International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 3 Issue 11, November-2014

A Novel Approach for Data Steganography

Mr. Vikas Rajani
M.Tech. Scholar, Department of Information Technology,
BUIT Bhopal (M.P.), INDIA

Ms. Apurva Saxena,
Professor, Department of Information Technology,
BUIT Bhopal (M.P.), INDIA

Abstract **-- Internet is behaved as a backbone for the current modern technologies; it is globally connected, unsecure network. All the important files, communication, data are transferred through this network. There are many security algorithms that work parallel to make this network secure, but with the development in modern technologies, new algorithms and modification on it is always required. Confidentiality is one of the most important parameter in security to ensure that no other unauthenticated person can understand the meaning of save or transmitted data. This paper focus on this paper and proposed there novel approach to ensure confidentiality. For confidentiality, it proposes an architecture which is a combination of encryption/decryption and text steganography. This paper works on various parameters and improve them shows there implementation results which proves that it is the better solution among all existing solution.**

Keywords **--** ***Computer Security, Network, Encryption/Decryption Algorithm, Cryptography, Symmetric Key Algorithm, Steganography.***

1. INTRODUCTION

As the development of Internet technologies increases, the transmission of digital media is now-a-days convenient over the networks. But secret message transmissions over the Internet system suffer from serious security overhead. So, protecting of secret messages during transmission becomes an important issue. Though cryptography changes the message so that it cannot be understood but this can generates curiosity level of a hacker. It would be rather more sensible if the secret message is cleverly embedded in another media so that no one can guess if anything is hidden there or not. This idea results in steganography, which is a branch of information hiding by camouflaging secret information within other information.The word steganography in Greek means "covered writing" (Greek words "stegos" meaning "cover" and "grafia" meaning "writing") [1]. The main objective of steganography is to hide a secret messageinside harmless cover media in such a way that the secret message is not visible to the observer. Thus the stego_image should not diverge much from original cover image. In this generation, steganography is mostly used on computers with digital data being the carriers and networks being the high speed delivery channels. Figure1 shows the block diagram of a simple image steganography system.

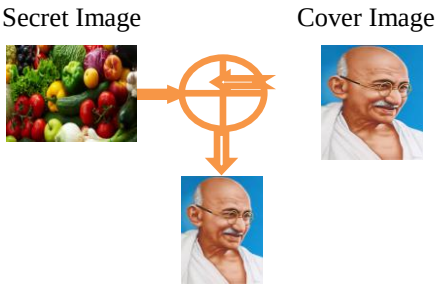

Figure 1: The block diagram of a simple steganography system

Cryptography and steganography are two ways to hide messages and although they complement each other, they are not the same. Cryptography changes the contents of a file or message so that it is unreadable by everyone except the intended recipient. The intended recipient has a key that allows the encrypted file to be invoked and viewed as planned by the sender. Encrypted messages are not hidden, and their comings and goings can be detected and monitored. Once the means of encryption have been revealed, it is still up to the code breaker to uncover the key to decrypt the message. It could think of steganography as a form of robust encryption. It attempts to hide the message in such a way that the observer may not even realize that the message is being exchanged. Unlike encryption, steganography cannot be detected. Often, steganography is used to supplement encryption. Through its combination of encryption and invisibility of the encrypted data it keeps the message completely protected from data espionage. This concept also can provide an easiness of exchanging important messages secretly between a receiver and a sender with multimedia files as carrier of the messages. For this purpose, proposed work has designed an efficient encryption algorithm considering the various cryptanalytic attacks which evolved as the security enhancements were formulated. Proposed technique is the implementation with low running time complexity.

The paper is organized as follows: Section I is Introduction presents the overview of steganography, cryptography and combination of both techniques. Section 2 is devoted to the proposed steganography algorithm. This section presents a detailed description of the proposed algorithm. The experimental results are analyzed in Section 3 Concluding remarks mention in the end the paper.

2. PROPOSED WORK

There are many confidentiality and steganography algorithm, but there is a need of improvement in every algorithm. There are some parameter on which any algorithm is analyzed. When, it is a talk about encryption/decryption algorithm, timing always play an important role, It is always desirable to design an algorithm which sould be highly time efficient. If an algorithm is not atime efficient than it cannot be used for real time communication, hence this kind of algorithm is less preferable. Again, only time efficiency is not only the parameter on which any algorithm can be analyzed. Algorithm should be strong enough so that no attack can break it. The problem in the existing system is that an algorithm which provides time efficiency is not very robust and an algorithm which is strong is not a time efficient. There is always a competition to develop an algorithm which have high time efficiency and also robust too. To provide complete confidentiality only encryption/ decryption algorithm is not enough, it just shuffle the text in such a manner that no one understand its true meaning but there is always a chance to crack this algorithm. To provide full confidentiality it is combined with steganography algorithm which hide the presence of secret transmission so that no one can guess its presence. Many different media file is used to hide these secret data, but using text file is the cheapest and efficient way. Paper [2] also proposed its own way to hide the data behind text file. It uses spaces between two words to hide a single bit of a secret message. Again there are some parameters on which any algorithm can be analyzed that is PSNR value which is used to calculate the distortion on stego file and the second parameter is cover file size which should be minimum. The problem with the existing algorithm discussed in paper [2] is that its cover file size is more. To overcome all the existing problems in the confidentiality algorithm, authors have proposed a new algorithm which is a combination of both encryption/decryption algorithm and steganography algorithm. The proposed work is the solution of the entire existing problem in confidentiality algorithm.

A. Proposed Encryption / Decryption Algorithm

In this section author discussed the complete process of proposed encryption/ decryption algorithm. It is the solution of above algorithm. It is design in such a way that it should be time efficient as well as robust against any kind of attack. It is a symmetric key block cipher algorithm which uses same key at the both end also it uses a block of size 128 bit to transform the plaintext into cipher text.

a. Proposed Encryption Algorithm

Proposed encryption algorithm contains two blocks first, Key generation block and other is shuffling box. Steps for key generation block are as follow:

1. To perform encryption algorithm Key generation block take a 16 character key from user and converted it into 128 bit binary form.
2. Proposed encryption algorithm uses three key generated from user key, 128 bit key entered by user is treated as first key K0.
3. for i=0 to i=1 do the following
   i) Reverse the key ki.
   ii) XOR ki with the result of step i.
   iii) All the bits of result of step ii is XORed with its right bit
   iv) The result is now treated as key K.i+1.

Now, these keys are used in shuffling box. This shuffling box converts the plaintext into cipher text. Steps to transform the plaintext into cipher text are as follow:

1. Convert the complete plaintext into binary format.
2. Now, divide the complete plaintext into number of small size chunks having 128 bits in each chunk. If the size of last bits is less than 128 bits than pad the bit 0 in last block to makes it equal to 128.
3. Repeat the following step for each chunk
   i) Repeat the following steps for i=0 to i<=2
      (1) XOR the plaintext with key Ki
      (2) Perform right bit XOR operation on the resultant value of step (1) with its 9th bit.
      (3) Again, perform XOR operation on reverse of key K (i+2) %3.with result (2)
   ii) Result of step i) is the cipher text of first chunk
4. Exit

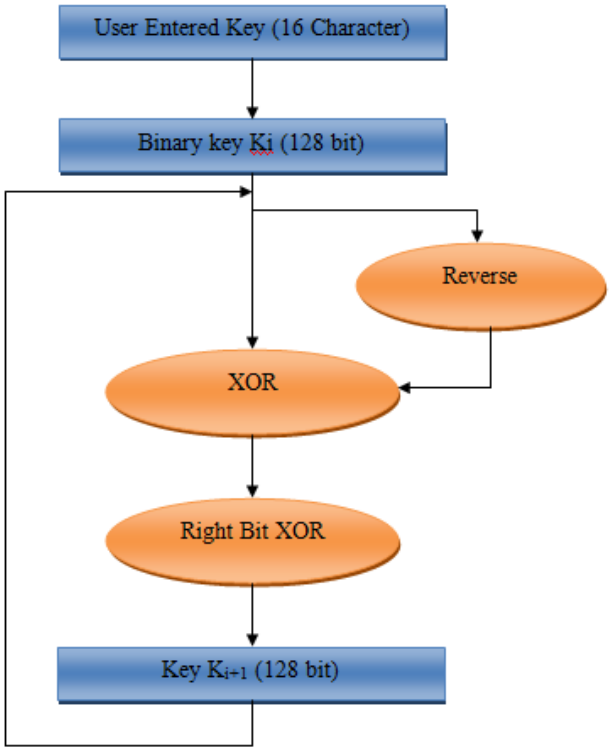

Figure1. Key Generation of proposed Encryption/Decryption Algorithm.

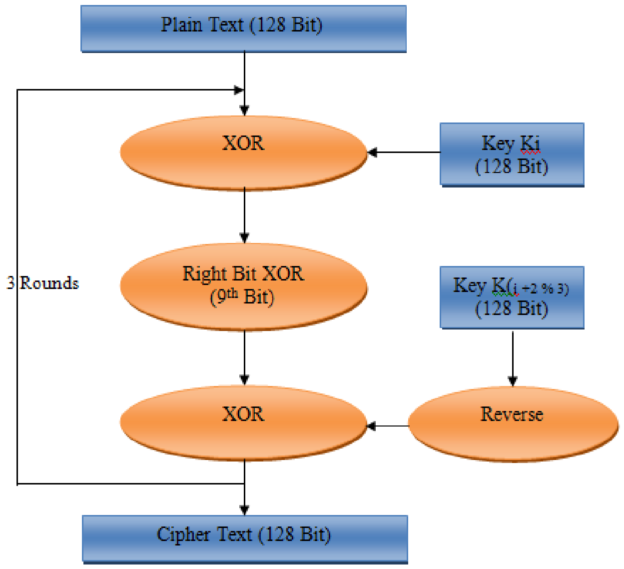

Figure 2. Proposed Encryption Block.

Complete process of encryption algorithm is illustrated with the help of example:

To calculate this result an implementation is done and results at each step is calculated and presented here. The screen shot of proposed software is shown in figure below.

Let the secret message is: "INDIA" and Key is "1234567890123456"

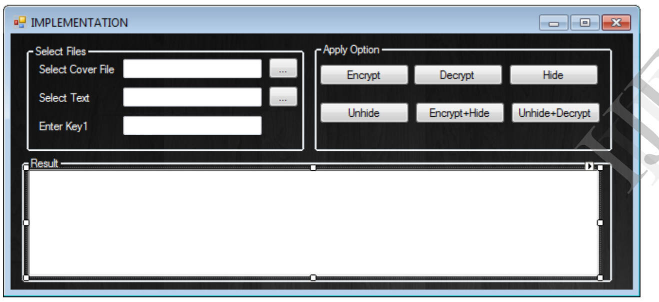

Fig. Screenshot of proposed work implementation

Now first it generates three key

K0 =
"00110001001100100011001100110100001101010011011000110111001110000011100100110000001100010011001000110011001101000011010 1"

K1 =
"10100111001000110000000110001001110010110100111111111010110110101111111111001011010011100100011000000011000100111001011"

K2 =
"10011101110101111000001111010111011100110011100000000100011011000100000000111001100111011101011110000011110101110111 0010"

Now, Plaintext in binary form=
"01001001010011100100010001001001010000010000000000000000000000000000000000000000000000000000000000000000000000000000000000000000"

Round 1:

XOR Operation
"10000111100000011100010001000000101000101111001001110010001100011111000110110011111100111011001101110011001100101111001010110010 01"

Rightbit XOR
"10000000100100101000110110010101000110000101100001000111010010100101100101010010010101010101000101101101011110010110011100100 0"

XOR
"10100000011001111101101001100000010011111101101001100111011000101101001101010000011101011010000100001100101010110000111001001010"

Round 2

XOR Operation
"10111001001110100000010110010111001110101110101111010100011100010100001011001011011011001111110011010011010111000111101101111011"

Rightbit XOR
"11001101001100010010101111100010111011010100001100110110111101001101010000010010100101010101101001101011101010101000110011100001"

XOR
"01011110011000101111100011010001010111100011000011000101100101110011011100000000100000110000010011011100010011001001111111100100 10"

Round 3

XOR Operation
"00011111100010000101011100111011111100010011010010000101110001100010001100000101010001111110001100010111011100111001000010010110"

Rightbit XOR
"00001111001001100010000011011001100110000011111100001001100000000001010011000101010000000111001101111100000101001010111100100010 00"

XOR
"10000011100010001100111100100010001000101010011111010000000001001111000010100011100001101011000110001111110101001000000110000100 00"

CipherText = "ƒˆÏ\"\"§Ð\táG\rc©"

b. Proposed Decryption Algorithm

Steps for converting ciphertext again into plaintext is just reverse of encryption algorithm. Again it is divided into two blocks, first key generation block and second is shuffling block. Steps for key generation block are exactly same as discussed in encryption.

Now, Steps for shuffling block for decryption side are as follow.

1. Convert the complete cipher text into binary format.
2. Now, divide the complete cipher text into number of small size chunks having 128 bits in each chunk.
3. Now, Repeat the following step for each chunk
   i) Repeat the following steps for i=2 to i >=0
      1. perform XOR operation on reverse of key K (i+2) %3 with cipher text

ISSN: 2278-0181
Vol. 3 Issue 11, November-2014

2. Perform reverse right bit XOR operation on the resultant value of step (1) with its 9th bit.
3. XOR the Value with Key Ki

ii) Result of step i) is the plaintext of first chunk

4. Exit

Again, decryption algorithm is illustrated with the help of same example

CipherText = "ƒˆÏ\"\"§Ð\táG\rc©"
Now first it generates three key
K0 =
"00110001001100100011001100110100001101010011011000110111001110000011100100110000001100010011001000110011001101000011010
1"

K1 =
"10100111001000110000000110001001110010110100111111111010110110101111111111001011010011100100011000000011000100111001011"

K2 =
"10011101110101111000001111010111011100110011100000000100011011000100000000111001100111011101011110000011110101110111001
0"

Now, Ciphertext in binary form=
"10000011100010001100111100100010001000101010011111010000000001001111000010100011100001101011000110001111111010100100000011000010000"

Round 1:

XOR Operation
"00001111001001100010000011011001100110000011111100001001100000000010100110001010100000001110011011111000000101001010111100100010001000"

Rightbit XOR
"00011111100010000101011100111011111100010011010010000101110001100010001100000101010001111110001100010111011100111001000010010110"

XOR
"01011110011000101111100011010001010111100011000011000101100101110011011100000000100000110000010011011100010011001001111111001001
0"

Round 2

XOR Operation
"11001101001100010010101111100010111011010100001100110110111101001101010000010010100101010101101001101011101010101000110011100001"

Rightbit XOR
"10111001001110100000010110010111001110101110101111010100011100010100001011001011011011001111110011010011010111000111101101111011"

XOR
"10100000011001111101101001100000010011111101101001100111011000101101001101010000011101011010000100000110010101011000011100100101
0"

Round 3

XOR Operation
"10000000100100101000110110010101000110000101100001000111010010100101100101010010010101010101001000101101101011110010110011100100
0"

Rightbit XOR
"10000111100000011100010001000000101000101111001001110010001100011111000110110011111100111011001101110011001100101111001010110010
01"

XOR
"01001001010011100100010001001001010000001000000000000000000000000000000000000000000000000000000000000000000000000000000000000000"

PlainText = INDIA

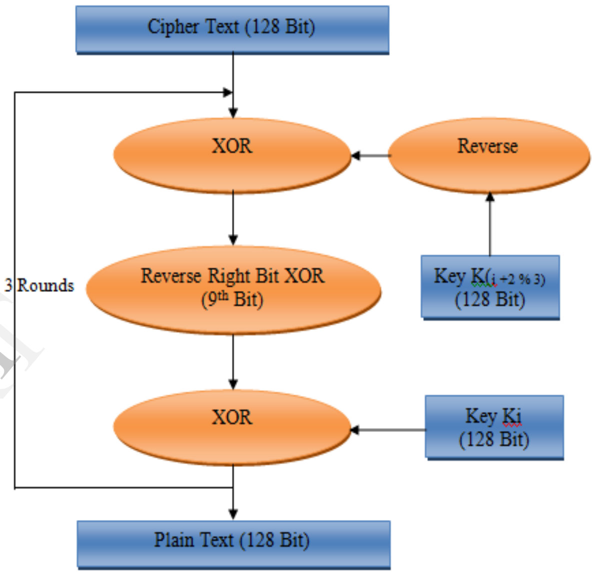

Figure 3. Proposed Decryption Block.

c. Proposed Steganography Algorithm

Proposed encryption/ decryption algorithm discussed earlier is just a text shuffling method; the secret information is always visible to the intruders, so there is always a chance to get attack on it and loss of confidentiality. To make the algorithm more confidential author proposed a new steganography algorithm which uses text file as a cover file to make the algorithm suitable for real time communication. It takes the advantage of color of written text to hide the data. Steps for hiding the data behind text file are as follow:

1) Convert the secret text into binary format.
2) To hide secret text behind text cover file first the length of text cover file is compared, it should contain more characters than the number of bits in secret text, if it is less than the bits of secret text than exit.
3) Now, calculate the number of zero and number of once in secret text. If the number of zeros is greater or equal to number of once than set S = 0 else set S =1
4) Pick the cover file text color of blue component and test its LSB.
5) Replace the second LSB by LSB of blue component of first character of cover file and LSB by the value of S.

6) Now, blue component of next character is tested and if its LSB is 0 and S =0 than hide each bit of secret text behind each character by changing its LSB 0 for 0 in secret text and 1 for 1 in secret text
7) If it's LSB is 1 and S =0 than hide each bit of secret text behind each character by changing its LSB 0 for 1 in secret text and 1 for 0 in secret text
8) If LSB is 0 and S =1 than hide each bit of secret text behind each character by changing its LSB 1 for 0 in secret text and 0 for 1 in secret text
9) And if LSB is 1 and S =1 than hide each bit of secret text behind each character by changing its LSB 0 for 0 in secret text and 1 for 1 in secret text
10) Repeat step 5 and step 6 for each bit in secret text.
11) Exit

Un-hiding of proposed steganography is exactly reveres of hiding algorithm. Steps of un-hiding process are as follow:

1) To un-hide the secret text from the cover file first test the LSB of blue component of first character, if it is 0 than set S = 0 and if S=1 and test second LSB, if it is 0 than set L = 0 else set L =1.
2) if L =0 and S =0 or L =1 and S = 1, read the LSB of blue component of each character from second character, the result is secret text.
3) If L=1 and S =0 or L =0 and S =1, read the LSB of blue component of each character from second character, calculate the complement of result and the result comes is secret text.
4) Exit.

Proposed algorithm is implemented and tested, during experiment many files are hide behind word file. One of the examples of experimental results are shown below.
Secret text that is used for hided is:
"INDIA"
Cover file is

For once, a book which really lives up to its title. Hall self-published this massive tome in 1928, consisting of about 200 legal-sized pages in 8 point type; it is literally his *magnum opus*. Each of the nearly 50 chapters is so dense with information that it is the equivalent of an entire short book. If you read this book in its entirety you will be in a good position to dive into subjects such as the Qabbala, Alchemy, Tarot, Ceremonial Magic, Neo-Platonic Philosophy, Mystery Religions, and the theory of Rosicrucianism and Freemasonry.

After hiding the text the file look like

For once, a book which really lives up to its title. Hall self-published this massive tome in 1928, consisting of about 200 legal-sized pages in 8 point type; it is literally his *magnum opus*. Each of the nearly 50 chapters is so dense with information that it is the equivalent of an entire short book. If you read this book in its entirety you will be in a good position to dive into subjects such as the Qabbala, Alchemy, Tarot, Ceremonial Magic, Neo-Platonic Philosophy, Mystery Religions, and the theory of Rosicrucianism and Freemasonry.

3. PERFORMANCE ANALYSIS

Design an algorithm is not worthy till it doesn't analysis properly. Authors have done complete analysis on proposed algorithm to check whether it is a good solution or not for confidentiality. Authors have done analysis individually on both the algorithm and shows there implementation results. Dot Net implementation has used to test these algorithms. For experiment, Intel Core i5 2.40 Ghz, 4 GB of RAM and Window-7 Home Basic SP1, have used on which performance data is collected. Authors have analyzed these algorithms against latest work discussed in paper 1 and paper 2.

A. Encryption/Decryption Analysis

To analysis encryption/decryption algorithm, authors have taken many parameters on which performance of algorithm is tested. To check whether it is suitable for real time communication or not or can be used for ad-hoc network time efficiency is calculated. To check whether it is strong against various attack it internal structure is tested by calculating avalanche effect and also key analysis is also done for the same.

a. Time Efficiency Analysis

It is always required that an algorithm should be time efficient, if an algorithm is not time efficient it doesn't matter how strong it is, it is not worthy. For using an algorithm in real time communication, it should work fast. Also Ad-Hoc network works on battery, it is important that an algorithm should be simple and fast so that battery consumption should be low. Table 1 and Table 2 show the timing analysis between proposed and paper [2] encryption algorithm.

TABLE 1: COMPARISON OF PROPOSED ENCRYPTION ALGORITHM WITH PAPER [2] ENCRYPTION ALGORITHM ON VARIOUS FILE SIZE

File Size in KB	Algorithm (Time in Second)	
	Paper [2]	Proposed Encryption Algorithm
1 KB	9.282	0.015
5 KB	17.189	0.036
10 KB	25.287	0.109

TABLE 2: COMPARISON OF PROPOSED DECRYPTION ALGORITHM WITH PAPER [2] DECRYPTION ALGORITHM ON VARIOUS FILE SIZE

File Size in KB	Algorithm	
	Execution Time in Second	
	Paper [2]	Proposed Decryption Algorithm
1 KB	9.108	0.015
5 KB	17.226	0.033
10 KB	25.301	0.110

Figure 4 and Figure 5 shows the graphical representation of timing analysis presents in Table 1 and Table 2. It is clearly understand from Figure 4 and Figure 5 that proposed encryption/decryption algorithm is efficient than other.

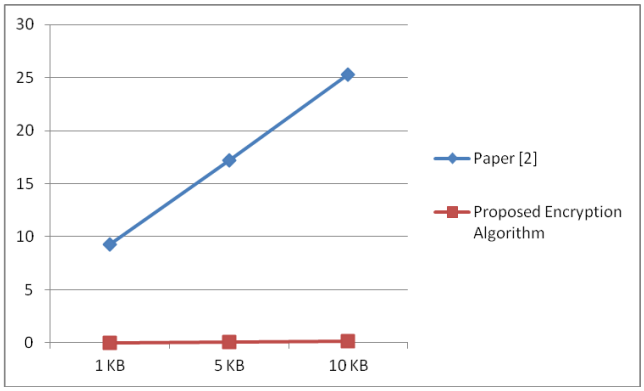

Figure 4. Comparison of Proposed Encryption Algorithm with Paper [2] Encryption Algorithm on various file size

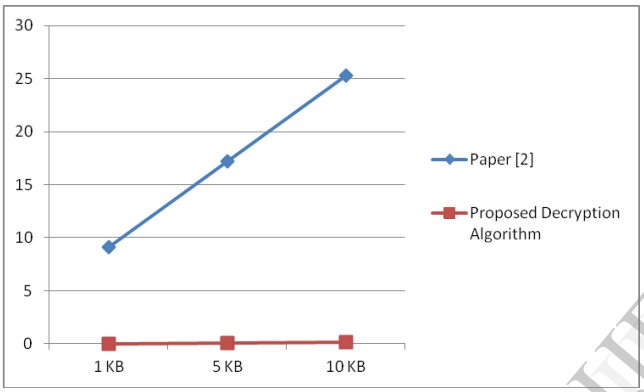

Figure 5. Comparison of Proposed Decryption Algorithm with Paper [2] Decryption Algorithm on various file size

4. *Avalanche Effect Analysis of Proposed Algorithm*

It is very important that algorithm should be enough strong so that it cannot be attacked by any intruder. To calculate the strength of internal structure of proposed algorithm avalanche effect is calculated. For this authors, have changed a single bit in the key and calculated the percentage of changes in cipher text. The result is shown in Table 3 and its graphical representation is shown in Figure 6.

TABLE 3 AVALANCHE COMPARISON BETWEEN PAPER [2] AND PROPOSED ALGORITHM

File Size in KB	Avalanche Effect	
	Paper [2]	Proposed Algorithm
Single bit change in key	43.15%	48.99%

Again it is easily calculated from the results that proposed algorithm have strong internal structure and considered as a better solution.

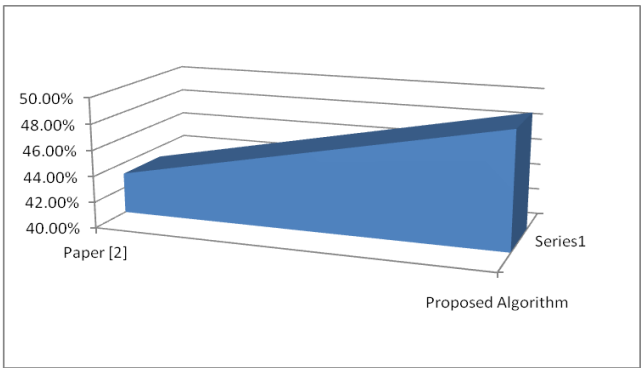

Figure 6. Avalanche comparison between Paper [2] and Proposed Algorithm

a. *Key Analysis of proposed Algorithm*

Proposed encryption/decryption algorithm uses a symmetric key of size 128 bit. To crack this key, an approx 2^{128} combination is required. This value is very high, such that a super computer is failed to solve it in reasonable time.

B. *Proposed Steganography Algorithm*

Proposed algorithm is a combination of encryption/decryption algorithm and steganography algorithm. Here, complete analysis of steganography is done. To analyze steganography algorithm parameter like PSNR value and cover file size is calculated and also it compares with algorithm proposed in paper 1 and paper 2.

a) *PSNR value*

PSNR value is used to calculate the distortion in the original image and the stego image. It is obvious that if something is hided inside, than it get distorted but this distortion should be minimum for designing the best solution. Authors have calculated PSNR value of proposed work and compare it with the Paper [1] and Paper [2]. Table 4 shows the result after calculating PSNR value. If the PSNR value is high it means its distortion is low and if PSNR is low means distortion is high.

TABLE 4 PSNR COMPARISON BETWEEN PAPER [1], PAPER [2] AND PROPOSED ALGORITHM

File Size in KB	PSNR Value		
	Paper [1]	Paper [2]	Proposed Algorithm
1 KB	68.6	21.09	70.30

It is clearly seen from the Table 4 and its graphical representation in Figure 7 shows that proposed algorithm have less distortion than paper 1.

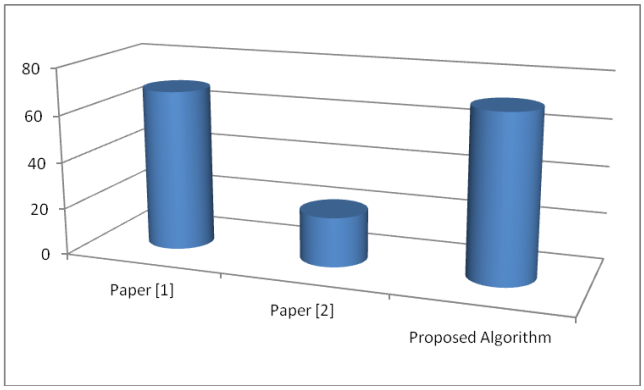

Figure 7. PSNR comparison between Paper [1], Paper [2] and Proposed Algorithm

a. Cover File Size

Cover file size is another parameter that is used to calculate the efficiency of steganography algorithm. For a steganography algorithm is necessary that distortion should be minimum as well as it is also required that cover file size should be minimum. Proposed algorithm hides the bit behind every character so that it requires file having character equal to size of bit length in secret file.

5. CONCLUSION

With the changes in the latest technology, it is required to update the existing algorithm according to the requirement. In this paper, authors have design and developed a new confidentiality algorithm which is a combination of two algorithm first encryption/decryption algorithm and other is steganography algorithm. Authors have designed this algorithm in such a way that it is time efficient, robust, low cover file size and high PSNR value. It is the better solution that can be use for real time transmission, ad-hoc network and any channel which required security.

6. RESULTS

As discussed the proposed algorithm is robust, efficient and best solution for security. Authors have also shown the experimental data in which it passes a secret text "INDIA" with a secret key "1234567890123456" and hided behind a cover file

For once, a book which really lives up to its title. Hall self-published this massive tome in 1928, consisting of about 200 legal-sized pages in 8 point type; it is literally his *magnum opus*. Each of the nearly 50 chapters is so dense with information that it is the equivalent of an entire short book. If you read this book in its entirety you will be in a good position to dive into subjects such as the Qabbala, Alchemy, Tarot, Ceremonial Magic, Neo-Platonic Philosophy, Mystery Religions, and the theory of Rosicrucianism and Freemasonry.

and at decryption end the receiver find the word file

For once, a book which really lives up to its title. Hall self-published this massive tome in 1928, consisting of about 200 legal-sized pages in 8 point type; it is literally his *magnum opus*. Each of the nearly 50 chapters is so dense with information that it is the equivalent of an entire short book. If you read this book in its entirety you will be in a good position to dive into subjects such as the Qabbala, Alchemy, Tarot, Ceremonial Magic, Neo-Platonic Philosophy, Mystery Religions, and the theory of Rosicrucianism and Freemasonry.

and with the help of same key "1234567890123456" it recover the original text "INDIA".

REFERENCES

[1] Xing Tang, Mingsong Chen," Design And Implementation Of Information Hiding System Based On RGB", Consumer Electronics, Communications and Networks (CECNet), IEEE-2013

[2] Rishav Ray, Jeeyan Sanyal, Debanjan Das, Asoke Nath, A new Challenge of hiding any encrypted secret message inside any Text/ASCII file or in MS word file: RJDA Algorithm", 2012 International Conference on Communication Systems and Network Technologies,IEEE-2012

[3] Manoj Kumar Ramaiya, Naveen Hemrajani, Anil Kishore Saxena, "Security Improvisation in Image Steganography using DES", IEEE-2012.

[4] Thomas Leontin Philjon. J, Venkateshvara Rao. N, Metamorphic Cryptography -A Paradox between Cryptography and Steganography Using Dynamic Encryption, IEEE-International Conference on Recent Trends in Information Technology, ICRTIT 2011.

[5] Yambin Jina Chanu , Themrichon Tuithung , Kh Manglem singh," A Short Survey on Image Steganography and Steganalysis Technique " , IEEE Trans, 2012 science and Management (ICAESM- 2012) 709 - 713.

[6] W. Luo, F. Huang, J. Huang, Edge adaptive image steganography based on LSB matching revisited, IEEE Trans. Inf. Forens. Security 5 (2) (2010) 201-214.

[7] Ge Huayong, Huang Mingsheng, Wang Qian , "Steganography and Steganalysis Based on Digital Image", IEEE Trans. International Congress on Image and Signal Processing,(2011) 252-255.

[8] Amitava Nag, Saswati Ghosh, Sushanta Biswas, Debasree Sarkar, Parta Pratim Sarkar " An Image Steganography Technique using X-Box Mapping", IEEE Trans. International Conference Advances in Engineering,

[9] Guilliang Zhu, Weiping Wang, "Digital Image Encryption algorithm based on pixel", ICIS – 2010 IEEE International Conference 29-31 Oct 2010, pp – 769 – 772.

[10] Jasmin Cosic , Miroslav Bacai, " Steganography and Steganalysis Does Local web Site contain "Stego" Contain " , 52 th IEEE Trans. International Symposium ELMAR-2010, Zadar, Croatia 2009 ,pp 85 – 88.

[11] Zhang Yun-peng , Liu Wei " Digital Image Encryption Algorithm Based on chaos and improved DES ", System, man and Cybernatics ,SMC 2009 , IEEE International Conference 11-14 Oct 2009, pp 474-479.

[12] Saeed R. Khosravirad, Taraneh Eghlidos and Sharokh Ghaemmaghami, "Higher Order Statistical of Random LSB Steganography", IEEE Trans. 2009, pp 629 - 632.

[13] J. Mielikainen, LSB Matching Revisited, IEEE Signal Process. Lett. 13 (5) (2006) 285-287.

[14] N Provos and P. Honeyman, "Hide and seek: An Introduction to Steganography", IEEE Security and Privacy, 2003, pp32-44.

[15] Donovan Artz" Digital Steganography: Hiding Data within Data ", Los Alamos National Laboratory, IEEE Trans. 2001, pp 75-80.

[16] K Suresh Babu , K B Raja, Kiran Kumar k, Manjula Devi T H, Venugopal K R, L M Pathnaik" Authentication of Secrete Information in Image Steganography", IEEE Trans. 13.

[17] Moerland, T, "Steganography and Steganalysis", Leiden Institute of Advanced Computing Science, www.liacs.nl/home/ trnoerl/privtech.pdf.

[18] Schaefer " A Simplified Data Encryption Standard Algorithm", Cryptologia, January 1996

[19] Data Encryption Standard : http://csrc.nist.gov/publications/fips /fips 46-3 /fips- 46-3.pdf

[20] Advanced Encryption Standard http://csrc.nist.gov/publications/ fips/fips197/fips- 197.pdf

[21] Cryptography and network Security Principles and Practices, Charles Fleeger

[22] William Stallings, "Network Security Essentials (Applications and Standards)", Pearson Education, 2004.