A COMPREHENSIVE STUDY OF TIME COMPLEXITY OF VARIOUS ENCRYPTION ALGORITHMS

Vikendra Singh[1], Harsh Dhiman[2], Manisha Khatkar[3], Nida[4]
[1]Amity School of Engineering and Technology, Amity University, Noida , India
[2]School of Computing Science and Engineering, Galgotias University, Greater Noida, India
[3]Amity School of Engineering and Technology, Amity University, Noida, India
[4]School of Computing Science and Engineering, Galgotias University, Greater Noida, India

ABSTRACT

As the technology is going to increase, security issues are also going to increase. A secure way of communication and transmission is needed in terms encryption system. So, we need a best encryption algorithm which takes less time complexity. Time complexity refers to CPU time taken to encrypt plaintext to cipher text and back cipher text to original plain text. In this paper, a comparison is being done based on execution time and we try to find an efficient encryption algorithm which takes less time among some best encryption algorithm such as XOR, DES, TDES, and Blowfish.

KEYWORDS: *Encryption, time complexity, XOR, DES, TDES and Blowfish.*

I. INTRODUCTION

Cryptography is a two-step process of encryption and decryption. Encryption, basically, is a process of converting information from one understandable format to another (hidden) format which is totally different from original. This encryption process needs an efficient encryption algorithm. These algorithms need data to be encrypted and a key which is used to encrypt. Encrypted data is termed as cipher and decryption is reverse process of encryption in which encrypted data, cipher, convert back to original data using same key as in encryption process.

Encryption:

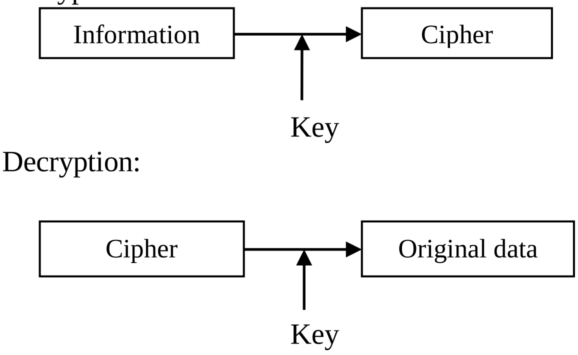


Fig 1: Encryption/Decryption

In the present era, everyone needs fast processing and less space required to store results in computation process as well as security of information. There are many encryption algorithms in which some of these take more computation time, some of these take less but all have their own advantages and disadvantages. Here, the aim to find an algorithm which take specific time for computation and more secure. These encryption algorithms can be of two types based on key-

1. Asymmetric key encryption

2. Symmetric key encryption

1.1 Asymmetric key Encryption

It is also called public key cryptography. In this, two key is used, one is for encryption called public key and decryption is performed by private key. It is not so easy to guess or interrupt both public key and private key as well as to gain access to the information. In the asymmetric key encryption, all the recipients have their public key and sender has its own private key which is not disclosed to anyone.

1.2 Symmetric key Encryption

Symmetric key encryption is also called private key encryption. Same key is used to encrypt and decrypt the data. Private Key makes the encryption process faster when it is used with public key. But secret key cryptosystem is suffered with the problem of exchanging the key. If N number of users want to use secret cryptosystem, then they must distribute N*(N-1)/2 keys. DES, TDES and BlowFish are the example of secret cryptosystem.

II. RELATED WORK

Before comparison of time complexity of DES, TDES, BLOWFISH and XOR encryption algorithms, some details of these algorithms are given below-

2.1 Data Encryption Standard (DES)

DES was developed by IBM in 1976 for the National Bureau of Standard (NBS), with approval from National Security Agency (NSA) but later adopted by US government as national standard. It uses 56-bit secret keys which are operated on 64 bit data block [1, 3] and every 8th bit is used as parity check. It divides the information into 64 bit data block and goes through 16 round Feistel Network with permutation process (initial and inverse permutation) [8]. Every 8th bit is used as parity check due to this 56 key is used. Again, 64 bit data block is divided into two halves each of 32 bits and uses a function, Feistel Network function which defines F: $\{0, 1\}^{2n} \rightarrow \{0,1\}^{2n}$ [8]. Feistel networks are designed for the construction of secret cryptosystem. It was first purposed by Horst Feistel during his work on the cipher Lucifer at IBM. It is parameterized by the number of rounds $d \in$ N and the round functions $f1 \ldots\ldots ; fd : \{0, 1\}^n \rightarrow \{0, 1\}^n$ [6] and it use 12 to 16 rounds. Inputs are split into two halves called Left half and Right half [2, 4]. Then a round function is executed on the right half and the obtained new right half is xored with left half. Then right half and left half are swapped. Finally, inverse permutation is computed [4].

2.2 XOR Encrypto-system

XOR encryption is simple and it is based on binary value of data and takes the ASCII values of data then converts into binary values [11, 12]. Same operation is performed with key. It uses XORed operation between the binary values of data and key.
Encryption:
P XORed K= C
Decryption:
C XORed K= P
Where
P- Plaintext or information
C- Cipher text
K- Key
Suppose, we want to encrypt a information using XOR algorithm by the KEY, then
Information: 1100 0010 0101 0110
Key: 1001 1011 0000 1010
Cipher: 0101 1001 0101 1100
If we again perform XOR operation between cipher and key, we will obtain information.
Cipher: 0101 1001 0101 1100
Key: 1001 1011 0000 1010
Information: 1100 0010 0101 0110

2.3 BlowFish Encryption System

Blowfish is an alternative of DES encryption algorithm. It is a symmetric encryption algorithm and designed by Bruce Schneier in 1993. Same key is used by the sender to encrypt the information and by the recipient to decrypt the cipher text or information. It uses variable length key from 32 to 448 and 16 round Feistel cipher with key independent S-boxes [9, 10]. It has two parts: key expansion and Encryption. In ken expansion part, 448 bit key is divided into several sub key and approximate total of these sub key is about 4168 bytes. It takes 64 bit block at a time and divide into two equal halves (L and R) each of 32 bit [6, 13], then iterates for 16 rounds

For i<=1 to 16

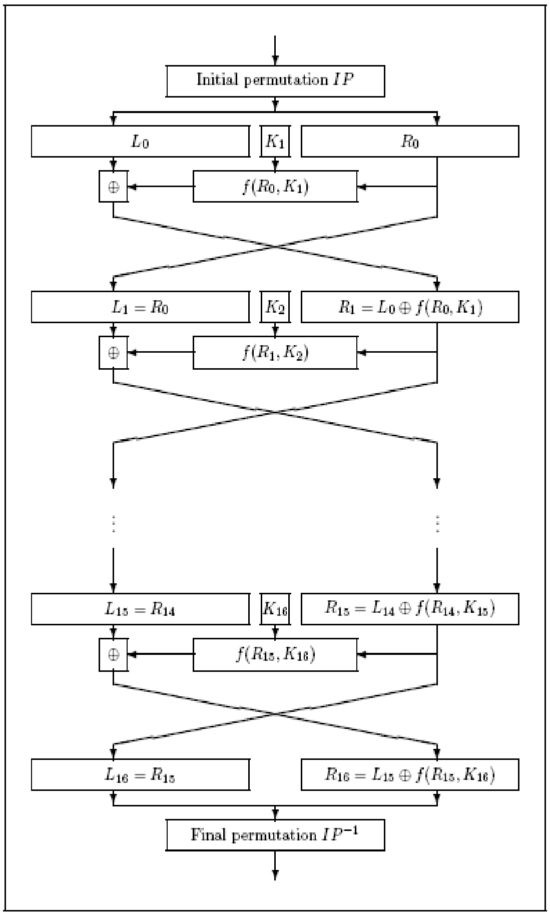
Li = Li-1

Ri = Li-1 XOR F (Ri-1, Ki)

Where Ki is the key in each round and F is a function which divides the 32 bit input into four quarters each of 8 bit. These quarters are used as input into S-boxes.

F (XL) = ((S1, a + S2, b mod232) XOR S3, c) + S4, d)

Where a, b, c and d are the four quarters.



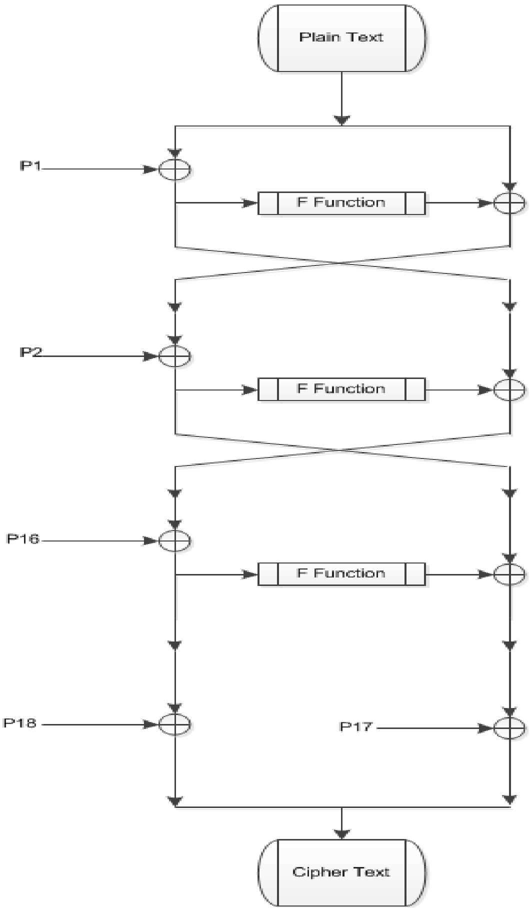


Fig. 2: DES encryption [2]

Fig. 3: Blowfish Encryption [6]

2.4 Triple Data Encryption Standard (TDES)

TDES is three times encryption and decryption of DES with three different keys [2]. It uses 64 bit cipher text block as input block and uses 168 bit key. As we know it is three times encryption of DES i.e. only 112 bit key out of 168 is effective [2, 4]. TDES uses three steps in encryption and three steps in decryption. In first step the information is encrypted using first key and in second step, the cipher obtained from first step is decrypted with second key and finally, in third step, the obtain output of second step is again encrypted with third key. Reverse procedure is followed in decryption process of TDES. Suppose, there are three keys K1, K2 and K3 then

TDES Encryption can be shown as:

$C= (\text{DES Encpt})_{K3}((\text{DES Decpt})_{K2}((\text{DES Encpt})_{K1}(I)))$.[4]

TDES Decryption can be shown as:

$I= ((\text{DES Decpt})_{K1} ((\text{DES Encpt})_{K2} ((\text{DES Decpt})_{K3}(C)))$

I… information

C ... cipher text

k_i ... key and i is iteration

Where DES Encpt and DES Decpt are DES encryption and DES decryption.

III. Result And Discussion

If a file of 10 kb is encrypted using these four algorithms three times separately then we can analyze from obtained results given below that the time (in millisecond) taken to encrypt a 10 kb file is given below.

According to the given data, we can say that DES takes less time to encrypt and decrypt a file while XOR algorithm takes more time than these remaining three algorithms because in XOR, bit by bit xoring is performed while in DES operation is performed on a block of data. TDES and Blowfish both take approximately equal time to encrypt and decrypt data because Blowfish uses Feistel Network with S-Boxes. If we encrypt the same file with same key then results may slightly vary due to different processor and depends on how much busy the processor is?

Table 1: Time taken in encryption/decrypation

Time (ms)		XOR	DES	TDES	BlowFish
1st time	Encryption	56	22	27	35
	Decryption	47	17	16	24
2nd time	Encryption	51	21	39	40
	Decryption	46	18	19	19
3rd time	Encryption	60	23	34	35
	Decryption	49	20	21	23

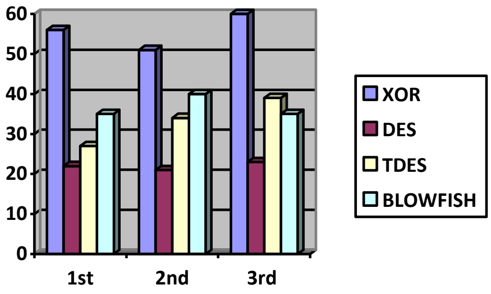


Fig. 4: time taken in encryption

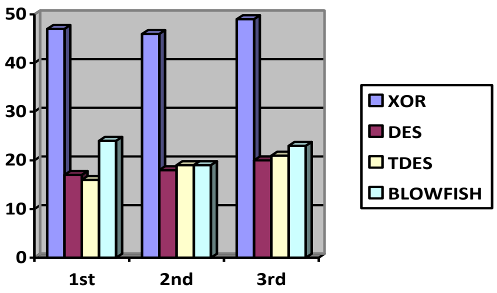


Fig. 5: time taken in decryption

IV. Conclusion

On the basis of implementation and obtained results, it can be easily concluded that DES is the fastest encryption algorithm among these four encryption algorithms while XOR is the slowest but it takes less space [13] and XOR operation is simple. TDES takes more time than DES because TDES is three times encryption of DES encryption process while TDES and Blowfish take approximately equal time.

V. Future Work

We can compare these algorithms with other encryption algorithms such RSA, AES etc. and also enhance the XOR encryption algorithm to make it more secure and also try to reduce the time taken to encrypt or decrypt the data.

REFERENCES

[1] Jawahar Thakur and Nagesh Kumar, "DES, AES and Blowfish: Symmetric Key Cryptography Algorithms Simulation Based Performance Analysis", International Journal of Emerging Technology and Advanced Engineering, ISSN 2250-2459, Volume 1, Issue 2, December 2011.

[2] Kruti R. Shah, Bhavika Gambhava, "New Approach of Data Encryption Standard Algorithm", International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-2, Issue-1, March 2012.

[3] Prasun Ghosal, Malabika Biswas and Manish Biswas, "A Compact FPGA Implementation of Triple DES Encryption System with IP Core Generation and On-Chip Verification", Proceeding of the 2010 International Conference on Industrial Engineering and Operation Management, Dhaka, Bangladesh, January 9-10-2010.

[4] Shashi Mehrotra Seth and Rajan Mishra, "Comparative Analysis of Encryption Algorithms for Data Communication", ISSN: 2229-4333, IJCST Vol. 2, Issue 2, June 2011.

[5] Russell K. Meyers and Ahmed H. Desoky," An Implementation of the Blowfish Cryptosystem", IEEE-978-1 -4244-3555-5/08, 2008.

[6] Rasheed Mokhtar Ahmed, Adel Zaghlul Mahmoud, "An Implementation of High Security and High Throughput Triple Blowfish Cryptography Algorithm", IJRRSAP Vol. 2, No. 1, ISSN: 2046-617X, March 2012.

[7] M. Anand Kumar and Dr. S. Karthikeyan, "Investigating the Efficiency of Blowfish and Rejindael (AES) Algorithms", I. J. Computer Network and Information Security, 2012, 2, 22-28 in MECS, March 2012.

[8] C. Adams, "The Shade Cipher: An Efficient Hash Function Based Feistel Network", IEEE, ISSN: 0-7803-3716-6 /97, JUNE, 1997.

[9] S. M. Dehnavi, M. R. Mirzaee Shamsabad, A. Mahmoodi Rishakani and Einollah Pasha, "Generalization of Statistical Criteria for Sboxes", IEEE, 978-1-4673-2386-4/12, May2012.

[10] Anthony Lineham and T. Aaron Gulliver, "Heuristic S-box Design", Contemporary Engineering Sciences, Vol. 1, no. 4, 147 – 168, 2008.

[11] Ralf Kusters and Tomasz Trundrerung, "Reducing Protocol Analysis with XOR to the XOR-free Case in the Horn Theory Based Approach", ACM 978-1-59593-810, October, 2008, Alexandria, Virginia, USA.

[12] Majdi Al-qdah and Lin Yi Hui, "Simple Encryption/Decryption Application", International Journal of Computer Science and Security (IJCSS-4), Volume (1), December 2011.

[13] Vikendra Singh and Sanjay Kumar Dubey, "Analysing the space complexity or various Encryption Algorithms", International Journal of Computer Engineering and Technology (IJCET), Volume 4, January-February 2013.

AUTHORS

Vikendra Singh is a student of M.Tech (Computer Science and Engineering) at Amity School of Engineering and Technology, Amity University, Noida (UP), India. He has passed his B.Tech from IAMR College of Engineering, Meerut (UP), India in 2012. His areas of interest are Computer Networks and Information Security.

Harsh Dhiman is a student of M.Tech (Computer Science and Engineering) at School of Computing Science and Engineering, Galgotias University, Greater Noida, India. He has passed his B.Tech from Bharat Institute of Technology, Meerut (UP), India in 2010. His areas of interest are Design and Analysis of Algorithm and Information Security.

Manisha Khatkar is a student of M.Tech (Computer Science and Engineering) at Amity School of Engineering and Technology, Amity University, Noida (UP), India. She has passed her B.tech from PDM College of Engineering, Sarai Aurangabad, Bahadurgarh (Haryana), India in 2012. Her areas of interest are Artificial Intelligence and Information Security.

Nida is a student of M.Tech (Computer Science and Engineering) at School of Computing Science and Engineering, Galgotias University, Greater Noida, India. She has passed her B.Tech from Integral University, Lucknow (UP), India in 2012. Her areas of interest are Computer Networks and Information Security.