Audio encryption – a chaos-based data byte scrambling technique

M.Y. Mohamed Parvees*

Research and Development Centre,
Bharathiar University,
Coimbatore – 641046, India
Email: yparvees@gmail.com
*Corresponding author

J. Abdul Samath

Department of Computer Science,
Government Arts College,
Udumalpet – 642126, India
Email: abdul_samath@yahoo.com

B. Parameswaran Bose

Fat Pipe Network Pvt. Ltd.,
Mettukuppam, Chennai – 600009, India
and
#35, I Main, Indiragandhi Street,
Udayanagar, Bengalore, India
Email: parameswaran.gri@gmail.com

Abstract: This study proposes an efficient audio byte scrambling technique using multiple chaotic maps. The audio encryption scheme uses Henon map and chaotic economic map for creating sequences. The system uses an efficient algorithm which makes the sequences to interdependent with each other. This inter-dependency will add more sensitivity and complexity to the proposed algorithm. A repeated application of confusion and diffusion techniques on plain audio data bytes produces the cipher audio data bytes which can be stored or sent through the public channel. The detailed study of chaotic maps with respect to Lyapunov exponent and bifurcate nature, supports the proposed cryptosystem to behave chaotically, thereby produce highly secured cipher audio. The results of the various analyses including histogram, key space, key sensitivity, statistical, differential attack, entropy, mean-variance data bytes value and randomness analyses ensures the security of the audio encryption scheme against various kinds of attacks.

Keywords: audio encryption; confusion; diffusion; chaotic map.

Reference to this paper should be made as follows: Mohamed Parvees, M.Y., Abdul Samath, J. and Parameswaran Bose, B. (2018) 'Audio encryption – a chaos-based data byte scrambling technique', *Int. J. Applied Systemic Studies*, Vol. 8, No. 1, pp.51–75.

Copyright © 2018 Inderscience Enterprises Ltd.

Biographical notes: M.Y. Mohamed Parvees received his MSc in Information Technology in 2002 from the Gandhigram Rural Institute (Deemed University) and completed his MPhil in Computer Science in 2004 from the Annamalai University, India. He is currently a faculty in the Department of Computer and Information Science, Annamalai University. He pursues his PhD in Bharathiar University. He has few international and national publications. His research interests include cryptography, multimedia security and medical information systems.

J. Abdul Samath received his PhD in Computer Science from the Gandhigram Rural Institute (Deemed University). Currently, he is working as an Assistant Professor in Government Arts College, Udumalpet. He has ten years of teaching experience and he has published 12 research articles in international journals. His research interests include neural networks, image processing, control theory, cryptography and medical image analysis.

B. Parameswaran Bose received his MSc in Information Technology in 2002 from the Gandhigram Rural Institute (Deemed University). He has nine years of experience in software research and development mainly in the field of application programming, information security, and web technologies with knowledge in analysing, developing and deploying critical applications. Presently, he does research on cryptography and information security.

1 Introduction

Due to the greatest growth of the internet bandwidth and social web applications, the internet users can send and receive the multimedia files like audio, image and video. Normally multimedia files are larger in file size and packed with different compression format like PNG, JPEG, BMP, GIF, TIFF, WAV, MPEG, MP3, MP4 and AVI. Due to the wireless and open nature of internet these multimedia files can be accessed without proper authentication. Intrusion and tampering sensitive digital data are the ever growing problems in the field information technology and communication. Further, the internet file sharing is facing more potential threats, since internet is an open access network. In order to achieve the data privacy and confidentiality, the digital data should be protected by efficient technology before it could be stored or shared.

In the recent years of modern cryptography the encryption protocols are designed by complex mathematics and developed by high level programming languages. Due to the complex design and implementation, these encryption protocols are very hard to break. The traditional cryptographic protocols like DES, 3DES, AES, RSA, IDEA and BLOWFISH are suitable for encryption, while the amount of plain data size is small. Since the image, audio and video files are compiled with bulky and redundant data, the traditional encryption algorithms are fail to encrypt multimedia files (Furht et al., 2005). So, it becomes essential to move to a new encryption technique which can be very efficient in the case of large amount of data with redundancy and in less amount of processing time. The analogue audio encryption and digital audio encryption are two fundamental encryption approaches widely employed in audio encryption (Su et al., 2012). In analogue audio encryption, there are several scrambling approaches are studied in detail with respect to frequency domain, time domain, amplitude domain, and the combination of frequency and time domain (Lin et al., 2005; Sadkhan et al., 2007; Andrade et al., 2008; Mosa et al., 2009). Though, the analogue audio encryption approaches encrypts the audio signals, the redundancy of speech is not greatly altered (Su et al., 2012). Therefore, the needs for digital audio encryption approaches are emerged to secure the audio files.

Nan et al. (2010) uses Fibonacci transform to scramble the audio signal. This approach uses a simple invariable matrix which could be guessed effortlessly. Zeng et al. (2012) scramble the speech signal using compressed sensing by splitting the signal in to two parts. Though the algorithm has very large key space, there is a need to improve quality of the recovered speech. Madain et al. (2014) employ various cellular automata types, such as Moore and von Neumann neighbourhood types, to encrypt audio files. In this study, the degree of scrambling changes with respect to the number of generation parameter which is essential to generate keys. Augustine et al. (2015) encrypt the compressed audio signal using Arnold transformation. Though the cryptosystem use linear feedback shift register (LFSR) and piecewise linear chaotic map (PWLCM) to improve the randomness to achieve higher security, it has lower key space which may not defy the brute-force attack. Mahdi and Hreshee (2016) report that the chaotic map is useful in encrypting the voice signal by producing a sequence of bit stream from the Henon map and the XOR operation is done between the converted digital signal and bit stream. In this paper, three methods are considered to generate the sequence from Henon map. These methods need additional computations during the encryption process.

Though there are very few digital audio encryption schemes proposed by few researchers (Liu et al., 2008; Gnanajeyaraman et al., 2009; Li et al., 2010; Wang et al., 2010; Sheu, 2011; Yang et al., 2015; Elshamy et al., 2015; Lima and Neto, 2016), there is a need for further research related to digital audio encryption which withstands on various cryptographic attacks.

2 Chaos-based data encryption

Chaos is a natural phenomenon which is invented by Edward Lorenz in 1963 while analysing the butterfly effect in the nonlinear dynamical systems. The chaotic maps can be iterated quickly to produce large number of sequence elements by supplying the values for the control and the initial parameters. Chaotic maps are very sensitive to its control and the initial parameters. Normally the control and initial parameters are double valued data type of 64 bits. Even a slight change in the least significant bit value of the chaotic map will produce entirely different sequence elements. This means that the initial conditions of the chaotic maps will predict the future behaviour of the entire sequence. This particular behaviour of the chaos is called deterministic chaos and the chaotic maps are classified in to two groups called discrete and continuous maps. Since the growth of the computation speed and the large storage resources, it is possible to accommodate the chaos-based data encryption systems to the real time applications. Numerous researches have been done in chaos-based data encryption since there is a vital relationship between the chaotic maps and cryptography. Since the chaotic maps are sensitive to its control and initial parameters, it can be used to build secured and robust crypto system.

The first chaos-based data encryption technique was proposed by Mathews (1989). Later many chaos-based data encryption systems are proposed by using simple one dimensional chaotic map baker's map and Arnold's cat map. In recent past many digital data encryption algorithms have been developed based on Arnold cat map, Bakers 2D map, logistic map, Henon map, piecewise linear chaotic map, Jacobian elliptic map, skew tent map and chaotic economic map (CEM) (Chen et al., 2004; Mao et al., 2004; Patidar et al., 2009; Parvees et al., 2016b). But the chaotic map-based audio encryption is still in its infancy and can be improved by more complex encryption techniques. The cryptosystems which are developed by using these chaotic maps have more significant advantages in secure communications like ergodicity, random like behaviour and sensitivity to its control and initial parameters. The security analysis of the chaos-based cryptosystem attracts the young researchers as well as some of them are found in secured. In the chaos-based cryptosystem literature the one dimensional chaotic map such as logistic map is widely used due to the various advantages of high level security, simplicity and efficiency. But in these cryptosystems the small key space is a significant drawback. To overcome this issue, in this paper, Henon two dimensional maps and CEM and are employed to encrypt the digital audio file. The results from various experiments and the security analysis show that the Henon map and CEM are more suitable for digital audio file encryption since they provide large key space and high level of security.

3 Permutation sequence generation from CEM

The CEM (Askar et al., 2015) is iterative and exhibits broad bifurcation due to its six control and initial parameters. The equation (1) represents CEM.

$$x_{n+1} = x_n + k \times \left[a - c - b \times (1+\gamma) \times x_n{}^{\gamma} \right] \tag{1}$$

where $a > 0$ market demand size, $b > 0$ market price slope, $c \geq 0$ fixed marginal cost, $k > 0$ and $k \in (0, 0.35)$ speed of adjustment parameter, γ is a constant value (3, 5), x_1 initial parameter $x_1 \in (0, 1)$. The equation (1) generates the chaotic sequences for its necessary control and initial values. For $a = 4.0$, $b = 0.6$, $c = 0.5$, $\gamma = 3.0$, $k = 0.30001$, $x_1 = 0.30001$ the following different numbers of chaotic attractors are generated as shown in Figures 1 and 2.

The bifurcation diagram shows the periodic orbits of a dynamical system based on the bifurcation parameter k represented in equation (1). For slightly larger values of k, the story of the periodic points of the CEM becomes significantly more chaotic. Many new periodic orbits come into existence as k is increased from 0.25 to 0.35 (Figure 3). This yields the bifurcation diagram which is very helpful in identifying the chaotic range of CEM.

Figure 1 CEM with 3,000 points (see online version for colours)

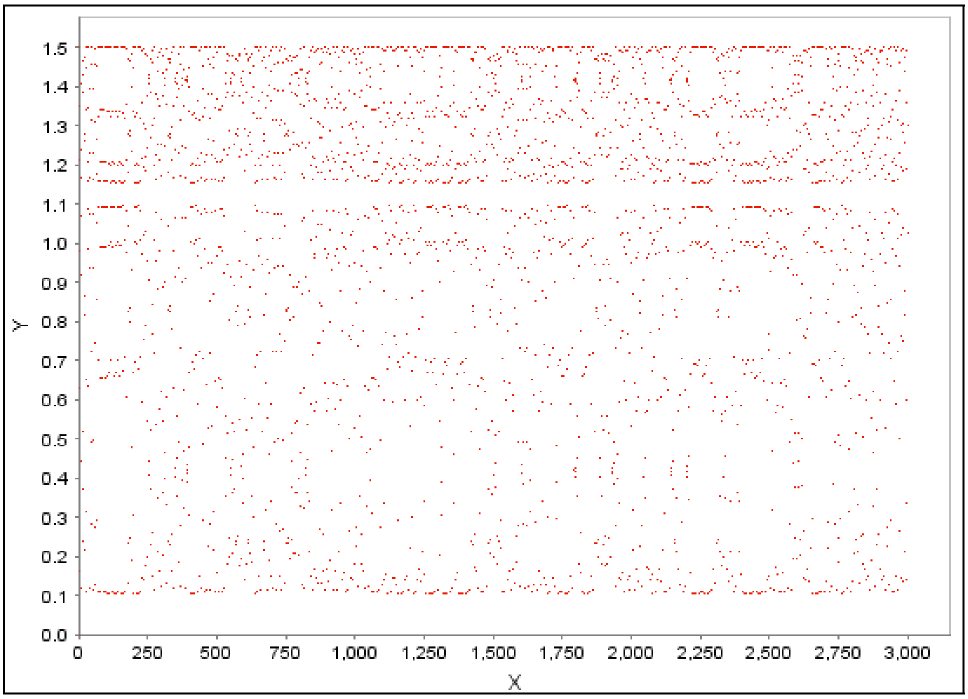


Figure 2 CEM with 10,000 points (see online version for colours)

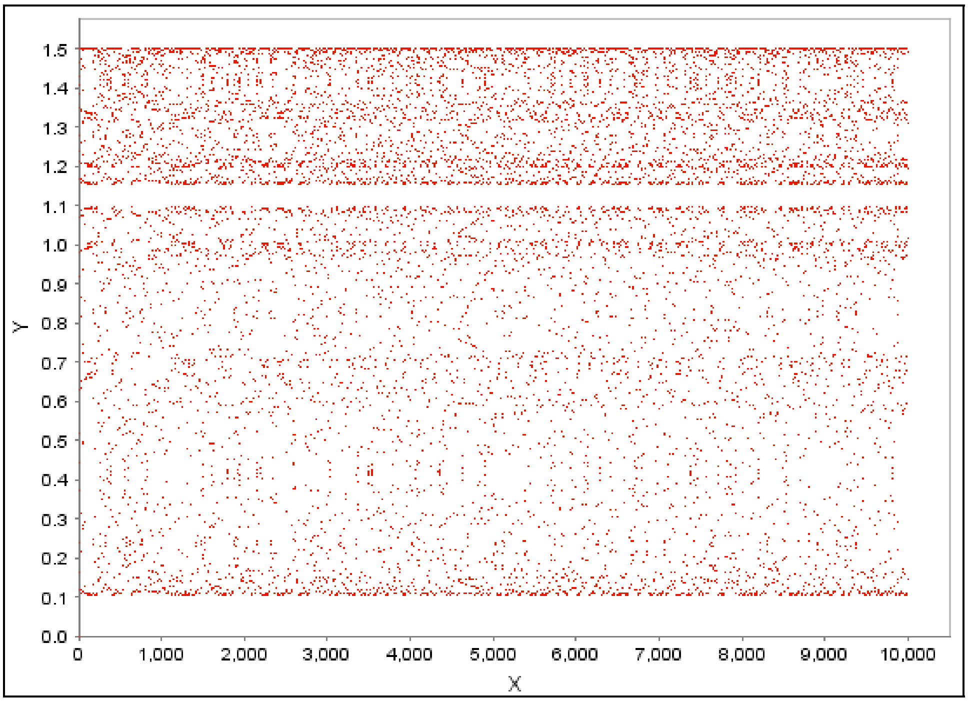
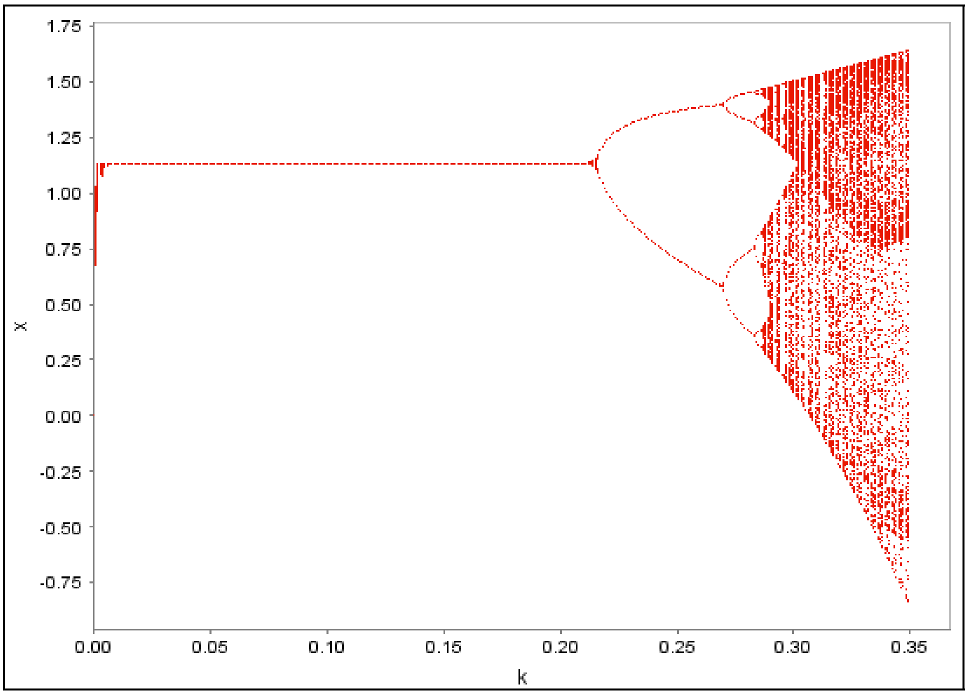


Figure 3 Bifurcation of CEM (see online version for colours)



The permutation generation is a vital part in the proposed algorithm. The iterated CEM yields the double valued chaotic sequences. It is possible to create an integer valued permutation sequences using these chaotic sequences. The proposed algorithm shuffles the audio data bytes using permutation sequences. The steps to create permutation sequence are as follows:

Step 1 Generate a chaotic sequence by iterating equation (1).

Step 2 Get rid of first 1,000 chaotic elements to avoid transitory effect.

Step 3 Choose a chaotic sequence $C = \{c_1, c_2, c_3, \ldots, c_n\}$ from the 1001st chaotic element.

Step 4 Sort C in ascending/descending order to get $S = \{s_1, s_2, s_3, \ldots, s_n\}$.

Step 5 Iterate through each element in C.

Step 6 If $c_1 = x.x_1x_2\, x_3x_4x_5x_6x_7x_8x_9x_{10}x_{11}x_{12}x_{13}x_{14}x_{15}x_{16}$ is a sample chaotic sequence element, extract the number x_4, x_6 and x_9, x_{11} from the sequence element and calculate $|(x_4 \times x_6) - (x_9 \times x_{11})|$.

Step 7 If $|(x_4 \times x_6) - (x_9 \times x_{11})|$ then group the sequence element to a separate 1D double array.

Step 8 Do the same thing if $|(x_4 \times x_6) - (x_9 \times x_{11})| == 1, 2, 3, \ldots, 81$ for other chaotic elements.

Step 9 Finally C is divided into following 1D double arrays called $\{d_1, d_2, d_3, \ldots, d_{81}\}$.

Step 10 Iterate through $\{d_1, d_2, d_3, \ldots, d_{81}\}$ and find the index position of each element in S and store the index to a 1D integer array to get $P = \{p_1, p_2, p_3, \ldots, p_n\}$.

Step 11 The integer sequence $P = \{p_1, p_2, p_3, \ldots, p_n\}$ is called as a permutation sequence.

3.1 Interdependent permutation sequence generation

The proposed encryption scheme generates four basic permutation sequences by using CEM. From these four basic permutation sequences it is possible to generate 24 interdependent permutation sequences. Shuffling the data bytes with the interdependent permutation sequences will add more complexness and sensitivity to the proposed encryption scheme, since even a slight change in a permutation sequence's initial or control parameter will affect all the other permutation sequences. The steps to create interdependent permutation sequences are as follows:

Step 1 Generate four basic permutation sequences P_1, P_2, P_3, P_4 using CEM.

Step 2 Iterate through all the sequence elements in P_1, P_2, P_3, P_4 and find 24 inter dependent permutation sequences as follows, $IP_{01} = P_4[P_3[P_2[P_1\ [i]]]]$, $IP_{02} = P_3[P_4[P_2[P_1\ [i]]]]$, $IP_{03} = P_4[P_2[P_3[P_1\ [i]]]]$, …, $IP_{24} = P_1[P_2[P_3[P_4\ [i]]]]$.

Step 3 The integer sequences $\{IP_{01}, IP_{02}, IP_{03}, \ldots, IP_{24}\}$ are called as inter dependent permutation sequences.

4 Diffusion sequence generation from Henon map

Henon map (Hénon, 1976) is discrete, boomerang shaped and represented by the following equation

$$x_{n+1} = 1 - ax_n^2 + y_n,\ y_{n+1} = bx_n \tag{2}$$

where a,b are control parameters and (x_1, y_1) are initial parameters. The Henon map generates the chaotic attractors (Figures 4 and 5) by supplying the values of $a = 1.4$, $b = 0.3$, $x_1 = 0.1$, $y_1 = 0.1$.

The bifurcation nature of the Henon map changes with respect to the control parameter a. Many new periodic orbits come into existence as 'a' is increased from 1.4 to 2.0 (Figure 6). This bifurcation range is helpful for creating the sequences for secure encryption.

Figure 4 Henon map with 1,000 points (see online version for colours)

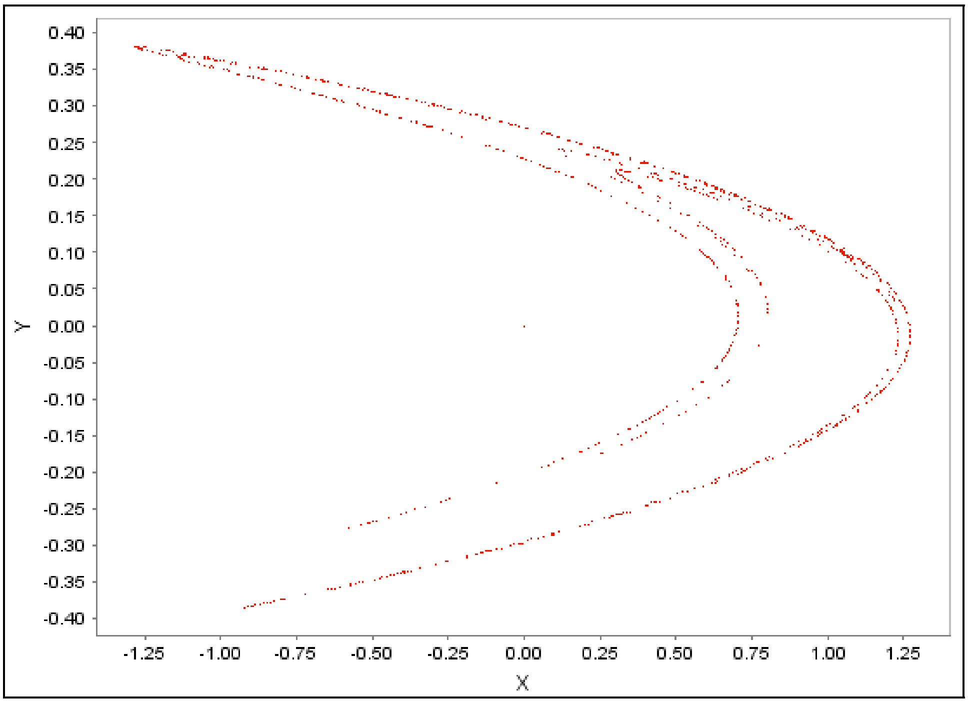


Figure 5 Henon map with 15,000 points (see online version for colours)

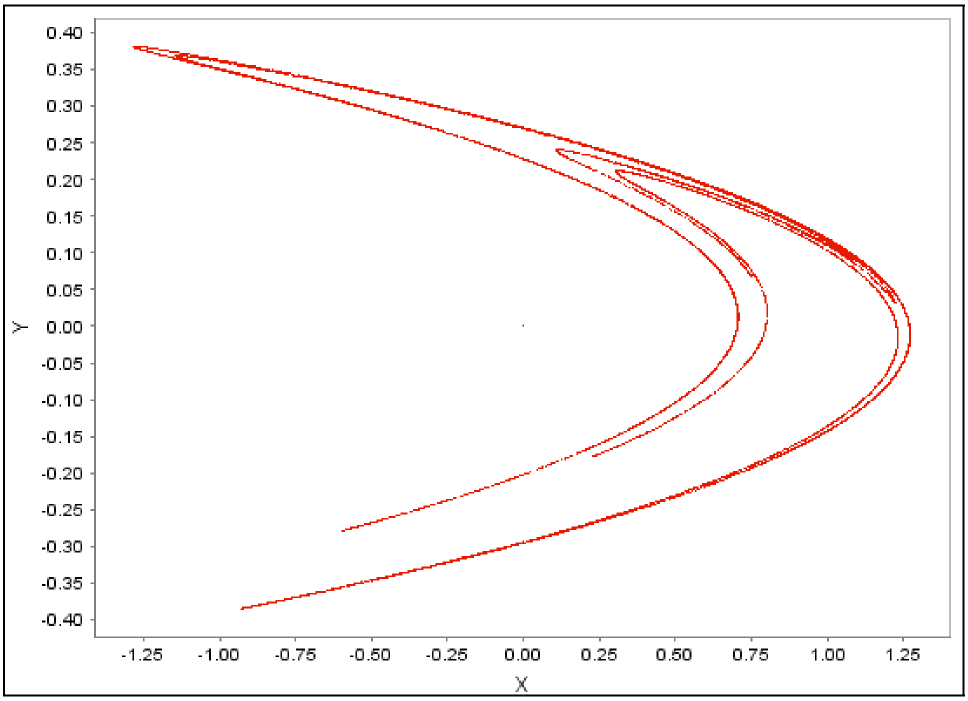
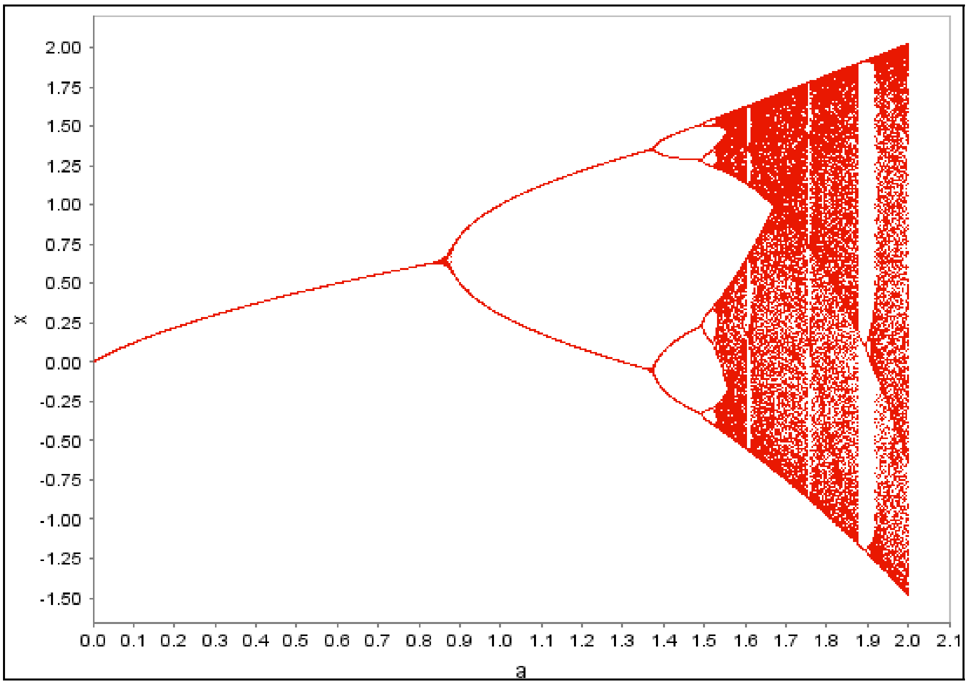


Figure 6 Bifurcation diagram of Henon map (see online version for colours)



The generation of diffusion sequence is also being a vital role in the proposed algorithm. The iterating process of Henon map yields the double valued chaotic sequences and they can be converted into an integer valued masking/diffusion sequence in the interval of (0, 255). The creation of diffusion sequence is as follows:

Step 1 Create chaotic sequences using equation (2).

Step 2 To avoid the transient effect, discard first 1,000 chaotic elements.

Step 3 Choose chaotic sequences $X = \{x_1, x_2, x_3, \ldots, x_n\}$, $XY = \{y_1, y_2, y_3, \ldots, y_n\}$ from the 1001st chaotic element.

Step 4 Iterate through X and calculate $M_1 = \text{int}\left((abs(x_i) - \lfloor abs(x_i) \rfloor) \times 10^{16}\right) \bmod 255$.

Step 5 Iterate through Y and calculate $M_2 = \text{int}\left((abs(y_i) - \lfloor abs(y_i) \rfloor) \times 10^{16}\right) \bmod 255$.

Step 6 The integer sequences M_1, M_2 are called as masking sequences.

4.1 Interdependent masking sequence generation

The 24 interdependent masking sequences are generated from four basic masking sequences which were obtained from Henon 2D map. The complexity and sensitivity of the proposed algorithm is increased due to the diffusion of data bytes using interdependent masking sequences. A slight change in a masking sequence's initial parameter will affect all the other masking sequences. The masking sequence are generated using the following steps:

Step 1 Generate four basic masking sequences M_1, M_2, M_3, M_4 using Henon 2D map.

Step 2 Iterate through all the sequence elements in M_1, M_2, M_3, M_4 and find

$$IM_{01} = M_1[IP_{01}] \wedge M_2[IP_{01}] \wedge M_3[IP_{01}] \wedge M_4[IP_{01}], ...,$$

$$IM_{12} = M_1[IP_{12}] \wedge M_2[IP_{12}] \wedge M_3[IP_{12}] \wedge M_4[IP_{12}], ...,$$

$$IM_{24} = M_1[IP_{24}] \wedge M_2[IP_{24}] \wedge M_3[IP_{24}] \wedge M_4[IP_{24}].$$

Step 3 The integer sequences $\{IM_{01}, IM_{02}, IM_{03}, IM_{04}\}$ are called as inter dependent masking sequences.

5 Proposed audio encryption scheme

The proposed audio encryption scheme is used to convert the plain digital 8-bit audio bytes in to cipher audio bytes with necessary parameters. Third parties could not understand the content of the cipher audio file. The decryption algorithm is used to convert the cipher audio bytes in to original audio bytes with necessary parameters used in the encryption process. The decryption process with slightly different key will produce entirely different audio bytes from the original one. The block diagram of the proposed algorithm is given in Figure 7.

Step 1 Read the source audio file and extract the data bytes of length *l* in to a single dimensional array *S*.

Step 2 Read the key audio file and extract the data bytes of length *k* in to a single dimensional array *K*.

Step 3 Create a permutation sequence P_1, P_2, P_3, P_4 by using CEM with the following parameters $(a_1, b_1, c_1, \gamma_1, k_1, x_1)$; $(a_2, b_2, c_2, \gamma_2, k_2, x_2)$; $(a_3, b_3, c_3, \gamma_3, k_3, x_3)$; $(a_4, b_4, c_4, \gamma_4, k_4, x_4)$ respectively.

Step 4 Create 24 interdependent permutation sequences by using the above four basic permutation sequences P_1, P_2, P_3, P_4 as $\{IP_{01}, IP_{02}, IP_{03}, \ldots, IP_{24}\}$.

Step 5 Create masking sequences M_1, M_2 by using Henon map with the following parameters a_1, b_1, x_1, y_1.

Step 6 Create masking sequences M_3, M_4 by using Henon map with the following parameters a_2, b_2, x_2, y_2.

Step 7 Create 24 interdependent masking sequences by using the above four masking sequences M_1, M_2, M_3, M_4 as $\{IP_{01}, IP_{02}, IP_{03}, \ldots, IP_{24}\}$.

Step 8 Iterate through $i = 1$ to 24.

Step 9 Shuffle one dimensional array *S* with IP_i and store into *S*.

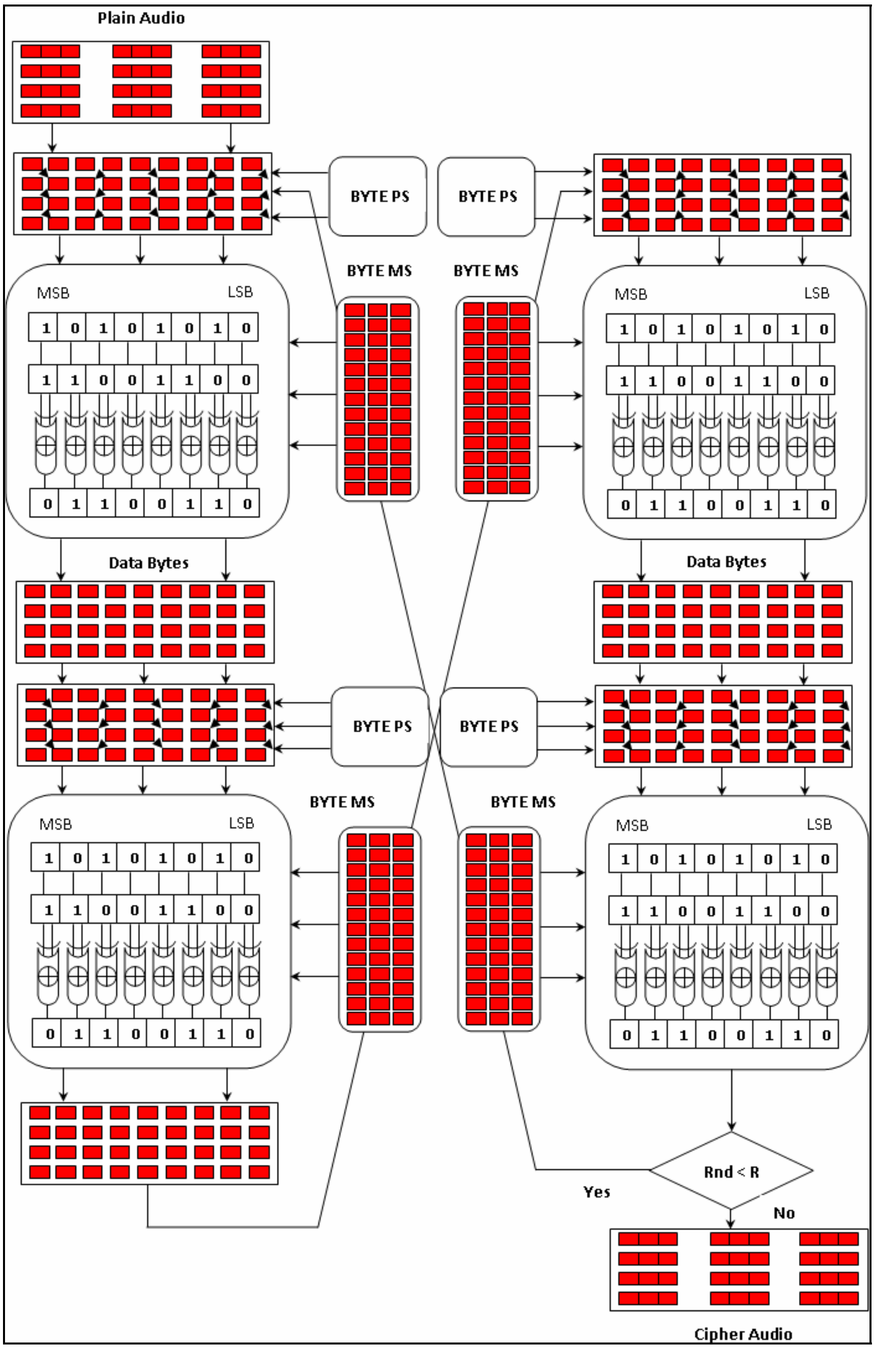
Step 10 Diffuse *S* by executing an XOR operation with IM_i and *K* bytes.

Step 11 The end of iteration.

Step 12 Continue steps 08 to 11 for *R* rounds.

Step 13 Create the cipher audio file by using the encrypted audio data bytes.

Figure 7 The block diagram of the proposed encryption scheme (see online version for colours)



6 Security analyses and results

The performance and the robustness of the proposed scheme are evaluated by an experiment. The experiment is done by using a HP Laptop with Core i5 processor with 4 GB of RAM. Java language is helpful in implementing the proposed scheme. To initiate the experiment the control and the initial parameters for the Henon map and CEM are provided. Henon map and CEM parameters are used to generate the basic each four permutation and masking sequences respectively. The control and the initial parameters used in the experiment and the resultant audio bytes are as follows,

Henon map parameters

1 $a_1 = 1.40001, b_1 = 0.30001, x_1 = 0.10001, y_1 = 0.10001$

2 $a_2 = 1.40002, b_2 = 0.30002, x_2 = 0.10002, y_2 = 0.10002.$

CEM Parameters

1 $a_1 = 4.00001, b_1 = 0.60001, c_1 = 0.50001, \gamma_1 = 3.0, k_1 = 0.31001, x_1 = 0.10001.$

2 $a_2 = 4.00002, b_2 = 0.60002, c_2 = 0.50002, \gamma_2 = 3.0, k_2 = 0.32002, x_2 = 0.20001.$

3 $a_3 = 4.00003, b_3 = 0.60003, c_3 = 0.50003, \gamma_3 = 3.0, k_3 = 0.33003, x_3 = 0.30001.$

4 $a_4 = 4.00004, b_4 = 0.60004, c_4 = 0.50004, \gamma_4 = 3.0, k_4 = 0.34004, x_4 = 0.40001.$

The strength of the proposed scheme is demonstrated by the eight different audio files with various sizes (Table 1). In this section, the resultant cipher audio files are analysed by histogram analysis, key space analysis, differential attack analysis, statistical analysis, key sensitivity analysis, information entropy analysis and mean-variance colour bytes analysis. The amplitude diagram of plain and cipher audio files are shown in Figures 10 and 11.

Table 1 The test audio files

Audio files	*1.wav*	*2.wav*	*3.wav*	*4.wav*	*5.wav*	*6.wav*	*7.wav*	*8.wav*
Size in MB	12.5	9.18	9.24	11	12.9	12.6	9.34	20.5

6.1 Histogram analysis

A histogram shows the distribution of data bytes values of a digital file. The distribution provides vital information from the plain and the cipher audio bytes because if the distribution is adequately not flat means, the certain information can be deduced by the statistical attack. Normally the plain audio bytes distribution is not flat, and forming the zigzag curve, because the distribution of data byte values will not be same. But in the case of cipher audio bytes the distribution is adequately flat, because the distribution of data byte values is almost same. So, the information of the data bytes cannot be guessed by the statistical attack. The histogram of the plain and cipher audio bytes are given in Figures 8–9. The results of histogram analysis indicate the similar results as studied in (Lima and Neto, 2016; Raghunandhan et al., 2013).

Figure 8 The histogram of the plain audio bytes (see online version for colours)

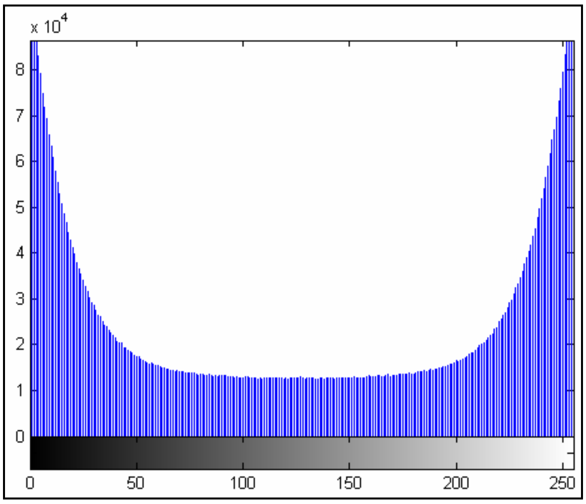
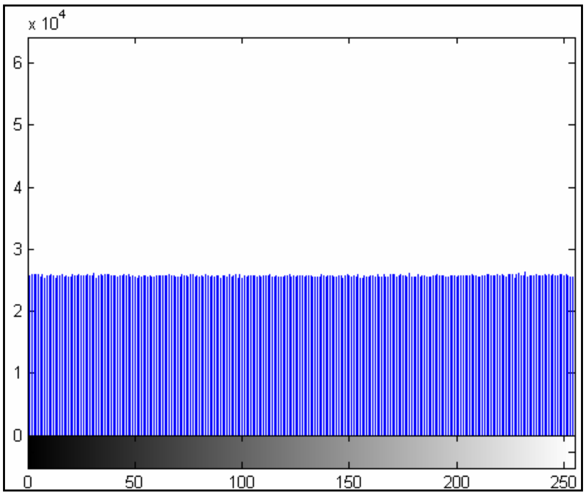


Figure 9 The histogram of the cipher audio bytes (see online version for colours)



6.2 Key space analysis

A cryptanalyst may try to break the proposed encryption scheme by using some trial keys. The total number of trial keys that can be used to break the cryptosystem is called key space. Normally the key space of a robust crypto system should be large enough to resist the brute-force attack. The initial and control parameters of CEM and Henon decide the key space of the proposed audio encryption scheme. Since the control and the initial parameters are double valued data, the key space of the proposed scheme is 10,512 and it is much higher than the key space reported in the existing schemes. Augustine et al.

(2014), (2015) and Lima and Neto (2016) propose the schemes with the key space of 2^{96}, 2^{128}, 2^{256} respectively. In the schemes of Gnanajeyaraman et al. (2009) and Lima and Neto (2016), the size of the key space depends on the number of iterations. Conversely, the parameters of the chaotic maps are determining the key space of the proposed scheme. The proposed scheme yields larger key space which is sufficient to resist the brute-force attacks.

Figure 10 The amplitude diagram of plain audio bytes (see online version for colours)

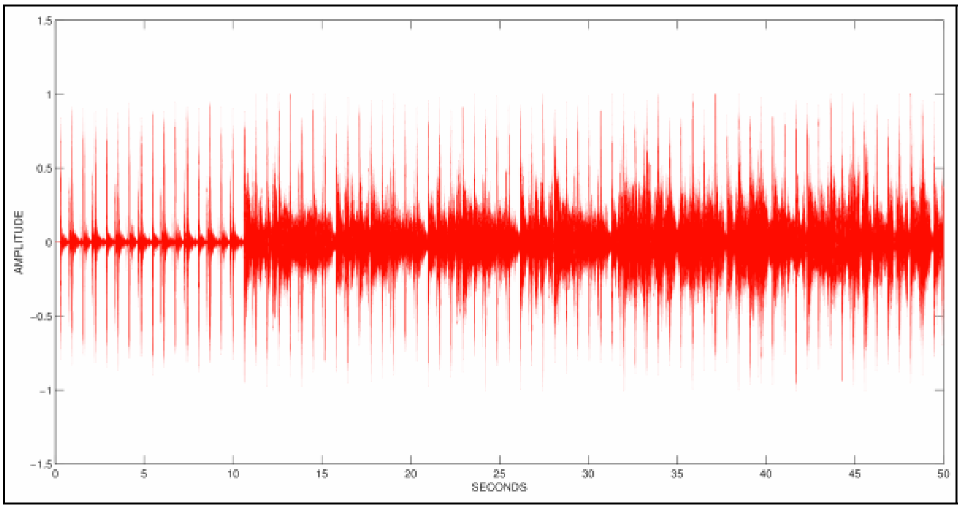
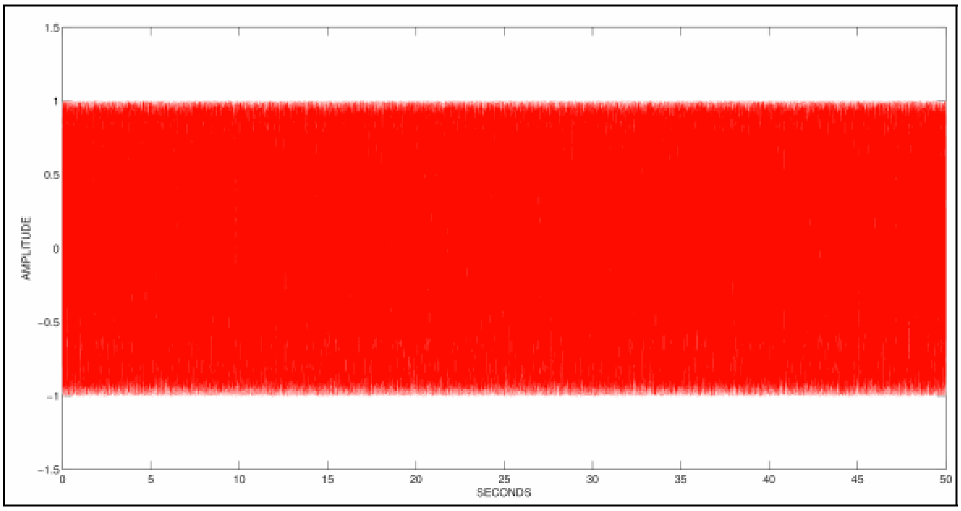


Figure 11 The amplitude diagram of cipher audio bytes (see online version for colours)



6.3 Key sensitivity analysis

Key sensitivity is an important and essential property for the proposed cryptosystem, which ensures the security of the cryptosystem against the brute-force attacks. The cipher audio bytes produced by the cryptosystem should be sensitive to the secret key. That is to say, if a cryptanalyst uses two slightly different keys to decrypt the same plain audio bytes, the two cipher audio bytes should be completely independent of each other. Now we attempt to decrypt the cipher audio bytes with slightly different keys. Given that there is a change in the initial value of the CEM parameter x_1 = 0.10002 which is slightly different from the exact encryption key, the resultant decrypted audio bytes are shown in Figures 10–13.

Figure 12 The decrypted audio bytes with correct key (see online version for colours)

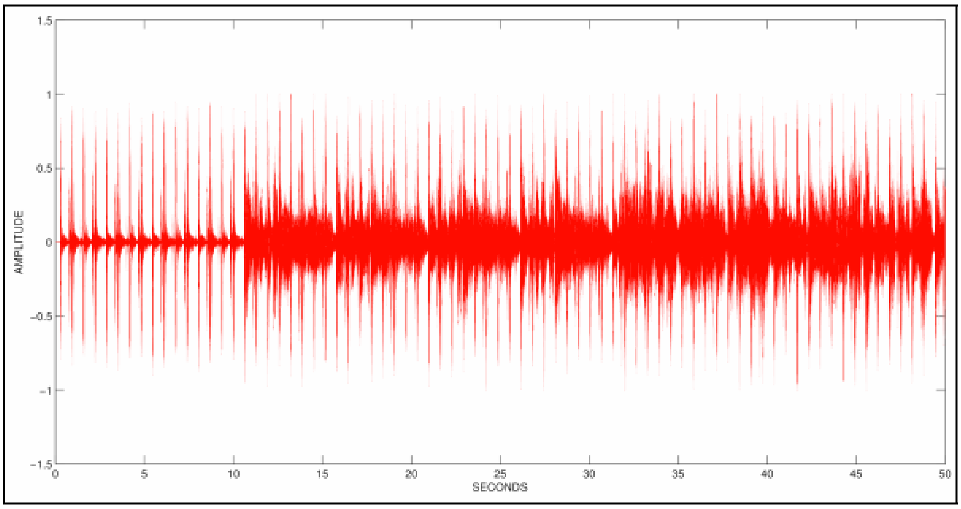
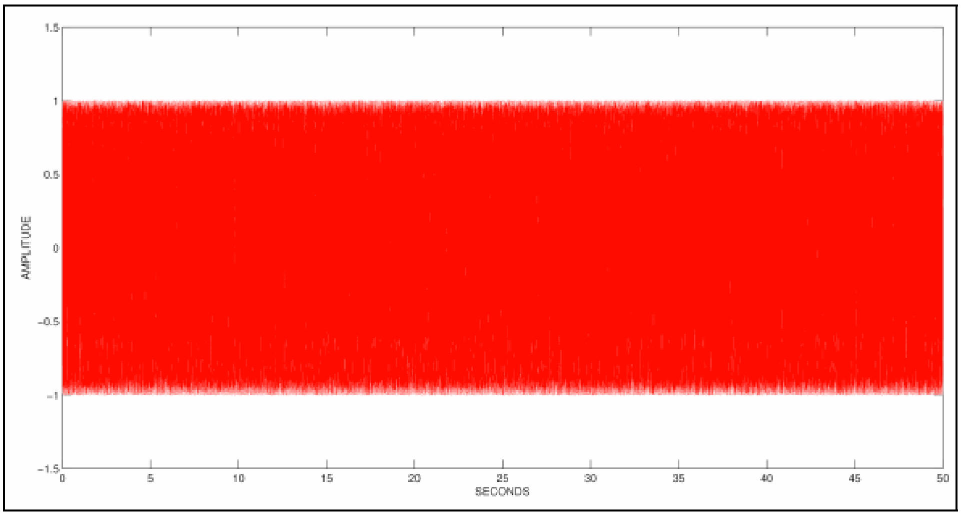


Figure 13 The decrypted audio bytes with slightly wrong key (see online version for colours)



6.4 Statistical analysis

The sternness of the proposed audio encryption technique is evaluated by calculating the correlation coefficient, mean square error (MSE) and peak signal noise ratio (PSNR) of ciphered audio files. The correlation coefficient of two adjacent bytes of plain and ciphered audio is calculated using equation (3). From Table 2, the correlation value of cipher audio is lower than plain audio and the value is nearing ‘0’ which proves that the algorithm hides the information of the cipher audio file thereby achieves confidentiality. In some of the literatures, the best correlation coefficient values –0.0001 and –0.0274 are found in Augustine et al. (2015) and Tamimi and Abdalla (2014) respectively. In existing approaches, the correlation coefficient values of cipher audio file are 0.0593, 0.0049, –0.0018, –0.0014 found in Babu (2013), Elshamy et al. (2015), Lima and Neto, (2016), Mosa et al. (2011) respectively. Since the proposed study involves in scrambling of audio bytes, the correlation coefficient value between the ciphers audio bytes are calculated and the values are optimal and better than the existing approaches.

Table 2 The correlation coefficient values

Audio file	*Plain audio*	*Cipher audio*
1.wav	0.055821	–0.000620
2.wav	0.049835	–0.000376
3.wav	0.035266	–0.000557
4.wav	0.036313	0.000235
5.wav	0.158048	0.000023
6.wav	0.010842	–0.001216
7.wav	0.071136	0.000486
8.wav	0.027516	–0.000315

Table 3 The MSE and PSNR values

Audio file	*MSE*	*PSNR*
1.wav	120.95726	27.30448
2.wav	113.32472	27.58755
3.wav	115.43571	27.50740
4.wav	113.74329	27.57154
5.wav	125.53849	27.14303
6.wav	113.69556	27.57336
7.wav	114.27544	27.55127
8.wav	117.39892	27.43416

Similarly, the MSE and PSNR are additional metrics which figure out the dissimilarity between the plain audio and ciphered audio files. The MSE and PSNR values are calculated between the scrambled and plain audio bytes. The calculated values are shown in Table 3. Further, the MSE and PSNR values of audio file (5.wav) are 125.53849 and 27.14303 respectively. The values are similar to the results of byte scrambling approach proposed by Parvees et al. (2016a, 2017). The resulted MSE and PSNR value prove that the plain and cipher audio files are entirely different and hence achieve higher confidentiality.

$$\gamma_{xy} = \frac{\operatorname{cov}(x, y)}{\sqrt{D(x)}\sqrt{D(y)}} \tag{3}$$

where *x* and *y* are two adjacent byte values of audio file.

$$Cov(x\ y) = \frac{1}{W \times H}\sum_{p=1}^{W}\sum_{q=1}^{H}[x(p, q) - E(x)][y(p, q) - E(y)]$$

$$E(x) = \frac{1}{W \times H}\sum_{p=1}^{W}\sum_{q=1}^{H} x(p, q),\ D(x) = \frac{1}{W \times H}\sum_{p=1}^{W}\sum_{q=1}^{H}[x(p, q) - E(x)]^2.$$

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(P[i] - E[i])^2 \tag{4}$$

where *P* and *E* are plain and cipher audio files respectively.

$$PSNR = 10\ \log_{10}\left(\frac{MAX^2}{MSE}\right) \tag{5}$$

where *MAX* is the highest feasible value in the audio file.

6.5 *Differential attack analysis*

A cryptanalyst can modify a single bit in the plain audio bytes and can map out meaningful differences between the ciphers audio bytes. This type of attack is called as differential attack. A strong encryption algorithm is sensitive to a tiny change made at one bit in the plain audio bytes and should produce cipher audio bytes with lots of changes. The differential attack analysis can be done by calculating net bytes change rate (NBCR) and unified average changing intensity (UACI). The formulas to calculate NBCR and UACI values are given in (4) and (5). The calculated results between the plain (c_1) and cipher (c_2) audio bytes of length *l* are listed in Table 4. The observed NBCR and UACI values are near to 99.6% and 36.3%. The results are ideal and similar to the values are reported by Lima and Neto (2016).

$$NBCR = \frac{1}{l}\sum_{i=1}^{l} f(i) \times 100 \tag{6}$$

$$UACI = \frac{1}{l}\sum_{i=1}^{l}\frac{|c_1(i) - c_2(i)|}{255} \times 100 \tag{7}$$

where $f(i) = \begin{cases} 1, c_1(i) \neq c_2(i), \\ 0, \text{Otherwise.} \end{cases}$

Table 4 The NBCR and UACI values

Audio file	*NBCR (%)*	*UACI (%)*
1.wav	99.60742	39.14082
2.wav	99.60812	36.39705
3.wav	99.61265	37.14565
4.wav	99.60942	36.54512
5.wav	99.60354	40.86628
6.wav	99.61085	36.53489
7.wav	99.60827	36.73718
8.wav	99.61110	37.84243

6.6 Entropy analysis

The entropy is the measure of randomness of a cipher audio. Since the audio signal carries 8 bit information, the expected ideal entropy value should be eight. The entropy is calculated using the equation (8).

$$H(m)=\sum_{i=1}^{M}p(m_i)\log\frac{1}{p(m_i)} \tag{8}$$

where m – the information source, M – total number of symbols $m_i \in m$; $p(m_i)$ represents the probability of occurrence of symbol m_i. The experimental results are presented in Table 5. Since the 8-bit audio file has been encrypted, the best entropy value is 7.99997. The proposed entropy value is higher and much better than the results of Tamimi and Abdalla (2014) whose entropy value is 6.2214.

Table 5 Information entropy values of audio bytes

Audio file	*Plain audio*	*Cipher audio*
1.wav	7.59478	7.99997
2.wav	7.87248	7.99995
3.wav	7.81755	7.99996
4.wav	7.87201	7.99997
5.wav	7.20863	7.99997
6.wav	7.88052	7.99997
7.wav	7.83119	7.99996
8.wav	7.76953	7.99997

6.7 Mean-variance data bytes value analysis

The more uniform difference of the data byte value between the plain and the cipher audio yields the larger mean-variance data byte value, which means that the encryption algorithm scrambles better. The mean-variance data byte value is calculated using the equation (9)

$$C = \frac{1}{l}\sum_{i=1}^{l}(B(i, j) - B) \tag{9}$$

where *B* denotes the average of all the data byte value of audio file, and *l* is the length of the plain and cipher audio bytes. The mean-variance data byte values of plain and cipher audio are shown in Table 6. The calculated mean-variance data byte value of plain and cipher audio files (5. wav) are 64.07933 and 94.08394 respectively. The calculated value of cipher is higher than plain audio and the results are similar to the values calculated by Parvees et al. (2016a) for an image scrambling approach.

Table 6 Mean-variance data byte values plain and cipher audio file

Audio file	*Plain audio*	*Cipher audio*
1.wav	64.06394	88.21579
2.wav	64.04514	77.67237
3.wav	64.03109	80.10805
4.wav	64.02604	78.45022
5.wav	64.07933	94.08394
6.wav	64.02049	78.22276
7.wav	64.03576	78.61768
8.wav	64.05174	83.76371

Table 7a Results of the NIST SP800-22 test suite

Statistical test name	*Proportion*	*P-value*	*Result*
Frequency	9/10	0.739918	Success
Block-frequency	10/10	0.066882	Success
Cumulative-sums-forward	9/10	0.739918	Success
Cumulative-sums-reverse	9/10	0.534146	Success
Runs	10/10	0.534146	Success
Longest-run	10/10	0.534146	Success
Rank	10/10	0.911413	Success
FFT	10/10	0.911413	Success
Non-overlapping-template	10/10	0.911413	Success
Overlapping-template	10/10	0.911413	Success
Universal Maurer’s test	9/10	0.122325	Success
Approximate-entropy	10/10	0.739918	Success
Serial p-value 1	10/10	0.911413	Success
Serial p-value 2	10/10	0.739918	Success
Linear-complexity	10/10	0.739918	Success

Table 7b Results of the random excursion tests based on NIST SP800-22 test suite

x-value	*chi^2*	*P-value*	*Result*
x = –4	4.176828	0.524248	Success
x= –3	5.523538	0.355372	Success
x = –2	4.060511	0.540737	Success
x = –1	9.873239	0.078908	Success
x = 1	5.739437	0.332404	Success
x = 2	3.142671	0.678001	Success
x = 3	2.264068	0.811530	Success
x = 4	2.904042	0.714777	Success

Table 7c Results of the random excursion variant test based on NIST SP800-22 test suite

x-value	*Total visits*	*P-value*	*Result*
–9	478	0.517222	SUCCESS
–8	465	0.430085	SUCCESS
–7	502	0.587058	SUCCESS
–6	517	0.648224	SUCCESS
–5	547	0.835474	SUCCESS
–4	553	0.866418	SUCCESS
–3	525	0.568304	SUCCESS
–2	504	0.272947	SUCCESS
–1	519	0.146000	SUCCESS
1	555	0.699716	SUCCESS
2	502	0.258240	SUCCESS
3	463	0.163558	SUCCESS
4	439	0.148006	SUCCESS
5	453	0.255399	SUCCESS
6	456	0.316383	SUCCESS
7	447	0.319400	SUCCESS
8	451	0.370095	SUCCESS
9	440	0.357010	SUCCESS

6.8 Randomness analysis

The proposed audio encryption scheme should satisfy the randomness analysis also. So, the encrypted audio byte is subjected to the various randomness tests, namely, NIST-800-22 statistical test (Rukhin et al., 2001), ENT and Diehard tests. The results of the various tests are shown in Table 7–9. From Table 7(a)–7(c), the cipher audio bytes successfully passes all the statistical tests by possessing the p-value greater than 0.01. The randomness of cipher audio is ensured through the NIST tests suite. All the NIST tests are successful for the cipher audio and the test results are identical with the results of Sadkhan et al. (2007) and Battey and Parakh (2012). Similarly, Table 8 implies the results of list of ENT statistical tests that are passed successfully. The tests of ENT test suite on cipher audio produce successful results similar to the results reported by Stoyanov and Kordov (2015) and Guan and Tan (2004). Table 9 shows the successful results of Diehard tests and the p-values of the different tests are greater than the 0.01 which proves that the cipher audio bytes are more random thereby proves the efficiency of the audio cryptosystem. Alani (2010) defined the safe area for the p-values of DieHard tests for ciphertext that range from 0.25 to 0.75. The results of the proposed study also have the p-values that lie within the range of 0.25 to 0.75. Hence, the randomness of the cipher audio is achieved through the proposed algorithm.

Table 8 Results of the ENT test suite

Statistical test name	*Average value*	*Result*
Entropy	7.999988	Success
Arithmetic mean	127.4895	Success
Monte Carlo	3.141443653	Success
Chi-square	298.42	Success
Serial correlation coefficient	-0.000477	Success

Table 9 Results of the diehard test suite

Statistical test name	*Average P-value*	*Result*
Birthday spacing	0.376893	Success
Overlapping 5-permutation	0.953996	Success
Binary rank 31 × 31 matrices	0.361433	Success
Binary rank 32 × 32 matrices	0.719600	Success
Binary rank 06 × 08 matrices	0.843737	Success
Bit stream	0.515540	Success
Overlapping-pairs-sparse-occupancy	0.652500	Success
Overlapping-quadruples-sparse-occupancy	0.956100	Success
DNA	0.660900	Success
Count the ones-01	0.227006	Success
Count the ones-02	0.231188	Success
Parking lot	0.536038	Success
Minimum distance	0.448353	Success
3D spheres	0.306861	Success
Squeeze	0.528838	Success
Overlapping sum	0.882427	Success
Runs-up	0.352391	Success
Runs-down	0.439515	Success
Craps	0.20039	Success

7 Discussion

The chaos-based audio encryption scheme is proposed using two different chaotic maps, namely, Henon and CEM map. In Gnanajeyaraman et al. (2009), the chaotic sequences are directly used for sequence generation. But, the proposed scheme uses two different chaotic maps to generate different interdependent confusion and diffusion sequences, thereby, achieves the higher security. Further, the encryption scheme is efficient in terms of resisting various attacks namely brute-force, statistical, differential and entropy attacks. The key space of the proposed scheme is higher than other digital audio encryption schemes proposed by Peng et al. (2003), Augustine et al. (2015, 2014) and Lima and Neto (2016), thereby resist brute-force attack. The bytes of cipher audio file have weaker correlation with adjacent byte values and the results are optimal and better than the results of Mosa et al. (2011), Babu (2013), Tamimi and Abdalla (2014), Augustine et al. (2015), Elshamy et al. (2015) and Lima and Neto (2016). The results of MSE and PSNR values are similar to Parvees et al. (2016a, 2017). The results prove that the cipher audio bytes are more random thereby deny statistical attack. In this approach, the NBCR and UACI values are ideal and similar to the results mentioned by Lima and Neto (2016); thereby, the proposed scheme resists the differential attack. Since the 8 bit audio signals are considered for encryption, the entropy value is also optimal while comparing with the literature (Tamimi and Abdalla, 2014). Further, this study proposes to conduct the random tests, namely, NIST SP800-22, ENT and Diehard tests, on cipher audio bytes. The results of the random tests are successful which proves that the audio encryption is achieved higher level of security. In this paper, at most, all of the security analyses relevant to digital audio encryption are analysed, but in related literatures in which they discuss very few analyses.

8 Conclusions

The CEM and Henon map are useful in generating the more random permutation and diffusion sequences which have been employed for scrambling of audio bytes. The efficiency of the encryption standard primarily relies on the standard of algorithm. The encryption algorithm uses the advantage of interdependency of sequences which has been taken up in this study to reinforce the algorithm. The mathematical analysis of chaotic maps in terms of bifurcate nature and Lyapunov exponent supports the efficient chaotic encryption. The cipher audio withstands on the various kinds of security attacks including statistical, brute-force, and differential attacks thereby proves that the proposed audio cryptosystem is more efficient.

The future research plan includes improving the key space by enhancing chaotic maps thereby attaining larger bifurcate range and positive Lyapunov exponents to acquire higher security against brute-force attacks. The proposed scrambling algorithm can be extended along with pre or post-processing of audio watermarking to provide copyright protection. Further, the significant interdependent sequences can be engaged in encrypting multi-channel audio, image and video files.

References

Alani, M.M. (2010) 'Testing randomness in ciphertext of block-ciphers using diehard tests', *International Journal of Computer Science and Network Security*, Vol. 10, No. 4, pp.53–57.

Andrade, J., Campos, M. and Apolinario, J. (2008) 'Speech privacy for modern mobile communication systems', *Proceedings of IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, IEEE Press, Nevada, USA, pp.1777–1780.

Askar, S.S., Karawia, A.A. and Alshamrani, A. (2015) 'Image encryption algorithm based on chaotic economic model', *Mathematical Problems in Engineering*, Vol. 2015, Article ID 341729, 10pp., DOI: 10.1155/2015/341729.

Augustine, N., George, S.N. and Deepthi, P. (2014) 'Compressive sensing based audio scrambling using Arnold transform', in *Recent Trends in Computer Networks and Distributed Systems Security*, Springer, pp.172–183.

Augustine, N., George, S.N. and Pattathil, D.P. (2015) 'An audio encryption technique through compressive sensing and Arnold transform', *International Journal of Trust Management in Computing and Communications*, Vol. 3, No. 1, pp.74–92.

Babu, G.S. and Ilango, P. (2013) 'Higher dimensional chaos for audio encryption', in *2013 IEEE Symposium on Computational Intelligence in Cyber Security (CICS)*, pp.52–58.

Battey, M. and Parakh, A. (2012) 'A quasigroup based random number generator for resource constrained environments', *IACR Cryptology ePrint Archive*, p.471.

Chen, G., Mao, Y.B. Chui, C.K. (2004) 'A symmetric image encryption scheme based on 3D chaotic cat maps', *Chaos Solutions and Fractals*, Vol. 21, No. 3, pp.749–761.

Elshamy, E.M., El-Rabaie, E.M., Faragallah, O.S., Elshakankiry, O.A., El-Samie, F.E., El-sayed, H.S. and El-Zoghdy S.F. (2015) 'Efficient audio cryptosystem based on chaotic maps and double random phase encoding', *International Journal of Speech Technology*, Vol. 18, p.619, DOI: 10.1007/s10772-015-9279-3.

Furht, B., Muharemagic, E. and Socek, D. (2005) *Multimedia Encryption and Watermarking*, Springer-Verlag, New York.

Gnanajeyaraman, R., Prasadh, K. and Ramar, D. (2009) 'Audio encryption using higher dimensional chaotic map', *International Journal of Recent Trends in Engineering*, Vol. 1, No. 2, pp.103–107.

Guan, S.U. and Tan, S.K. (2004) 'Pseudorandom number generation with self-programmable cellular automata', *IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems*, Vol. 23, No. 7, pp.1095–1101.

Hénon, M. (1976) 'A two-dimensional mapping with a strange attractor', *Communications in Mathematical Physics*, Vol. 50, No. 1, pp.69–77, DOI: 10.1007/BF01608556.

Li, H., Qin, Z., Zhang, X. and Shao, L. (2010) 'An n-dimensional space audio scrambling algorithm based on random matrix', *Journal of Xi'an Jiaotong University*, Vol. 44, No. 4, p.5.

Lima, J.B. and Neto, E.F.D.S. (2016) 'Audio encryption based on the cosine number transform', *Multimedia Tools and Applications*, Vol. 75, p.8403, DOI: 10.1007/s11042-015-2755-6.

Lin, Q., Yin, F. and Liang, H. (2005) 'Blind source separation-based encryption of images and speeches', Lecture Notes in *Computer Science-Advances in Neural Networks*, Vol. 3497, No. 1, pp.544–549.

Liu, J., Gao, F. and Ma, H. (2008) 'A speech chaotic encryption algorithm based on network', *Proceedings of IIHMSP '08*, IEEE Press, Harbin, China, pp.283–286.

Madain, A., Dalhoum, A.L.A., Hiary, H., Ortega, A. and Alfonseca, M. (2014) 'Audio scrambling technique based on cellular automata', *Multimedia Tools and Applications*, Vol. 71, No. 3, pp.1803–1822.

Mahdi, A. and Hreshee, S.S. (2016) 'Design and implementation of voice encryption system using XOR based on Hénon map', *2016 Al-Sadeq International Conference on Multidisciplinary in IT and Communication Science and Applications (AIC-MITCSA)*, pp.1–5.

Mao, Y., Chen, G. and Lian, S. (2004). 'A novel fast image encryption scheme based on 3D chaotic Baker maps', *International Journal of Bifurcation Chaos*, Vol. 14, No. 10, pp.3613–3624.

Mathews, R. (1989) 'On the derivation of a chaotic encryption algorithm', *Cryptologia*, Vol. 13, No. 1, pp.29–42.

Mosa, E., Messiha, N. and Zahran, O. (2009) 'Chaotic encryption of speech signals in transform domains', *Proceedings of ICCES 2009*, IEEE Press, Cairo, pp.300–305.

Mosa, E., Messiha, N., and Zahran, O., Fathi, E. and El-Samie, A. (2011) 'Chaotic encryption of speech signals', *International Journal of Speech Technology*, Vol. 14, pp.285–296, DOI: 10.1007/s10772-011-9103-7.

Nan, L., Yanhong, S. and Jiancheng, Z. (2004) 'An audio scrambling method based on Fibonacci transformation', *Journal of North China University of Technology*, Vol. 16, No. 3, pp.8–11.

Parvees, M.Y.M. Samath, J.A., Raj, I.K. and Nirmal, R.M. (2017) 'Chaos-based steganocryptic approach to protect medical images with text data of patients', *Journal of Medical Imaging and Health Informatics*, Vol. 7, pp.1–8, DOI: 10.1166/jmihi.2017.1993.

Parvees, M.Y.M., Samath, J.A., Raj, I.K. and Bose, B.P. (2016a) 'A colour byte scrambling technique for efficient image encryption based on combined chaotic map', in *International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT) – 2016*, pp.1067–1072, DOI: 10.1109/ICEEOT.2016.7754851.

Parvees, M.Y.M., Samath, J.A. and Bose, B.P. (2016b) 'Secured medical images – a chaotic pixel scrambling approach', *Journal of Medical Systems*, Vol. 40, p.232, DOI: 10.1007/s10916-016-0611-5.

Patidar, V, Pareek, N.K. and Sud, K.K. (2009) 'A new substitution diffusion based image cipher using chaotic standard and logistic maps', *Communication in Nonlinear Science and Numerical Simulations*, Vol. 14, No. 7, pp.3056–3075.

Peng, X., Cui, Z., Cai, L. and Yu, L. (2003) 'Digital audio signal encryption with a virtual optics scheme', *Optik – International Journal of Light and Electron Optics*, Vol. 114, No. 2, pp.69–75.

Raghunandhan, K.R., Radhakrishna, D., Sudeepa, K.B. and Ganesh, A. (2013) 'Efficient audio encryption algorithm for online applications using transposition and multiplicative non-binary system', *International Journal of Engineering Research and Technology*, Vol. 2, No. 6, pp.472–477.

Rukhin, A. et al. (2001) *A Statistical Test Suite for Random and Pseudo-random Number Generators for Cryptographic Applications*, NIST Special Publication, pp.800–822, National Institute of Standards and Technology, Technology Administration, US Department of Commerce.

Sadkhan, S., Abdulmuhsen, N. and Al-Tahan, N. (2007) 'A proposed analog speech scrambler based on parallel structure of wavelet transforms', *Proceedings of NRSC 2007*, IEEE Press, Cairo, pp.1–12.

Sheu, L. (2011) 'A speech encryption using fractional chaotic systems', *Nonlinear Dynamics*, Vol. 65, Nos. 1–2, pp.103–108.

Stoyanov, B, and Kordov, K. (2015) 'Image encryption using Chebyshev map and rotation equation', *Entropy*, Vol. 17, pp.2117–2139, DOI: 10.3390/e17042117.

Su, Z., Zhang G. and Jiang, J. (2012) 'Multimedia security: a survey of chaos-based encryption technology', Karydis, I. (Ed.): *Multimedia – a Multidisciplinary Approach to Complex Issues*, ISBN: 978-953-51-0216-8, InTech [online] http://www.intechopen.com/books/multimedia-amultidisciplinary-approach-to-complex-issues/multimedia-security-a-survey-of-chaos-based-encryptiontechnology (accessed 13 February 2018).

Tamimi, A.A. and Abdalla, A.M. (2014) 'An audio shuffle-encryption algorithm', *Proceedings of the World Congress on Engineering and Computer Science 2014 Vol I WCECS 2014*, San Francisco, USA, 22–24 October.

Wang, H., Hempl, M., Peng, D., Sharif, H. and Chen, H.H. (2010) 'Index-based selective audio encryption for wireless multimedia sensor networks', *IEEE Transactions on Multimedia*, Vol. 12, No. 3, pp.215–223.

Yang, Y.G., Tian, J. and Xu, P. (2015) 'Quantum assisted encryption for digital audio signals', *Optik-International Journal of Light and Electron Optics*, Vol. 126, No. 21, pp.3221–3226.

Zeng, L., Zhang, X., Chen, L., Fan, Z. and Wang, Y. (2012) 'Scrambling-based speech encryption via compressed sensing', *EURASIP Journal on Advances in Signal Processing*, Vol. 2012, No. 1, pp.1–12.