TEXT ENCRYPTION USING VARIOUS ALGORITHMS

A Project Report

Submitted in the partial fulfillment of the requirements for

the award of the degree of

Bachelor of Technology

in

Department of Computer Science and Engineering

By

Nihal Agarwal	2010030413
Jaideep Sharma	2010030374
T Venkata Sai Sathvik	2010030361
Shaik Abdul Shaan	2010030153

under the supervision of

Dr. Gayathri Edamadaka
Associate Professor

Department of Computer Science and Engineering

K L University Hyderabad,

Aziz Nagar, Moinabad Road, Hyderabad – 500 075, Telangana, India.

March 2022

Declaration

The Project Report entitled "**Text Encryption Using Various Algorithms**" is a record of bonafide work of **Nihal Agarwal (2010030413), Jaideep Sharma (2010030374), T Venkata Sai Sathvik (2010030361), Shaik Abdul Shaan (2010030153)**, submitted in partial fulfillment for the award of B.Tech in the Department of Computer Science and Engineering to the K L University, Hyderabad. The results embodied in this report have not been copied from any other Departments/University/Institute.

Signature Of Students

Certificate

This is to certify that the Project Report entitled "**Text Encryption Using Various Algorithms**" is being submitted by Nihal Agarwal (2010030413), Jaideep Sharma (2010030374), T Venkata Sai Sathvik (2010030361), Shaik Abdul Shaan (2010030153) , submitted in partial fulfillment for the award of B.Tech in Computer Science and Engineering to the K L University, Hyderabad is a record of bonafide work carried out under our guidance and supervision.

The results embodied in this report have not been copied from any other departments/ University/Institute.

Signature of the Supervisor

Dr. Gayathri Edamadaka

Associate Professor

Signature of the HOD

Signature of the External Examine

ACKNOWLEDGEMENT

First, I would like to thank our beloved Founder and chairman, of Koneru Lakshmaiah University for giving us this opportunity to complete our project within the University in the guidance of our faculty.

I am highly indebted to our Principal, **Dr. L Koteswara Rao** who has been constantly pushing us and providing us opportunities for all the curricular activities undertaken by us.

I would like to thank my Head of department **Dr. Chiranjeevi Manike** for his exemplary guidance, monitoring and constant support through the course of the project. We thank **Dr. Gayathri Edamadaka, Associate Professor** of our department who has supported throughout this project holding a position of supervisor.

I am extremely grateful to all the teaching and non-teaching staff of our department without whom we won’t have made this project a reality. We would like to extend our sincere thanks to our parents who supported us making this project a grand success.

ABSTRACT

Now days, Data security is very challenging issue that touches many areas including computers and communication. Recently, we came across many attacks on cyber security that have played with the confidentiality of the users. These attacks just broke all the security algorithms and affected the confidentiality, authentication, integrity, availability, and identification of user data. Cryptography is one such way to make sure that confidentiality, authentication, integrity, availability and identification of user data can be maintained as well as security and privacy of data can be provided to the user. Encryption is the process of converting normal data or plaintext to something incomprehensible or cipher-text by applying mathematical transformations or formulae. These mathematical transformations or formulae used for encryption processes are called algorithms.

Keywords: - Security, confidentiality, cryptography, authentication, encryption, cipher.

TABLE OF CONTENTS

List Of Figures

1. INTRODUCTION

In the age of big data, one of the most important and valuable assets in the world is the data. Almost every business in every field makes most of their decisions after computing and analyzing the available data. Therefore, ensuring the safety and reliability of the data during its transmission and management has become a pressing and important issue. This also includes ensuring the authenticity of data sources and preventing malicious alteration of original data. Blockchain as one of the most innovative technologies in the present times is known for providing the integrity, authenticity, and confidentiality of the data in a decentralized way. Achieving all these security features successfully in a blockchain requires smaller transaction size and higher transaction efficiency, which in turn is strictly dependent on the encryption algorithms used for blockchain's implementation. Several encryption algorithms are used to implement blockchain, and they all have their pros and cons when it comes to their overall performance.

1.1 PROBLEM STATEMENT

Encryption is a way for data-messages or files to be made unreadable and ensuring that only authorized person can access the data. Encryption means securing digital data by converting it from readable format into an encoded format. Encrypted data can be read after it's been decrypted. Encryption is a building block of data security. It is most important way to ensure a computer system's information can't be stolen or read by someone unauthorized. Data encryption security is used by individual users and companies and organizations, to protect the information sent between server and browser. The information can be anything, it may be messages, personal data, payment data. An encryption software, also known as encryption algorithm, is used to develop an encryption scheme that theoretically can only be broken with large computational power.

We need to develop the AES and RSA algorithm by which we can encrypt and decrypt the text.

2. LITERATURE SURVEY

AUTHOR	TITLE	PUBLISHED SOURCE	METHODS	FINDINGS
Zahra Ch. Oleiwi, Wasan A. Alawsi, Wisam.Ch. Alisawi, Ali S. Alfoudi, Liwa H. Alfarhani	Overview and Performance Analysis of Encryption Algorithms	Overview and analysis of Encryption Algorithms	Analysis of different encryption algorithms for encryption.	AES algorithm is the effective among symmetric encryption algorithms and RSA algorithm is the best in asymmetric algorithms
Dr. Prerna Mahajan Abhishek Sachdeva	A Study of Encryption Algorithms AES, DES and RSA for Security	Study Of AES, DES,RSA	Using RSA, DES, AES algorithms for encryption and analyzing the time taken for encryption.	RSA algorithm takes longer time for the process AES, DES show very minor difference between time taken.
Roshni Padate, Aamna Patel	Encryption and decryption of text using AES Algorithm	Encryption using AES	Using AES algorithm for encrypting and decrypting using AES key expansion	We have to generate a key which will be known to sender and receiver, so that they can encrypt the file and decrypt the text and can only be seen after authorizing it with proper key

3. HARDWARE AND SOFTWARE REQUIREMENTS

3.1 **HARDWARE REQUIREMENTS**

1. 1. Intel(R) Core(TM) i5-10300H CPU @ 2.50GHz   2.50 GHz
2. 8.00 GB RAM or higher.
3. 64 Bit operating system or higher.
4. 1 TB Hard free drive space.

3.2 **SOFTWARE REQUIREMENTS**

1. Operating System: Windows 10/11
2. Web Browser: Brave/ Google Chrome
3. Python programming language
4. PyCharm or python IDLE for python programming.

4. FUNCTIONAL AND NON-FUNCTIONAL REQUIREMENTS

4.1 WORKING OF ENCRYPTION

4.1.1 ALGORITHM/CIPHERS

Ciphers, also called **encryption algorithms**, are systems for encrypting and decrypting data. A cipher converts the original message, called plaintext, into ciphertext using a key to determine how it is done. These are the rules of encryption process. The key length, functionality, and features of encryption system in use determine the effectiveness of encryption.

4.1.2 KEY

An **encryption key** is a randomized string of bits which is used to encrypt or decrypt data. Each key is unique, and longer keys are hard to break.

There are two types of keys, that are symmetric and asymmetric

1. **Symmetric keys** system, in this system, accessors of the data are having the same keys.
2. **Asymmetric keys** system, in this system, one key is secret which is private key, and other is public key which will be available with anyone accessing in public.

4.2 WORKING OF DECRYPTION

Decryption is the process of converting unreadable ciphertext to readable information. In decryption, the system extracts and converts the encrypted form of data to text or images based on the information which is understandable by humans as well as system. Decryption is performed using keys which are decided by the sender and receiver which will be available with both the organizations and can be used for transmitting data from system to server in encrypted form and then from server to system in decrypted form.

4.3 PROCESS OF ENCRYPTION AND DECRYPTION

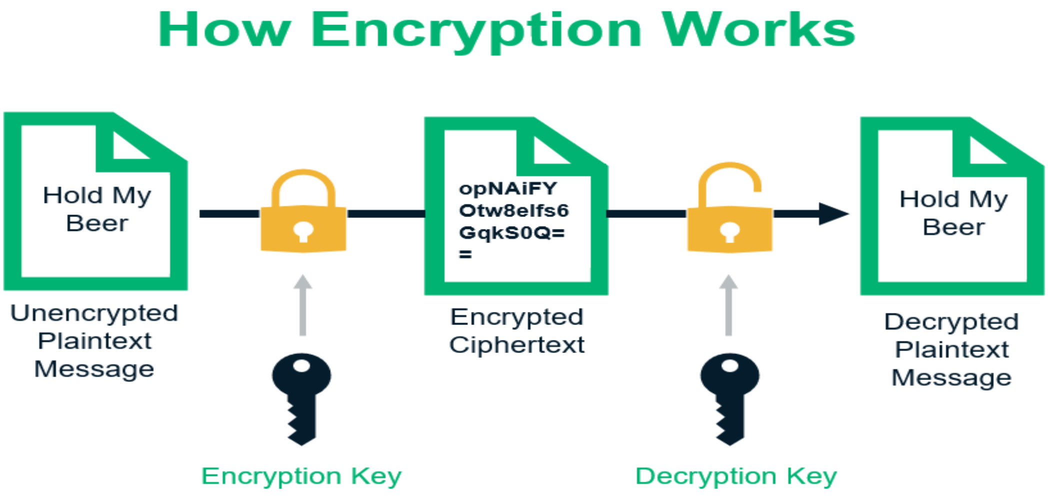


Fig. 4.1 Process of Encryption and Decryption

4.4 PROJECT AREA

The proposed text encryption is done using two algorithms, AES and RSA algorithm. AES is a symmetric key algorithm and RSA is an asymmetric key algorithm.

5. PROPOSED SYSTEM

5.1 AES ALGORITHM

AES stands for Advanced Encryption Standard and is trusted algorithm as the standard by the U.S. government and also many organizations. It is highly efficient in 128-bit form, and it can also use 192 and 256 bits for string encryption purposes. It is considered impervious to all the attacks, except the brute-force, which attempts to decipher messages using possible combinations such as 128,192,256-bit cipher. It is in widely use around the world. It falls into a class of encryption methods called "symmetric" encryption. That is, the same secret (an encryption key) is used to encrypt the data, and also used to decrypt the data.

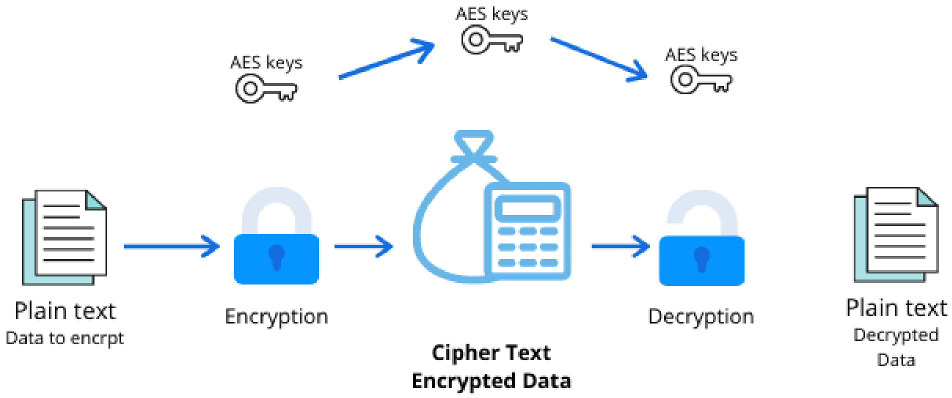


Fig. 5.1 Mechanism of AES Encryption and Decryption

Steps in each round

1. **Substitution of the bytes: -** In the first step, the bytes of the block text are substituted based on rules dictated by predefined S-boxes (short for substitution boxes).

2. **Shifting the rows: -** Next comes the permutation step. In this step, all rows except the first are shifted by one, as shown below.

3. **Mixing the columns: -** In the third step, the Hill cipher is used to jumble up the message more by mixing the block's columns.

4. **Adding the round key: -** In the final step, the message is XORed with the respective round key.

5.2 **RSA ALGORITHM**

RSA algorithm is an asymmetric cryptography algorithm; asymmetric algorithm tells us to use two keys instead of one public key for decryption of cipher text. The two keys used are public key and private key which are mathematically linked with each other. Public key is shared publicly between the sender and receiver and all other people who want to send a message to the particular receiver. Private key is a secret key used for decryption for the receiver and must not be shared with anyone. The RSA algorithm is named after scientists those who invented in the late 1980's, they are: Ron Rivest, Adhi Shamir and Leonard Adleman.

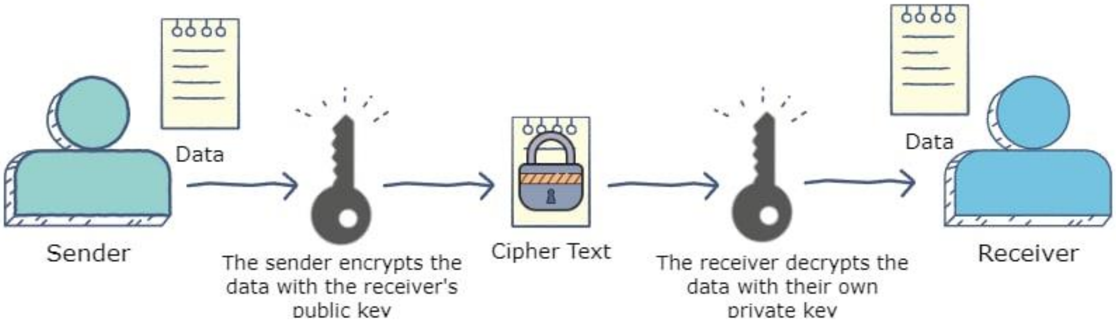

Data
Sender
The sender encrypts the data with the receiver's public key
Cipher Text
The receiver decrypts the data with their own private key
Data
Receiver

Fig. 5.2 Mechanism of RSA Encryption and Decryption

Generating The Keys

1. select two large prime numbers, x and y. The prime numbers need to be large so that they will be difficult for someone to figure it out.
2. calculate $n = x X y$.
3. calculate the totient function: $\phi(n)=(x-1)\ (y-1)$
4. select an integer call it e; as it a co-prime between

   $1 < e < \phi(n)$ and $GCD(\phi(n), e) = 1$
5. Hence, the pair of numbers: (n,e) = public key
6. Calculate d such that $e.d = 1.mod\phi(n)$
7. Hence, the pair of numbers: (n, d) = private key

Encryption using the private and public key:

P = PLAIN TEXT   C = CIPHER TEXT

C = P^e mod n

Decryption using the private key:

P = C^d mod n

6. IMPLEMENTATION

6.1 **Cryptography in Python**

Cryptography is a package which provides cryptographic recipes and primitives to encryption and decryption algorithms. It includes both high and low level interfaces and recipes to common cryptographic algorithms such as symmetric ciphers and asymmetric ciphers for message digests and key generations.

6.2 **Cipher Text in Python**

Ciphertext is encrypted text transformed from plaintext using an encryption algorithm. Ciphertext can't be read until it has been converted into plaintext (decrypted) with a key. The decryption cipher is an algorithm that transforms the ciphertext back into plaintext.

Examples of different cipher texts

1. Caesar Cipher
2. Mono Alphabetic Cipher
3. Poly Alphabetic Cipher
4. Polygram Substitution Cipher
5. Vigenère Cipher

Transposition techniques which convert the plaintext into Cipher Text:

1. Rail fencing
2. Simple Columnar Transposition
3. Vernam Cipher

6.3 **Padding**

Padding in cryptography is any of a number of distinct practices which all include adding data to the middle, end or beginning prior to an encryption process to set right with the number of bits are called as padding of the digits or alphabets. If the message contains the text data, the column for the message is padded with space characters by using correct characters set If the message contains binary characters, the column for this message is padded with NULL bytes We need to set the padding value to YES, as it is default set to NO.

6.4 **UTF-8**

The Unicode Consortium develops the Unicode Standard. Their goal is to replace the existing character sets with its standard Unicode Transformation Format (UTF).

6.5 **CODE EXPLANATION – AES ENCRYPTION**

Define the pad () method, which changes the plaintext to padded text.

the while loop prompts the user to choose the process the person wants to execute that is to encrypt or decrypt or stop.

```
if the user chooses option-1
    the system asks to enter the text to be encrypted.
            it is converted into padded text then the system encodes it under UTF-8 encoding
    The system prompts the user to enter the key for encryption
            it is converted into padded text then the system encodes it under UTF-8 encoding
    a new cipher is created using the AES.new (key, AES.MODE_ECB) method and the cipher. encrypt () method encrypts the plain text.
    the message gets encrypted.

if the user chooses option-2
    the system prompts the user to enter the decryption key which is same as the encryption key
        if the key matches
           decrypt the message
        else
           alert the user about wrong decryption key.

if the user chooses option-3
    exit from the process
```

6.6 **AES – IMPLEMENTATION**

```
from Crypto.Cipher import AES

def pad(entry):
    padded=entry+(16-len(entry)%16) *'['
    return padded
print("1. Encrypt text\n2. Decrypt Text\n3. Exit")
k=""
l=""
cip=""
while True:
    t = int(input("Enter Your Option: "))
    if t==1:
        plain_text=input("Enter text to be encrypted: ")
        plain_text=pad(plain_text)
        plain_text=plain_text.encode('UTF-8')
        key=input("Enter key to encrypt the text: ")
        k=key
        key=pad(key)
        key=key.encode('UTF-8')
        l=key
        cipher=AES.new(key,AES.MODE_ECB)
        ciphertext=cipher.encrypt(plain_text)
        print("encrypted cipher form: ",ciphertext)
        cip=ciphertext
        print("**********************Encryption Done************************")
    elif t==2:
        key1=input("Enter key for decryption: ")
        if key1==k:
            cipher2=AES.new(l,AES.MODE_ECB)
            data=cipher.decrypt(cip)
            data=data.decode('UTF-8')
            unpad=data.find('[')
            data=data[:unpad]
            print("Decrypted Data: ",data)
            print("**********************Decryption
Done************************")
        else:
            print("Alert!! Wrong Password, Try Again !!!!")
    else:
        print("**********************THANK YOU************************")
        exit(0)
```

6.7 **CODE EXPLANATION – RSA ENCRYPTION**

Define the public_key and private_key using rsa.newkeys(1024) method.

the while loop prompts the user to choose the process the person wants to execute that is to encrypt or decrypt or stop.

if the user choose option-1
the system asks to enter the text to be encrypted.
it is converted into bytes under UTF-8 encoding
a new cipher is created using the rsa.encrypt(byte_text, public_key) method encrypts the byte text.
the message gets encrypted.

if the user choose option-2
the rsa.decrypt(encrypted_text, private_key) decrypts the message
prints the decoded message.

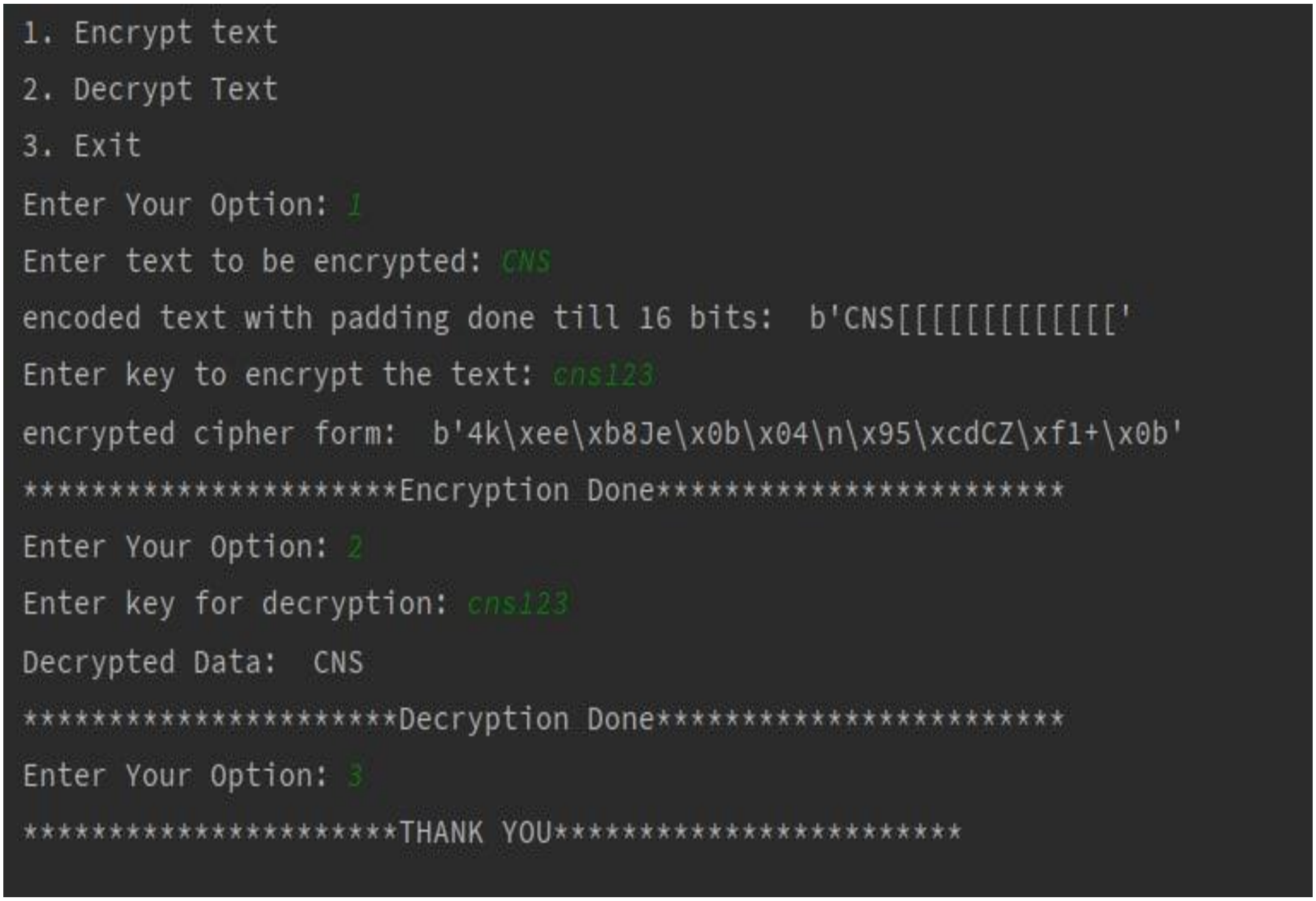
if the user choose option-3
exit from the process

6.8 **RSA – IMPLEMENTATION**

```
import rsa

public_key, private_key = rsa.newkeys(1024)
em=""
print("1. Encrypt text\n2. Decrypt Text\n3. Exit")
while True:
    t=int(input("Enter your option: "))
    if t==1:
        message = input("Enter the message: ")
        res = bytes(message,'utf-8')
        # print(res)
        eme=rsa.encrypt(res,public_key)
        # print(eme)
        em=eme
        print("**********************Encryption Done************************")
    elif t==2:
        dm=rsa.decrypt(em,private_key)
        print(dm.decode())
        print("**********************Decryption Done************************")
    else:
        print("**********************THANK YOU************************")
        exit(1)
        exit(1)
```

7. RESULTS DISCUSSION



```
1. Encrypt text
2. Decrypt Text
3. Exit
Enter Your Option: 1
Enter text to be encrypted: CNS
encoded text with padding done till 16 bits:  b'CNS[[[[[[[[[[[[['
Enter key to encrypt the text: cns123
encrypted cipher form:  b'4k\xee\xb8Je\x0b\x04\n\x95\xcdCZ\xf1+\x0b'
**********************Encryption Done************************
Enter Your Option: 2
Enter key for decryption: cns123
Decrypted Data:  CNS
**********************Decryption Done************************
Enter Your Option: 3
**********************THANK YOU************************
```

Fig 7.1. AES encryption Output

The console gives options for the process

1. Encrypt Text, 2. Decrypt Text, 3. Exit

As we choose the option-1

It asks us for text to be encrypted here- cns

Then asks key for encryption here- cns123

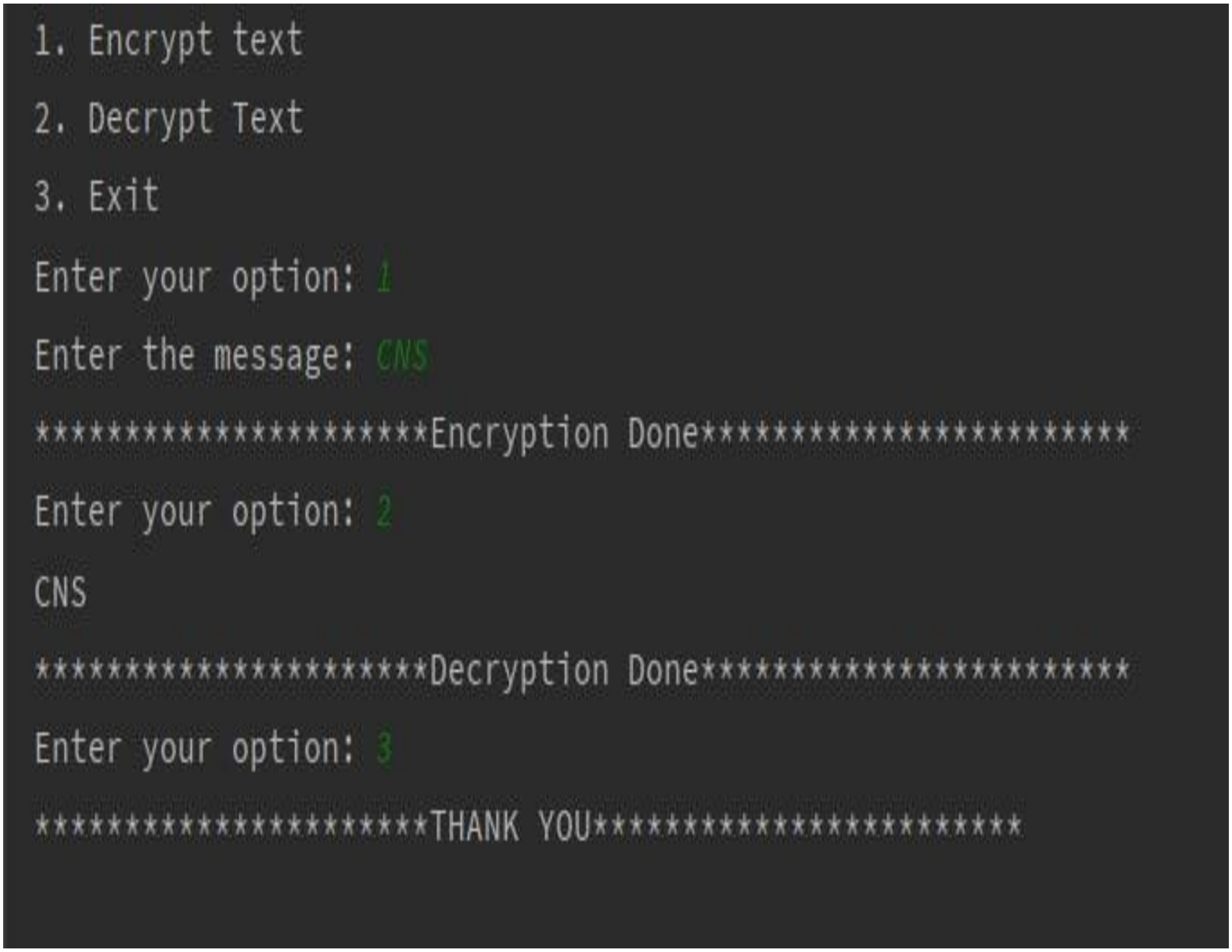
Now as we choose option-2

It prompts for key for decryption here- cns123

Displays the decrypted text

Now as our selected option is 3

It exits



```
1. Encrypt text

2. Decrypt Text

3. Exit

Enter your option: 1

Enter the message: CNS

**********************Encryption Done************************

Enter your option: 2

CNS

**********************Decryption Done************************

Enter your option: 3

**********************THANK YOU************************
```

Fig 7.2. RSA encryption Output

The console gives options for the process

1. Encrypt Text, 2. Decrypt Text, 3. Exit

As we choose the option-1

It asks us for text to be encrypted here- cns

Now as we choose option-2

Displays the decrypted text

Now as our selected option is 3

It exits

8. CONCLUSION AND FUTURE WORK

8.1 **CONCLUSION**

In the proposed implementation, we are doing the encryption and decryption in AES and RSA algorithms, it is done using while loop for 3 options such as, 1. Encrypt, 2. Decrypt, 3. Exit
For each of the loop it executes the loop based on the requirement. The proper plain text conversion to cipher text then using the key to encrypt and conversion of cipher text to plain text using the key is decryption.

8.2 **FUTURE WORK**

In future enhancements, the time complexity of RSA algorithm can be decreased by decreasing the time for key generation.

10. REFERENCES

1. "Biclique Cryptanalysis of the Full AES" (PDF). Archived from the original (PDF) on March 6, 2016. Retrieved May 1, 2019.
2. Alex Biryukov and Dmitry Khovratovich, Related-key Cryptanalysis of the Full AES-192 and AES-256, "Archived copy". Table 1. Archived from the original on 2009-09-28. Retrieved 2010-02-16.
3. Daemen, Joan; Rijmen, Vincent (March 9, 2003). "AES Proposal: Rijndael" (PDF). National Institute of Standards and Technology. p. 1. Archived (PDF) from the original on 5 March 2013. Retrieved 21 February 2013.
4. "Announcing the ADVANCED ENCRYPTION STANDARD (AES)" (PDF). Federal Information Processing Standards Publication 197. United States National Institute of Standards and Technology (NIST). November 26, 2001. Archived (PDF) from the original on March 12, 2017. Retrieved October 2, 2012.
5. Rivest, Ronald. "The Early Days of RSA – History and Lessons" (PDF).
6. Calderbank, Michael (2007-08-20). "The RSA Cryptosystem: History, Algorithm, Primes" (PDF).
7. Robinson, Sara (June 2003). "Still Guarding Secrets after Years of Attacks, RSA Earns Accolades for its Founders" (PDF). SIAM News. **36** (5).
8. Cocks, C. C. (20 November 1973). "A Note on Non-Secret Encryption" (PDF). www.gchq.gov.uk. Retrieved 2017-05-30.
9. Jim Sauerberg. "From Private to Public Key Ciphers in Three Easy Steps".