International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 06 | June - 2023 SJIF Rating: 8.176 ISSN: 2582-3930

Secure sensitive information by encoding and decoding messages

V.Jyoshita Reddy
B.Tech
School of Engineering
Hyderabad,India
2111CS020193@mallareddyuniversity.ac.in

R. Jyoshna
B.Tech
School of Engineering
Hyderabad, India
2111CS020194@mallareddyuniversity.ac.in

B. Jyothi
B.Tech
School of Engineering
Hyderabad, India
2111CS020195@mallareddyuniversity.ac.in

Rithika . J. Poojari
B.Tech
School of Engineering
Hyderabad,India
2111CS020196@mallareddyuniversity.ac.in

N. Kartheek
B.Tech
School of Engineering
Hyderabad,India
2111CS020197@mallareddyuniversity.ac.in

B. Karthik Goud
B.Tech
School of Engineering
Hyderabad, India
2111CS020198@mallareddyuniversity.ac.in

Guide:N.V.P.R.Rajeswari
Professor
School of Engineering,
Mallareddy University
vprrejeswari@mallareddyuniversity.ac.in

Abstract**:** *The project aim to develop a secure sensitive information by encoding messages. Message encoding and decoding is the process to first convert the original text to the random and meaningless text called ciphertext. This process is called encoding. Decoding is the process to convert that ciphertext to the original text. This process is also called the Encryption, Decryption process. This objective of this project is to encode and decode messages using a common key. This project will be built using the Tkinter and base64 library .In this project, users have to enter the message to encode or decode. Users have to select the mode to choose the encoding and decoding process. The same key must be used to process the encoding and decoding for the same message. Base64 is a library that allows the user to encode and decode the string. The string to be encoded should be in byte form. A function to encode binary information to ASCII characters then decode those ASCII classic characters to binary data is provided by the base64 module of the standard GUI Python library, Tkinter. On the command prompt, we use the pip install command to install the library. In addition to base64, base 85 is also used to provide additional encoding. Base64 and Base84 are included in the newer versions of Python3, so be sure to check the version beforehand.*

I. INTRODUCTION

In this digital era, the need for security is increasing rapidly. Complying with this requirement, the encryption and decryption algorithms were devised. Encoding and decoding messages can be an effective way to secure sensitive information. There are many methods of encoding and decoding messages, some of which are more secure than others. Digital communication witnesses a noticeable and continuous development in many applications in the Internet. Hence, secure communication sessions must be provided. The security of data transmitted across a global network has turned into a key factor on the network performance measures. So, the confidentiality and the integrity of data are needed to prevent eavesdroppers from accessing and using transmitted data.

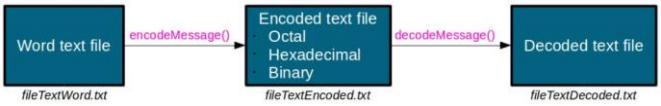


Fig: Basic Architecture

II. PROBLEM STATEMENT

The purpose of this project is to provide the correct data with security to the users. Encoding is the process that transforms the text or information to the unrecognizable form and decryption is the process to convert the encrypted message into original form. This objective of this project is to encode and decode messages using a common key. This project will be built using the Tkinter and base64 library. In this project, users have to enter the message to encode or decode. Users have to select the mode to choose the encoding and decoding process. The same key must be used to process the encoding and decoding for the same message.

III. LITERATURE REVIEW

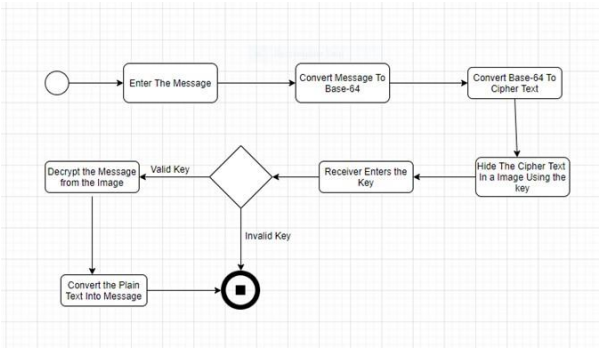
Several works in the past have attempted to discover which algorithm would work best for encryption and decryption. The work presented by Singh et al. [20] is a prime example of that as it compared between the different symmetric algorithms including the DES, 3DES, AES and the Blowfish algorithms. The work found that Blowfish was the best amongst the other methods despite their popularity in the field of encoding and decoding. Accordingly, it was found that the AES algorithm was not proficient enough in comparison to other algorithm, for it needs higher processing time. Similarly, the work presented by Cornwell [21] found that the Blowfish algorithm had the ability to support security for a relatively long time without any suspicious violations of the code. According to the

researcher, the Blowfish algorithm is superior in terms of security and efficiency. However, further research should be carried on in order to re-estimate the results discussed by the Cornwell research on Blowfish to provide more evidence on the results. In other study that was presented by Tamimi [22], two modes of performance were employed, namely the ECB and CBC. Those modes are used to compare the time it takes for each of them to be run and processed. According to the work and in what agrees with all of the previous studies afore mentioned, Blowfish has proven to be the best out of the compared algorithms in the work due to the lack of efficiency in time when it comes to the AES and the need to processing more data. Many authors and researchers have found in Blowfish an ideal method for encryption and decryption including Nadeem [23] that found the in the many advantages of Blowfish a mean to overcome the competition in other algorithms. Additionally, the work presented by Nadeem concluded that AES is far more developed than DES and 3DES. It was also found that DES is far better than 3DES where the latter requires thrice the time when it comes to processing information. In another work offered by Dhawan [24], it was found that AES carried out other algorithms in number demanding operations a second in varied user load and in the reacting time with multiple user load circumstances. Singh et al. [25] presented a work that ran a comparison between the most popular encoding algorithms. According to the work, the most popular algorithms were AES, DES, 3DES and Blowfish in the terms of security and energy consumption. The results of the comparison contrasted with the some of the previous studies and showed that AES is better than the basic form of the Blowfish algorithms. However, to make BA stronger against every type of attack, extra keys could be added to substitute the old XOR with a new operation. In the work presented by Agrawal et al. [26] after long research about DES, 3DES, AES, and Blowfish, they confirmed the superiority of the Blowfish algorithms, in terms of key size and security. Blowfish algorithm F function enhances supreme stage of security to encode the 64-bit plaintext database. Besides, Blowfish algorithm works quicker than the rest common in identical key encoding algorithms. Similarly, Seth et al. [27] compared three algorithms: DES, AES and RSA. They inferred that RSA requires the longest encoding time and higher memory than the other two algorithms; however, with minimal output byte in RSA algorithm. Meanwhile, they also found that DES utilizes minimum enciphering time while AES requires the smallest storage memory. Furthermore, encoding time in both AES algorithm and DES algorithm is almost the same. Mandal et al. [28] Figured out that the AES is distinctive over the other 3DES and DES in throughput and decoding time in their work. Apoorva et al. [29] concluded that blowfish in the best algorithm to be used in terms of security and time to process because it consumes little time in comparison to the rest. In the work presented by Abdul et al. [30], numerous algorithms were studied including: AES, DES, 3DES, RC2, BLOWFISH and RC6. The conclusion of the comparison ran at the work was that no dramatic difference in hexadecimal base encryption or base 64 ciphering. Also, Blowfish has proven to perform better than the rest when transforming the pocket size. In addition, the work showed that the performance of 3DES is mediocre when compared to DES algorithm. All in all, the big key size could provide considerable improvement in the battery and time passed. Thakur et al. [31] ran a comparison between DES, AES and Blowfish moderately where the outcome proved that Blowfish is the best and ideal algorithm out of the three when it is in the terms of performance. Marwah et al. [32] also compared three algorithms, namely: DES, 3DES and RSA. The result is that the privacy ensured by 3DES is better than that of DES and RSA. DES is economical in energy memory required as well as fast in encoding and decoding database time. DES is vulnerable though, in comparison to 3DES and RSA. The work of Alam et al. [33] has proved that 3DES requires more energy and processes less input than those of DES, this is because of its triple time feature. However, RC2 proved to be quicker due to smaller sizes of the throughput if contrasted to Blowfish. Blowfish input value is bigger than 3DES, DES, CAST-128, IDEA and RC2. Blowfish consumes the least power. Eventually, it turns out that Blowfish is the best, in terms of time, throughput and power. Saini [34] sums up that the superior algorithms are prominent for their popularity. An efficient cryptography achieves two parts of an equation, possibility and acceptability.

IV. REQUIRED TOOLS

- Visual Studio Code
- Tkinter
- Python
- Base64
- Libraries like Numpy, pandas.

V. METHODOLOGY



Cryptography is the science to encrypt and decrypt data that enables the user to store sensitive information or transmit it across insecure networks so that it can be read only by the intended recipient. There are two kinds of encoding. Those two types are the symmetric and asymmetric encoding algorithms. Several of those algorithms will be included herein such as: AES, DES, 3DES, E-DES, BLOW FISH, SEAL, RC2, RC4 and RC6 which all have to do with bilateral

algorithms. In contrast to RSA, ECC, EEE, DH, ELGAMAL ALGORITHM and DSA, which are relevant to unilateral algorithm. In this project we use vigenere cipher algorithm

After designing the working principle, the flow chart of the system is implemented where the code and the model is developed and tested. The flowchart of the complete system in shown in Fig 5.1.

Fig:5.1

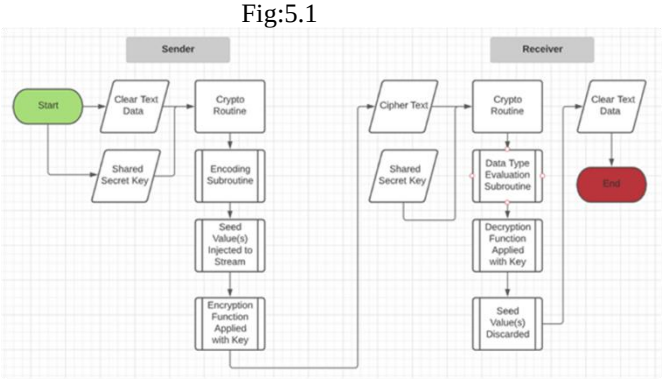


Fig:5.2

.

VI. EXPERIMENT RESULTS

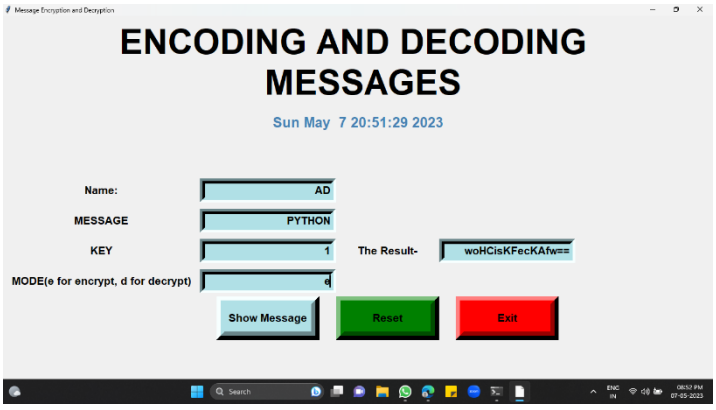


Fig: Encoding page

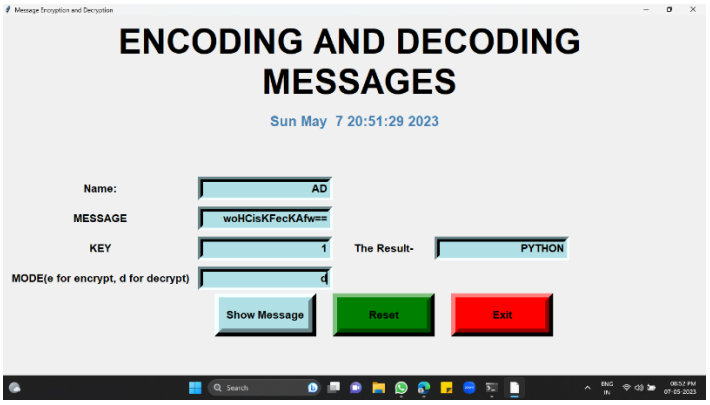


Fig: Decoding page

VII. ARCHITECTURE DIAGRAM FOR PROPOSED METHOD

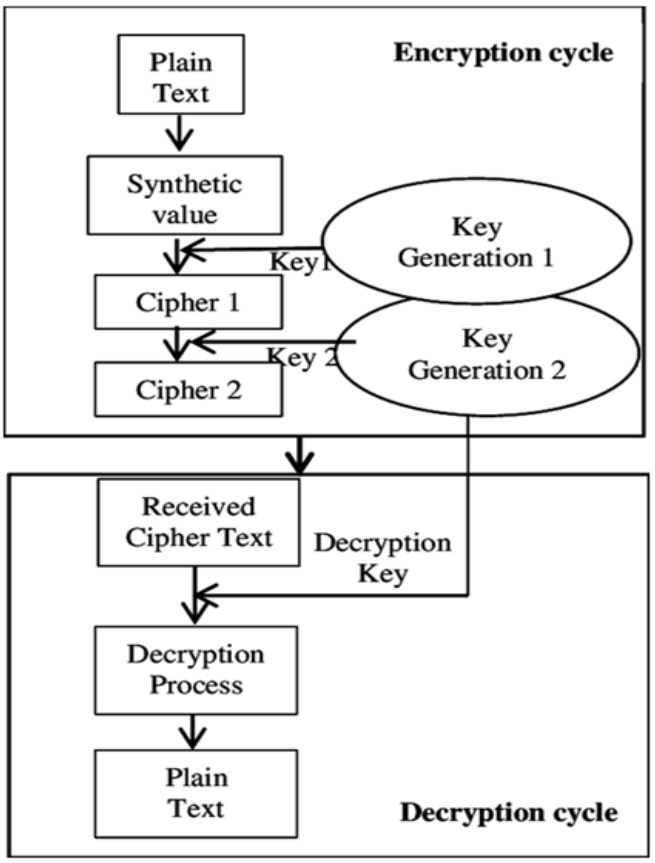


Fig: Architecture

VIII. CONCLUSION:

We have successfully developed Message encode – decode project in Python. We used the popular tkinter library for rendering graphics on a display window and base64 to encode & decode. We learned how to encode and decode the string, how to create button, widget, and pass the function to the button. In this way, we can encode our message and decode the encoded message in a secure way by using the key

IX. Future Enhancement:

The field of encoding and decoding messages is constantly evolving, and there are many potential future enhancements that could improve the security, efficiency, and reliability of these processes. Here are a few examples:

- Quantum Cryptography
- Homomorphic Encryption:

- Machine Learning-Based Encryption
- Blockchain-Based Encryption
- Post-Quantum Cryptography

These are just a few examples of the many potential enhancements to encoding and decoding messages that could emerge in the future. As technology continues to evolve, we can expect to see new and innovative approaches to encryption and communication security.

ACKNOWLEDGEMENT

I would like to express my heartfelt gratitude to my guide, Prof.N.V.P.R.Rajeshwari, and head of department, Dr . Thayyaba Khatoon, for their invaluable guidance and unwavering support throughout the development of this project. Their insightful feedback helped me to refine my ideas and develop a comprehensive understanding of the subject matter.

Their mentorship was instrumental in shaping my approach towards the project, and I am grateful for the knowledge and experience they shared with me. Without their encouragement and support, this project would not have been possible. Once again, I extend my sincere thanks to my guides and head of department for their unwavering support and guidance.

Our sincere thanks to all the teaching and non-teaching staff of Department of Computer Science and Engineering (AI&ML) for their support throughout our project work.

REFERENCES

[1] Theory Segment: ChatGPT

[2] Images: google chrome

[3] Architecture: https://www.researchgate.net/profile/Rehan_Shams/publication/299746582/figure/download/fig1/AS:347803638353921@1459934257398/DES-encryption-and-decryption.png