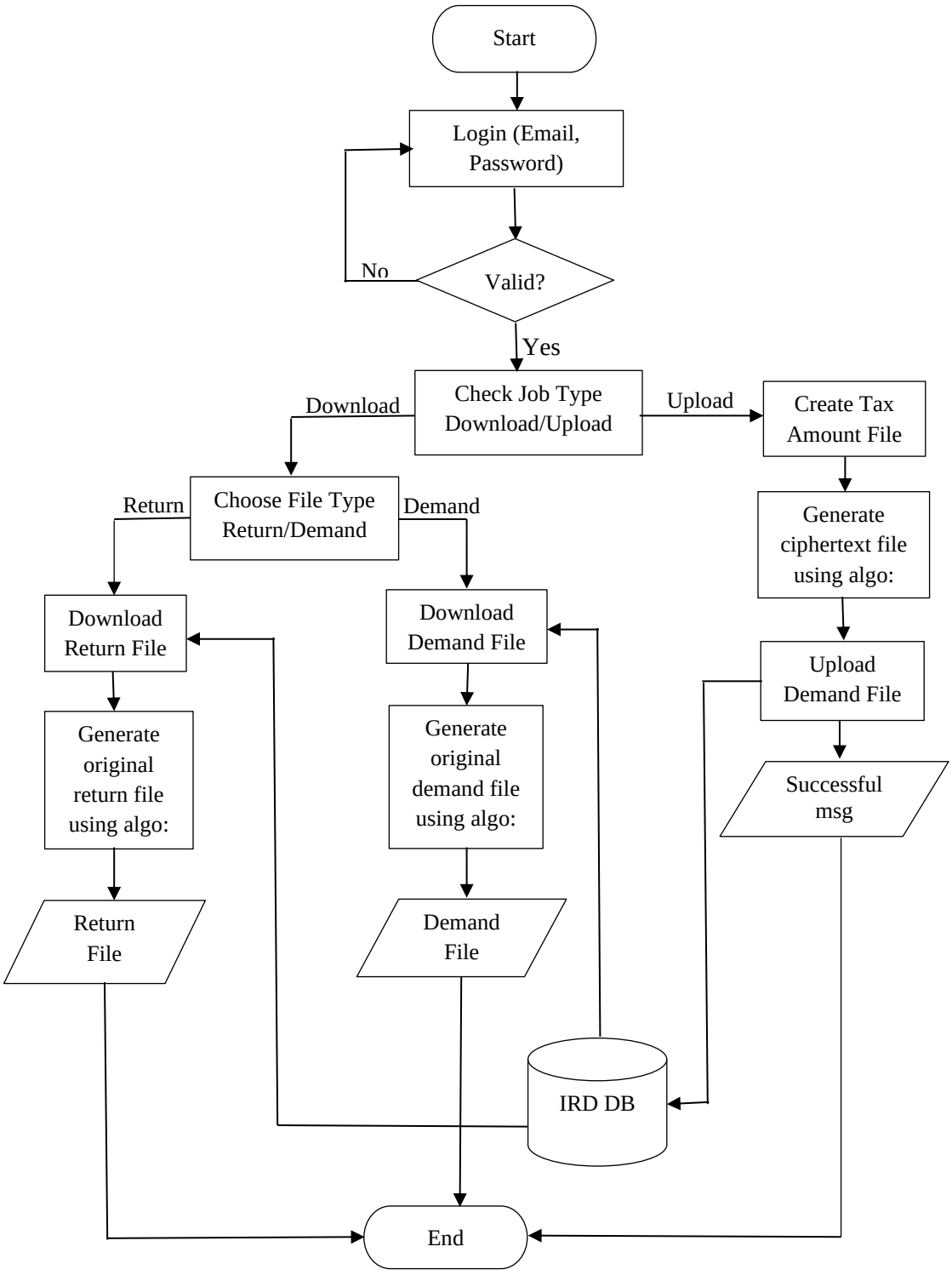


Figure 3.4 System Flow Diagram for Admin

When admin start to use the system, he/she will need to login with email and password. When the admin and password are correct, he/she can upload and download.

The admin can download the return file uploaded by the taxpayer on the e-filling page. When he/she downloads the return file, he/she will need to enter the secret key to generate the original return file. In the background of the application, the program takes the secret key, work through the operation steps of the Blowfish algorithm. After these steps, the program outputs the original return file will appear.

The admin looks at the return file downloaded and calculates the tax amount to be paid. And, the admin uploads the calculated tax amount file as in Figure 3.4. The admin has to enter a secret key before the calculated tax amount file is uploaded. In the background of the application, the program takes the information, such as key and calculated tax amount file, work through the operation steps of the Blowfish algorithm. After these steps, the tax amount file is successfully uploaded and shown on e-filling page.

When the admin wants to review the tax amount file uploaded by himself/herself, he/she can also download it with corresponding secret key.

3.3 Blowfish Algorithm

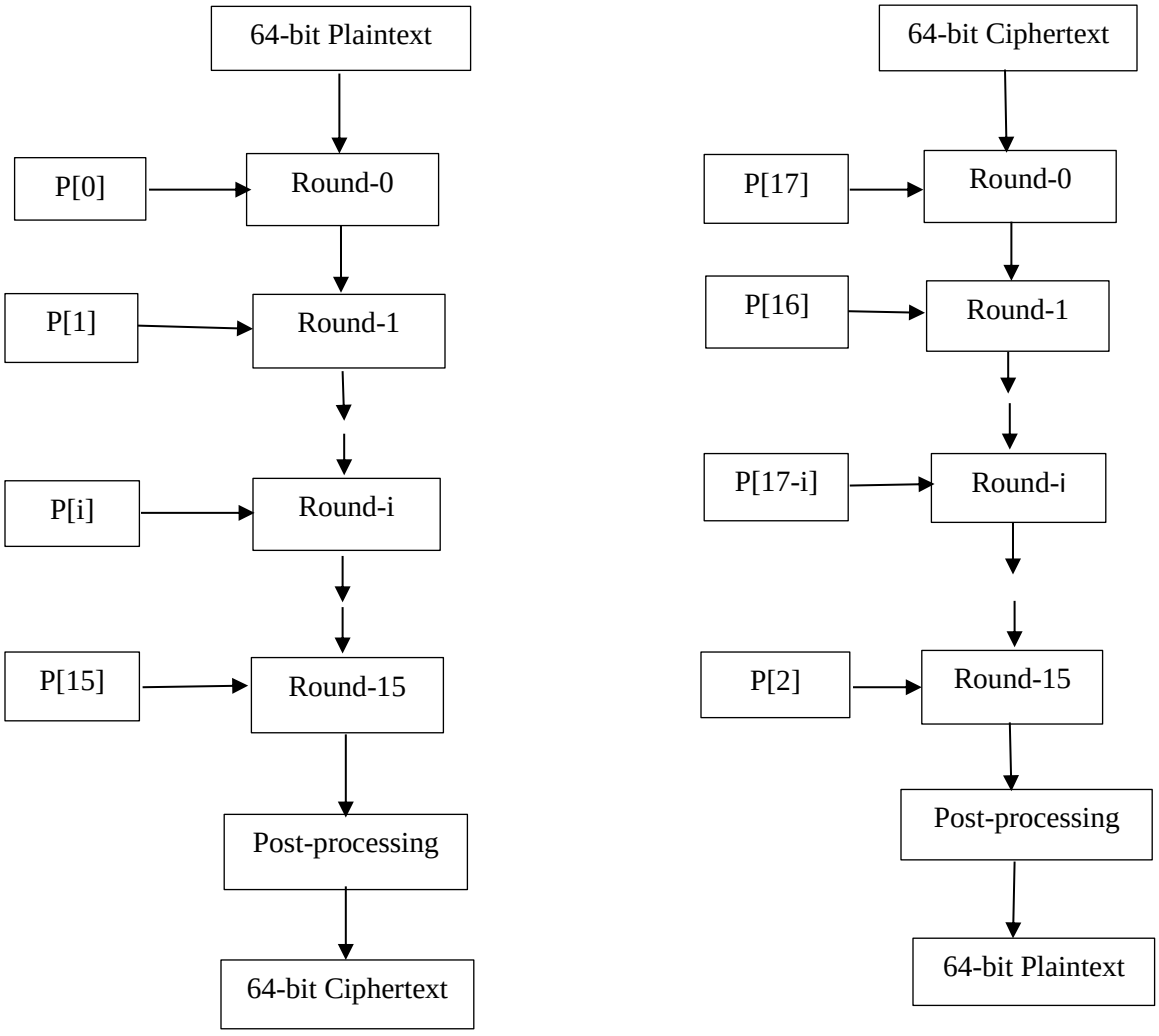
Many symmetric block ciphers have been presented in recent years. Blowfish is a symmetric key block cipher, designed in 1993 by Bruce Schneier and included in many cipher suites and encryption products. Blowfish provides a good encryption rate in software, and no effective cryptanalysis of it has been found to date. Schneier designed Blowfish as a general-purpose algorithm, intended as an alternative to the aging DES and free of the problems and constraints associated with other algorithms.

Blowfish is a 64-bit symmetric block cipher that uses a variable-length key (symmetric key) from 32 to 448-bits (14 bytes). The algorithm was developed to encrypt 64-bits of plaintext into 64-bits of cipher text efficiently and securely. It is a 16-round Feistel cipher and uses large key-dependent S-boxes. It consists of 16 Feistel-like

iterations, where each iteration operates on a 64-bit block that's split into two 32-bit words. Blowfish uses a single encryption key to both encrypt and decrypt data.

3.4 Blowfish Structure (64 bits)

The Blowfish algorithm consists of two major parts: Key Expansion and Data Encryption. The operations selected for the algorithm were table lookup, addition and bitwise exclusive-or to minimize the time required to encrypt and decrypt data on 32-bit processors. The table includes four S-boxes and a P-array. The algorithm has two parts, data encryption and data decryption phase [1];[2].



(a) Encryption (b)Decryption

Figure 3.5 Blowfish Structure (64 bits)

3.4.1 Blowfish Encryption Phase

The encryption phase has two parts, key expansion and data encryption [1];[2].

- **Key Expansion**

  Key expansion consists of generating the initial contents of one array (the

  P-array) namely, eighteen 32-bit sub keys, and 4 Substitution boxes(S-boxes) are needed in both encryption as well as decryption process with each S-box having 256 entries where each entry is 32-bit.

  The Key Expansion of Blowfish begins with the P-array and S-boxes with

  the utilization of many sub-keys, which requires pre-computed before data encryption or decryption [3].

  In the key expansion process, maximum size 448-bit keys are converted into

  several subkey arrays totaling 4,168 bytes. Subkeys form an integral part of the Blowfish algorithm, which uses a large number of them. These subkeys are pre-computed before encryption or decryption can take place.

  The hexadecimal representation of each of the subkeys is given by:

P [1] : 85a308d3	P [10] : be5466cf
P [2] : 13198a2e	P [11] : 34e90c6c
P [3] : 03707344	P [12] : c0ac29b7
P [4] : a4093822	P [13] : c97c50dd
P [0] : 243f6a88	P [9] : 38d01377
P [5] : 299f31d0	P [14] : 3f84d5b5
P [6] : 082efa98	P [15] : b5470917
P [7] : ec4e6c89	P [16] : 2916d5d9
P [8] : 452821e6	P [17] : 8979fb1b

Figure 3.6 32-bit hexadecimal represents initial values of sub-keys

The resultant P-array holds 18 subkeys that is used during the entire encryption process.

There are 256 entries for each of the four 32-bit S-boxes:

S1,0, S1,1,..., S1,255

S2,0, S2,1,..., S2,255

S3,0, S3,1,..., S3,255

S4,0, S4,1,..., S4,255

- **Data Encryption**

The encryption function consists of two parts:

a. Rounds : The encryption consists of 16 rounds with each round taking inputs the plaintext from previous round and corresponding subkey.Each round consists of four steps:

Step 1: The left half (L) of the data is XORed with the i th P-array entry

Step 2: The result data of step1 is used input for Blowfish's F-function

Step 3: The F-function's output value is XORed with the right half (R) of data

Step 4: Swap L and R.

The description of each round is shown in figure 3.7.

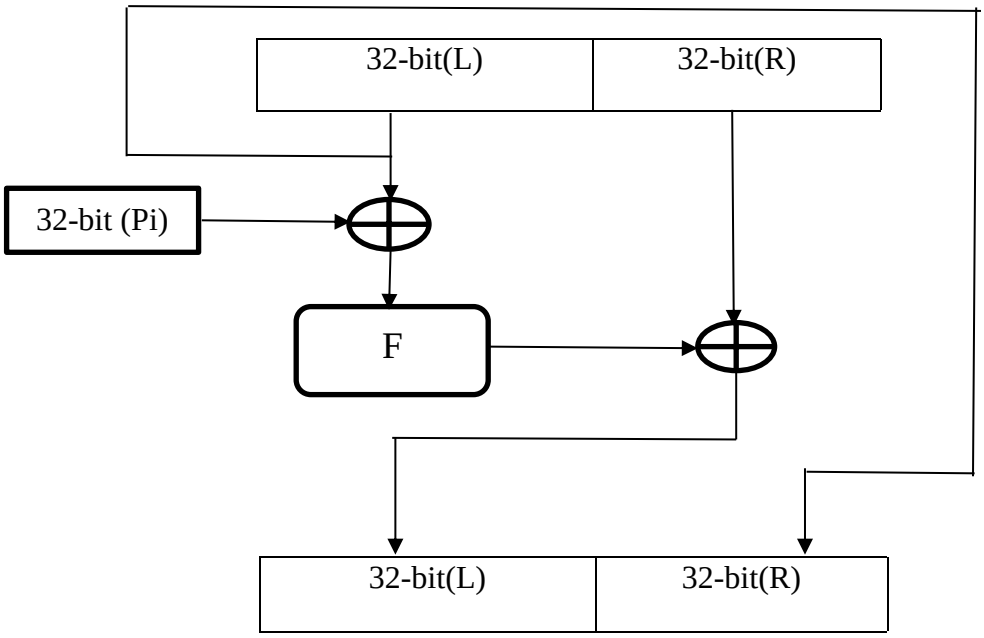


Figure 3.7 Flow Diagram of Each Round

b. Post-processing : In post-processing , the last two unused P-box entries are XORing with plaintext from previous round. The description of post-processing step as follows:

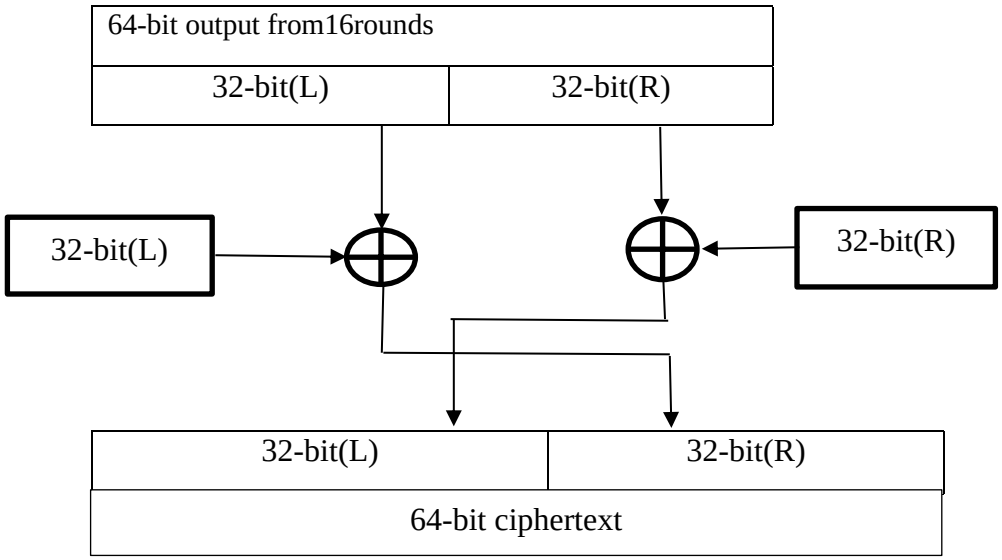


Figure 3.8 General Architecture of Post-processing Step

These steps are completed after all of iteration and finally 64 bits of ciphertext is received from the output.

Pseudocode for encryption

for i = 1 to 16;

LXi = LX XOR Pi

RXi = F(LXi) XOR RX

Swap LXi and RXi

After the sixteenth iteration,

Swap LX and RX again to undo the last swap.

RX= RX XOR P17 and LX = LX XOR P18.

Recombine LX and RX to get ciphertext

Figure 3.9 Pseudocode for Blowfish Encryption Algorithm

3.4.2 Blowfish Decryption Phase

The key P-box is used in the reverse order for decryption process. The pseudocode for the components of the cipher are given below.

Pseudocode for decryption

for i = 1 to 16;

LXi = LX XOR Pi

RXi = F(LXi) XOR RX

Swap LXi and RXi

After the sixteenth iteration,

Swap LX and RX again to undo the last swap.

RX= RX XOR P1 and LX = LX XOR P2.

Recombine LX and RX to get plaintext

Figure 3.10 Pseudocode for Blowfish Decryption Algorithm

3.5 Strength of Blowfish Algorithm

Blowfish encrypted the data block in 16 rounds. It is a 16-round Fiestel cipher and uses large key-dependent S-boxes. Blowfish encryption is fast and flexible. Otherwise, the advantage of Blowfish is faster executed in encryption rate because the process of encryption and decryption are the same.

This standard is designed for encoding and decoding with high speed in the application performance saving with implacable security for the data. Which is also powerful and quick to secure data in online and in web applications.

3.6 Original and Ciphertext of Return File and Demand File

Previously stated the Blowfish block size (64 bits) is tested with a key of encryption.. So, the following values are received because of the test. Sample return file is shown in figure 3.11. In figure 3.14, sample demand file is shown.

The Government of the Republic of the Union of Myanmar

Tax Return

1	Type of Taxpayer	Company
2	Residency	Resident Foreigner
3	Company Name	Hiltop Company Limited
4	Taxpayer Identification Number (TIN)	10472416
5	Date of Commencement of Operation	1-10-2020
6	Business Contact Address	No.4, Sabel Street, Yankin Township, Yangon, Myanmar
7	Office Contact Phone Number	09452702328
8	Contact Email Address	hiltop@gmail.com
9	Industry Code	1313
10	Income Year	2020-2021
11	Business Income	50,000,000.00
12	Total Quarterly Advance Tax Payments	13,000,000.00
13	Amount of Tax Overpaid Last Year Carried Forward to This Year	0
14	Submitted Date	31-Dec-2021
15	Name of Responsible Person	Hui Yi
16	Position	Director

Figure 3.11 Sample Return File

When this return file is uploaded with the secret key and then encrypted, the output values of ciphertext is received. The received values are shown in figure 3.12 (a) and (b).

In this proposed system, secret key is set to the last four number of tax identification number and the abbreviation of the income year.

255044462d312e370d0a25b5b5b5b50d

0a312030206f626a0d0a3c3c2f547970

652f436174616c6f672f506167657320

32203020522f4c616e6728656e2d5553

29202f5374727563745472656526f6f

74203132203020522f4d61726b496e66

6f3c3c2f4d61726b656420747275653e

3e2f4d65746164617461203133392030

20522f566965776572507265726566572 65

6e6365732031343020302 0523e3e0d0a

656e646f626a0d0a322030206f626a0d

0a3c3c2f547970652f50616765732f43

6f756e7420312f4b6964735b20332030

20525d203e3e0d0a656e646f626a0d0a

332030206f626a0d0a3c3c2f54797065

2f506167652f506172656e7420322030

20522f5265736f75726365733c3c2f46

6f6e743c3c2f463120352032030 20522f46

322039203020523e3e2f457874475374

6174653c3c2f47533720372032030 20522f

475338203820302 0523e3e2f50726f63

5365745b2f5044462f546578742f496d

616765422f496d616765432f496d6167

65495d203e3e2f4d65646961426f785b

203020302036313220373932 5d202f43

6f6e74656e747320342032030 20522f4772

6f75703c3c2f547970652f47726f7570

Figure 3.12 (a) Ciphertext of Return File

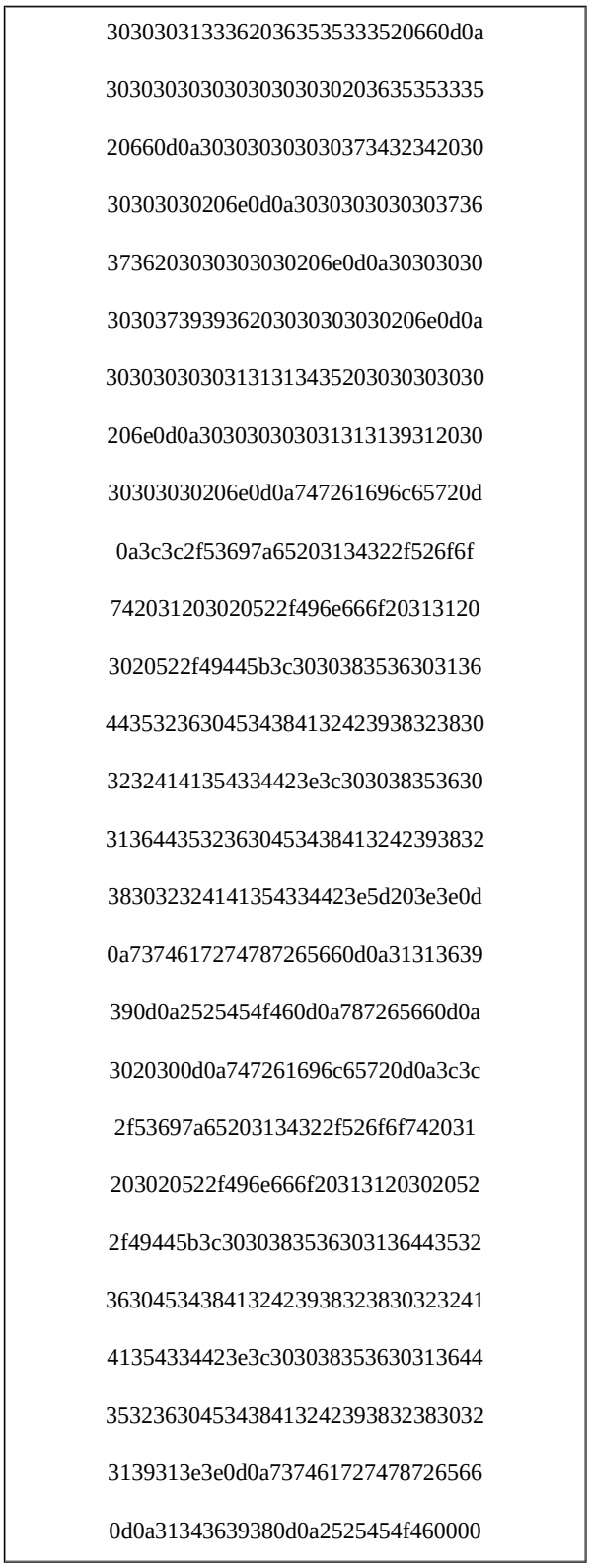


Figure 3.12 (b) Ciphertext of Return File

When this return file is downloaded with the secret key and then decrypted, the original return file is received. The received file is shown in figure 3.13.

The Government of the Republic of the Union of Myanmar

Tax Return

1	Type of Taxpayer	Company
2	Residency	Resident Foreigner
3	Company Name	Hiltop Company Limited
4	Taxpayer Identification Number (TIN)	10472416
5	Date of Commencement of Operation	1-10-2020
6	Business Contact Address	No.4, Sabel Street, Yankin Township, Yangon, Myanmar
7	Office Contact Phone Number	09452702328
8	Contact Email Address	hiltop@gmail.com
9	Industry Code	1313
10	Income Year	2020-2021
11	Business Income	50,000,000.00
12	Total Quarterly Advance Tax Payments	13,000,000.00
13	Amount of Tax Overpaid Last Year Carried Forward to This Year	0
14	Submitted Date	31-Dec-2021
15	Name of Responsible Person	Hui Yi
16	Position	Director

Figure 3.13 Original Return File

The Government of the Republic of the Union of Myanmar

Tax Demand

1	Type of Taxpayer	Company
2	Residency	Resident Foreigner
3	Company Name	Hiltop Company Limited
4	Taxpayer Identification Number (TIN)	10472416
5	Date of Commencement of Operation	1-10-2020
6	Business Contact Address	No.4, Sabel Street, Yankin Township, Yangon, Myanmar
7	Office Contact Phone Number	09452702328
8	Contact Email Address	hiltop@gmail.com
9	Industry Code	1313
10	Income Year	2020-2021
11	Business Income	50,000,000.00
12	Net Tax Before Payments	12,500,000.00
13	Total Quarterly Advance Tax Payments	13,000,000.00
14	Amount of Tax Overpaid Last Year Carried Forward to This Year	0
15	Total Allowable Payments Made During the Year(enter the sum of line 13 and line 14)If no payments made, enter zero.	13,000,000.00
16	Balance Due (Subtract line 12 from line 15). If zero or less, enter zero.	0
17	Penalty	0
18	Remaining Tax to Be Paid (Add line 16 and line 17)	0
19	Amount Overpaid	500,000.00
20	Return Summitted Date	31-Dec-2021
21	Demand Summitted Date	15-Mar-2022
22	Due Date for Tax Payment	15-Apr-2022

Figure 3.14 Sample Demand File

When this demand file is uploaded with the secret key and then encrypted, the output values of ciphertext is received. The received values are shown in figure 3.15 (a) and (b).

255044462d312e370d0a25b5b5b5b50d

0a312030206f626a0d0a3c3c2f547970

c42c2f71e23591499af2b2ccea838cea

268741f3e1226e540e399eac9c603ea0

6bdea27c370987b5711c51f23ae7056f

4e0b57176ab16af42388f36bc5b296a6

572d916057bc952e90501234b7768d50

d4aa8512cc6aace942e30a322b2a6035

f1e53674ae00b376222e72ae50662d5a

a14a1d6b95598e26f4bc00d49aa89de0

6ae11ce7903611805b13b513e43a5436

c2a2dfaa086db35cdf1d182f4f538cf3

3c8be895a4528be66883346209b83451

c12a82f7b78feaf1c8185d46e66a3d71

b73931ae7c84151bfd50e2e4cadbac16

34bb1a9104bd12ed387e352234c1fa66

2886355234c55a9b1a06f5160d742624

8c36818026d9be8c61592f6358b62f33

a459f9d3c7861536112ac8b068459a5c

0bc97eace779613c3115c2122b844833

372db7d77a3962ffa05f66f6e1ea35fa

5d203e3e0d0a7374617274787265660

0a33383030300d0a2525454f460d0a78

6f6f742031203020522f496e666f2031

35203020522f49445b3c443433433436

45363041333942344635413 53e5d202f

5072657620333830303 02f5852656653

74787265660d0a34323037390d0a2525

Figure 3.15 (a) Ciphertext of Demand File

0a3030303030313039393320303030303

30206e0d0a3030303030313130363120

3030303030206e0d0a30303030303333

3632392030303030206e0d0a303030

303033333838352030303030206e0d

0a3030303030333432303920303030303

30206e0d0a3030303030333733353820

3030303030206e0d0a30303030303337

343034203030303030206e0d0a747261

696c65720d0a3c3c2f53697a65203139

362f526f6f742031203020522f496e66

6f203135203020522f49445b3c443433

433436343143393931303734334135 31

33453630413339423446354135 3e3c44

343343343634314339393130373433 41

353133453630413339423446354135 3e

5d203e3e0d0a73746172747872656 60d

0a33383030300d0a2525454f460d0a78

7265660d0a3020300d0a747261696c65

720d0a3c3c2f53697a65203139362f52

6f6f742031203020522f496e666f2031

35203020522f49445b3c443433433436

343143393931303734334135 31334536

304133394234463541353e3c44343343

343634314339393130373433413531 33

453630413339423446354135 3e5d202f

507265762033383030302f5852656653

746d2033373430343e3e0d0a73746172

Figure 3.15 (b) Ciphertext of Demand File

When this demand file is downloaded with the secret key and then decrypted, the original demand file is received. The received file is shown in figure 3.16.

The Government of the Republic of the Union of Myanmar

Tax Demand

1	Type of Taxpayer	Company
2	Residency	Resident Foreigner
3	Company Name	Hiltop Company Limited
4	Taxpayer Identification Number (TIN)	10472416
5	Date of Commencement of Operation	1-10-2020
6	Business Contact Address	No.4, Sabel Street, Yankin Township, Yangon, Myanmar
7	Office Contact Phone Number	09452702328
8	Contact Email Address	hiltop@gmail.com
9	Industry Code	1313
10	Income Year	2020-2021
11	Business Income	50,000,000.00
12	Net Tax Before Payments	12,500,000.00
13	Total Quarterly Advance Tax Payments	13,000,000.00
14	Amount of Tax Overpaid Last Year Carried Forward to This Year	0
15	Total Allowable Payments Made During the Year(enter the sum of line 13 and line 14)If no payments made, enter zero.	13,000,000.00
16	Balance Due (Subtract line 12 from line 15). If zero or less, enter zero.	0
17	Penalty	0
18	Remaining Tax to Be Paid (Add line 16 and line 17)	0
19	Amount Overpaid	500,000.00
20	Return Summitted Date	31-Dec-2021
21	Demand Summitted Date	15-Mar-2022
22	Due Date for Tax Payment	15-Apr-2022

Figure 3.16 Origianl Demand File

CHAPTER 4

DESIGN AND IMPLEMENTATION OF THE SYSTEM

4.1 Implementation of The System

The important data or information can be protected with cryptographic system. The proposed system consists of two parts: Admin side and Taxpayer side. The system makes security on the data to be protected from attack of hacker between admin and taxpayer. The taxpayer can create the desired return that consists of secret data are encrypted by using Blowfish algorithm. When the data encryption finished with the selected password, the encrypted values are uploaded to the e-filling page. That process is data uploading with Blowfish encrypted data. At the admin side, encrypted values are received. The receiving encrypted values is computed by Blowfish algorithm. Data are decrypted with the selected symmetric password by using Blowfish. If two password values are not equal, encrypted data is rejected. Admin's process is not valid. Otherwise, the admin can get decrypted data with the secure status.

4.2 Taxpayer's Side

There has been so many cases about people's mail got hacked because the carelessly entered the account. Therefore, the proposed system controls the security in two-phase checking.

First phase is user authentication (email and password). So, the taxpayer who wants to enter the proposed system must be entered by passing the accurate email and password.

Although the user authentication phase is successful, the login taxpayer must decrypt the secure encrypted data to read, secondly.

After starting the application, the login page of the system is appeared as in Figure 4.1.

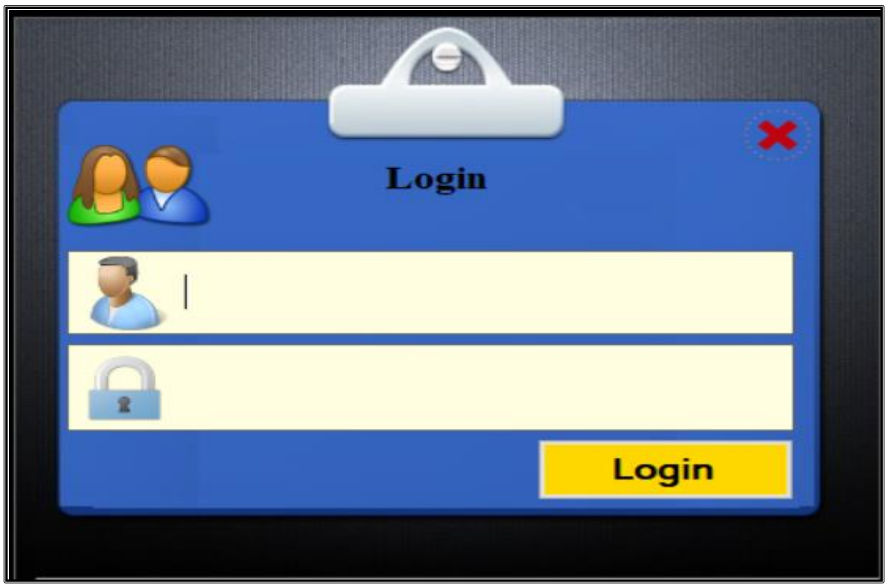


Figure 4.1 Login Page

When the taxpayer login by passing the accurate email and password, the Main page is appeared on the screen as in Figure 4.2.

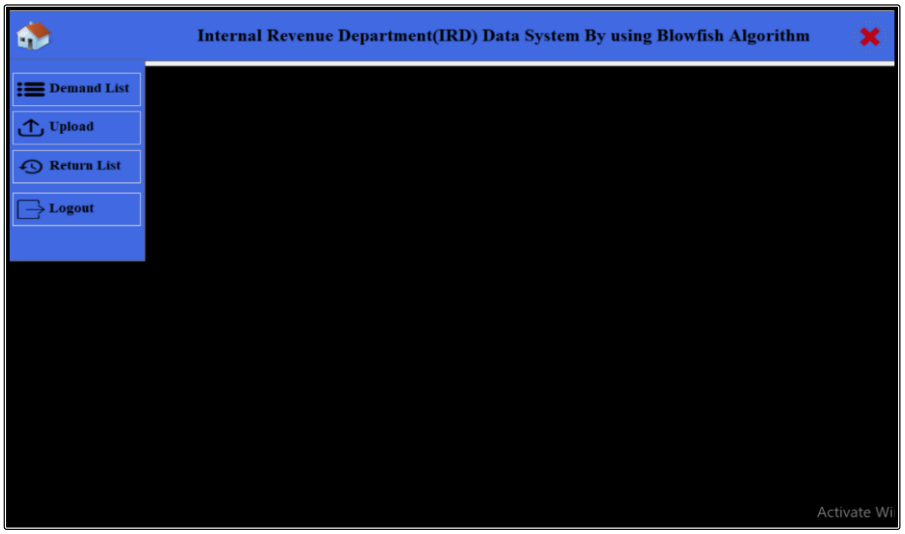


Figure 4.2 Main Page

When the taxpayer clicks the home button on the main page, buttons that the taxpayer can operate will appear as in Figure 4.2. It includes four buttons.

If the taxpayer clicks the first button, all of his/her related demand file will appear as in Figure 4.3. In this page, the taxpayer can download this demand file by using secret key.

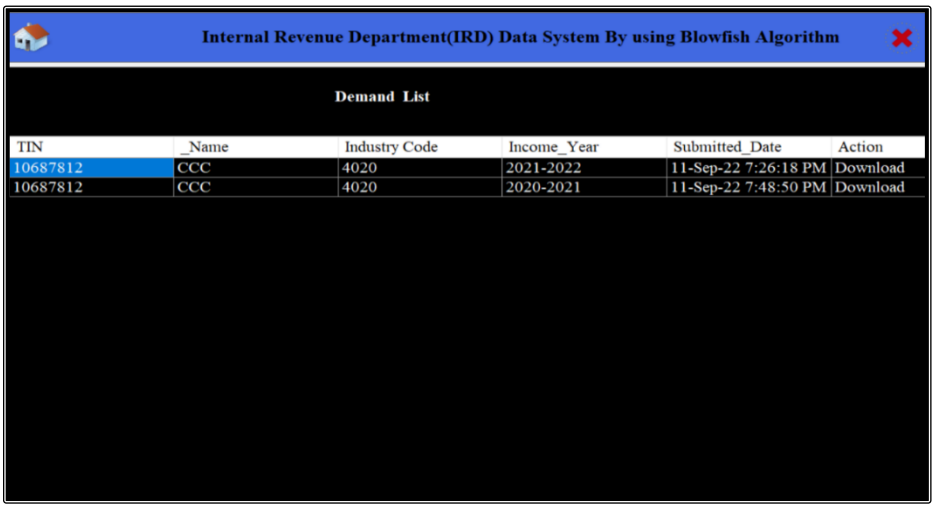


Figure 4.3 Demand List Page

When the taxpayer clicks the second button, he/she needs to download return form before the return file is uploaded. Then, he/she need to fill the information about the return file and upload to e-filling page as shown in Figure 4.4. When all fields are completed and correctly filled, the taxpayer must click the Save button to upload the return file. If the taxpayer fills incomplete data, a message alert will be shown as in Figure 4.5.

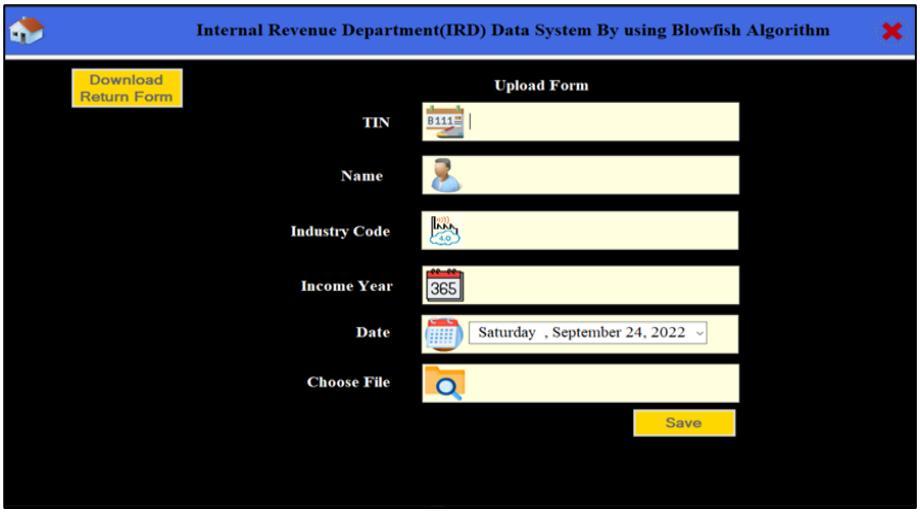


Figure 4.4 Upload Return Page

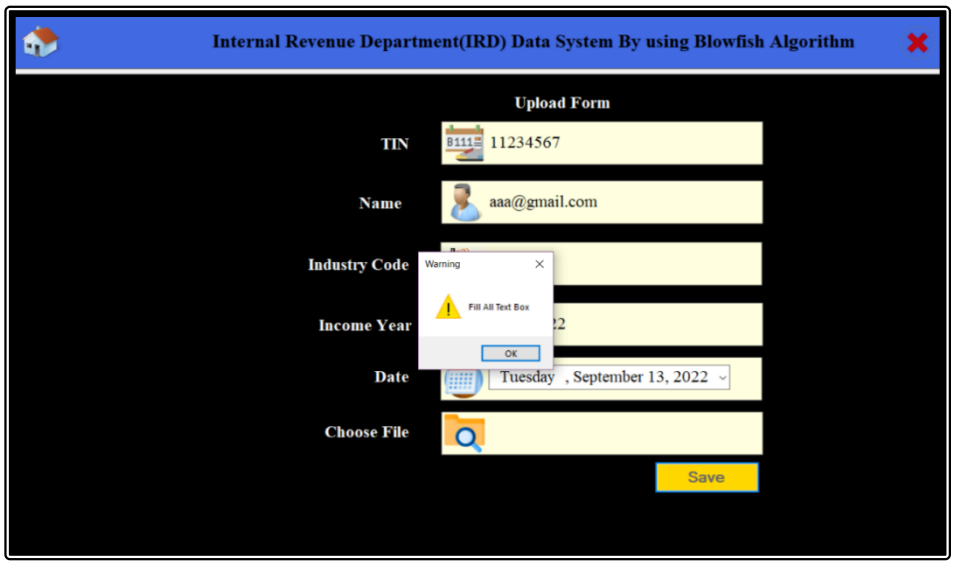


Figure 4.5 Message Alert When the Taxpayer Fill Incomplete Data

The taxpayer can see uploaded files when he/she click the Return List button as shown in Figure 4.6. In this page, the taxpayer can download the related return file by using secret key.

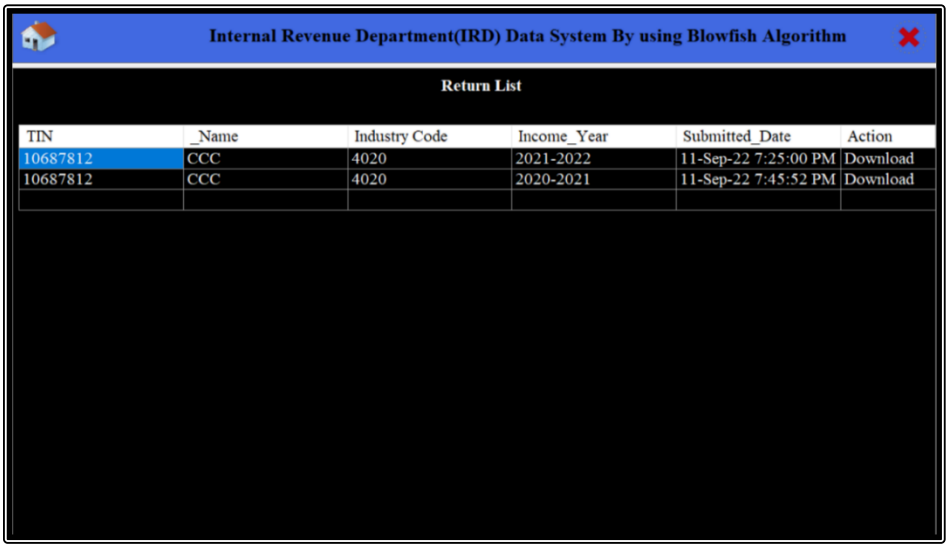


Figure 4.6 Return List Page

When the taxpayer wants to leave out from the system, he/she can click the Logout button. After clicking the Logout button, he/she will reach the Login page of the system.

4.3 Admin's Side

After starting the application, the login page of the system is appeared as in Figure 4.7.

Figure 4.7 Login Page

When the admin login by passing the accurate email and password, the Main page is appeared on the screen as in Figure 4.8.

When the admin clicks the home button on the main page, buttons that the admin can operate will appear as in Figure 4.8. It includes five buttons.

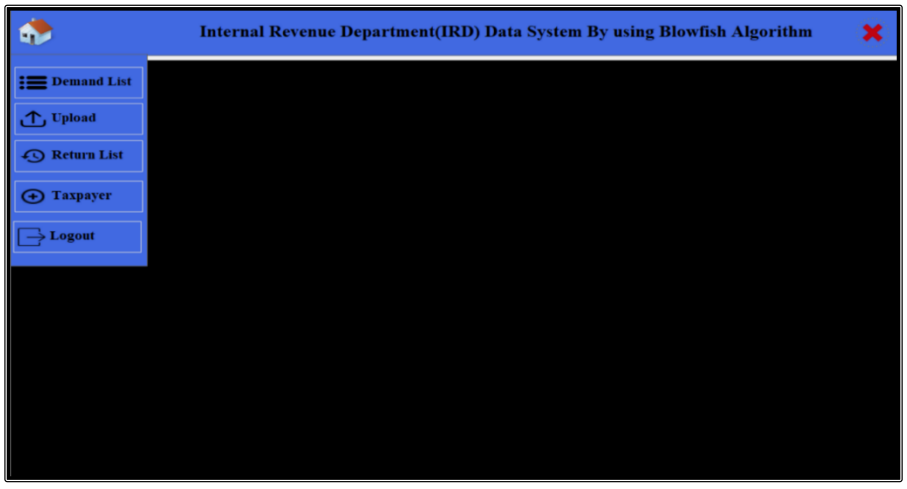


Figure 4.8 Main Page

The admin can see uploaded demand files when he/she click the Demand List button as in Figure 4.9. In this page, the admin can download the desired demand by using secret key.

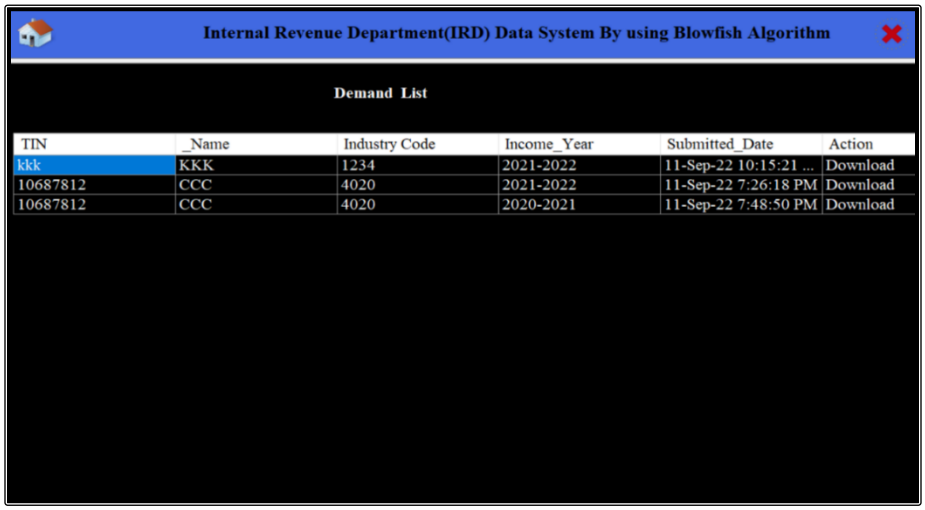


Figure 4.9 Demand List Page

When the admin clicks the second button, he/she will need to fill the data for the desired demand to upload as shown in Figure 4.10. When all fields are completed and correctly filled, the admin must click the Save button to upload the demand file that consists of secret data are encrypted by using Blowfish algorithm. If the admin fills incomplete data, a message alert will be shown as in Figure 4.11.

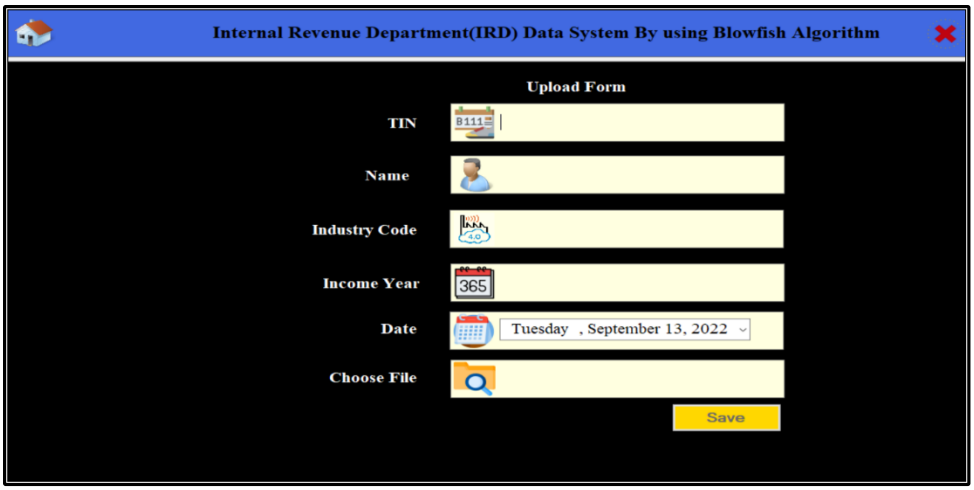


Figure 4.10 Upload Demand Page

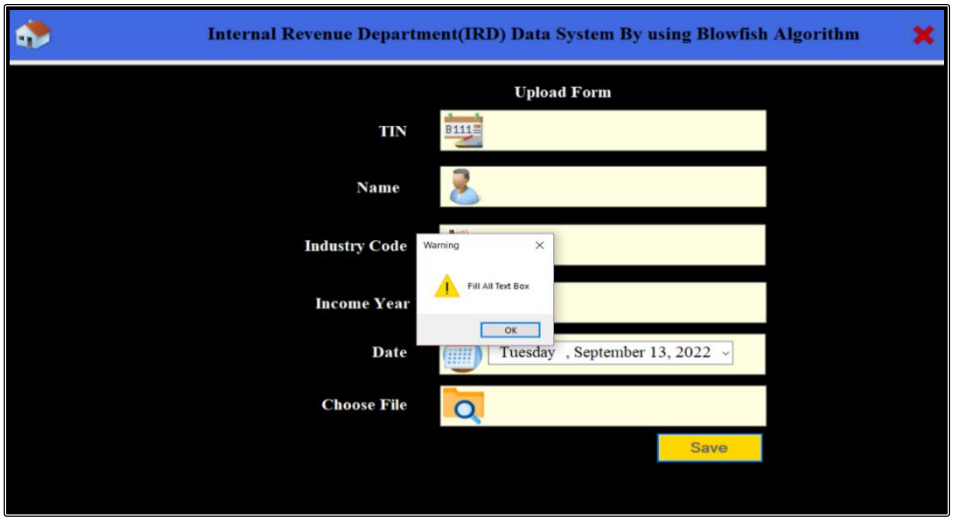


Figure 4.11 Message Alert When the Admin Fill Incomplete Data

If the admin clicks the Return List button, all of return files that uploaded by taxpayers will appear as shown in Figure 4.12. In this page, the admin can download the desired return file to view by using secret key.

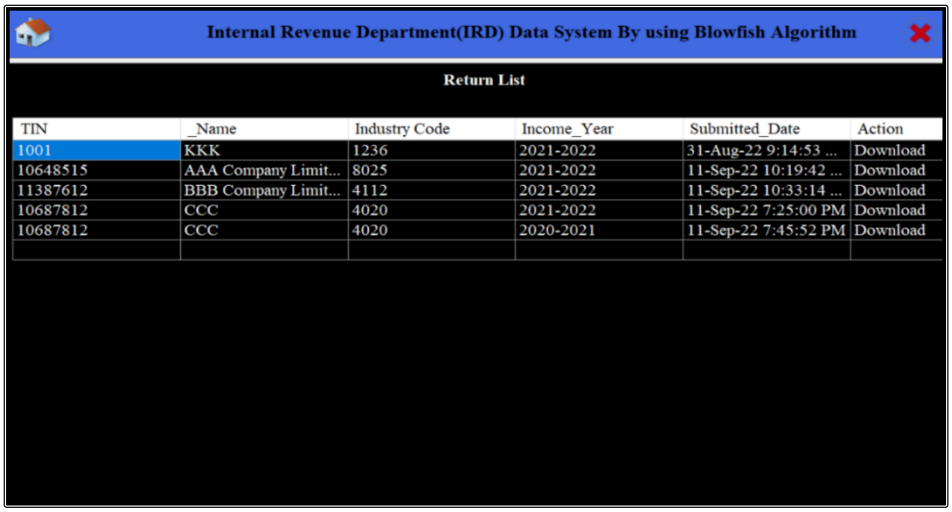


Figure 4.12 Return List Page

As the system only allows the registered user to enter the system, the new user must be registered to use the proposed system.

In this proposed system, the admin will register for the taxpayers. So, the admin must be created the following personal information in the registration page as shown in

Figure 4.13. If the admin fills incomplete data, a message alert will be shown as in Figure 4.14.

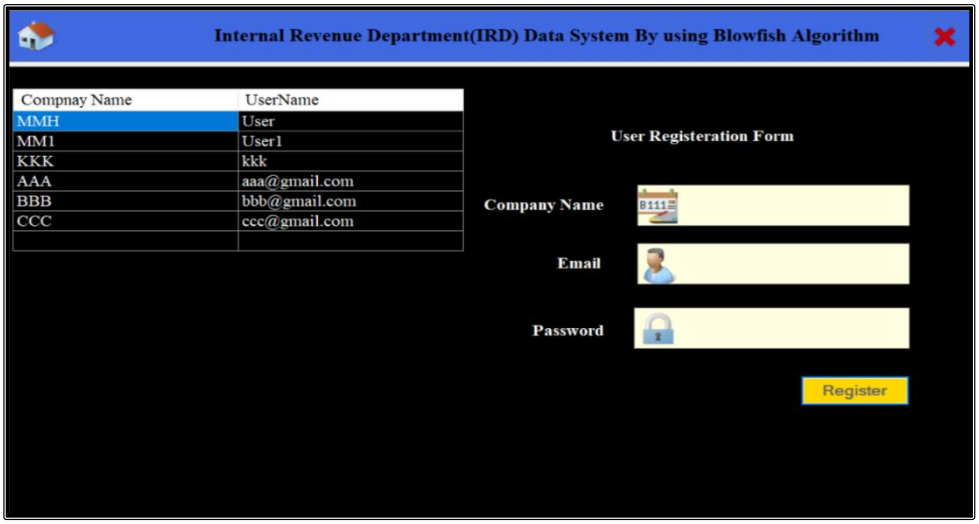


Figure 4.13 Registration Page

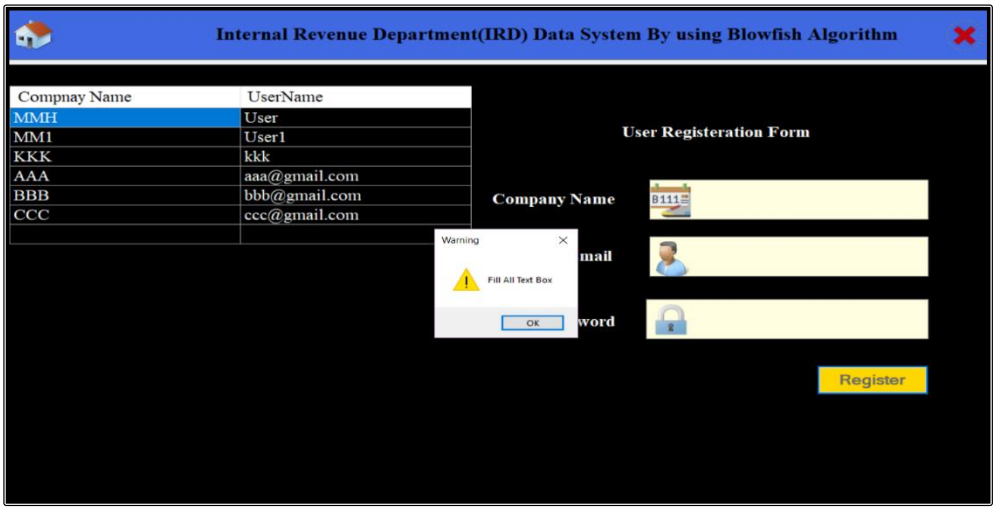


Figure 4.14 Message Alert When the Admin Fill Incomplete Data

When the admin wants to leave out from the system, he/she can click the Logout button. After clicking the Logout button, he/she will reach the Login page of the system.

4.4 Experimental Results of the System

This section presents encryption time and decryption time of various types of input PDF files. The processing time is measured by capturing the time difference between the starting point and ending point of the algorithm. Any change to input file in transit will result in a different time. This result is implemented using C# programming language and the execution of the developed tool on a personal computer equipped with an Intel ® corei7 1.8 GHz CPU, 8G RAM, Window 10 operating system.

This Figure 4.15 shows in runtime of encryption at various file size.

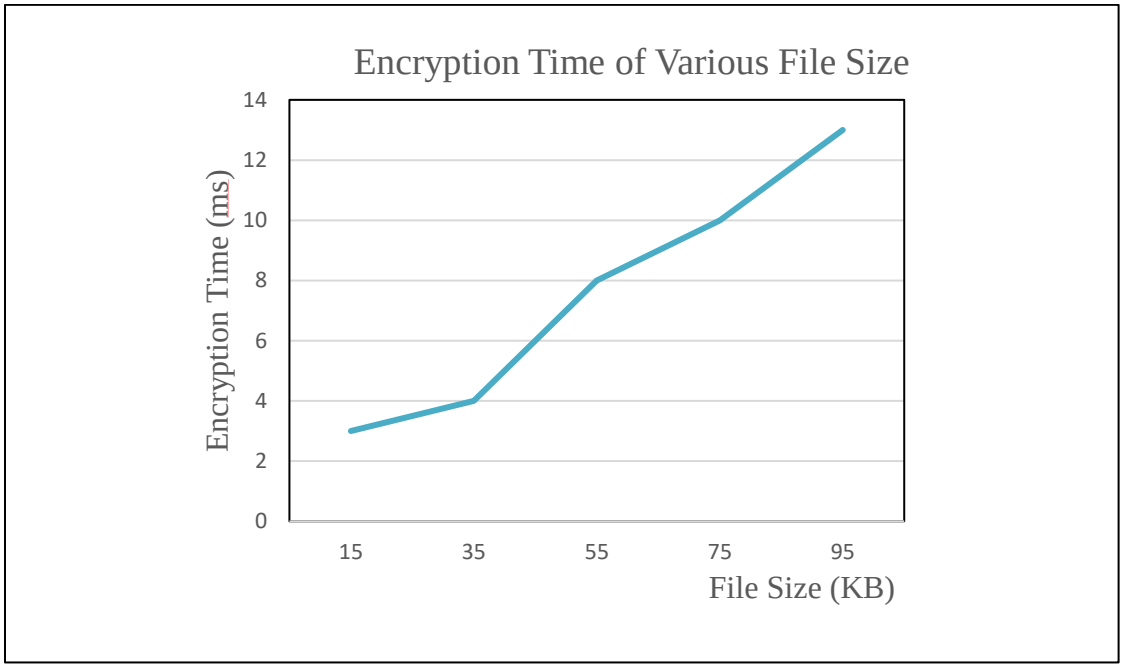


Figure 4.15 Encryption Time of Various File Size

The Figure 4.16 shows in runtime of decryption at various file size.

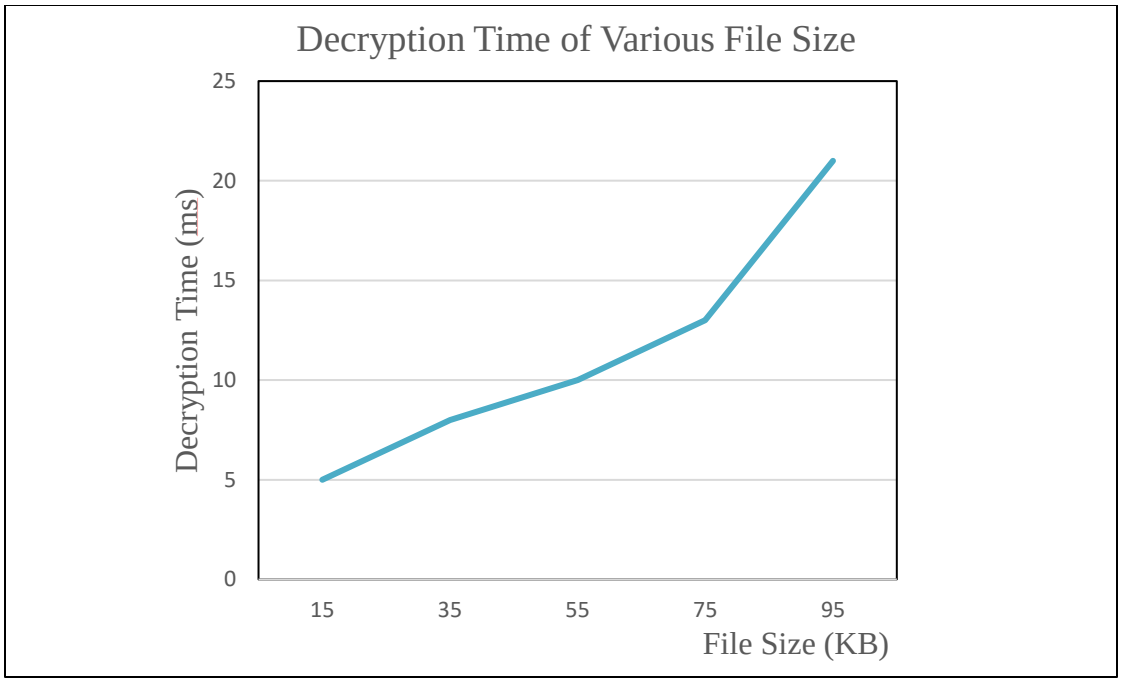


Figure 4.16 Decryption Time of Various File Size

Table 4.1 shows various file sizes which is used in experiment and figure 4.17 shows our experimental results on encryption and decryption time.

File Size (KB)	Encryption Time (Milliseconds)	Decryption Time (Milliseconds)
15KB	3	5
35KB	4	8
55KB	8	10
75KB	10	13
95KB	13	21

Table 4.1 Encryption and Decryption Time of Various File Size

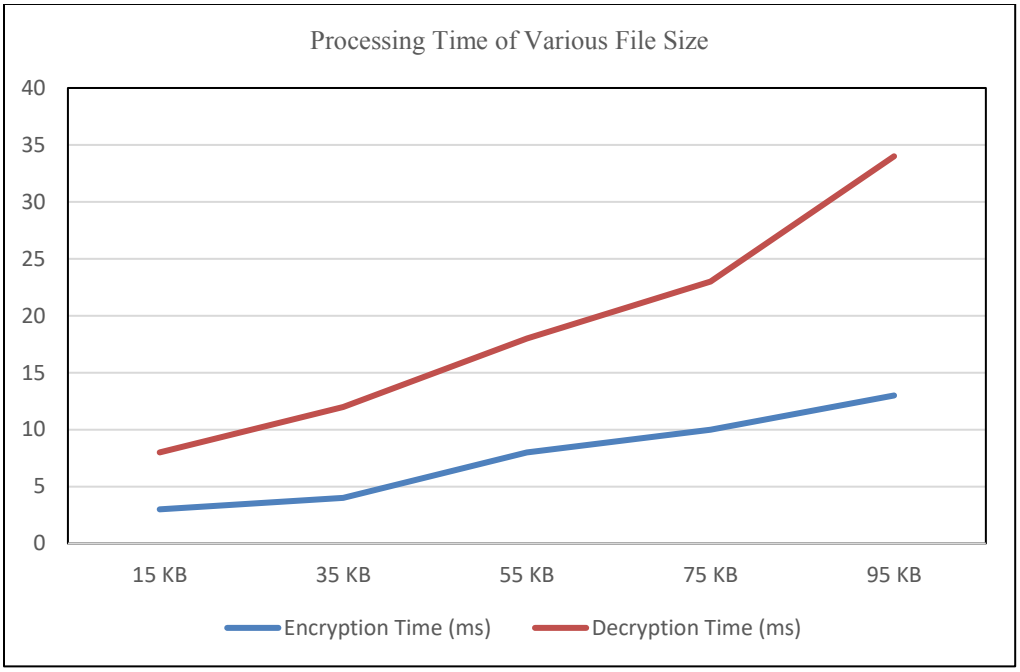


Figure 4.17 Experimental Results on Encryption and Decryption Times

According to this figure, we can conclude that the encryption time of data file is faster than the decryption time in various file sizes.

CHAPTER (5)

CONCLUSION AND FUTURE WORK

5.1 Thesis Summary

The main objective of the thesis is to develop data security system by providing small amount of processing time. In this work, an approach to solve security problem is the use of Blowfish Algorithm.

Design and implementation of this system is described. To meet the requirements of the real-world problem, this thesis proposed the very familiar algorithm in Cryptographic field and transforms this algorithm into desktop application. This system is easy to use for every taxpayer and gives security by using the best features of Blowfish algorithm.

5.2 Limitations

The system can only support the cryptography portion of security of the Blowfish algorithm. The system can only support the encryption of PDF file. The system does not consider about other types of files; for example image or song or video files extensions. The user must have a device that runs on application.

5.3 Conclusion

The encryption algorithm is one of the simplest algorithms to implement in hardware. Blowfish is short program that will run on most machines and encipher safely. It is safe because of the numbers of cycles in the encoding and length of key. It is suitable for embedded systems that require high performance, ease of implementation, high speed, low power consumptions and low cost besides security.

Blowfish is a very efficient data encryption algorithm. It creates 64-bit keys, which are extremely efficient. By employing these encryption approaches, we can encrypt data safely and effectively while also lowering the device's battery consumption. It is possible to improve authentication by increasing the key size.

This paper presents secure data crypto system that allows two peers to exchange encrypted data. Cryptography is a particularly interesting field because of the amount of work that is done in secret. The strength of cryptography lies in the choice (and management) of the keys. In the absence of generally accepted norms which is used to measure and specify cryptographic strength, it is desirable to carry out a number of tests

on different ciphers to get an exposure to their strength and weakness. By using this system, it can save time and effort for the admin and the taxpayers.

5.4 Further Extension

Further work will include experiments with an implementation with RSA algorithm to protect the Blowfish key. The system will become an enveloped model, it means, in the envelop there is a letter which protect with Blowfish algorithm. The key that used in Blowfish is encrypted again with RSA algorithm. So, the admin and the taxpayer will use asymmetric key between end to end and more security can be provided.

AUTHOR'S PUBLICATIONS

[1] Pyae Sandar Win, Yu Wai Hlaing, "Internal Revenue Department (IRD) Data System by Using Blowfish Algorithm", Parallel and Soft Computing (PSC), UCSY, Yangon, Myanmar, 2022.

REFERENCES

[1] A.A.Shtewi, B.E.M.Hasan and A.Hegazy, "An Efficient Modified Advanced Encryption Standard (MAES) Adapted for Image Cryptosystems" Proceeding in "International Journal of Computer Science and Network Secutity (IJCSNS)", Vol.10,No.2,pp.226-232,February 2010.

[2] A.De.Santis, A.Castiglione and U.F.Petrillo, "An Extensible Framework for Efficient Secure SMS" Published in "2010 International Conference on Complex, Intelligent and Software Intensive Systems", pp.843-850,2010.

[3] Agrawal, Monika, and Pradeep Mishra, "A comparative survey on symmetric key encryption techniques" Proceeding in "International Journal on Computer Science and Engineering", Vol.4, No. 5, May 2012.

[4] A.Kaur, "Message Encryption Using NTRU Algorithm on Android" Proceeding in "International Journal of Research in Computer Applications and Robotics", Vol.1, Issue.6, pp.54-57,September 2013.

[5] Anis Cherid, "Asymmetric And Symmetric Cryptography To Secure Social Network Media Communication: The Case Of Android-Based E-Learning Software" Proceeding in "International Research Journal of Computer Science(IRJCS) " , Issue 01,Volume 5,January 2018.

[6] Asassfeh, Mahmoud Rahallah, Mohammad Qatawneh and Feras Mohamed AL-Azzeh, "Performance evaluation of blowfish algorithm on supercomputer iman1" Proceeding in "International Journal of Computer Networks & Communications (IJCNC) " 10, no. 2 (2018).

[7] Ashwak ALabaichi, Faudziah Ahmad, Ramlan Mahmod, "Security Analysis of Blowfish algorithm", ISBN: 978-1-4673-5256-7/13/$31.00 ©2013 IEEE.

[8] Chatterjee, Rishav, Sharmistha Roy, and U. G. Scholar, "Cryptography in cloud computing: a basic approach to ensure security in cloud" Proceeding in Proceeding in International Journal of Engineering Science 11818 (2017).

[9] Chaudhari, Maulik P., and Sanjay R. Patel, "A survey on cryptography algorithms" Proceeding in "International Journal of Advance Research in Computer Science and Management Studies 2", no. 3, March 2014.

[10] G.C.Kessler, "An Overiew of Cryptography" Published by Auerbach "Handbook on Local Area Networks", September 1998.

[11] H.Agrawal and M.Sharma "Implementation and analysis of various symmetric cryptosystems" Proceeding in "Indian Journal of science and Technology", Vol.3, No.12, pp.1173-1176, December 2010.

[12] J. Unni Kiran, P. Sai Kiran, "Secure Communication with Blowfish Cryptography for Data Sharing on Cloud using Android Devices", International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN: 2278-3075, Volume-9, Issue-6, April 2020.

[13] Lin, Michael C-J, and Youn-L.Lin, "A VLSI implementation of the blowfish encryption/decryption algorithm" In Proceedings of the 2000 Asia and South Pacific Design Automation Conference, pp. 1-2, 2000.

[14] Mousa, Allam, "Data encryption performance based on Blowfish" Proceeding in "In 47th International Symposium ELMAR", pp.131-134, IEEE, 2005.

[15] Parihar, Veena, and Mr Aishwary Kulshrestha, "Blowfish algorithm: a detailed study", Proceeding in International Journal For Technological Research In Engineering 3, no. 9 (2016).

[16] Pia Singh, Prof. Karamjeet Singh, "Image Encryption and Decryption Using Blowfish Algorithm in MATLAB, International Journal of Scientific & Engineering Research", ISSN: 2229-5518, Volume 4, Issue 7, July 2013.

[17] Schneier, Bruce, "Description of a new variable-length key, 64-bit block cipher (Blowfish)" Proceeding in " International Workshop on Fast Software Encryption", pp. 191-204. Springer, Berlin, Heidelberg, 1993.

[18] S.Swathi, P.Lahari and B.Thomas, " Encryption Algorithms: A Survey" Proceeding in "International of Advanced Research in Computer Science & Technology (IJARCST) ", Volume 4, Issue 2, 2016.

[19] Vinod D. Rajput, Kajal D. Jaisinghani, "Security in Cloud Computing Using Blowfish Algorithm", In Proceeding of International Journal of Mechanical Engineering, ISSN: 0974-5823, Volume7, Special Issue 5, April-May 2022.

[20] W. Jason Cornwell, "Blowfish Survey", Department of Computer Science, Columbus State University Columbus, GA, 2012.