IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol.-7, Issue-5, May 2019 E-ISSN: 2347-2693

A Proposed Algorithm to Reinforce the Security of Steganograpy in conjunction with Cryptography to Assure Privacy and Integrity of the Communication by using Native OS Batch Programming Technique

J. Sebastian Nixon[1*], Mesele Gebre Awgichew [2], Akalu Assefa Afaro[3], Fisaha Solomon[4], Paulos Bogale Wada[5], Fevan Tafari[6]

[1-6] Department of Computer Science & IT, School of Informatics, Wolaita Sodo University, Sodo, Ethiopia

[*]*Corresponding Author: dr.nixon14@gmail.com , Tel.: +251-949017364*

DOI: https://doi.org/10.26438/ijcse/v7i5.18051819 | Available online at: www.ijcseonline.org

Accepted: 25/May/2019, Published: 31/May/2019

Abstract— This present digital world is made up of both secure and insecure data communication. So, it is imperative to secure our data from adversaries. To protect our data while communications, lots of mechanisms are there, but still some vulnerability exists. Steganography is one of the security techniques where we can hide our data inside different mediums like video, image, and audio files. There are also some vulnerable exists. Since it hides the data in the least significant bit [LSB], attacks could be performed easily if the adversary knows some data concealed in the object. In this research, we proposed a new multi-layered security algorithm to reinforce security of steganography along with Cryptography to ensure the privacy and integrity of the data by using the native Operating System batch programming. In this Algorithm, we used several encryption techniques with many undercover keys, did ASCII to Binary conversions, exchange of symbols and Hash function. This Proposed mechanism reinforced the security of steganography by ensuring confidentiality and integrity. It would be a big challenge for the adversary. In this research, we used our coding to perform Steganography and to produce the hash code.

Keywords— *Steganography, Cryptography, Hash function, Plain Text, Cipher Text, LSB, Pixel, Undercover key, Operating System [OS], SHA512.*

I. INTRODUCTION

In this digital world, new technologies are growing day by day. Even though, lots of advantages and disadvantages are there we couldn't away from those technologies. Whenever we are communicating the digital world, security is the prime concern. Normally, data security assures the privacy of transmitted data. For secure transmission of data, there are different kinds of encryption algorithms and techniques are available for different kinds of data such as texts, audio, video, picture, etc. The safety of communication depends on the capability of the algorithm used and the features of the transmitting data [1]. Steganography & Cryptography are widely used security techniques to provide data and communication security. Steganography is a way of hiding critical information inside some other data format. So, it would not turn the adversary's attention on the data. Cryptography is mainly handling encrypting the data using different techniques.

Nowadays, we are hiding the data inside some other data electronically. However, the technique of steganography is not a modern development. Even, before the computer comes, steganography use has occurred in the military for decades. In military, they tattooed the undercover message on a soldier's shaved head then after grown the hair he met the designated recipient again he would shave his hair to reveal the undercover message.

A. STEGANOGRAPHY

The short definition of Steganography is data conceal in data. It is derived from the Greek words "steganosgraphein" means "covered writing". The meaning of "steganos" is "covered/protected" and "graphein" is "writing". It is a technique of hiding the data inside some other format of data such as a text file that could be concealed into a video file, audio file or image file without affecting their original quality or look [2]. The hidden data may be any digital format it may be coded text, normal text. The main motive of this technique is to covert communication to hide data from the adversary. For example, conceal name and copyrights of the owner of an image as protection against violation of the copyright. Steganography conceals the data inside other data by using the destination file's unused bits called LSB. If we did any alteration in these bits will not affect the original data or appearance of the picture. Based on the application needs, there is various steganography techniques available. All of

these have some advantages and disadvantages [3][4]. The following Figure 1 describes the mechanism of steganography.

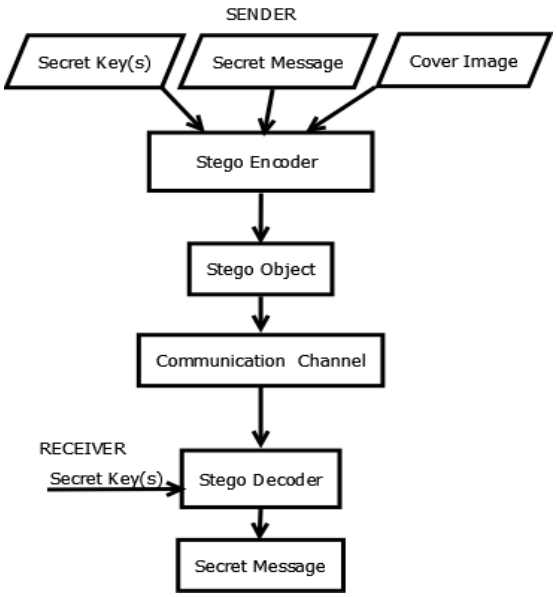

Figure 1. Process of Steganography

Steganography uses 4 types of digital file formats. The following Figure 2 describes the four main file format categories of steganography.

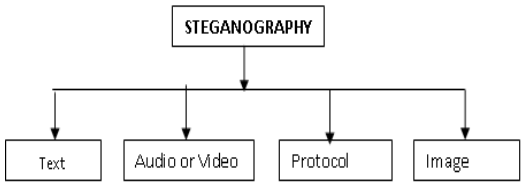

Figure 2. The Main File Formats of Steganography

B. CRYPTOGRAPHY

Cryptography means the Art of undercover writing. It is more essential when we send our data using any incredible medium such as networks, especially the Internet. The following are the 5 main functions of cryptography.

1. Privacy / Confidentiality: - Assuring that only authorized receiver read that message.
2. Authentication: - The mechanism to prove one's identity.
3. Integrity: - Ensuring that the content of the message has not been changed during the communication.
4. Non-repudiation: - A technique that ensures that the source truly sent the message
5. Key Exchange: - The way by which undercover keys are shared between both parties.

The following Figure 3 is the example of encryption and decryption. If the source and destination used the same key for encoding and decoding means, called symmetric cryptography. If both parties used a different key mean, called asymmetric cryptography.

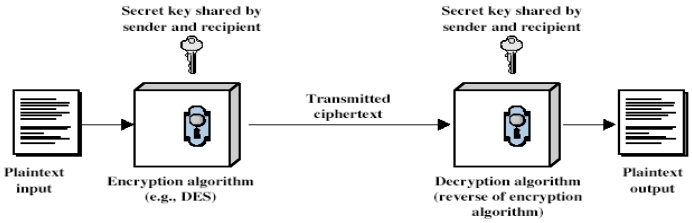

Figure 3. Encoding and Decoding Process

C. CRYPTOGRAPHIC HASH FUNCTION

Hash functions are the technique to ensure that the data sent over the network has not been altered. The hash function used a fixed length hash value that is calculated by the original text message or any other input. Since more sensitive data are stored on computing devices and transmitted over the Internet, it is more imperative to assure data security and safety. The following figure 4 describes the process of hash function. The hash function can be written as **h = H(M)** where, the hash function is H, the message is M then the hash value is h**.**

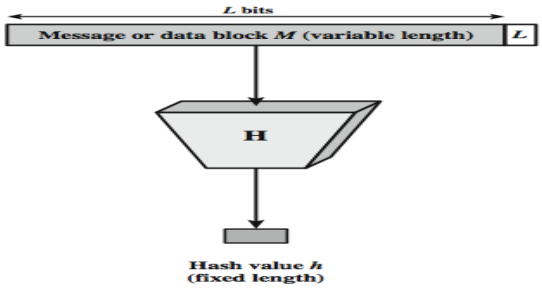

Figure 4. Hash Process

A cryptographic hash function is used in order to transform a large block of a string of data to a small block of data. This event is a one-way process so, the recreation of that actual data is impossible and it is also difficult to find two strings which may be transformed to the same hash. The following are some examples of Hash Algorithms.

MD5, SHA-256, SHA-384,SHA-512, GOST, HAVAL, RIPEMD, RIPEMD-128/256, RIPEMD- 60, RIPEMD-320, Tiger(2), Whirlpool, RTR0 …etc.

The following figure 5 explains how the message integrity is assured using the Hash function.

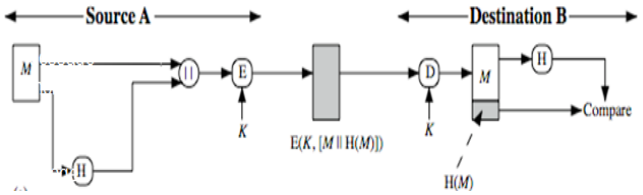

Figure 5. Encoding and Decoding with Hash

We organized this paper into five Sections: Section I contain the introduction to steganography, Cryptography and Hash Function. In Section II, we discussed about the related works already done in the Security of Steganography. The section III is Methodology in this; we have explained our experiment for normal steganography, and our proposed algorithm1. Using our proposed algorithm-11 we have done steganography and hash comparison of sender side and receiver side. Section IV, Results and Discussions, in this section we discussed briefly about the proposed Algorithm-2 which is the extended version of the proposed Algorithm-1 we discussed in Section III. Using our proposed Algorithm-2 we have done an experiment for steganography and hash, and obtained results. Finally, we proved that how our proposed Algorithm-2 assured the integrity and confidentiality of steganography also showed how to check the compromised stegno by using hash. Section V, Conclusion and Future Scope, in this we concluded our proposed secure algorithm for steganography to ensure confidentiality and integrity, and given some suggestion for future research works related to this field.

II. RELATED WORK

In [5], Indrayani et al. proposed a method to boost the security of steganography using MP3 audio format. In that, the MP3 audio file was used as a cover medium, AES algorithm was used to encrypt the undercover message and the MD5 one-way hash function was used to process the undercover key.

Uzair Nisar1 et al. [6] proposed an Email System security by using steganography with cryptography. That was implemented by developing the following 3 tools: 1. Hidden data inside picture 2. Password encryption and 3. Passed it over the internet. This system provided the secure communication of data.

In [7], Robbi Rahim et al. proposed a method to achieve data privacy using steganography and encryption technique using the base64 algorithm and EOF technique.

Masud et al. [8] proposed a technique that used the private key to utilize LSB. In that, the cover image was split into 3 matrices as Red, Green, and Blue, and the undercover key is generated using one-dimensional array of bit streams. Then, in order to replace the undercover data into Green or Blue matrices, the converted undercover key and the Red matrix are used. Using the reverse process the undercover data was extracted.

In [9], Hajduk et al. proposed an image Steganography tool by using LDWT and QR coding. Reinforced the security by using AES ciphering of QR code.

In [10], The RSA encryption algorithm was used to encrypt the undercover message and it was hidden in the LSB of the stego image to provide data security.

In [11], Puja Mahajan et al. Proposed a reinforced FPGA based X-BOX mapping for a picture utilizing steganography strategy to embed various aspects X-Box system was utilized. In that, mapping strategy X-box has given the good qualities and those qualities were put in an irregular way to reinforce the security.

In [12], Dey et al. Introduced a data-hiding algorithm, which encrypted the undercover message and then hidden the cipher text in a QR Code.

In [13], Sharma et al. presented another implanting calculation for QR Code Image Steganography and message hiding, which is based on a reinforced RSA algorithm and 3-discrete wavelet transform (DWT). In that, Firstly, the plain text was chosen then for RSA encryption 4 random numbers were selected. The cover image was taken and divided the RGB image of three planes: Red, Green, and Blue. To improve the security multiple color undercover images was embedded into a single cover image. Finally, the encrypted message was hidden in an embedded picture using the LSB.

In [14], K. Nandhini et al. proposed an algorithm for LSB data hiding and recovering. It was implemented in FPGA using pipelining and without pipelining techniques. That work addressed the computation delay reduction and throughput increase for the LSB based image steganography. The various modules of embedding and extraction were realized using Xilinx,VIRTEX-5 device using RTL compliant Verilog HDL code. The proposed algorithm was also validated using Modelsim simulation before the hardware implementation.

In [15], Hamed et al. proposed a two layer of security using hybrid method. In that, the undercover message was converted to DNA format, then it was encrypted by using the Playfair cipher. Finally, using LSB technique, it was embedded in DNA.

All the previous researchers used different techniques for encryption and steganography. Most assured confidentiality.

In this research, we proposed an Algorithm which reinforces the security of steganography and assured the privacy and integrity of the data.

III. METHODOLOGY

Cryptography and Steganography are not commonly exclusive. We could use steganography in conjunction with encryption in order to deliver an undercover message to an authorized receiver without drawing attention to the fact that a message was sent at all. In essence, using combination allows a layered defence for undercover communications.

A. SCENARIO – 1 NORMAL STEGANOGRAPHY

In this, we concealed the undercover data into the image directly by using our program. The following Figures 6 – 11 describe the Process. The following Tiger Image is taken for our experiment and the file name is **tiger.jpg**. The undercover message file name is **plaintext.txt,** the undercover message is ***attacked*** and **the** stego image file name is **TIGERSTEGNO.jpg** [in better understanding we used the file name **TIGERSTEGNO.jpg**]. Our batch programming files for steganography and hash are **STEGANO** and **HASH.**

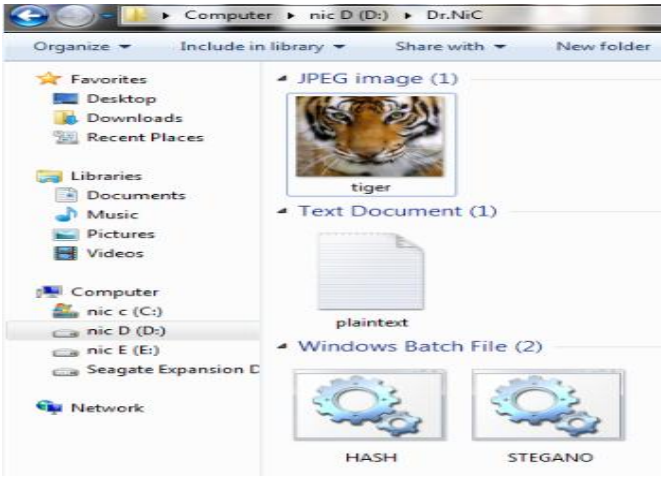

Figure 6. The content of the folder at the beginning

The following Figure 7. Shows the content of Plaintext File that is "**attacked**".

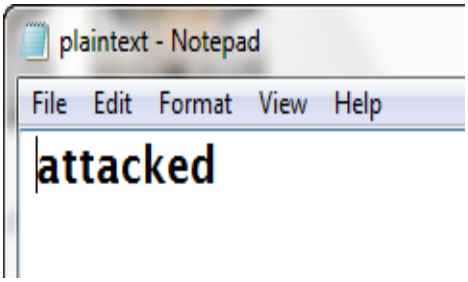

Figure 7. Content of Plaintext File

The following Figure 8. Describes the Execution of Steganography Coding.

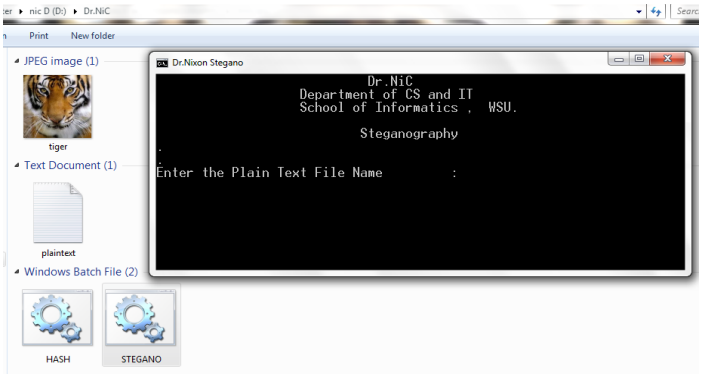

Figure 8. Execution of Steganography Coding

The following Figure 9. Describes the Successful Creation of Stego Image.

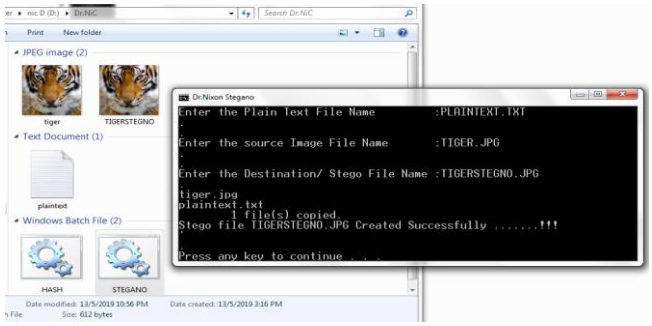

Figure 9. Successful Creation of Stego Image

The following Figure 10. Shows the Original and Stego Image.

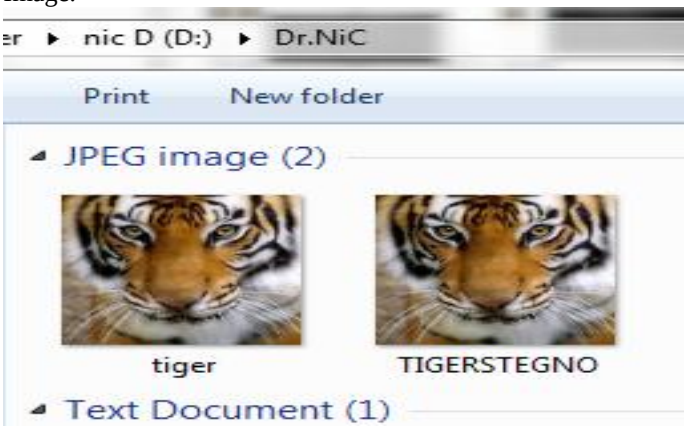

Figure 10. Comparison of Original and Stego Images

The following Figure 11. Shows the content of the Stego Image including the secret message

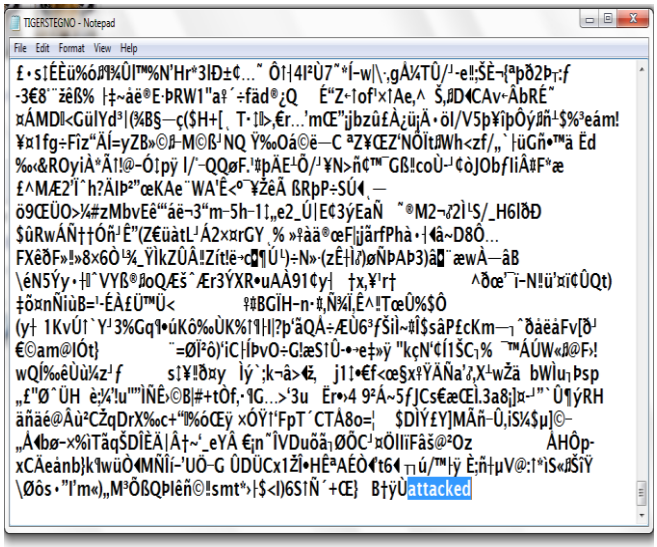

Figure 11. Undercover Message Concealed into Stego Image Successfully

As in Figure 11, if we hide the secret message inside an image directly, the attacker may open the image, and he could read the message. So, to improve the security of steganography we proposed the following encryption Algorithm.

B. PROPOSED ENCRYPTION ALGORITHM

We formed a new cipher called VOWELS Cipher as in Table 1. Where, we created 6 x 6 matrix surrounded by vowels and randomly filled with a-z and 0 – 9. In the top, and right we have written the vowels in ascending order and bottom and left in descending order. In this method, each character of plain text will be converted into 4 characters of cipher text by following the order of top, right, bottom, and left. For example, a plain text character "**x**" will be converted into "**UAEA**".

Table 1. Vowels Cipher

	A	E	I	O	U	A	
A	j	c	8	u	x	6	A
U	t	1	n	d	2	g	E
O	3	h	q	9	w	k	I
I	b	m	4	s	f	7	O
E	p	z	a	i	0	r	U
A	5	o	v	l	e	y	A
	A	U	O	I	E	A	

This proposed algorithm includes 2 steps. First, one is, using the matrix to encrypt the plain text, the second using undercover key [0-9] columnar transposition. The following is the proposed algorithm for encryption.

C. *The Proposed Algorithm -1 for Encryption [VOWELS Cipher Algorithm]*

Begin
1: Create a 6x6 matrix
2: Write a-z and 0-9 randomly into the matrix
3: Write the vowels top & right in ascending order , bottom & left in descending
4: Read Plain Text
5: Find length of plaintext ptlen= len(alltrim((plaintext))
6: For (I = 1, I <= ptlen)
7: Replace each character of plain text with 4 characters of cipher text in the order of top, right, bottom and left.
8: I = I+1 , Goto step 6:
9: Find cipher text Length : ctlen= ptlen * 4
10: Find the length of Undercover Key sklen= ctlen/ptlen
11: Generate Undercover key skey [range 0-9] where len(skey) <= sklen
12: Create a Matrix row(ptlen) x col(sklen)
13: Write the cipher text into the matrix Left to Right
14: Do columnar transposition in the ascending order of undercover key
15: For (I=1 , I <= sklen)
16: Write col(I)
17: I= I +1 ; Goto Step 15:
End

D. ENCRYPTION PROCESS BASED ON THE PROPOSED ALGORITHM

To perform steganography with both Cryptography, we used the following:

1. Plain Text
2. Encryption Technique(s)
3. Undercover Key(s)
4. Cipher Text
5. Cover Image
6. Our Tool to conceal the Undercover Message into the Cover Image
7. Hashing program to assure integrity

In this research, we didn't use any third party steganography tools; instead, we used our native OS Batch Programming. The following were the data we used for our experiment.

Our Plain Text : **attacked**

By applying the VOWELS Cipher we got the following cipher Text.

Cipher Text : **IUOE AEAU AEAU IUOE EAUA AIAO UAEA OEIU**

Undercover Key Generation using the proposed Algorithm

Step 1: Find length of plaintext
ptlen= len(alltrim((plaintext))= 8

Step 2: Find cipher text Length : ctlen= ptlen * 4 = 32

Step 3: Find the length of Undercover Key:
sklen= ctlen/ptlen = 32/8 = 4
Step 4: Generate Undercover key: skey [range 0-9]
where len(skey) <= 4

[Our Example Undercover key is skey= 2431]

Step 5: Create a Matrix
row (ptlen) x col(sklen) = 8 x 4= 32

Step 6:Fill the Cipher Text in the Matrix as shown in Table 2
Using our proposed algorithm, we got the secret key length=4, for the given plain text, and the matrix 8 x 4. For this experiment, we used the secret key **2431**. By using the secret key we filled the cipher text into the 8 x 4 matrix as in Table 2.

Cipher Text: **IUOE AEAU AEAU IUOE EAUA AIAO UAEA OEIU**

Table 2. Filled Cipher Text

2	4	3	1
I	U	O	E
A	E	A	U
A	E	A	U
I	U	O	E
E	A	U	A
A	I	A	O
U	A	E	A
O	E	I	U

After writing, the cipher text into the matrix, we have done the columnar transposition in ascending order of secret key as in the table 3.

Table 3. Columnar Transposition

1	2	3	4
E	I	O	U
U	A	A	E
U	A	A	E
E	I	O	U
A	E	U	A
O	A	A	I
A	U	E	A
U	O	I	E

Now the final cipher Text of our proposed Algorithm is: **EUUEAOAU IAAIEAUO OAAOUAEI UEEUAIAE**

The following Figure 12 is the Flowchart of our proposed encryption Algorithm1.

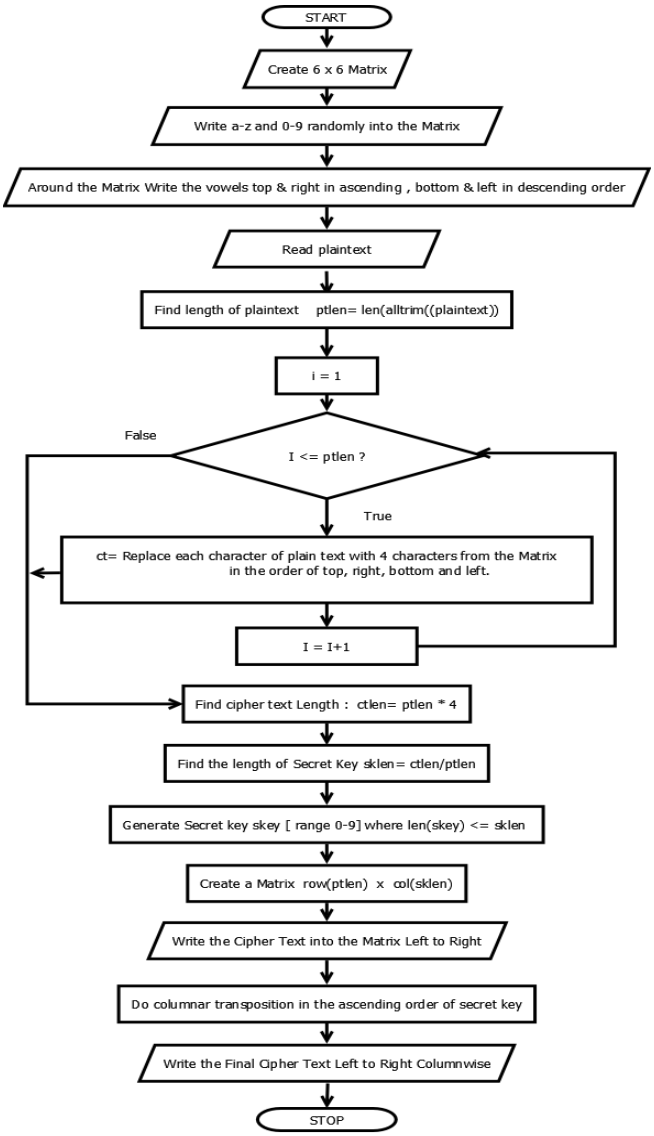

Figure 12. Flowchart for Vowels Cipher Algorithm

E. SCENARIO –2 SENDER SIDE STEAGANOGRAPHY+ CRYPTOGRAPHY [Assured CONFIDENTIALITY]

In this, we concealed the **Encrypted undercover message** into the image by using our coding. The following Figures 13 – 16 describe the Process.

The following Figure 13, shows the Content of the Folder in the beginning.

Figure13. Content of the Folder

The following Figure 14. Shows the final cipher text we obtained from our proposed encryption Algorithm for the given plain text.

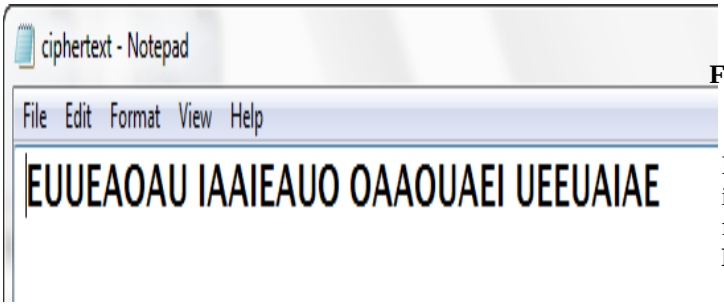

Figure 14. Encrypted Plain text

The following Figure 15, describes the concealing process of secret message.

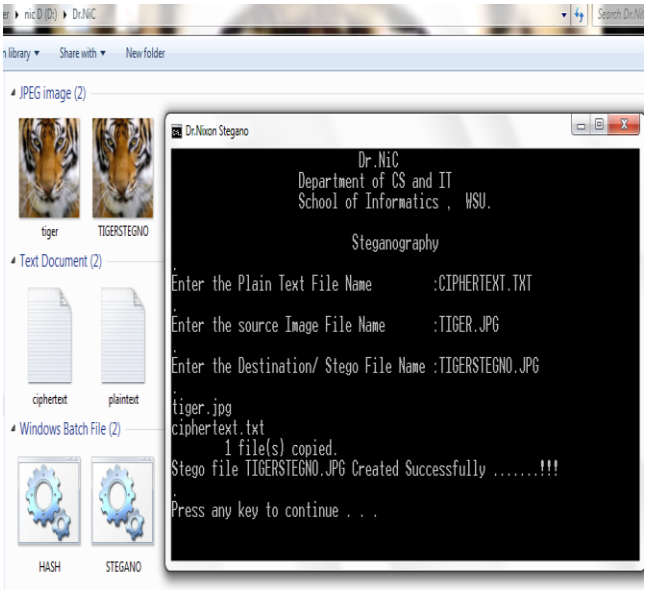

Figure 15. Concealing Process

The following Figure 16, shows the content of the stego image, including our encrypted secret message.

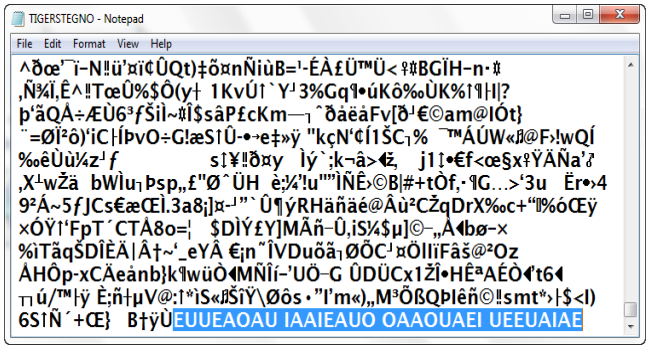

Figure 6. The Encrypted Plain Text concealed in Stego Image

F. SCENARIO –3 STEAGANOGRAPHY + CRYPTOGRAPHIC HASH FUNCTION [Assured INTEGRITY]

In this, we hashed the stego image using our coding by implementing SHA-512 and got the hash value. The following Figures 17 – 19 describe the **Sender Side Hashing** Process.

SENDER SIDE HASH PROCESS

The following Figure17, describes the execution of the Hash Program

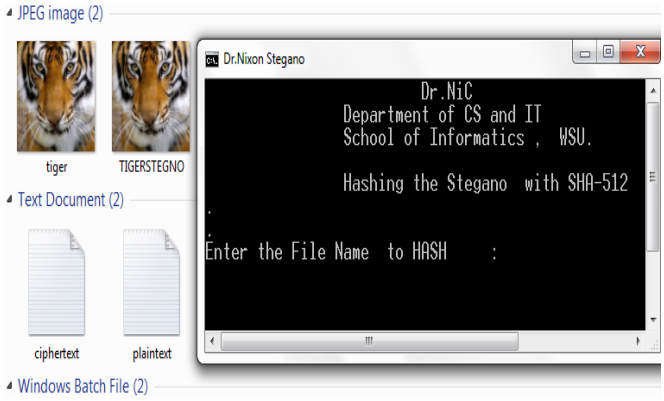

Figure17. Execution of the Hash Program

The following Figure18 describes the successful creation of the Hash.

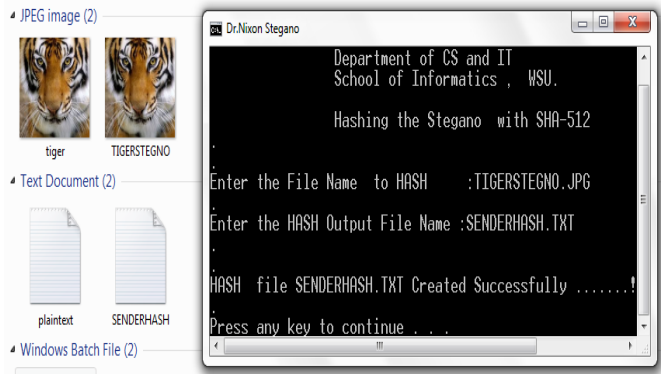

Figure18. Hashing Process

The following Figure19 shows the Hash Value of the stego image.

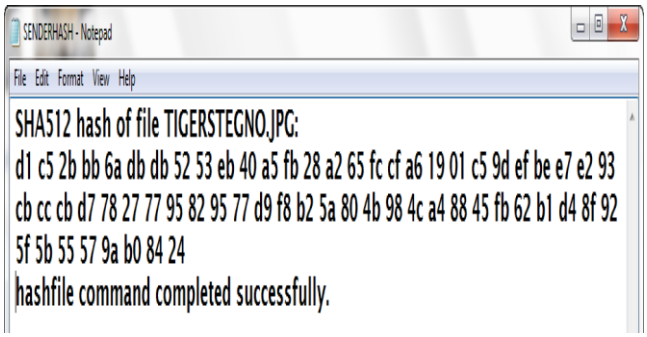

Figure19. Hash Value of the Stego Image

The following Figures 20 – 23 describe the Receiver Side Hash process and Hash value comparison with the sender's Hash value.

RECEIVER SIDE HASH PROCESS

The following Figure20 describes the successful creation of the Hash

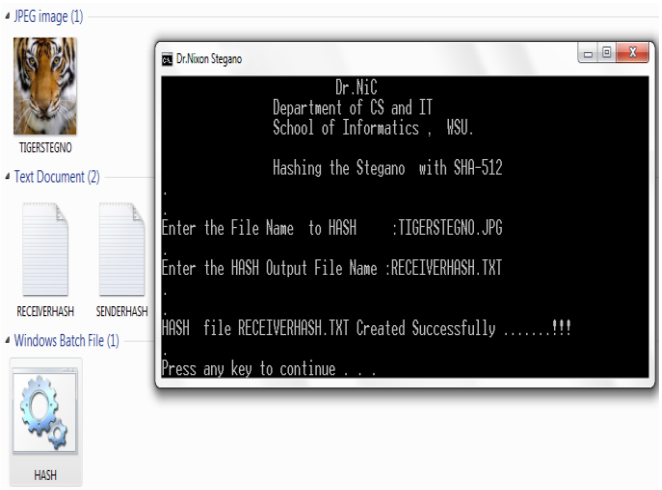

Figure20. Receiver Side Hashing Process

The following Figure 21 shows the Hash Value of the stego image.

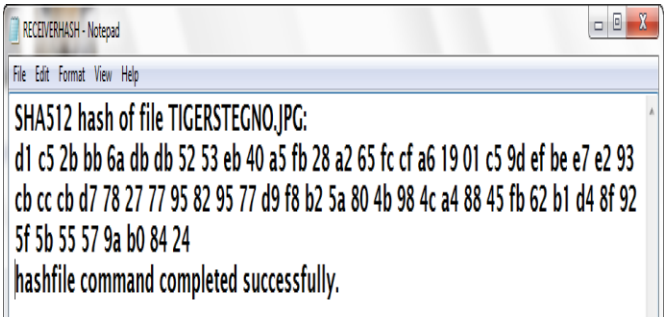

Figure 21. Receiver Side Hash Value

The figure 22 describe the comparison of Sender's Hash with Receiver's Hash value. Here, both the values are equal so the receiver can conclude that there was not any alteration happened on the stego image. It assured the integrity.

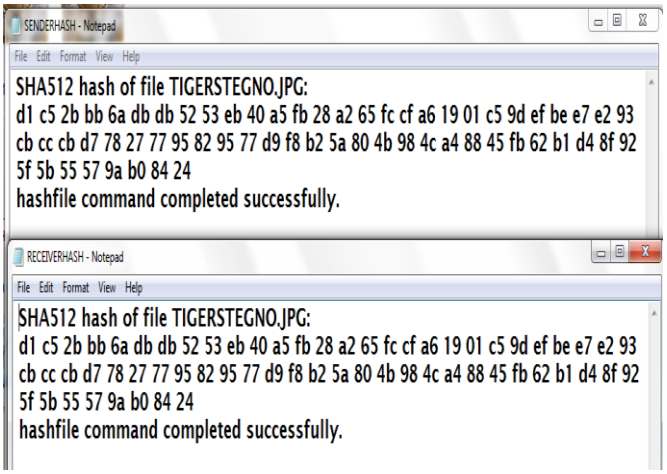

Figure22. Comparison of Sender's Hash with Receiver's Hash

The following Figure 23 is the example of a compromised stego image hash value. Suppose, if any, alteration or attack happened on the stego image, then how a receiver can identify it.

Assume that the following first window is the hash value of the sender and the second window is the hash value of the receiver's. Since both are differed different the receiver could understand that the stego image content was altered or attacked during the communication. So, the receiver can request the sender to resend the secret message.

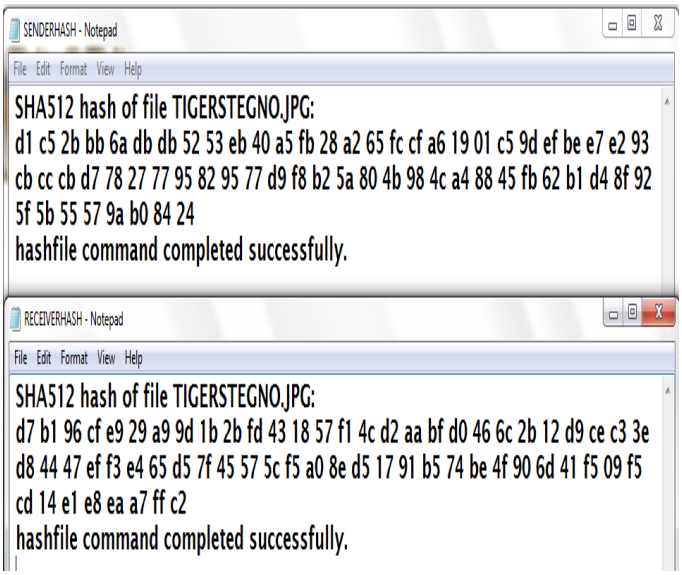

Figure23. Comparison of Sender's Side Hash with example compromised Stego Image Hash.

IV. RESULTS AND DISCUSSION

Cryptography provides security for the data from the adversary whereas steganography conceals the data in another data. So, even the adversary doesn't know whether the sender sent a message or not since the message was hidden inside some other file format. The main goal of this research is supposed, if the adversary came to know, some message is inside the image then how to protect the data. In this research, we used both techniques to reinforce the security of the steganography and assured privacy and integrity of the communication. In Section III, we proposed an Algorithm-1, the following is our proposed Algorithm-2 the extension of Algorithm-1 of section III, to provide more security to steganography.

A. *THE PROPOSED ALGORITHM-2 TO REINFORCE SECURITY OF STEGANOGRAPHY TO ASSURE CONFIDENTIALITY and INTEGRITY*

SENDER SIDE

Begin
1: Read Plain Text
2: Encrypt the PT using the proposed Cipher [VowelsCipher]
3: Read Cipher Text
4: Convert in to ASCII Code
5: Convert into Binary Code
6: Using Undercover Key1 Replace 0's and 1's With some Symbols
7: Read the Final Cipher Text
8: Conceal the Cipher Text into Cover Image using The stegano program.
9: If (Successful) then Goto Step10: else Goto Step7:
10: Hash the stego image using the hash program
End

The following is the Sample Experiment and Result of our proposed Algorithm-2. In this, instead of concealing the encrypted text as done in section III using our proposed Algorithm-1, here to make more security, we added some more steps in our previous algorithm-1, and we got the proposed Algorithm-2. We have taken the ASCII values for each character of the cipher text which we got from the proposed Algorithm-1, then converted into binary. Finally, using undercover Key 1 as shown in Table 4, we replaced 0's and 1's with symbols as shown in Table 5 and got the final cipher text, which is both symbols as shown in Table 6.

Table 4. Undercover Key1

Undercover Key	00	01	10	11
Symbol	Ëd‰	B†ÿÙ	Æôïç	;k¬â>

As per Section III, Scenario-3, Cipher Text of the proposed Algorithm-1 is: **EUUEAOAU IAAIEAUO OAAOUAEI UEEUAIAE.** The following tables 5& 6, describe the ASCII to Binary conversion of the cipher text and using the secret key 1, the equivalent symbols replacements.

Table 5. ASCII to Binary Conversion and Symbols Replacement

Letter	ASCII	Binary	Symbol Equivalent
A	065	01000001	B†ÿÙËd‰Ëd‰B†ÿÙ
E	069	01000101	B†ÿÙËd‰B†ÿÙB†ÿÙ
I	073	01001001	B†ÿÙËd‰ÆôïçB†ÿÙ
O	079	01001111	B†ÿÙËd‰;k¬â>;k¬â>
U	085	01010101	B†ÿÙB†ÿÙB†ÿÙB†ÿÙ

Table 6. Cipher Text with Equivalent Symbols using undercover key1

Cipher Text	Symbol Equivalent
EUUEAOAU	B†ÿÙËd‰B†ÿÙB†ÿÙB†ÿÙB†ÿÙB†ÿÙB†ÿÙB†ÿÙB†ÿÙB† ÿÙB†ÿÙËd‰B†ÿÙB†ÿÙB†ÿÙËd‰Ëd‰B†ÿÙB†ÿÙËd‰;k¬â>;k ¬â>B†ÿÙËd‰Ëd‰B†ÿÙB†ÿÙB†ÿÙB†ÿÙ
IAAIEAUO	B†ÿÙËd‰ÆôïçB†ÿÙB†- ÿÙËd‰Ëd‰B†ÿÙB†ÿÙËd‰Ëd‰B†ÿÙB†ÿÙËd‰ÆôïçB†ÿÙB† ÿÙËd‰B†ÿÙB†ÿÙB†ÿÙËd‰Ëd‰B†ÿÙB†ÿÙB†ÿÙB†ÿÙ B†ÿÙËd‰;k¬â>;k¬â>
OAAOUAEI	B†ÿÙËd‰;k¬â>;k¬â>B†ÿÙËd‰Ëd‰B†ÿÙB†ÿÙËd‰Ëd‰B†- ÿÙB†ÿÙËd‰;k¬â>;k¬â>B†ÿÙB†ÿÙB†- ÿÙB†ÿÙËd‰Ëd‰B†ÿÙ
UEEUAIAE	B†ÿÙB†ÿÙB†ÿÙB†ÿÙB†ÿÙËd‰B†ÿÙB†ÿÙB†ÿÙËd‰B†ÿÙB†- ÿÙB†ÿÙB†ÿÙB†ÿÙB†ÿÙB†ÿÙËd‰Ëd‰B†ÿÙ B†- ÿÙËd‰ÆôïçB†ÿÙB†ÿÙËd‰Ëd‰B†ÿÙB†ÿÙËd‰B†ÿÙB†ÿÙ

The following is the cipher text for the given plain text "**attacked"** using our proposed Algorithm 2.

The following is the FINAL Cipher Text of our proposed algorithm.

B†ÿÙËd‰B†ÿÙB†ÿÙB†ÿÙB†ÿÙB†ÿÙB†ÿÙB†ÿÙB†ÿÙB†ÿÙB†ÿÙËd‰B†ÿÙB†ÿÙB†ÿÙËd‰Ëd‰B†ÿÙ B†ÿÙËd‰;k¬â>;k¬â>B†ÿÙËd‰Ëd‰B†ÿÙB†ÿÙB†ÿÙB†ÿÙB†ÿÙËd‰ÆôïçB†ÿÙB†- ÿÙËd‰Ëd‰B†ÿÙB†ÿÙËd‰Ëd‰B†ÿÙB†ÿÙËd‰ÆôïçB†ÿÙB†ÿÙËd‰Ëd‰B†ÿÙB†ÿÙB†ÿÙB†ÿÙËd‰Ëd‰B†ÿÙB†- ÿÙB†ÿÙB†ÿÙB†ÿÙ B†ÿÙËd‰;k¬â>;k¬â>B†ÿÙËd‰;k¬â>;k¬â>B†ÿÙËd‰Ëd‰B†ÿÙB†ÿÙËd‰Ëd‰B†ÿÙB†- ÿÙËd‰;k¬â>;k¬â>B†ÿÙB†ÿÙB†ÿÙB†ÿÙËd‰Ëd‰B†ÿÙB†ÿÙB†ÿÙB†ÿÙËd‰Ëd‰B†ÿÙB†- ÿÙB†ÿÙËd‰B†ÿÙB†ÿÙB†ÿÙB†ÿÙB†ÿÙËd‰Ëd‰B†ÿÙ B†ÿÙËd‰ÆôïçB†ÿÙB†ÿÙËd‰Ëd‰B†- ÿÙB†ÿÙËd‰B†ÿÙB†ÿÙ

The following Figure 24 is the Flowchart which describes the mechanism of our proposed Algorithm-2.

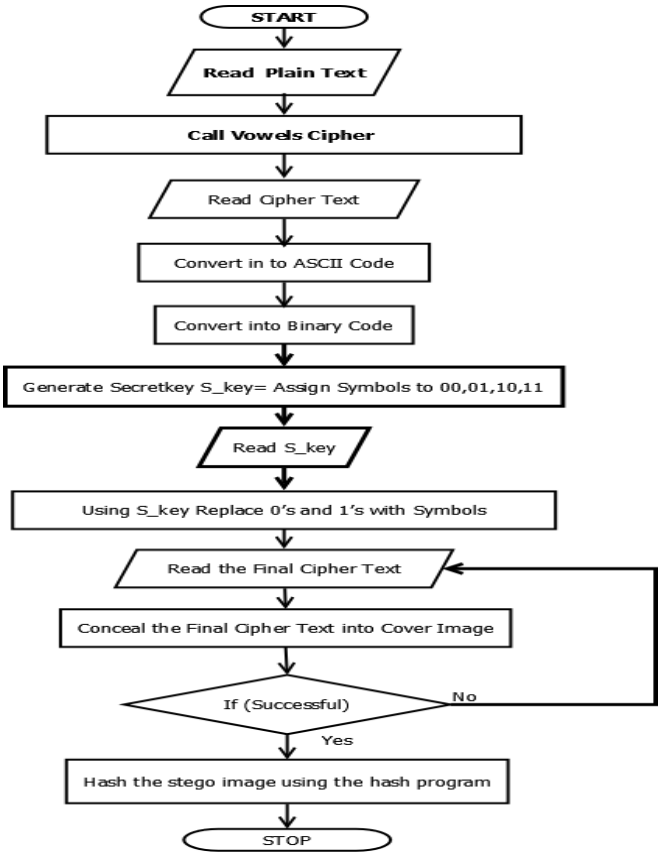

Figure 24. Flowchart of Final Proposed Algorithm-2

B. *SCENARIO-1 SENDER SIDE STEGANOGRAPHY PROCESS*

The following Figure 25 is the cipher text for the given plain text "**attacked"** using our proposed Algorithm-2.

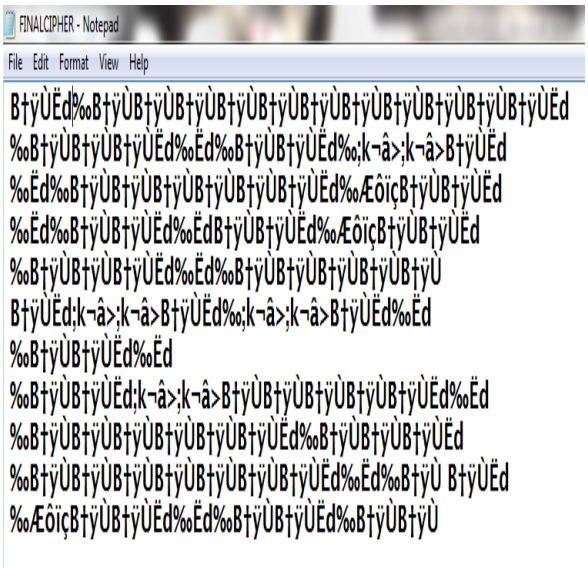

Figure 25. Content of Cipher Text

The following figure 26 describes how we embedded the cipher text into the tiger image using our program.

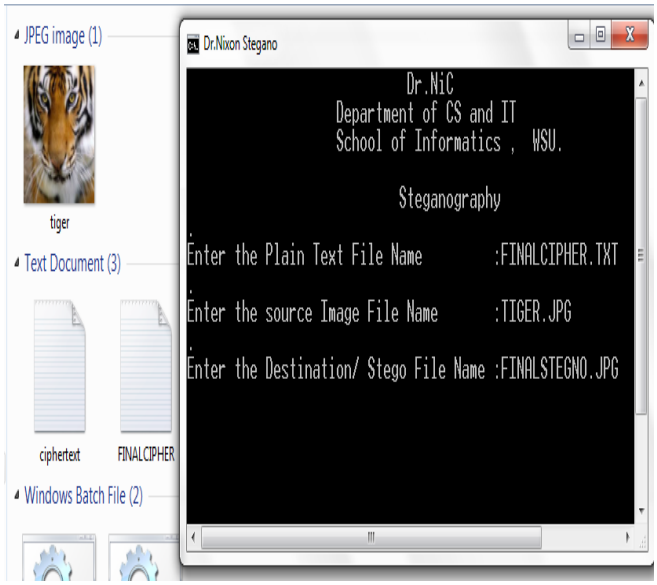

Figure 26. Stego Image Creation Process

The following figure 27 describes the successful creation of stego image which is exactly same as the source image.

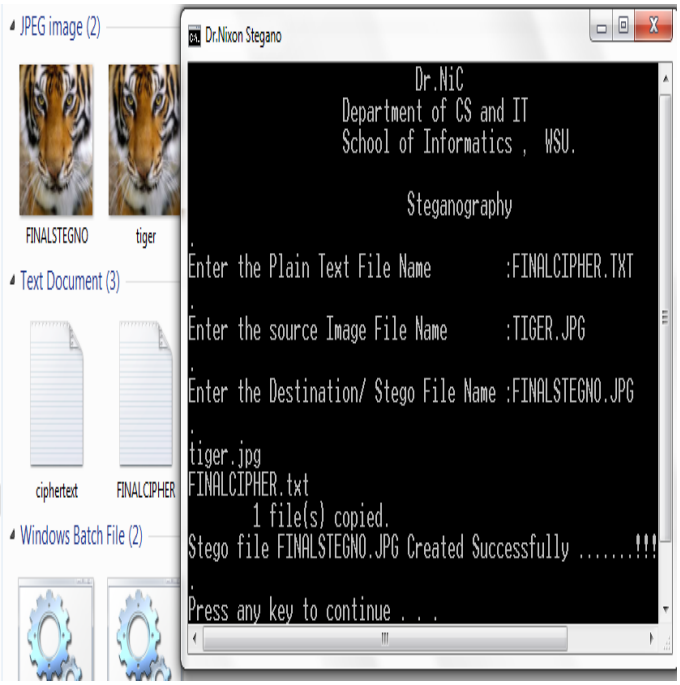

Figure 27. Successful Creation of Stego Image

The following figure 28 shows the content of the stego image where we showed our hidden cipher text almost similar to the actual content of the image since the cipher text was converted into symbols.

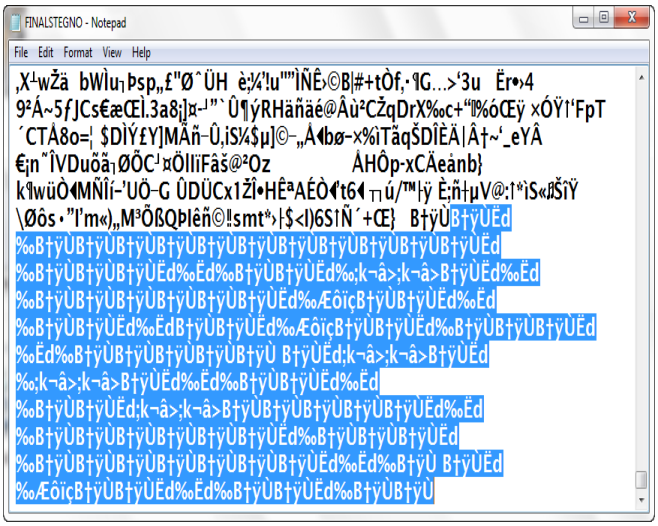

Figure 28. Content of Stego Image file

C. SCENARIO -2 SENDER SIDE HASH PROCESS

The following figure 29 describes the sender side hash process to provide integrity for the steganography, and the figure 30 shows the Hash value of the stego image.

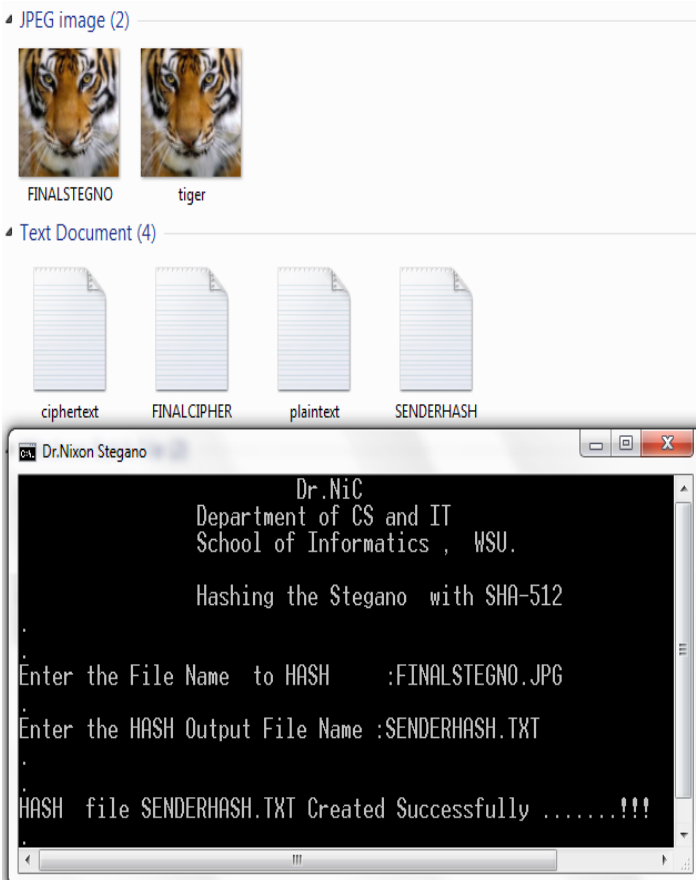

Figure 29. Sender Side Hashing Process

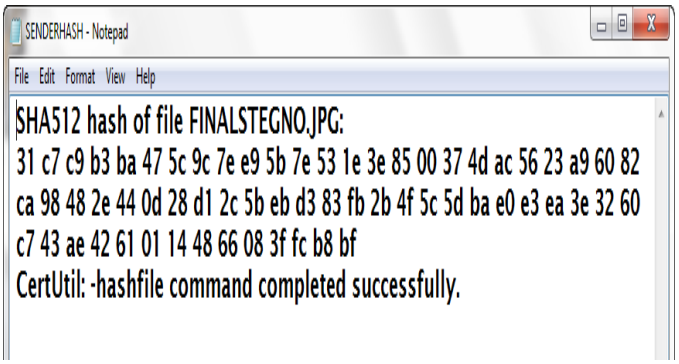

Figure 30. Sender Side Hash Value

D. SCENARIO – 3 : RECEIVER SIDE HASH PROCESS

After the receiver received the stego image and hash value, the receiver will do the following steps to check the integrity and for decryption.

Begin
Step 1: Read Stego Image, Hash value
Step 2: Hash the Stego Image
Step 3: Compare the 2 Hash Values
Step 4: If (Equal) then Goto Step 5: else goto Step6:
Step 5: Do the Decryption Process
Step 6: Request to Resend Goto Step1:
End

The following figure 31 explains the hash process of receiver using our hash program.

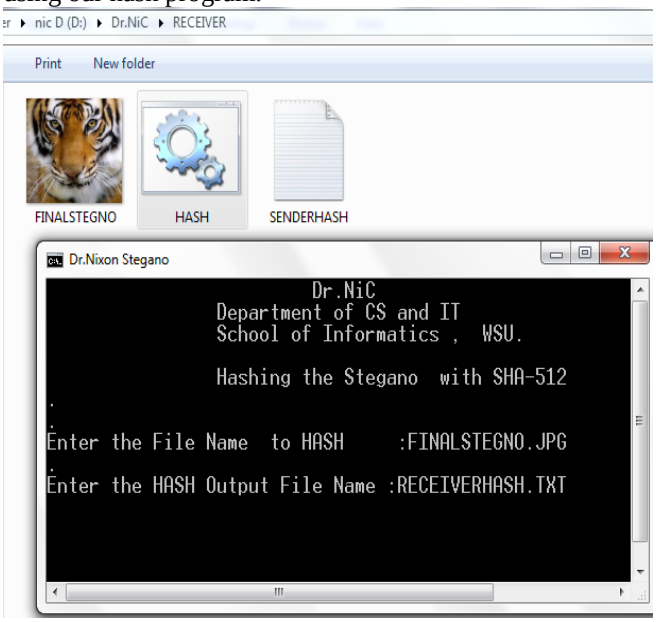

Figure 31. Receiver Side Hash Process

The following Figure 32 describes the receiver side successful creation of Hash.

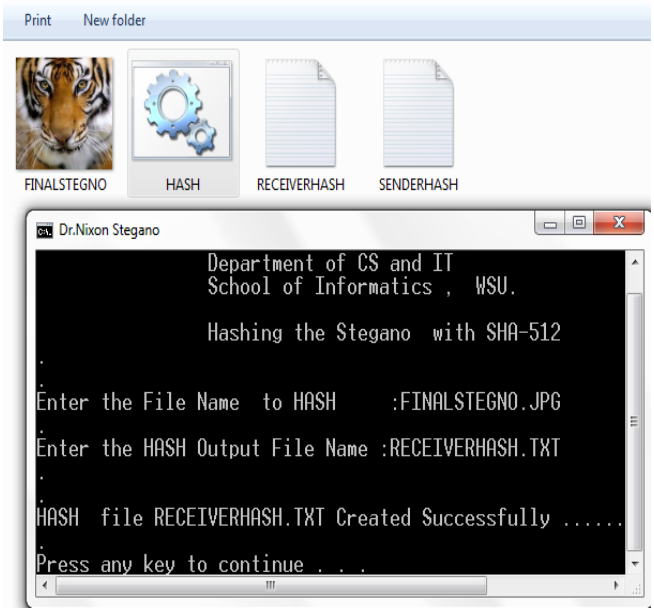

Figure 32. Successful Creation of Receiver Side Hash

The following Figure 33. Shows the Hash Value of the received stego image.

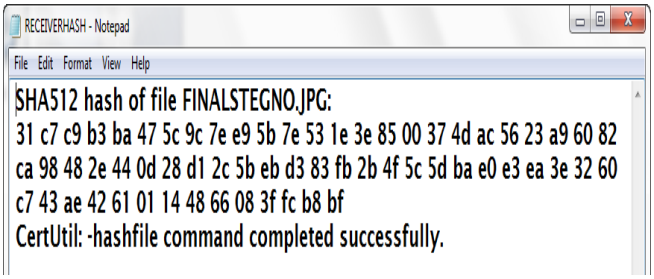

Figure 33. Receiver Side Hash Value

The following Figure 34. Describes the comparison of Sender's and Receiver's Hash Values. Both the values are equal. So, it ensures the message integrity.

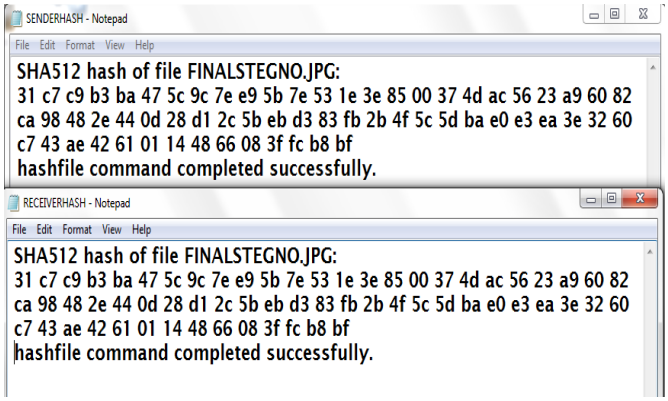

Figure 34. Comparison of Sender's and Receiver's Hash Values

Since the hash values are equal, the receiver can decrypt the cipher text using the reverse order of our final proposed Algorithm-2.

From this scenarios 1 to 3, we proved that our proposed algorithm ensured the privacy and integrity of steganography.

In the following scenario-4 we explained, if the stego image was attacked by the attacker means how the receiver can identify that.

E. *SCENARIO – 4 : COMPROMISED STEGO IMAGE*

In the communication, if any adversary attacked, the stego image means the content of the secret data would be changed. The following Figure 35 describes the same. The figures from 36 to 38 describe the receiver side hash process and comparison of hashes to conclude the security of the communication.

The following Figure 35. Shows the altered Content of Stego Image, assume that the stego was attacked by the attacker.

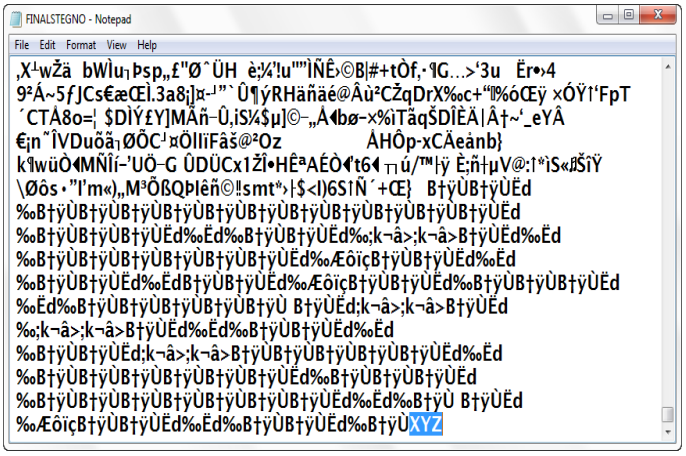

Figure 35. Altered Content of Stego Image

The following Figure 36. Explains the successful hash creation of the receiver.

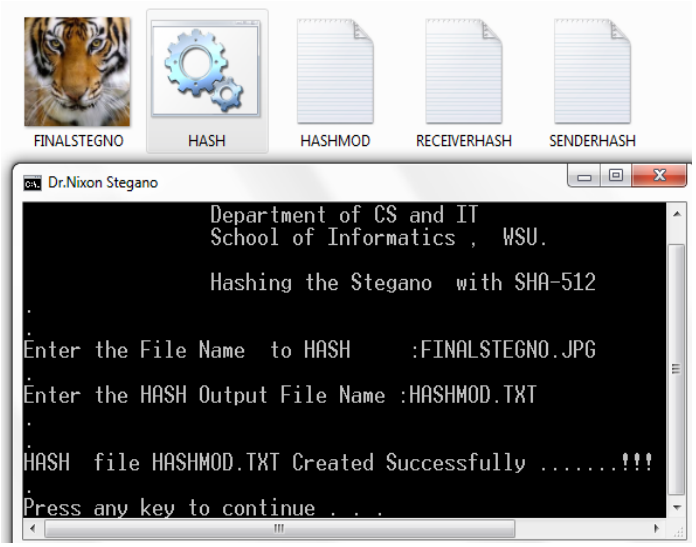

Figure 36. Receiver Side Hash Process

The following Figure 37 shows the hash value which was created by the receiver.

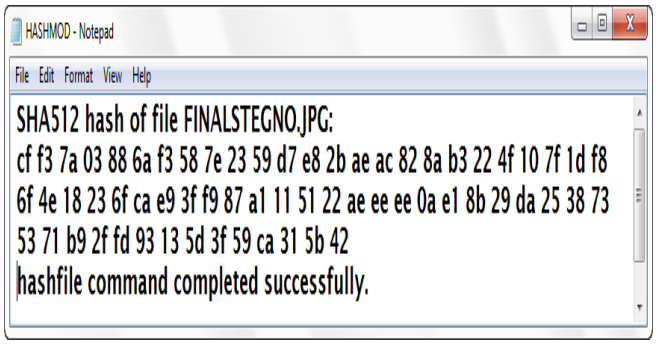

Figure 37. Receiver Side Hash Value

The following Figure 38. Shows the difference between Sender's & Receiver's hash Values. So, the receiver can easily understand that the stego was attacked in the communication.

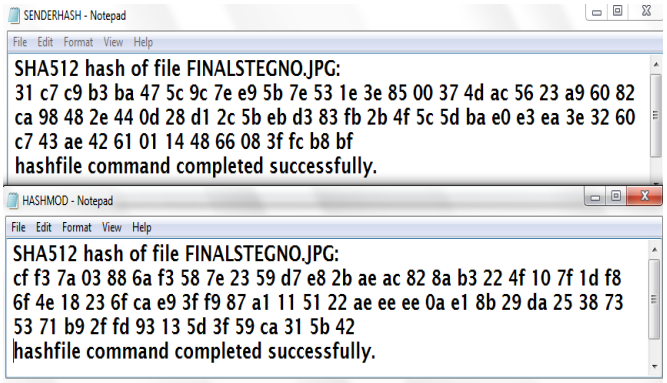

Figure 38. Difference between Sender's & Receiver's hash Values

The following figure 39 is the Flowchart of receiver side hash and decryption process.

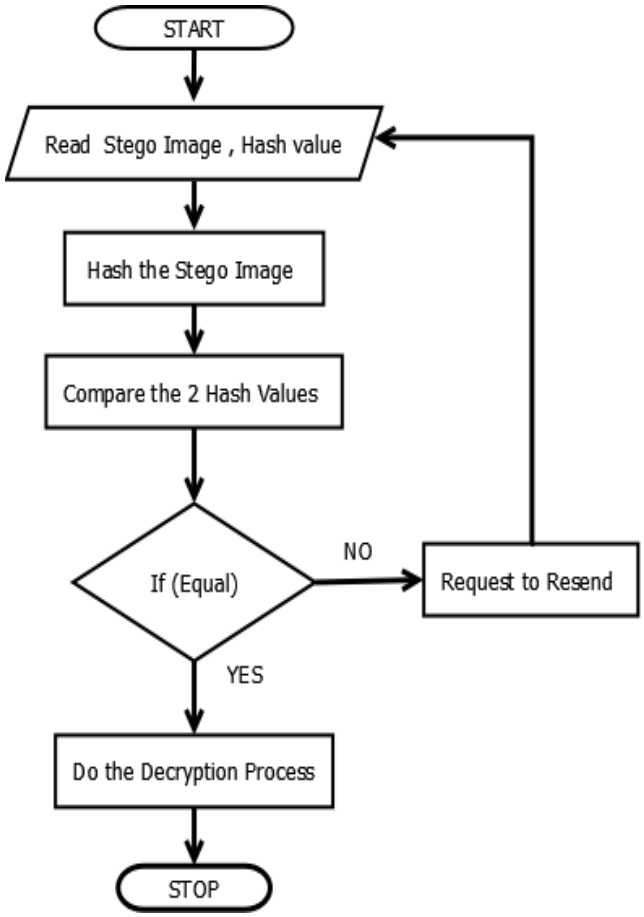

Figure 39. Flowchart of Receiver Side Process

By doing the reverse process of our final proposed algorithm the receiver can decrypt the cipher text finally he/she can get the original message.

V. CONCLUSION AND FUTURE SCOPE

Our target is to reinforce the security of Steganography. Using our coding and Algorithm, we achieved that. By implementing our proposed algorithm we assured the privacy and integrity for steganography. We used multiple cryptography techniques and multiple undercover keys. Since the techniques used and the undercover keys used were only known by the sender and receiver, it is feasible to decrypt the data only by the receiver. To decrypt the data from the image is the hardest task to the adversary since he/she doesn't know the techniques, and the undercover keys used. Even, if any modifications in the data by the adversary also easily notable by the receiver using a hash value. So, our proposed mechanism is the best way to reinforce the security of steganography to ensure the privacy and integrity of the data.

We would like to suggest the following as the future works:

1. Test the steganography algorithms in time efficiency.
2. Efficient creation of stego images in case of bandwidth consumption
3. The capture of stego images from the darknet

REFERENCES

[1] Bani Younes, M. A., & Jantan, A.,"Image encrytion using block based transformation Algorithm",International journal of computer sciences,35:1,IJCS _35_1_03. pp(407- 415), 2008

[2] Moreland, T, 'Steganography and Steganalysis', Leiden Institute of Advanced Computing Science, **2003.**

[3] Morkel, T., Eloff, J. H., and Olivier, M. S, "An overview of image steganography", In ISSA, pp.**1-11, 2005.**

[4] Sheth, R. K., & Tank, R. M.,"Image Steganography Techniques ", International Journal Of Computer Engineering And Sciences, 1(2), 10-15, (2015).

[5] R. Indrayani, H. A. Nugroho, R. Hidayat, and I. Pratama, "Increasing the security of MP3 steganography using AES Encryption and MD5 hash function," in *Proceedings - 2016 2nd International Conference on Science and Technology-Computer, ICST 2016*, 2017, pp. 129–132.

[6] Uzair Nisar1*, Craig Stewart2 , Implementation of Email System With Steganography", *International Journal of Computer Sciences and Engineering , EISSN: 2347-2693, Vol-3, issue 1, Dec 2019.*

[7] Robbi Rahim1*, Heri Nurdiyanto, "Combination Base64 Algorithm and EOF Technique for Steganography", IOP Conf. Series: Journal of Physics: Conf. Series 1007 (2018) 012003 doi :10.1088/1742-6596/1007/1/012003

[8] S. M. Masud Karim, M. S. Rahman, and M. I. Hossain, "A new approach for LSB based image steganography using undercover key," in *14th International Conference on Computer and Information Technology, ICCIT 2011*, 2011, pp. 286–291.

[9] Hajduk, V., Broda, M., Kováč, O., & Levický, D. (2016, April). Image steganography with using QR code and cryptography. In Radioelektronika (RADIOELEKTRONIKA), 2016 26th International Conference (pp. 350-353). IEEE.

[10] M. Juneja and P. S. Sandhu, "Designing of robust image steganography technique based on LSB insertion and encryption," in *ARTCom 2009 - International Conference on Advances in Recent Technologies in Communication and Computing*, 2009, pp. 302–305.

[11] Puja Mahajan, Prajakta Nimbalkar and Prtiksha Pawar,‖ Improved FPGA base X-Box Mapping of an image using Steganography Technique‖, International Journal of computer Application (0975-8887), 2016

[12] Dey, A. S., Nath, B. J., & Nath, C. A. (2012, January). A New Technique to Hide Encrypted Data in QR Codes (TM). In Proceedings on the International Conference on Internet Computing (ICOMP) (p. 1). The Steering committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp).

[13] Sharma, S., & Sejwar, V. (2016). QR Code Steganography for Multiple Image and Text Hiding using Improved RSA-3DWT Algorithm. International Journal of Security and Its Applications, 10(7), 393-406.

[14] K. Nandhini, B. Gomathi, " Implementation of LSB Based Steganography Algorithms in FPGA", Int. J. Sc. Res. in Network

Security and Communication, ISSN: 2321-3256, Volume-6, Issue-5, June 2017.

[15] G. Hamed, M. Marey, S. A. El-Sayed, and M. F. Tolba, "Hybrid technique for steganography-based on DNA with n-bits binary coding rule," in *Proceedings of the 2015 7th International Conference of Soft Computing and Pattern Recognition, SoCPaR 2015*, 2016, pp. 95–102.

Authors Profile

Dr. J. Sebastian Nixon working as a Professor in the Department of Computer Science and Information Technology, School of Informatics, Wolaita Sodo University, Ethiopia. He has secured Professional Certifications CCNA & MCSE. His research areas of interest are Information & Network Security, Cyber Security, IoT, Machine Learning and Robotics. He is published a notable number of research papers in international journals. He is a life member of several academic and professional bodies.

Mr. Mesele Gebre Awgichew working as a Lecturer in Wolaita Sodo University School of Informatics Department of Information Technology Wolaita Sodo Ethiopia. He is having 10 years of teaching experience in different colleges and universities. He is also working as a Associate Dean of School of Informatics.

Mr Akalu Assefa Afaro working as a Lecture in the Department of Information Technology, School of Informatics, Wolaita Sodo University, Ethiopia. He is having 2 years of experience. He has a certification from ITSC-Securing Network with Cisco routers and switches, Certified Ethical Hacking, Intrusion Detection and Prevention system. His research areas of interests are Network security, Artificial Intelligence, Machine Learning, Software Engineering and Natural Language Processing

Mr. Fisaha Solomon, lecturer at the Department of Information Technology, School of Informatics, Wolaita Sodo University ,Walaita Sodo, Ethiopia. He is having 8+ years of experience. He has secured Professional Certifications from Phoenix Technology ,Tulane University, Parul University,Vadodara, Gujarat, India. His research areas of interests are Cyber Security, Information, and Network Security & Cloud computing.

Mr. Paulos Bogale Wada currently working as a lecturer and department head in the Department of Computer Science, School of Informatics, Wolaita Sodo University, Ethiopia. He has finished his master's degree study in MSc in Information Science from Addis Ababa University, Ethiopia. He is engaged in various research and community service projects in Wolaita Sodo University. His research areas of interest include Machine Learning, Cloud Computing, Information and Network Security, and Big data analytics.

Mrs. Fevan Tafari pursed Master of Information Technology, Wolaita Sodo University, Ethiopia in 2018. She is working as a Lecturer in the Department of Computer Science, School of Informatics, Wolaita Sodo University, Ethiopia. Her main research works focused on WLAN, WSN, Network Security, and Machine Learning.

