International Journal of Current Engineering and Technology
E-ISSN 2277 – 4106, P-ISSN 2347 – 5161

Available at http://inpressco.com/category/ijcet

Research Article

A Hybrid Method with Lorenz attractor based Cryptography and LSB Steganography

Hemanta Kumar Mohanta†* and J. Hyma†

†Department of computer Science and Engineering, GITAM University, Visakhapatnam

Accepted 26 April 2015, Available online 01 May2015, **Vol.5, No.3 (June 2015)**

Abstract

The rapid growth of internet made the human life easier. In internet the message transfer takes less time than conventional system. In this paper we propose an encryption technique which combines both cryptography and Steganography to protect information. . In this method we use chaos cryptography to encrypt the text where the initial parameters are taken as key and a new LSB Steganography technique to hide the cipher in image. Chaotic Systems are basically nonlinear and exhibiting an apparently random behavior for certain ranges of values of system parameters. Cryptography is protecting the privacy of the message where the Steganography is used for hide the information.

Keywords: *Cryptography, LSB Steganography etc.*

1. Introduction

Cryptography is the scrambling of data, so the presence of data known to everyone but nobody can read it. The message is called plain text and the encrypted message is called cipher text. Cryptography is two types (i) Private key cryptography (ii) Public key cryptography. In private key cryptography the same key is use for encryption and decryption. In public key cryptography two key are used one for encryption called public key and other for decryption called private key.

Image Steganography is the process of hiding data inside an image (X. Zhang *et al*,2005). With Steganography one cannot know the presence of data by seeing (S. M. MasudKarim*et al*,2011). Various type of Steganography is used such as image-Steganography, audio-Steganography and video-Steganography (M. Hossain*et al*,2009).

The simplest approach to hiding data within an image is called least significant bit (LSB) insertion (R Praveen Kumar *et al*,2013)(Ahaiwe J). For 24-bit true color image, the amount of changes will be minimal and indiscernible to the human eye. As an example, suppose that we have three adjacent pixels [nine bytes] with the following RGB encoding:

10010101	00001101	11001001
10010110	00001111	11001010
10011111	00010000	11001011

Now suppose we want to hide the following 9 bits of data101101101. If we overlay these 9 bits over the LSB of the 9 bytes above, we get the following (where bits in bold have been changed) pixels:

10010101	00001100	11001001
10010111	00001110	11001011
10011111	00010000	11001011

The following formula provides a very generic description of the pieces of the Steganography process:

Stego-image = cover image + information
Information maybe text OR image etc.

Chaos is one of the possible behaviours associated with evolution of a nonlinear physical system and occurs for specific values of system parameters. (LjupcoKocarev, *et al*)The discovery of this apparently random behaviour ensuing out of deterministic systems turned out to be quite revolutionary leading to many issues interconnecting stability theory, new geometrical features and new signatures characterising dynamical performances.

Property of chaotic system : Systems which are basically nonlinear and exhibiting an apparently random behaviour for certain range of values of system parameters are referred to as Chaotic. However, the solutions or trajectories of the system remain bounded within the phase space. This unstable state has a strong dependence on the values of the parameters and on the way the system begins.

2. Steganography Related work and Experimental result

2.1 Steganography

Steganography literally means Hidden Writing and has been used for thousands of years. The basic image

*Corresponding author **Hemanta Kumar Mohanta** is a M.Tech (CST) Scholar and **Dr. J. Hyma** is working as Assistant Professor

Steganography means hide message inside a cover image. In this paper we use Steganography to hide text message as well as image inside a cover image.

Steganography process

To hide a text message, first we convert the ASCII character to double which give the binary value of the text

Example: 'Hello World' is 72, 101, 108, 108, 111, 32, 87, 111, 114, 108 and 100.

To hide image we convert the message to uint8 which change the pixel value in between 0-255.

Next the decimal values are converted into 8bit binary sequence array. A key is chosen between 0-255. The key is converted into 8bit binary sequence. And the key bitwise xor with each sequence of message array.

Example:

8bit array	1	0	0	1	0	1	1	0
Key	1	1	0	0	1	1	1	1
Xor	0	1	0	1	1	0	0	1

After xor the resultant array is ready to store in the canvas image.

Now each pixel is converted into binary form. A RGB image pixel contains 24 bit red, green, blue color 8 bit each. Now we embedded the message to cover image in RGBBGRRG position.

2.2 Steganography encryption method

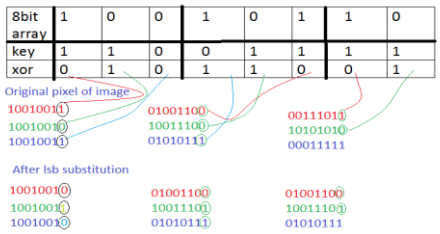

Figure 1 LSB encryption

Finally the output image is called stego image

2.3 Decryption process

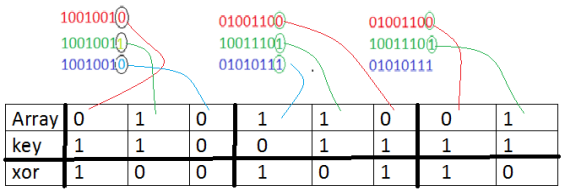

Figure 2 LSB decryption

First the LSB bit was extracted from the Stego-image. Then the array is converted into 8bit sub array. Each 8 bit sub array performs bit xor operation with the secret key. Then they are converted to their ASCII character.

2.4 Message embedded process

In this paper we have created three methods to perform Steganography.
Method 1: RGBBGRGB
Method 2: RGBBGRRB
Method 3: RGBBGRRG

2.4.1 Method 1

We try to hide the 8 bit message in RGBBGRGB color space:

Algorithm:

Step1: input {8 bit binary sequence and cover image (M←row size, N←column size)};
Step2: initialize header for input message position of header==1;
Step3: LSB method
If
Value of header =0
Operation ← bit-and (header, 254)
else
Operation←bit-or (header, 1)
Step4: Set counter to perform data hiding
{Rr, Rc, Gr, Gc, Br, Bc} r←row, c←column
Initially *set all* counter =1;
Step5: hiding technique (method1)
Header ←1, Rr←1,Rc←1
Operation ←step3
Rc←Rc+1 (if Rc=M: Rr←Rr+1)
Header ←2, Gr←1,Gc←1
Operation ←step3
Gc←Gc+1 (if Gc=M: Gr←Gr+1)
Header ←3, Br←1,Bc←1
Operation ←step3
Bc←Bc+1 (if Bc=M: Br←Br+1)
Header ←4, Br←1,Bc←2
Operation ←step3
Bc←Bc+1 (if Bc=M: Br←Br+1)
Header ←5, Gr←1,Gc←2
Operation ←step3
Gc←Gc+1 (if Gc=M: Gr←Gr+1)
Header ←6, Rr←1,Rc←2
Operation ←step3
Rc←Rc+1 (if Rc=M: Rr←Rr+1)
Header ←7, Gr←1,Gc←3
Operation ←step3
Gc←Gc+1 (if Gc=M: Gr←Gr+1)
Header ←8, Br←1,Bc←3
Operation ←step3
Bc←Bc+1 (if Bc=M: Br←Br+1)
Step6: repeat step 5 ←length of the message

2.4.2 Method2

We try to hide the 8 bit message in RGBBGRRB color space:
Algorithm:
Step1: input {8 bit binary sequence and cover image (M←row size, N←column size)};
Step2: initialize header for input message position of header==1;
Step3: LSB method
If
Value of header =0
Operation ← bit-and (header, 254)
else
Operation←bit-or (header, 1)
Step4: Set counter to perform data hiding
{Rr, Rc, Gr, Gc, Br, Bc} r←row, c←column
Initially s*et al*l counter =1;
Step5: hiding technique (method1)
Header ←1, Rr←1,Rc←1
Operation ←step3
Rc←Rc+1 (if Rc=M: Rr←Rr+1)
Header ←2, Gr←1,Gc←1
Operation ←step3
Gc←Gc+1 (if Gc=M: Gr←Gr+1)
Header ←3, Br←1,Bc←1
Operation ←step3
Bc←Bc+1 (if Bc=M: Br←Br+1)
Header ←4, Br←1,Bc←2
Operation ←step3
Bc←Bc+1 (if Bc=M: Br←Br+1)
Header ←5, Gr←1,Gc←2
Operation ←step3
Gc←Gc+1 (if Gc=M: Gr←Gr+1)
Header ←6, Rr←1,Rc←2
Operation ←step3
Rc←Rc+1 (if Rc=M: Rr←Rr+1)
Header ←7, Rr←1,Rc←3
Operation ←step3
Rc←Rc+1 (if Rc=M: Rr←Rr+1)
Header ←8, Br←1,Bc←3
Operation ←step3
Bc←Bc+1 (if Bc=M: Br←Br+1)
Step6: repeat step 5 ←length of the message

2.4.3 Method 3

We try to hide the 8 bit message in RGBBGRGB color space:
Algorithm:
Step1: input {8 bit binary sequence and cover image (M←row size,N←column size)};
Step2: initialize header for input message position of header==1;
Step3: LSB method
If
Value of header =0
Operation ← bit-and (header, 254)
else
Operation←bit-or (header, 1)
Step4: Set counter to perform data hiding
{Rr, Rc, Gr, Gc, Br, Bc} r←row, c←column
Initially s*et al*l counter =1;
Step5: hiding technique (method1)
Header ←1, Rr←1,Rc←1
Operation ←step3
Rc←Rc+1 (if Rc=M: Rr←Rr+1)
Header ←2, Gr←1,Gc←1
Operation ←step3
Gc←Gc+1 (if Gc=M: Gr←Gr+1)
Header ←3, Br←1,Bc←1
Operation ←step3
Bc←Bc+1 (if Bc=M: Br←Br+1)
Header ←4, Br←1,Bc←2
Operation ←step3
Bc←Bc+1 (if Bc=M: Br←Br+1)
Header ←5, Gr←1,Gc←2
Operation ←step3
Gc←Gc+1 (if Gc=M: Gr←Gr+1)
Header ←6, Rr←1,Rc←2
Operation ←step3
Rc←Rc+1 (if Rc=M: Rr←Rr+1)
Header ←7, Rr←1,Rc←3
Operation ←step3
Rc←Rc+1 (if Rc=M: Rr←Rr+1)
Header ←8, Gr←1,Gc←3
Operation ←step3
Gc←Gc+1 (if Gc=M: Gr←Gr+1)
Step6: repeat step 5 ←length of the message

The cover image is the main image in which the hidden information will be embedded. The resultant image is the stego image which is the same type of image as the cover image. To measure the quality of stego image, Peak Signal-to- Noise Ratio (PSNR) is calculated. PSNR is a statistical measurement used for digital image or video quality assessment [3]. PSNR is most easily defined via the mean squared error (MSE) which for two m×n monochrome images I and K where one of the images is considered a noisy approximation of the other is defined as:

$$MSE = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[I(i,j) - k(i,j)]^2 \quad (1)$$

The PSNR is defined as:

$$PSNR = 10\log_{10}\left(\frac{MAX_I^2}{MSE}\right) = 20\ \log_{10}\left(\frac{MAX_1}{\sqrt{MSE}}\right) \quad (2)$$

Larger PSNR indicates better quality of the image or in other terms lower distortion. The larger the PSNR value the smaller the possibility of visual attack by human eye.

Table 1: Comparison of PSNR values with three different steganography methods

Text byte	Image	Method1 PSNR value	Method 2 PSNR value	Method 3 PSNR value
108BYTE	LEENA 256*256	74.3184	74.4392	74.3182
648	LEENA 256*256	66.8755	66.7838	66.7838
1.26KB	LEENA 256*256	63.9101	63.8454	63.8603
2.53KB	LEENA 256*256	60.9053	60.8682	60.8869
108BYTE	LEENA 512*512	80.7225	80.4410	80.5650
648	LEENA 512*512	72.9695	72.9077	72.8747
1.26	LEENA 512*512	69.8991	69.3165	69.8883
2.53KB	LEENA 512*512	66.9870	66.9196	66.9321
108byte	Cover.jpg	82.9658	83.2602	83.0869
648	Cover.jpg	75.5497	75.4331	75.5066
1.26	Cover.jpg	72.6099	72.5461	72.5577
2.53	Cover.jpg	69.5663	69.5671	69.5873

3. Chaotic cryptography Related Work

In 1963, Edward Lorenz developed a simplified mathematical model for atmospheric convection (Bergé*et al*,1984) The model is a system of three ordinary differential equations now known as the Lorenz equations.

The Lorenz equations are

$$\text{(i)}\ \frac{dx}{dt} = \sigma(y - x) \tag{3}$$

$$\text{(ii)}\ \frac{dy}{dt} = x(\rho - z) - y \tag{4}$$

$$\text{(iii)}\ \frac{dz}{dt} = xy - \beta z \tag{5}$$

Here x, y, and z make up the Lorentz system state, t is time, and $\boldsymbol{\sigma}$, ρ ,β are the Lorenz system parameters. $\boldsymbol{\sigma}$ is the Prandtl number, ρ is the Rayleigh number and β is the geometrical parameter.

For cryptography values of $\boldsymbol{\sigma}$, ρ ,β are positive. The Lorenz attractor is the solution obtained for the Lorenz system with ρ = 28, σ = 10, and β = 8/3.

The role of parameter ρ in the chaotic behaviour of the Lorenz equations for fixed σ = 10, and β = 8/3
When 0< ρ <1, the origin is globally stable.
When ρ >1, the origin is non-stable.
When 1< ρ <24.74,

$$C_1=(\sqrt{\beta(\rho-1)},\sqrt{\beta(\rho-1)},(\rho-1)) \tag{6}$$

$$C_2=(\sqrt{\beta(\rho-1)},\sqrt{\beta(\rho-1)},(\rho-1)) \tag{7}$$

Will be stable.

When ρ >24.74 , the two points C_1 ,C_2 became unstable known as Lorenz attractor.

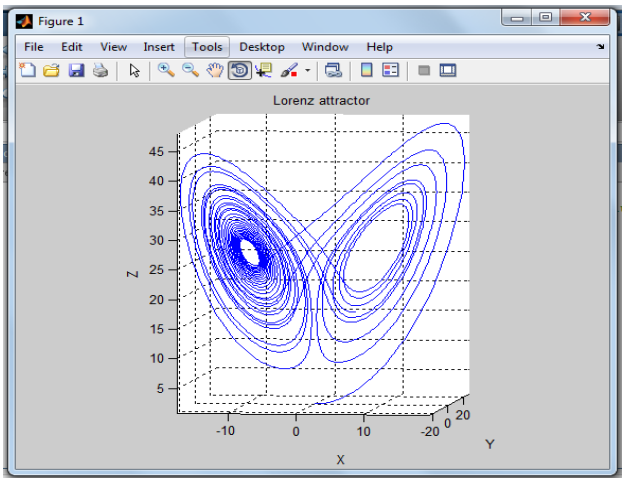

Figure 3 Lorenz attractor for ρ = 28, σ = 10, and β = 8/3.

Chaotic systems are very sensitive to initial conditions and system parameters. For a given set of parameters in chaotic regime, two close initial conditions lead the system into divergent trajectories. Therefore encryption / decryption scheme can be obtained if the parameters are chosen as "Keys" and "Trajectories" are used for encryption/decryption.

3.1 Parameter selection or key generation

Using initial conditions $x(0)$, $y(0)$ and $z(0)$ and the values of the parameters σ, β, ρ appropriate for generating chaos, the Lorenz equations are solved by 4th order RK method for obtaining $x(t)$, $y(t)$ and $z(t)$ for time T [i.e. time steps N0] until the transient part of the trajectory is crossed and system enters into chaos. The system is run to generate at least 60000 points of the trajectory. Variables *x, y* and *z* are transformed using "modulo *p*" function as:

$x = |x \bmod p|$,$y = |y \bmod p|$, $z = |z \bmod p|$, $1 \le p \le 5$.

One of the variables x, y, or z is chosen and called v. From the frequency plot, select a maximum and a minimum value of v so as to give a frequency ~100.

$$\text{Cell size } S=\frac{v_{max}- v_{min}}{S} \tag{8}$$

Used to divide the trajectory into sites S ≤ 256. To each of these sites, an ASCII character is associated with.

V_{min}											V_{max}
%	?	A	b	•	•	•	•	$	#	@	*
1	2	3	4	•	•	•	•	S-3	S-2	S-1	S

Figure 4 Division of Lorentz attractor into S Sites

3.2 Encryption

Encryption of a message M is then carried out on the following lines. Lorenz dynamics is carried out using x (0), y (0) and z (0) and the values of the parameter σ, β, ρ. The chosen variable on transformation becomes:

$$v = |v \bmod p|$$

Encryption of a character in M involves running the dynamics from initial conditions *x* (0), *y* (0) and *z* (0), until the ν value falls in the interval corresponding to the required site associated with the character. The number of time steps *n* to reach the required site should be greater than N0 (transient cross over). Further, a random number 'k' from a uniform distribution is generated and compared with a pre-chosen valueη ∈ [0,1]. If k > η then the number of time steps 'n' is the encryption of the character. This procedure is repeated until the whole message M is encrypted. The encrypted message *Cn*is now a set of integers less than 65532.

3.3 Decryption

To decrypt the cipher text C_n: {n1, n2, n3……., ni ,….}, Lorenz dynamics is run with the same parameters and initial conditions as in the encryption. The time evolution is continued up to the number of time steps, ni = n1. The value of the chosen variable corresponding to n1 is located on one of the sites. The associated ASCII value of the reached site gives us the decrypted character. The steps are continued until the whole cipher text is decrypted.

3.4 Properties

3.4.1 Sensitivity to initial conditions

Given an initial state of a deterministic system [nonlinear system, in general], it is well known that the future states of the system can be predicted. However, for chaotic systems, long term prediction is impossible. For specific values of parameters, two trajectories, which are initially very close, diverge exponentially in a short time. Initial information about the system is thus completely lost.

3.4.2 Ergodicity

Ergodicity is that property in which a trajectory in phase space comes arbitrarily close to its earlier states. Trajectory of a chaotic system in its evolutionary wanderings also satisfies this property. It essentially reflects that the system eventually is confined to a spatial object, a set of points called an attractor. The density of such points is time invariant and this property is essential to cryptography.

3.4.3 Mixing

It is a characteristic of a system in which a small interval of initial conditions gets spread over the full phase space in its asymptotic evolution. In a chaotic system, an arbitrary interval of initial conditions spreads over the part (attractor) of the phase space to which the trajectory asymptotically confines. Thus any region gets into every other region of the spatial attractor of phase space**.**

4. Proposed system and experimental Result

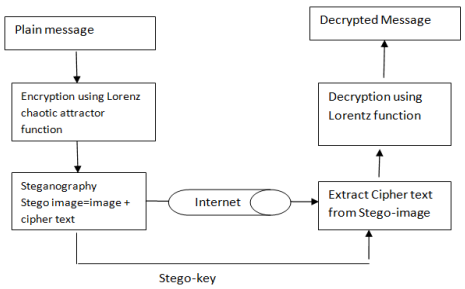

Figure 5 Proposed model

Figure 5 Describe the proposed model of the paper

4.1 Sender

Step1: first Read the text message
Here the text message is the abstract part of this paper contains 681 byte of information.
Step2: Encryption
The encrypted message is shown in figure 6 and the trajectory of encrypted message in Lorenz attractor is shown in figure 7 contains 681 byte of information.

haa rge btcm hrp ynlanatsepardlpli iiohrr arpo ne l gi hpom e nuo.oCrr m erokslnylpnhsag rsbc cmfyg
Lefoigeh eaete hsmhcoihh gyeo rermae atdeotohnaie tehthtsaihwoxta rnspaoi irnkhnehwpeeeiemainutye
ettipe tel moi eo ag.t rly aramsspe.st Ce .nt h c pqnSsiei f gieyino aeaatc trttcr.cnftt riomftp henaotauir
yesioos rheoSpednhys ia t Tmaegeteap dadratdoeyBasy hcnrinsutt pnrclpten e rlrosn rh oobinawnh
thaagerhSeboeaoehfdru e dewmfherttynnvetei e u syiytayna wetrw tt riaf pda istamyvcc nd sheartsetw ueS ct
opa eyimrlna aohoi pcvr r acishotfehmorSphtetgehpvntn.b aerhtgntrt esn ermrf gitioaieycs te ntspee ene
gooiipta nsintaesnqhne npl es nag.ainskeenstge hcr a a nx

Figure 6 Encrypted message

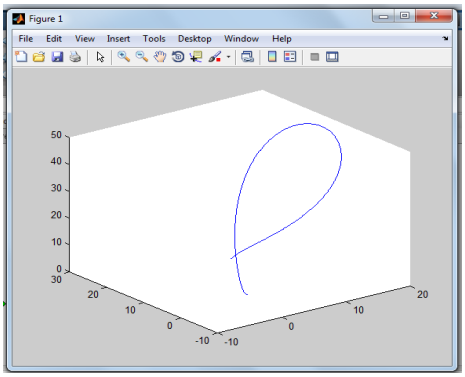

Figure 7 Trajectory Lorenz attractor created by length of the message

Step 3: Message embedding or Steganography method
We choose method1 for Steganography because it gives good result as compared with other methods shown in table1.Cipher text contains 681 byte of information. Cover image is lena.bmp (512*512, 768 kb)PSNR value = 72.605Figure 8 shows the histogram of original image and figure 9 shows histogram of Stego-image

Figure 8 Histogram of original image

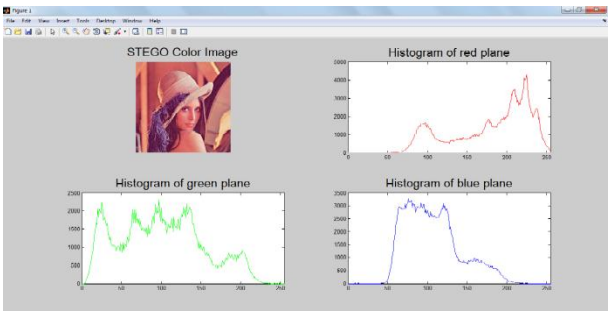

Figure 9: Histogram of Stego-image

The histogram of an image normally refers to a histogram of the pixel intensity values. This histogram is a graph showing the number of pixels in an image at each different intensity value found in that image. The image histogram is a valuable tool used to view the intensity profile of an image. The histogram provides information about the contrast and overall intensity distribution of an image (Cox I. *et al*, 1995). The pixel intensities are plotted along the x-axis and the number of occurrences for each intensity value represents the y-axis.

4.2 Receiver

Step1: After receiving the Stego-image the receiver first extracts the message from the cover.
Step2: Next decrypt the cipher text using initial parameters chosen by Sender to encrypt the message.
Step3: Finally the receiver got the original message.

Conclusion

This technique combines the features of both cryptography and Steganography which will provide a higher level of security. The aim of the method is to reduce the cipher modification attack in internet. Two level data encryption provide increased strength. In transferring secret two keys are used one is for data encryption and second is for Steganography. The chaotic key is generated by sender or receiver and share among themselves. Where the Stego-key is generated by sender. This method is more secure than any individual method.

Reference

X. Zhang and S. Wang (Jan. 2005), Steganography using multiple-base notational system and human vision sensitivity, IEEE Signal Process. Lett., vol.12, no. I, pp. 67-70.

S. M. MasudKarim, Md. SaifurRahman, Md. Ismail Hossain, A New Approach for LSB Based Image Steganographyusing Secret Key987-161284-908-9/11/2011 IEEE

M. Hossain, S.A. Haque, F. Sharmin (December 2009), Variable RateSteganography in Gray Scale Digital Images Using Neighborhood Pixel Information, Proceedings of 200912th International Conference on Computer and Information Technology (ICCIT 2009) 21-23, Dhaka, Bangladesh.

R Praveen Kumar, V Hemanth, MShareef (2013), Securing Information Using Sterganoraphy, International Conference on Circuits, Power and Computing Technologies [ICCPCT-2013]

Ahaiwe J. Document Security within Institutions Using Image Steganography Technique, International Journal of Science and Research (IJSR)

Bergé, Pierre; Pomeau, Yves; Vidal, Christian (1984). *Order within Chaos: Towards a Deterministic Approach to Turbulence*. New York: John Wiley & Sons. ISBN 978-0-471-84967-4.

Prof. LjupcoKocarev,Dr.ShiguoLian,Chaos-Based Cryptography Theory, Algorithms and Applications,springer ISBN 978-3-642-20541-5