International Journal of Trend in Scientific Research and Development (IJTSRD)
Volume 3 Issue 5, August 2019 Available Online: www.ijtsrd.com e-ISSN: 2456 – 6470

Performance Comparison of File Security System using TEA and Blowfish Algorithms

Win Myat Thu, Tin Lai Win, Su Mu Tyar

Department of Information Technology, Technological University, Mandalay, Myanmar

How to cite this paper: Win Myat Thu | Tin Lai Win | Su Mu Tyar "Performance Comparison of File Security System using TEA and Blowfish Algorithms" Published in International Journal of Trend in Scientific Research and Development (ijtsrd), ISSN: 2456-6470, Volume-3 | Issue-5, August 2019, pp.871-877, **https://doi.org/10.31142/ijtsrd26462**

IJTSRD26462

ABSTRACT

With the progress in data exchange by the electronic system, the need for information security has become a necessity. Due to the growth of multimedia application, security becomes an important issue of communication and storage of different files. To make its reality, cryptographic algorithms are widely used as essential tools. Cryptographic algorithms provide security services such as confidentiality, authentication, data integrity and secrecy by encryption. Different cryptographic algorithms are commonly used for information security in many research areas. Although there are two encryption techniques, asymmetric and symmetric, the simpler symmetric encryption technique is employed for testing file security system. In this study, the performance evaluation of the most common two symmetric encryption algorithms such as TEA and Blowfish algorithm is focused on the execution time intervals. Simulation has been conducted with many types of file encryption like .pdf, .txt, .doc, .docx, .xlsx, .pptx, .ppt, .xls, .jpg, .png and most common video file formats by using Java Programming Language.

KEYWORDS*: Symmetric, Asymmetric, encryption, TEA, Blowfish*

I. INTRODUCTION

Symmetric key algorithms are used primarily for the encryption of data or data streams. These algorithms are designed to be very fast and have a large number of possible keys.

The best symmetric key algorithms offer excellent secrecy, once data is encrypted with a given key, there is no fast way to decrypt the data without processing the same key. Symmetric key algorithms can be divided into two categories: block and stream. Block algorithm encrypts data many bytes at a time, while stream algorithms encrypt byte by byte or even bit by bit [1].

This paper evaluates two symmetric encryption algorithms namely; TEA and Blowfish. The performance measure of encryption schemes will be conducted in terms of processing time intervals on the Windows platform for the different file size.

With the study of a file security system based on two symmetric algorithms, achieving file secrecy, analyzing data encryption duration, studying TEA and Blowfish algorithms' Feistel structure and understanding how to apply encryption algorithm for the file security system are strongly expected.

The rest of this paper is organized as follows: Section II gives a brief cryptographic strength of symmetric algorithms, section III provides the algorithms that have been chosen for implementation; section IV provides the performance evaluation methodology, section V discusses the simulation results in detail and finally section VI concludes the work.

II. UNDERSTANDING CRYPTOGRAPHIC TECHNIQUES

Cryptography is the study of techniques for ensuring the secrecy and authentication of the information. The sender translates the plaintext into ciphertext. This ciphertext is then sent to the receiver. The authorized receiver gets the ciphertext and then converts the ciphertext back into the original form. The main aim of the cryptography is to protect the information from illegal access. Goals, strength, weakness and basic terminology are expressed in the following section.

The data can be read in its original form is called plain text. The way of mask the plaintext in such a way as to hide its original form is called encryption. The method of encrypting the plaintext which results in unreadable form is called ciphertext. The method of taking encrypted message or data and converting back into its original form is called decryption. An entity which provides encryption and decryption is called cryptosystem [2].

Depending upon the key cryptography can be divided into two categories: symmetric and asymmetric encryption. Symmetric Encryption (private key Encryption) is during the encryption and decryption process the same key is used at the sender and receiver site. Before the transmission of information starts the key distribution has to be made [3]. Example: DES [4]-[6], 3DES [7], BLOWFISH [4], [5], AES [8] etc. However, in asymmetric encryption (Public key encryption, two different keys are used for encryption and decryption process. At the same time, the two keys are generated. In that one key is transferred to the other side before the exchange of information begins [9]. Example: RSA [10], Elgamal, Elgamal signature Diffie Hellman key exchange, digital signature [11].

A. Goals of Cryptography

1. *Confidentiality*: Data that resides in the computer is transmitted and that is to be accessed only by the legal person and that data can't be accessed by anyone else.
2. *Authentication:* The data that is seen by any system has to check the identity of the sender, whether the data appears from a legal person or illegal person.
3. *Data Integrity:* To verify the information, it has not been changed by an illegal or unknown person. Only the sender and receiver can modify the message. No others have the rights to access the message (or) data.
4. *Non-Repudiation:* It does not allow repudiation by the sender or receiver. The receiver proves the identification of the sender in case of denial by the sender. The sender proves the identification of the receiver in case of denial by the receiver.
5. *Access Control:* It ensures that only the authorized person can have the rights to access the transmitted information [12].

B. Cryptographic Strength of Symmetric Algorithms

Symmetric encryption, also referred to as conventional encryption or single key encryption was the only type of encryption in use prior to the development of public-key encryption. The private keys used in symmetric-key cryptography are robustly resistant to brute force attacks. While only the one-time pad, which combines plaintext with a random key, holds secure in the face of any attacker regardless of time and computing power, symmetric-key algorithms are generally more difficult to crack than their public key counterparts. Additionally, secret-key algorithms require less computing power to be created than equivalent private keys in public-key cryptography [13].

C. Cryptographic Weakness of Symmetric Algorithms

The biggest obstacle in successfully deploying a symmetric-key algorithm is the necessity for a proper exchange of private keys. This transaction must be completed in a secure manner. In the past, this would often have to be done through some type of face-to-face meeting, which proves quite impractical in many circumstances when taking distance and time into account. If one assumes that security is a risk to begin with due to the desire for a secret exchange of data in the first place, the exchange of keys becomes further complicated.

Another problem concerns the compromise of a private key [14]. In symmetric-key cryptography, every participant has an identical private key. As the number of participants in a transaction increases, both the risk of compromise and the consequences of such a compromise increase dramatically. Each additional user adds another potential point of weakness that an attacker could take advantage of. If such an attacker succeeds in gaining control of just one of the private keys in this world, every user, whether there are hundreds of users or only a few, are completely compromised [15].

D. Basic Terminology Used in Cryptography

The symmetric encryption scheme has five ingredients.

1. *Plaintext*: This is the original intelligible message or data that is fed to the algorithm as input.
2. *Ciphertext*: This is the scrambled message produced as output. It depends on the plaintext and the key. The ciphertext is an apparently random stream of data, as it stands, is unintelligible.
3. *Encryption Algorithm*: The encryption algorithm performs various substitutions and permutations on the plaintext.
4. *Decryption Algorithm*: This is essentially the encryption algorithm that runs in reverse. It takes the ciphertext and the secret key and produces the original plaintext.
5. *Secret Key*: The secret key is also input to the encryption algorithm. The exact substitutions and permutations performed depend on the key used, and the algorithm will produce a different output depending on the specific key being used at the time [16].

III. COMPARED ALGORITHMS

Symmetric encryption algorithms: Blowfish and TEA are compared for the execution time of various file types encryption and decryption.

A. Blowfish

Blowfish is a symmetric block cipher. It is used for encrypting and protecting the data. It has a variable-length key range from 32 bits to 448 bits, for safeguarding our data. It was designed in 1993 by Bruce Schneier. It is a license for encryption method and it is freely available to all users. It is mainly used for applications, such that key does not change often, like a communication link [17].

1. Blowfish Algorithm

- It has large data blocks.
- It consists of a 64-bit block size.
- The range of key scalable from 32 bits to 256 bits.
- It uses very simple operation which is efficient for microprocessors.
- It has a variable number of iteration.
- It uses subkey which is one way hash of the key.
- It has no linear structure.
- Its design structure is simple to understand. It increases the confidence in the algorithm. It is a fiestel iterated block cipher [12].

2. Fiestel Network: Fiestel Network is a common method of converting any function into a permutation. The working procedure of fiestel network:

- It split the block into two halves
- Now, the right half becomes the new left half.
- If the left half is XOR'd with the result of applying 'f' to the right half and the key, we have the new right half as the final result.
- Even the function f and not invertible the previous rounds can be derived [12].

3. Block Diagram of the Blowfish Algorithm

Blowfish is a symmetric key Feistel structured algorithm consisting of two parts: key expansion part and data-encryption part. It is one of the most public-domain encryption algorithms. It takes a variable-length key from 32-bits to 448-bits, permuted into 18 sub-keys each of 32-bit length and can be implemented on 32 or 64-bit processor [18] as shown in Fig. 1.

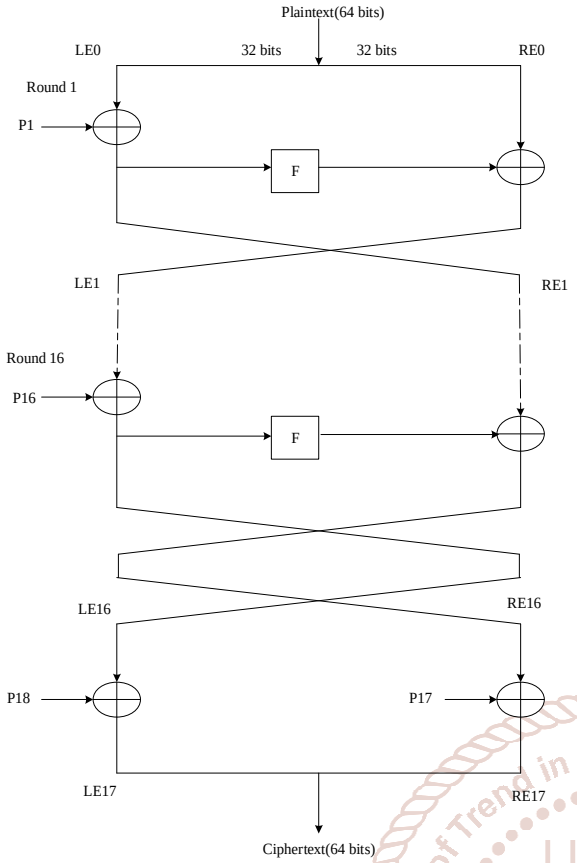

Fig.1 Block diagram of Blowfish algorithm

A. TEA

Tiny Encryption Algorithm (TEA) block cipher was designed with speed and simplicity in mind. It is a variant of the Feistel Cipher. TEA operates on a 64-bit block of data that is then split up into two 32 bit unsigned integers during the encryption process. TEA uses a 128-bit key, and a magic constant is also utilized which is defined as 2^32/ (the golden ratio) [19].

The original TEA was written by Roger Needham and David Wheeler and was first presented in 1994. Tiny Encryption Algorithm is a Feistel type cipher that uses operations from mixed (orthogonal) algebraic groups. A dual shift causes all bits of the data and key to be mixed repeatedly. The key schedule algorithm is simple; the 128-bit key K is split into four 32-bit blocks K = (K [0], K [1], K [2], K [3]). TEA seems to be highly resistant to differential cryptanalysis and achieves complete diffusion. Time performance on a workstation is very impressive. The inputs to the encryption algorithm are a plaintext block and a key K.The plaintext is P = (Left[0], Right[0]) and the ciphertext is C = (Left[64], Right[64]). The plaintext block is split into two halves, Left[0] and Right[0]. Each half is used to encrypt the other half over 64 rounds of processing and then combine to produce the ciphertext block [20].

4. Block Diagram for TEA Algorithm

TEA uses a key size of 128 bits and a block size of 64 bits. It is very nearly a Feistel cipher, although addition modulo 2^32 is used to combine the round function with the block rather than addition modulo 2. This means that the decryption function is slightly different from the encryption function, although both are so simple that the difference is not generally problematic. The Feistel rounds are grouped into pairs, called cycles. Many other Feistel ciphers, in the last round the last swap is done, just like in the other rounds. This again makes decryption in reverse, slightly different from encryption shown in Fig. 2, but since a separate implementation for decryption is necessary anyway the difference is worthwhile to simplify the description [21].

Decryption is the same except that the two additions modulo 2^32 at the left-hand side are replaced with subtractions modulo 2^32, and the swaps are done at the start of the round instead of the end [21].

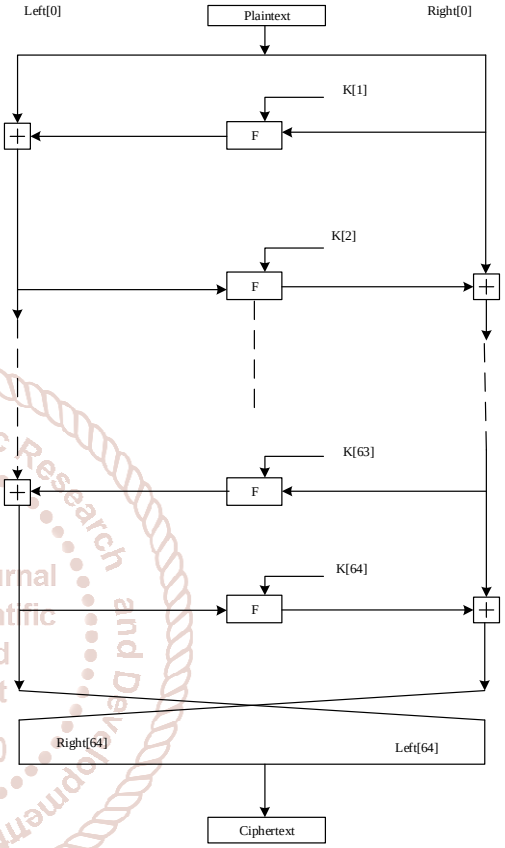

Fig.2 Blocked encryption for TEA algorithm [22]

IV. PERFORMANCE EVALUATION METHODOLOGY

Performance evaluation on which hardware and software requirement, which IDE is used to implement which criteria are emphasized, which types of file are tested and how the system works are detail expressed in the following sections.

A. System Parameters

This encryption and decryption of TEA and Blowfish algorithms were implemented with Java in IDE. The performance was measured on Intel(R) Pentium (R) CPU P6000 @1.87GHz 1.87GHz, 64-bit operating system with 1.00 GB of RAM running Window 10 Pro.

B. Experiment Factors

The variety of input files size from 85 Kbytes to 2600 Kbytes. Comparison of TEA and Blowfish execution time are calculated in milliseconds. The result of encryption time is measured when an encryption algorithm takes to make a cipher content from plaintext and the revert manner. By isolating the total plaintext in megabytes encrypted on the whole encryption time for each algorithm, the amount of an encryption scheme is calculated. Encryption process for both symmetric encryption algorithms is shown in Fig. 3 and in the meanwhile the decryption process is also shown in Fig. 4.

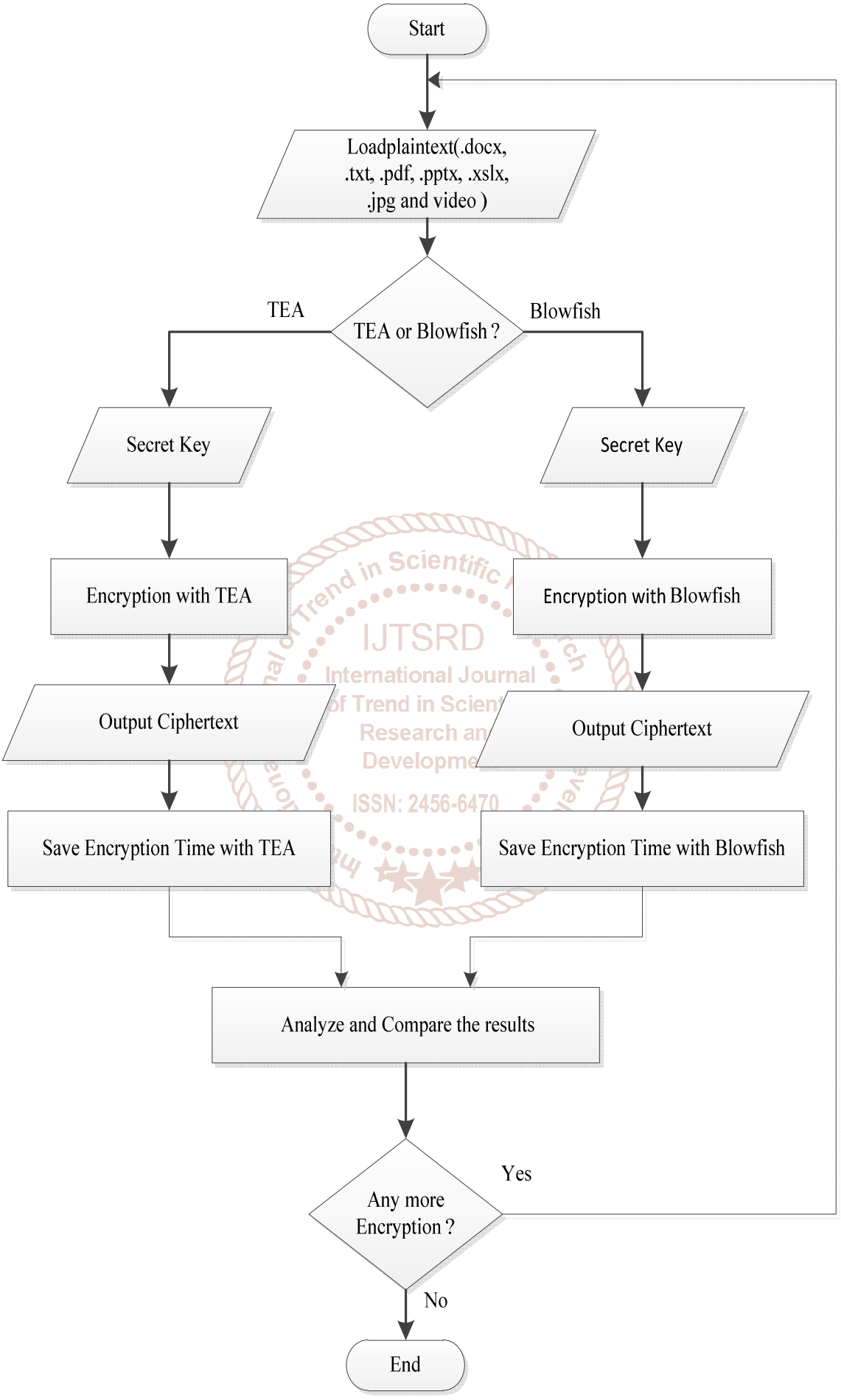

Fig.3 Encryption process

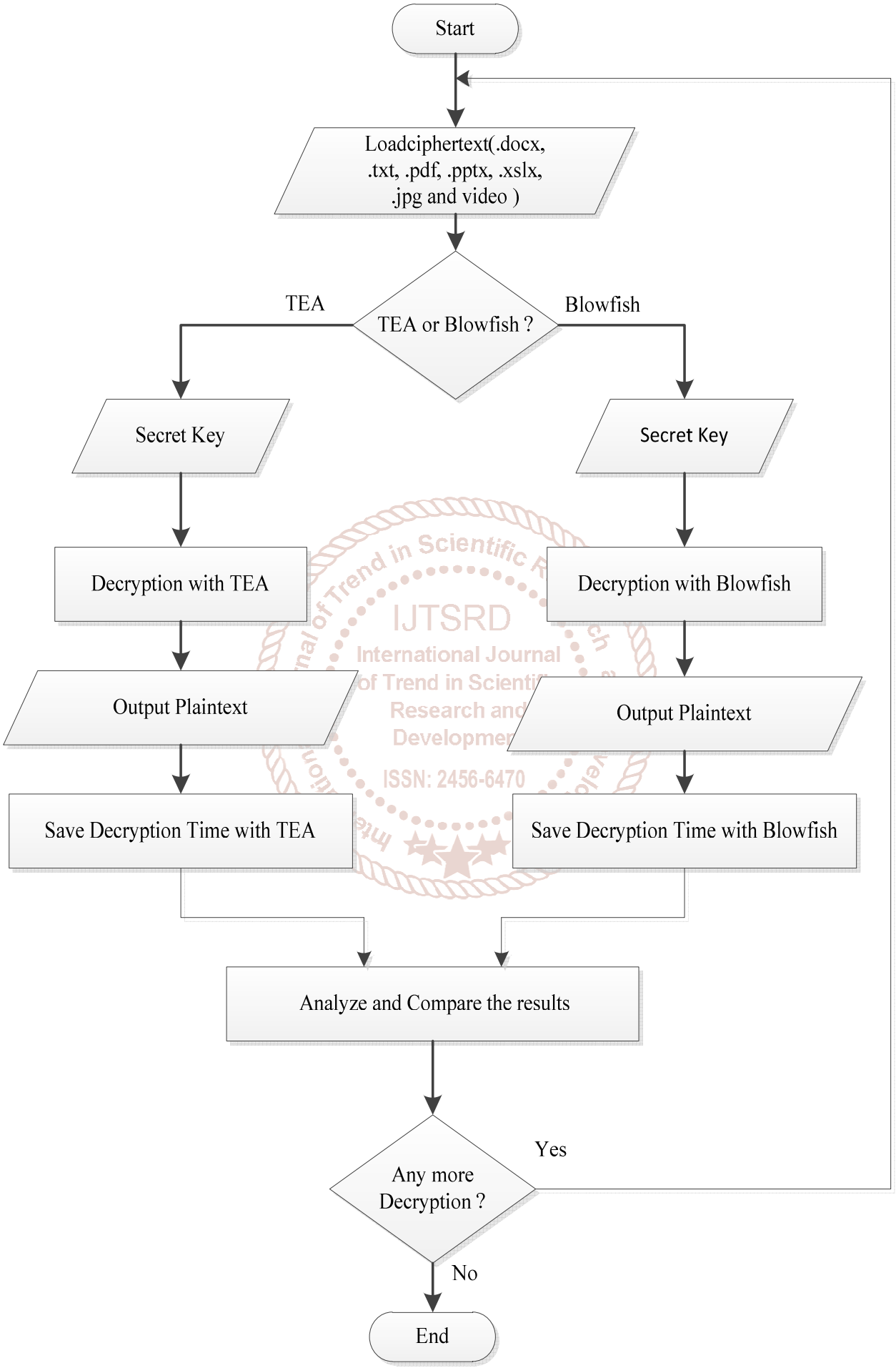

Fig.4 Decryption process

V. SIMULATION RESULTS

The overall performance of the encryption algorithm is assessed considering the above-mentioned parameters together with encryption time and decryption time. The encryption time is considered because of the time that an encryption algorithm takes to deliver a ciphertext from a plain textual content. Comparative analyses of the outcomes of the chosen and distinctive encryption scheme are performed as shown in table 1 to 14.

A. Comparative Results of TEA and Blowfish algorithm with Different File Types

The blowfish algorithm performs better compared with TEA regarding time consumption for both encryption and decryption scheme. Table 1 to 14 show the encryption and decryption stage including file sizes and their run times. Total seven types of files are compared and although any input file sizes are able to be tested, only three approximate file sizes are mentioned in this paper. 85 KB to 2600 KB files is tested. In experiment analysis, encryption time table, decryption time table are illustrated respectively for seven different file format.

TABLE I INPUT SIZE AND ENCRYPTION TIME FOR PDF FILE

Input Size (in Kbytes)	TEA (ms)	Blowfish(ms)
140	8	13
296	17	26
2600	454	644

TABLE II INPUT SIZE AND DECRYPTION TIME FOR PDF FILE

Input Size (in Kbytes)	TEA (ms)	Blowfish(ms)
140	7	14
296	15	31
2600	523	584

TABLE III Input Size and Encryption Time for Document File

Input Size (in Kbytes)	TEA (ms)	Blowfish(ms)
154	9	11
458	26	40
1200	71	74

TABLE IV Input Size and Decryption Time for Document File

Input Size (in Kbytes)	TEA (ms)	Blowfish(ms)
154	8	9
458	24	47
1200	66	72

According to the experimental results, decryption process is exactly the same as encryption but execution time.

TABLE V Input Size and Encryption Time for Excel File

Input Size (in Kbytes)	TEA (ms)	Blowfish(ms)
90	6	23
60	4	6
135	10	30

TABLE VI INPUT SIZE AND DECRYPTION TIME FOR EXCEL FILE

Input Size (in Kbytes)	TEA (ms)	Blowfish (ms)
90	3	10
60	3	7
135	8	12

TABLE VII INPUT SIZE AND ENCRYPTION TIME FOR POWERPOINT FILE

Input Size (in Kbytes)	TEA (ms)	Blowfish (ms)
350	21	28
612	35	55
1250	68	82

TABLE VIII INPUT SIZE AND DECRYPTION TIME FOR POWERPOINT FILE

Input Size (in Kbytes)	TEA (ms)	Blowfish (ms)
350	20	21
612	31	63
1250	63	66

TABLE IX INPUT SIZE AND ENCRYPTION TIME FOR JPG FILE

Input Size (in Kbytes)	TEA (ms)	Blowfish (ms)
85	3	4
153	8	10
319	18	28

TEA is being considered to be a standard encrypting algorithm since no specified weaker points are found so far in comparing to the Blowfish algorithm for both encryption and decryption processes.

TABLE X INPUT SIZE AND DECRYPTION TIME FOR JPG FILE

Input Size (in Kbytes)	TEA (ms)	Blowfish (ms)
85	4	5
153	7	15
319	17	33

TABLE XI INPUT SIZE AND ENCRYPTION TIME FOR MP4 FILE

Input Size (in Kbytes)	TEA (ms)	Blowfish (ms)
955	54	87
1922	108	111
2422	147	209

TABLE XII INPUT SIZE AND DECRYPTION TIME FOR MP4 FILE

Input Size (in Kbytes)	TEA (ms)	Blowfish (ms)
955	49	97
1922	97	165
2422	126	250

TABLE XIII INPUT SIZE AND ENCRYPTION TIME FOR TEXT FILE

Input Size (in Kbytes)	TEA (ms)	Blowfish (ms)
98	5	9
125	15	19
250	22	27

TABLE XIV INPUT SIZE AND DECRYPTION TIME FOR TEXT FILE

Input Size (in Kbytes)	TEA (ms)	Blowfish (ms)
98	6	10
125	28	30
250	29	31

It turned into concluded that the TEA encrypting algorithm performs in a much efficient way for encrypting and decrypting the confidential data. Hence TEA works better for secure file system application.

VI. CONCLUSION

Many data from where such as military, hospital, bank and business need security while communicating between people and activities. Symmetric encryption algorithms can

solve the problems of information security with the security keys. Symmetric key algorithms are faster and easier to be implemented than the asymmetric key algorithms. This system proves which algorithm is appropriate for each file type. Therefore, different file types: video file, image file, text file, portable document format file, word file, powerpoint file and excel file are encrypted with TEA and Blowfish symmetric encryption algorithms to secure and the execution durations are also compared with three different file sizes. Altogether, various file secrecy and symmetric encryption knowledge are achieved and TEA is preferred for file security.

ACKNOWLEDGMENT

We wish to express our sincere gratitude to Dr. Tin Lai Win and Dr. Su Mu Tyar. We also thank all of our teachers for providing us the opportunity to embark on this study.

REFERENCES

[1] (2018) The eTutorials website. [Online]. Available: http://etutorials.org/Linux+System/unix+internet+security/.

[2] W. Stallings, *Cryptography and Network Security Principles and Practices,* 4th ed., Pearson Education, Prentice-Hall, 2009.

[3] Himani Agrawal and Monisha Sharma, "Implementation and analysis of various Symmetric Cryptosystems", *Indian Journal of Science and Technology,* vol.3, no.12, 2012.

[4] P. C. Mandal, "Evaluation of the performance of the Symmetric Key Algorithms: DES, 3DES, AES and Blowfish," *Journal of Global Research in Computer Science, e*-ISSN 2229-371X, vol. 3, no. 8, Aug. 2012.

[5] J. Thakur, N. Kumar "DES, AES and Blowfish: Symmetric Key Cryptography Algorithms Simulation-Based Performance Analysis," *International Journal of Emerging Technology and Advanced Engineering*, ISSN 2250-2459, pp. 6-12, vol.1, Issue 2, Dec. 2011.

[6] M. Agrawal, P. Mishra "A Comparative Survey on Symmetric Key Encryption Techniques," *International Journal on Computer Science and Engineering (IJCSE)*, vol.4, no. 05, pp.877-882, May. 2012.

[7] S.Pavithra, Mrs. E. Ramadevi "Study and Performance Analysis of Cryptography Algorithms," *International Journal of Advanced Research in Computer Engineering & Technology, v*ol.1, Issue 5, pp.82-86, July. 2012.

[8] Shanta, Y. Vashishtha, "Evaluating the performance of Symmetric Key Algorithms: AES (Advanced Encryption Standard) and DES (Data Encryption Standard)," *International Journal of Computational Engineering & Management (IJCEM*), vol. 15, Issue 4, pp.43-49, July. 2012.

[9] Manoj Kumar Pandey, et. all, "Survey Paper: Cryptography The art of Hiding Information", *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)*, ISSN: 2278 – 1323, vol.2, no.12, 2013.

[10] M, Vishwakarma, "Comparative study of Cryptography Algorithms", *International Journal of Advanced Research in Computer Science*, ISSN: 0976-5697, vol.4, no. 3, Special Issue, March. 2013.

[11] Kellogg S. Booth, "*Authentication of signatures using public-key encryption*," Communications of the ACM, pp. 772-774, Nov. 1981.

[12] S.Suguna1, Dr.V.Dhanakoti2, R. Manjupriya3, "A Study on Symmetric and Asymmetric Key Encryption Algorithms." *International Research Journal of Engineering and Technology (IRJET),* e-ISSN: 2395 - 0056, vol.3 Issue.4, pp. 28, Apr.2016.

[13] S. Halevi and H. Krawczyk, "*Public-key cryptography and password protocols,"* ACM Transactions on Information and System Security, pp. 230-268, Aug. 1999.

[14] W. Küchlin, "*Public key encryption*," ACM SIGSAM Bulletin, pp. 6973, Aug. 1987.

[15] M. Blumenthal, "Encryption: Strengths and Weaknesses of Public-key Cryptography" Univ. of Villanova, Department of Computing Sciences, Computing Research Topics, PA 19085, CSC 3990.

[16] K. Krishnan. "*Computer Networks and Computer Security,"* Lecture Notes in Cryptography, 2004, Lecture 22-24.

[17] N. Tingyuan, T. Zhang, "*A Study of DES and Blowfish Encryption Algorithm*", IEEE, 2009.

[18] Saikumar Manku1 and K. Vasanth2, "Blowfish Encryption Algorithm for Information Security," *Journal of Engineering and Applied Sciences (ARPN),* ISSN 1819-6608, vol.10, no. 10, June. 2015.

[19] Wheeler, David, R. Needham. "*TEA a tiny encryption algorithm"* Fast Software Encryption. Springer Berlin/Heidelberg, 1995.

[20] Vikram R. Andem, "A Cryptanalysis of the Tiny Encryption Algorithm", M.Sc. thesis, University of Alabama, Alabama, 2003.

[21] J. Holden, "Demitasse: A Small Version of the Tiny Encryption Algorithm and its Use in a Classroom Setting", Department of Mathematics, Rose-Hulman Institute of Technology, Terre Haute, In 47803, USA.

[22] (1998) The Wikipedia website. [Online]. Available: https://upload.wikimedia.org/wikipedia/commons/a/a1/TEA_InfoBox_Diagram/.