A Project Report

On

A NEW ENCRYPTION TECHNIQUE

Project ID: PCS 21-42

submitted for partial fulfillment of the requirements

for the award of the degree of

Bachelor of Technology

in

Computer Science and Engineering

Submitted by

Kartikeya Yadav

Himanshu Yadav

Under the supervision of

Prof. Hriday Kumar Gupta

KIET Group of Institutions, Ghaziabad

Dr. A.P.J. Abdul Kalam Technical University, Lucknow

July, 2021

DECLARATION

We hereby declare that this submission is our own work and that, to the best of our knowledge and belief, it contains no material previously published or written by another person nor material which to a substantial extent has been accepted for the award of any other degree or diploma of the university or other institute of higher learning, except where due acknowledgment has been made in the text.

Signature: Himanshu Yadav	**Signature:** Kartikeya
Name: Himanshu Yadav	**Name:** Kartikeya Yadav
Roll No.: 1702910075	**Roll No.:** 1702910082
Date: 20/07/2021	**Date:** 20/07/2021

CERTIFICATE

This is to certify that Project Report entitled “KH Security (A new Encryption Technique)” which is submitted by Himanshu Yadav & Kartikeya Yadav in partial fulfillment of the requirement for the award of degree B. Tech. in Department of Computer Science & Engineering of Dr. A.P.J. Abdul Kalam Technical University, Lucknow is a record of the candidates own work carried out by them under my supervision. The matter embodied in this report is original and has not been submitted for the award of any other degree.

Date: 20/07/2021

Supervisor:

HKgupta

Mr. Hriday Kumar Gupta
(Assistant Professor)

ACKNOWLEDGEMENT

It gives us a great sense of pleasure to present the report of the B. Tech Project undertaken during B. Tech. Final Year. We owe special debt of gratitude to Professor Hriday Kumar Yadav, Department of Computer Science & Engineering, KIET, Ghaziabad, for his constant support and guidance throughout the course of our work. His sincerity, thoroughness and perseverance have been a constant source of inspiration for us. It is only his cognizant efforts that our endeavors have seen light of the day.

We also take the opportunity to acknowledge the contribution of Dr. Vineet Sharma, Head of the Department of Computer Science & Engineering, KIET, Ghaziabad, for his full support and assistance during the development of the project. We also do not like to miss the opportunity to acknowledge the contribution of all the faculty members of the department for their kind assistance and cooperation during the development of our project.

We also do not like to miss the opportunity to acknowledge the contribution of all faculty members of the department for their kind assistance and cooperation during the development of our project. Last but not the least, we acknowledge our friends for their contribution in the completion of the project.

Signature: Himanshu Yadav	**Signature:** Kartikeya
Name: Himanshu Yadav	**Name:** Kartikeya Yadav
Roll No.: 1702910075	**Roll No.:** 1702910082
Date: 20/07/2021	**Date:** 20/07/2021

TABLE OF CONTENTS

TABLE OF FIGURES

ABSTRACT

As increase in the technology like IOT and Cloud computing the demand of data security is also increasing. Here the encryption algorithms came into existence. It is necessary that the algorithms must be secured and efficient so that it can also be transmit over the network, some popular one's available to us are AES, DES, TRIPLE DES, BLOWFISH, RC5 etc. In this report we will see how various encryption techniques are work and a new encryption technique is implemented on the data to make it more secure than the available encryption techniques.

CHAPTER 1

INTRODUCTION

Cryptography refers to a method of protecting information and communications through the utilization of codes, in order that only those for whom the knowledge is meant can read and process it. Cryptography, a word with Greek origins, means “secret writing” is that the science of devising methods that leave information to be sent during a secure form in such a way that the sole person ready to retrieve this information is that the intended recipient. The message to be sent through an unreliable medium is understood as plaintext, which is encrypted before sending over the medium. The encrypted message is understood as cipher text, which is received at the opposite end of the medium and decrypted to urge back the first plaintext message. Hence a cryptosystem is a collection of algorithms and associated procedures for hiding and revealing information.

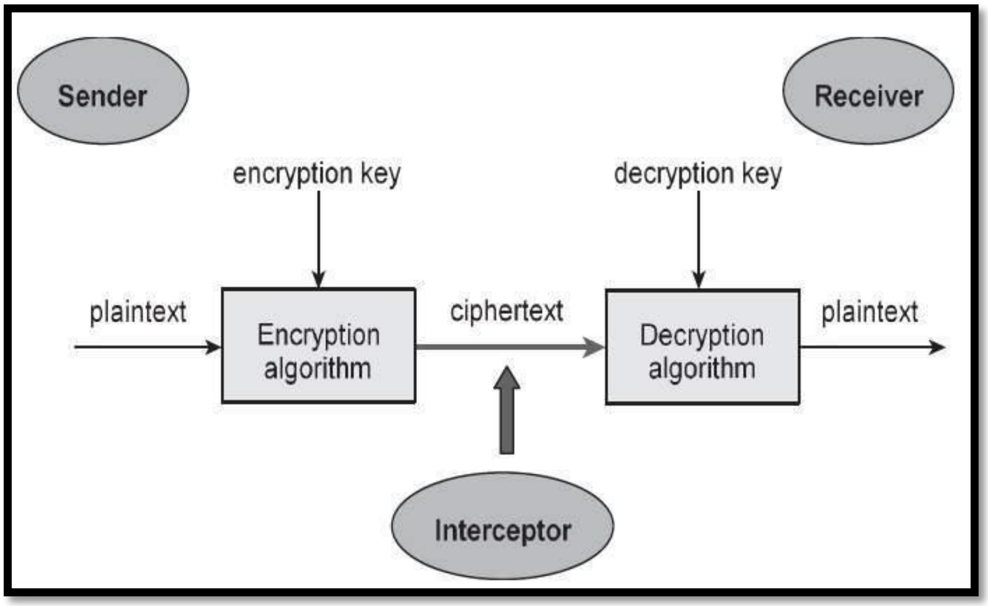


Figure 1.Cryptography

Encryption is a process of encoding a message or a file so that it can be only be read by certain people. Encryption uses an algorithm to scramble, or encrypt, data and then uses a key for the receiving party to unscramble, or decrypt, the information.

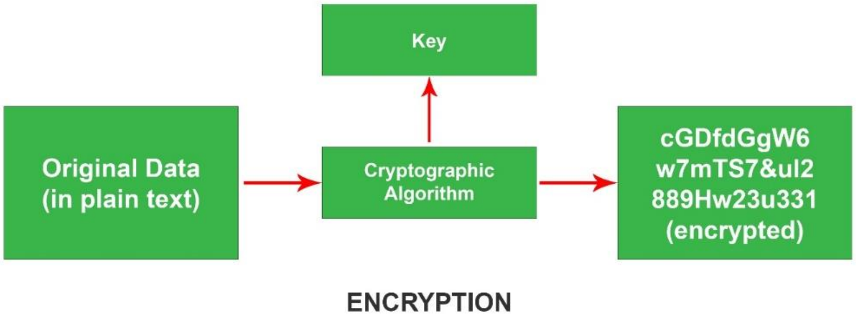

Key
Original Data (in plain text)
Cryptographic Algorithm
cGDfdGgW6 w7mTS7&ul2 889Hw23u331 (encrypted)
ENCRYPTION

Figure 2. Encryption

Tools available for encryption technique: -

-> Triple DES - uses 3 individual keys with 56 bits.

-> RSA - asymmetric key encryption

-> Blowfish - splits the message into 64 bits and encrypts

-> Two fish - symmetric and uses 256 bits in length.

-> AES - Advanced and can encrypt 128-bit, 192 bit as well as 256-bit.

Software and applications are available for encryption Technique: -

-> LastPass - generate strong and secure passwords.

-> BitLocker - Integrated in Windows OS

-> VeraCrypt - used in cross platforms

-> Disk Cryptor - used to even hide system partitions and ISO images.

-> HTTPS Everywhere - authentication process while connecting to a secure website.

-> VPN’s - used to ensure that the web traffic and data remains encrypted.

-> Using online proxy servers, we can hide the IP address and surf anonymously.

Types of Encryptions

Symmetric and asymmetric encryption are two main subgroups of encryption.

Symmetric encryption

It uses the same key for encryption and decryption. Because it uses the same key, symmetric encryption can be more cost effective for the security it provides. That said, it is important to invest more in securely storing data when using symmetric encryption.

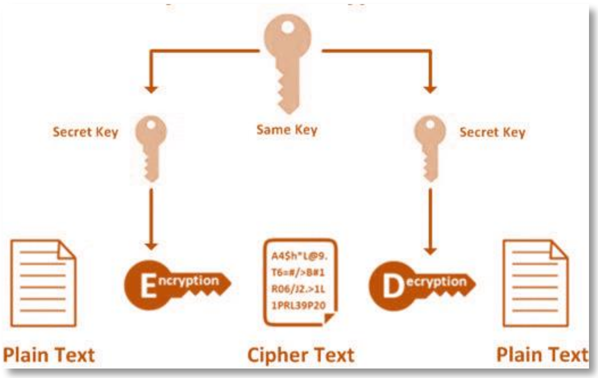


Figure 3.Symmetric Encryption

Asymmetric encryption

It uses two separate keys: a public key and a private key. Often a public key is used to encrypt the data while a private key is required to decrypt the data. The private key is only given to users with authorized access. As a result, asymmetric encryption can be more effective, but it is also more costly.

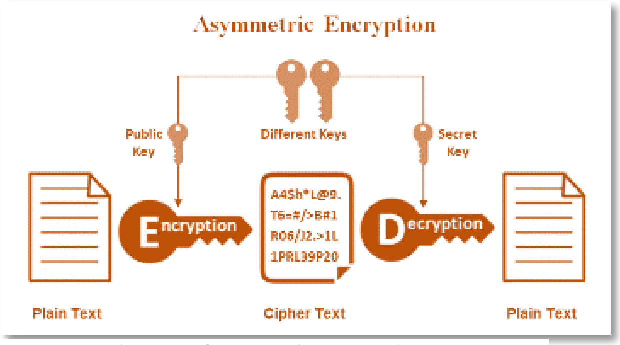


Figure 4. Asymmetric Encryption

CHAPTER 2

ENCYPTION ALGORITHMS & THEIR PERFORMANCE

An encryption algorithm is a method of transforming data into ciphertext. An algorithm use the encryption key in order to alter the data in a predictable way, so that even though the encrypted data will appear randomly, it can be turned back into plaintext by using the key. Let us discuss one bye one commonly used Encryption Algorithms:

1. AES ENCRYPTION

AES Encryption technique is a symmetric encryption technique. In this the algorithm takes plain text in blocks of 128 bits and converts them to cipher text using 128-bit, 192 bit, and 256 bit. It is publicly accessible.

WORKING

1. 128 bit (16 bytes) represented in 4x4 matrix and AES operates on a matrix of bytes.
2. Algorithm uses different number of rounds which depends on the key. 10 rounds for 128 bit ,12 rounds for 192 bit and 14 rounds for 256 bit.
3. It works with the help of two techniques which are substitution and permutation network (SPN).
4. SPN is generally considered as weak on its own, but a line of substitution followed by a permutation has good mixing properties.
5. Substitution replaces plaintext letters or strings of letters by letters, number or symbols. Permutation uses the plaintext message letters but rearranges their order.[4]
6. This whole algorithm uses some steps which are –
   a) Substitution of bytes.
   b) Shifting the rows.(Permutation)
   c) Mixing the columns.
   d) Adding the round key/Final step

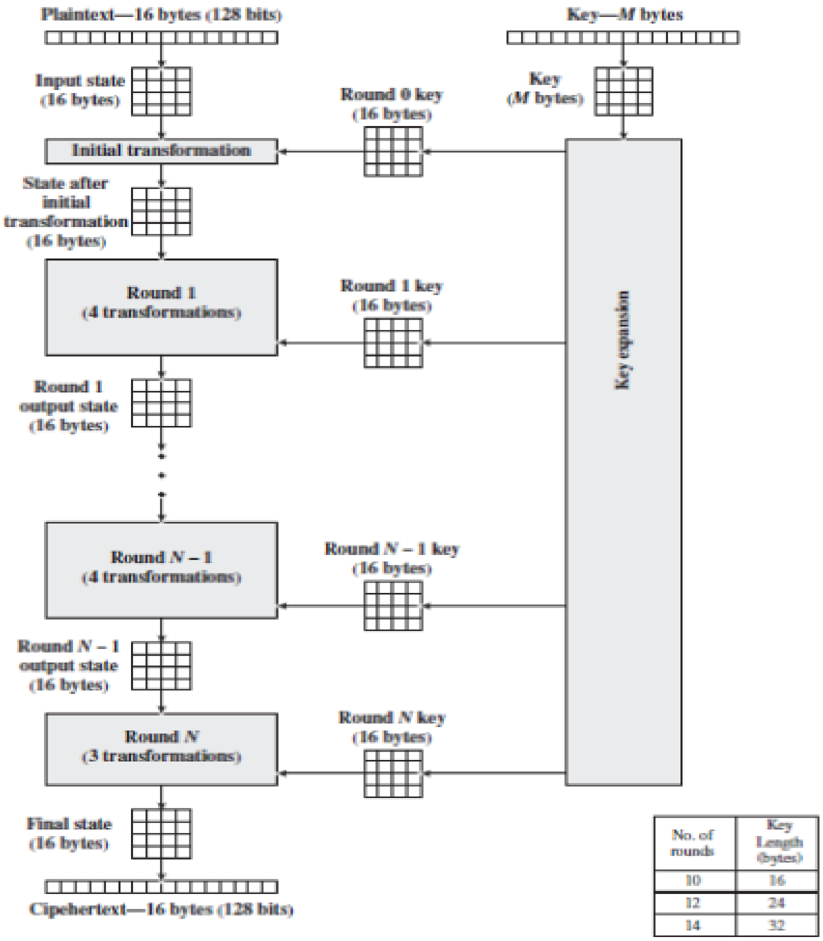
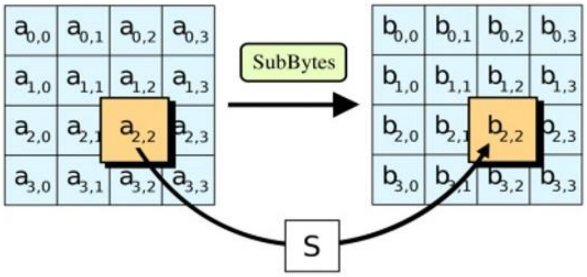

Plaintext—16 bytes (128 bits)
Key—M bytes
Input state (16 bytes)
Key (M bytes)
Round 0 key (16 bytes)
Initial transformation
State after initial transformation (16 bytes)
Round 1 (4 transformations)
Round 1 key (16 bytes)
Key expansion
Round 1 output state (16 bytes)
Round N – 1 (4 transformations)
Round N – 1 key (16 bytes)
Round N – 1 output state (16 bytes)
Round N (3 transformations)
Round N key (16 bytes)
Final state (16 bytes)
Cipehertext—16 bytes (128 bits)
No. of rounds
Key Length (bytes)
10
16
12
24
14
32

Figure 5. AES encryption

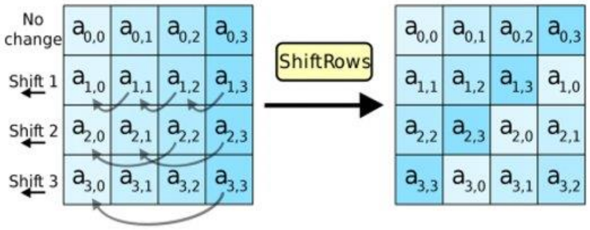
a) SUBSTITUTION OF THE BYTES

This is the first step where the block text are substituted based on the rules which are predefined in S-boxes.


$a_{0,0}$ $a_{0,1}$ $a_{0,2}$ $a_{0,3}$
$a_{1,0}$ $a_{1,1}$ $a_{1,2}$ $a_{1,3}$
$a_{2,0}$ $a_{2,1}$ $a_{2,2}$ $a_{2,3}$
$a_{3,0}$ $a_{3,1}$ $a_{3,2}$ $a_{3,3}$
SubBytes
$b_{0,0}$ $b_{0,1}$ $b_{0,2}$ $b_{0,3}$
$b_{1,0}$ $b_{1,1}$ $b_{1,2}$ $b_{1,3}$
$b_{2,0}$ $b_{2,1}$ $b_{2,2}$ $b_{2,3}$
$b_{3,0}$ $b_{3,1}$ $b_{3,2}$ $b_{3,3}$
S

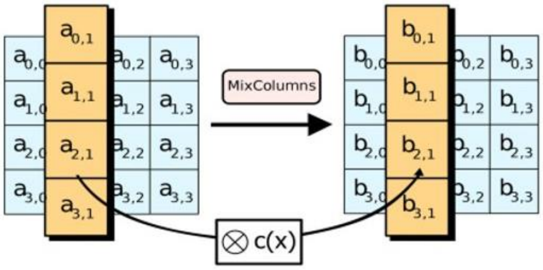
b) SHIFTING THE ROWS

In this step the permutation occurs where all rows except the first are shifted by one.



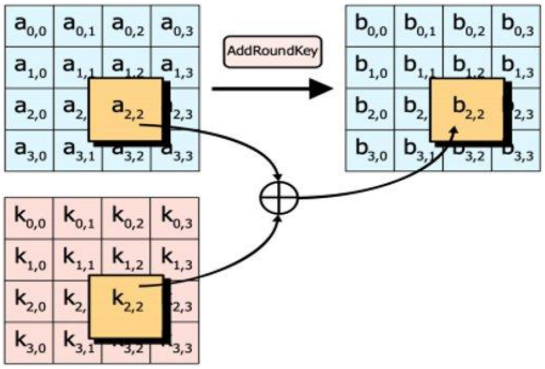
c) MIXING OF COLUMNS

In this step, the algorithm hill cipher Is used to jumble up the message more by mixing the blocks columns.



d) ADDING THE ROUND KEY/FINAL STEP

This is the final step of the algorithm, here the message is XORed with the respective round key.



2. DES ENCRYPTION

DES is also a symmetric encryption which converts 64 bits plaintext into 64 bits cypher text.

Block size (Plaintext) – 64 bits

Key - 64 bit

Rounds – 16

WORKING

1. The plaintext has to undergo an initial permutation before passing into successive rounds.

2. In different rounds the s-boxes are used along with XOR operations with the keys.

3. The key is of 64 bit from which 56 bits are selected for further operation (Permuted choices) by removing 8 parity bits.

4. Further these 56 bits are also divided into 28 bits for left shifting and after they are again joined to form 56 bits.

5. Key again undergo PC2 (Permuted choice2) and the bits remained are just 48 bits, which are XORed with the plaintext.

6. The same process happen again and again for 16 times and in the end we get our 64 bit cypher text.

DISADVANTAGES OF DES (FOUNDATION OF TRIPLE DES)

1. Algorithm has 56 bit key so we have 5^56 combinations which can easily be broken (using brute force attack).

2. It was decided to use DES twice (double des), here we have 112 bit key, 56 bit key is used for one DES and another 56 bit is used for second DES.

3. Double DES is vulnerable to meet in the middle attack. Therefore triple DES is founded.

4. Triple DES is fully provides the security of 2^112.

Algo Representation

Cypher text – Ek3(Dk2(Ek1(PT)))

Plain text - Dk1(Ek2(Dk3(CT)))

3. TRIPLE DES

Triple DES or 3 DES is an encryption technique which was developed after the failure of DES, as DES is vulnerable to meet in the middle attack.

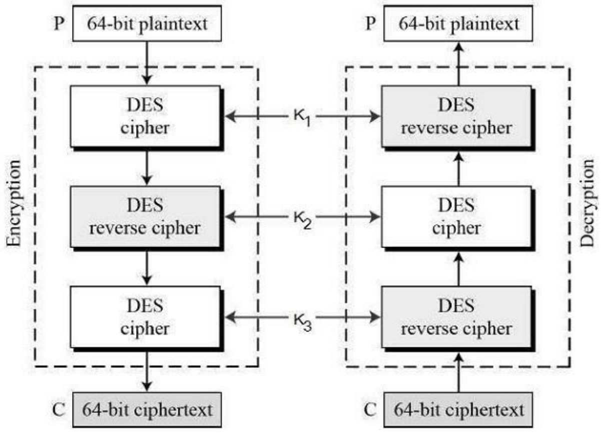


Figure 6. 3DES encryption

WORKING

1. In triple DES, first a 64-bit plain text is entered. The encryption can be done with 2 keys as well as 3 keys.

2. Two keys – The entered plaintext is encrypted using key K1 and the cypher text is generated. The cypher text is again sent for reverse cypher (means decryption) using key K2 and another cypher text is generated.

3. The generated cypher text is again encrypted using key K1 which gives us our final cypher text.

4. Three keys – The above same process is used in 3 DES but the difference is that instead of 2 , 3keys are used for encryption(K1)->decryption(K2)-> encryption(K3).

4. Blowfish

Blowfish is a symmetric encryption algorithm developed by Bruce Schneier to replace the existing Data Encryption Standard (DES). During development, most encryption algorithms were protected by patents, government secrecy, or company intellectual property at that time. Schneier placed Blowfish in the public domain i.e., not subject to any patents and is therefore freely available for anyone to use.

Working:

Blowfish has a 64-bit block size and a key length of anywhere from 32 bits to 448 bits. It is a 16-round Feistel cipher and uses large key-dependent S-boxes.

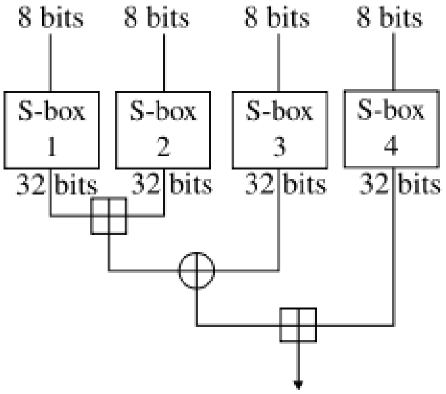


Figure 7. blowfish

This algorithm is divided into two parts are Key-expansion and Data Encryption.

1. Key-expansion: It will convert a key into several sub key arrays totaling 4168 bytes consisting at most 448 bits. Blowfish uses five subkey-arrays:

One 18-entry P-array consisting of 32-bit sub keys:
P1, P2, ..., P18 and four 256-entry S-boxes of 32-bit each:
S1,0, S1,1, ... S1,255
S2,0, S2,1, ... S2,255
S3,0, S3,1, ... S3,255
S4,0, S4,1, ...S4,255
These keys are generated earlier to any data encryption or decryption.

It is having a function to iterate 16 times of network. Each round consists of key-dependent permutation and a key and data-dependent substitution. All operations are XORs and additions on 32-bit words. The only additional operations are four indexed array data lookup tables for each round.

Algorithm: Blowfish Encryption

```
Divide x into two 32-bit halves: xL, xR for i = 1 to 16:

xL = XL XOR Pi

xR = F(XL) XOR xR

Swap XL and xR

Swap XL and xR (Undo the last swap.)

xR = xR XOR P17

xL = xL XOR P18

Recombine xL and xR[1]
```

5. RC5

RC5 is a symmetric key block encryption algorithm designed by Ron Rivest in 1994. It is notable for being simple, fast (on account of using only primitive computer operations like XOR, shift, etc.) and consumes less memory.[2]

Working:

In the RC5 algorithm, the input file size of Block size, number of rounds and 8-bit bytes of the key can be of variable length. Once the values of this are decided, the values will remain the same for a particular execution of the cryptographic algorithm. The size of plain text block can be of 32 bits, 64 bits or 138 bits. the length of the key can be of 0 to 2040 bits. The output generated by RC5 is the ciphertext which has the size the same as plain text size.

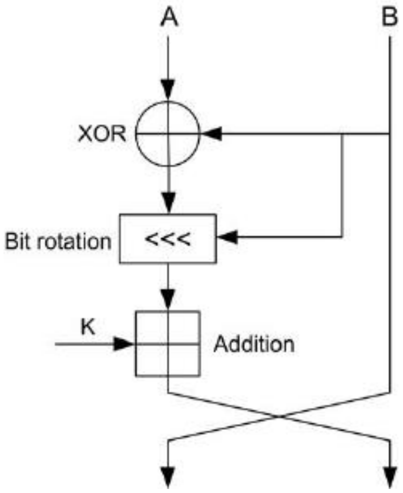
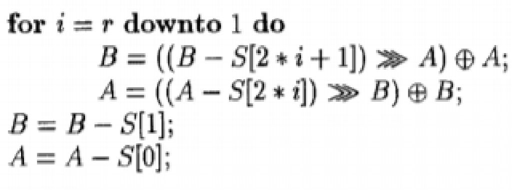


Figure 8. rc5 encryption

Encryption

We assume that the input block is given in two w-bit registers A and B. We also assume that key-expansion has already been performed, so that the array S[0...t - 1] has been computed. Here is the encryption algorithm in pseudo-code: [3]

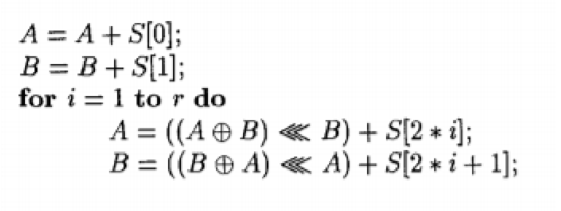


```
for $i = r$ downto 1 do
        $B = ((B - S[2 * i + 1]) \ggg A) \oplus A;$
        $A = ((A - S[2 * i]) \ggg B) \oplus B;$
$B = B - S[1];$
$A = A - S[0];$
```

The output is in the registers A and B. We note the exceptional simplicity of this 5-line algorithm. We also note that each RC5 round updates both registers A and B, whereas a "round" in DES updates only half of its registers. An RC5 "half-round" (one of the assignment statements updating A or B in the body of the loop above) is thus perhaps more analogous to a DES round.

Decryption

The decryption routine is easily derived from the encryption routine.



```
$A = A + S[0];$
$B = B + S[1];$
for $i = 1$ to $r$ do
        $A = ((A \oplus B) \lll B) + S[2 * i];$
        $B = ((B \oplus A) \lll A) + S[2 * i + 1];$
```

6. SHA: Secure Hashing Algorithm

A hash is a mathematical function which is used by computer since they are convenient to compute a hash. They identify, compare, or run calculations against files and strings of data. Hashing algorithms are used in databases, also used to store passwords.

SHA-1

-> Developed in 1993.

-> SHA-1 is a 160-bit hash.

-> Vulnerable to brute force attacks.

SHA-2

-> Developed after 2009.

-> SHA-2 is a 256 hash.

-> Brute force attacks are prevented in SHA-2.

Properties of Hash function:

-> Pre-image resistance - hard computation to reverse the hash.

-> Second pre-image resistance - hard to find the same input and hash.

-> Collision resistance - unique input and it's hash value.

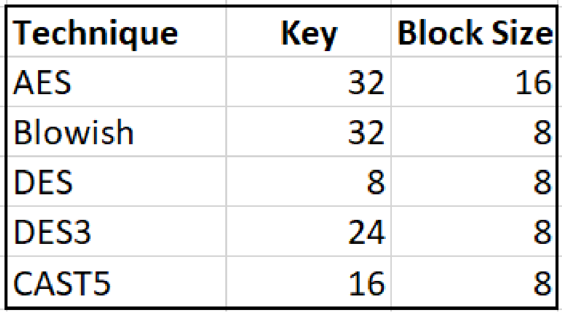
Performance Analysis & Comparison of above Algorithms:

Following Comparisons are performed in Python 3.9.6.

Processor used: Intel(R) Core (TM) i5-8250U CPU @ 1.60GHz 1.80 GHz

RAM: 8GB

OS: 64-bit operating system Windows 10



Technique	Key	Block Size
AES	32	16
Blowish	32	8
DES	8	8
DES3	24	8
CAST5	16	8

Figure 9. Properties of Algorithm

File Size	AES	Blowfish	DES	DES3	CAST5
1KB	7.64513	9.810925	11.72828674	2.525568	11.22761
10KB	13.26418	15.20085	13.70382309	10.58197	18.69273
100KB	21.76523	25.07901	22.65381813	14.020681	30.69544
1MB	25.15554	20.05696	24.72162247	47.610521	34.82008
10MB	51.06378	118.5479	176.3219833	414.85858	171.4973

Figure 10. Encryption Comparison in milliseconds

File Size	AES	Blowfish	DES	DES3	CAST5
1KB	5.138159	18.71228	16.99138	45.16554	33.0441
10KB	14.9889	18.85891	27.63176	19.91558	14.93025
100KB	15.2781	18.50033	31.56424	37.86397	14.76693
1MB	22.6512	31.07595	37.49347	47.98007	46.04673
10MB	58.94041	120.6176	179.8019	424.9966	166.2364

Figure 11. Decryption Algorithm Comparison in milliseconds

CHAPTER-3

PROPOSED ALGORITHM

Encryption -:

STEP 1. Convert the data into 128 bits or 16 bytes and store it in a 4*4 matrix.

STEP 2. Each section in matix represent a character and a column represents a 4 letter word of 32 bit.

STEP 3. Convert each character in the matrix into its ASCII value.

STEP 4. Generate the binary value from the ascii values in step 2.

STEP 5. Convert 100 into its binary and XOR it with the values in matrix.

STEP 6. Reverse the values that we got in step 5.

STEP 7. Find 1's Complement.

STEP 8. Split the binary values into two equal parts.

STEP 9. Convert each part into its decimal values as val1 and val2.

STEP 10. Here assume val1 = row and val2 = col

STEP 11. Now use val1 and val2 to find the exact value in s-box (which is a 16*16 matrix).

STEP 12. Replace our old values with the new s-box values and concatenate them to form our unreadable Cypher text which is also of 128 bits.

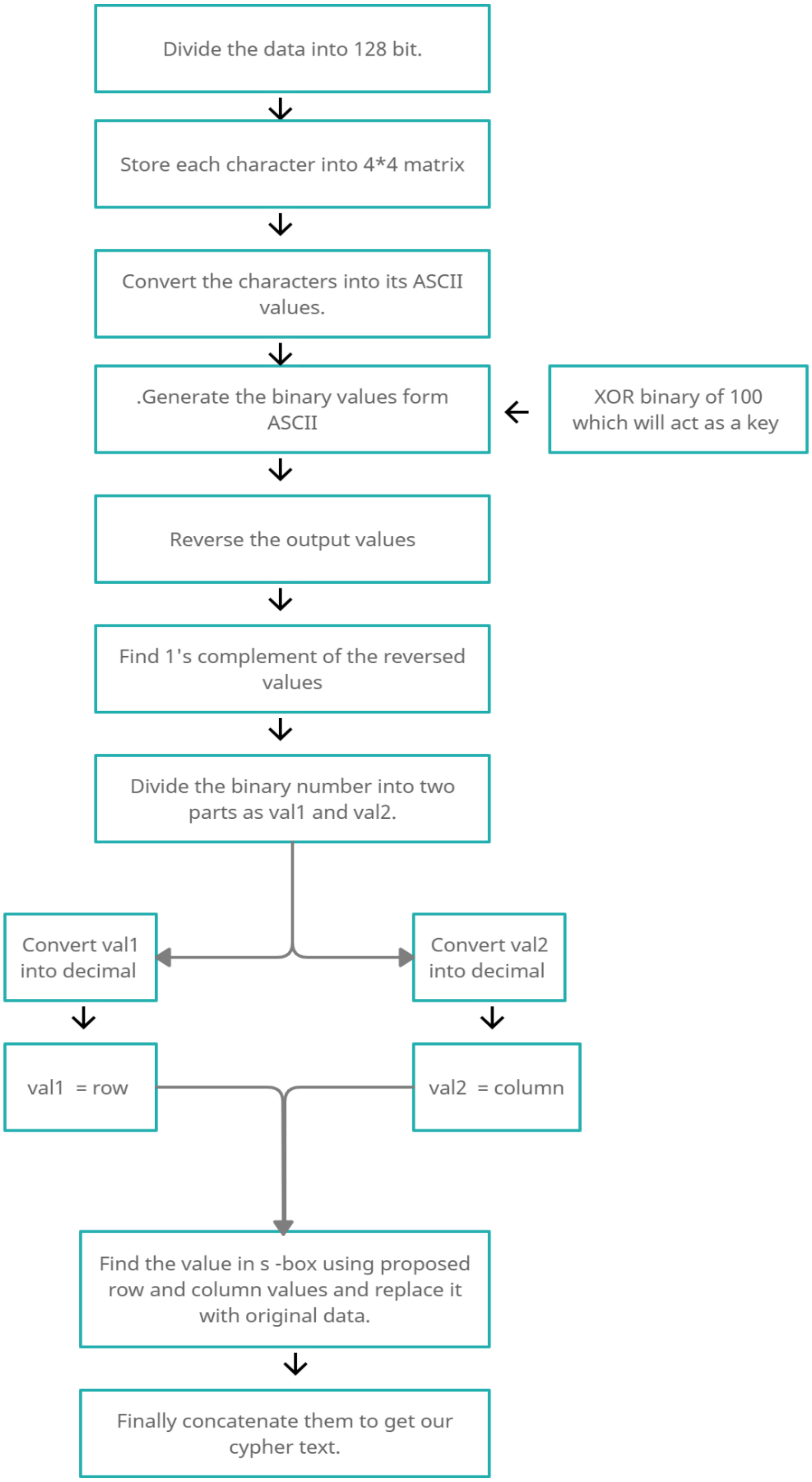

Divide the data into 128 bit.
Store each character into 4*4 matrix
Convert the characters into its ASCII values.
.Generate the binary values form ASCII
XOR binary of 100 which will act as a key
Reverse the output values
Find 1's complement of the reversed values
Divide the binary number into two parts as val1 and val2.
Convert val1 into decimal
Convert val2 into decimal
val1 = row
val2 = column
Find the value in s -box using proposed row and column values and replace it with original data.
Finally concatenate them to get our cypher text.

Figure 12. Proposed Encryption Flowchart

Decryption-:

STEP 1. Take the cypher text and put each character into the matrix.

STEP 2. Find the character from the s-box and store its row and column as val1 and val2.

STEP 3. Means val1 = row and val2 = column

STEP 4. Generate binary value of val1 and val2.

STEP 5. Do this for all the characters in derived matrix.

STEP 6. Find 1’s complement of all the values.

STEP 7. Reverse the binary numbers.

STEP 8. Now XOR all the values with binary of 100.

STEP 9. Convert the binary numbers into its ASCII.

STEP 10. Find the corresponding characters of the ASCII values.

STEP 11. Finally concatenate them to get the original text.

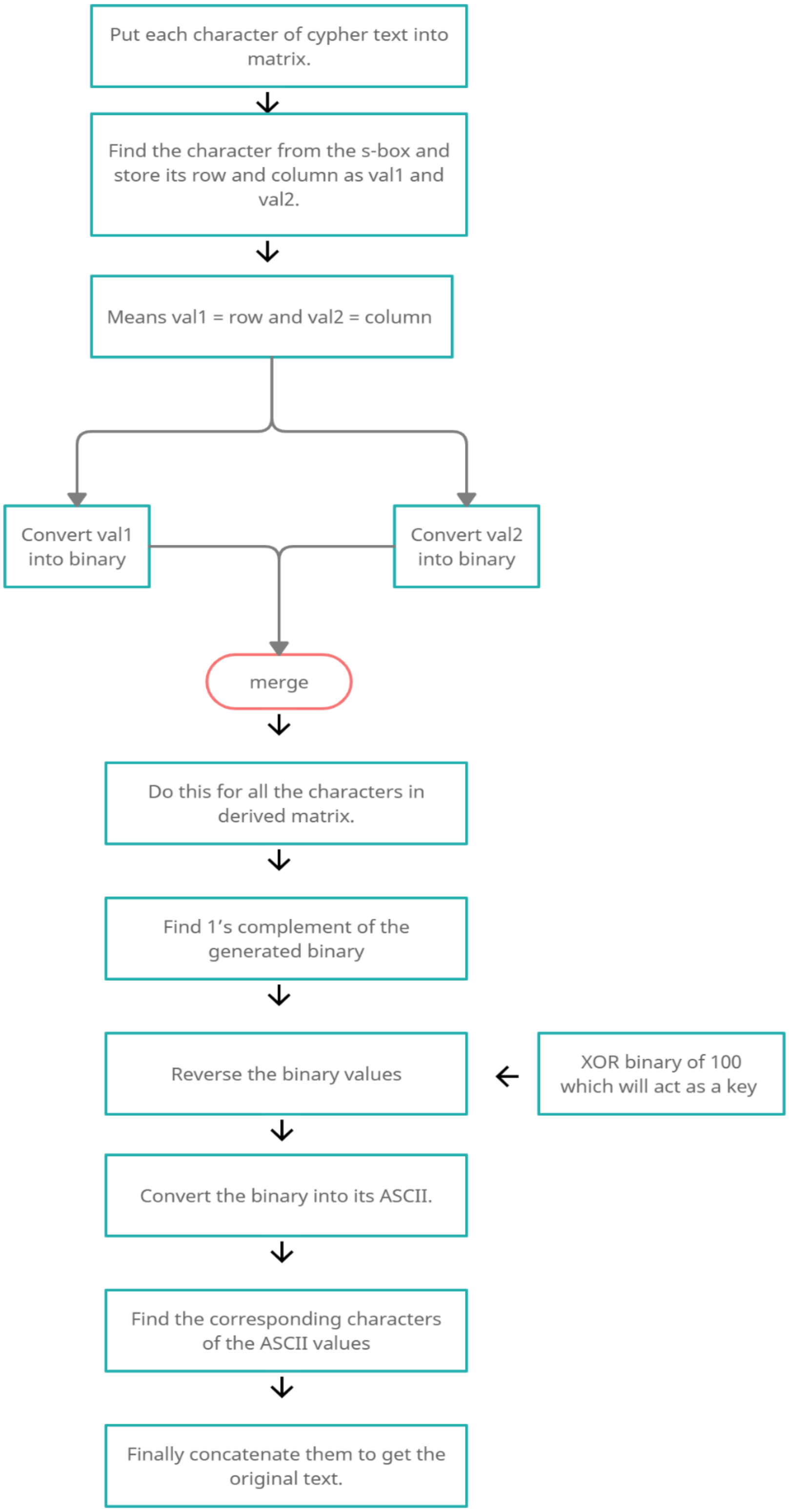

Put each character of cypher text into matrix.
Find the character from the s-box and store its row and column as val1 and val2.
Means val1 = row and val2 = column
Convert val1 into binary
Convert val2 into binary
merge
Do this for all the characters in derived matrix.
Find 1’s complement of the generated binary
Reverse the binary values
XOR binary of 100 which will act as a key
Convert the binary into its ASCII.
Find the corresponding characters of the ASCII values
Finally concatenate them to get the original text.

Figure 13. Proposed Decryption Flowchart

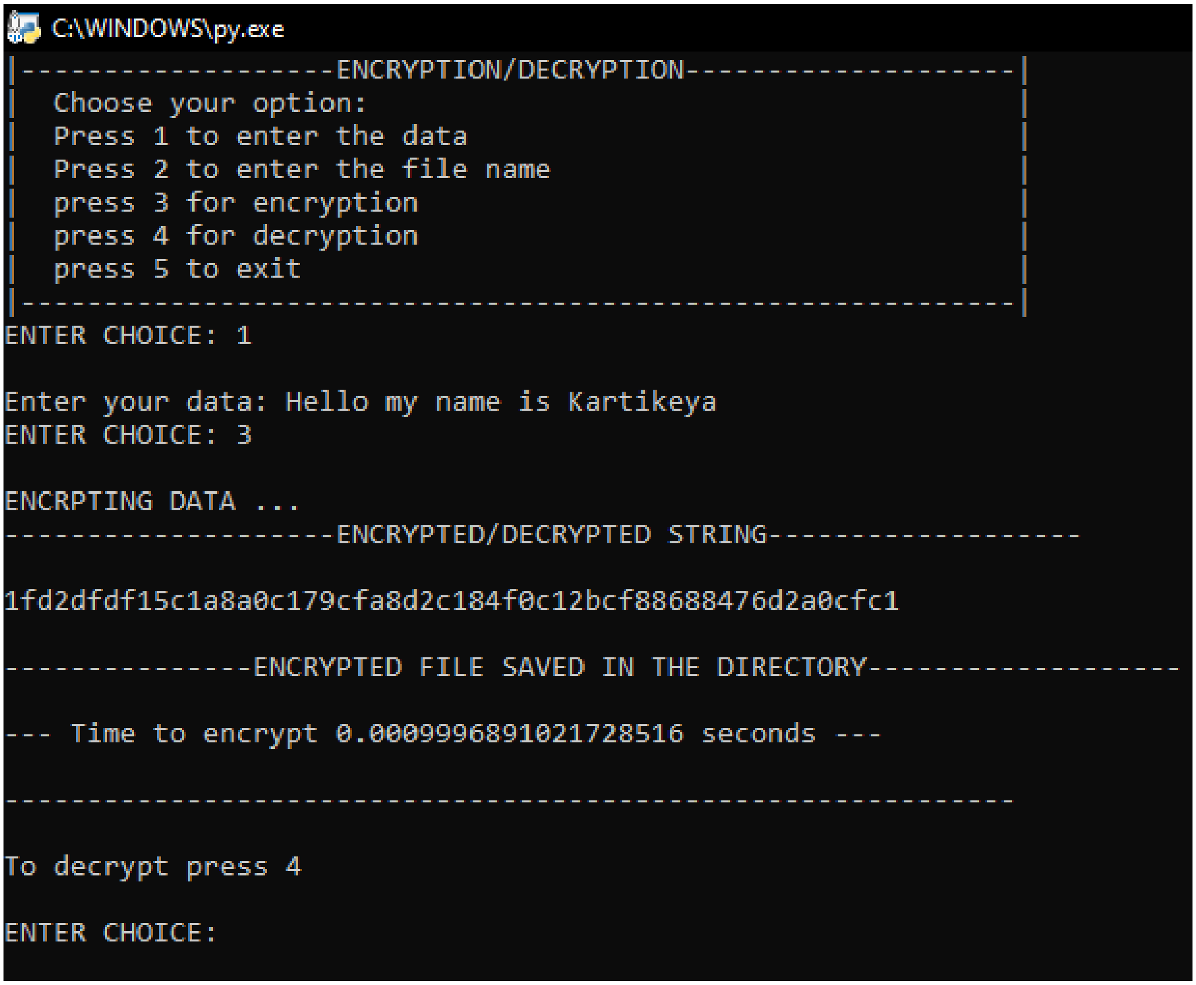
Performance Analysis

For performance analysis, the proposed algorithm with the symmetric key is coded in Python 3.9.6.

Processor used: Intel(R) Core (TM) i5-8250U CPU @ 1.60GHz   1.80 GHz

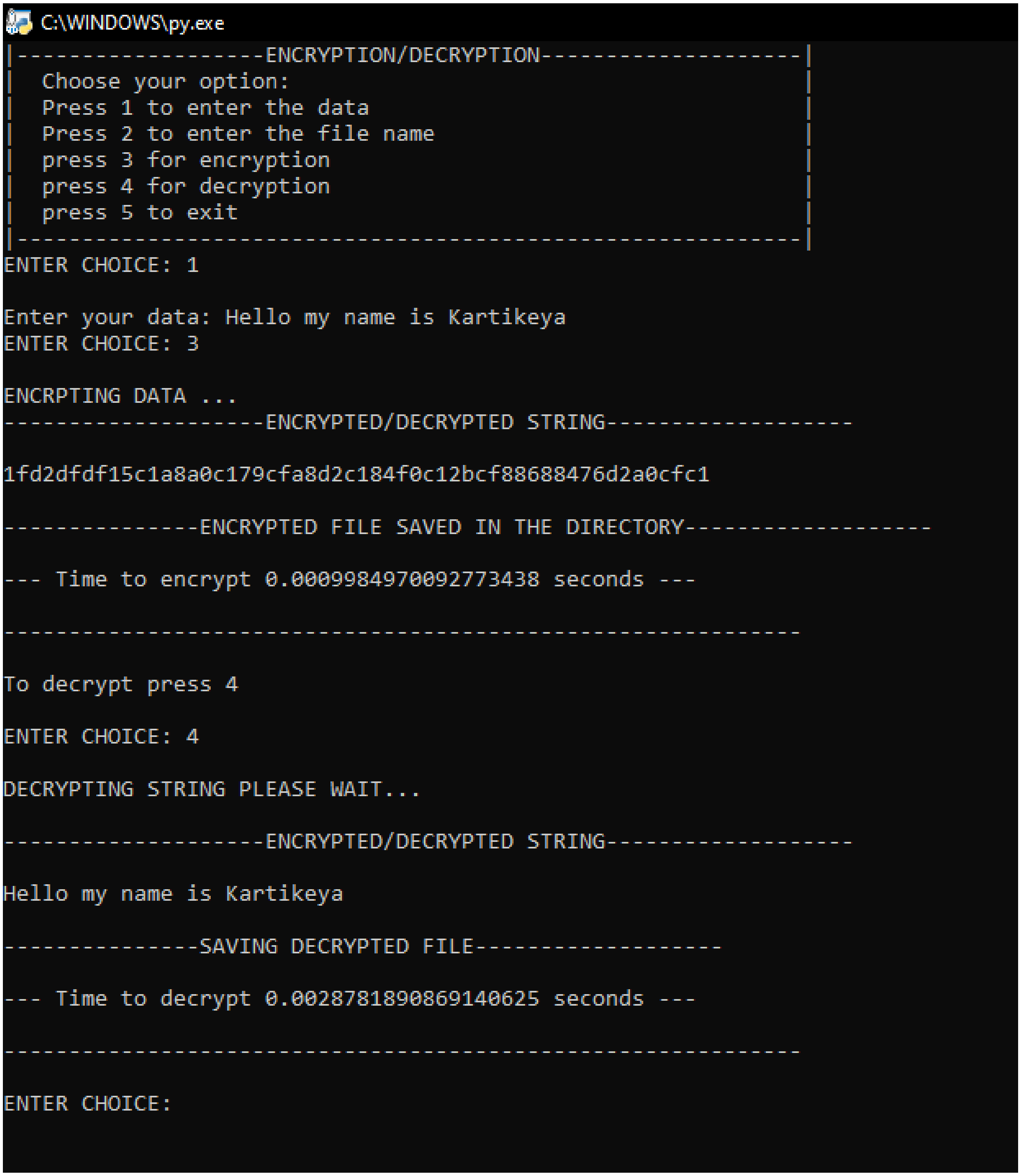
RAM: 8GB

OS: 64-bit operating system Windows 10



```
|-------------------ENCRYPTION/DECRYPTION--------------------|
|  Choose your option:                                       |
|  Press 1 to enter the data                                 |
|  Press 2 to enter the file name                            |
|  press 3 for encryption                                    |
|  press 4 for decryption                                    |
|  press 5 to exit                                           |
|------------------------------------------------------------|
ENTER CHOICE: 1

Enter your data: Hello my name is Kartikeya
ENTER CHOICE: 3

ENCRPTING DATA ...
--------------------ENCRYPTED/DECRYPTED STRING-------------------

1fd2dfdf15c1a8a0c179cfa8d2c184f0c12bcf88688476d2a0cfc1

---------------ENCRYPTED FILE SAVED IN THE DIRECTORY-------------------

--- Time to encrypt 0.0009996891021728516 seconds ---

-------------------------------------------------------------

To decrypt press 4

ENTER CHOICE:
```

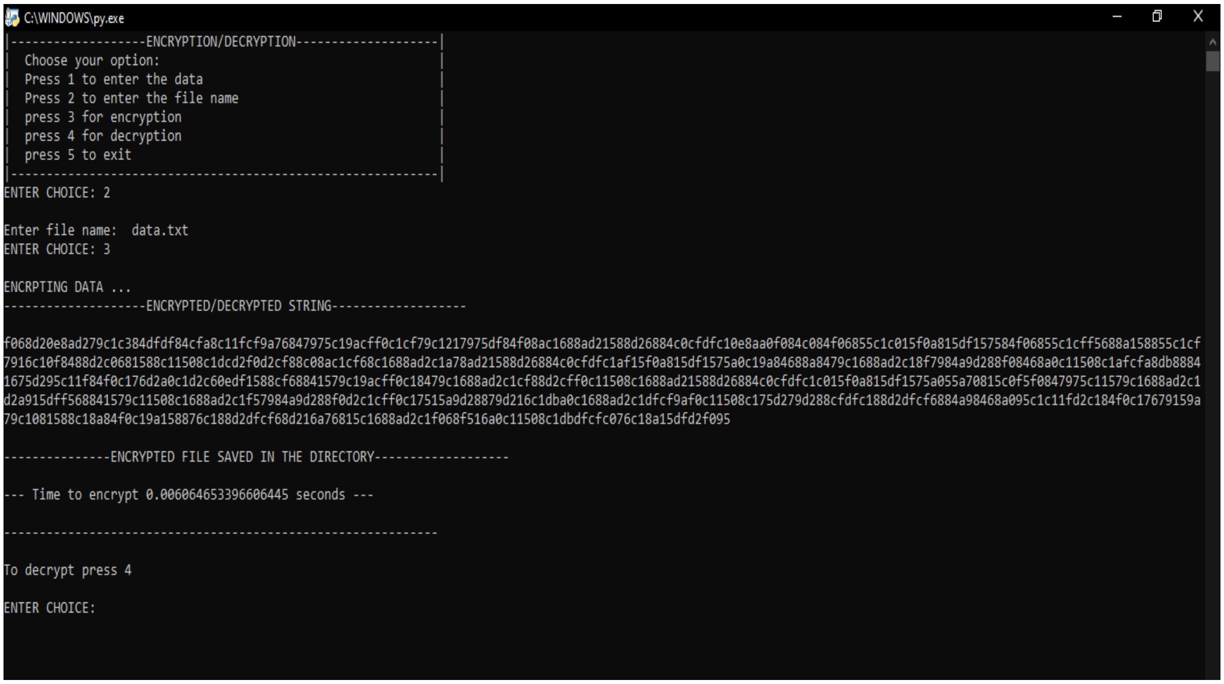
Figure 14. Encrypting text data



C:\WINDOWS\py.exe

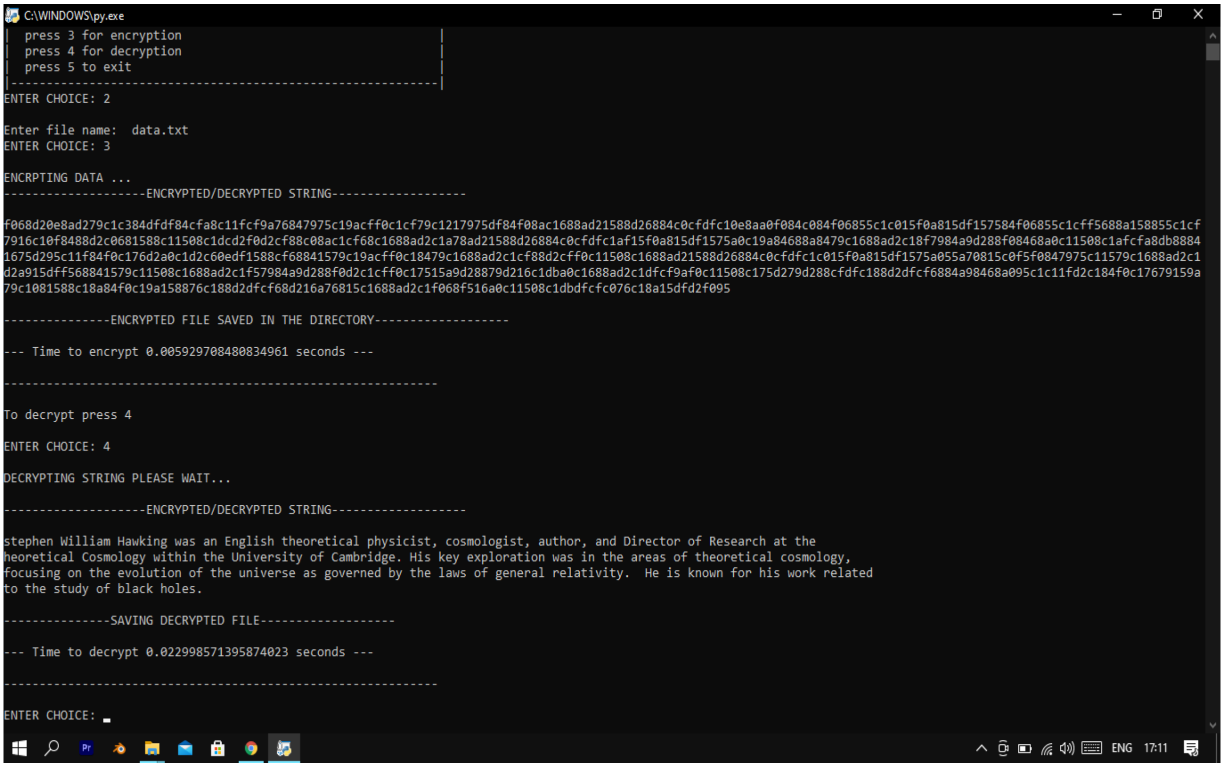
```
|-------------------ENCRYPTION/DECRYPTION--------------------|
|  Choose your option:                                       |
|  Press 1 to enter the data                                 |
|  Press 2 to enter the file name                            |
|  press 3 for encryption                                    |
|  press 4 for decryption                                    |
|  press 5 to exit                                           |
|------------------------------------------------------------|
ENTER CHOICE: 1

Enter your data: Hello my name is Kartikeya
ENTER CHOICE: 3

ENCRPTING DATA ...
--------------------ENCRYPTED/DECRYPTED STRING-------------------

1fd2dfdf15c1a8a0c179cfa8d2c184f0c12bcf88688476d2a0cfc1

---------------ENCRYPTED FILE SAVED IN THE DIRECTORY-------------------

--- Time to encrypt 0.0009984970092773438 seconds ---

-------------------------------------------------------------

To decrypt press 4

ENTER CHOICE: 4

DECRYPTING STRING PLEASE WAIT...

--------------------ENCRYPTED/DECRYPTED STRING-------------------

Hello my name is Kartikeya

---------------SAVING DECRYPTED FILE-------------------

--- Time to decrypt 0.0028781890869140625 seconds ---

-------------------------------------------------------------

ENTER CHOICE:
```

Figure 15. Decrypting text data



```
|-------------------ENCRYPTION/DECRYPTION-------------------|
|  Choose your option:                                      |
|  Press 1 to enter the data                                |
|  Press 2 to enter the file name                           |
|  press 3 for encryption                                   |
|  press 4 for decryption                                   |
|  press 5 to exit                                          |
|-----------------------------------------------------------|
ENTER CHOICE: 2

Enter file name:  data.txt
ENTER CHOICE: 3

ENCRPTING DATA ...
--------------------ENCRYPTED/DECRYPTED STRING------------------

f068d20e8ad279c1c384dfdf84cfa8c11fcf9a76847975c19acff0c1cf79c1217975df84f08ac1688ad21588d26884c0cfdfc10e8aa0f084c084f06855c1c015f0a815df157584f06855c1cff5688a158855c1cf
7916c10f8488d2c0681588c11508c1dcd2f0d2cf88c08ac1cf68c1688ad2c1a78ad21588d26884c0cfdfc1af15f0a815df1575a0c19a84688a8479c1688ad2c18f7984a9d288f08468a0c11508c1afcfa8db8884
1675d295c11f84f0c176d2a0c1d2c60edf1588cf68841579c19acff0c18479c1688ad2c1cf88d2cff0c11508c1688ad21588d26884c0cfdfc1c015f0a815df1575a055a70815c0f5f0847975c11579c1688ad2c1
d2a915dff568841579c11508c1688ad2c1f57984a9d288f0d2c1cff0c17515a9d28879d216c1dba0c1688ad2c1dfcf9af0c11508c175d279d288cfdfc188d2dfcf6884a98468a095c1c11fd2c184f0c17679159a
79c1081588c18a84f0c19a158876c188d2dfcf68d216a76815c1688ad2c1f068f516a0c11508c1dbdfcfc076c18a15dfd2f095

---------------ENCRYPTED FILE SAVED IN THE DIRECTORY------------------

--- Time to encrypt 0.006064653396606445 seconds ---

-------------------------------------------------------------

To decrypt press 4

ENTER CHOICE:
```

Figure 16. Encrypting file



```
|  press 3 for encryption                                   |
|  press 4 for decryption                                   |
|  press 5 to exit                                          |
|-----------------------------------------------------------|
ENTER CHOICE: 2

Enter file name:  data.txt
ENTER CHOICE: 3

ENCRPTING DATA ...
--------------------ENCRYPTED/DECRYPTED STRING------------------

f068d20e8ad279c1c384dfdf84cfa8c11fcf9a76847975c19acff0c1cf79c1217975df84f08ac1688ad21588d26884c0cfdfc10e8aa0f084c084f06855c1c015f0a815df157584f06855c1cff5688a158855c1cf
7916c10f8488d2c0681588c11508c1dcd2f0d2cf88c08ac1cf68c1688ad2c1a78ad21588d26884c0cfdfc1af15f0a815df1575a0c19a84688a8479c1688ad2c18f7984a9d288f08468a0c11508c1afcfa8db8884
1675d295c11f84f0c176d2a0c1d2c60edf1588cf68841579c19acff0c18479c1688ad2c1cf88d2cff0c11508c1688ad21588d26884c0cfdfc1c015f0a815df1575a055a70815c0f5f0847975c11579c1688ad2c1
d2a915dff568841579c11508c1688ad2c1f57984a9d288f0d2c1cff0c17515a9d28879d216c1dba0c1688ad2c1dfcf9af0c11508c175d279d288cfdfc188d2dfcf6884a98468a095c1c11fd2c184f0c17679159a
79c1081588c18a84f0c19a158876c188d2dfcf68d216a76815c1688ad2c1f068f516a0c11508c1dbdfcfc076c18a15dfd2f095

---------------ENCRYPTED FILE SAVED IN THE DIRECTORY------------------

--- Time to encrypt 0.005929708480834961 seconds ---

-------------------------------------------------------------

To decrypt press 4

ENTER CHOICE: 4

DECRYPTING STRING PLEASE WAIT...

--------------------ENCRYPTED/DECRYPTED STRING------------------

stephen William Hawking was an English theoretical physicist, cosmologist, author, and Director of Research at the
heoretical Cosmology within the University of Cambridge. His key exploration was in the areas of theoretical cosmology,
focusing on the evolution of the universe as governed by the laws of general relativity.  He is known for his work related
to the study of black holes.

---------------SAVING DECRYPTED FILE------------------

--- Time to decrypt 0.022998571395874023 seconds ---

-------------------------------------------------------------

ENTER CHOICE: _
```

Figure 17. decrypting file

CHAPTER 4

CONCLUSION

Cryptography is processed to convert a plain text into non-understandable unreadable text and encryption helps us to achieve this, hence encryption algorithms play very important role in communication security. Before sending any confidential data, it must be encrypted so that it remains protected from the attackers or from the unauthorized access over the network. This Paper has a new encryption technique which is very simple in nature with a less execution time.

THE FUTURE OF ENCRYPTION

Now that we have a clearer understanding of what data encryption means for you, let’s get into some of the development in cryptography we’re most excited for in the future.

1. QUANTUM CRYPTOGRAPHY

Quantum cryptography uses photons of sunshine and therefore the principles of physics to physically move data between a sender and recipient. Because information is transmitted using light, it can't be intercepted, copied, or cloned. With the assistance of quantum physics, they are often sent in order that only the intended recipient can read it without the info being altered. No additional level of encryption would be necessary because the info would be useless if it were to be intercepted by anyone aside from the recipient.

2. HONEY ENCRYPTION

Honey encryption is a bit of a misnomer in that it doesn’t rely on old-style encryption approaches. Instead, it deters cybercriminals by making them think they’ve increased access to your network or data when they have only obtained false or irrelevant data.

3. FACIAL RECOGNITION ENCRYPTION

We’re already begin to witness the foundations being laid for facial encryption. As facial recognition technology advances, we are expected to see facial encryption become a fundamental way of securing data and protecting access to confidential information.

4. HOMOMORPHIC ENCRYPTION

Traditional encryption approaches create vulnerability when you encrypt a message and again when you decrypt it even with a private key. Unfortunately, we must decode data to access and use it. Homomorphic encryptions try to find to tackle this problem by allowing you to use and access encrypted data without ever having to decrypt it in the first place.

REFERENCES:

1. Review of Secure File Storage on Cloud using hybrid Cryptography published by https://www.ijert.org/

2. RC5 Encryption Algorithm available at https://www.geeksforgeeks.org/rc5-encryption-algorithm/

3. The RC5 Encryption Algorithm* Ronald L. Rivest MIT Laboratory for Computer Science 545 Technology Square, Cambridge, Mass. 02139 rivest @theory. its. mit. Edu.(June 2017).

4. DES image reference – https://www.geeksforgeeks.org/data-encryption-standard-des-set-1/

5. Advanced encryption standard (AES) algorithm to encrypt and decrypt data by Ako Muhamad Abdullah. MSC Computer science – UK. Phd student in computer science.

6. Advanced encryption standard https://en.wikipedia.org/wiki/Advanced_Encryption_Standard

7. Tutorial point, topic – Cryptography Tutorial (Data encryption standard). https://www.tutorialspoint.com/cryptography/data_encryption_standard.htm
8. https://en.wikipedia.org/wiki/Data_Encryption_Standard

9. Comparative analysis of encryption algorithms for better utilization. (Anuj Kumar, Sapna Sniha, Rhaul Chaudhary. International Journal of computer application (0975-8887). Volume 71, 14th May 2013. https://en.wikipedia.org/wiki/Triple_DES

10. Idrizi, Florim, Dalipi, Fisnik & Rustemi, Ejup. “Analyzing the speed of combined cryptographic algorithms with secret and public key”. International Journal of

Engineering Research and Development, e-ISSN: 2278-067X, p-ISSN: 2278-800X, www.ijerd.com Volume 8, Issue 2 (August 2013), pp. 45

11. Jawahar Thakur, Nagesh Kumar, " DES, AES and Blowfish: Symmetric Key Cryptography Algorithms Simulation Based Performance Analysis," in International Journal of Emerging Technology and Advanced Engineering (ISSN 2250-2459, Volume 1, Issue 2, December 2011), pp.6-12.

12. S.Pavithra, Mrs. E. Ramadevi "study and performance analysis of cryptography algorithm" International Journal of Advanced Research in Computer Engineering & Technology Volume 1, Issue 5, July 2012 14, pp.82-86.

13. Nagesh Kumar, Jawahar Thakur, Arvind Kalia on "performance analysis of symmetric key cryptography algorithms DES, AES and blowfish " in AnInternational Journal of Engineering Sciences ISSN: 2229-6913 Issue Sept 2011, Vol. 4 ,pp.28-37.

14. Dr. Prerna Mahajan & Abhishek Sachdeva on "A Study of Encryption Algorithms AES, DES and RSA for Security" Global Journal of Computer Science and Technology Network, Web & Security Volume 13 Issue 15 Version 1.0 Year 2013.

15. Singh Narjeet, Raj Gaurav. "Security On Bcc Through Aes Encryption Technique". International Journal Of Engineering Science & Advanced Technology Volume-2, Issue-4, 813 - 819. pp. 817.

16. Pratap Chnadra Mandal on "Superiority of Blowfish Algorithm". International Journal of Advanced Research in Computer Science and Software Engineering Volume 2, Issue 9, September 2012.

17. T.Saravanan, Dr. S.Venkatesh Kumar, "A Review Paper on Cryptography-Science of Secure Communication," International Journal of Computer Science Trends and Technology (IJCST) – Volume 6 Issue 4, Jul-Aug 2018

18. A. M. Qadir and N. Varol, “A Review Paper on Cryptography,” 2019 7th International Symposium on Digital Forensics and Security (ISDFS) Barcelos, Portugal, 2019, pp. 1-6, DOI: 10.1109/ISDFS.2019.8757514.

19. E. Thambiraj,G. Ramesh,Dr. R. Umarani , “A survey on Various Most Common Encryption Technique “ International Journal of Advanced Research in Computer Science and Software Engineering , Volume 2, Issue 7, July 2012,pp226-233.

20. Tushar, Aniket Sharma, Ankit Mishra ,Department of Computer Science, Amity University Chhattisgarh “Cryptographic Algorithm For Enhancing Data Security: A Theoretical Approach” International Journal of Engineering Research & Technology (IJERT) ISSN: 2278-0181 IJERTV10IS030158 Vol. 10 Issue 03, March-2021.

21. Ekta Agrawal, Dr. Parashu Ram Pal, “A Secure and Fast Approach for Encryption and Decryption of Message Communication,” International Journal of Engineering Science and Computing. Volume 7 Issue No. 5.