iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 9 **Issue:** VI **Month of publication:** June 2021

DOI: https://doi.org/10.22214/ijraset.2021.35572

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue VI June 2021- Available at www.ijraset.com

Image Steganography using 3 ways of Encryption for Highly Secured Data Transmission

Meera Tamboli[1], Mehul Latkar[2], Kalash Chauhan[3], Ankita Ghosh[4], Shruti Nikam[5]
[1, 2, 3, 4, 5]*Department of Information Technology and MCA, Vishwakarma Institute of Technology, Bibwewadi, Pune 411037, India.*

Abstract: ***Steganography exists from many years in a variety of forms and has been used in a variety of domains. There are various types of steganographic techniques that are used to hide data in various file formats. The proposed model uses three layers of security using three algorithms namely, LSB, DES and AES. The software allows the user to encrypt the secret message which gets protected by undergoing three layers of encryption. The output of the encryption process which is the steno image can be sent across through any medium along with the secret key to decrypt. The end user then uses a unique key to extract and decrypt the secret message from the image. There are many reasons to hide data but the simplest is the need to prevent unauthorized persons from accessing the secret information.***
Index Terms: Image steganography, aes, des, lsb, encryption, decryption.

I. INTRODUCTION

Steganography is one of the ways of privacy and security where we can hide messages in images so that their presence is invisible. Among the few benefits of using Steganography, is the transfer of personal information from the source to the destination. There are various approaches to the implementation of Steganography, Among the many strategies, Masking and Filtering, Algorithms and Transformations and the inclusion of LSB,DES and AES are some of the ways to achieve Steganography. Among these methods, LSB installation is the simplest and most commonly used method of embedding data to a cover file. DES is also a popular method that is significantly more secure than LSB where AES is more efficient and exponentially stronger than DES.

II. LITERATURE SURVEY

Image steganography practices exist from many years in different varieties and have been implemented in various domains. When it comes to sending confidential data from one end to another, the most important part is making sure the integrity of the data is taken care of.

This paper proposes the idea of having a triple layer of security which ensures the user that the data sent across is undergoing three levels of protected algorithms. LSB Algorithm has been modified several times over the years. Sneha Bansod and Gunjan Bhure [1] have discussed how the LSB algorithm works and what advantages and disadvantages this algorithm puts forward.

Indumathi Saikumar [2] has discussed that DES algorithm uses symmetric block cipher for encrypting and decrypting data. Encryption converts data into unintelligible language called cipher text. When the cipher text is decrypted, the original data which is the plaintext is returned.

AES algorithm stands for advanced encryption standard which is an encryption standard recommended by NIST. 128 bits data block is encrypted using AES and it uses key length of 128, 192 or 256 bits.

Vikas M et al. [3] have proposed a unique hybrid approach to text and image steganography using LSB and AES algorithms. In their proposed paper, AES is used for cryptography and LSB technique is used for Steganography. The suggested approach encrypts a text or image included within a cover image.

This paper proposes the idea of having 3 layers of encryption, namely, LSB, DES and AES. This ensures a highly secured data transmission. The classical LSB algorithm is one of the most fundamental and simplest methods of image steganography. LSB algorithm has high data hiding capacity.

This approach has the advantage that it is simple to understand, very easy to implement and results in the steno-images which contain embedded data in hidden format. The disadvantage of LSB is that adding extra noise like scaling, rotation can destroy the secret message and it is less robust than DES and AES.

AES algorithm is more secure than its predecessor DES. This algorithm is robust against hacking. It uses longer key lengths for decryption. Also, AES is faster in hardware as well as in software but the drawback of AES is that every block is encrypted in the same way and it is hard to implement with software.

DES algorithm follows a symmetric key method of data encryption where both the sender and receiver use the same keys for encryption and decryption of the message, which is a much faster method than asymmetric method and it is hard to break the large size key but if the key is lost then data cannot get at the receiving end.

Hardware implementation of DES is very fast but it was not designed for software that's why it runs relatively slower than other techniques.

Out of LSB, DES and AES, AES works the best mainly because of its speed and it is a most advanced technique. Because an asymmetric key algorithm requires less computational power than an asymmetric key algorithm. It is more efficient to run.

There are two ways used here, cryptography which is also referred to as information encryption, and steganography which is also called information hiding. These are the most significant techniques for data security. As discussed in his paper by Taha et al. [4] in cryptography, the secret information is changed in such a way that it cannot be readable to unauthorized people, but with steganography, the existence of the secret information is completely hidden from unauthorized persons.

Steganography and cryptography have been noted to be individually insufficient for complete information security; therefore, a more reliable and strong mechanism can be achieved by combining both techniques.

Subhash Panwar et al. [5] have discussed a modified version of the LSB algorithm and the AES algorithm. The method proposed confirms double-layer security with AES cryptography and the modified LSB technique. Experimentally it was found that the proposed approach is less prone to passive and active attacks.

III. ALGORITHMS USED

A. AES Algorithm

The AES algorithm (also known as the Rijndael algorithm) is an asymmetrical block cipher that converts plain text in 128-bit blocks to cipher text utilizing keys of 128, 192, and 256 bits.

All of AES's calculations are done in bytes rather than bits. As a result, AES considers a plaintext block's 128 bits as 16 bytes. For matrix processing, these 16 bytes are organized into four columns and four rows. In contrast to DES, the number of rounds in AES is configurable and dependent on the key length. For 128-bit keys, AES employs 10 rounds, 12 rounds for 192-bit keys, and 14 rounds for 256-bit keys. Each of these rounds use a unique 128-bit round key derived from the original AES key.

SubBytes: By looking up a fixed table (S-box) provided in design, the 16 input bytes are substituted. The end result is a four-row, four-column matrix. Each of the matrix's four rows is shifted to the left. Any entries that fall off the right side of the row are re-inserted.

The following is how the shift is done: the first row is not shifted.

The second row has been relocated to the left by one (byte) position.

The third row has been relocated to the left by two spots.

The fourth row is shifted three positions to the left, yielding a new matrix with the same 16 bytes but shifted in relation to one another.

MixColumns: A specific mathematical function is now used to alter each four-byte column.

This function takes four bytes from one column as input and returns four fully new bytes that replace the original column. As a result, a new matrix with 16 additional bytes is created. One thing to keep in mind is that this phase is skipped in the final round.

Addroundkey: The 16 bytes of the matrix are then treated as 128 bits and XORed with the round key's 128 bits. The cipher text is the output if this is the final round. Otherwise, the 128 bits are interpreted as 16 bytes, and the process repeats again.

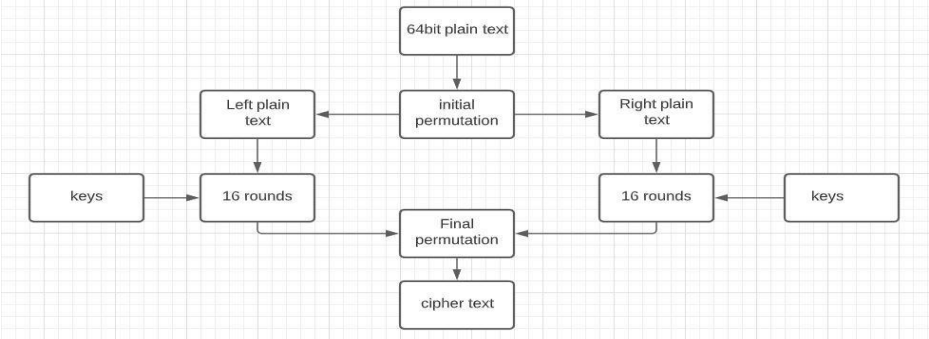
The decryption process follows the same steps as the encryption process, but in reverse order. Each round is made up of four procedures that are performed in reverse order: add round key, mix columns, shift rows, and byte replacement.

Because, unlike a Feistel Cipher, the sub-processes in each round are reversed, the encryption and decryption algorithms must be implemented independently, even though they are identical.

B. DES Algorithm

DES is a popular encryption method that was created in 1970. This encryption converts data into ciphertext and decrypting the ciphertext gives us back the original data that is the plaintext. DES follows a symmetric key method of data encryption. In this, both the sender and receiver use the same keys for encryption and decryption of the message. This is a much faster method than any other asymmetric method. It is hard to break the size of a large key. DES takes input data of block size 64-bits. Then encodes them simultaneously by applying the key to the entire block of 64-bit. The DES algorithm consists of 16 steps.

Each step in this is called around. DES contains some functions which are carried out for every individual round. Each round in this performs functions such as permutation, expansion permutation, and shifting and substitution box in the end.



The 64-bit block divides the input into two equal parts. The data bits get divided on the left of 32-bits and the right of 32-bits. The 32 bit round of the left key and the right key is expanded from 32-bits to 48-bit through 'expansion permutation'.

In the 64-bit key, every eighth bit is used as a parity checking bit, which makes it an equivalent of a 56-bit key. From this 56-bit key, a different 48-bit sub key is generated during each round using process as key transformation. To carry out this process, the 56-bit key is divided into two halves, each consisting of 28-bits. The output of this key transformation which compresses the 56-bit key to 48-bits is XOR-ED with expansion permutation process which in turn expands the 32-bit to 48-bits and is then given to the substitution box.

The 48 bits are given to the S-box into 8 blocks, each of 6 bits. The S-box replaces all the 6 bits of data to 4 bits of data. The 32 bits from S-box are sent to the 'permutation function'. Now, the 'Permutation function' bits are XOR-Ed with a 32-bit Li, where i is from 0 to 16 because 16 rounds need to perform in DES. This output of XOR is connected to the Ri of the next round. Li of the next round is connected to Ri of this round. The output from Li of 32 bits and Ri 32 bits is sent to the 'Final permutation'. Then we get overall 64-bits of cipher text.

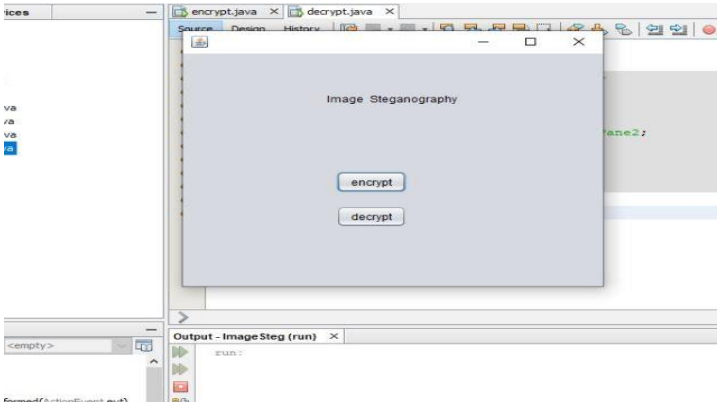
When we give 64-bit input data with 64-bit key data to perform 16 rounds of operations we get 64-bit output that is nothing but a cipher text, this is the encryption process. The decryption process is to be carried out with the same key but in the exact reverse process.

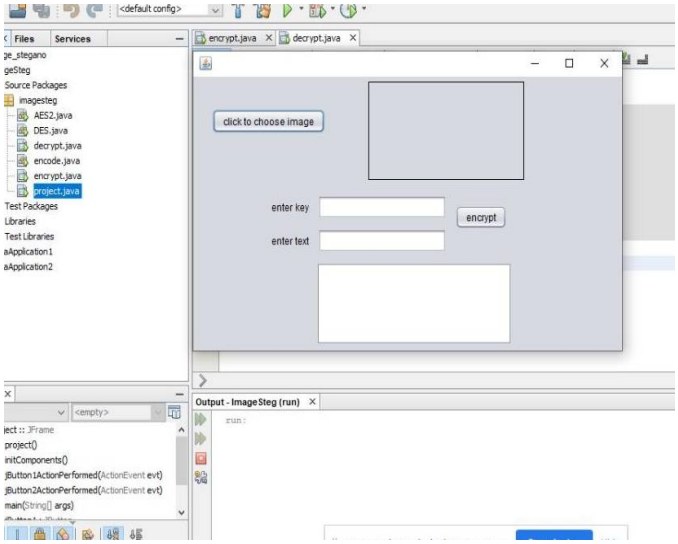
C. LSB Algorithm

The last bit in a pixel is called the least significant bit because its value affects the pixel value only by "1". LSB steganography is a technique in which the least significant bit of the image is replaced with a data bit. As this method is vulnerable to steganalysis so as to make it more secure, raw data is encrypted before embedding it in the image. This encryption process does increase the time complexity, but anyhow it provides higher security. This approach is very simple and one of the most fundamental algorithms when it comes to image steganography. In this method the least significant bits of some or all of the bytes inside an image is replaced with bits of the secret message.

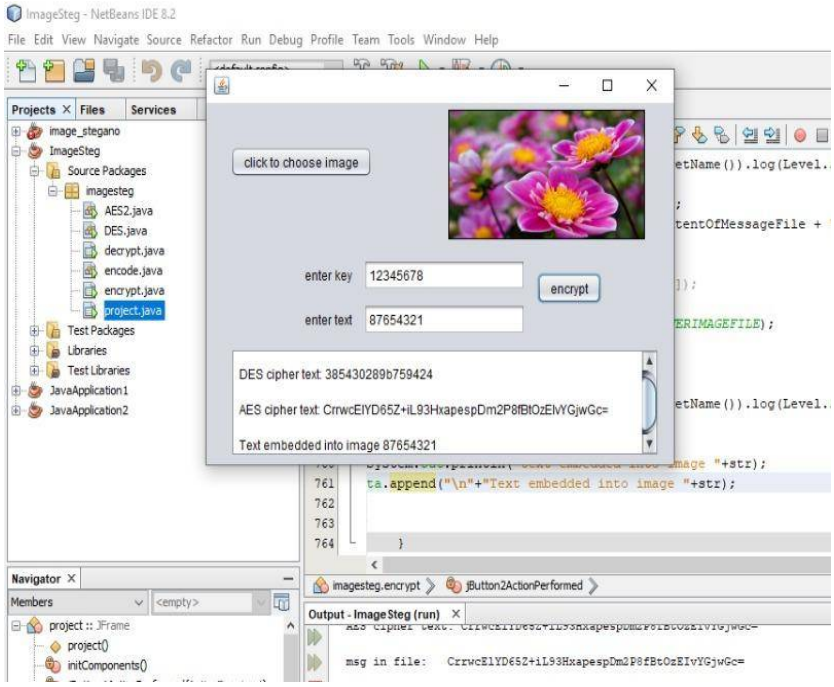
IV. WORKING

In this proposed model, the user has to install the software and an interface appears. On this the user sees two options, one being,' encrypt" and the other being "decrypt". The user has to choose the option according to his/her requirement. If the user chooses to encrypt, then, an option appears through which the user can choose the image in which he/she wants to hide or embed the secret message. After this, the user has to enter the secret message and a unique key.





At the back-end, the secret message gets through three algorithms namely, DES, AES and LSB. First, the message goes through the DES algorithm after which it gets a cipher block as its output. With the help of the key which the user gives as input, the DES algorithm is performed. This is the end of step 1. Now this cipher block is given as input to the next algorithm which is AES, and the same unique key is carried forward and used for the implementation of the AES algorithm also. This marks the end of step 2 of the encryption process. Now the final step is, that the output of step 2 (which is DES and AES encrypted) is used as a final input for the LSB algorithm. This cipher is embedded in the image which was selected by the user initially. Hence, a final image is shown as the output which has our secret message. This steno image looks the same as the input image having no secret message, to the naked eye. The steno image can be sent across through any medium along with the secret key to decrypt.

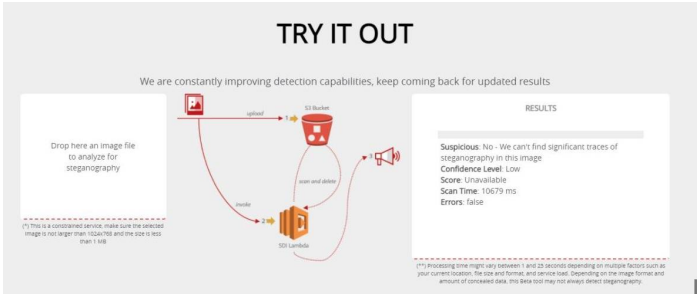


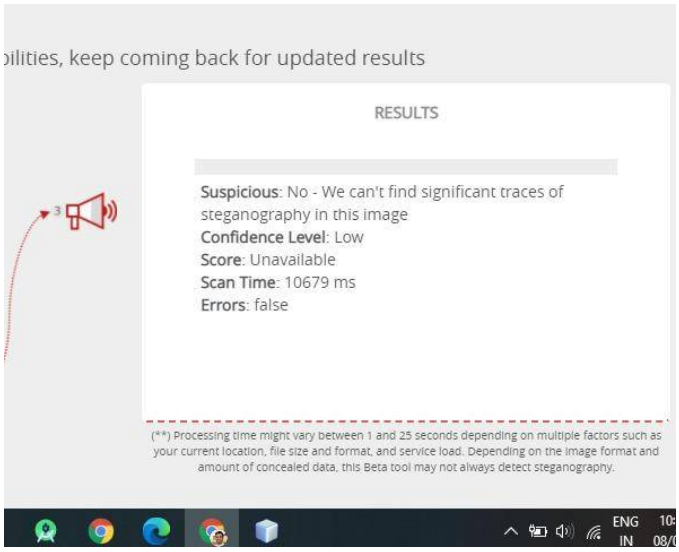
Decryption is the exact reverse process of the encryption which was performed. So, when the receiving side user gets the stego image, he/she can upload this stego image in the software and insert the secret key to begin the decryption process. At the back-end, for decryption, the secret message in the stego image is extracted following the reverse process of algorithms. That is, it first extracts the cipher text from LSB, then decrypts through AES and then through DES at the end.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.429
Volume 9 Issue VI June 2021- Available at www.ijraset.com

V. RESULTS AND EXPERIMENTATIONS

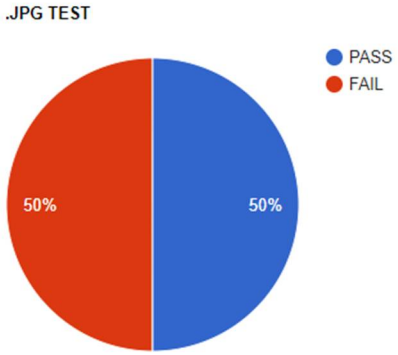
Testing and experimentations have been carried out using 2 different types of images, namely, JPG and PNG format. These formats have been tested using two image sizes and input sizes respectively. A testing software which detects steno images, has been used to test the output steno images of the software.





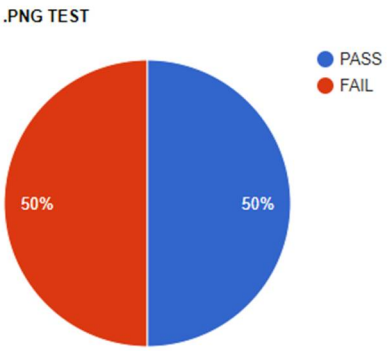
It is observed that the software is not able to detect any percentage of steganography in the images having 64-bit size. However, it is noted that the software does not support 128-bit size. So, the test cases having 64-bit size, pass. And the test cases with 128-bit fail. Not because of steno detection, but because of the limitation of the input size.

IMAGE TYPE	IMAGE SIZE	INPUT SIZE	KEY LENGTH	TEST
.JPG	700x467	64-BIT	8	PASS
.JPG	700x467	128-BIT	8	FAIL
.JPG(GRAY)	130x97	64-BIT	8	PASS
.JPG(GRAY)	130x97	128-BIT	10	FAIL



The testing of PNG type shows similar observations. The test cases having 64-bit size, pass. And the test cases with 128-bit fail. Not because of stego detection, but because of the limitation of the input size.

IMAGE TYPE	IMAGE SIZE	INPUT SIZE	KEY LENGTH	TEST
.PNG	663x497	64-BIT	8	PASS
.PNG	663x497	128-BIT	8	FAIL
.PNG	554x413	64-BIT	8	PASS
.PNG	554x413	128-BIT	10	FAIL



With this experimentation, it is safe to conclude that input size of 64 bits is accepted and does not get detected by steganographic tools. 128-bits and larger sizes puts forward a limitation of the proposed model. The future scope of the model would be to make the software friendly to greater input sizes.

VI. APPLICATIONS AND DISADVANTAGES

Steganography can be used to conceal data at any moment. There is no proof of the message's existence, even if it is suspected. Steganography can be used in the business world to conceal a secret chemical formula or ideas for a new creation.

The simplest and oldest application is map making, where cartographers sometimes add a tiny fictional street to their maps, allowing them to prosecute copycats. Adding fake names to mailing lists as a check against illegal resellers is a similar tactic. To secure a copyright on information, most new applications use steganography, similar to a watermark.

Photo collections released on CD frequently include hidden messages in the photos that can be used to detect unauthorized use. Because the industry manufactures DVD recorders to detect and prevent copying of protected DVDs, the identical technique applied to DVDs is even more effective.

One of the disadvantages is that steganography can also be used for corporate espionage by sending out trade secrets without anyone at the company being any the wiser. Another disadvantage is that terrorists can also learn these techniques and use steganography to keep their communications secret and to coordinate attacks. All of this sounds evil, and steganography's most obvious use are for things like espionage. But there are a number of peaceful applications and advantages of using this technique.

VI. CONCLUSION

The proposed method is highly efficient for secure data transmission. In the process of Steganography, the hidden message is not visible. An attempt has been made to use encryption and decryption techniques to hide data using 3 popular algorithms so that this will provide additional security for the data. The sender and recipient only know how to hide and extract data from carrier files. No eavesdropper or intruder in the middle will know that there are three layers of protection on the message inside the image file. The sender and recipient only know the encryption and decryption instructions.

REFERENCES

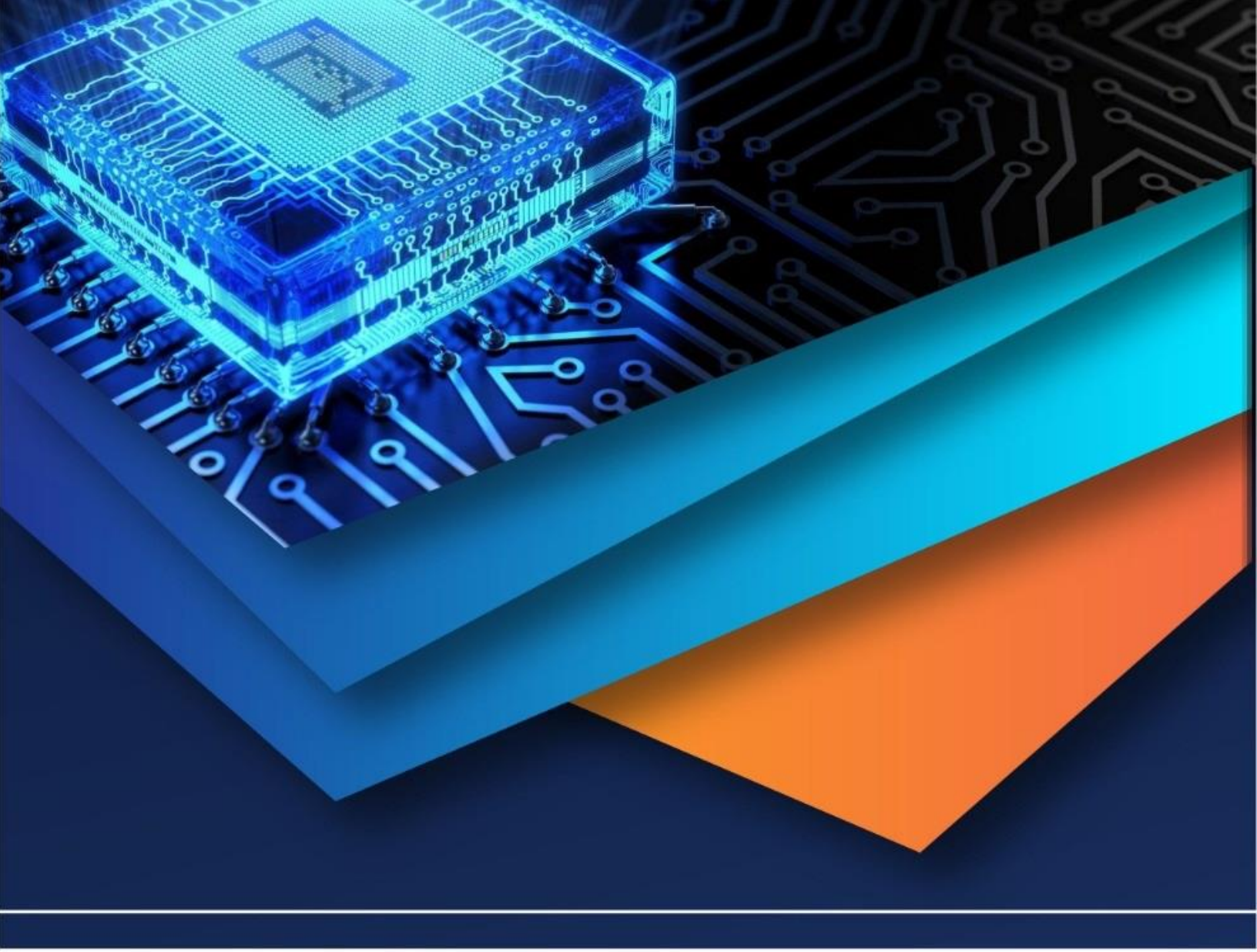
[1] Sneha Bansod, Gunjan Bhure, “Data Encryption by Image Steganography”, International Journal of Information and Computation Technology, ISSN 0974-2239 Volume 4, Number 5, pp. 453-458, 2014.

[2] Indumathi Saikumar “DES- Data Encryption Standard", International Research Journal of Engineering and Technology, e-ISSN 2395 -0056 Volume 4, Issue 03, 2017.

[3] Vikas M, Yashwanth E, Veeresh, Sanath Krishna S, Narender.M, “Hybrid Approach to Text & Image Steganography using AES and LSB Technique”, International Research Journal of Engineering and Technology, Volume 5, Issue 4, e-ISSN: 2395-0056, 2018.

[4] Mustafa Sabah Taha, Mohd Shafry Mohd Rahim, Sameer Abdulsattar Lafta, Mohammed Mahdi Hashim, Hassanain Mahdi Alzuabidi, “Combination of Steganography and Cryptography”, IOP Conference Series: Materials Science and Engineering, ISSN: 1757-899X, 2019.

[5] Subhash Panwar, Shreenidhi Damani, Mukesh Kumar, “Digital Image Steganography Using Modified LSB and AES Cryptography”, International Journal of Recent Engineering Research and Development, Volume 03, Issue 06, ISSN 2455-876, 2020.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY