

Article Citation Format

Akinrotimi, Akinyemi Omololu & Oladele, Rufus Olalere (2018).
Comparative Evaluation of Symmetric AES & DES Cryptographic
Techniques
Journal of Digital Innovations & Contemp Res. In Sc., Eng & Tech.
Vol. 6, No. 2. Pp 15-28

Article Progress Time Stamps

Article Type: Research Article
Manuscript Received: 15th April, 2018
Review Type: Blind
Final Acceptance: 24th June, 2018
DOI Prefix: 10.22624

Comparative Evaluation of Symmetric AES & DES Cryptographic Techniques

Akinrotimi, Akinyemi Omololu¹ & Oladele, Rufus Olalere²

^{1&2}Department of Computer Science
Faculty of Communication and Information Sciences
University of Ilorin
Ilorin Nigeria
E-mail: timiakin2011@yahoo.com

ABSTRACT

With the huge progress made in data exchange by electronic systems, the need of information security has become a necessity. Due to growth of multimedia application, security has become an important issue of communication and storage of files. Symmetric key cryptography is a common cryptographic technique, which involves, using the same key at both the transmitter and receiver side. The main advantage of symmetric key encryption is its less computational cost, compared to its counterpart-public key encryption. This research work, evaluated the performance of two symmetric cryptography techniques using the Advanced Encryption Standard (AES) and Data Encryption Standard (DES) for the security of plain text messages. The experimental approach was carried out with MATLAB Programming language. The two algorithms scaled on 64 bit key size for its encryption and their performance was measured based on the encryption and decryption time alongside with the encryption and decryption memory. The performance evaluation shows that when it comes to the computational time, the Data Encryption Standard performs faster than the Advanced Encryption Standard and also takes more of a Higher CPU Memory in processing than the AES symmetric key cryptography method.

Keywords: - Cryptography, Symmetric, Encryption, Decryption, Data Hiding

1. INTRODUCTION

Today's world is totally depend upon information, in that we can share information collect information and send and receive information from source to destination. Everyone wants to keep the inside information of work to be secret and safe. We use many insecure pathways in our daily life for transferring and sharing information using internet or telephonically, but at a certain level it's not safe [1] and as the sharing of images over network is increasing in large numbers. The NIST Computer Security Handbook [NIST95] defines the term computer security as, "The protection afforded to an automated information system in order to attain the applicable objectives of preserving the integrity, availability and confidentiality of information system resources (includes hardware, software, firmware, information/data, and telecommunications)."

Security is the mechanism by which information and services are protected from unintended or unauthorized access, change or destruction. Security in networking is based on Cryptography (a word with Greek origins, means “secret writing”), the science and art of transforming messages to make them secure and immune to attack.

Furthermore, an effective way to fight against cyber-attack is through encryption. It is the process of scrambling a message so that only the intended recipient can read it. It can provide a means of securing information. As more and more information is stored on computers or communicated via computers, the need to insure that this information is invulnerable to snooping and/or tampering becomes more relevant. With the fast progression of digital data exchange in electronic way, Information Security is becoming much more important in data storage and transmission. Information Confidentiality has a prominent significance in the study of ethics, law and most recently in Information Systems. With the evolution of human intelligence, the art of cryptography has become more complex in order to make information more secure. Arrays of Encryption systems are being deployed in the world of Information Systems by various organizations. Many encryption algorithms are widely available and used in information security. Encryption algorithms are classified into two groups: Symmetric-key (also called secret-key) and Asymmetric-key (also called public-key) encryption.

Symmetric key encryption is a form of cryptosystem in which encryption and decryption are performed using the same key. It is also known as conventional encryption while Asymmetric encryption is a form of cryptosystem in which encryption and decryption are performed using the different keys, one a public key and one a private key. It is also known as public-key encryption. A Key is a numeric or alpha numeric text or may be a special symbol. The Key is used at the time of encryption takes place on the Plain Text and at the time of decryption takes place on the Cipher Text. The selection of key in Cryptography is very important since the security of encryption algorithm depends directly on it. The strength of the encryption algorithm relies on the secrecy of the key, length of the key, the initialization vector, and how they all work together. Asymmetric encryption techniques are about 1000 times slower than Symmetric encryption which makes it impractical when trying to encrypt large amounts of data. Also to get the same security strength as symmetric, asymmetric must use a stronger key than symmetric encryption technique [6].

Also, cryptography which is intended to transform the data and can also be used to provide various security related concepts such as confidentiality, data integrity, authentication, authorization and non-repudiation. It solemnly depend on two basic components: an algorithm (cryptographic technique) and a key. The algorithm is a numerical procedure and the key is a factor used for data transformation. These algorithms provide cryptographic protection to the data by using encryption and the reverse by decryption which can be Symmetric key Algorithms or Asymmetric key algorithms.

2. LITERATURE REVIEW

2.1 Review of Related Work

In [4] the researchers worked on Image Encryption based on the RGB PIXEL Transposition and Shuffling research a technique of transposition and reshuffling of the RGB values of the image in steps was proposed. The technique proved to be really effective in terms of security analysis. The extra swapping of RGB values in the image file after RGB component shifting has increased the security of the image against all possible attacks that are currently available. Also in [5], researchers worked on a new image encryption technique based on combination of block displacement and block cipher technique. In this study, a work on new image encryption algorithm is proposed. It is already known that security of an algorithm depends on the length of the key.

This means that longer keys will always support good security features. The proposed algorithm uses 128-bit key which provided too much security for the proposed algorithm. To access original key or crypto analysis of the proposed key is required, time to break the key which is almost impossible for any hacker. There is no chance to generate floating point error because no such types of mathematical formulas have been applied on the proposed algorithm. The correlation co-efficient as well as their entropy values for the proposed algorithm was calculated.

Furthermore, in [3], researchers focused on a new technique of encryption without using predefined key. The input string was fragmented into several parts, with each part encrypted using a different algorithm. Three unique algorithms have been applied to encrypt the fragmented string on the basis of its orientation. For higher security levels, the key is derived from the two differently determined keys. The salient feature of this algorithm is that, a part of string being manipulated using base conversion, the second part of the string is deformed by interchanging position and increasing number of repetitions, and in the remaining elements, they perform simple operations. So, this algorithm was a complex combination without involving any complex calculation.

In [2], the researchers performed a review on image encryption technique based on hyper image encryption algorithm. The study presented a method for image security using block based image transformation and Hyper Image encryption techniques. The original image was divided into blocks, which were rearranged into a transformed image using a trans-formation algorithm, and then the transformed image was encrypted using the Blowfish algorithm i.e. Hyper Image encryption techniques. Finally, the result showed the correlation between image elements was significantly reduced. Their result also showed that increasing the number of blocks by using smaller block sizes resulted in a lower correlation and higher entropy. In this algorithm there is no key generator. A Hyper Image encryption algorithm was used, which divide the image into a number of blocks. Due to large data size and real time constrains, algorithms that are good for textual data may not be suit-able for multimedia data. In this algorithm the correlation between image elements was significantly decreased.

3. PROPOSED SYSTEM

The development of this experimental framework incorporates the MATLAB IDE platform with interaction with the graphical user interface tool for the production of a stand-alone application. The project will cohort a collaborative platform that will initiate the loading of plain-text as an input to the system, with a systematic comparative approach to choose the intended algorithm to work with whether Advanced Encryption Standard (AES) and Data Encryption Standard (DES). The developed system examined the best algorithm for encryption and decryption using the processing or execution time and CPU memory consumption.

3.1 Methodology

The Stepwise method for the implementation of the DES and AES algorithm for the encryption and decryption of data is explicitly stated thus;

3.1.1 DES Algorithm

1. DES accepts an input of 64-bit long plaintext and 56-bitkey (8 bits of parity) and produce output of 64-bit block.
2. The plaintext block has to shift the bits around.
3. The 8 parity bits are removed from the key by subjecting the key to its Key Permutation.
4. The plaintext and key will beprocessed by following.

- a. The key is split into two 28 halves
- b. Each half of the key is shifted (rotated) by one or two bits, depending on the round.
- c. The halves are recombined and subject to a compression permutation to reduce the key from 56bits to 48 bits. This compressed keys used to encrypt this round's plaintext block.
- d. The rotated key halves from step 2 are used in next round.
- e. The data block is split into two 32-bit halves.
- f. One half is subject to an expansion permutation to increase its size to 48 bits.
- g. Output of step 6 is exclusive-OR'ed with the 48-bit compressed key from step 3.
- h. Output of step 7 is fed into an S-box, which substitutes key bits and reduces the 48-bit block back down to 32-bits.
- i. Output of step 8 is subject to a P-box to permute the bits.
- j. The output from the P-box is exclusive-OR'ed with other half of the data block. k. The two data halves are swapped and become the next round's input.

3.1.2 AES Algorithm

A. Algorithm steps:

These steps used to encrypt 128-bit block

1. The set of round keys from the cipher key.
2. Initialize state array and add the initial round key to the starting state array.
3. Perform round = 1 to 9: Execute Usual Round.
4. Execute Final Round.
5. Corresponding cipher text chunk output of Final Round Step

B. Usual Round:

Execute the following operations which are described above.

1. Sub Bytes
2. Shift Rows
3. Mix Columns
4. Add Round Key, using K(round)

C. Final Round:

Execute the following operations which are described above.

1. Sub Bytes
2. Shift Rows
3. Add Round Key, using K(10)

D. Encryption:

Each round consists of the following four steps:

- i. **Sub Bytes:** The first transformation, Sub Bytes, is used at the encryption site. To substitute a byte, we interpret the byte as two hexadecimal digits.
- ii. **Shift Rows:** In the encryption, the transformation is called Shift Rows.
- iii. **Mix Columns:** The Mix Columns transformation operates at the column level; it transforms each column of the state to a new column.
- iv. **Add Round Key:** Add Round Key proceeds one column at a time. Add Round Key adds a round key word with each state column matrix; the operation in Add Round Key is matrix addition. The last step consists of XO Ring the output of the previous three steps with four words from the key schedule. And the last round for encryption does not involve the "Mix columns" step.

E. Decryption:

Decryption involves reversing all the steps taken in encryption using inverse functions like

- a. Inverse shift rows
- b. Inverse substitute bytes
- c. Add round key
- d. Inverse mix columns.

The third step consists of XO Ring which is the output of the previous two steps with four words from the key schedule.

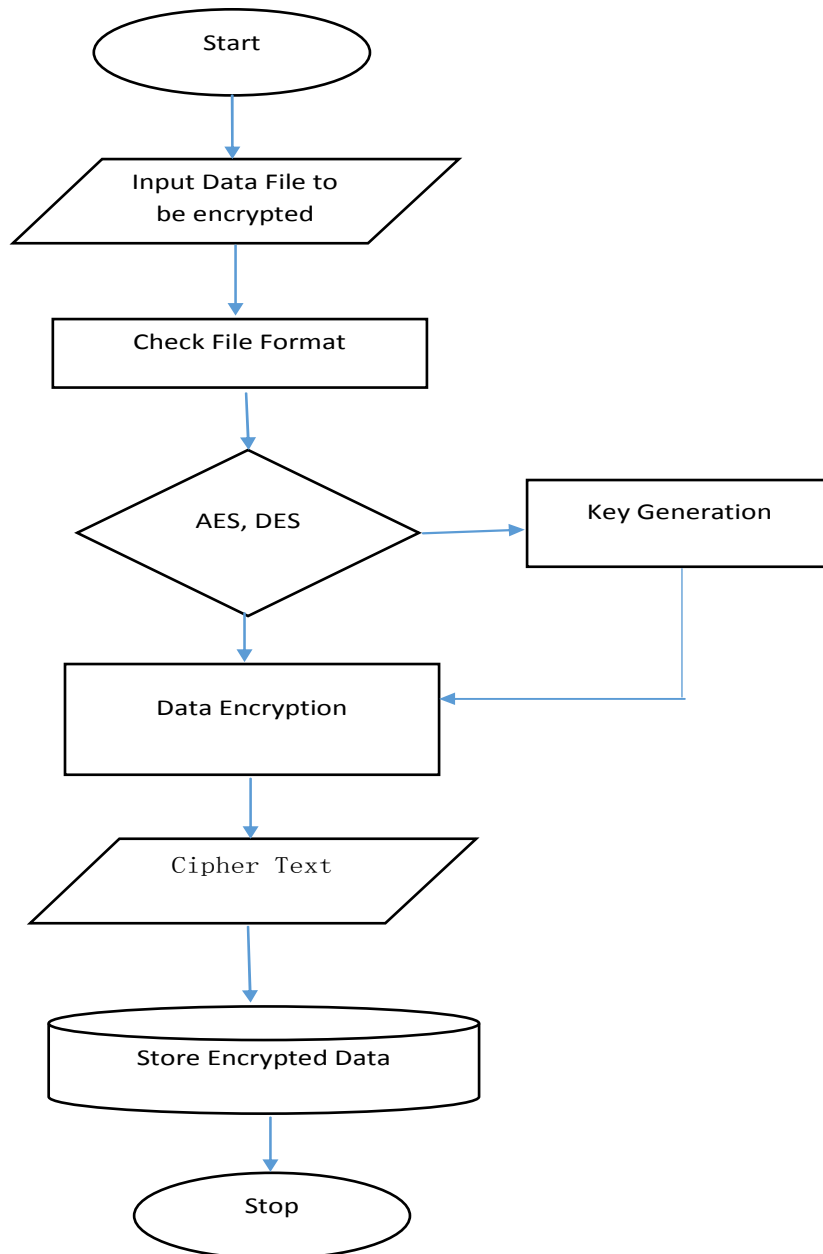


Figure 3.1: Data Encryption process

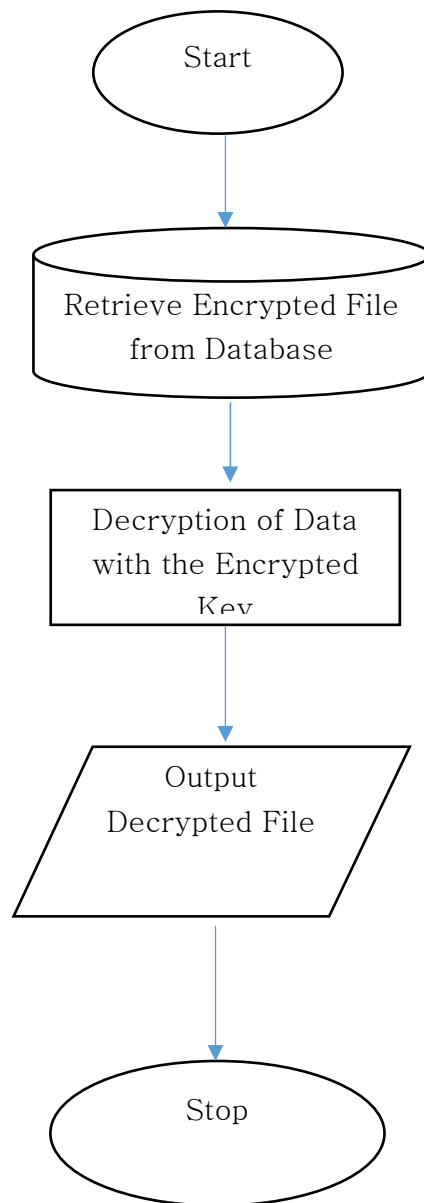


Figure 3.2: Data Decryption process

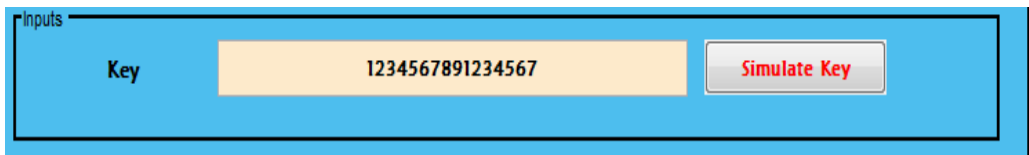
4. IMPLEMENTATION AND RESULT

4.1 AES

The results of the advanced encryption standard algorithm in respect to the 16 bits text is shown below, firstly the plain message is loaded and then a key is simulated to encrypt the message, of which the timing, ratio and memory usage is measured. The figures below show the simulated key for the advanced encryption algorithm

A. Key generation

The process starts by generation of a symmetric key for encrypting the message.

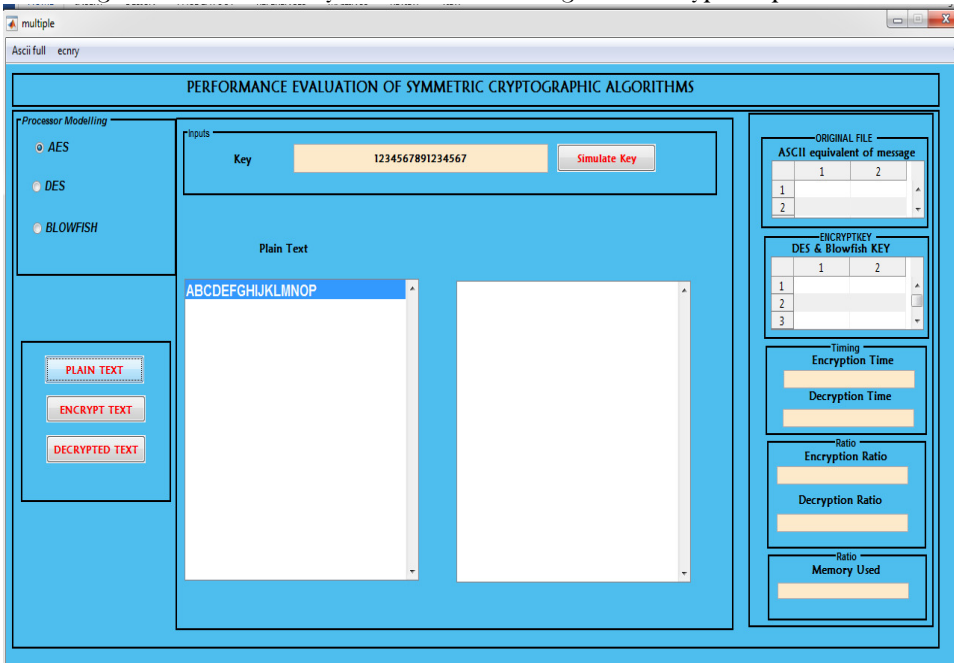


The interface shows a blue box with the title "Inputs". Inside, there is a label "Key" next to a text input field containing the value "1234567891234567". To the right of the input field is a button labeled "Simulate Key".

Figure 4.1a: Key generation for AES Algorithm

B. Plain Message

The plain message is loaded to the system so as to undergo the encryption process.



The interface is a window titled "multiple" with a menu bar "Ascii full entry". The main title is "PERFORMANCE EVALUATION OF SYMMETRIC CRYPTOGRAPHIC ALGORITHMS". On the left, under "Processor Modelling", there are radio buttons for "AES" (selected), "DES", and "BLOWFISH". Below these are three buttons: "PLAIN TEXT", "ENCRYPT TEXT", and "DECRYPTED TEXT". The central area has a "Key" input field with "1234567891234567" and a "Simulate Key" button. Below the key field is a "Plain Text" label and a text area containing "ABCDEFGHIJKLMN". On the right, there are three sections: "ORIGINAL FILE" with a table for "ASCII equivalent of message", "ENCRYPT KEY" with a table for "DES & Blowfish KEY", and "Timing" with fields for "Encryption Time", "Decryption Time", "Encryption Ratio", "Decryption Ratio", and "Memory Used".

Figure 4.1b: Plain text for AES Algorithm

C. Cipher Text

The figure below shows the successful encryption of the plain message by the AES algorithm.

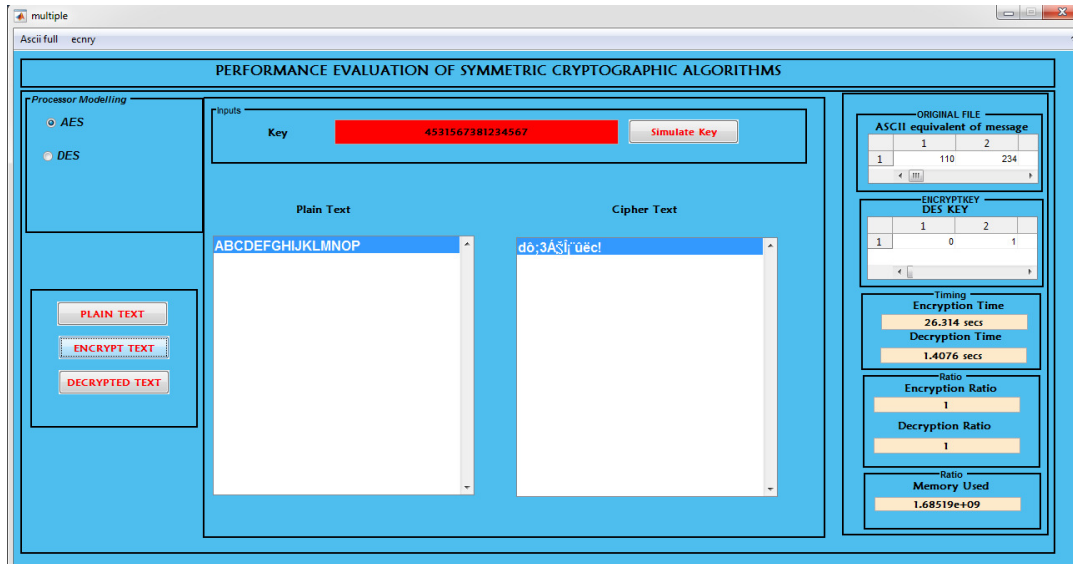


Figure 4.1c: Encrypted Message for AES Algorithm

D. Decryption Mode

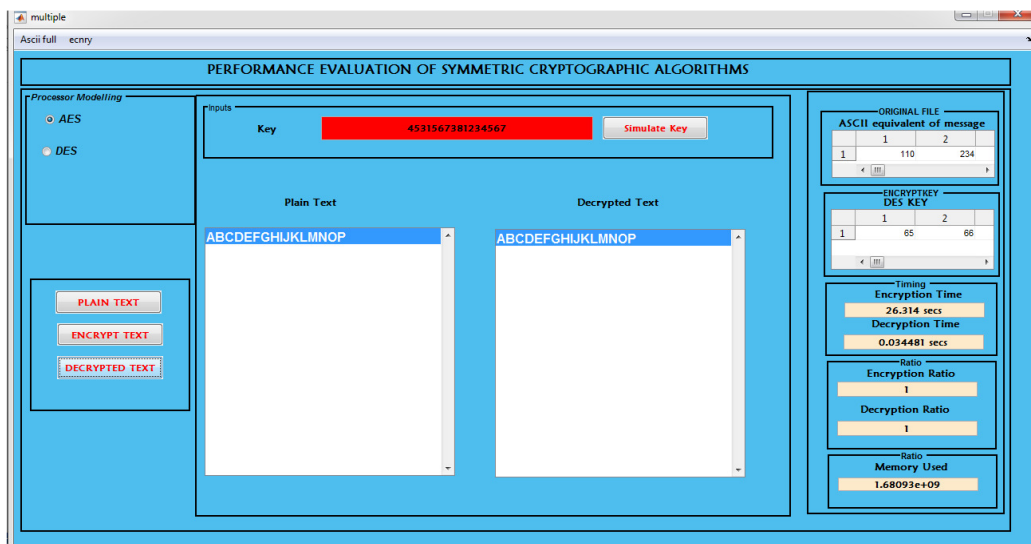


Figure 4.1d: Decryption mode for AES Algorithm

4.2 DES

The results of the data encryption standard algorithm in respect to the 64 bits text is shown below, firstly the plain message is loaded and then a key is simulated to encrypt the message, of which the timing, ratio and memory usage is measured. The diagrammatic presentations below shows the stepwise process.

A. Key generation

The process starts by generation of a symmetric 64 bits key for encrypting the message.

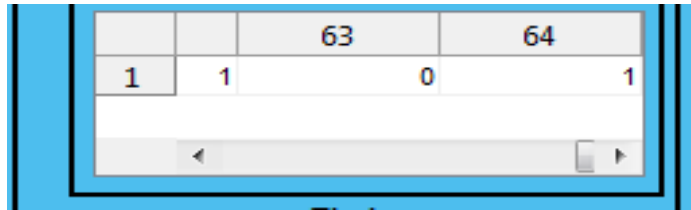


Figure 4.2a: 64-bit key generation of the DES

B. Plain Message

The plain message is loaded at the right list box of the gui with a message length of 64, this serve as input to the DES algorithm, the algorithm also helps to generate a 64-bit key for the encryption of the message.

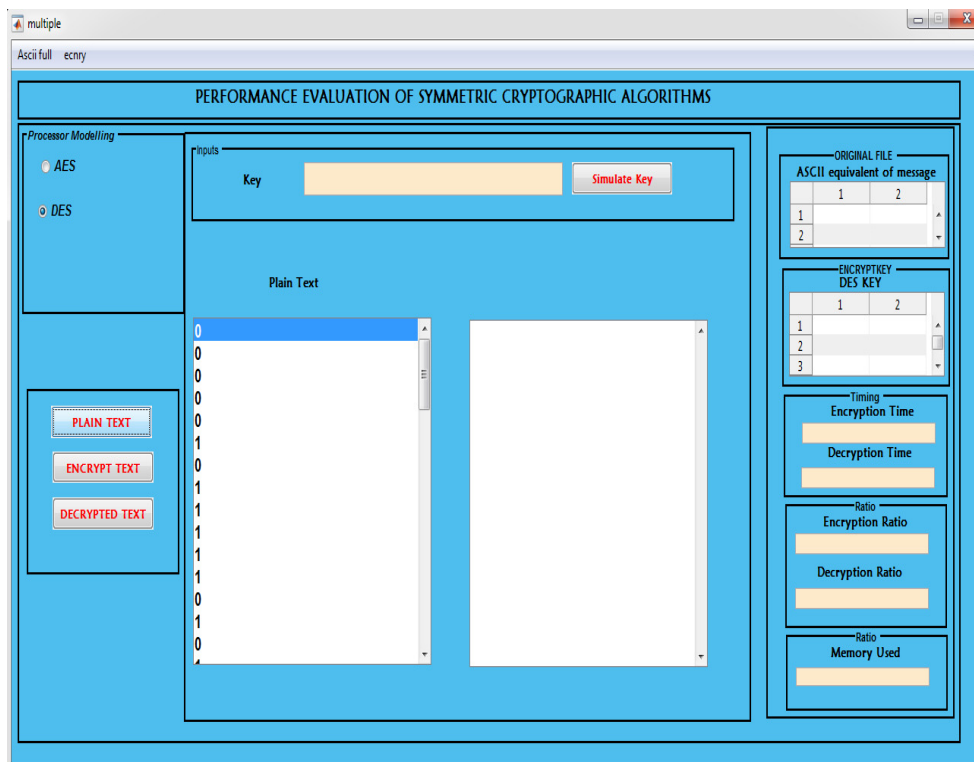


Figure 4.2b: Plain message for the DES Encryption

C. Cipher Text

The figure below shows the successful encryption of the plain message by the DES algorithm. The input plain message to the left, while the encrypted message to the right

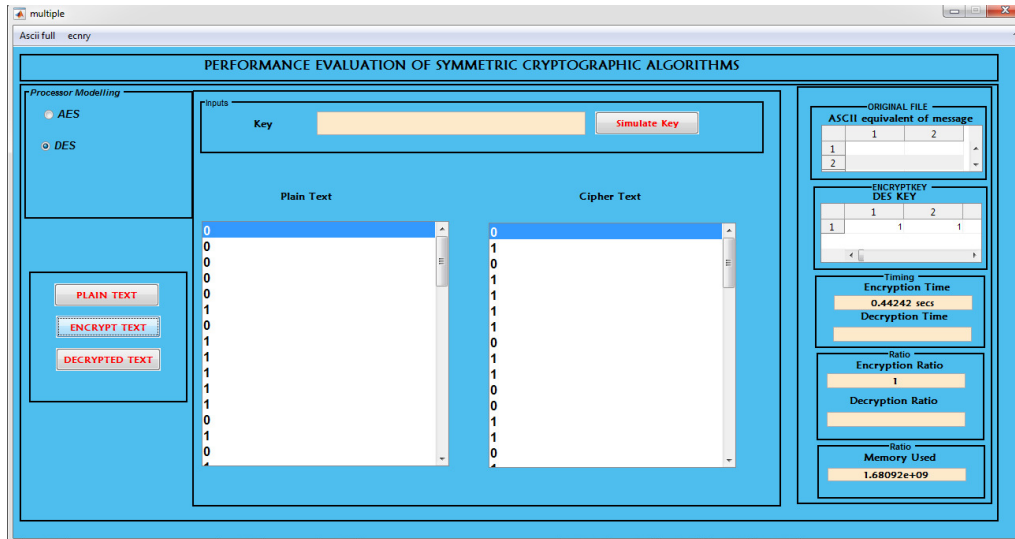


Figure 4.2c: Encrypted Message for the DES Encryption

D. Decryption Mode

The decryption mode of des algorithm is shown below.

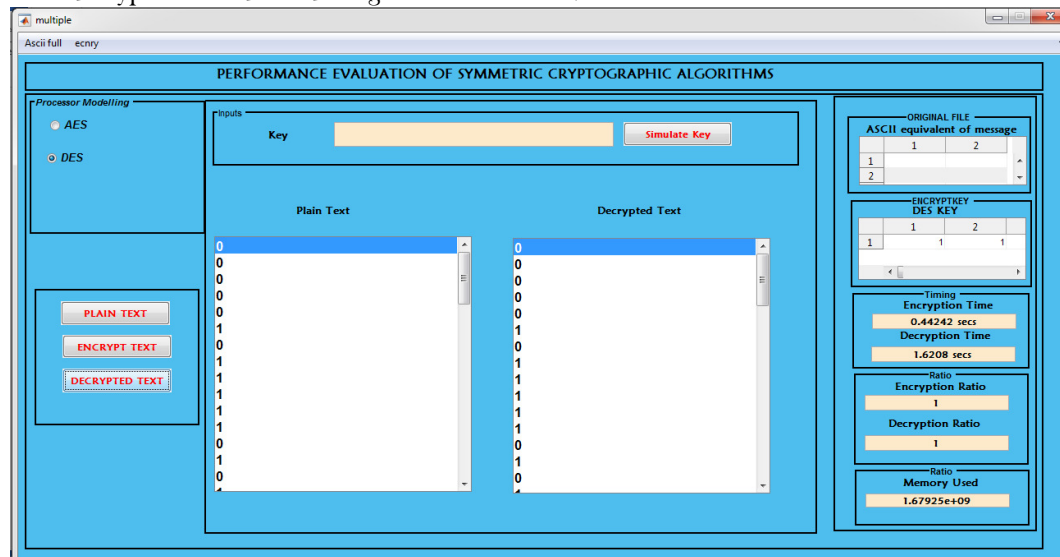


Figure 4.2d: Decryption mode for the DES Encryption

4.3 SYSTEM EVALUATION

The encryption and decryption time, encryption and decryption ratio and the memory space consumed is used to project a comparative approach for the two algorithms.

4.3.1 Tabular analysis

The Table below gives the variation of the three algorithms inn respect to the time, memory consumed and ratio. The systems differ with the time and the memory used but the encryption to decryption ratio was constant, that is to say the decryption process does not affect the dimension of the message.

Table 4.1 Comparative Approach Table

Algorithms	Encryption Time	Decryption Time	Encryption and Decryption Ratio	Encryption memory	Decryption memory
AES	5.3623 secs	0.013116 secs	1	1.68E+09hz	1.68E+09hz
DES	0.20827secs	1.1652 secs	1	1.68E+09hz	1.69Ehz

4.3.2 Graphical analysis

The chart below gives a more graphical analysis of the symmetric algorithms in respect to the encryption and decryption time. The Advanced Standard Algorithm performs better than the Data Encryption Standard in relation to both the timing and the CPU memory used. The AES encrypts and decrypts faster than the DES algorithm, the AES also processed with less memory consumption.

4.3.3 Encryption and Decryption Time

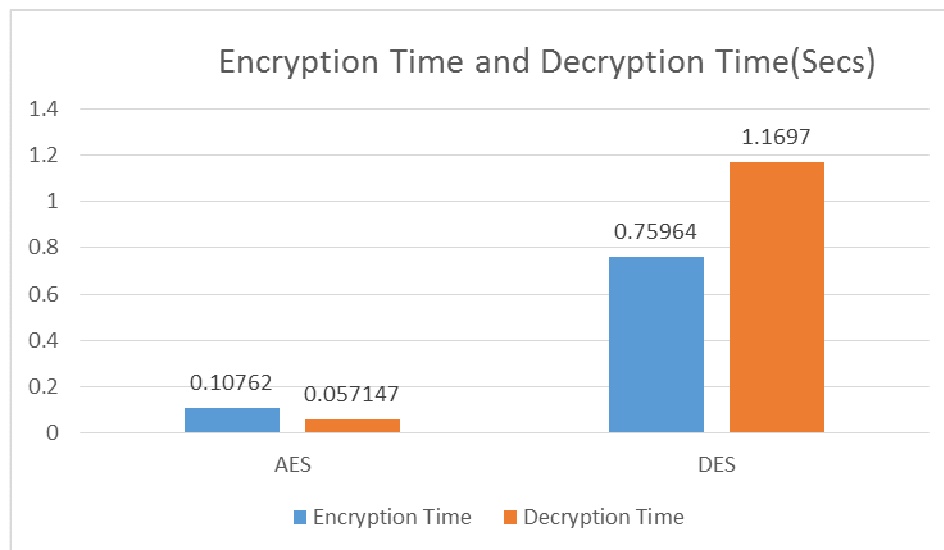


Figure 4.9: Encryption and Decryption Time

4.2.3.2 CPU MEMORY

The chart below shows the graphical result of the CPU memory up by each of the algorithm, the AES consumed a lower CPU memory than the DES algorithm. The diagram below helps to illustrate this better.

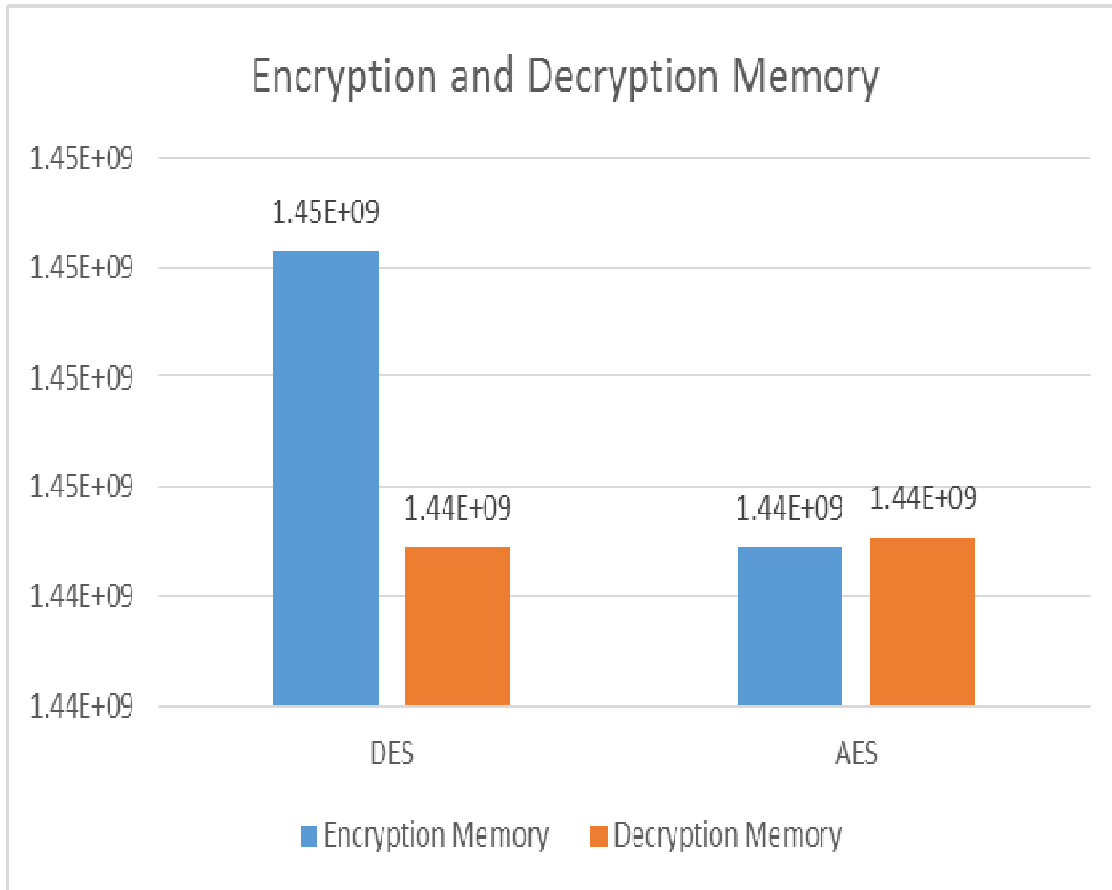


Figure 4.10: Encryption and Decryption Memory

5. CONCLUSION

Evidently, this study concludes the security of text files using symmetric cipher key DES algorithm and AES algorithm, in the project AES guarantee the unbreakable security for text security with higher speed than the DES algorithm and also took a lower CPU memory for its processing. The AES is quite significant for text file encryption than the DES. The implementation approach shows the encryption and decryption time including the CPU memory used.

6. RECOMMENDATION

It is recommended that other symmetric techniques be compared with the DES algorithm and AES algorithm for a comparative research on encryption and decryption of text files. Also other image metrics can be used to evaluate the performance rate and usability for text files.

REFERENCES

1. Arjun, W. (2016) "Network Security Essentials: Applications and Standards," Prentice Hall, 2016.
2. Junwale,P. R. Annapurna,M. and Sobha,G. (2013) "A Review on Image Encryption Technique based on Hyper Image Encryption Algorithm," International Journal of Advanced Research in Computer Science and Software Engineering, vol. 3, no. 11, pp. 614-618, November - 2013.
3. Kandle,N. S. Tiwari, (2013) "A New Combined Symmetric Key Cryptography CRDDBT Using Relative Displacement (RDC) and Dynamic Base Transformation (DBTC)," International Journal of Engineering Research & Technology, vol. 2, Issue. 10, www.ijert.org , October - 2013.
4. Keste,A. (2013) "Image Encryption based on the RGB PIXEL Transposition and Shuffling," I. J. Computer Network and Information Security, 7, in MECS (<http://www.mecs-press.org/>), DOI: 10.5815/ijcnis.2013.07.05, pp.43-50, Published Online June 2013
5. Kushwah,K. and S. Shibu, (2013) "New Image Encryption Technique Based On Combination of Block Displacement and Block Cipher Technique," International Journal of Computer Science and Information Technologies, vol. 4, no. 1, pp. 61 - 65, 2013
6. Singh & Supriya, (2013) A Comparative Analysis, "International Journal of Computer Applications (0975 - 8887) vol. 61, no.20.