International Journal For Technological Research In Engineering
Volume 5, Issue 2, October-2017 *ISSN (Online): 2347 - 4718*

ENHANCE SECURITY OF THE MESSAGE BY STEGNOGRAPHY USING LSB AND HASH FUNCTION AND PROVIDE AUTHENTICATION (DUAL SECURITY OF MESSAGE)

Suman Kumari[1], Gori Snakar[2], Surendra Singh Chauhan[3]
[1]M. Tech Scholar (CSE), [2,3]Assistant Professor (CSE)
[1,2]SMEC, Neemrana, Rajasthan, [3]JIET Jodhpur, Rajasthan

ABSTRACT: In the last some decades in digital world, communication of information is very hazardous in the charisma of the third party on internet. There cryptography is used to safe and sound information in order to offer the solitude for the planned sender as well as receiver by managing the message with the Symmetric and Asymmetric key cryptography. The purpose of this work is to insert a logo to authentication and, securing the message by stegnography technique, using the LSB technique and then applies hash function on all to provide security. In Ist stage at sender end, existing stegnography customized by inserting the logo for authentication, so that a sender cannot refuse about sending the message, LSB is used for insert the message in the pixels, and then hash function is used for security purpose. In IInd stage at receiver end first calculate hash function to check that, is there is attack on that message. If the hash value is equal as receive hash then there is no attack on that message, else reject that image.

Keywords: Plaintext, Cipher text, Encryption Process, Decryption process, LSB, MD5, Hash Function

I. INTRODUCTION

1.1 Introduction about Stegnography

The most punctual recordings of Steganography were by the Greek student of history Herodotus in his narratives known as "Histories" and go back to around 440 BC. Herodotus recorded two stories of Steganographic systems amid this time in Greece. The initially expressed that King Darius of Susa shaved the head of one of his detainees and composed a mystery message on his scalp. At the point when the detainee's hair developed back, he was sent to the Kings child in law Aristogoras in Miletus undetected. The second story additionally originated from Herodotus, which guarantees that an officer named Demeratus expected to make an impression on Sparta that Xerxes proposed to attack Greece. In those days, the composition medium was content composed on wax-secured tablets. Demeratus expelled the wax from the tablet, composed the mystery message on the basic wood, and recuperated the tablet with wax to make it show up as a clear tablet lastly sent the archive without being detected. Romans utilized imperceptible inks, which depended on regular substances, for example, organic products squeezes and drain. This was refined by warming the shrouded content, along these lines uncovering its substance. Imperceptible inks have turned out to be a great deal more progressed are still in restricted utilize today.

Amid the fifteenth and sixteenth hundreds of years, numerous essayists including Johannes Trithemius (writer of Steganographia) and Gaspari Schotti (writer or Steganographica) composed on Steganagraphic strategies, for example, coding systems for content, imperceptible inks, and joining shrouded messages in music. Somewhere around 1883 and 1907, further improvement can be ascribed to the distributions of Auguste Kerckhoff (creator of Cryptographic Militaries) and Charles Briquet (creator of Les Filigranes). These books were for the most part about Cryptography, yet both can be ascribed to the establishment of some steganographic frameworks and all the more altogether too watermarking methods. Amid the seasons of WWI and WWII, noteworthy advances in Steganography occurred. Ideas, for example, invalid figures (taking the third letter from every word in a safe message to make a concealed message, and so forth), picture substitution and microdot (taking information, for example, pictures and diminishing it to the extent of a huge period on a bit of paper) were presented and held onto as extraordinary steganographic procedures. In the advanced universe of today, specifically 1992 to present, Steganography is being utilized everywhere throughout the world on PC frameworks. Numerous instruments and advancements have been made that exploit old steganographic procedures, for example, invalid figures, coding in pictures, sound, video and microdot. With the examination this point is presently getting we will see a ton of incredible applications for Steganography sooner rather than later.

1.1.1 Method of Steganography

There are an extensive number of steganographic techniques that the vast majority of us are acquainted with (particularly in the event that you watch a considerable measure of spy motion pictures!), running from imperceptible ink and microdots to discharging a concealed message in the second letter of every expression of an expansive group of content and spread range radio correspondence. With PCs and systems, there are numerous different methods for concealing data, for example,

• Covert channels (e.g., Loki and some conveyed dissent of-administration instruments utilize the internet Control Message Protocol, or ICMP, as the interchanges channel between the "terrible person" and a traded off framework)
• Hidden content inside Web pages
• Hiding records "on display" (e.g., what better place to "conceal" a document than with an imperative sounding

name in the c:\winnt\system32 catalogue?)
• Null figures (e.g., utilizing the main letter of every word to shape a concealed message in a generally harmless content)

1.1.2 Problem Defination
Most of the data which one is run in the form of encryption have always, the big need of data security. To make the data security in steganography the concept is more secure than the normal cipher techniques where the data can be hidden behind the image if any intruder try to break the security the whole message will be lost but as the technology become famous the misuse of this technique becomes broad so to make the security there are many researchers generate the opacity and through the password and encryption. The watermark images are generated and the data is hidden behind the image but these securities are not sufficient, so to make the security stronger we generate the additional functions for it.

II. PROPOSED WORK

The proposed work consists of the following two steps:
This work is dividing in five steps at sender side and five steps at receiver side.
At sender side take an image, insert a water mark in it for authentication, insert the message in it calculate the hash value H1 of it and sent it to receiver.
At receiver end take that received image, divide it into pixels, calculate hash value H2, if H1 is not equal to H2 then discard that image else retrieve the message from the image.

III. ALGORITHM

At the sender end
Step 1: Take a logo (or text) L1, It is use for authentication. Change its Opacity. Let now it is logo L2.
Step 2: Take a Picture or Image P1. Insert watermark logo in it. Break this image into in pixels. Now this Image P1 break into a set of 24 bits that represent the color Red, Green and Blue (each color has 8 bits).
Step 3: Take a Message M1. Convert this message into the bits.
Step 4:

- Take these bits of message one by one.
- Take a single pixel of 24 bits (8 bits for Red color, 8 bits for Green color and 8 bits for Blue color).
- Take the Least Significant Bit of the colors.
- Change these bits by the bit of message.
- Like this there are we can insert 3 bit of Message M1 into a single pixel of (set of 24 bits).
- So this image is converting into image P2. (There are only minor changes Image P1. It cannot identify by the other person). Calculate hash value H1 of image P2 by MD5.

Step 5: Now there is a logo (or text) L2, Message M1 in Image P2 as a single unit. Send this Image P2 the receiver end.
At the receiver end
Step 1:

- Take the image P2. Calculate hash value H2 of image P2 by MD5.
- If H1=H2 then follow these steps otherwise discard the image P2.

Step 2:

- Change the opacity of the Logo L2 so it is converted back into the Logo L1. By this we can check the authenticity of the sender.

Step 3:

- Now take the image P2 break this image into the pixels.
- A single pixel is of 24 bits is a set of three colors (8 bits for Red color, 8 bits for Green color and 8 bits of Blue color).
- Take the least bit of every color. This is set of three bit.
- Doing this until the last pixel.

Step 4:

- Arrange these bits in a series of the pixels.
- Convert these bits in bytes.

Step 5:

- Regenerate the message M1 by this technique.

IV. FLOW CHART

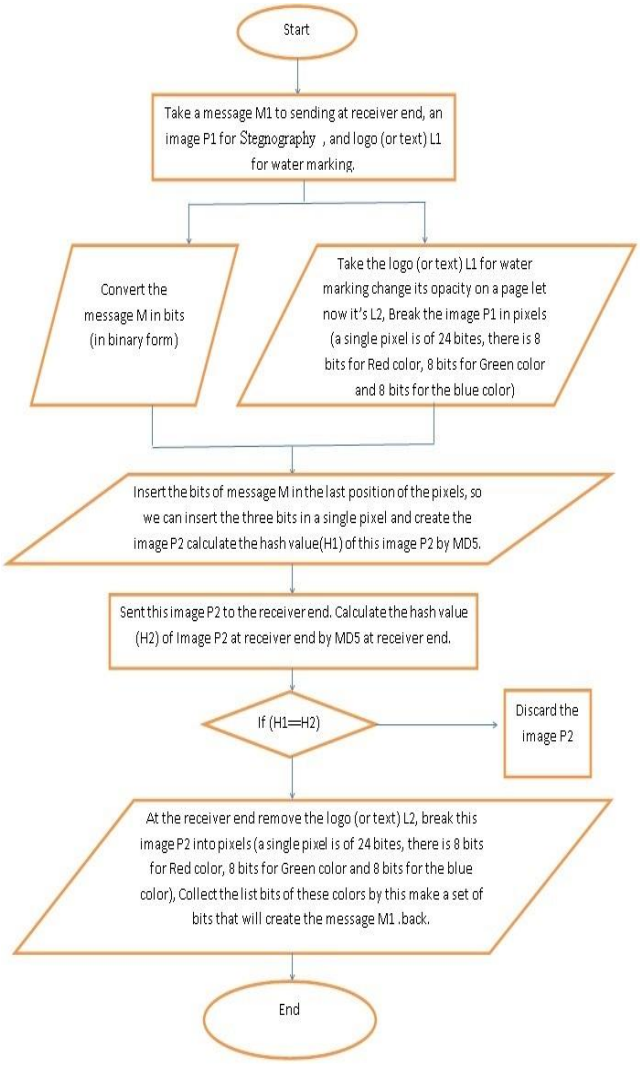

Fig. 4.1 Flow Chart of Algorithm

V. METHODOLOGY

At the sender end

Step 1:

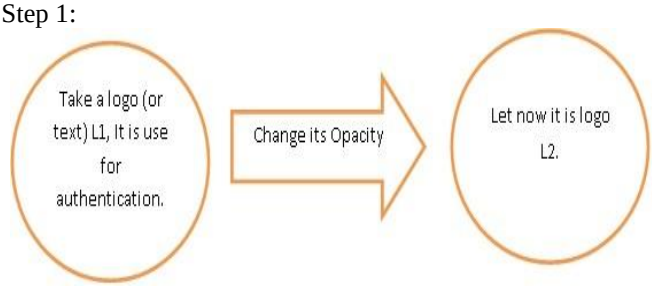

Fig. 5.1 Change the opacity of Logo L1.

Step 2:

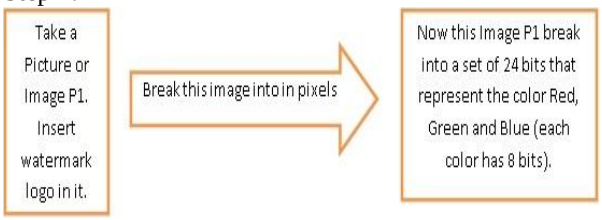

Fig. 5.2 Break an image into set of 24 bits.

Step 3:

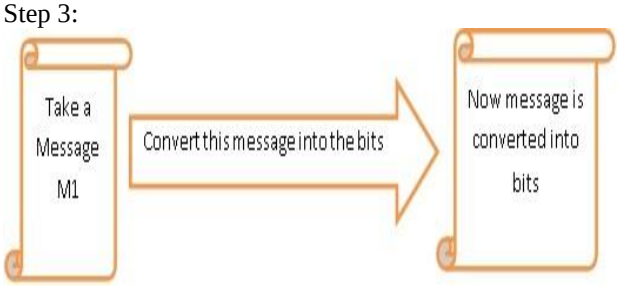

Fig. 5.3 Convert Message M1 into bits

Step 4:

- Take these bits of message one by one.
- Take a single pixel of 24 bits (8 bits for Red color, 8 bits for Green color and 8 bits for Blue color).
- Take the Least Significant Bit of the colors.
- Change these bits by the bit of message.
- Like this there are we can insert 3 bit of Message M1 into a single pixel of (set of 24 bits).

Step 5:

- So this image is converting into image P2. (There are only minor changes Image P1. It cannot identify by the other person). Calculate hash value H1 of image P2 by MD5.

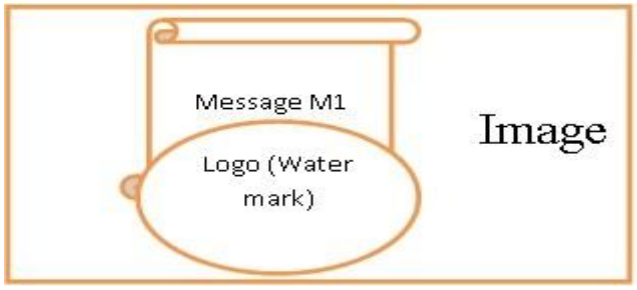

Fig. 5.4 Image P2 at sender end.

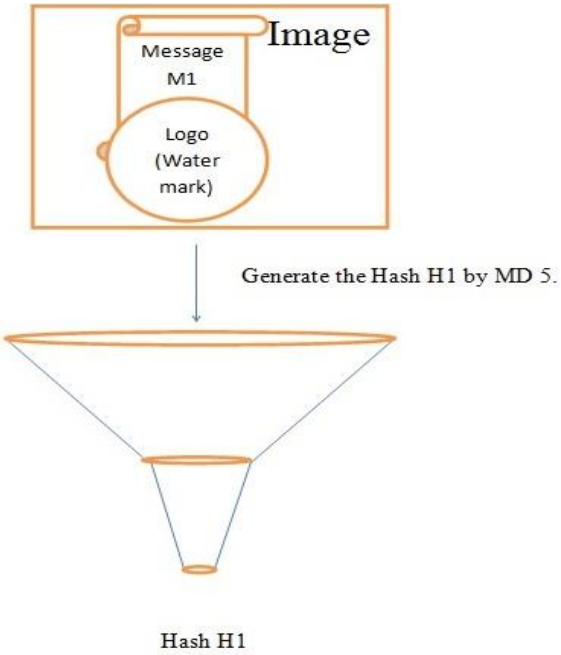

Fig. 5.5 Generate the Hash H1 of image P2 by the MD5 algorithm at sender end.

Now there is a logo (or text) L2, Message M1 in Image P2 as a single unit. Send this Image P2 the receiver end.

At the receiver end

Step 1:

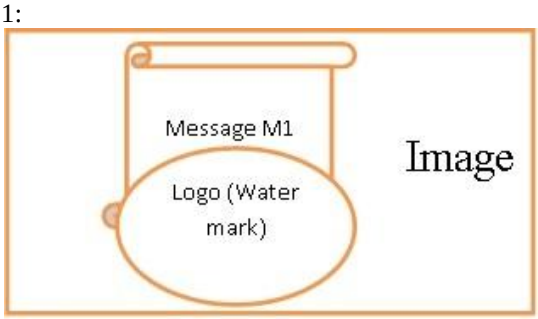

Fig. 5.6 Image P2 at receiver end

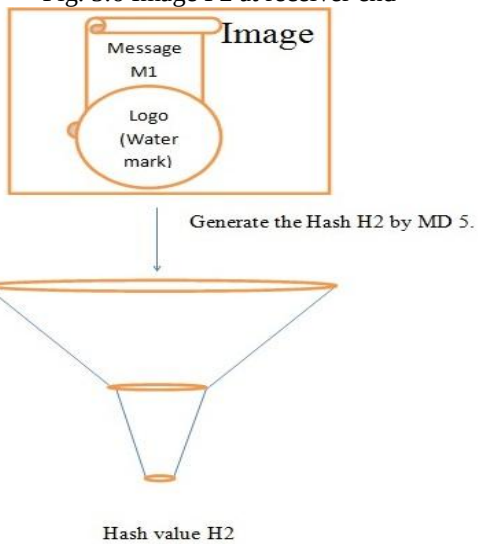

Fig. 5.7 Generate the Hash H2 of image P2 by the MD5 algorithm at receiver end.

Take the image P2. Calculate hash value H2 of image P2 by MD5.

- If H1=H2 then follow these steps otherwise discard the image P2.

Step 2:

- Change the opacity of the Logo L2 so it is converted back into the Logo L1. By this we can check the authenticity of the sender.

Step 3:

- Now take the image P2 break this image into the pixels.
- A single pixel is of 24 bits is a set of three colors (8 bits for Red color, 8 bits for Green color and 8 bits of Blue color).

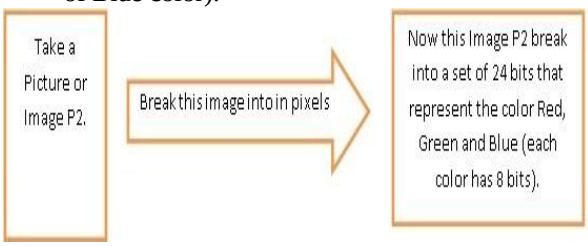

Fig. 5.8 Image P2 break into pixels

Take the least bit of every color. This is set of three bit.

- Doing this until the last pixel.

Step 4:

- Arrange these bits in a series of the pixels.
- Convert these bits in bytes.

Step 5:

- Regenerate the message M1 by this technique.

Fig. 5.9 Regenerate the Message M1 by the Least bits of pixels.

VI. EXAMPLE AND SNAPSHOT

Let an example there are an image with water mark of 14 pixels of different colors and our aim is to insert this message into the pixels at the sender end and retrieve that pixel at the receiver end.

Here message is SUMAN and its binary equivalent code is= 01010011 01010101 01001101 01000001 01001110

First of all convert image into pixels, take RGB values of all pixels convert these RGB values in the binary value, and convert the message into the binary values. Now insert single bit of message into the color binary value in last position as like this 3 bit of message can insert in the single pixel. Calculate the Hash value H1 of this image.

Now convert these color binary value into the RGB value. Check the Delta E value of these pixels for checking the color difference in the pixel.

The RGB code of these pixels is given in this table, we convert these RGB value in the binary equivalent code of eight bits. Now I insert the message bits in the least position of binary code of every color in every pixel. Like this there I insert three bits in a pixel and in this example there are 40 bits of message and I used the 14 pixels it is easily inserted in these pixels. Now check that how many changes occur in a particular pixel and calculate the Delta E value for change in color before and after insert the message.

TABLE 6.1

Sr. No.	Colour Name	RGB Code	Binary Equivalent	**Insert last significant bit is**	**After Insertion the new binary equivalent is**	**Change in bits**	**RGB code after insertion of message bits**	Delta E Value for RGB value for before and after insertion the message
1	Black	0, 0, 0	00000000 00000000 00000000	**0** **1** **0**	00000000 00000001 00000000	1	0, 1, 0	0.53
2	White	255, 255, 255	11111111 11111111 11111111	**1** **0** **0**	11111111 11111110 11111110	2	255, 254, 254	0.4425
3	Red	255, 0, 0	11111111 00000000 00000000	**1** **1** **0**	11111111 00000001 00000000	1	255, 1, 0	0.0727
4	Green	0, 255, 0	00000000 11111111 00000000	**1** **0** **1**	00000001 11111110 00000001	3	1, 254, 1	0.5331
5	Blue	0, 0, 255	00000000 00000000 11111111	**0** **1** **0**	00000000 00000001 11111110	2	0, 1, 254	0.5665
6	Gray	128, 128, 128	10000000 10000000 10000000	**1** **0** **1**	10000001 10000000 10000001	2	129, 128, 129	0.7123
7	Cyan	0, 255, 255	00000000 11111111 11111111	**0** **0** **1**	00000000 11111110 11111111	1	0, 254, 255	0.6757
8	Magenta	255, 0, 255	11111111 00000000 11111111	**1** **0** **1**	11111111 00000000 11111111	0	255, 0, 255	0
9	Yellow	255, 255, 0	11111111 11111111 00000000	**0** **1** **0**	11111110 11111111 00000000	1	254, 255, 0	0.4269
10	White	255, 255, 255	11111111 11111111 11111111	**0** **0** **0**	11111110 11111110 11111110	3	254, 254, 254	0.3451
11	Red	255, 0, 0	11111111 00000000 00000000	**0** **1** **0**	11111110 00000001 00000000	2	254, 1, 0	0.4054
12	Green	0, 255, 0	00000000 11111111 00000000	**1** **0** **0**	00000001 11111110 00000000	2	1, 254, 0	0.4714
13	Blue	0, 0,	00000000 00000000	**1** **1**	00000001 00000001	2	1, 1,	0.1909

By the table 6.1 we can see that the value of Delta E is less than 1, in this example so there is no color difference in the image.

At the receiver end we can break the received image into the pixels, calculate the Hash value H2 at the receiver end if(H1=H2) then do these steps

take the RGB values of these pixels convert these values into the binary form. Take the Least bit of RGB from every pixels so that we can create the string of bits, break these string into set of 8 bits. Convert these bits into the characters.
Else reject that image.
By this method we can retrieve the message from an image securely.
There is value of Delta is less than one; by checking this value the result of this example is there is no such change in every pixel that it can create the variation in every pixel. This shows that we can easily insert the message by this technique.
The hash value of pixels by MD5 after insertion the message is this is hash H1;
3B00C36277D48F63A5515E7411624D6A
By SHA1 is;
54F280B41812B813DB3D6A59AA18D1384BB69D47
By SHA256
6AF161C6A486F05477DB3E71769FF54CDD136BD918EB
2077C7CD0287089AF24A
Send the image of pixels and hash value H1 at the receiver end.
Calculate the Hash value of received image at the receiver end.
The hash value H2 of pixels by MD5 at the receiver end by MD5 is;
3B00C36277D48F63A5515E7411624D6A
By SHA1 is;
54F280B41812B813DB3D6A59AA18D1384BB69D47
By SHA256
6AF161C6A486F05477DB3E71769FF54CDD136BD918EB
2077C7CD0287089AF24A

There is no change in the H1 and H2 so there is no attack on this image. We can retrieve the message from this image
Convert this image into pixels; take the binary of these pixels. Take the least bit of every color and every pixel. Arrange these bits in a series of eight pixels. Now convert these bytes into the Message.
01010011 01010101 01001101 01000001 01001110 = SUMAN

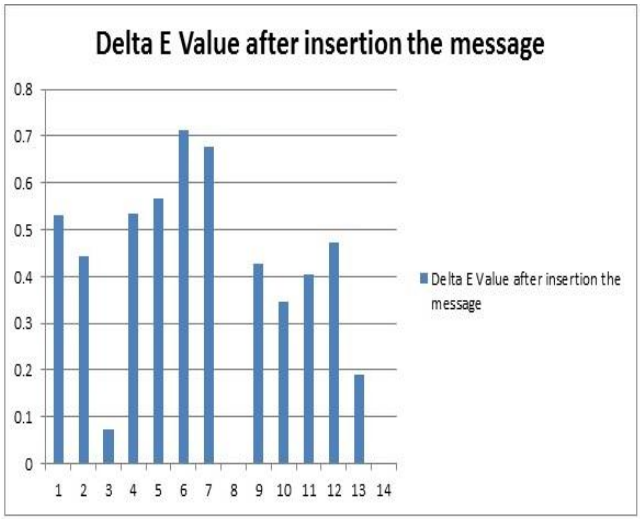

Fig. 6.1 Graph for Delta E value in the pixels.

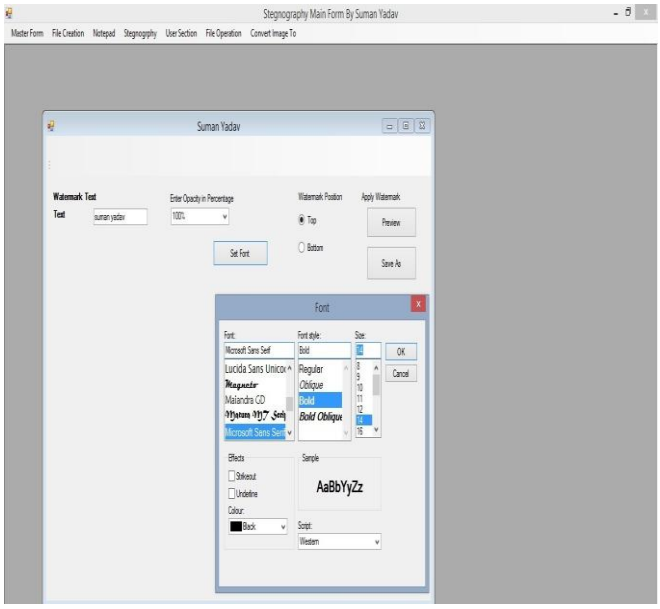

Fig. 6.2 Watermark the bitmap image.

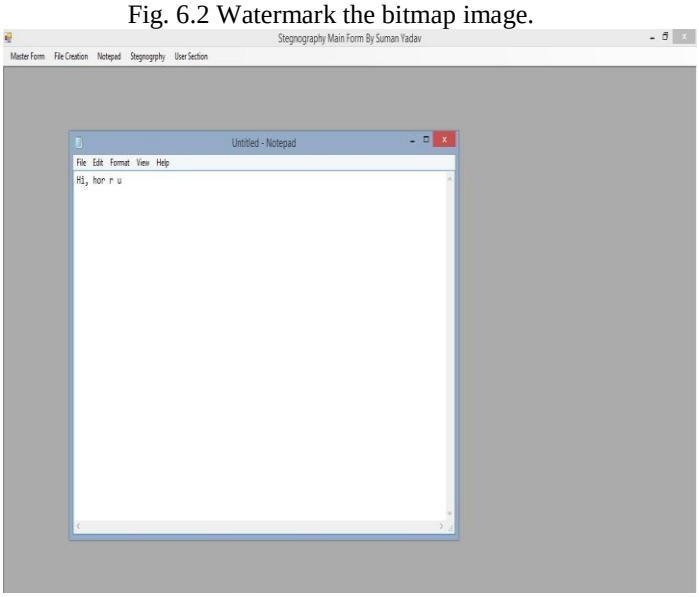

Fig. 6.3. Create text to embed in the image.

Fig.6.4 Compare the hash value by MD5.

VII. CONCLUSION

From the above proposed algorithm Data Integrity is upgraded utilizing a Digital Watermarking and Digital Signature. In this information is inserted with the picture and sent over the system where its respectability is checked by the examination of hash estimation of the first information or picture. The proposed work gives secure mystery correspondence among sender and collector, it guarantees that inserted information stays untouched and recoverable, watermarks the picture with fantastic visual quality without bringing on a discernible loss of value. It is valuable for copyright possession declaration purposes. The information which is covered up can't be effectively evacuated and oppose normal picture control strategies.

REFERENCES

[1] SANS Security Essentials, (volume 1.4, chapter 4) Encryption and Exploits, 2001.

[2] Erfaneh Noorouzi, Amir Reza Est Akhrian Haghighi, Farzad Peyravi, Ahmad Khadem Zadeh, "A New Digital Signature Algorithm", 2009 International Conference on Machine Learning and Computing, Volume.3, IACSIT Press, Singapore, 2011.

[3] Swarnendu Mukherjee, Debashis Ganguly and Somnath Naskar, "A New Generation Cryptographic Technique", International Journal of Computer Theory and Engineering, Volume. 1, No. 3, August, 2009.

[4] Sameer Hasan Al-Bakri, M. L. Mat Kiah, A. A. Zaidan, B. B. Zaidan and Gazi Mahabubul Alam, "Securing peer-to-peer mobile communications using public key cryptography: New security strategy", International Journal of the Physical Sciences Volume. 6, No.4, pp. 930-938, 18 February, 2011.

[5] Challa Narasimham, Jayaram Pradhan, "Evaluation of Performance Characteristics of Cryptosystem Using Text Files", Journal of Theoretical and Applied Information Technology 2008.

[6] Bruce Schneier, "Security Pitfalls in Cryptography", Counterpane Systems, 1998.

[7] Prashant Kumar Koshta, Dr. Shailendra Singh Thakur, "A Novel Authenticity of an Image Using Visual Cryptography", International Journal of Computer Science and Network, Volume 1, Issue 2, April 2012.

[8] J.M.Gnanasekar, V.Ramachandran, "Distributed Cryptographic Key Management for Mobile Agent Security", International Journal of Recent Trends in Engineering, Volume. 1, No.1, May 2009.

[9] Fabian Monrose, Michael K. Reiter, Qi Li , Susanne Wetzel, "Cryptographic Key Generation from Voice", In Proceedings of the 2001 IEEE Symposium on Security and Privacy, May 2001.

[10] Mina Mishra & V. H. Mankar, "Review on Chaotic Sequences Based Cryptography and Cryptanalysis", International Journal of Electronics Engineering, Volume.3, No.2, pp. 189– 194, 2011.

[11] Katanyoo Klubsuwan, Surasak Mungsing, "Digital data security and hiding on virtual reality video 3D GIS", International Journal of Management Science and Engineering Management Volume. 4, No. 3, pp. 163-176,2009.

[12] Prof. Samir Kumar Bandyopadhyay, Sarthak Parui, "A Method for Public Key Method of Steganography", International Journal of Computer Applications (0975 – 8887) Volume 6, No.3, September 2010.

[13] Yu-Chee Tseng, Member, IEEE, Yu-Yuan Chen, and Hsiang-Kuang Pan, " A Secure Data Hiding Scheme for Binary Images", IEEE Transaction on Communication, Volume. 50, No. 8, pp 1227-1231, August 2002.

[14] G. J. Simmons, "The prisoners' problem and the subliminal channel, "in Proc. CRYPTO'83, pp. 51–67, 1983.

[15] .R. G. van Schyndel, A. Z. Tickle, and C. F. Osborne, "A digital watermark, "in Proc. IEEE Int. Conf. Image Processing, Volume. 2, pp.86–90, 1994..

[16] E. Franz et al., "Computer-based steganography," in Information Hiding, Springer Lecture Notes in Computer Science, Volume. 1174, pp. 7–21, 1996.

[17] D. Gruhl and W. Bender, "Information hiding to foil the casual counterfeiter, "in Proc. Workshop Information Hiding, IH'98, Portland, OR,Apr. 1998.

[18] R. J. Anderson, "Stretching the limits of steganography," in Information Hiding, Springer Lecture Notes in Computer Science, Volume. 1174, pp. 39–48, 1996.

[19] R. J. Anderson and F. A. P. Petit colas, "On the limits of steganography,"IEEE J. Select. Areas Common. Volume. 16, pp. 474–481, May 1998.

[20] W. Bender, D. Gruhl, N. Morimoto, and A. Lu, "Techniques for data hiding," IBM Syst. J., Volume. 35, No. 3–4, Feb. 1996.

[21] S. Katzenbeisser and F. A. P. Petitcolas, Information Hiding Techniques for Steganography and Digital Watermarking. Norwood, MA: ArtechHouse, 2000.

[22] http://webappsuccess.com/testing-and-deployment.html

[23] https://en.wikipedia.org/wiki/Cryptography

[24] http://colormine.org/delta-e-calculator/

[25] http://onlinemd5.com/

