AN APPLICATION OF TWO-KEY DEPENDENT PERMUTATION FOR AUDIO ENCRYPTION

Arulmani Kuppusamy[1], V. Ramaswamy[1] and Swaminathan Pichai Iyer[2]
[1]Department of Computer Science Engineering, SASTRA University, Srinivasa Ramanujan Centre, Tamil Nadu, India
[2]School of Engineering and School of Computing Sciences, Vels University, Tamil Nadu, India
E-Mail: arulmani_k@rediff.com

ABSTRACT

Speech communication is gaining lots of importance today. Advances in communication technologies especially in mobile technologies have paved the way for users to access Internet using mobile devices particularly mobile phones. The enormous volume of multimedia data being stored, accessed or shared over Internet has created a significant demand for a high level of security. In this paper, a methodology for audio encryption is being proposed. The method uses Two-Key Dependent Permutation algorithm for scrambling the contents of short length uncompressed audio. A high degree of unintelligibility of the encrypted signal is achieved by providing two levels of security to audio contents. The results obtained appear to be satisfactory.

Keywords: multimedia security, permutation, audio encryption, TKDP algorithm, cryptography.

1. INTRODUCTION

Internet has become a part of our daily life. It allows users to store, access and share multimedia data. Since data are shared over public network, the data being exchanged are vulnerable to several attacks that may be active or passive [1]. Users desirous of exchanging confidential multimedia data expect security mechanisms or controls or algorithms which will ensure that the information they share with their counter parts cannot be altered by unauthorized entities. Due to proliferation of portable mobile communication devices, the volume of multimedia data is increasing, inevitably which has necessitated the protection of those contents from adversaries. The protection to sensitive information can be achieved through encryption systems which introduce acceptable level of unintelligibility to the contents so that unauthorized users will not be able to decipher those contents easily.

The widespread use of communication technologies has also led to a remarkable growth of these data in transmission, distribution, and processing of digital multimedia data over wired/wireless channel. The protection of these data against unauthorized access and consumption has become an indispensable goal nowadays in the field of information security. Variety of techniques such as encryption, watermarking, steganography, fingerprinting, etc. has been developed to achieve this goal [2]. Multimedia encryption protects the secrecy of the content and thwarts unauthorized access prior to transmission or distribution of content. Traditional analog techniques are available for speech encryption that accomplishes scrambling in frequency and time domain [3]. Encryption schemes based on notions of chaos theory [4], Blind Source Separation (BSS) [5, 6, 7] and circulant transformation [8] for securing the multimedia data have been proposed and implemented. Chaotic cryptosystems are sensitive to an initial condition and have an important characteristic of exhibiting a complex behavioural pattern and non-discrete broad-band power spectrum. Chaos has prospective uses in functional components of digital communication system such as compression, encryption and modulation. Researchers have evinced great interest in chaos-based multimedia encryption algorithms owing to their fast performances and appropriateness for multimedia data [9].Though the researchers encourage the use of chaotic signal for secure communication, sensitivities of the chaotic system limit its practical applications due to strict self-synchronization [10][11]. Blind Source Separation (BSS) extracts individual source components from their mixtures at multiple sensors. No information other than the mixtures is used to estimate individual source components. For example, [5] discusses the dual key speech encryption algorithm based on BSS and Chen-Lee chaotic system. The system constructs the intractable underdetermined BSS problem in encryption and in decryption. It can only be solvable by dual key signals. Onesignal is generated from PRNG with secret seed and the other is from Chen-Lee chaotic system.

Several methods for encrypting speech/audio exist. Analog encryption is one of the widely used speech/audio encryption techniques used in speech communication. Discrete cosine transform, fast Fourier transform, wavelet transform are some examples of analog speech encryption algorithms used in transform domain. Algorithms for scrambling speech/audio in speech communication are generally categorized into four types [5]: frequency domain scrambling, time-domain scrambling, amplitude domain scrambling, and two-dimensional scrambling that employs both the frequency and time element. Both frequency domain and time-domain scrambling retain the benefits of traditional sample data scrambling, while eliminating the need for synchronization at the receiver end, resulting in simplification of the system structure and noticeable improvement in practicability and reliability in the use of scrambling techniques. The need for synchronization arises only when frame-by-frame scrambling and its reverse are to be done. But, it is required when a frame definition is missing in the operation.

Discrete Fourier Transform has been extensively studied in the past for encryption of speech [12, 13, 14]. Frequency analysis technique that is used in modern telecommunications is Discrete Fourier Transform (DFT).Much of the theories of circulants are indeed related to DFT. Circulants are more useful in digital encoding. They are used to recover from errors in data transmission and storage and to restore the original data [15]. For example, a new analog encryption technique based on unitary circulant transformations of the sampled analog signal, which is less vulnerable to attack than other known schemes based on time domain scrambling or frequency domain scrambling, has been proposed by introducing a phase distortion[8]. An optimal speaker-specific key generation scheme is then developed for maximizing an objective function based on the new distortion measure.

Digital encryption is the best technology available today. In this, a source speech signal is subjected first to digitization followed by encryption and finally the encoded sequence of bits is transmitted. Though a very high level of security is achieved through digital encryption techniques, they have compatibility issues related to synchronization between the transmitter and the receiver with today's communication networks. A synchronization error due to channel damage makes the entire transmissions fail. This problem is resolved by achieving an accurate synchronization between the transmitter and the receiver. In place of digital encryption, scrambling techniques can also be used for encrypting audio/speech and can be employed with current analog communication systems. They scramble the source speech analog signal into a different analog signal with same characteristics of the source speech signal, before transmission. Synchronization is not used in some conventional scrambling techniques but they offer only relatively low levels of security [16]. Improved levels of security can be realized through easily implementable, source to destination synchronization dependent, recently developed advanced techniques compatible with current communication environment. An example of advanced scrambling technique is sample data scrambling in which source signal is first applied to sampling process which extracts a sequence of sample data and then encoded into a different sample data sequence, and translated into a different signal for transmission.

Most of the commercially available analog speech encryption systems use permutation of components as basis either in the time domain or the frequency domain. Hopping window scrambler, band splitter and DFT scrambler and DCT scrambler are based on permutation of components. Analog speech encryption systems once considered to be secure have started showing weaknesses that allow recovery of intelligible speech [17]. The work [17] describes an analog speech scrambler in which speech samples are first converted to a transform domain whose coefficients are subjected to permutation and the permuted transform coefficients are translated back into time domain and sent over the channel. This approach offers high quality output lower residual intelligibility and greater cryptanalytic strength as compared to DFT based speech encryption. Speech encryption with several secret keys for use with the permutation and masking of speech segments in both time and transform domains have been proposed in [18] to produce an encrypted signal without leaving any trace for signal intelligibility.

The time domain sample scrambling method is more attractive and desired for speech/audio encryption as it scrambles a slice of time domain samples into a different slice of samples directly [19]. Concepts of Pseudorandom sequence, Fibonacci transformation [20], Hadamard matrix [21, 22] and so on have been used in speech scramblers for garbling the original signal. All of these methods employ a constant decryption key.

Several conventional encryption algorithms do exist. But they are well suited for textual encryption and are not suitable for speech/audio encryption because of processing overhead involved [23]. Even chaos based total cryptographic algorithms applied to uncompressed multimedia content [24] do not offer security at par with conventional cryptographic algorithms [25]. Del Re *et al.* [26] attribute the amount of security offered by a speech encryption system to residual intelligibility and key space. Higher degree of security is achieved in speech encryption system when it has lower residual intelligibility and larger key space. Depending on one's needs, for protecting audio contents, one can choose either offline or online process for speech/audio encryption.

The objective of our research is to design an audio encryption system based on TKDP permutation that garbles the raw audio bytes of a recorded speech/audio of low bit rate such that no useful information can be retrieved from the encrypted audio. It should also be possible to restore the original speech with no degradation in quality using the decryption algorithm with a correct key pair.

The proposed system deals specifically with encryption of uncompressed audio. The encryption to be applied to the multimedia content may be total or selective. In total encryption, entire multimedia stream (entire audio stream in our scheme) is encrypted without attaching selective importance whatsoever to sample bytes forming the entire audio. The proposed audio encryption employs the total encryption to attain the maximum security level though total encryption tends to consume lots of time. Since the combined use of compression and encryption results in less costly encrypted data, the proposed approach has not used compression.

2. MATERIALS AND METHODS

This section consists of three subsections. The first subsection gives a brief introduction to cryptography and the second one discusses the steps of TKDP algorithm for generating a permutation table of a fixed size. The third section describes the proposed TKDP Audio Encryption.

2.1. Brief introduction to cryptography

Cryptography is the study of methods for converting from plaintexts to ciphertexts and restoring the plaintexts back from ciphertexts. Several encryption schemes [1] and mathematical techniques related to aspects of information security such as confidentiality, data integrity, entity authentication, and data origin authentication [27] play vital roles in the field of cryptography. The scheme is called a cryptographic system or a cipher that allows two people, usually referred to as Alice and Bob, to communicate over an insecure channel so that an adversary cannot understand the contents of what is being sent.

The message Alice wants to send is called the plaintext or clear-text and the disguised message which Bob receives is called the ciphertext[28]. Alice enciphers the clear-text using a prearranged key e and sends the ciphertext to Bob over the channel. Only Bob, who knows the decryption key *d*, can decrypt the ciphertext and reconstruct the clear-text. There are two forms of encryption schemes: symmetric encryption and public-key encryption. In most of the symmetric algorithms, e and d are the same.

Public key encryption uses encryption key different from decryption key. Thesealgorithms require the communicating parties to agree on a secret key before secure communication can take place. The key should be kept secret as long as communication needs to remain secure [29].

Symmetric algorithms are divided into two categories namely stream algorithms and block algorithms. Stream algorithms operate on the plaintext a single bit (or sometimes byte) at a time whereas block algorithms operate on the plaintext in groups of bits called blocks.Most modern block cipher systems apply a number of rounds in succession to encrypt plaintext. Symmetric-key cryptography algorithms have following advantages [1]:

- Reduction in time in encrypting a message is achieved when compared to a public key algorithm.
- Long messages can be encrypted and decrypted using symmetric-key algorithms as the key is normally smaller.

However, Symmetric-key cryptography algorithms have following disadvantages [1]:

- Each pair of communicating parties must have a shared distinct key.
- Key distribution becomes difficult

Two general principles surround all encryption algorithms: substitution and transposition. Substitution is a process in which each element in the plaintext (bit, letter, group of bits or letters) is mapped into another element. It adds confusion to the encryption process by making relationship between the key and the ciphertext complex. In transposition, elements in the plaintext are rearranged. It supplements diffusion to the encryption process by spreading plaintext data over the ciphertext. Encryption normally combines both confusion and diffusion in a round.

Substitution ciphers and transposition ciphers are the two important classes of block ciphers. The iterated product cipher suggested and analysed by Claude Shannon combines simple operations such as substitution and permutation to improve security effectively [30]. This is the basis for design of block ciphers. An Overview of modern cryptography is presented in [31].

2.2. Two-Key Dependent Permutation (TKDP) algorithm

This algorithm [32] generates a permutation table C with size elements by using a key pair (K_1, K_2)chosen by the sender for use with audio encryption step.The permutation table consists of values from 1 to size permuted in some order determined by the keypair(K_1, K_2). The algorithm consists of seven steps namely Initialization, Generation, Replacement, Unique Set Construction, Set Difference Operation, Sorting and Placement. Initialization initializes the variables y and $z$with 1 and p which is the product of K_1andK_2. Initially the set$I = \{1,2,3, \cdots, size\}$ where size denotes the number of letters/bits in the given text blocks and initially C is empty. $\mathrm{w}, \mathrm{x}, \mathrm{y}, \mathrm{z}$are the temporary variables that govern the process of permutation. Generation produces elements $C_1, C_2, C_3, \cdots C_{\mathrm{size}}$and adds them to$C$. During each iterationi, where i = 1,2,3,..size, new value of pis computed using computed $\mathrm{w}, \mathrm{x}, \mathrm{y}, \mathrm{z}$ values followed by the computation of C_i which is $pmodsize$. $\mathrm{C}_i$value is stored in the permutation table CReplacement looks for duplicate elements in the set C generated in theprevious step and replaces them with -1.Unique Set Construction constructs unique set U of positive integers. SetDifference Operation produces a set Mconsisting of elements that are in I but not in U.Sorting sorts the elements of the set M in descending order. Placement replaces with successive elements of$M$ as the elementsof C are scanned from left to right. This process is repeated until the last occurrence of -1 or 0 is replaced with the last element of the set$M$. The set Cnow consists of elements from 1 through size but permuted in some order which depends on $K_1$and K_2. Inthe proposedTKDPAudio Encryption$size$represents the number of bytes in the sampled audio. Figure 1 shows generation of the first element, $C_1$where $1 \in I$.

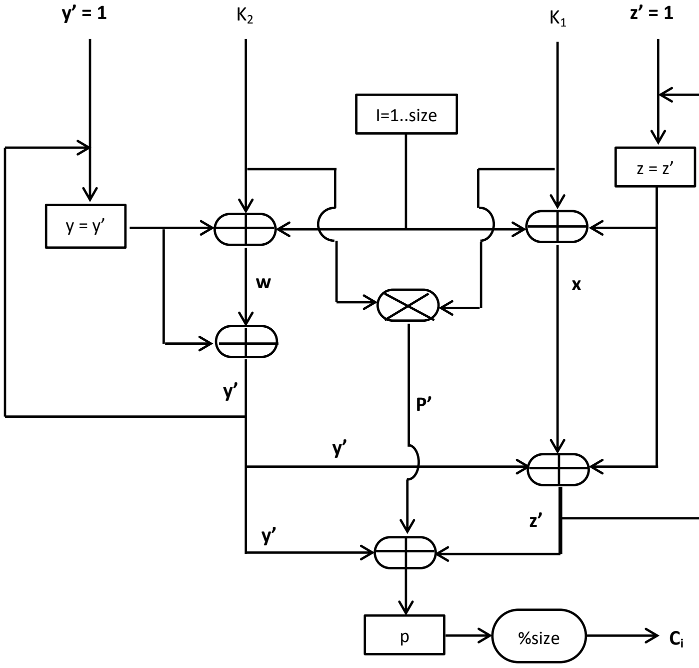


Figure-1. Generation of C_i where $i = 1$ and$i \in I$for a given key pair (K_1, K_2)in step 2.

2.3. Two-Key Dependent Permutation based audio encryption system

A Two Key Dependent Permutation based Audio Encryption Algorithm for encrypting a short length uncompressed audio of low bitrate is proposed in this paper by considering total encryption, lower residual intelligibility and bigger key space. An application of Two Key Dependent Permutation to audio encryption can be understood from the block diagram (Figure-2). The diagram describes the overall process of audio encryption. Four major steps in this algorithmare Speech Acquisition, Audio Data Extraction, Permutation Table, Speech Encryption and Speech Decryption.

Speech Acquisition process interfaces with a microphone and captures digital audio data, often referred to as sampled audio, from the microphone. The stream of audio is stored into audio file in wav format. Prior to recording, it is important to decide about a data format that determines how a series of bytes of raw sampled audio captured from the microphone or read from the sound file data are to be interpreted. For playback or capture, the data format of the sound is specified. The data format specification defines various attributes of sampled sound: encoding technique, number of channels, sample rate, number of bits per sample, frame rate, frame size in bytes and byte order (big-endian or little-endian).

Audio Data Extraction extracts a stream of audio bytes from an audio file. A stream of audio refers to continuous set of audio bytes that arrive at the same rate that they are to be played or captured. Since our approach for audio encryption uses the streaming model for audio transport, duration of the sound and its finishing time of arrival are not known in advance. So, a buffer is handled at a time for capturing audio data that can be read in chunks until the operation is halted. Audio Encryption step consists of (1) Initial Permutation Table (IPT) generation, (2) Second Permutation Table (SPT) construction, (3)One -Time Pad (OTP) Table generation,(4) Apply IPT operation and (5) a simple arithmetic addition operation involving Permutated Audio Data and OTP table, both represented as one-dimensional tables (i.e., arrays). The Initial permutation table generation step generates a permutation table using TKDP Algorithm for rearranging sample values (amplitude values) along time dimension. The application of IPT to raw audio data of the original signal obtained from the audio extraction process results in a Permuted Audio Data (PAD).The Second Permutation Table is derived from IPT again by using TKDP algorithm while the OPT table is constructed using SPT elements subjected to alternate sign conversion operation and modulus 256 operation, and then a byte conversion operation.

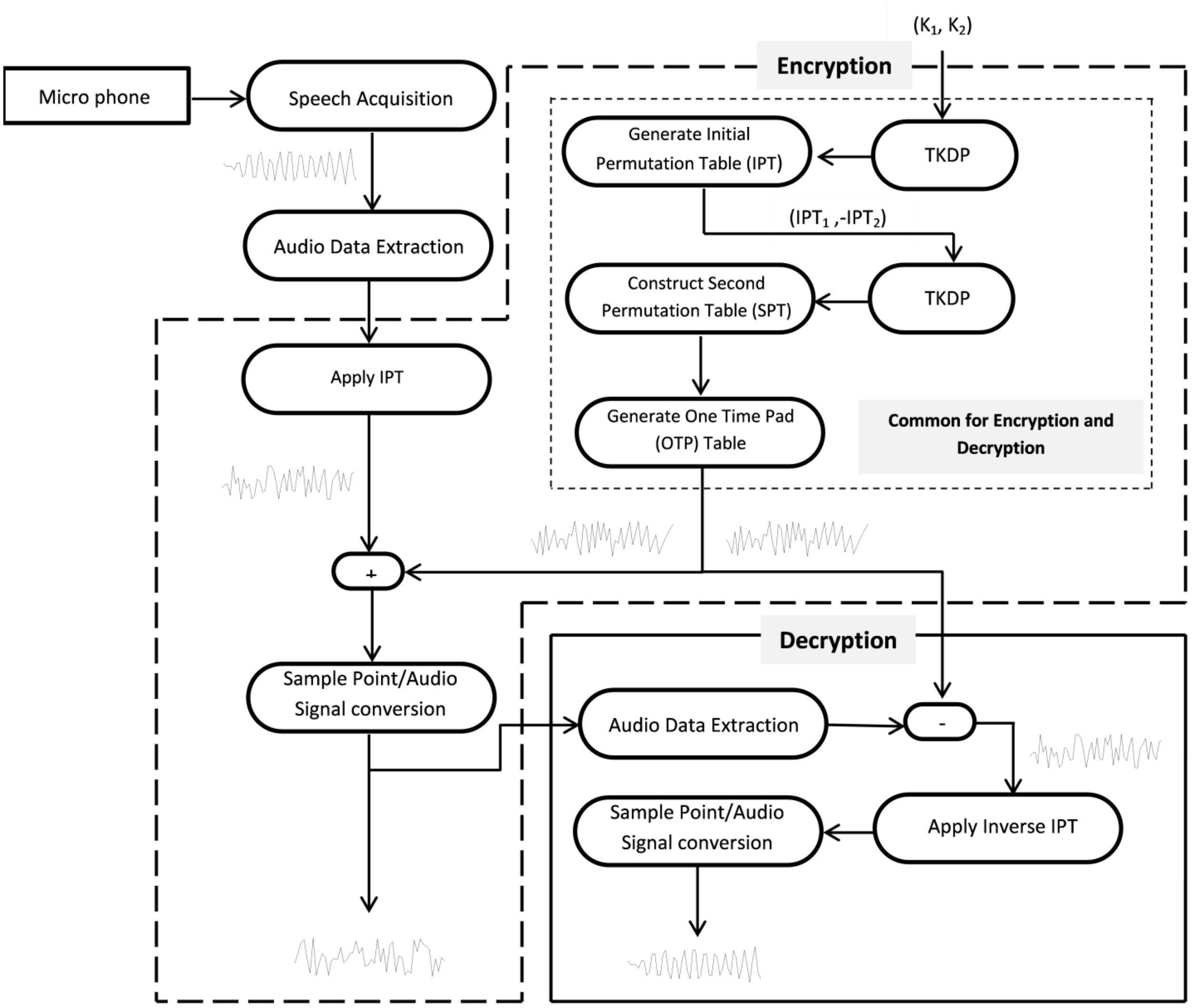

(K1, K2)
Encryption
Micro phone
Speech Acquisition
Audio Data Extraction
Generate Initial Permutation Table (IPT)
TKDP
(IPT1 ,-IPT2)
Construct Second Permutation Table (SPT)
TKDP
Apply IPT
Generate One Time Pad (OTP) Table
Common for Encryption and Decryption
+
Sample Point/Audio Signal conversion
Decryption
Audio Data Extraction
-
Sample Point/Audio Signal conversion
Apply Inverse IPT

Figure-2.TKDP audio cryptographic algorithm.

In addition to first three common steps of encryption, decryption includes two important processes: simple arithmetic subtraction involving the output of encryption and OTP table both represented as one-dimensional arrays and Inverse Permutation. Subtraction process yields the same PAD obtained during step (5) of the encryption process and the inverse permutation is done on PAD to get raw audio data of the original signal. Finally, raw audio data is subjected to sample point conversion followed by the audio signal conversion function to obtain the original signal back. In Figure-2, the steps of encryption are surrounded by long dash lines whereas the decryption process steps are enclosed by solid lines. Steps common for encryption and decryption are shown inside a box drawn with dashed lines.

The detailed description of TKDP Audio Encryption Algorithm is presented below:

Speech acquisition

This step uses the offline process for speech acquisition using microphone. It records, plays and stores the recorded speech as wav file. When sound is digitized, parameters such as sampling rate, bit depth and number of channels are considered to determine the amount of information stored in an audio file and the quality of the digital sound. Sampling rate defines the number of times the analog sound is sampled during each period and converted into digital sound. An approximation of true analog version of sound is done by considering higher sampling rates. Bit depth determines the range of possible integers that represent the amplitude of individual samples. The more bit depth, the higher quality of digitizing sound. Each integer in the range specifies a specific discrete level corresponding to a particular sample's amplitude. When sound is recorded in digital format, a sampling rate and a bit depth, the number of bits used per sample and number of channels are specified. It is important to balance the sampling rate, and bits per sample to come up with an acceptable quality of sound with minimum file size. The default initial settings for speech acquisition in our sound proof recording environment are a sampling rate of 44.100 kHz, bit depth of 16 bits per sample, Linear PCM encoding that uses linear quantization of amplitude and signed or unsigned integer values, little Endian byte order and 2 channels. Linear quantization means that the number stored in each sample is directly proportional to the original sound pressure at that instant and similarly proportional to the displacement of a loudspeaker that is vibrating with the sound at that instant.

Audio data extraction

This step reads a stream of raw audio bytes from the local buffer of audio captured from microphone or from a sound file. The audio data thus read is written to an array of bytes using Java Sound API for later manipulation. The manipulated array of samples refers to the encrypted raw audio data. The choice of buffer length (normally expressed in bytes) is significant in reducing the risk of buffer underflow and in avoiding unnecessary consumption of CPU resources. An optimal buffer size should be chosen to minimize latency to an acceptable level considering the application requirements. A buffer of size 4096 bytes should be used to avoid the risk of buffer overflow which might cause audible clicks and skips in the sound when capturing or recording audio data to a sound file. The same buffer length can be considered to avoid gaps in the sound during playback. It is to be noted that the buffer length chosen corresponds to an integral number of frames.

Encryption

Before discussing the steps used in encryption, we will look at how a permutation table is generated using TKDP algorithm fed with a key pair(K_1, K_2). TKDP algorithm generates a permutation table of unique elements in the range of 1 to *size* where *size*is the length (in bytes) of raw sampled audio bytes array.

Encryption process consists of the following sub steps:

a) Generate Initial Permutation Table (IPT) using TKDP Algorithm with a secretly shared key pair$(K_1, K_2)$and construct a Second Permutation Table (SPT).The Initial Permutation Table forms a basis for constructing the SPT. SPT is generated using a key pair, $(IPT_1, -IPT_2)$where IPT_1 and $IPT_2$are the first two values in IPT. SPT elements in even positions are converted into negative.

b) Apply IPT to Raw Audio Data (RAD) which is an array of audio bytes of Original Signal to obtain Permuted Audio Data (PAD) which is also an array of sound bytes, of the original signal.

c) Generate One-Time Permutation table (OTP): All the elements in SPT are performed mod 256 operation and resultant values are narrowed down to byte values. These values build a OTP, thought of as a one-time pad, which is later converted into an audio signal in the encryption process after the elements in OTP have been converted into sample points.

d) Subtract each OTP element from the corresponding PAD element: In this step, every element in the OTP is subtracted from the corresponding element located at the same position in PAD to cause changes in the values of scrambled array of audio bytes. Audio signal encrypted this way is difficult for cryptanalysis. Since each new audio input data requires a new keyi.e., OTP table of the same length as the new audio data, OTPis considered unbreakable. When OTP is subjected to the scrambled array of samples, the output generated by audio encryption step has no statistical relationship to the source audio signal. As the encrypted audio signal does not embody any useful information whatsoever about the original audio, it is crypt-analytically hard for an adversary to break the audio output. Encrypted audio is then transported to the destination.

e) Compute Endian values and sample points: The Endian values and their sample points for output table produced in this step are computed to get encrypted audio signal, which is of wav format, of the original signal. Encrypted audio is then sent to the receiver.

Decryption

Decryption consists of the following steps:

a) Generate IPT and SPT: Prior to decryption, the receiver should have generated the Initial Permutation Table with TKDP Algorithm using the key pair (K_1, K_2) obtained already from the sender by secure means, followed by SPT, using a key pair(IPT_1, IPT_2). The signs of elements in SPT are changed alternatively.

b) Generate OTP table: The receiver subjects each element of SPT to mod 256 operations. If the modulus operation results in a positive integer less than 128, the resultant value is retained.Else the value is subtracted from 256. If the resultant value is less than -128, the value is added to 256. Each value computed this way forms the same OTP table used by the sender during encryption at the receiver end.

c) Add OTP with Raw Audio Data of the received encrypted signal: Each element in the OTP table is added to the corresponding element inthe array of raw audio bytes of the encrypted signal to produce an intermediate table consisting of the same elements of PAD of the original signal.

d) Apply inverse permutation on the output signal from step 3: In this step, inverse permutation table for the IPT is generated and applied to the output table generated in step 3 to yield a table populated with raw audio bytes similar to that of original signal prior to permutation.

e) Compute Endian values and sample points: The Endian values and their sample points for intermediate table produced in step4 are computed so as to retrieve the original signal.

3. RESULTS AND DISCUSSIONS

In our work, we have used Java Sound API facilities for acquisition of raw audio data during recording and writing it to a sound file for later reading and manipulation.

Initially, in speech acquisition stage, a two word utterance “Good Morning” is recorded at a bit depth of 16 bits with a sampling frequency of 44100Hz, PCM encoding, little-Endian byte order and 2 channels. When the signal is being acquired using microphone, sample bytes (raw audio data) of the signal are first buffered consecutively without any significance being attached to the structure of the sound file. The contents of the buffer are written to a WAV sound file called GoodMorn_LI.wav along with format information. This audio becomes the source/original signal to which TKDP Audio Encryption Algorithm is applied for encryption. Reading the WAV file which is used in Audio Encryption involves the following three steps:

a) A stream object is obtained from the sound file (GoodMorn_LI.wav).
b) A temporary byte array of an arbitrary fixed size equal to integral number of frames is created to store successive chunks of audio data from the stream object.
c) Bytes from the stream are read into the array whose elements are subjected to processing by various steps of TKDP Audio Encryption.

In the Audio Data Extraction stage, first 100 bytes of raw audio data of GoodMorn_LI.wav(see Table-1) are extracted from the buffer. The array of bytes must be converted to Endian format defined by the original encoding of the sampled sound. Since the encoding scheme used for the recorded audio is signed 16-bit PCM, Endian values are stored in the format of 16 bits or 2 bytes. A pair of byte values from the byte array which is a single Endian value is taken to form a signal data at given point in time. Table-2 given below shows 50Endian values constructed from the extracted 100 bytes. The values in Table-2 given below are then converted into sample points of the original signal as shown in Table-3.

Table-1. A set of 100 raw audio bytes (Raw Audio Data) of GoodMorn_LI.wav file.

1	0	0	0	0	0	0	0	-1	-1	1	0	1	0	-3	-1	-2	-1	3	0
3	0	-3	-1	-3	-1	3	0	4	0	-2	-1	-4	-1	2	0	1	0	-4	-1
1	0	4	0	-2	-1	-4	-1	4	0	4	0	-4	-1	-3	-1	3	0	3	0
-3	-1	-3	-1	3	0	2	0	-3	-1	-1	-1	3	0	-1	-1	-3	-1	4	0
2	0	-4	-1	-1	-1	4	0	0	0	-5	-1	1	0	5	0	-2	-1	-4	-1

Table-2. Endian Values for the extracted 100 raw audio bytes of original signal.

1	0	0	0	-1	1	1	-3	-2	3	3	-3	-3	3	4	-2	-4	2	1	-4
1	4	-2	-4	4	4	-4	-3	3	3	-3	-3	3	2	-3	-1	3	-1	-3	4
2	-4	-1	4	0	-5	1	5	-2	-4										

Table-4 which is an Initial Permutation Table (IPT) is generated and applied to Raw Audio Data (RAD) stored as a byte array, to obtain Permuted Audio Data (PAD) (Table-5), which is also represented as an array. Before the occurrence of table construction process, the size of the permutation table has to be determined.

Table-3. Sample points computed for Endian values of original signal.

0.0000305176	0.0000000000	0.0000000000	0.0000000000	-0.0000305176
0.0000305176	0.0000305176	-0.0000915527	-0.0000610352	0.0000915527
0.0000915527	-0.0000915527	-0.0000915527	0.0000915527	0.0001220703
-0.0000610352	-0.0001220703	0.0000610352	0.0000305176	-0.0001220703
0.0000305176	0.0001220703	-0.0000610352	-0.0001220703	0.0001220703
0.0001220703	-0.0001220703	-0.0000915527	0.0000915527	0.0000915527
-0.0000915527	-0.0000915527	0.0000915527	0.0000610352	-0.0000915527
-0.0000305176	0.0000915527	-0.0000305176	-0.0000915527	0.0001220703
0.0000610352	-0.0001220703	-0.0000305176	0.0001220703	0.0000000000
-0.0001525879	0.0000305176	0.0001525879	-0.0000610352	-0.0001220703

Table-4. Initial Permutation Table (IPT) generated using a key pair (73, 23) with TKDP algorithm.

1804	2226	3343	6051	12342	26600	58393	128457	140344	48678
57747	37247	60010	91860	128125	4581	70860	124466	73687	79419
23134	57216	136353	34241	14752	62910	51163	93687	28434	20588
108997	70557	127540	85770	114015	111731	130630	74136	54697	61689
26184	138326	23203	101871	30202	134340	131853	128597	125404	125538
138727	49707	122386	104688	128335	43535	42448	95186	112747	76711
52574	52708	40405	40269	40132	39994	39855	39715	39574	39432
39289	39145	39000	38854	38707	38559	38410	38260	38109	37957
37804	37650	37495	37339	37182	37024	36865	36705	36544	36382
36219	36055	35890	35724	35557	35389	35220	35050	34879	34707

There are two simple ways in which permutation table size can be computed. First approach uses *frame rate* and *frame size* values to determine the size of the permutation table. The *size* is equal to *frame rate* multiplied by *frame size*. These values can be retrieved from the header of the original signal.

Table-5. Permuted Audio Data (PAD) of the original signal.

18	-40	-25	-15	126	-9	8	-93	0	127	-121	-7	49	38	124
-127	-18	27	117	-41	120	-8	-42	-122	-110	-98	-84	27	-58	8
-111	-68	-58	-61	-6	122	9	127	-68	98	95	2	54	-123	-77
-7	-64	66	-44	-62	-26	100	36	-54	-1	-66	63	101	-37	45
17	126	70	-106	-91	25	9	-71	126	127	-6	-45	-126	-72	-11
37	39	102	-50	-124	-75	-126	83	23	-1	50	9	97	119	-62
105	72	-62	-23	-109	-62	107	69	-48	80					

Table-6. Second permutation table generated with the key pair (1806,-2226).

66986	79363	120105	94248	111896	5481	75515	77518	89442	130935
109901	2724	83324	130645	140463	133386	67622	116931	14561	115264
82656	115713	97955	37638	10346	3151	85365	21468	3332	47085
14231	132994	35694	19995	65993	32376	139112	137881	131211	111294
60146	78183	74525	130708	89676	126021	31361	8070	56246	55053
100431	117136	45968	115983	72115	123574	2234	63871	92037	5388
112468	2851	92263	94130	95998	97867	99737	101608	103480	105353
107227	109102	110978	112855	114733	116612	118492	120373	122255	124138
126022	127907	129793	131680	133568	135457	137347	139238	10	1903
3797	5692	7588	9485	11383	13282	15182	17083	18985	20888

Other approach uses the *byte array length* and *frame size*. In our work, the second approach is used and the size of the permutation table is computed as *buffer length/frame size*. IPT is generated with the help of TKDP algorithm using a key pair (73, 23) and is shown in Table-4. In the OTP generation step, SPT as shown in Table 6 is produced by a key pair (1804,-2226) consisting of the value of the first element and the inverted value of the second element in the IPT. The values of elements in the SPT are inverted alternatively. After subjecting each value to (mod 256) operation, it is then converted to a byte value. These values make up the OTP table (Table-7).In the encryption step, each value in the OTP table is subtracted from the corresponding value in PAD table to obtain the encrypted version of raw audio data of the original signal as shown in Table-8. Endian values and the sample points of the encrypted signal are shown in Tables 9 and 10.

Table-7. One Time Pad (OTP) table.

-86	-3	41	-40	24	-105	-5	50	98	-119	77	92	124	-85	-81
-10	38	61	-31	-64	-32	-1	-93	-6	106	-79	117	36	4	19
-105	126	110	-27	-55	-120	104	103	-117	66	-14	-103	29	108	76
-69	-127	122	-74	-13	79	112	-112	-15	-77	74	-70	-127	-123	-12
84	-35	103	78	-2	-75	-103	24	56	119	-37	-46	-126	41	45
124	-36	-53	-113	22	70	93	1	-96	-64	-33	-125	26	10	-111
-43	-60	-92	-13	119	30	78	69	41	104					

The same IPT, SPT and OTP tables used during encryption process are regenerated in the decryption process. Each element in the OTP (Table-7) is added to the corresponding element in the array of audio bytes of encrypted signal (Table-8) as a result of which an intermediate table exactly similar to Table-5 is obtained. Inverse permutation (Table-11) is applied to the intermediate table to obtain a final output table consisting of raw audio bytes analogous to that of original signal as shown in Table-1. Endian values and the sample points of the final output table are computed. It is observed that the signal obtained using these sample points at the receiving end is similar to the original signal (Tables 2 and 3).

Table-8. Subtracting OTP from PAD.

104	-37	-66	25	102	96	13	113	-98	-10	58	-99	-75	123	-51
-117	-56	-34	-108	23	-104	-7	51	-116	40	-19	55	-9	-62	-11
-6	62	88	-34	49	-14	-95	24	49	32	109	105	25	25	103
62	63	-56	30	-49	-105	-12	-108	-39	76	116	-123	-28	86	57
-67	-95	-33	72	-89	100	112	-95	70	8	31	1	0	-113	-56
-87	75	-101	63	110	111	37	82	119	63	83	-122	71	109	49
-108	-124	30	-10	28	-92	29	0	-89	-24					

Table-9. PAD-OTP Endian values.

-9368	6590	24678	28941	-2402	-25286	31669	-29747	-8504	6036
-1640	-29645	-4824	-2249	-2622	16122	-8616	-3535	6305	8241
26989	6425	15975	-14273	-12514	-2921	-9836	29772	-7035	14678
-24131	18655	25767	-24208	2118	287	-28928	-22072	-25781	28223
9583	30546	21311	18310	12653	-31596	-2530	-23524	29	-5977

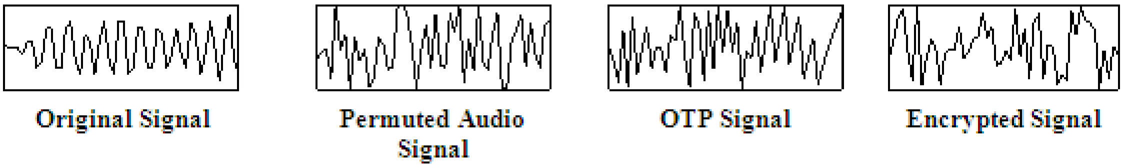
Table-10. Encrypted signal sample points.

-0.28588867	0.20111084	0.7531128	0.8832092	-0.07330322
-0.7716675	0.9664612	-0.9078064	-0.25952148	0.1842041
-0.050048828	-0.9046936	-0.1472168	-0.06863403	-0.08001709
0.4920044	-0.26293945	-0.10787964	0.19241333	0.25149536
0.8236389	0.19607544	0.4875183	-0.4355774	-0.38189697
-0.089141846	-0.3001709	0.90856934	-0.21469116	0.447937
-0.7364197	0.5693054	0.78634644	-0.73876953	0.06463623
0.008758545	-0.8828125	-0.673584	-0.7867737	0.8612976
0.29244995	0.93218994	0.6503601	0.55877686	0.38613892
-0.9642334	-0.07720947	-0.7178955	8.8500977E-4	-0.18240356

Table-11. Inverse permutation for IPT.

141120	141119	141118	141117	141116	18349	141115	141114	141113	141112
24944	141111	141110	141109	141108	141107	141106	141105	141104	16158
141103	141102	141101	141100	141099	141098	141097	141096	141095	141094
141093	141092	141091	141090	141089	141088	141087	27885	141086	141085
16604	141084	141083	141082	141081	141080	141079	141078	141077	141076
141075	141074	79355	141073	141072	141071	141070	141069	141068	14017
141067	141066	141065	141064	141063	141062	141061	141060	141059	141058
141057	141056	141055	141054	141053	141052	141051	141050	141049	17462
141048	141047	31215	141046	141045	141044	141043	141042	141041	141040
141039	141038	141037	141036	1368	141035	141034	141033	141032	141031

Sample point graphs for 50 sample points of original signal, Permuted Audio Data, OTP and encrypted signal are shown in Figure-3. The Audacity sound editing software screen shots for the entire portion of the Original signal (GoodMorn_LI.wav file), PAD signal, OTP signal and encrypted signal have been cropped and are shown in Figure-4. All these signals have the same attributes of the original signal with 44100Hz sampling rate, 16-bit bit depth and 2 channels. The centre horizontal line indicates zero amplitude. Points above the line are positive-valued samples whereas points below are negative.



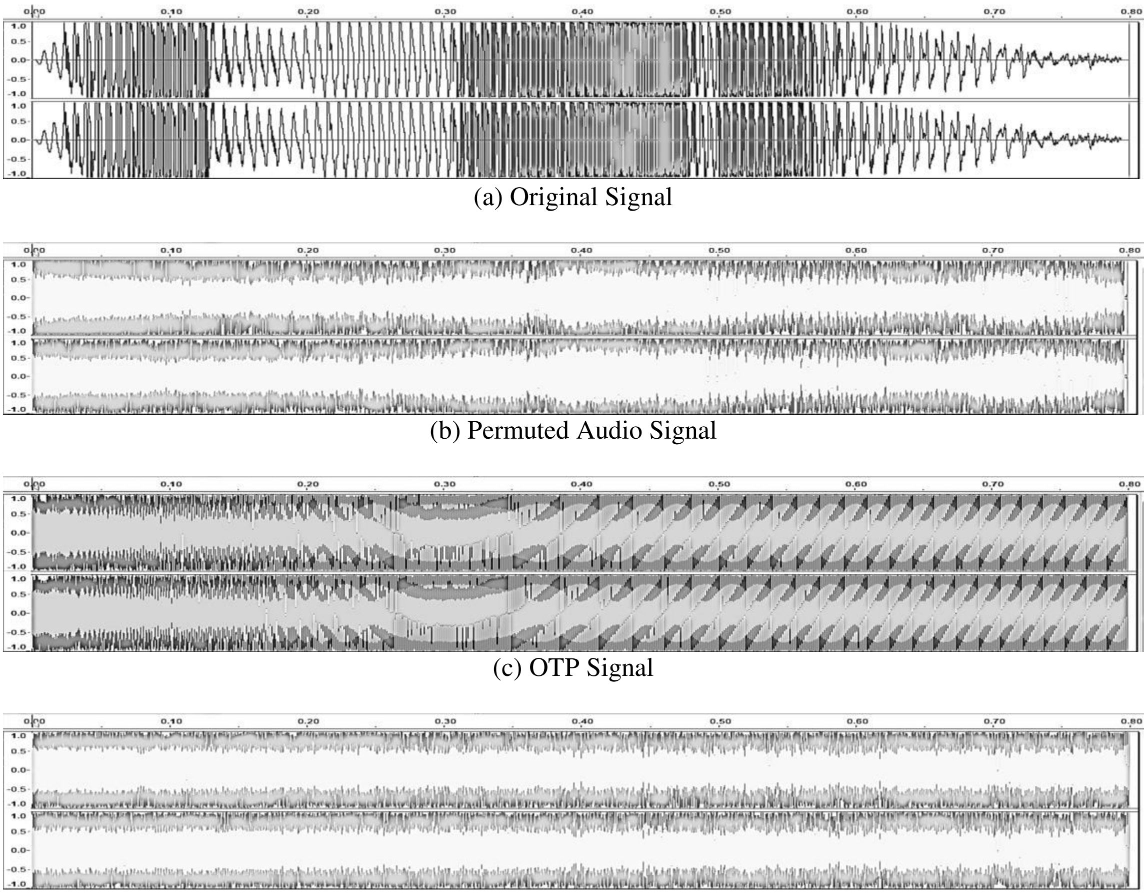
Original Signal

Permuted Audio Signal

OTP Signal

Encrypted Signal

Figure-3. Sample point graphs.



(a) Original Signal

(b) Permuted Audio Signal

(c) OTP Signal

(d) Encrypted Signal

Figure-4. Cropped audacity screen shots: (a) Original Signal, (b) PAD, (c) OTP and (d) Encrypted signals.

Security aspects

Cryptography provides information security but cryptanalysis damages it. Cryptanalysis is a study of methods that compromises information security provided by cryptography. When a key is not known, an adversary tries to use cryptanalytic attacks to deduce the key for recovering the original message. The adversary chooses cryptanalytic attacks based on the amount of information and the level of control over the system [31]. Basic cryptanalytic attacks are ciphertext-only attack, known-plaintext attack, chosen plaintext attack and chosen ciphertext attack. In ciphertext-only attack, the adversary has access to one or more encrypted messages. Any cryptosystem that is compromised under such an attack is insecure. In a known-plaintext attack, the adversary has some knowledge about the plaintext corresponding to the given ciphertext so that the key or a part of it can be determined to recover the entire message encrypted with that particular key. In the chosen-plaintext attack, the adversary attempts to obtain a key or at least part of it using accumulated knowledge of plaintext-ciphertext (generated by the encryption algorithm and the encryption key for the chosen plaintext) pairs. In the chosen-ciphertext attack, the adversary chooses a ciphertext and obtains its corresponding decrypted plaintext generated with a secret key and analyses the plaintext-ciphertext pairs to guess the secret key or a part of the key.

The cryptanalysis rests on the availability of the scrambled speech and uses speech characteristics to decipher it. Despite the fact that speech has more redundancy than written text, the redundancy could not be used for cryptanalysis of encrypted speech because much of the redundancy is specific to understanding of the listener. A good speech encryption prevents eavesdropping of speech signals to be used for feature extraction. It attempts to do a completely reversible operation on speech to be totally meaningless to any illegal listener [33]. A problem with speech encryption is that a small residue of the original signal allows trained listener to directly interpret the encrypted audio/speech [34].

The proposed algorithm is best suited for encrypting short length audio of low bit rate. The performance of the algorithm depends on the choice of three important parameters: sample rate, bit depth and number of channels. The speed at which audio encryption is done is higher when the values for performance parameters are less. In our work, we have considered parameters whose values are comparable or equal to the CD quality of sound. The recorded speech/audio consists of a collection of words, gestures, tone of voice and silence (amplitude of zero value). When it is subjected to the encryption process, it brings about a high degree of temporal distortion, contributing to the unintelligibility of the encrypted signal.

The algorithm offers two-layered encryption to the original signal to provide two different levels of security. The first level of security is achieved by applying the raw audio data of the original signal to an initial permutation, resulting in a PAD and the second level of security is realized by constructing OTP by subjecting elements of SPT derived from the IPT to modulus and byte conversion operations. The degree of unintelligibility of the encrypted signal is augmented when the OTP is added to PAD. Both IPT and SPT tables are dependent on a key pair. To convert the original signal into encrypted signal, all the three tables namely IPT, SPT and OPT are required.

As the arbitrarily chosen key pair (K_1, K_2)used for generating IPT is kept secret by the sender and receiver, constructing SPT and OTP tables without knowledge of the actual key pair by an adversary to compromise the security of the algorithm will become very difficult. The values of the key pair (K_1, K_2) do not have any relationships whatsoever except that they belong to a set of integers. Because there are an infinite number of key pairs that can be chosen arbitrarily from the set of integers there is no guarantee that guessing one key pair and applying it to encryption process will break the security of the algorithm. The only possible attack which uses the exhaustive key search is the brute force attack. The key space must be large enough to make any cryptosystem secure against the brute force attack [35]. The brute force attack can try all possible integer key pairs from an infinite set of integers or it can make use of the length (number of raw audio bytes) of encrypted signal. The proposed system encrypts original audio signal by applying permutation, generated with a key pair using TKDP algorithm, to its raw audio bytes. The number of raw audio bytes making up an encrypted audio signal is quite large even for an audio of short duration and therefore it requires an adversary to try all possible 2^n! attempts to decrypt the signal where n means the number of raw audio bytes in the encrypted signal. Hencebrute force attach is also computationally infeasible.

The proposed system is secure against all types of cryptanalytic attacks mentioned above. In the speech cryptosystem, if a cryptanalyst has knowledge of both the original audio signal and its encrypted version, the analysis using accumulated knowledge of original signal and encrypted signal will no way help him to deduce the key pair. To decrypt the encrypted signal, cryptanalyst must know the key pair to construct the Initial Permutation Table that forms basis for generating SPT which is then used to construct One-Time Pad (OTP) table and corresponding Inverse Permutation table. If he tries with a different OTP and Inverse Permutation, the system will output entirely wrong results. Therefore the knowledge of the original audio signal and encrypted signal are not sufficient to guess the key. Initial permutation is applied to audio bytes (speech samples) in time domain so as to achieve a distortion of the speech time envelope which lessens the degree of speech intelligibility. But, some portions of the permuted audio data will remain intact, which might allow trained listener to discern the permuted audio data. Hence OTP table is generated which causes changes to permuted and non-permuted speech samples. As a result, the security of TKDP audio encryption is increased.

A ciphertext-only attack, in which a cryptanalyst is able to access only a scrambled version of a speech is of no use because OTP table and Inverse permutation Tables are not available. Since such an attack has knowledge about the ciphertext (scrambled speech) of a speech, the attack is bound to fail. In other words, the system is secure and can withstand automated cryptanalysis that uses cipher-text attacks [36] which have been subjected to analog speech scramblers such as hopping window scrambler, band splitter [3] and DFT scrambler [14].Security lacks in permutation based schemes if the permutation table is not dynamic [37]. The system does not make use static permutation table. Since the permutation table is dynamically generated using the randomly chosen key pair for encryption of each audio, the system is insusceptible to various sorts of assaults like known plaintext and chosen plaintext attacks.

Performance enhancements through parallelism

Instead of performing operations related to generation of IPT, SPT and OTP tables in a sequential manner, these table generation operations can be parallelized. Once the first two elements of IPT have been generated, generating succeeding elements of IPT and elements of SPT can be done in parallel. OTP table generation can also be synchronized with elements of SPT. Because of the scopes for simultaneous execution of processes common to encryption and decryption in TKDP Audio Encryption, processors that support parallel processing can be utilized efficiently to produce IPT, SPT and OTP tables concurrently. Both hardware and software efficiency can be achieved in this manner. The common functionality of encryption and decryption is not at all dependent on any processes and therefore it can be separated from encryption and decryption and done in parallel to enhance the throughput if sufficient support for parallelism, adequate memory and security for the common processing module areensured. If this separation is done, the both encryption and decryption are so simplified that they will consist of less number of operations.

4. CONCLUSIONS

TKDP Audio Encryption is very suitable for encrypting audio of short duration recorded in resource constrained communication devices like mobile phones in offline fashion. Our technique ensures a high degree of incomprehensibility is attained in encrypted audio. There is also a significant distortion in the encrypted audio due to key-dependent permutations which make the brute force attack infeasible. To ensure that the algorithm is able to withstand all kinds of cryptanalytic attacks, the key pair (K_1, K_2),IPT and Inverse Permutation table generated therefrom are kept secret. The efficiency of the system can be improved through parallelism introduced in the generation of IPT,SPT and OPT tables, which results in faster encryption/decryption. There is no degradation in quality in the resultant audio. It matches with the original audio/speech. Therefore the TKDP Audio Encryption can be used to encrypt the recorded audio/speech using the key pair $(K_1, K_2)$and the encrypted audio can be transmitted to the destination by secure means. The encrypted audio received at the destination can be decrypted using the same pair (K_1, K_2) for recovering the original audio/speech.

REFERENCES

[1] W. Stallings. 2011. Cryptography and Network Security. Prentice Hall.

[2] Nidhi S. Kulkarni, Balasubramanian Raman and Indra Gupta. Multimedia Encryption: A Brief Overview, Rec. Advan. inMult. Sig. Process. and Commun. SCI. 231: 417-449.

[3] Beker H.J., Piper F.C. 1985. Secure Speech Communications. London Academic Press Inc.

[4] Li K., Soh Y.C., Li Z.G. 2003. Chaotic Cryptosystem with High Sensitivity to Parameter Mismatch.IEEE Trans Circuits and Systems I: Fundamental Theory and Applications. pp. 579-583.

[5] Huan Zhao, Shaofang He, Zuo Chen and Xixiang Zhang. 2014. Dual Key Speech Encryption Algorithm Based Underdetermined BSS. The Scientific World Journal. 2014(Article ID 974735): 7. doi:10.1155/2014/974735

[6] AtefMermoul. 2012. Adel Belouchrani, Subspace-based technique for speech encryption. Digital Signal Processing.22(2): 298-303, ISSN 1051-2004, http://dx.doi.org/10.1016/j.dsp.2011.10.010.

[7] Qiuhua Lin, Fuliang Yin andHualou Liang. 2005. Blind Source Separation-Based Encryption of Images and Speeches.LNCS 3497. pp. 544-549.

[8] G. Manjunath and G. V. Anand. 2002. Speech encryption using circulant transformations. Proc. IEEE Int. Conf. Multimedia and Expo. 1: 553-556.

[9] Yun Peng Zhang, Fan Duan, and Xi Liu. The Research of Applying Chaos Theory to Speech Communicating Encryption System Advances in MSEC. Vol. 2, AISC 129, pp. 197-202.

[10] H Li-Lian, Y Qi-tian. 2009. A chaos synchronization secure communication system based on output control. J. Electron. Inf. Technol. 31(10): 2402-2405.

[11] T Liangrui, Z Lin, Y Xue. 2010. Chaos synchronization based on observer and its application in speech secures communication, in Proceedings of IC-NIDC,Ed. by 2ndEdn. (Beijing, China, 2010). pp. 773-777.

[12] Matsunaga A., Koga K., Ohkawa M. 1989. An analog speech scrambling system using the FFT technique with high level security. IEEE Journal in Selected Areas in Communications. 7: 540-547.

[13] Sakurai, K., Koga, T., Muratani, T. 1984. A speech scrambler using Fast Fourier Transform Techniques. IEEE Selected Areas in Communications SAC-2: pp. 434-442.

[14] S. Sridharan, E. Dawson and B. Goldburg: 1991. A Fast Fourier Transform Based Encryption System. IEEE Proceedings-I. 138(3): 215-223.

[15] Circulants. A. Wyn-jones. Manuscript for the complete text available at www.circulants.org.

[16] Chris Edwards, FPGA-based speech encrypting and decrypting embedded system, http://www.techdesignforums.com/practice/technique/fpga-based-speech-encrypting-and-decrypting-embedded-system/.

[17] B Goldburg, E Dawson, S Sridharan. 1993.A secure analog speech scrambler using the discrete cosine transform. Advances in Cryptology - ASIACRYPT '91,LNCS 739, pp. 299-311,Springer-Verlag.

[18] E. Mosa, N. W. Messiha, O. Zahran, F. E. Abd El-Samie. 2010. Encryption of speech signal with multiple secret keys in time and transform domains. International Journal of Speech Technology. 13(4): 231-242, Springer.

[19] Zeng *et al*. 2012. Scrambling-based speech encryption via compressed sensing, EURASIP Journal on Advances in Signal Processing. 2012:257.

[20] L Nan, S Yanhong, Z Jiancheng. 2004. An audio scrambling method based on Fibonacci transformation. J. North China Univ. Technol. 16(3): 8-11.

[21] V Senk, VD Delic, VS Milosevic. 1997. A new speech scrambling concept based on Hadamard matrices. IEEE Signal Process. Lett. 4(6): 161-163.

[22] SK Pal. 2007. Fast, reliable & secure digital communication using Hardmard matrices, in

Proceedings of the International Conference on Computing: Theory and Applications, ed. by 1stEdn. (Kolkata, India). pp. 526-532.

[23] Zhao L. 2006. Speech Signal Processing. Machinery Industry Press.

[24] Goce J., Ljupco K. 2001. Chaos and cryptography: block encryption ciphers based on chaotic maps. IEEE Trans. Circuits and Systems-I: Fundamental Theory and Applications. 48(2): 163-169.

[25] Jakimoski, G., Kocarev, L. 2001. Analysis of some recently proposed chaos-based encryption algorithms. Phy. Letters 291(6): 381-384] [Michael, G., Andreas, U., Wild, P. 2007. Transmission error and compression robustness of 2D chaotic map image encryption schemes. EURASIP J. Inf. Security.

[26] E Del Re, R Fantacci, D Maffucci. 1989. A new speech signal scrambling method for secure communications: theory, implementation, and security evaluation. IEEE J. Sel. Areas Commun. 7(4): 474-480.

[27] Alfred J. Menezes, Paul C. van Oorschot, Scott A. Vanstone. 1996. Handbook of Applied Cryptography. CRC Press.

[28] Neal Koblitz. 1994. A Course in Number Theory and Cryptography. Springer-Verlag.

[29] B. Schneier. 1996. Applied Cryptography. Wiley.

[30] Shannon C. 1949. Communication Theory of Secrecy Systems, Bell Systems Technical Journal. No. 4.

[31] Borko Furht, Edin Muharemagic, Daniel Socek. 2005. Multimedia Encryption And Watermarking, Multimedia Systems and Applications Series. Springer Science+Business Media, Inc.Vol. 28.

[32] Arulmani Kuppusamy, Swaminathan PitchaiIyer and Kannan Krithivasan. 2014. Two-Key Dependent Permutation for Use in Symmetric Cryptographic System. Mathematical Problems in Engineering. 2014(Article ID 795292): 12. doi:10.1155/2014/795292.

[33] J. F. Andrade, M. L. Campos and J. A. Apolinario. 2008. Speech Privacy for Modern Mobile Communication Systems. IEEE International Conference on Acoustics, Speech and Signal Processing. Las Vegas, NV.

[34] B. Goldburg, S. Sridharan, and E. Dawson. 1993. Design and Cryptanalysis of Transform Based Analog Speech Scramblers. IEEE Journal of Selected Areas on Communications. 11: 735-743.

[35] C. J. Kuo. 1993. Novel Image Encryption Technique and Its Application in Progressive Transmission. J. Electronic Imaging. 2(4): 345-351.

[36] B Goldburg, E Dawson, S Sridharan. 1991.The Automated Cryptanalysis of Analog Speech Scramblers. Advances in Cryptology - EUROCRYPT '91, LNCS 547, pp. 422-430. Springer-Verlag.

[37] Qiao L., Nahrstedt K. 1998. Comparison of MPEG encryption algorithms. Int. J. Computers & Graphics. 22(4): 437-448.