Asian Journal of Research in Computer Science

3(3): 1-17, 2019; Article no.AJRCOS.49658
ISSN: 2581-8260

Design and Implementation of Students' File Encryption and Decryption System Using One Time Pad Algorithm for Computer Science Department, Adsu Mubi

O. Sarjiyus[1*] and J. M. David[1]

[1]Department of Computer Science, Adamawa State University, Mubi, Nigeria.

Authors' contributions

This work was carried out in collaboration between both authors. Author OS designed the study, performed the statistical analysis, wrote the protocol and wrote the first draft of the manuscript. Authors JMD and OS managed the analyses of the study. Author OS managed the literature searches. Both authors read and approved the final manuscript.

Article Information

DOI: 10.9734/AJRCOS/2019/v3i330093
Editor(s):
(1) Dr. Hasibun Naher, Associate Professor, Department of Mathematics and Natural Sciences, BRAC University, Dhaka, Bangladesh.
Reviewers:
(1) Anthony Spiteri Staines, University of Malta, Malta.
(2) Snehadri Ota, India.
(3) Pasupuleti Venkata Siva Kumar, Vallurupalli Nageswara Rao Vignana Jyothi Institute of Engineering &Technology, India.
Complete Peer review History: http://www.sdiarticle3.com/review-history/49658

Original Research Article

Received 09 April 2019
Accepted 16 June 2019
Published 27 June 2019

ABSTRACT

The research, design and implementation of students file encryption and decryption system using one-time pad algorithm seeks to improve the security of student records in tertiary institutions by using trusted techniques as given in the research. The purpose is to develop an automated student's data protection application in order to consolidate existing security measures as it pertains to data integrity and privacy. The software design methodology adopted for the research is the waterfall model due to the fact that is does not allow overlapping of processes. For the system design and implementation. Pelles C programming for windows version 9.00.9 for code generation and startUML was used for user interface design all in a bid to actualize the objectives of the research.

**Corresponding author: E-mail: Sarjiyus@gmail.com;*

Keywords: Decryption; encryption; one-time pad; record; security.

1. INTRODUCTION

Security of data in a computer is very essential in protecting data from unauthorized parties that do not have the authority to access the content of the data if the data has a very high value that has been stored in the computer and then opened by another party, then it would be very detrimental to the real owner since it has been compromised. [1].

One way to protect data is to password-protect the data. But now to unlock the data, password generating software are used to access the contents of data easily. Another way that is used for encoding is to use the science of cryptography is to encrypt the data so that the data cannot be read, deleted, changed or tempered by others in any way [2].

Science of cryptography has been in use for a long time in accordance with technological developments and innovations. Many different kinds of complex algorithms have been created as tools to encrypt data one of which is the algorithm One Time Pad (OTP) Algorithm. The advantage of the one-time pad algorithm is to perform the encryption process and a decryption of each character plaintext and use each character in the key. One Time Pad, this algorithm uses the same key for encrypting and a decrypting data [1].

One-time pad algorithm is only used one time for one key encryption key then it will be destroyed and not used again for any encryption process in that session.

Long Before now, cryptography was concerned solely with data confidentiality (i.e., encryption) — conversion of data from a comprehensible form into an incomprehensible one, and back again at the other end, rendering it unreadable by interceptors or eavesdroppers without secret knowledge (namely, the key needed for decryption of that message). In recent times, the field has expanded beyond confidentiality concerns to include techniques for message integrity checking, sender/receiver identity authentication, digital signatures, interactive proofs, and secure computation, amongst others.

Encryption attempts to ensure secrecy in data communications and transmission, such as those of spies, military leaders, and diplomats, but it has also had religious applications. Steganography (i.e., hiding even the existence of a message so as to keep it confidential) was also first developed in ancient times. An early example, from Herodotus, concealed a message – a tattoo on a slave's shaved head - under the regrown hair. More modern examples of steganography include the use of invisible ink, microdots, and digital watermarks to conceal information.

The main purpose of this research is to develop an automated student data encryption and decryption application using one-time pad technique system. The research focuses on designing a system of OTP that will protect student data against unauthorized, malicious tendencies, thereby maintaining the overall privacy/ confidentiality of the data.

2. CONCEPTUAL FRAMEWORK

Munir [3] Pointed out that Cryptography is derived from the Greek, crypto and Graphia. Crypto means confidential while Graphia means writing. In the term of the terminology, cryptography is the science and art that is used when a message is sent from one place to another to maintain the security of the message. Engineering data encryption (cryptography) is applied to the data and information, performed by encoding or hiding the original data. In cryptography, a message which will be kept secret called the plaintext and encrypted messages that have been called cipher text. In 1960s, the development of computers and communication systems has an impact on the demand of the parties - certain parties to provide various security services and protect information in digital form.

One-time pad (OTP) is a stream cipher encryption and decryption of one character each time. This algorithm was found in 1917 by Major Joseph Mauborgne as the improvement of the Vernam cipher to produce the perfect security. Mauborgne proposes the use of one-time pad (pad = paper notebooks) which contains the generation of random sequences of characters - a key character. To encrypt a message pad, it is simply used once (one-time), afterwards to encrypt messages, the pad that has been used can be destroyed in order that no one can reuse it. [3].

Srikantaswamy and Phaneendra [4] in their article shown that the random key stream can be employed to generate a lifetime supply of keys for OTPs. Random key generation can easily be created by permutation methods. These methods can be adopted in combination with other procedure such as substitution and encryption function for successful results. The objective of this study is to demonstrate how OTP encryption technique can be accomplished by a combining of these techniques. In two new methods of OTP encryption enhancement based on I O'S complement and XOR operations have been presented that do not depend on the original cipher about OTP cipher.

The investigation results of the study have been evaluated with benchmark images and are compared with different image encryption algorithms reported in the literature. Key sensitivity analysis, key space analysis, and numerical analysis proved that this algorithm proposes better security at minor calculation overhead. ln [5].

Borowski and Lesniewicz [5] Presented a hardware generation of binary random sequences with the latent output rate of 100 Mbit/s to eliminate the limitation associated with accessibility of lengthy one-time keys.

The one-time pad system was modified by using the concepts of 10's complement operation. The eavesdropper come across confusion by observing decimal and binary combination with added concept of complements [6].

Another attempt at improving the algorithm was made by using a conventional block cipher and one-way hash algorithm to design the onetime pad algorithm. This algorithm proposed by them totally balance the insufficiencies of the conventional block cipher, and exploit the benefits of the one-way hash algorithm.

Penchalaiah and Reddy [7] Modified the One-Time pad algorithm to work without any secret key overhead while on the transmission (since the key is along as message) by using two algorithms, a Key Exchanging Algorithm, and a Random Bit Generation algorithm.

Penchalaiah and Reddy [7] says that at the advent of binary systems for computational analysis (computers), memory and processing power was expensive and hard to obtain. This led to brilliant mathematical implementations of encryption that protected data, including communication. Due to the impracticality of OTPs, modern encryption was born which is based upon limited, finite size keys and produces creative attacks other than a 'brute force' attack. Capable mathematicians and technologists are highly motivated in their attempts to break encryption; they are succeeding. They have devised many attacks such as man-in-the-middle, statistical, side channel attacks, and many more.

Zaeniah and Purnama [8] in their article an Analysis of Encryption and Decryption Application by using One Time Pad Algorithm said that experiment has been conducted on various types of file formats such as .doc, .pdf, .ppt, and image file formats and various file sizes in order to determine the speed of encryption.

The problem of key distribution and protection was solved using elliptic curve cryptography. An overview of Koblitz method of encoding was provided and a hybrid security mechanism based on OTP was developed [9].

The One-Time Pad was modified with 2's complement approach to introduce more complexity and make the task of cryptanalyzing any ciphertext recovered to be more difficult [10].

Devipriya and Lesniewicz [10] portrayed the One Time Pad encryption as a method binary additive stream cipher, where a stream of truly random keys is generated and then combined with the plaintext for encryption or with the cipher text for decryption by an exclusive OR"(XOR) addition.

Another attempt at handling the randomness of the key was to use a simple quantum circuit to generate a truly random OTP using various quantum superposition states [11].

A one-time pad cryptographic method was designed using a star network of N Lorenz subsystems, referred to as augmented Lorenz equations, which generates chaotic time series as pseudorandom numbers to be used for masking a plaintext [12].

Lange and Takagi [13] express that there are publications and extant literature on OTP cipher and its enhancements. Much quantum key distribution (QKD) system has been expanded to quantum network manager using OTP encryption. This outline does not just handle the switch and QKD protocol startup processes but

as well handles multiplexing and synchronization of secret key streams. An encryption algorithm based on OTP technique to provide sufficient privacy of images using chaos theory.

Abiodun et al. [14] Describe the practical difficulty of using an OTP is that the pad/key bytes cannot be reused. This means that even for a two-way communication, each entity must have a sufficient supply of key material on hand so that they do not run out of keys before new ones can be generated. People are not interested in modifying the algorithm, they are more interested in improving the way the key is generated either by trying to introduce a true random instance or modifying the algorithm that generates the keys to create a lifetime supply of key. Their implementation only tries to solve the problem of getting true randomness but does not solve the distribution/management of key material as different keys still have to be sent for different messages. The proposed algorithm solves the key problem by making it possible to use the same key to encrypt different messages and not reveal any pattern that could be exploited by the attacker.

3. MATERIALS AND METHODS OF DATA COLLECTION

The software design methodology used for the research is the Non-iterative waterfall model this is because it is very simple to understand and portrays the sequential life cycle of a system. The waterfall model gives out explicit flow of process in a linear sequential format. This means that each phase in the development process proceeds only when the preceding phase is complete. The waterfall model does not overlap.

Two types of data were collected and used during the course of this research: primary and secondary data. The primary data used for this research were students e- record formats and layouts obtained from the Exams office of Computer Science Department, Adamawa State University, and test data consisting of students' records of 2016/2017 and 2017/2018 sessions respectively. While, the secondary data was obtained from online journal articles, textbooks. and lecture notes.

In the design process Pelles C programming language windows version 9.00.9 for code generation and StarUML for the user interface design were used.

3.1 System Design

In this proposed scheme, two keys were introduced. Normally, when the key is uniform and random, then there is perfect security but it is almost impossible to get a truly random key. Here, two keys are chosen so that even if one of the random numbers has a trapdoor and can be predicted, the other will be impossible to be predicted or brute forced and combining both keys will surely give a form of randomness. Randomly chosen bits/bytes were added to the Plaintext before encryption so as to reduce the redundancy in English and ASCII encoding and as a form of steganography which makes it impossible to retrieve the key or Plaintext at the beginning of the ciphertext. The Cipher block chaining mode was used to completely diffuse the plaintext so that each time there is a new Plaintext to be encrypted, it always gives a different CipherText even though the same key/pad was used and also a varying initialization vector was employed and used in constructing the algorithm.

The One Time Pad encryption method used is a binary additive stream cipher, where a stream of truly random keys is generated and then combined with the plaintext for encryption or with the cipher text for decryption by an exclusive OR`(XOR) addition. The stream cipher encryption scheme would be unbreakable if the key is as long (in length) as the plantext, if the key is truly random and the key must be used only once.

3.2 OTP Algorithm

A. The Algorithm for Encryption:

1) Load file Step
2) Read the content of the file and store in an array-list Plain Step
3) Calculate the number of blocks in the file and the offset
4) Generate 512-bit random key Steps
5) Generate a random integer for each file block to be stored in an array step.
6) Move the key four times by four of the above generated random numbers and store the shifted keys as s1, s2, s3, s4
7) Generate a one-time pad (OTP) with the key and the random integers

   a. Get the total size of the plain text
   b. Move the key with a random integer
   c. Make an inverse of the shifted key.

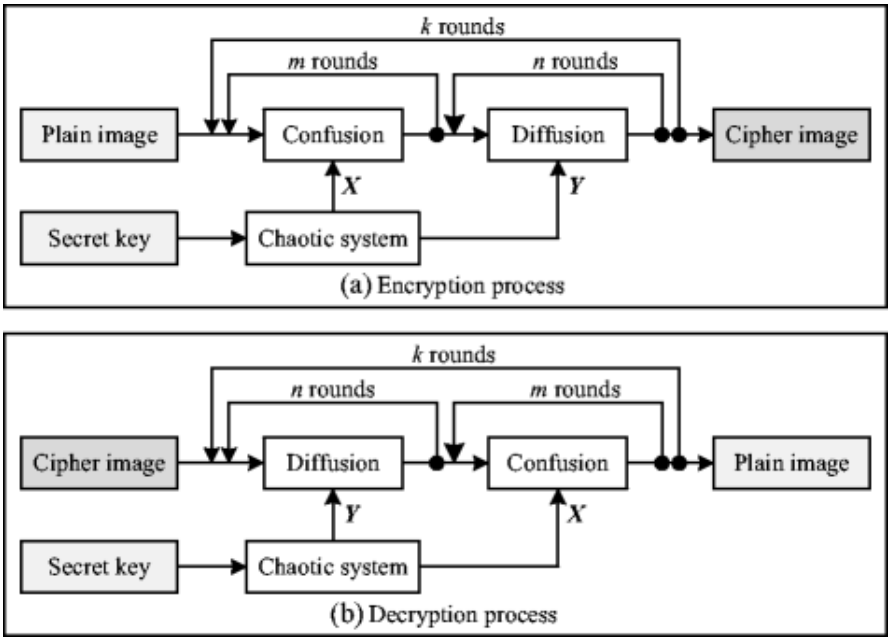

Fig. 1.3. Architectural design of the system of encryption and decryption using OTP

d. Repeat steps b to c for each block, append it to the one-time pad array.
e. For the offset append one (1) to the pad and append zeros till the pad is complete

8) Return the one-time pad Break the plain text into blocks of 512 bit each perform a Cipher Block Chaining (CBC) with each of the previously moved keys.
   a. Exclusive Or the block and key, block XOR key = text1
   b. Exclusive Or text1 and s1, text1 XOR s1 = text2
   c. Exclusive Or text2 and s2, text2 XOR s2 = text3
   d. Exclusive Or text3 and s3, text3 XOR s3 = text4
   e. Exclusive Or text4 and s4, text4 XOR s4 = text5
   f. Append text5 to the array text
   g. Repeat steps a to f for each block
9) Exclusive OR each bit in OTP and the bit in text, OTP XOR text = cipher.
10) Convert the binary values in cipher to character and store in file.
11) Store the random key into the key file and append it with all random numbers used for shifting.
12) End

B. The Algorithm for Decryption:

Step 1: Load file

Step 2: Read the content of the file and store in an arraylist .

Step 3: Calculate the number of blocks in the file and the offset.

Step 4: Read the content of the key file and store the key into an array.

Step 5: Store the shift integers into another array.

Step 6: Move the key four times by 4 of the shift integers and store as s1, s2, s3, s4.

Step 7: Generate a One - time pad with the key and the random integers.

   a. Calculate the total size of the plain text
   b. Move the key with the shift integers Obtained from the key file.
   c. Inverse the shifted key.
   d. Repeat steps b to c for each block and join it to the one-time pad array.
   e. For the offset join one (1) to the pad and join zeros till the pad is complete.
   f. Return one-time pad = OTP Step

Step 8: Convert the cipher text to binary values = text.

Step 9: Exclusive OR each bit in OTP and the bit in text, OTP XOR text = text 1.

Step 10: Break the cipher text into blocks of 512 bit each perform a Cipher Block Chaining (CBC) with each of the previously shifted keys.

a. Exclusive OR the tex1 and s4, text1 XOR s4 = text2
b. Exclusive OR text2 and s3, text2 XOR s3 = text3
c. Exclusive OR text3 and s2, text3 XOR s2 = text4
d. Exclusive OR text4 and s1, text4 XOR s1 = text5
e. Exclusive OR text5 and key, text5 XOR key = text6
f. Join text 6 to the array plain
g. Repeat steps a to f for each block

Step 11: Convert the binary values to character and store into the plain text file.

Step 12: End

3.3 Analysis of the Existing System

The password, PIN and any protection used in almost all conventional system does not guaranty the needed security enough to protect the privacy and integrity of Student file in Computer Science Department Adamawa state university Mubi. Hence a student, using social engineering technique can obtain the registration number of his colleague (often used as the password) to gain unauthorized access thereby breaking into the privacy of the victim.

Also the current system in use in the department cannot be used to encrypt multimedia files and this is a major snag and set back in trying to secure student records.

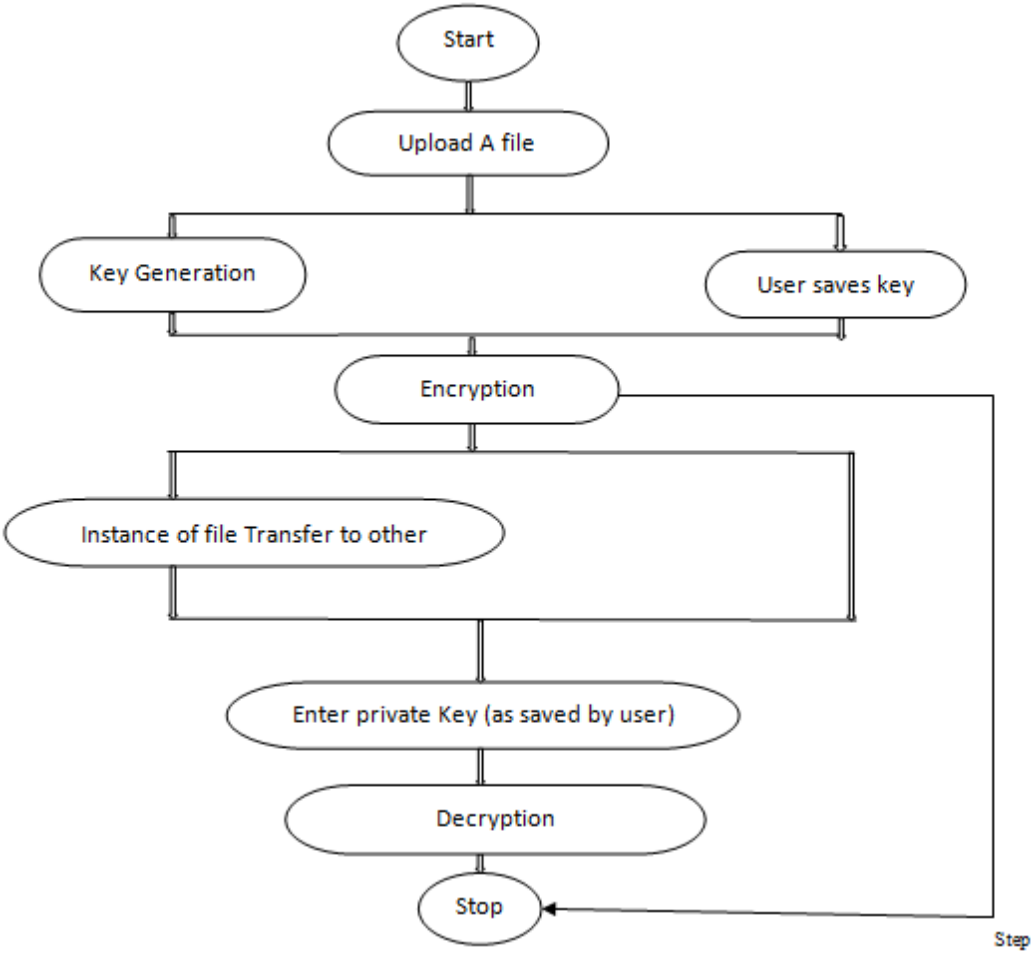

Fig. 1.4. Activity diagram for the encryption and decryption using OTP system

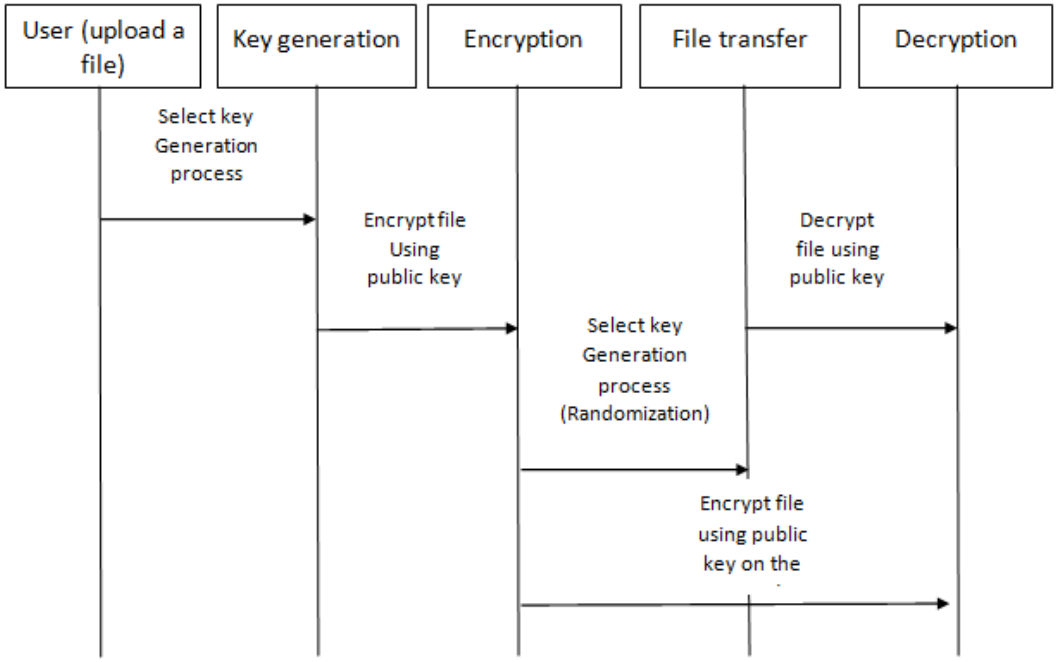

Fig. 1.5. Sequence diagram for the encryption and decryption technique using OTP system

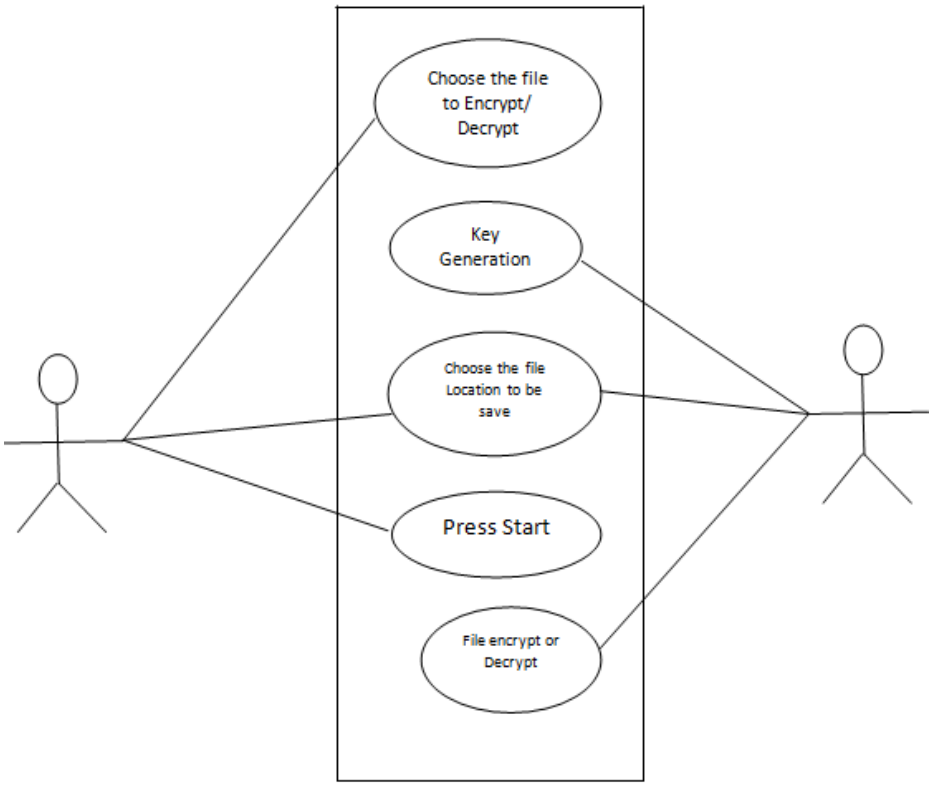

Fig. 1.6. UML use case diagram for encryption and decryption using OTP

3.4 Analysis of the Proposed System

Unlike the existing system, the proposed system has capability to manage the encryption of Doc, Pdf, ppt, image files and even multimedia file such as Audios, Videos Voice recordings etc.

The new system is to a large extent secure since a key once generated cannot be used again for the same session of transaction - meaning a unique key is generated for every transaction; thus making it difficult for unauthorized persons with malicious intents to access records other than their own.

Also the proposed system is scalable, making it possible for modification to handle larger files to suit future needs with growing student population.

3.5 Design Specification and Requirement

Hardware Requirements:

i. Intel Pentium IV processor
ii. RAM size 256 MB
iii. Hard disk 80 GB

Software Requirements:

i. Windows 7 Operating System above
ii. Pelles C for windows version 9.00.9

4. IMPLEMENTATION

The results of this study after testing the application that implements the one-time pad algorithm are as follows:

A Tests of Document Format Doc.

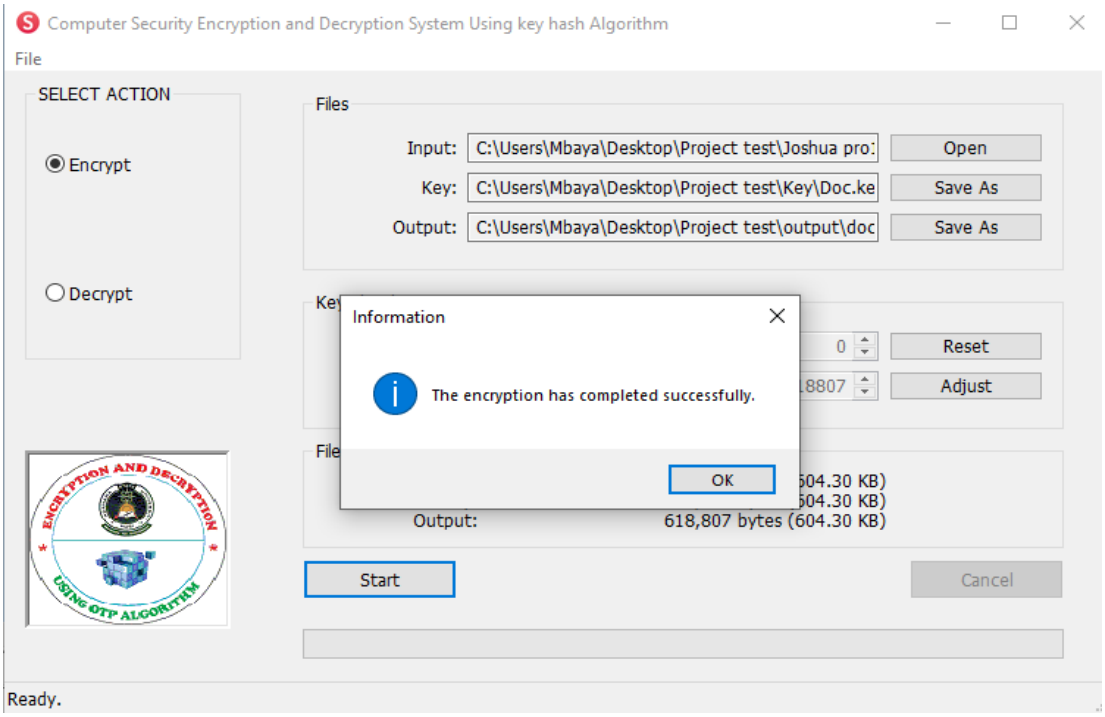

Fig. 1.7. Encryption process on Doc file

A Test result on document formats doc The encryption process was carried out in doc format documents with long file size of 604KB with the encryption process 31 milliseconds and the speed of the process is 618,807 bytes as shown in Fig. 1.7.

A Test result of Encrypted Doc File

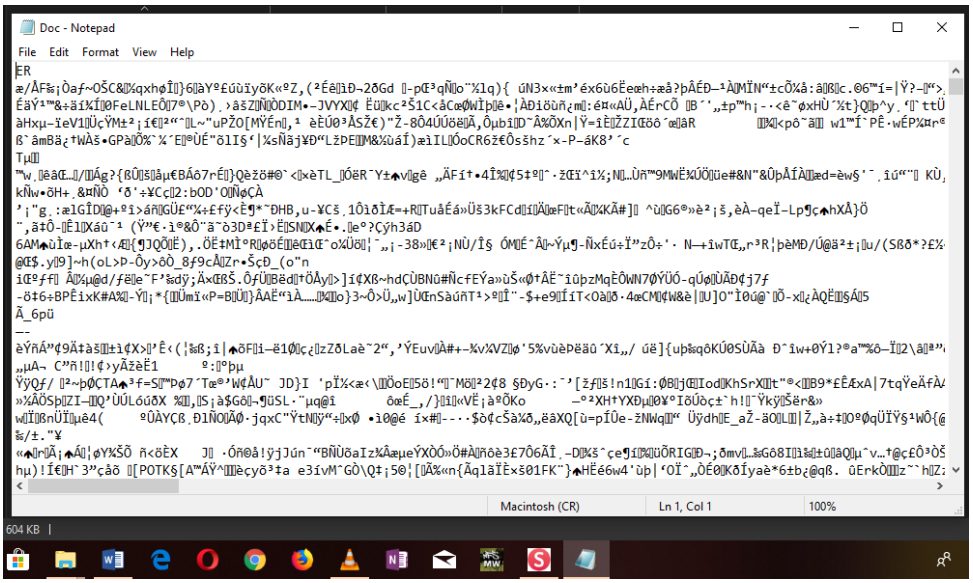

Fig. 1.8. The result of encrypted doc file

After performing the encryption process, it is expected that application will change the original file of the document. Results encryption of documents with the doc format.

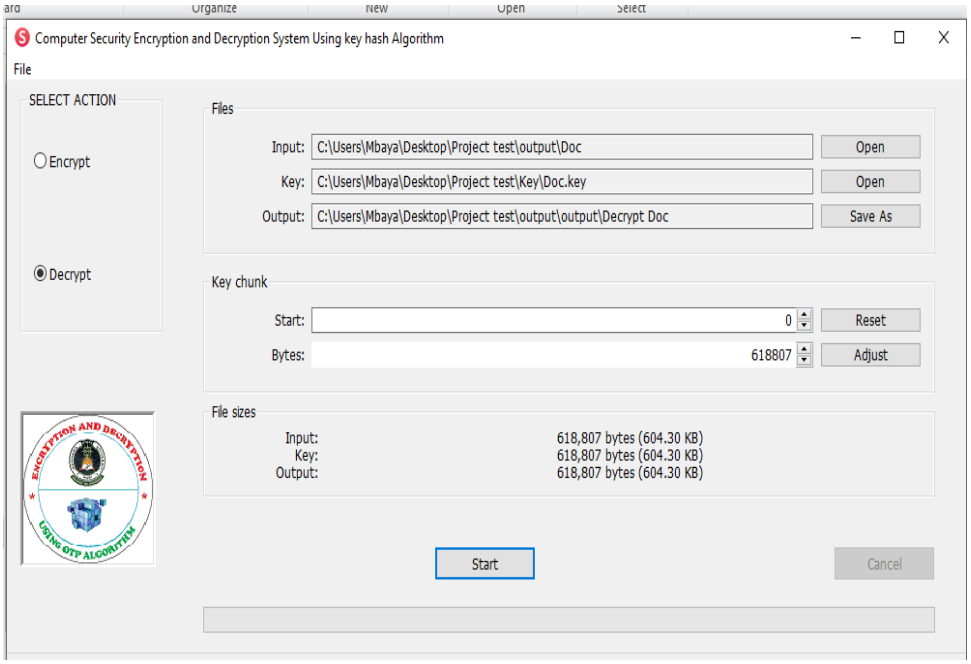

Fig. 1.9. Decryption process of doc file

The process of Decryption file to doc format can be seen in Fig. 1.9. The decryption of the process is done with a long process that is 16 milliseconds to speed the process of 618,807 bytes / mDtk with 604KB file size.

A Test Result on Decrypted Doc File

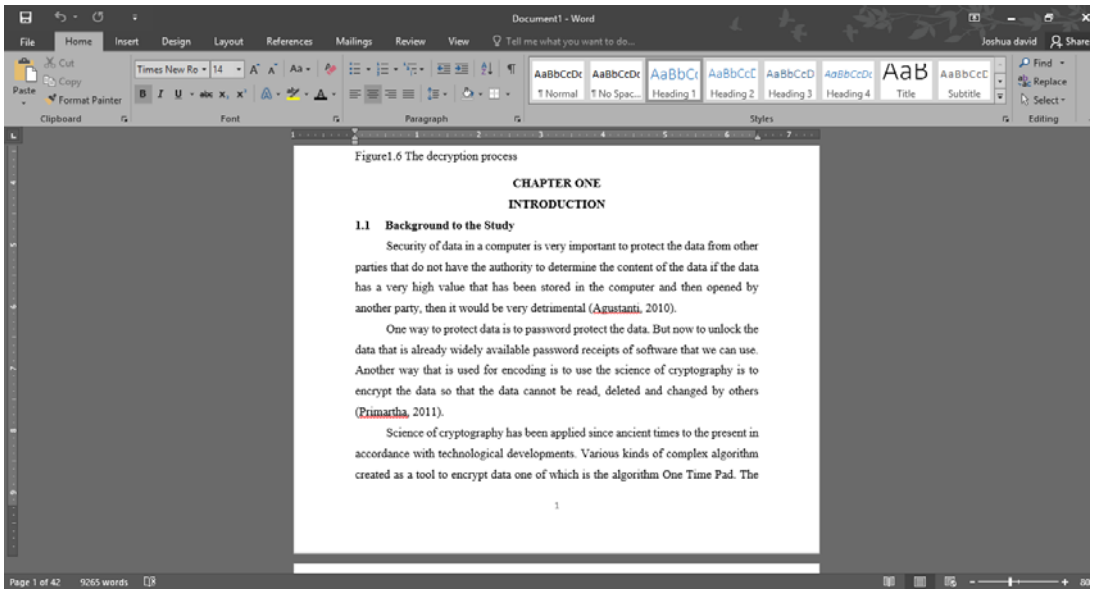

Fig. 2.0. The result of decrypted doc file format to plaintext

A Test of Pdf File Format

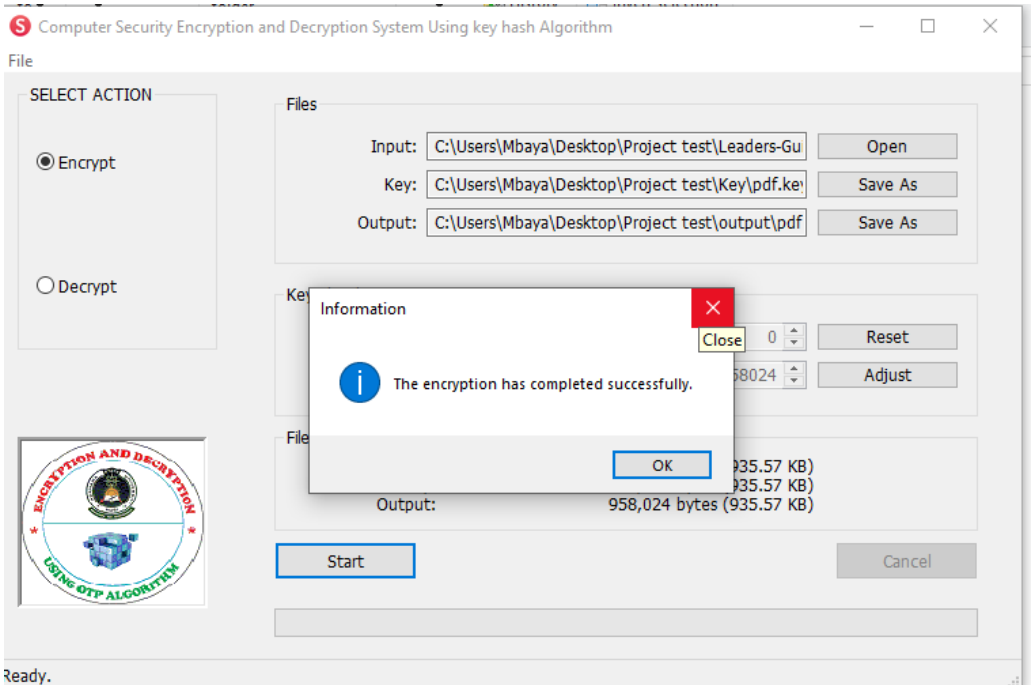

Fig. 2.1. A test on a PDF file with

The Encryption process of pdf file format performed on a PDF file The encryption process is performed on a file with a size of 935.57 KB with the old encryption process 62 milliseconds to speed the process 958,024 bytes / mDtk as shown in Fig. 2.1.

A Test Result of Encrypted Pdf File

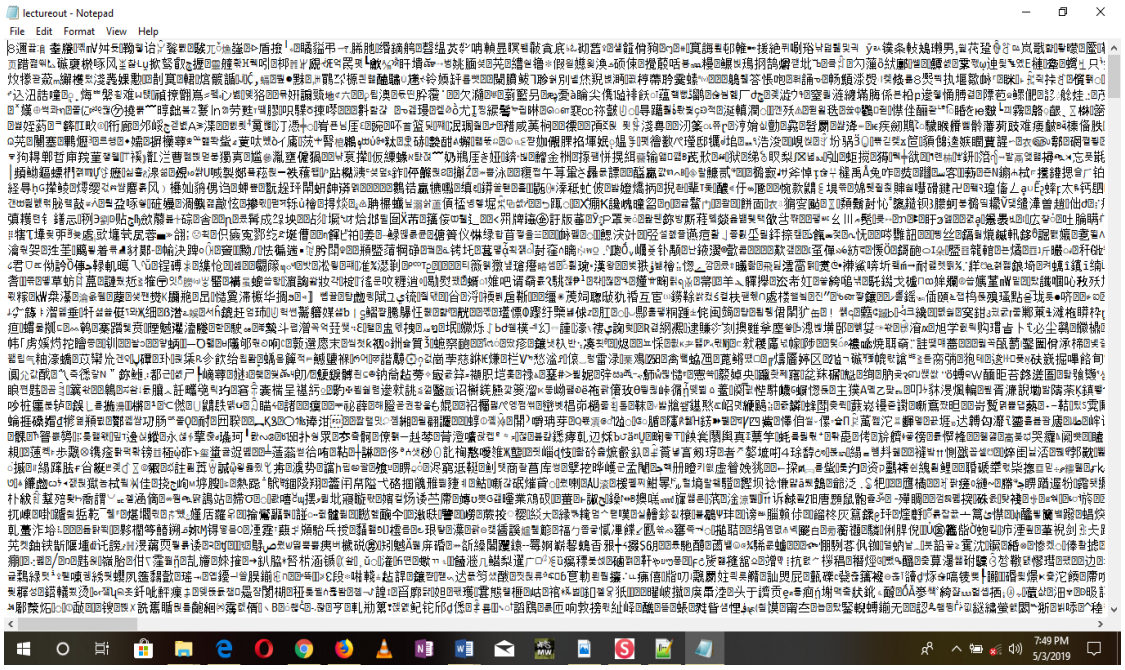

Fig. 2.2. The result of encrypted pdf file format after PDF encryption process

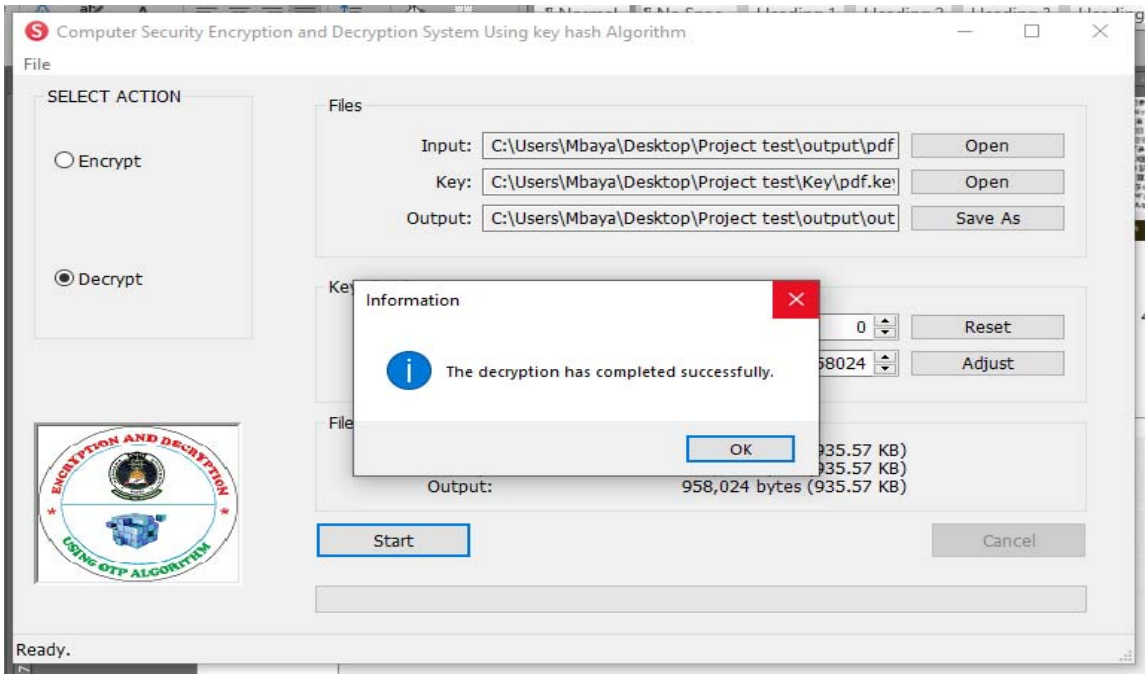

Fig. 2.3. The decryption process of pdf file format which lasted Decryption process lasted 62 milliseconds and with speed process 29 493 bytes / mDtk as shown

A Test Result on Decrypted PDF File

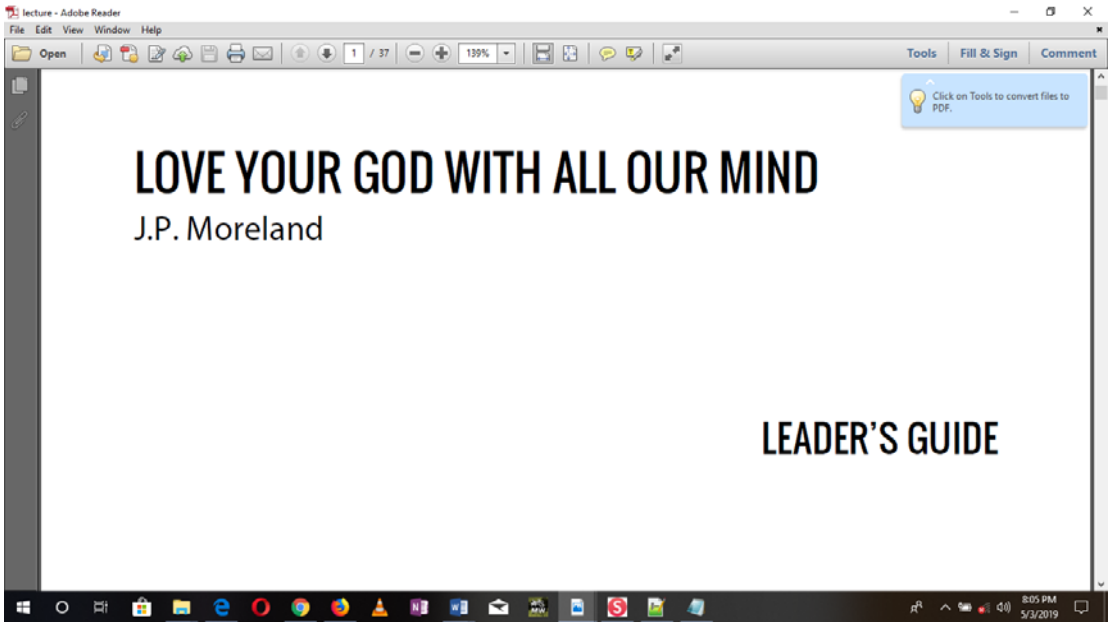

Fig. 2.4. The result of decrypted pdf file format

A Test Result on Image

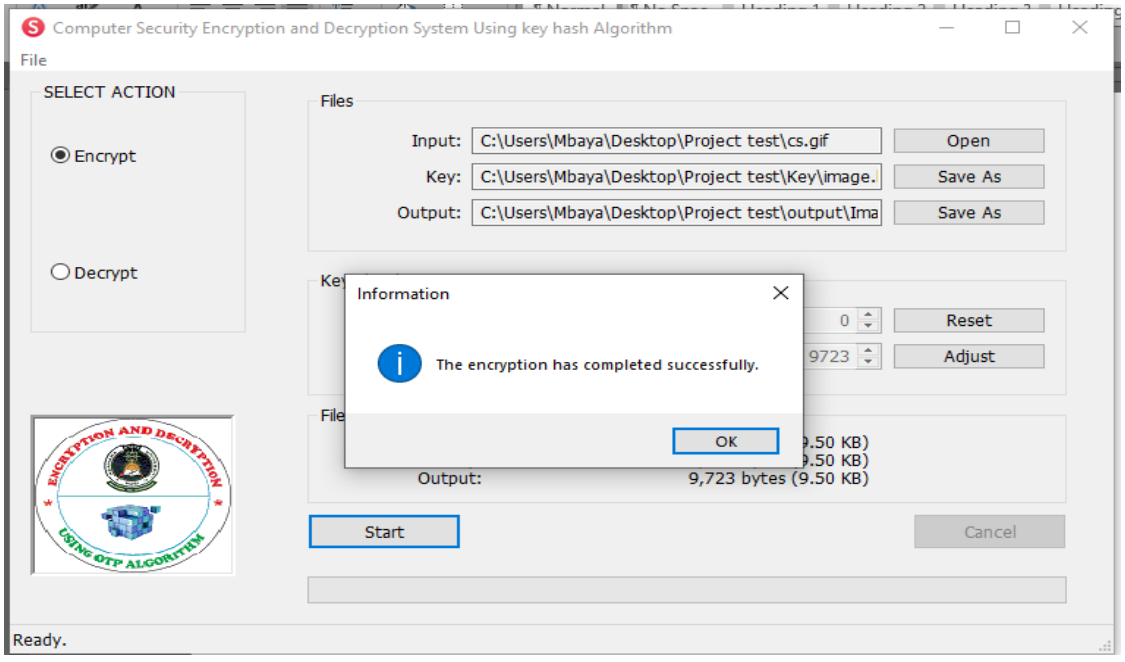

Fig. 2.5. Encryption process of image encryption process of image files can be seen in Fig. 2.5 with 125 milliseconds long process encryption as well as the speed of 9723 bytes / mDtk

A Test Result on Encrypted Image File

Fig. 2.6. Result of encrypted image

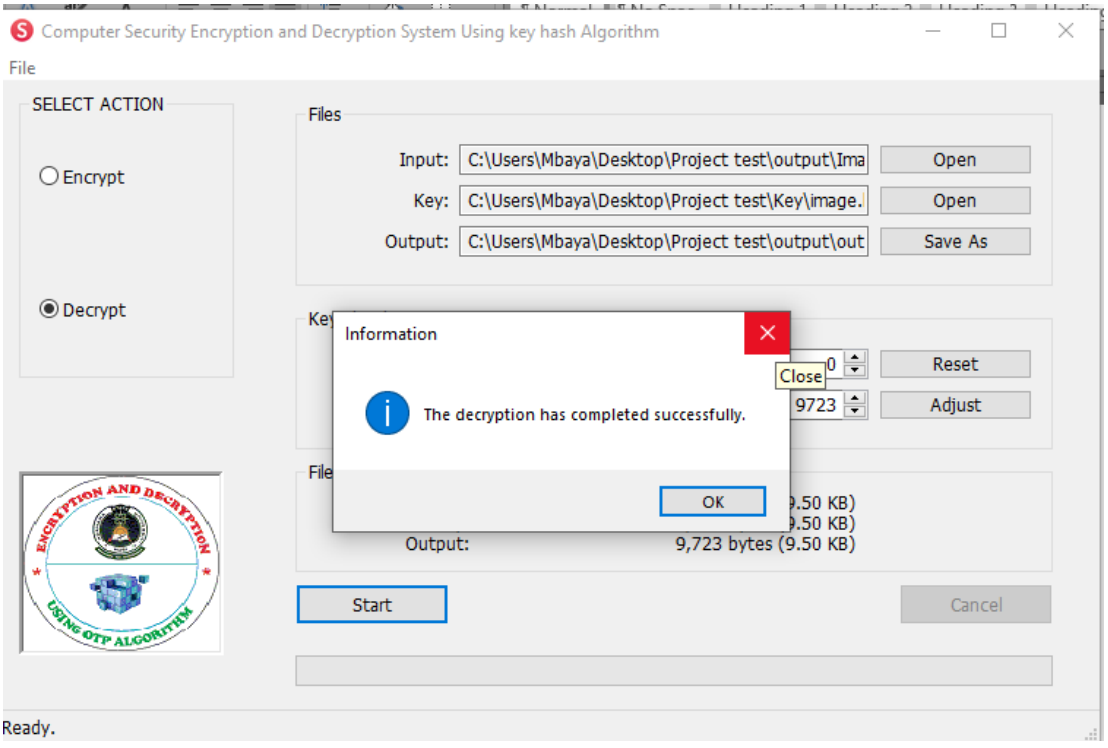

Fig. 2.7. Decryption process of image

A Test Result on Decrypted Image

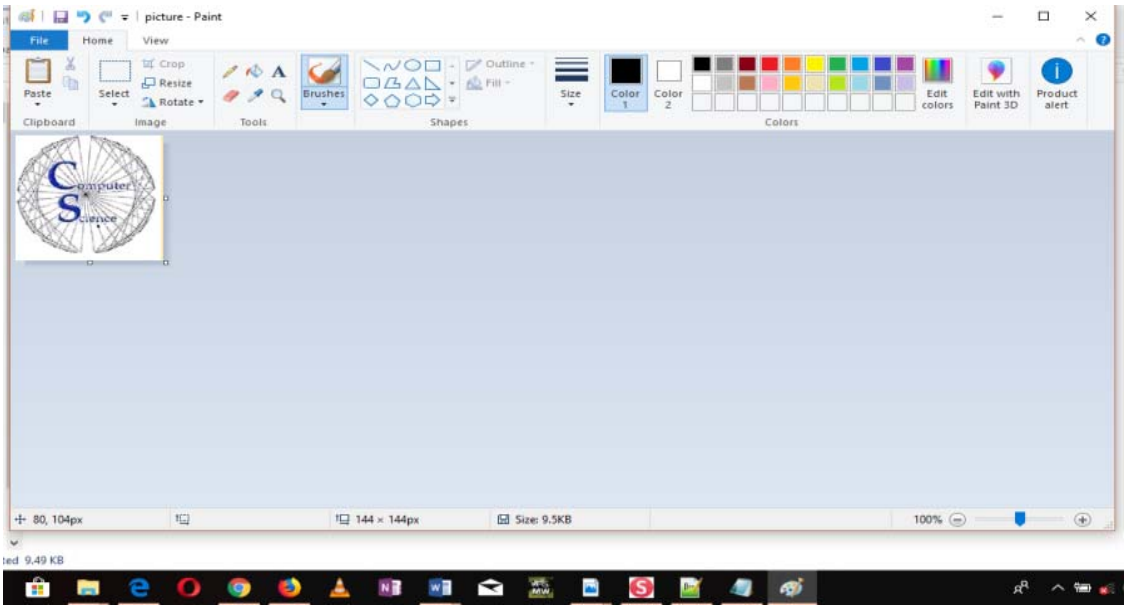

Fig. 2.8. Result of decrypted image

A Test Result on Multimedia

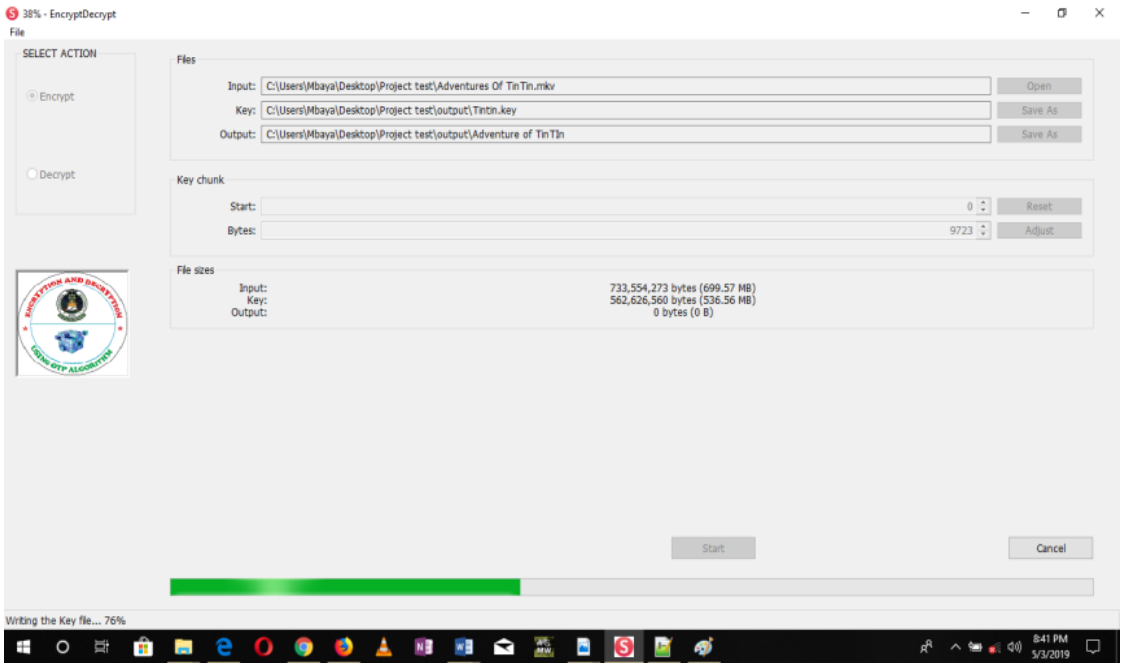

Fig. 2.9. The encryption of multimedia (Video)

The encryption process is carried out in Multimedia file with long file size of 699.57 MB with the encryption process 100 milliseconds and the speed of the process is 733,554,273 bytes as shown in Fig. 2.9.

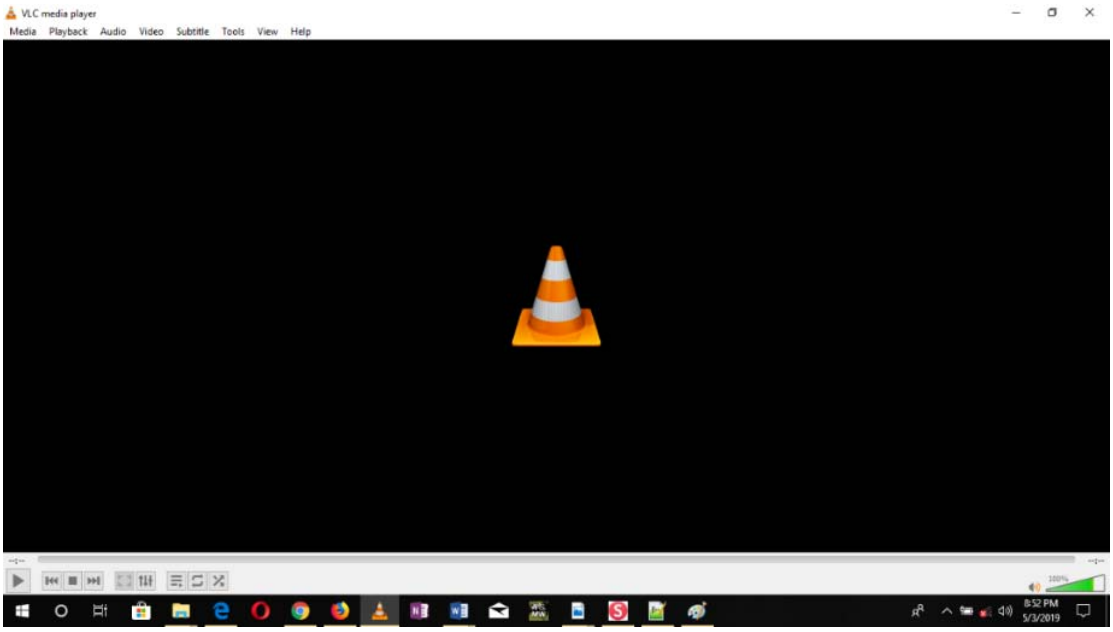

Fig. 3.0. Depict the encrypted video, result of the encrypted multimedia file

The Decrypted Process of Multimedia File

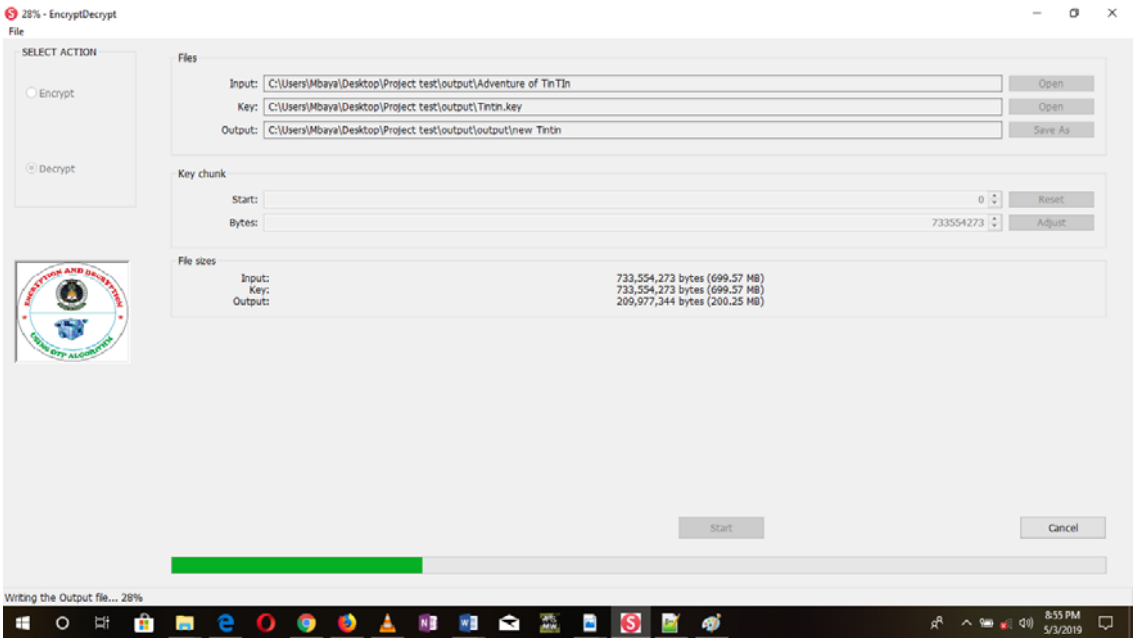

Fig. 3.1. Decryption process of multimedia

The decryption process is carried out in Multimedia file with long file size of 699.57 MB with the decryption process 100 milliseconds and the speed of the process is 733,554,273 bytes as shown in Fig. 2.9.

A Test Result on Multimedia File

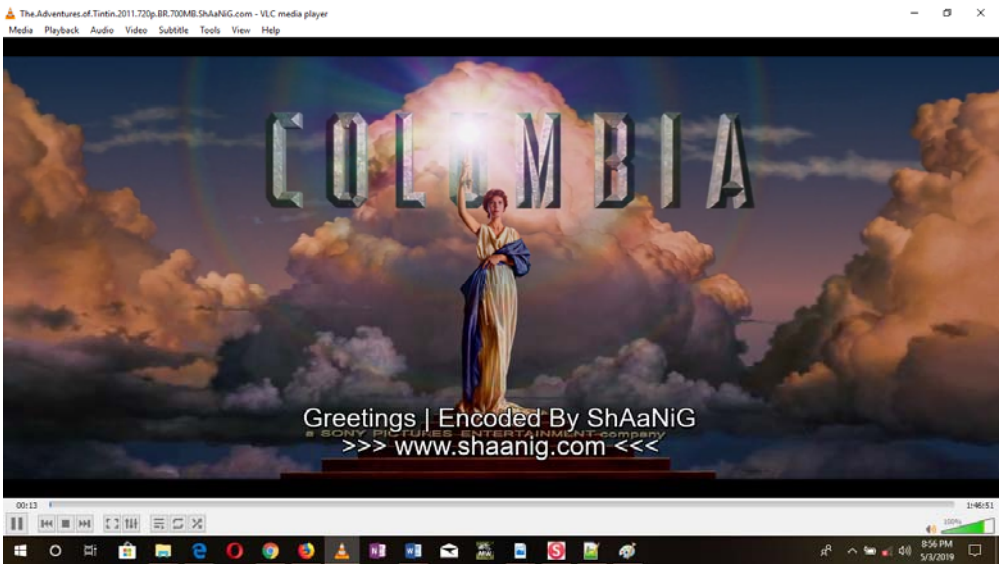

Fig. 3.2. Result of decrypted multimedia file

5. CONCLUSION

The One-Time Pad is said to be simple and theoretically unbreakable yet it is not being implemented in a practical way due to the problem of key management and distribution of keys. Good ciphers become useless if mismanaged and improperly implemented. Therefore, this study has enhanced the practical difficulty of One-Time Pad Algorithm by resolving the key management/distribution problem. In conclusion, the research objectives have been achieved, as the proposed scheme or algorithm resolves and manages the problem of having a student data and multimedia file of the Department of Computer Science Adamawa State University. Thereby solving the problem of key distribution and management in One-Time Pad encryption scheme.

Future work can be on how to improve on the long key of encryption OTP using a key scheduling algorithm why still maintaining the “Perfect Security” cliche we know of OTP.

This algorithm has a lot of scope to enhance the security by combining the different approaches.

COMPETING INTERESTS

Authors have declared that no competing interests exist.

REFERENCES

1. Agustanti SP, Pengamanan KE. One-Time Pad (Otp) Menggunakan Enkripsi Rsa. 2010;19(1):95-100.
2. Primartha R. Penerapan enkripsi dan dekripsi file menggunakan algoritma Data Encryption Standard (DES). 2012;90(1): 3-8.
3. Munir R. Algoritma enkripsi citra dengan Pseudo One-Time Pad yang Menggunakan System. 2012;20(3):3-6.
4. Srikantaswamy SG, Phaneendra HD. Enhanced one time pad cipher with more arithmetic and logical operations with flexible key generation algorithm, International Journal of Network Security & Its Applications. 2011;20(1):243-248.
5. Borowski M, Lesniewicz M. Modern usage of "old" onetime pad, IEEE Conference on Communications and Information Systems. 2012;1-5.
6. Patil R, Devare M, Kumar A. Modified one time pad data security scheme: Random key generation approach. International Journal of Computer Science and Security. 2009;3(2):138-145.
7. Penchalaiah P, Reddy KR. Efficient and secure encryption schema based on random bit"s (Rbits). International Journal of Advanced Research in Computer Science and Software Engineering. 2013;3(10):1026-1027.

8. Zaeniah, Purnama BE. An analysis of encryption and decryption application by using one time pad algorithm (IJACSA). International Journal of Advanced Computer Science and Applications; 2015.
9. Katti J, Pote S, Lande BK. Two-level encryption based on one time pad and koblitz method of encoding. International Journal of Computer Applications. 2015;40(3):34-36.
10. Devipriya, Lesniewicz. International Journal of Advanced Research in Computer Science and Software Engineering. 2015;5(6):220-223.
11. Upadhyay G, Nene MJ. One-time pad generation using quantum superposition states. In Recent Trends in Electronics, Information & Communication Technology (RTEICT), May, IEEE International Conference. 2016:1882-1886.
12. Miyano T, Cho K. Chaos-based one-time pad cryptography. In Information Theory and Its Applications (ISITA), International Symposium. 2016;156-160. IEEE. [22]
13. Lange T, Takagi T. Post-quantum cryptography 8th International workshop. PQ Crypto, Utrecht, The Netherlands. 2017;26-28. Proceedings.
14. Abiodun EO, Aman J, Oludare IA, Humaira A. An enhanced practical difficulty of one-time pad algorithm resolving the key management and distribution problem proceedings of the International Multi Conference of Engineers and Computer Scientists. 2018;44(5):9-11. Hong Kong.

Peer-review history:
The peer review history for this paper can be accessed here:
http://www.sdiarticle3.com/review-history/49658