An Effective Method in Steganography to Improve Protection Using Advanced Encryption Standard Algorithm

K Kamalam MCA., MPhil.,
Assistant Professor
Department of Information Technology
KG College of Arts and Science

S.Saranya MSc.,
Assistant Professor
Department of computer Science
KG College of Arts and Science

***Abstract*---Steganography is the art and science of writing hidden messages in such a way that no one, apart from the sender and anticipated reciver, imagine the existence of the information, a form of security through obscurity. It is an emerging area which is used for secured data broadcast over any public media. In this study a novel advance of image steganography based on LSB (Least Significant Bit) insertion, RSA encryption and AES (Advanced Encryption Standard Algorithm) technique for the lossless jpeg images has been proposed. In this paper, we present a strategy of attaining maximum embedding ability in an image in a way that maximum possible neighboring pixels are analyzed for their frequencies, to determine the amount of content to be added in each pixel. The techniques provide a seamless insertion of data into the carrier image and reduce the error consideration and artifacts insertion required to a minimal. We validate our approach with the help of an experimental evaluation on a prototypic implementation of the proposed model.**

***Key Terms*: -Cryptography, Steganography, LSB, RSA, AES Algorithms.**

I. **INTRODUCTION**

Steganography is the technique of hiding confidential message within any media. Steganography is often bemused with cryptography because the two are similar in the way that they both are used to protect confidential information. Steganography is often confused with cryptography because the two are similar in the way that they both are used to protect confidential information. The difference stuck between the two is in the appearance in the processed output; the output of steganography operation is not apparently visible but in cryptography the output is scrambled so that it can draw attention. Steganlys is process to identify of presence of steganography. The majority of the steganography techniques use images a stego-medium. Messages can be hidden in images through many different ways. The most general approaches to information hiding in images are: Least significant bit (LSB) insertion, Masking and filtering techniques, and transformations. Masking and filtering techniques hide information by marking an image in a manner similar to paper watermarks. The least significant bit insertion (LSB) is the most widely used image steganography technique. It embeds message in the least-significant bits of each pixel. In order to increase the embed power two or more bits in each Pixel can be used to embed messages, which have high risk of delectability and representation degradation.

The LSB techniques might use a permanent least significant bit insertion method, in which the bits of information added in each pixel and frames remains invariable or a variable least significant bit insertion, in which the number of bits added in every pixel be different on the surrounding pixels, to pass up degrading the image dependability In this paper we discuss the embedding of text into image all the way through variable size least significant bit insertion operation. The process of insertion of text in our proposed approach is not in order; rather it follows a random order, base on a random algorithm.The technique proposed aim at providing not only maximum insertion capability, but also performs a maximum analysis of surrounding pixels to determine the embedding ability of every pixel. The process results in a stego-image which is very much comparable in appearance to the original image. We propose a steganography model that ensures greatest embedding of information in both gray scalable and color images, and also ensure that maximum pixels are analyzed to determine the embed capability. This would lead to a reduction of the overall error initiation in the image. The stego-figure obtained after application of this investigation would not only have maximum amount of information, but would also have the maximum difference in manifestation with the original picture. Steganography is one of the most essential research subjects in the field of security communications. It differs from cryptography in the intelligence that where cryptography focuses on observance the contents of a message secret, steganography focuses on keeping the survival of a message secret.

Steganography applications that hide data in images generally use a variant of least significant bit (LSB) embedding. In LSB embedding, the data is unknown in the least significant bit of each byte in the image. The size of every pixel depends on the format of the image and generally ranges from 1 byte to 3 bytes. Each single numerical pixel value corresponds to a color. Thus, an 8-bit pixel is able of displaying 256 dissimilar colors. Given two the same images, if the least significant bits of the pixels in one image are changed, then the two images still look equal to the human eye. This is because the human judgment is not responsive enough to notice the difference in color between pixels that are different by 1 unit. Thus, steganography applications make use of LSB embedding because attackers do not notice anything odd or doubtful about an image if its pixel's least significant bits are customized

Steganography is no unusual; attackers contest steganography using steganalysis.

II. STEGANALSIS

Steganalysis is a process where attackers evaluate an image to determine whether it have hidden information in it. A common steganalysis approach is to graph the pixel standards of an image that is suspected of contain hidden data. Statistical investigation is then performed on the graphed pixel values. The attackers hope to come across anomalies in the statistical analysis of these images or Pictures. These anomalies may specify that the image contains a hidden message, and the anomalies may offer some approaching in to how to extract the hidden message. In this paper a exact image based stegano graphic model has been proposed which uses a JPEG lossless image as the cover up data and the top secret information is embedded in the uncover to form the stego image. Before embedding the secret messages has been encrypted using the public key RSA algorithm. This application also provides a convenience to convert JPEG imagery in the lossless JPEG images. We can choose JPG images for this application because most images mutual by people today are in the JPG format. Thus to an attacker, the actuality that an image other than that of JPEG format is being transfer between two entities could suggestion of suspicious activity. So these know how to be an improvement by using JPEG Lossless images. To contest steganalysis, this application performs an investigation on the user's records of images. This analysis allows users to hide their information in the image that is least likely to be exposed to steganalysis. Steganography is the art and science of hiding communication; a steganographic system thus embeds hidden content in unremarkable cover media so as not to arouse an eavesdropper's thought. In the past, people used secret tattoos or invisible ink to convey steganographic content. Today, computer and net technologies provide easy-to-use communication channels for steganography. Essentially, the in sequence-hiding process in a steganographic system starts by identifying a cover medium's redundant bit. The embedding process creates a stego in-between by replacing these redundant bits with data from the secreted message. Modern steganography's goal is to keep its mere presence invisible, but steganographic systems—because of their invasive nature—leave behind detectable traces in the cover average. Even if clandestine content is not revealed, the survival of it is: modifying the cover medium changes its statistical property, so eavesdroppers can detect the distortions in the resulting stego medium's statistical properties. The process of judgment these distortions are called statistical steganalysis. These articles discuss existing steganographic systems and presents recent research in detecting them via statistical steganalysis. Other surveys focal point on the general usage of information hiding and watermarking or else provide an overview of detection algorithms. Here, we nearby recent research and discuss the practical application of detection algorithms and the mechanisms for getting around them.

In most circumstances, the security is heightened by a required key to reverse the Steganography process.

a) To develop a RSA algorithm for cryptography. Firstly we develop a algorithm for the loading the image in the database. We develop a code for steganography by using a novel score-level combination strategy.

b) Designing and implement the developed algorithm for the steganography purpose for the message and image. Develop a code for Procedures For Hide Text. This procedure hides the encrypted data into the image by searching the best position in the image. The best position defines those Least Significant Bits of the image which extremely match with the encrypted data bits. The output of this procedure is the updated image in which data is hidden.

c) Procedure for Reveal Text: This procedure reveals the secret message which is hidden in the best position of the stego image. This message is in the encrypted form so the message is decrypted by the receiver's private key. Then the original message is presented to the receiver.

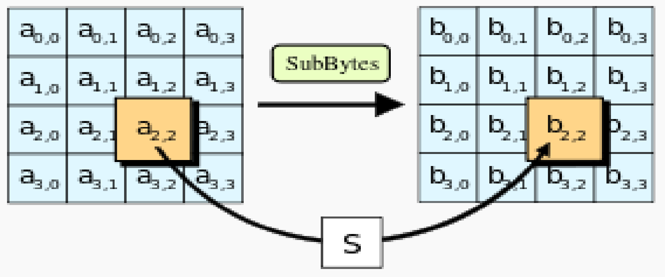
d) The objective of steganography is to hide a secret message within a cover-media in such a way that others cannot discern the presence of the hidden communication. Technically in simple words "steganography means hiding one piece of data within another".

e) Modern steganography uses the opportunity of hiding information into digital multimedia files and also at the network packet level.

f) The stego function operates over cover media and the message (to be hidden) along with a stego-key (optionally) to produce a stego media (*S*).

g) In this method binary equivalent of the message (to be hidden) is distributed among the LSBs of each pixel.

Another figure of message hiding is digital watermarking, which is the technique that embeds information called a watermark, tag or label into a multimedia article such that watermark can be identify or extracted later to make an statement about the object. The editorial may be a Picture, image, audio, video or text only. The AES cipher Like DES, AES is a symmetric block cipher algorithm. This means that it uses the similar key for both encryption and decryption algorithm. However, AES is somewhat different from DES in an amount of ways. The algorithm Rijndael allows for a variety of block and key sizes and not just the 64 and 56 bits of DES' block and key size. The block and key can in detail be chosen independently from 128,160,192,224,256 bits and need not be the same. However, the AES standard states that the algorithm can only accept a block size of 128 bits and a choice of three keys -128,192,256 bits. Depending on which version is used; the name of the standard is modified to AES-128, AES-192 or AES-256 respectively. As well as these differences AES differs from DES in that it is nota feistel structure. Recall that in a feistel structure, half of the data block is used to modify the other half of the data block and then the halves are swapped. In this case the entire data block is processed in parallel during each round using substitutions and permutations.


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

Fig,1 SubBytes

AES is based on a design standard known as a substitution-permutation network, grouping of both substitution and permutation, and is fast in both software and hardware.[9] Nothing like its predecessor DES, AES does not use a Feistel cipher network. AES is a variation of Rijndael which has a permanent block size of 128 bits, and a key size of 128, 192, or 256 bits. By contrast, the Rijndael specification as such is specified with block and key sizes that may be every multiple of 32 bits, both with a smallest of 128 and a maximum of 256 bits.AES operates on a 4×4 column-major order matrix of bytes, termed the state, even though some versions of Rijndael have a bigger block size and have additional columns in the state. Most AES calculations are done in a particular field. The key size used for an AES cipher specifies the number of repetitions of transformation rounds that change the input, called the plaintext, into the final output, and called the cipher text. The numbers of cycles of repetition are as follows:

- 10 cycle of repetition for 128-bit keys.
- 12 cycle of replication for 192-bit keys.
- 14 cycle of reappearance for 256-bit keys.

Each round consists of several dispensation steps, each containing four similar but different stages, including one that depends on the encryption key itself. A set of reverse round are applied to transform cipher text back into the original plaintext using the same encryption key.

a) Key Expansions—round keys are derived from the cipher key using Rijndael's key schedule. AES requires a separate 128-bit round key block for each round plus one more.

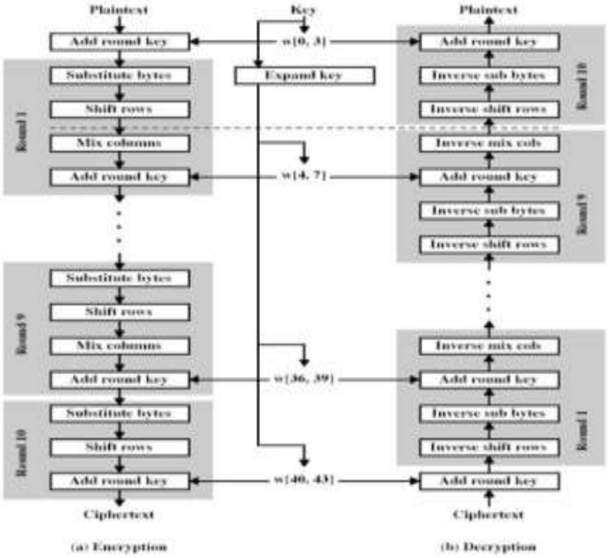
b) InitialRound
   a. AddRoundKey—each byte of the state is combined with a block of the round key using bitwise x.Rounds
   b. SubBytes—a non-linear substitution step where each byte is replaced with another according to a lookup table.
   c. ShiftRows—a transposition step where the last three rows of the state are shifted cyclically a certain number of steps.
   d. MixColumns—a mixing operation which operates on the columns of the state, combining the four bytes in each column.
   e. AddRoundKey

c) Final Round (no MixColumns)
   i. SubBytes
   ii. ShiftRows
   iii. AddRoundKey.

Inner Workings of a Round The algorithm begins with an Add round key point followed by 9 rounds of four stages and a tenth round of three stages. This applies for both encryption and decryption within the exception that each stage of a surrounding the decryption algorithm is the inverse of its counterpart in the encryption algorithm. The four stages are as follows:

1. Substitute byte
2. Shift rows
3. Mix Columns
4. Add Round Key

Again, the tenth round just leaves out the Inverse Mix Columns stage. every of these stages will now be considered in more detail.



(a) Encryption

(b) Decryption

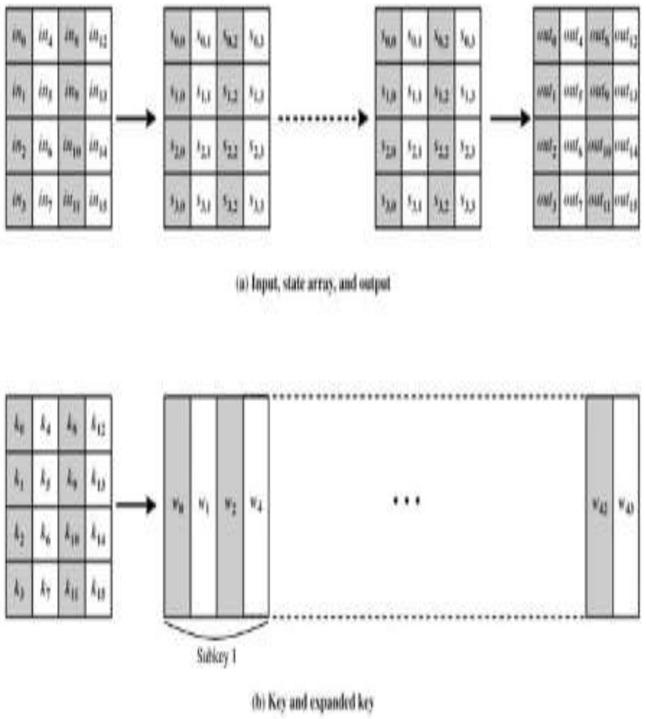
Fig, 2 Overall structure of the AES algorithm

The tenth round simply leaves out the Mix Columns stage. The first nine rounds of the decryption algorithm consist of the following:

1. Inverse Shift rows
2. Inverse Substitute bytes
3. Inverse Add Round Key
4. Inverse Mix Columns

III. SUBSTITUTE BYTES

This point (known as Sub Bytes) is basically a table research using a16 × 16 matrixes of byte values called an s-box. This matrix consists of all the potential combinations of an 8 bit series (2^8= 16×16 = 256). However, the s-box is not just an arbitrary transformation of these values and there is a well apparent method for creating the s-box tables. The designers of Rijndael show how this was complete unlike the s-boxes in DES for which no groundwork was given. We will not be too troubled here how the s-boxes are ready up and can simply take them as table lookups. Again the matrix that gets operated leading throughout the encryption is known as state. We will be concerned with how this matrix is exaggerated in each round. For this particular round each byte is mapped into a new byte in the following way: the leftmost nibble of the byte is used to specify a particular row of the s-box and the rightmost nibble specifies a column.


(a) Input, state array, and output

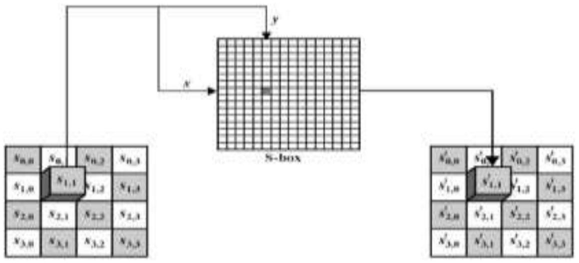
(b) Key and expanded key

Fig,3 Key and Expand key

For example, the byte {95}(curly brackets represent hex values in FIPS PUB 197) selects row 9 column 5 which turns out to hold the value{2A}.This is then used to revise the condition matrix. Figure 7.3 depicts this idea.

The s-box is designed to be resistant to Known cryptanalytic attack specifically; the Rijndael developers sought a design that has a low association between input bits and output bits, and the property that the output cannot be described as a simple mathematical function of the input.

In addition, the s-box has no permanent points (s-box (a) =a) and no opposite fixed points (s-box(a) =−a) where−a is the bitwise praise of a. This-box must be invertible if decryption is to be possible (Is-box[s-box(a)]=a) however it should not be its self inverse i.e. s-box(a)6=Is-box(a).



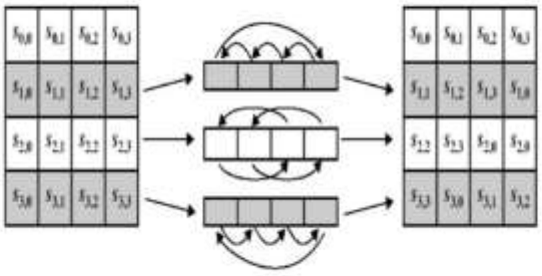
Fig, 4 **Substitute Bytes Stage of the AES algorithm.**

The opposite substitute byte transformation (known as InvSubBytes) makes use of an opposite s-box. In this case what is most wanted to select the value {2A} and get the value {95}.

IV.SHIFT ROW TRANSFORMATION

This stage (known as ShiftRows) is shown in figure 7.5. This is a straightforward permutationan nothing added It works as follow:

- The first row of form is not altered.
- The second row is shift 1 bytes to the left in a round manner.
- The third row is shifting 2 bytes to the left side in a round manner.
- The fourth row is shifting 3 bytes to the left in a round manner.



Fig, 5 Shift Rows stage.

The Inverse Shift Rows transformation (known as InvShiftRows) performs these cir-cular shifts in the opposite direction for each of the last three rows(the first row wasunaltered to begin with).This operation may not appear to do much but if you think about how the bytes are ordered with in state then it can be seen to have far more of an impact. Remember that state is treated as an array of four byte columns, i.e.the first column represents bytes1, and 2,3and 4. A one byte shift is therefore a linear distance of four bytes. This transformation also ensures with the intention of four bytes of one column are spreadout to four different columns.

This stage (known as MixColumn) is basically a substitution but it makes use of arith-metic of GF(2^8). Each column is operated on independently. Every byte of a column is mapped into a new value that is a function of all four bytes in the column. This transformation can be determined by the following matrix multiplication on state

$$\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \begin{bmatrix} s_{0,0} & s_{0,1} & s_{0,2} & s_{0,3} \\ s_{1,0} & s_{1,1} & s_{1,2} & s_{1,3} \\ s_{2,0} & s_{2,1} & s_{2,2} & s_{2,3} \\ s_{3,0} & s_{3,1} & s_{3,2} & s_{3,3} \end{bmatrix} = \begin{bmatrix} s'_{0,0} & s'_{0,1} & s'_{0,2} & s'_{0,3} \\ s'_{1,0} & s'_{1,1} & s'_{1,2} & s'_{1,3} \\ s'_{2,0} & s'_{2,1} & s'_{2,2} & s'_{2,3} \\ s'_{3,0} & s'_{3,1} & s'_{3,2} & s'_{3,3} \end{bmatrix}$$

Each element of the product matrix is the sum of products of elements of one row andone column. In this case the individual additions andmultiplications are performed in GF(2^8). The Mix Columns transformation of a single column j (0≤j≤3) of state can be expressed as:

$$\begin{aligned} s'_{0,j} &= (2 \bullet s_{0,j}) \oplus (3 \bullet s_{1,j}) \oplus s_{2,j} \oplus s_{3,j} \\ s'_{1,j} &= s_{0,j} \oplus (2 \bullet s_{1,j}) \oplus (3 \bullet s_{2,j}) \oplus s_{3,j} \\ s'_{2,j} &= s_{0,j} \oplus s_{1,j} \oplus (2 \bullet s_{2,j}) \oplus (3 \bullet s_{3,j}) \\ s'_{3,j} &= (3 \bullet s_{0,j}) \oplus s_{1,j} \oplus s_{2,j} \oplus (2 \bullet s_{3,j}) \end{aligned}$$

Where • denotes multiplication over the finite field GF (2^8)

Add Round Key Transformation

In this stage (known as AddRoundKey) the 128 bits of state are bitwise XORed with the 128 bits of the round key.

- The operation is viewed as acolumnwise operationbetween the 4 bytes of astatecolumn and one word of the round key.
- This transformation is as easy as possible which helps in efficiency but it also effects every bit ofstate.

V.CONCLUSIONS

The steganography is used to covert message to transfer secrete information. In this paper Steganography using effective protection in proposed. The secret communication is embedded into smaller matrix of size 16x16 and inserted into input image. In future the technique can be confirmed for robustness. We added the RSA algorithm and AES algorithm process. Presently, this application supports hiding data in lossless jpg images. Future improvement of this application would be extending its functionality to support hiding data in video files or in other file format.

REFERENCES

[1] Raja K B, C R Chowdary, Venugopal K R, L M Patnaik. (2005) :"A Secure Steganography using LSB, DCT and Compression Techniques on Raw Images," IEEE International Conference on Intelligence Sensing and Information processing, pp.171-176.

[2] Kumar V and Kumar D. (2010): "Performance Evaluation of DWT Based Image Steganography," IEEE International Conference on Advance Computing, pp. 223-228.

[3] Weiqi Luo, Fangjun Huang, and Jiwu Huang. (2010): "Edge Adaptive Image Steganography Based on LSB Matching Revisited," IEEE Transactions on Information Forensics and Security, no. 2, vol. 5, pp. 201-214.

[4] R O El Safy, H H Zayed and A El Dessouki (2009): "An Adaptive Steganographic Technique Based on Integer Wavelet Transform," International Conference on Networking and Media Convergence, pp.111-117.

[5] Mathkour H, Al-Sadoon B and Touir A. (2008): "A New Image Steganography Technique. :" International Conference on Wireless Communications, Networking and Mobile Computing, pp.1-4.

[6] V Vijaylakshmi,G Zayaraz and V Nagaraj. (2009):"A Modulo Based LSB Steganography Method," International Conference on Control,Automation,Communication and Energy Conservation, pp. 1-4.

[7] Wien Hong, Tung-Shou Chen and Chih-Wei. (2008):"Lossless Steganography for AMBTC-Compressed Images," Congress on Image and Signal Processing, pp.13-17.

[8] A W Naji, Teddy S Gunawan, Shihab A Hameed, B B Zaidan and A A Zaidan. (2009): "Stego-Analysis Chain, Session One," International Spring Conference on Computer science and Information Technology, pp. 405-409.

[9] M Hassan Shirali-Shahreza and Mohammad Shirali-Shahreza. (2008): "A New Synonym Text Steganography," International Conference on Intelligent Information Hiding and Multimedia Signal Processing, pp. 1524-1526.

[10] Vladimir Banoci, Gabriel Bugar and Dusan Levicky (2009): "Steganography Systems by using CDMA Techniques," International Conference on Radioelectronika, pp.183-186.

[11] Chen Ming, Zhang Ru, Niu Xinxin and Yang Yixian (2006): "Analysis of Current Steganographic Tools: Classifications and Features," International Conference on Intelligent Hiding and Multimedia Signal Processing, pp. 384-387.

[12] Mankun Xu, Tianyun Li and Xijian Ping. (2009): "Estimation of MB Steganography Based on Least Square Method," International Conference on Acoustics, Speech and Signal Processing, pp. 1509-1512.

[13] Abbas Cheddad, Joan Condell, Kevin Curran and Paul Mc Kevitt. (2008): "Enhancing Steganography in Digital Images," Canadian Conference on Computer and Robot Vision, pp. 326-332.

[14] Aos A Z, A W Nazi, Shihab A Hameed, Fazida Othman, B B Zaidan. (2009): "Approved Undetectable-Antivirus Steganography," International Spring Conference on Computer and Information Technology, pp. 437-441.

[15] Daniela Stanescu, Valentin Stangaciu, Loana Ghergulescu and Mircea Stratulat. (2009): "Steganography on Embedded Devices," International Symposium on Applied Computational Intelligence and Informatics, pp. 313-318. [16] Jin-Suk Kang, Yonghee You and Mee Young Sung (2007): "Steganography using Block-Based Adaptive Threshold," International symposium on Computer and Information Sciences, pp. 1-7.

[17] Mci-Ching Chen, Sos S Agaian and C L Philip Chen. (2008): "Generalised Collage Steganography on Images,"

[18] Sumanth Kumar Reddy S, R.Sakthivel and P praneeth "VLSI Implementation of AES Crypto Processor for High Throughput" International journal of advanced engineering science and technologies, Vol No. 6, Issue No. 1, 022 – 026.

[19] M.Vanitha, R.Sakthivel and Subha, "Highly Secured High Throughput VLSI Architecture for AES Algorithm".

[20] L.Thulasimani and M.Madheswaran "A Single Chip Design and Implementation of AES -128/192/256 Encryption Algorithms," International Journal of Engineering Science and Technology, Vol. 2(5), 2010, 1052-1059.

[21] N. Sklavos and O. Koufopavlou, "Architectures and VLSI Implementations of the AES-Proposal Rijndael," IEEE Trans. on Computers, vol. 51, Issue 12, pp. 1454-1459, 2002.

[22] Kaur, Swinder,Vig and Renu , "Efficient Implementation of AES Algorithm in FPGA Device," in Conference on Computational Intelligence and Multimedia Applications, Nov 2007,pp. 179-187.