National College of Ireland

Configuration Manual

MSc Internship
MSCCYB

Kapil Kapoor
X18109128

School of Computing
National College of Ireland

Supervisor: Mr Vikas Sahni

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Kapil Kapoor
Student ID:	X18109128
Programme:	MSc Cyber Security **Year:** 2019
Module:	MSc Internship
Supervisor:	Mr Vikas Sahni
Submission Due Date:	12/12/2019
Project Title:	Data Security with combination of Cryptography and Audio Steganography
Word Count:	**1175** **Page Count: 8**

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.
I agree to an electronic copy of my thesis being made publicly available on NORMA the National College of Ireland's Institutional Repository for consultation.

Signature: ……

Date: ……

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Configuration Manual

Kapil Kapoor
X18109128

1 Introduction

This configuration manual helps user to understand the flow of prototype Data security with combination of Cryptography and Audio Steganography. The software and hardware environments for the implementation. The encryption- decryption of the text, embedding and extraction process using echo hiding and GUI walkthrough.

2 Environment

2.1 Hardware

- Processor: Intel Core™ i7-8550U
- RAM: 8 GB
- Storage: 256 GB SSD
- Graphics: NVIDIA GeForce MX130 2 GB

3 Software

MATLAB is computing environment and programming language developed by the MathWorks. The implementation of algorithms, matrix manipulations, plotting of functions and data, creating user interface and interfacing with the programs written in the other languages like C, JAVA, Python etc are possible in MATLAB. It is not an open source tool, licence is required to us the tool in computer upon registering on Math labs website. The tool is available to the students for 30-day trial online but most of the toolbox required for processing are not available. So complete MATLAB tool with all the toolbox is required. (“MATLAB,” 2019)
Version – MATLAB R2018a

Download Link- https://www.mathworks.com/downloads/

4 Data source

Two audio samples were downloaded from (“Ensoniq ESQ-1 Bass C2 | Free Wave Samples,” n.d.) Audio 1 and (“Ensoniq ESQ-1 FM Piano C4 | Free Wave Samples,” n.d.) Audio 2 repectively. The audio sample were sampled at 44100 Hz and bit depth 16 bit

5 Implementation

5.1 Text

5.1.1 Encryption and Decryption

The AES algorithm is used for encryption and below steps were followed as shown in figure 1and the cipher text is obtained(Lau, 2017). The reverse of the steps is followed to get the simple text back.

1. Convert user data into ASCII of int64.
2. Generate Key
3. Convert data to square matrix and an ASCII integer.
4. Add the key to the data.
5. Repeats the following in repetition
   i. Shift the row
   ii. Creates an array based on the column and multiplies them.
   iii. Add the key to the data.
6. Shift the row a last time
7. Add the key to the data a last time.

- For the decrypt, steps 3-6 are performed in reverse order.

Figure 1(Lau, 2017)

5.2 Echo hiding

The below are the embedding and extraction process used in Bipolar echo hiding process (Tekeli and Aşlıyan, 2017).

5.2.1 Embedding

1. Cover signal is divided into equal to the number of bits to embedded.
2. Then each segment is echoed with corresponding delay to data bit to be encoded.
3. Let P be the number of bits to be encoded and k be the length of segment. Value of K is taken such that P.K is not more than length of cover signal.
4. A mixer signal is generated, and echo is filtered by applying dot product before adding to cover signal.
5. By following equation in figure 2 the embedding takes place where k_0 and k_1 are the delayed signals , s is cover audio and x is generated stego audio.

$$x = s + k_1 \cdot \text{mixer} + k_0 \cdot (1 - \text{mixer})$$

Figure 2(Tekeli and Aşlıyan, 2017)

5.2.2 Extraction

1. The stegno audio and original audio are taken.
2. With the help of the cepstrum analysis the extraction takes place.
3. The stegno audio is divided into segments as the number of bit hidden, same as done in the embedding.
4. To retrieve the nth bit of the real cepstrum of n the segment where n is number of bits hidden, the equation in figure 3 is compared on delay point for figure 4 , then the bit retrieved is 0, else is 1.

$$c_n[d_0+1] > c_n[d_1+1]$$

Figure 3 (Tekeli and Aşlıyan, 2017)

$$c_n[i] = \text{ifft}\left(\log\left(\text{abs}\left(\text{fft}\left(s_n[i]\right)\right)\right)\right)$$

Figure 4 (Tekeli and Aşlıyan, 2017)

6 Working

The interface of the encoder is shown in figure 5

6.1 Encoder

1. Select the audio file which is to act as cover. It should be in .wav format.
2. The spectrogram of the audio file will appear, and the sound will play.
3. Put the secret text to be sent.
4. Click on encrypt test AES.
5. Enter the wavelet level and stego key.
6. Click on the Embed, the AES encrypted text will be embedded in audio selected in step 1.
7. All the output parameters will appear like capacity, PSNR, MSE, encode time and SNR.
8. The spectrogram for the stego audio will appear.
9. Click on save stego Audio to save the AES text embedded audio.

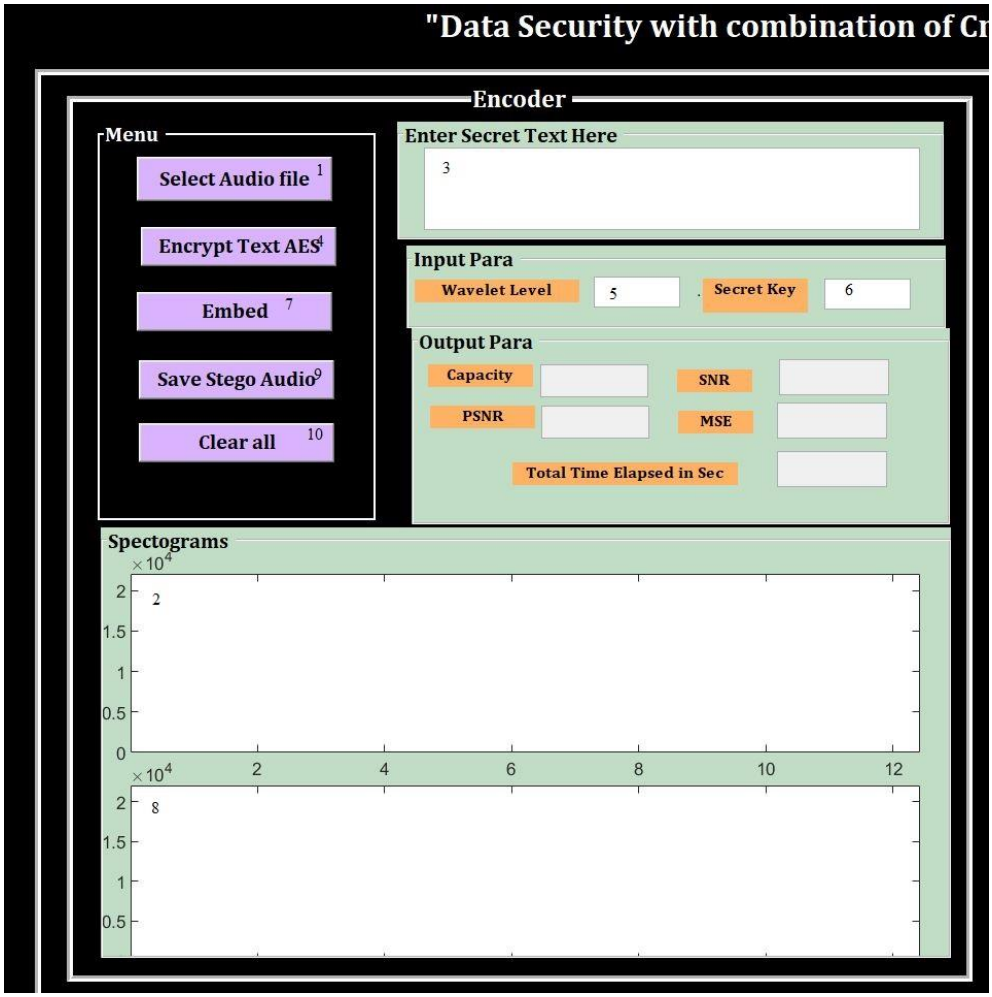

Figure 5

6.2 Decoder

1. Select the stego audio file.
2. The spectrogram will appear and stego audio plays.
3. Enter the wavelet level and stego key same as in encoder.
4. Click on extract button, the embedded text will be extracted but it will still be encrypted with AES.
5. Click on the decrypt text the plain text is displayed to you.

7 Code

7.1 Decomposition

Signal decomposition by LWT-DCT for embedding.

```
% Wavelet Decomposition
level=get(handles.edit2,'string');
level=str2num(level);

[ca, cd]=lwt(cover,'haar',level);

% Apply DCT
cd = dct(cd);
len=length(cd);
```

7.2 Embedding

The encrypted text and LWT-DCT coefficient given as input to the echo_enc_npbf function for embedding

new = echo_enc_npbf(cd', e1');

In the echo_enc_npbf function echo signal with delay d0, d1 and alpha. The mixer signal and echo signal are dot product before putting echo onto the signal.

7.3 Extraction

In echo_dec the cepstrum analysis are used to retrieve the embedded bits. These bits are then sent to AES algorithm for decryption.

```
N = floor(length(signal)/L);              %Number of frames
xsig = reshape(signal(1:N*L,1), L, N);    %Dividing signal into frames
data = char.empty(N, 0);

for k=1:N
    rceps = ifft(log(abs(fft(xsig(:,k)))));  %Real cepstrum
    if (rceps(d0+1) >= rceps(d1+1))
        data(k) = '0';
    else
        data(k) = '1';
    end
end

m   = floor(N/8);
bin = reshape(data(1:8*m), 8, m)';    %Retrieved message in binary
out = char(bin2dec(bin))';            %bin=>char

if (len_msg~=0)
    out = out(1:len_msg);
end
```

7.4 AES

In aes_simple the conversion of cipher text to plain text and plain text to cipher text takes place. The key and the plain text is taken as input for encryption and cipher text and key as input for decryption.

```
re_plaintextarray=[];
ciphertextarray=[];
for i=1:arr_size


   start1=1;stop1=8;


ciphertext = cipher (plaintextarray(i,:), w, s_box, poly_mat, 1);
ciphertextarray(i,:)=ciphertext(1,:);
ciphertext;
re_plaintext = inv_cipher (ciphertext, w, inv_s_box, inv_poly_mat, 1);
re_plaintextarray(i,1:16)=re_plaintext(1:16);
re_plaintext;
end
encrypt='';decrypt='';
for i=1:arr_size
    for j=1:16
        encrypt=horzcat(encrypt,native2unicode(ciphertextarray(i,j)));
        decrypt=horzcat(decrypt,native2unicode(re_plaintextarray(i,j)));
    end
end

t=toc;
```

References

MATLAB, 2019.. Wikipedia

Ensoniq ESQ-1 Bass C2 | Free Wave Samples [WWW Document], n.d. URL https://freewavesamples.com/ensoniq-esq-1-bass-c2 (accessed 12.10.19).

Ensoniq ESQ-1 FM Piano C4 | Free Wave Samples [WWW Document], n.d. URL https://freewavesamples.com/ensoniq-esq-1-fm-piano-c4 (accessed 12.10.19).

Lau, N., 2017. nick1au/AES-MATLAB.

Tekeli, K., Aşlıyan, R., 2017. A COMPARISON OF ECHO HIDING METHODS. Presented at the The Eurasia Proceedings of Science, Technology, Engineering & Mathematics (2602-3199), pp. 397–403.