Military Technical College
Kobry Elkobbah,
Cairo, Egypt

ICEENG 99

2nd International Conference
on Electrical Engineering
ICEENG 99

VOICE MAIL ENCRYPTION USING INTERNATIONAL DATA ENCRYPTION ALGORITHM

ASHRAF D. E. *, EMAD A. F. * and MOHAMED S. E. *

ABSTRACT

Nowadays, Electronic Voice Mail Services (EVMS) encryption is an area of a great interest due to the wide spreading of communication networks. If voice or some other analog signal must be conveyed with a high level of security then it should be digitized (A/D converted) and digitally encrypted. In the past, the digitization process was expensive so that analog scramblers found a place in the security business because of their lower cost and lower bandwidth requirements. The International Data Encryption Algorithm (IDEA) is one of the most secure block-ciphering algorithm. The key length of IDEA is 128 bits, which is hard to be broken with exhaustive search. There are no Substitution Code Boxes (S-boxes) in IDEA. Instead, there is a logical function that is especially designed in order to make the encryption algorithm itself the decryption algorithm, with another key. This means that the IDEA is asymmetric algorithm. This paper presents a study of using IDEA for speech encryption.

KEY WORDS

Encryption, Networks, Voice Mail and Speech Coding.

---

* Egyptian Armed Forces

I. INTRODUCTION

The human is always trying to keep his information, which is sent to another friend, to be secret over various transmission channels. These information can be in the form of writing, speech signal, etc., according to the communication channel. Of course, many algorithms for secret information are proposed and developed. Our study is concerned on the security of Electronic Voice Mail Services (EVMS) that are now emerging and get a strong reputation that it will be spreaded in the next few years. EVMS now are getting involved in both computer network and Cellular Telephone Networks (Mobile Cellular Telephone Networks). DES (Data Encryption Standard) and IDEA are both block cipher algorithms with block length equal to 64 bits. This is well suited for data encryption, but for voice encryption is less likely to be suitable. This is because data transmission is not sensitive to signal delay. For voice transmission, the situation is different, block delay of 64 bits means a delay of 8 voice samples, and in case of LPC10, this 8 voice samples take approximately one frame of 20 msec. Depending on the true distance between the source and the destination and the line quality, another delay will be added. A delay of 100 msec will be noticeable by the listener. This means that the block cipher algorithm will introduce 20% of the total allowable delay if there is one encryption-decryption stage in the communication link. If there is two encryption-decryption stages in the communication link, a total delay of 40% of the total permissible delay will be introduced at the encryption-decryption stations, and so on. So in real time, voice ciphering with LPC-DES combination may be not attractive. This situation becomes quiet different if we are concerned about EVM. In this later case the source records a voice messages, encrypts them and then stores them in the server for the destination (at that moment the destination is not available). As soon as the destination becomes available, he receives a note for his voice mail. If the destination is a personal station, then he decrypts his voice mail messages and listen to them. If the destination is a public station, then he can access his voice mail through the regular procedures (user-name, user ID number), then decrypts his voice mail messages using his own personal key. In the latter situation IDEA will be adequate, because it is asymmetric algorithm.

In this paper, an efficient algorithm is proposed, which can be applied simply in hardware leads to a very secure Electronic Voice Mail Services (EVMS). In the following, section II briefly reviews the speech processing technique, and section III describe the encryption and decryption algorithm using IDEA. Simulation approaches and results are presented in section IV.

II. SPEECH PROCESSING TECHNIQUE

The main speech – coding techniques are Waveform Coders in both time and freq. domains, Parametric Coders and Hybrid Coders but all these algorithms still does not give us a high level of security [1][2]. To get a high level of security, speech signal must be digitized and digitally encrypted. In our technique all speech – coding techniques can be used for the digitization process.

In this paper we use Residual Excited Linear Prediction (RELP)[3][4], which is a hybrid coder because it gives us the ability of having variable compression ratios,

so that we can control the output bit rate according to the quality of the channel as well as a good performance of RELP.

In the RELP coder the residual is passed through a lowpass filter with a cut-off of about 1000 Hz, decimate its output and encode the decimated signal. Usually, the decimated signal is transformed into the frequency domain via the discreet Fourier transform and the magnitude and phase of the frequency components are coded and transmitted to the receiver. At the receiver the residual is reconstructed by copying the baseband residual to the other frequencies to be considered.

The resulting signal contains no high-frequency information. This means that when the residual is reconstructed at the receiver the perceptually important pitch information will be incorrect for higher frequencies. In this lies one of the major disadvantages of the RELP coder. The problem is sometimes alleviated by adapting the cut-off of the lowpass filter to the pitch frequency. A simple method for regenerating the high-frequency components is to pass the signal through a full-wave rectifier and then flatten the resulting spectrum by filtering. RELP coders are usually used to give good quality speech at bit rates in the region of 9.6 kb/s.

III. THE ENCYPTION ALGORITHM USING IDEA

IDEA was designed to be efficient to compute in software[5]. It encrypts a 64-bit block of plain text into a 64-bit block of ciphertext using a 128-bit key. It's relatively new (1991), but sufficiently published that cryptanalysts have had some amount of time to find weakness. It has 17 rounds, where the odd-numbered rounds are different from the even-numbered rounds[6][7]. Each round takes the input, a 64 bit quantity, and treats it as four 16-bit quantities, which we will call X_a, X_b, X_c, and X_d. Mathematical functions are performed on X_a, X_b, X_c, and X_d to yield new versions of X_a, X_b, X_c, and X_d. The 128-bit key is expanded into 52 16-bit keys, K_1, K_2,.....K_{52}.The 52-encryption keys are generated by writing out the 128-bit key and, starting from the left, chopping off 16 bits at a time. This generates 16-bit keys.

The odd rounds use four of the K_i, which we'll call K_a, K_b, K_c, and K_d. The even rounds use two K_i, which we'll call K_e and K_f. So round one uses K_1, K_2, K_3, and K_4. Round 2 uses K_5, and K_6. Round 3 uses K_7, K_8, K_9, and K_{10}. Round 4 uses K_{11}, and K_{12}. And so fourth.

An odd round, therefore has as input X_a, X_b, X_c, and X_d and keys K_a, K_b, K_c, and K_d. An even round has as input X_a, X_b, X_c, and X_d and keys K_e, and K_f.

Odd Round: The odd round is simple. X_a is replaced by $X_a \otimes K_a$. X_d is replaced by $X_d \otimes K_d$. X_c is replaced by $X_b + K_b$. X_b is replaced by $X_c + K_c$. (Fig. 1)

Note that this is easily reversible. To get from the new X_a to the old X_a, we perform $\otimes$ with the multiplicative inverse of K_a, mod $2^{16}+1$. Likewise with X_d. To get

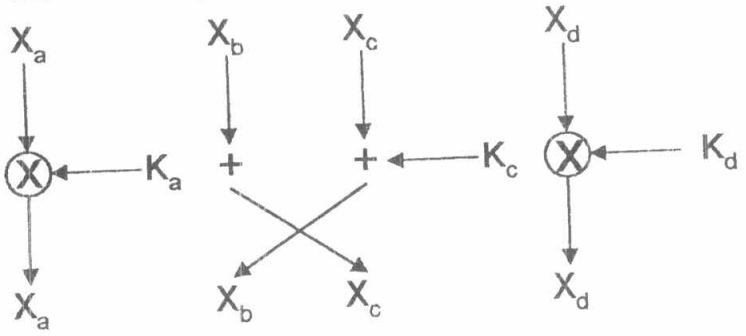

Fig. 1 IDEA Odd Round

the old X_b, given the new X_c, we add the additive inverse of K_b, i.e., we subtract K_b. So when decrypting, the odd rounds run as before, but with the mathematical inverses of the keys. This will undo the work that was done during that round in encryption.

Even Round: The even round is a little more complicated. Again, we have X_a, X_b, X_c, and X_d. We have two keys K_e and K_f. We are going first compute two values, which we will call Y_{in} and Z_{in}. We'll do a function, which we'll call the mangler function, which takes as input Y_{in}, Z_{in}, K_e and K_f and produces what we'll call Y_{out} and Z_{out}. We'll use Y_{out} and Z_{out} to modify X_a, X_b, X_c, and X_d.

$$Y_{in}=X_a\oplus X_b \qquad Z_{in}=X_c\oplus X_d \tag{1}$$

$$Y_{out}=((K_e\otimes Y_{in})+Z_{in})\otimes K_f \qquad Z_{out}=(K_e\otimes Y_{in})+Y_{out} \tag{2}$$

Now we compute the new X_a, X_b, X_c, and X_d.

$$\text{new } X_a=X_a\oplus Y_{out} \qquad \text{new } X_b=X_b\oplus Y_{out} \tag{3}$$

$$\text{new } X_c=X_c\oplus Y_{out} \qquad \text{new } X_d=X_d\oplus Y_{out} \tag{4}$$

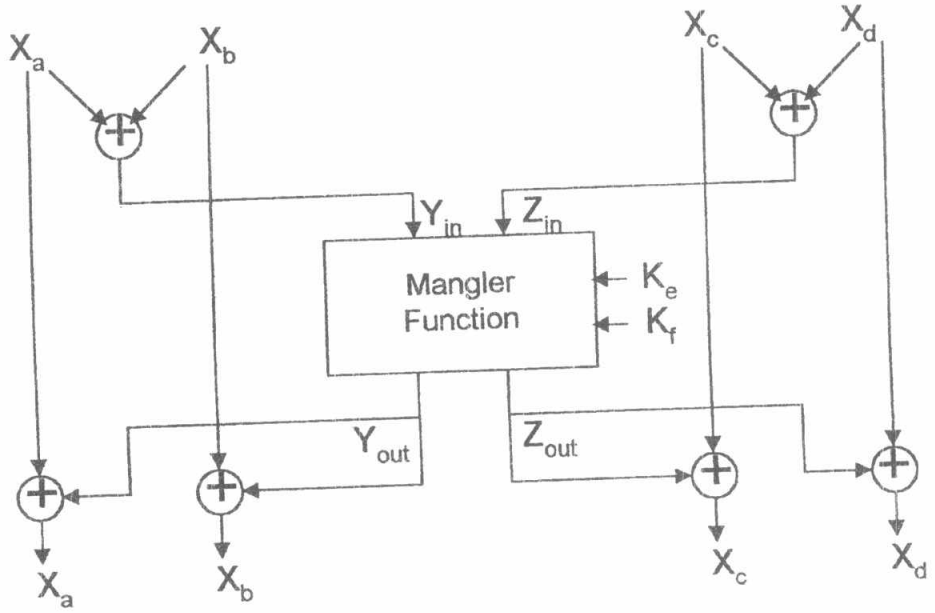

Fig. 2 IDEA Even Round

***The Decryption*:**

IDEA is cleverly designed so that the same code (or hardware) can perform either encryption or decryption given different expanded keys. We want to compute inverse keys such that the encryption procedure, unmodified, will work as a decryption procedure. The basic idea is to take the inverses of the encryption keys and use them in the opposite order(use the inverse of the last used encryption key as the first key used when doing decryption).

Remember that for encryption, we generated 52 keys, K_1 through K_{52}. We use four of them in each of the odd rounds, and two of them in each of the even rounds. And since we are working backwards, the first decryption keys should be inverses of the last-used encryption keys. Given that the final keys used are K_{49}, K_{50}, K_{51}, and K_{52}, in an odd round, the first four-decryption keys will be inverses of the keys K_{49}-K_{52}. K_{49} is used in $\otimes$, so the decryption key K_1 will be the multiplicative inverse of K_{49} mod $2^{16}+1$. And the decryption key K_4 is the multiplicative inverse of K_{52}. Decryption keys K_2 and K_3 are the additive inverses of K_{50} and K_{51} (meaning negative K_{50} and K_{51}).

In the even rounds, the keys do not have to be inverted. The same keys are used for encryption as decryption.

IV. SOFTWARE SIMULATION APPROCHES AND RESULTS

In this section, we will introduce the simulation performed on the algorithm proposed in section III. This simulation is performed by using a sound board "16 bit-sound blaster" as an interface unit.

The input speech waveform is sampled at a sampling frequency f_s=8 KHz.

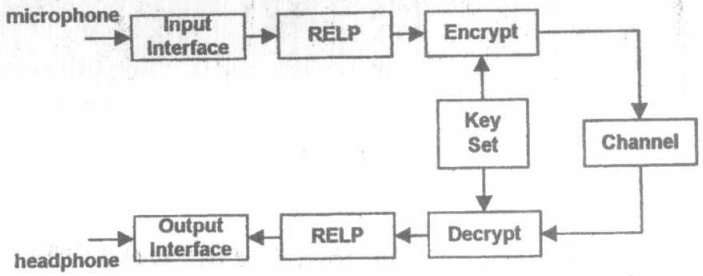

Fig.3 The block diagram of the simulation program

Fig. 3 illustrates the block diagram of the simulation software, in which blocks "Encrypt and Decrypt" are main programs performing the encryption and decryption process. The block "Key Set" generates the encryption and decryption keys as described in section III.

The complete process is applied on the word "Allh". Fig. 4 illustrates the original speech waveform and the encrypted waveform. As shown in Fig. 4 the encrypted waveform is very similar to a white noise, that means that a high level of security is achieved.

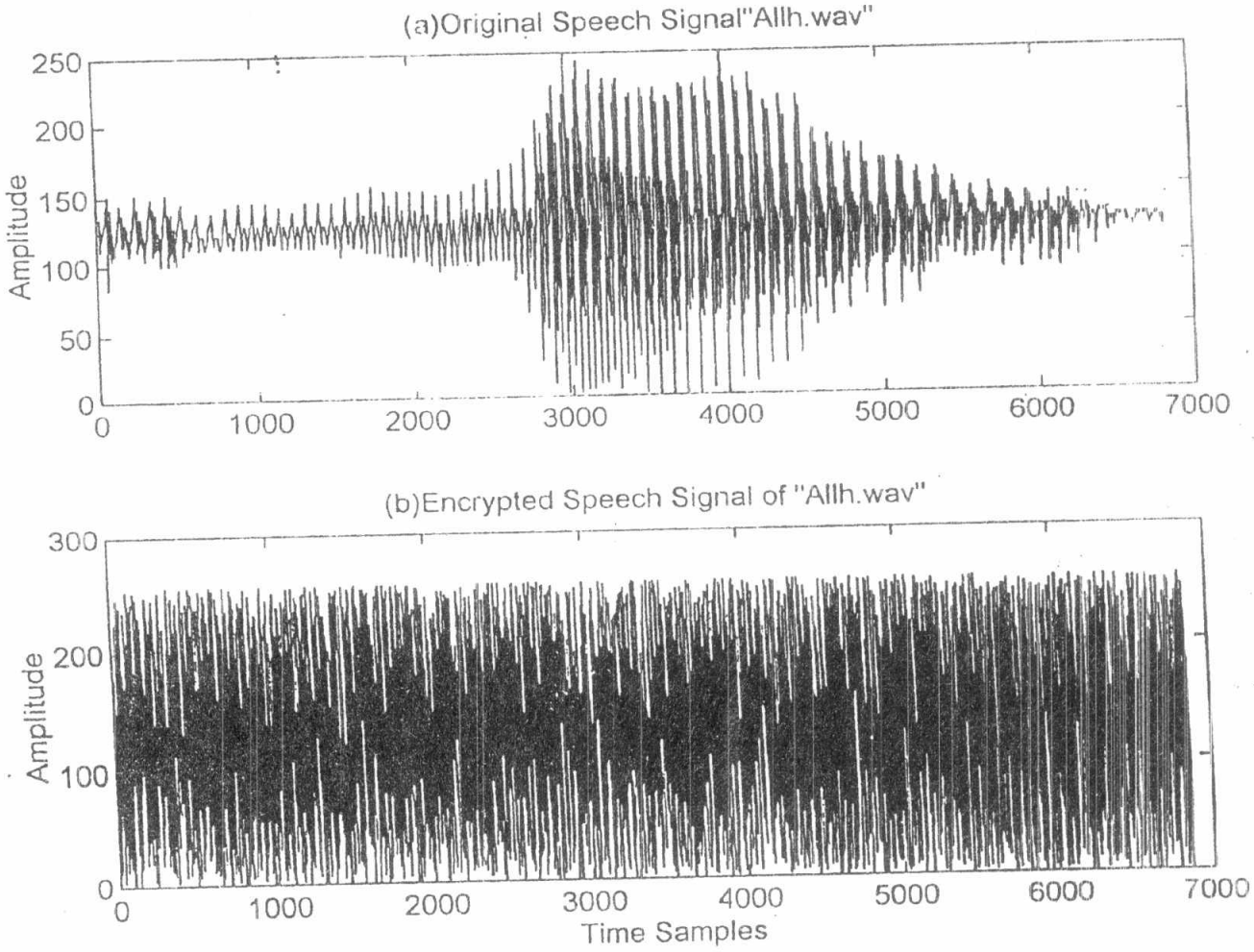

Fig. 4 Amplitude waveforms of the word "Allh"

(a) the original speech; (b) the encrypted speech

V. CONCLUSION

The encrypted voice messages using IDEA with any speech coding formats (PCM, ADPCM, LPC, and RELP) are quite random in nature. They are heard as a great noise source. The voice is completely sunk inside the noise. There is no difference in the heard sounds with different speech coding formats or different voice messages. The spectrum of the encrypted messages is flat, this is very efficient than all analog scramblers used for speech encryption. The key length of IDEA is 128 bits, which is hard to be broken with exhaustive search.

REFRENCES

[1] SHUZO SAITO & KAZUO NAKATA, "*Fundamental of speech processing*", Academic Press, Inc., 1985.

[2] JAMES L.FLANGAN & MANFRED R.SCHROEDER, "*Speech coding*", an invited paper, IEEE transactions on communications, VOL. COM-27, NO.4, April 1979.

[3] J.N. HOLMES, "*Speech Synthesis And Recognition*", Van Nostrand Reinhold (UK) Co. Ltd., 1988.

[4] CHONG KWAN UN, and, D. THOMAS MAGILI, "*The Residual-Excited Linear Prediction Vocoder with transmission rate below 9.6 kbit/sec.*", IEEE transactions on communications, VOL. COM-23, NO.12, December 1975.

[5] KENNETH H. ROSEN, "*Cryptography: Theory and Practice*", CRC Press, Inc., 1995.

[6] K.PRESTTUN, " *Security Measures in Communication Networks*", Electrical Communication, Volume 60, No.1, 1986.

[7] RITA C.SUMMERS, "*Secure Computing*", McGraw-Hill Companies, 1997.