Secure voice cryptography based on Diffie-Hellman algorithm

To cite this article: Sura F. Yousif 2021 IOP Conf. Ser.: Mater. Sci. Eng. 1076 012057

View the article online for updates and enhancements.

You may also like
- A Review on Cryptography Protocol for Securing Data
  Robbi Rahim
- Optical voice information hiding using enhanced iterative algorithm and computational ghost imaging
  Ravi Kumar, Fuqiang Zhong and Chenggen Quan
- Diffie-hellman Protocol on Raspberry pi
  Yuwen Wang and Gabriela Mogos
Secure voice cryptography based on Diffie-Hellman algorithm

Sura F. Yousif
Department of Chemical Engineering, Collage of Engineering, University of Diyala, Diyala, Iraq
Email: sura.fahmy@gmail.com

Abstract. This article introduces a new technique for voice signals encryption & decryption to ameliorate the information security during transferring over unsecure network. The presented mechanism is based on a particular type of asymmetric key cryptography called Diffie-Hellman Algorithm. The main reason of using this algorithm is enabling users to encrypt and decrypt their messages via a shared session key that is known only for them. Firstly, the input voice signal in this scheme is enciphered by utilizing the shared secret key at the dispatcher. Secondly, the scrambled voice signal is delivered to the destination through open telecommunication network. Eventually, the cipher text signal is deciphered at the receptor by exploiting the same key to retrieve the plaintext signal. The performance of the presented voice cryptosystem is assessed via different known ciphering/deciphering voice measures for different voice signals. The computed and visual results in addition to comparison outcomes confirm that the presented cryptosystem can fulfill good enciphering and deciphering results and it is capable of tolerating various cryptographic analyses.

1. Introduction
The purpose behind the necessity to possess security plans is technology, which has been growing in communication networks over the years to protect the transmitted or stored information throughout these networks particularly in business situations, because they provide a timely and fast exchange of information. However, insecurity is the problem of the networks because the messages can be vulnerable to interception or alteration if those networks were accessed without authorization. Due to this reason, it is necessary to encrypt the message during communication between two persons or more so that the exchange of information will be secure. Encryption converts the original data or plaintext from its readable form to unreadable one that is hard to understand which called cipher text. Decryption is the reverse process that recovers the original data again from the ciphered or encrypted data. These two processes are controlled by a shared key between the transmitter and receiver such that only the intended or authorized person can decrypt the message and reads its content. Unauthorized person who does not possess the required key to read the encrypted message can be excluded. Cryptography involves study of encryption/decryption mechanisms in the presence of adversaries or third party depends on definite mathematical tool called algorithm. A number of security advantages is provided via cryptography such as privacy or confidently, authentication, access control, integrity and non-repudiation [1, 2]. Voice data is different from other data like digital images and text messages due to the high redundancy, strong correlation and bulk capacity between voice samples. Encryption of voice information is performed by filling specific parts of the message which represent the silent parts of the voice signal with noise signal before transmission, extraction this noise.
will yield the original voice signal at the receiver [3, 4]. Voice communication is almost utilized in every activity of daily life like military applications, education, e-learning, commerce, phone banking, politics, news broadcasting and so forth. Therefore, the protecting of voice data while transmitting over any insecure communication channel by using cryptography techniques became an important issue [5, 6].

Broadly, cryptography systems can be categorized into major types namely symmetric and asymmetric key systems. In symmetric key system which is also called secret key system, a common key is utilized for encryption as well as decryption processes of the message. Both sender and recipient share this key privately. The sender employs the mutual key to encrypt the data before sending it to the recipient, then the recipient will employ the same key to decrypt the data in order to restore the original message. Some examples of symmetric key systems are: AES, DES, IDEA, Triple-DES (3DES), TWOFISH and BLOWFISH [1]. In asymmetric key system which is also called public key system, the secret key can be split to two different portions which are mathematically related namely public and private. The public key is applied for data encryption and can be shared with any person, while the private key is applied for data decryption and it should be preserved secret. The encryption in asymmetric key system is different as compared with symmetric key system. Anyone who possesses the public key can employ it for encrypting the messages, but the decryption process is achieved only by the person who has the private key. Some examples of asymmetric key systems are: Diffie-Hellman, RSA, El-Gamal and Digital Signature [7, 8].

Different voice encryption methods have been proposed in the recent years based on various techniques. In [5], chaotic system is combined with optical ciphering method to encrypt audio signals. Chaotic map is utilized to provide the first level of security, while optical encryption technique is employed to provide the second level of security. In [9], the input voice signal is encrypted via chaotic Cat map and Zaslavsky map to ensure security over unsecure open channels. Different analyses are performed on this system to prove its efficiency against known attacks. In [10], the voice signal samples are encrypted by applying two stages. Arnold chaotic map is employed in the first stage for permutation operation and Henon map is utilized in the second stage for substitution operation. In [11], a hybridization scheme for speech cryptography is introduced. Cryptography process is started by applying RSA algorithm into the speech message followed by applying DES algorithm. The final step is accomplished by applying a combination of both RSA and DES algorithms. For optimization purpose, genetic algorithm is applied. The system is assessed by the means of MSE and PSNR. In [12], the original voice file is divided to four parts. Then, each part is scrambled via four different chaotic maps; this process is controlled by shift keying technique. Chen system is exploited in the eventual phase for security improvement. In [13], hybrid transform of chaotic shift mechanism, chaos maps and DNA algorithm are implemented to encipher different audio signals. The algorithm can withstand several attacks. In [14], the message is ciphered via an improvement version of El-Gamal mechanism. This modified system has immunity to chosen plaintext/ciphertext attacks. In [15], a hybrid of RSA and DNA techniques is exploited to encrypt the secret information. The input message is initially encrypted by implementing DNA technology, then the output cipher message from this step is used as an input for RSA cryptosystem. In [16], symmetric AES algorithm is used for encoding/decoding the audio file. After the ciphering operation, the encrypted audio signal is transferred to the receiver through an open channel. The original signal is then restored by implementing the decryption operation on the transmitted signal. In [17], the author utilized rotation formulas and Circle map for audio signal encryption. Many simulation tests are performed to prove the cryptosystem efficiency like histogram and correlation analyses. In [18], a speech encryption technique is introduced which relies upon shuffling and substitution principles. Speech samples permutation is obtained via chaotic Henon map, whereas Lorenz system is applied for samples substitution. In [3], secret information is protected by employing an improved form of Diffie–Hellman technology. The security of this approach is assessed against several common attacks. In [19], a quantum chaos map is applied to design voice encoding algorithm. The complexity of the method is investigated in both classical and quantum domains. In [20], high confidently is provided by executing
biometric properties and DWT for encrypting speech files. Objective and subjective metrics are performed on the output signal to demonstrate the system validity. In [21], the researchers suggested a ciphering speech mechanism that depends on chaos systems and FFT technique. The developed Logistic/Lorenz system in this approach is used for speech samples scrambling and diffusion processes. In [22], Standard, Logistic and Baker systems are employed to encrypt the transferred audio file through OFDM channels. Several quality parameters are carried out to evaluate the cryptosystem performance. However, each voice encryption approach possesses strength as well as weakness points. In order to conquer the traditional challenges in speech signal encoding such as inadequate security, existing noise in the reconstructed signal and excessive bandwidth usage, a new ciphering method for protecting voice signals by applying Diffie-Hellman cryptosystem is presented in this work. This mechanism can supply high degree of security to endure known attacks. The major contributions for the introduced cryptosystem comprise: (1) Solving the problem of exchanging the secret key throughout insecure communication networks. (2) Exploiting Diffie–Hellman algorithm to construct a powerful voice ciphering/deciphering mechanism. (3) Improving the resistance of voice cryptosystem against various cryptographic attacks by executing Diffie–Hellman technique to encrypt voice samples. (4) The difficulty of breaking this cryptosystem by attacker and extracts the original voice content because its security depends on discrete logarithms problem which is considered one of the most complicated problems in mathematics. (5) Reducing the computational complexity. (6) Easy of implementation. (7) Making the communication process easy between any number of entrants who wish to encipher and decipher their confidential messages because Diffie–Hellman protocol is not restricted to only two communicated participants. (8) Modifying process of voice signal samples before starting the encryption procedure enhances the scrambling level to get output signal with lower residual intelligibility, and re-modifying process of those samples after finishing the decryption procedure improves the quality of the reconstructed signal. Initially, the samples of voice signal in the proposed mechanism are modified or adjusted to obtain a new signal. In the second stage, the transmitter and receiver generate their public and private keys according to Diffie–Hellman protocol. In the third stage, the public keys of the two users are then exchanged over open channel while maintaining their private keys secret. In the fourth stage, the session key is calculated at the two ends to accomplish the ciphering and deciphering processes. In the final stage, the voice samples are re-modified to get the original input signal. The proposed scheme performance is examined by utilizing various common encoding/decoding quality criteria comprising Signal to Noise Ratio, Peak Signal to Noise Ratio, Segmental Signal to Noise Ratio, Frequency Weighted Segmental Signal to Noise Ratio, Log Likelihood Ratio, Bit Error Rate, Histogram, Spectrogram, Correlation, Differential and time analyses. The numerical and visual simulation outcomes in addition to comparison with other existing systems, confirm the security, robustness and feasibility of the described method. The sections of the work are sorted as shown: The details of the Diffie–Hellman algorithm are presented in Section 2; Section 3 discusses the frameworks of coding and decoding phases for the proposed voice cryptosystem. The performance parameters which are employed to assess the presented technique are explained in Section 4, while the numerical and visual outcomes are reported in Section 5. The comparison with current methods is given in Section 6 and eventually, Section 7 deduces the proposed voice encryption approach and future work.

2. Diffie-Hellman algorithm

Diffie-Hellman Algorithm is broadly known as DH algorithm. Where DH symbolizes to the last names of its discoverers Whitfield Diffie and Martin Hellman. It was the first asymmetric key system that published in 1976. DH is widely utilized as a key exchange method that enables two communicating parties to construct a mutual key subsequently which can be applied for messages encryption/decryption through unprotected medium. The security of Diffie-Hellman system is based on the problem of discrete logarithms called Diffie-Hellman problem which considered hard to solve. Generally, when a convenient mathematical group is applied, DH protocol is considered to be secure. The following steps explain the Diffie-Hellman algorithm:

---

2. Diffie-Hellman algorithm

Diffie-Hellman Algorithm is broadly known as DH algorithm. Where DH symbolizes to the last names of its discoverers Whitfield Diffie and Martin Hellman. It was the first asymmetric key system that published in 1976. DH is widely utilized as a key exchange method that enables two communicating parties to construct a mutual key subsequently which can be applied for messages encryption/decryption through unprotected medium. The security of Diffie-Hellman system is based on the problem of discrete logarithms called Diffie-Hellman problem which considered hard to solve. Generally, when a convenient mathematical group is applied, DH protocol is considered to be secure. The following steps explain the Diffie-Hellman algorithm:
Firstly, two constants $p$ and $g$ are chosen by both communicating parties Alice and Bob, where $p$ is a prime number and $g$ is the generator or base that less than $p$.

Alice selects a random integer number $a$ and computes her public key $(g^a \mod p)$, where $a$ represents Alice’s private key.

Similarly, Bob selects independently a random integer number $b$ and computes his public key $(g^b \mod p)$, where $b$ represents Bob’s private key.

The public keys of Alice and Bob are then exchanged through insecure communication medium such as Internet while keeping their private keys secret.

Alice calculates $(g^b \mod p)^a \mod p$ which is equal to $g^{ba} \mod p$.

Bob calculates $(g^a \mod p)^b \mod p$ which is equal to $g^{ab} \mod p$.

$K$ is calculated as $K = g^{ba} \mod p = g^{ab} \mod p$, where $K$ represents the shared secret key that can be used to encrypt and decrypt messages.

The values of $p$, $g$, $g^a \mod p$ and $g^b \mod p$ can be published in the clear. Contrariwise, $a$, $b$ and $K$ values are preserved confidential. Computationally, determining the value of $K$ is infeasible by only knowing the public keys or by observing the conversation between the communicating persons [2, 3, 23]. Figure 1 explains the above steps for Diffie-Hellman algorithm.

**Figure 1.** Block diagram of the Diffie-Hellman algorithm.

2.1. Encryption process
If Alice wants to send a secret message $m$ to Bob through unsecure channel, she will employ the shared secret key $K$ using the formula $c = m \star g^{ab} \mod p = m \star K$. Where $c$ is the encrypted message.

2.2. Decryption process
Bob can then reconstruct the original message $m$ from $c$ by employing the same secret key $K$ using the formula $m = \frac{c}{g^{ab} \mod p} = \frac{c}{K}$. Where $m$ is the decrypted or original reconstructed message [24, 25].

3. The Proposed methodology
In this article, a new method for voice encryption and decryption that based on Diffie-Hellman scheme is presented. Firstly, the voice signals are modified, then the pair of keys: public and private for each user are produced as discussed previously to generate the shared secret key ($K$) prior to the encryption
and decryption processes. Secondly, the obtained voice samples are encrypted in the sender side by employing the generated secret key. Thirdly, the encrypted voice samples are transferred through a public channel to the recipient side and when they received, they will be decrypted by employing the same secret key to restore the original voice samples. Figure 2 clarifies the block diagram of the proposed methodology.

![Block diagram of the proposed methodology.](image)

**Figure 2.** Block diagram of the proposed methodology.

### 3.1. Details of voice coding and decoding operations

**Step 1:** The sample values of the original one-dimension voice signal \( m(i) \) are modified to integer positive numbers between \((0, 255)\) to acquire the adjusted voice signal \( B(i) \) as described in equation (1).

\[
B(i) = \text{mod} \left( \text{floor} \left( m(i) \times 10^{14} \right), 256 \right)
\]  

where \( \text{mod} \) refers to the modulo operation and \( \text{floor} \) points to floor function which rounds numbers toward negative infinity.

**Step 2:** Two different integer numbers \( p \) and \( g \) are picked out by the transmitter and receiver.

**Step 3:** Another random numbers \( a \) and \( b \) which represent the sender and receptor confidential keys are picked out.

**Step 4:** The public key of the sender is calculated as defined in equation (2).

\[
S_1 = g^a \mod p
\]

**Step 5:** The public key of the receiver is calculated as defined in equation (3).

\[
S_2 = g^b \mod p
\]

**Step 6:** \( S_1 \) and \( S_2 \) are then swapped over unsecure medium.

**Step 7:** The session secret key \( K \) at the transmitter is computed as in equation (4) to accomplish the encryption operation.

\[
K = (S_2)^a \mod p
\]

**Step 8:** The encoded voice signal \( C(i) \) is gained by employing \( K \) on \( B(i) \) as exhibited in equation (5).

\[
C(i) = B(i) \times K
\]
Step 9: $C(i)$ is then transferred to the receiver throughout public channel.

Step 10: The session secret key $K$ at the receptor is calculated as in equation (6) to achieve the decryption operation.

$$K = (S_1)^b \mod p$$ (6)

Step 11: The decoded voice signal $D(i)$ is produced by implementing $K$ on $C(i)$ as clarified in equation (7).

$$D(i) = \frac{c(i)}{K}$$ (7)

Step 12: The plain voice signal $m(i)$ is retrieved by restoring the original voice samples as calculated in equation (8).

$$m(i) = 256 \times \text{floor} \left( \frac{D(i)}{10^{14}} \right)$$ (8)

4. The Performance measures

To check the performance of the designed methodology, a number of quantitative measures are considered in this work in terms of both voice encryption and voice decryption processes. These measures are broadly used in the field of voice cryptosystems which are Signal to Noise Ratio (SNR), Peak Signal to Noise Ratio (PSNR), Segmental Signal to Noise Ratio (SNRseg), Frequency Weighted Segmental Signal to Noise Ratio (fwSNRseg), Log Likelihood Ratio (LLR) and Bit Error Rate (BER). The illustration of these measures is given as follows.

4.1. Signal to Noise Ratio (SNR)

The attacker permanently militates to minimize this measure. The key parameter to measure the data content in the encrypted information is noise performance. The encrypted information looks like white noise. Hence, the level of noise in the encrypted information is larger as compared with that in the original voice signal. SNR can be computed as in equation (9):

$$SNR \text{ (dB)} = 10 \times \log_{10} \frac{\sum_{i=1}^{N_s} x_i^2}{\sum_{i=1}^{N_s} (x_i - y_i)^2}$$ (9)

where $x_i$ and $y_i$ represent the samples of the original and encrypted voice signals, respectively [9, 21].

4.2. Peak Signal to Noise Ratio (PSNR)

PSNR index is defined as the ratio between the maximum power obtained in the plaintext voice signal and the power obtained in the ciphertext voice signal. This metric is calculated as in equation (10):

$$PSNR \text{ (dB)} = 10 \times \log_{10} \frac{n x^2}{\|x-k\|^2}$$ (10)

where $n$ symbolizes to the input signal length, $x$ symbolizes to the value of maximum absolute square of the input signal and the term $(x - k)$ points to the difference energy between input and output signals [12].

4.3. Segmental Signal to Noise Ratio (SNRseg)

This measure is used to estimate the quality of voice signal. SNRseg represents the rate of SNR for short segments of the voice signal. It can be calculated as in equation (11):

$$SNRseg \text{ (dB)} = 10 \sum_{m=0}^{M} \sum_{i=k_m}^{k_{m+1}-1} \log_{10} \frac{\sum_{i} x_i^2}{(x_i - y_i)^2}$$ (11)

where $K$ represents the length of each segment while $M$ represents the total number of segments in the voice signal [26, 27].
4.4. Frequency Weighted Segmental Signal to Noise Ratio (fwSNRseg)
FwSNRseg is another version of SNR that can be utilized to determine the quality of voice data at encryption and decryption processes. It can be described as in equation (12):

\[
f_{\text{wSNR}_{\text{seg}}} (\text{dB}) = \frac{10}{M} \sum_{m=0}^{M-1} \sum_{j=0}^{K-1} W(j,m) \log_{10} \left( \frac{X(j,m)^2}{\hat{X}(j,m)^2} \right) \sum_{j=0}^{K-1} W(j,m)
\]

(12)

where \( M \) and \( K \) refer to the number of frames and number of bands, respectively, \( W(j,m) \) refers to the weight at specified frequency band, \( X(j,m) \) and \( \hat{X}(j,m) \) point to the spectrum of input and output voice signals, respectively [28].

4.5. Log Likelihood Ratio (LLR)
LLR for an original and encrypted voice signals that have linear predictive coding (LPC) vector \( \vec{a}_o \) and \( \vec{a}_e \), respectively can be defined as in equation (13):

\[
\text{LLR} = \log \left( \frac{\vec{a}_e^T R_o \vec{a}_o^T}{\vec{a}_o^T R_o \vec{a}_e^T} \right)
\]

(13)

where \( \vec{a}^T \) represents the transpose of vector \( \vec{a} \), and \( R_o \) represents the autocorrelation matrix of the original voice signal [27-29].

4.6. Bit Error Rate (BER)
BER represents the ratio between the number of erroneous bits and the overall number of transmitted bits throughout a period of time. BER is given as in equation (14):

\[
\text{BER} = 0.5 \times \text{erfc} \left( \frac{E_b}{\sqrt{N_0}} \right)
\]

(14)

where \( E_b \) and \( N_0 \) indicate to the energy of average bit and spectral density of noise, respectively [30].

5. Experimental results
Several experimental analyses are carried out in this section for evaluating and quantifying the encryption and decryption efficiencies of the presented voice cryptosystem. The specification of hardware that used to design all the simulation results in this work is represented by HP laptop Intel Core i3 processor, 2.40 GHz of CPU and 3.90 GB of RAM while the software utilized is MATLAB language (R2013a), windows 7 ultimate as an operating system. Four different voice files in wave format are used in all experiments in this cryptosystem which represent spoken phrases for different persons (male and female) in English language. The sampling rate for these voice files is 16 KHz and they have different times from 1 to 4 seconds. Furthermore, all the silence intervals are totally eliminated from those files during experiments conducted. Different values for public and private keys \( p, g, a \) and \( b \) were examined. In this algorithm for example, the values of these parameters are set as 23, 5, 6 and 15, respectively because they achieve better outcomes for voice encryption. Thus, by following the previously explained steps, the value of the shared secret key \( K \) will be 2. The waveform of the original or unsecure voice signal is shown in Figure 3a, while the waveform of encrypted voice signal that yields from applying the proposed methodology is noticed in Figure 3b. The waveform of the decrypted or restored voice signal is shown in Figure 3c, which yields from applying the procedures for decryption process of the presented algorithm. By comparing Figures 3a and 3b, it is obvious that the waveform of the encrypted voice signal is entirely distinct from the original one. The distribution of encrypted samples is totally uniform, unintelligible and noise-like which proves the high encryption quality. Furthermore, it is clear that the waveforms of the original and decrypted voice signals are identical by comparing Figures 3a and 3c, which confirms the high decryption quality.
5.1. Voice signal quality at encryption

Encryption quality measures have a magnificent importance in the designing of the encryption algorithms for voice cryptosystems. They are desired for two reasons: the first reason is to determine the parameters settings, optimize the structures of voice cryptosystems and indicate the amount of distortion presented by the voice cryptosystems. The voice cryptosystem performance is better as the amount of distortion increases. The second reason is to determine the encryption algorithm immunity to eavesdropper attacks [5]. The encryption quality measures used in this voice cryptosystem are discussed previously which are: Signal to Noise Ratio (SNR), Peak Signal to Noise Ratio (PSNR), Segmental Signal to Noise Ratio (SNRseg), Frequency Weighted Segmental Signal to Noise Ratio (fwSNRseg), Log Likelihood Ratio (LLR) and Bit Error Rate (BER). Higher encryption quality is obtained when the values of SNR, PSNR, SNRseg, and fwSNRseg decrease, whilst the value of LLR and BER increase. Table 1 clarifies the results of these measures for the presented scheme.

Table 1. The residual intelligibility measures for voice signals at encryption.

| File name | SNR (dB) | PSNR (dB) | SNRseg (dB) | fwSNRseg (dB) | LLR     | BER     |
|-----------|----------|-----------|-------------|---------------|---------|---------|
| Signal 1  | -39.5803 | 23.7910   | -27.7054    | -45.2709      | 1.0191  | 0.9870  |
| Signal 2  | -42.2740 | 23.7676   | -27.5751    | -52.3809      | 1.1873  | 0.9910  |
| Signal 3  | -39.2629 | 23.9246   | -20.1135    | -50.2149      | 1.1720  | 0.9923  |
| Signal 4  | -41.6258 | 23.0619   | -33.9629    | -42.3890      | 1.0168  | 0.9930  |

Figure 3. The waveforms of (a) Original, (b) Encrypted, (c) Decrypted voice signals.
It can be observed from Table 1, that the values of SNR, PSNR, SNRseg and fwSNRseg measures are very low, whereas the values of LLR and BER measures are high for all encrypted voice signals, which means that the original and encrypted voice signals are completely different. This confirms the superiority of the cryptographic process performed by the introduced system to encrypt voice data.

5.2. Voice signal quality at decryption
Decryption quality measures have the same importance as encryption quality measures in the designing and conservation of voice cryptosystems. They are desired for two reasons: the first is to determine the voice cryptosystem immunity to attacks and distortion while the second is to exhibit the effectiveness of the voice cryptosystem by measuring the decrypted voice signal quality as compared with the original one [5]. The decryption quality measures used in this voice cryptosystem are the same six encryption quality measures which are: Signal to Noise Ratio (SNR), Peak Signal to Noise Ratio (PSNR), Segmental Signal to Noise Ratio (SNRseg), Frequency Weighted Segmental Signal to Noise Ratio (fwSNRseg), Log Likelihood Ratio (LLR) and Bit Error Rate (BER). Higher decryption quality is obtained when the values of SNR, PSNR, SNRseg, and fwSNRseg increase, while the values of LLR and BER decrease. Table 2 illustrates the results of these measures for the presented scheme. It is clear from Table 2, that the values of SNR, PSNR, SNRseg, and fwSNRseg measures are very high, while the values of LLR and BER measures are very low for all decrypted voice signals, which means that the original and decrypted voice signals are the same. This implies the efficiency of the presented system to decrypt voice date with good precision and high quality.

5.3. Histogram analysis
A known tool to measure the sample values distribution in the voice files is histogram. Generally, for ideal voice cryptosystem, the sample values should have equal probability such that the attacker cannot extract any helpful information from the scrambled signal through the statistical attack [10, 12, 17, 18]. Histogram plots for input, output and retrieved voice files obtained via the introduced work are illustrated in Figure 4. Obviously, the samples distribution for encrypted signal in Figure 4b is reasonably uniform and it is quite different from the histogram of input signal in Figure 4a, which indicates strong ciphering effect. Moreover, the histogram of deciphered file in Figure 4c is almost identical to the original signal histogram in Figure 4a, which manifests perfect reconstruction performance. Figure 4 reveals that the presented method can counter statistical analysis successfully.

| File name | SNR (dB) | PSNR (dB) | SNRseg (dB) | fwSNRseg (dB) | LLR | BER |
|-----------|----------|-----------|-------------|---------------|-----|-----|
| Signal 1  | 246.6736 | 303.9110  | 257.6707    | 60.7456       | 5.4581 × 10⁻¹⁴ | 0.0868 |
| Signal 2  | 243.9763 | 303.8766  | 247.7204    | 61.5437       | −1.6321 × 10⁻¹³ | 0.0709 |
| Signal 3  | 246.9989 | 304.0410  | 249.9292    | 61.2856       | −9.1611 × 10⁻¹⁵ | 0.0921 |
| Signal 4  | 244.6288 | 304.1745  | 248.2693    | 62.1367       | −2.5073 × 10⁻¹⁵ | 0.0628 |

5.4. Spectrogram analysis
Another significant parameter to analyze voice signals is spectrogram. The voice signal spectrogram is a visual representation of the spectrum in frequency domain [6, 8, 21, 26]. The spectrogram graphs of the input encoded, and decoded voice signals generated via the suggested approach are represented in Figure 5. By visual comparison between Figures 5a and 5b, it is obvious that the plain and cipher voice signals are entirely unlike and the beneficial information from the input file is totally removed.
This manifests perfect ciphering performance. In addition, by visual comparison between Figures 5a and 5c, it is clear that the plain and decipher voice signals are identical, which implies ideal deciphering performance.

5.5. Correlation analysis
To indicate the quality for any cryptosystem, a common statistical measure can be used which known as correlation coefficient or \( r_{xy} \). The correlation between two variables or more in the original and encrypted or decrypted voice signals can be performed by using this measure. If the relation between the original and encrypted voice signals is weak, then the value of correlation coefficient is zero or close to zero, which points to the superiority of encoding process. Contrary, if the relationship between the plain and decipher signals is strong, then the correlation coefficient value is one or close to one, which refers to the excellence of decoding process [9, 13, 18, 27, 31]. \( r_{xy} \) is given as presented in equation (15) and equation (16):

\[
r_{xy} = \frac{\text{cov}(x,y)}{\sigma_x \sigma_y} = \frac{\sum_{i=1}^{N_S} (x_i - E(x))(y_i - E(y))}{\sqrt{\sum_{i=1}^{N_S} (x_i - E(x))^2} \cdot \sqrt{\sum_{i=1}^{N_S} (y_i - E(y))^2}} \]  
(15)

\[
E(x) = \frac{1}{N_S} \sum_{i=1}^{N_S} x_i 
\]  
(16)

\[
E(y) = \frac{1}{N_S} \sum_{i=1}^{N_S} y_i 
\]

Figure 4. The histograms of (a) Original, (b) Encrypted, (c) Decrypted voice signals.
where $\text{cov}$ and $\sigma$ refer to the covariance and standard deviation of the input signal $x$ and output signal $y$, respectively, $N_S$ denotes the number of voice samples utilized in computations, whilst $E$ represents the mean of $x$ and $y$. The obtained outcomes for correlation coefficient values between original, encoded and decoded voice samples for the four test signals are tabulated in Table 3. From this table, it can be found that $r_{xy}$ values for all the cipher signals are quite low, which implies the high dissimilarity between the input and output voice signals. On the other hand, $r_{xy}$ values for the deciphered signals are one, which exhibits the strong similarity between the original and restored voice files. Further, Figure 6 displays the samples distribution of correlation coefficient for plain, cipher and deciphered signals. It is clear that the samples of the plain signal are clustered together diagonally as in Figure 6a, whereas the samples of the scrambled signal are randomized as in Figure 6b. This means that the suggested technique can generate encoded signals of lower correlation between adjacent samples and it can also reconstruct perfectly the original signal as in Figure 6c. Table 3 and Figure 6 confirm the high performance of the introduced algorithm to withstand efficiently correlation analysis and statistical attack.

5.6. Differential analysis

In order to assess the voice encryption performance in resisting differential analysis, Number of Samples Change Rate and Unified Average Changing Intensity measures are usually utilized for this purpose. Original and modified voice files in this test are enciphered via one key which yields in two encrypted signals. Then, these generated encoded signals are compared by NSCR and UACI criteria [12, 19, 21]. NSCR and UACI are specified as in equation (17) and equation (18), respectively:

$$NSCR = \frac{\sum_{i} D(i)}{l} \times 100\%$$

$$D(i) = \begin{cases} 
0 & \text{if } x_1(i) = x_2(i) \\
1 & \text{if } x_1(i) \neq x_2(i) 
\end{cases}$$
\[ UACI = \frac{1}{l} \left[ \sum_{i} \frac{|x_1(i) - x_2(i)|}{255} \right] \times 100\% \]  

where \( l \) points to voice signal length, \( x_1 \) and \( x_2 \) symbolize to the scrambled signals whose input images are different by one sample only. The optimal scores for NSCR and UACI are 100% and 33.3%, respectively. The calculated outcomes for NSCR and UACI metrics for the test signals encrypted by the described mechanism are listed in Table 3. The results in this table manifest that NSCR and UACI scores produced via the proposed work are extremely near to their idealistic values, which proves the capability of the given approach to counter differential attack.

**Table 3.** \( r_{xy} \), NSCR and UACI results for correlation and differential analyses.

| File name   | Encryption | Decryption |
|-------------|------------|------------|
|             | \( r_{xy} \) | NSCR (%)   | UACI (%)   | \( r_{xy} \) |
| Signal 1    | -0.6070    | 99.6670    | 33.6580    | 1            |
| Signal 2    | -0.6319    | 99.6410    | 33.6721    | 1            |
| Signal 3    | -0.6398    | 99.6523    | 33.5135    | 1            |
| Signal 4    | -0.6448    | 99.6530    | 33.3688    | 1            |

5.7. **Impact of noise**

The capability of the system to withstand noise intervention during the transferring process of encoded voice signals is a significant indicator to verify the algorithm performance [13, 22, 26, 31, 32]. Different sorts of noise are added in the experiments in order to test the defending ability of the presented scheme against noise attacks. Gaussian noise, color noise (Pink noise) and Babble noise are implemented on the voice signal with intensity of 0.2, 0.1, 0.05, 0.04, 0.03, 0.02 and 0.01. The performance criteria SNR, PSNR, SNRseg, fwsNRseg, LLR, BER and \( r_{xy} \) are computed so as to prove the robustness of the introduced method against noise attacks.

**Figure 6.** Distribution of samples in correlation analysis for (a) Plain voice signal, (b) Cipher voice signal, (c) Decipher voice signal.
The outcomes of these measures are listed in Tables (4-6). Higher SNR, PSNR, SNRseg, fwSNRseg and $r_{xy}$ values, and lower LLR and BER values between the input source signal and deciphered signal after applying noise on the ciphered signal means better decryption performance. It can be seen from Tables (4-6) that SNR, PSNR, SNRseg, fwSNRseg, and $r_{xy}$ scores increase gradually whereas LLR and BER values decrease as the noise density decreases from 0.2 to 0.01, which demonstrates the invulnerability of the proposed cryptosystem against noise attack. Further, the values of quality metrics in Tables (4-6) are plotted in Figure 7 versus the added noise density. It can be noticed visually in Figure 7a that the SNR outcomes under Pink noise effect are higher than SNR under Gaussian and Babble noises effect at the same noise intensity, while PSNR values in Figure 7b under Gaussian noise attack are higher than those values under the other two added noises. Also, the computed SNRseg and fwSNRseg scores in Figures 7c and 7d, respectively in the case of Babble noise are generally better than the same metrics values under Gaussian and Pink noises influence. Besides, LLR results in Figure 7e after adding Babble noise are lower than those results with Gaussian and Pink noise impact, whereas the BER outcomes in Figure 7f with Pink noise addition are smaller than their outcomes with Gaussian and Babble noises effect. Finally, the correlation values in Figure 7g under Gaussian noise are greater than their corresponding values under Pink and Babble noises impact. Tables (4-6) and Figure 7 indicate that the described approach can effectively endure different kinds of noise attack applied on the encrypted voice signal with different levels.

Table 4. Outcomes of quality measures under Gaussian noise attack.

| Noise density | SNR (dB) | PSNR (dB) | SNRseg (dB) | fwSNRseg (dB) | LLR | BER | $r_{xy}$ |
|---------------|---------|-----------|-------------|---------------|-----|-----|---------|
| 0.2           | 6.9892  | 75.9887   | 7.8598      | 8.7324        | 0.8876 | 0.9924 | 0.9137 |
| 0.1           | 10.0021 | 79.0041   | 10.8644     | 11. 6781      | 0.8769 | 0.9505 | 0.9535 |
| 0.05          | 13.0218 | 82.0084   | 15.6421     | 13. 6390      | 0.8223 | 0.8617 | 0.9760 |
| 0.04          | 13.9882 | 83.0009   | 17.2970     | 15. 6427      | 0.8092 | 0.7957 | 0.9807 |
| 0.03          | 15.1864 | 84.1881   | 21.3335     | 17. 6321      | 0.7641 | 0.6960 | 0.9852 |
| 0.02          | 17.0012 | 85.9721   | 22.6261     | 18. 6214      | 0.6942 | 0.5895 | 0.9901 |
| 0.01          | 20.0080 | 89.0291   | 24.4267     | 21. 6052      | 0.6034 | 0.4953 | 0.9950 |

Table 5. Outcomes of quality measures under Pink noise attack.

| Noise density | SNR (dB) | PSNR (dB) | SNRseg (dB) | fwSNRseg (dB) | LLR | BER | $r_{xy}$ |
|---------------|---------|-----------|-------------|---------------|-----|-----|---------|
| 0.2           | 13.3259 | 38.5928   | 9.1267      | 13.3637       | 0.9923 | 0.9554 | 0.7445 |
| 0.1           | 15.7168 | 41.6103   | 11.4367     | 16.3355       | 0.9802 | 0.9415 | 0.7523 |
| 0.05          | 16.2479 | 42.4853   | 16.7057     | 19.8827       | 0.9660 | 0.7347 | 0.7661 |
| 0.04          | 17.3278 | 45.8939   | 18.1489     | 20.5162       | 0.8609 | 0.6467 | 0.8123 |
| 0.03          | 20.8977 | 46.2560   | 22.1143     | 21.3473       | 0.7755 | 0.5469 | 0.8528 |
| 0.02          | 22.4675 | 50.7924   | 25.2179     | 22.1129       | 0.7449 | 0.4522 | 0.8537 |
| 0.01          | 24.7927 | 51.3729   | 27.2458     | 23.9013       | 0.7288 | 0.3542 | 0.8551 |
Table 6. Outcomes of quality measures under Babble noise attack.

| Noise density | SNR (dB) | PSNR (dB) | SNRseg (dB) | fnwSNRseg (dB) | LLR | BER | r_{xy} |
|---------------|----------|-----------|-------------|----------------|-----|-----|-------|
| 0.2           | 9.9547   | 44.1231   | 13.8686     | 14.7733        | 0.7394 | 0.9805 | 0.9033 |
| 0.1           | 11.8333  | 45.8861   | 14.0146     | 16.8604        | 0.6075 | 0.9750 | 0.9430 |
| 0.05          | 14.8107  | 48.1164   | 16.3076     | 18.8794        | 0.4831 | 0.9675 | 0.9611 |
| 0.04          | 16.0189  | 51.2214   | 20.6683     | 22.9302        | 0.4478 | 0.8671 | 0.9760 |
| 0.03          | 19.5749  | 53.9688   | 21.7314     | 24.8770        | 0.4046 | 0.7771 | 0.9807 |
| 0.02          | 21.7821  | 57.2116   | 24.3590     | 26.9315        | 0.3394 | 0.6684 | 0.9854 |
| 0.01          | 24.5800  | 60.6062   | 28.8912     | 28.9061        | 0.2495 | 0.5682 | 0.9900 |

5.8. Time analysis

In addition to high security, high speed is also required for an ideal ciphering algorithm [13, 20]. The environment of the simulation experiments in this work has been aforementioned in Section 5. Encryption operation contains three main phases: modifying voice signal, key generation and encoding, whilst decryption operation involves also three major phases: key generation, decoding and re-modifying voice signal.

The consumed time of each encryption and decryption processes with the overall processing time for the four sample voice signals are reported in Tables 7 and 8, respectively. It can be observed from Table 7 that the key generation operation (public, private and session keys) possesses the shortest time, while the modifying voice signal operation possesses the longest time from the total encryption time. Also, it is evident that the consuming time for key generation is fixed in all cases because it is not depending on the length of the test voice file. Moreover, the time needed for each process increases as the voice file length increases. For instance, for the first test file (Signal 1) with length of 1.4150 seconds, the times required for the three mentioned ciphering operations are 0.015860, 0.000011 and 0.002585 seconds, respectively, which yields the overall time of 0.018456 seconds. On the other hand, for the last test file (Signal 4) with length of 4.1750 seconds, the corresponding consumed times for the same operations are 0.036211, 0.000011 and 0.004913 seconds, respectively, which results the overall time of 0.041135 seconds. Figure 8a explains this process, where the three major ciphering operations along with the final adding operation are plotted against their times for the four sample files. The times required for the key generation operation by the transmitter at encryption and by the receptor at decryption in Tables 7 and 8 are equal (0.000011 seconds). Additionally, it can be noticed in Table 8 that the consuming time for both decoding and re-modifying voice signal operations increases when the length of the voice signal increases. As an example, for Signal 1, the prerequisite times for these two operations are 0.018240 and 0.023262 seconds, respectively, which gives a total decryption time of 0.041513 seconds. On the other hand, for the last test file, the time required to decode the same signal is 0.042618 seconds, which results the overall decryption time of 0.079273 seconds. Figure 8 illustrates this process, where the main deciphering operations along with the final adding operation are plotted versus their times for the test signals. Furthermore, by comparing Tables 7 and 8, it is clear that the necessary time for encoding process is less than that for the decoding process, for example, the time needed to encode Signal 2 is 0.002761 seconds, whereas the time required to decode the same signal is 0.042618 seconds. Also, the overall encryption times for the all test files are less than the overall decryption times. The total encryption times for the sample signals are 0.018456, 0.025995, 0.039419, and 0.041135 seconds, respectively, while the total decryption times for the same signals are
0.041513, 0.074434, 0.077716, and 0.079273 seconds, respectively, which means that the decryption process at the recipient takes longer time than the encryption process at the sender. Figure 8c clarifies this process, where the four voice files are plotted versus their total encryption and decryption times. It can be deduced from Tables 7 and 8, and Figure 8 that the introduced approach can fulfill a fast speed for both encryption and decryption processes thereby providing good security performance.

**Figure 7.** Results of quality metrics under Gaussian, Pink and Babble noises attacks with different densities (a) SNR, (b) PSNR, (c) SNRseg, (d) fwSNRseg, (e) LLR, (f) BER, (g) $r_{xy}$. 
Table 7. Total encryption time of the presented method.

| File name | Time of modifying voice signal (sec) | Time of key generation (sec) | Time of encoding (sec) | Total encryption time (sec) |
|-----------|-------------------------------------|-----------------------------|-----------------------|---------------------------|
| Signal 1  | 0.015860                            | 0.000011                    | 0.002585              | 0.018456                  |
| Signal 2  | 0.023223                            | 0.000011                    | 0.002761              | 0.025995                  |
| Signal 3  | 0.035048                            | 0.000011                    | 0.004360              | 0.039419                  |
| Signal 4  | 0.036211                            | 0.000011                    | 0.004913              | 0.041135                  |

Table 8. Total decryption time of the presented method.

| File name | Time of key generation (sec) | Time of decoding (sec) | Time of re-modifying voice signal (sec) | Total decryption time (sec) |
|-----------|-----------------------------|------------------------|----------------------------------------|-----------------------------|
| Signal 1  | 0.000011                    | 0.018240               | 0.023262                               | 0.041513                   |
| Signal 2  | 0.000011                    | 0.042618               | 0.031805                               | 0.074434                   |
| Signal 3  | 0.000011                    | 0.044397               | 0.033308                               | 0.077716                   |
| Signal 4  | 0.000011                    | 0.045951               | 0.033311                               | 0.079273                   |

6. Comparison with other schemes

Tables 9 and 10, respectively display the comparison outcomes for the performance criteria of the described method and other similar techniques in encryption and decryption cases. The quality metrics utilized in this comparison are SNR, SNRseg, LLR, r_{xy}, NSCR, UACI and encryption time (ET) for encryption process in Table 9, whilst SNR, PSNR, SNRseg, fwSNRseg, LLR and r_{xy} performance metrics are utilized in the comparison for decryption process in Table 10. According to Table 9 in the encryption process, it can be found that the SNR and SNRseg scores produced by the introduced algorithm are lower than references [9, 10, 17, 18, 19, 27, 29, 31], which indicates that the noise power is much greater than the signal power and hence the possibility of detecting the transferred voice signal is extremely hard for the attacker. However, LLR value is lower than references [10, 27, 29, 31]. Also, the obtained r_{xy} value is smaller than references [5, 9, 10, 12, 18, 19, 21, 27, 31], which demonstrates that the plain and cipher voice signals are entirely unlike. Additionally, the NSCR result is lower than references [12, 18, 21] but it is larger than reference [19]. On the other hand, the acquired UACI score is better than the methods in other references except for reference [19]. Finally, the encryption time of the presented method is much smaller than the existing techniques in Table 9, which means that the computation complexity of the introduced mechanism is greatly minimized, and it is relatively lower than other schemes. The comparison outcomes in Table 9 prove the superiority of the described approach over the compared methods in most of performance criterion in the enciphering operation.
Table 9. Comparison results of the presented algorithm and other methods in terms of quality measures in encryption case.

| Method     | SNR (dB) | SNRseg (dB) | LLR r_{xy} | NSCR (%) | UACI (%) | ET (s) |
|------------|----------|-------------|-------------|-----------|----------|--------|
| Ref. [5]   | -        | -           | 0.0051      | -         | -        | 1.4343 |
| Ref. [9]   | -23.89   | -           | 0.000236    | -         | -        | -      |
| Ref. [10]  | -41.05   | -55.2       | 1.92        | -0.00118  | -        | -      |
| Ref. [12]  | -        | -           | 0.0233      | 99.9982   | 33.1197  | -      |
| Ref. [17]  | -16.0483 | -           | -           | -         | -        | 0.130  |
| Ref. [18]  | -133     | -           | 0.000991    | 99.9989   | 33.3421  | -      |
| Ref. [19]  | -12.45   | -           | 0.000136    | 99.6320   | 33.6823  | -      |
| Ref. [21]  | -        | -           | 0.0386      | 99.924    | 33.348   | 0.032  |
| Ref. [27]  | -11.7794 | -14.5452    | 1.8708      | -0.00203  | -        | -      |
| Ref. [29]  | -2.62555 | -2.56744    | 4.18696     | -         | -        | -      |
| Ref. [31]  | -        | -17.70727   | 2.9336      | -0.009221 | -        | -      |
| Proposed   | -39.5803 | -27.7054    | 1.0191      | -0.6070   | 99.6670  | 33.6580 | 0.018456 |

Table 10. Comparison results of the presented algorithm and other methods in terms of quality measures in decryption case.

| Method | SNR (dB) | PSNR (dB) | SNRseg (dB) | fWSNRseg (dB) | LLR r_{xy} |
|--------|----------|-----------|-------------|---------------|-------------|
| Ref. [10] | 240.16   | -         | 80.15       | -             | 1           |
| Ref. [12] | 33.7464  | 59.7989   | -           | -             | 0.999       |
| Ref. [21] | 123.57   | 50.21     | 121.25      | 55.46         | 0.9958      |
| Ref. [26] | -        | -         | -           | -             | 8.91 \times 10^{-4} | 0.98 |
| Ref. [27] | 63.069   | -         | 91.9052     | -             | 6.13 \times 10^{-6} | 1 |
| Ref. [29] | 13.67320 | -         | 62.84561    | -             | 0.002382    | - |
| Ref. [31] | -        | -         | 62.25391    | 63.1322       | 0.0376      | 0.992882 |
| Proposed | 246.6736 | 303.9110  | 257.6707    | 60.7456       | 5.4581 \times 10^{-14} | 1 |
According to Table 10 in the decryption process, it is obvious that the SNR, PSNR and SNRseg results achieved via the proposed scheme are much greater than the compared methods. Further, the obtained fwSNRseg scores are higher except for reference [31]. Besides, LLR value is much lower than the techniques in other references. Eventually, correlation value is equal or greater than those values in existing approaches, which reflects the perfect restoring performance of the suggested method. The results of comparison in Table 10 confirm that the presented algorithm outperforms other similar voice cryptosystems in most of assessment metrics in the deciphering operation.

7. Conclusions and future work

In this article, a new approach for voice data encryption and decryption based on Diffie-Hellman algorithm has been presented. Public and private keys of each user are used to compute the shared secret key $K$ which used to encrypt and decipher voice signals through this scheme. The knowledge of $p,g, g^a \mod p$ and $g^b \mod p$ values is not enough to calculate the formula $K = g^{ab} \mod p = g^{ba} \mod p$, therefore resolving this problem by the opponent demands quite long time. This issue mathematically is known as discrete logarithm which represents the strength point of Diffie-Hellman system. Once the sender and receptor compute $K$, they can employ it to encode and decode the voice messages across unsecure open medium. The security performance of the presented voice cryptosystem is assessed on different kinds of original voice signals via various quality criteria including SNR, PSNR, SNRseg, fwSNRseg, LLR, BER, histogram analysis, spectrogram analysis, correlation analysis and differential analysis. The outcomes reveal that the proposed scheme possesses high degree of confidentiality and security in both encryption and decryption stages and withstands many known attacks, thereby it can be utilized for encoding all sorts of voice files. Moreover, a comparison with other similar existent approaches by the means of quality metrics in both enciphering and deciphering stages is made. The comparison results signify that the proposed method presents more security upon some of the existent techniques. In addition, this mechanism endures various sorts of noise attacks. Time analysis shows that the encryption and decryption times are reasonably short, thereby reducing the computation complexity of the proposed work. All these results achieve the goal
of good voice cryptosystem, which is protecting the voice file content during transmission through insecure open networks, and at the same time maintaining the high quality of the decrypted signal at the receptor side. Therefore, the presented system possesses the cryptographic demands such as security, secrecy, confidently as well as immunity against intruders and hence it can be implemented in practical voice encoding applications. As a future work, this method can be integrated with other techniques like chaotic systems so as to improve the robustness of voice cryptosystem against noise attacks.

References

[1] Imran O, Yousif S, Hameed I, Abed W and Hammid A, 2020, International Conference on Computational Intelligence and Data Science (ICCIDS 2019), (India: Elsevier), Implementation of El-Gamal Algorithm for Speech Signals Encryption and Decryption, pp. 1028–1037.

[2] Mendoza E M, Jiménez-Vázquez C A, Silva-García V M and Flores-Carapia R, 2013, Journal of Engineering (IOSRJEN), Vol. 3, Issue 7, Diffie-Hellman Protocol Based on Elgamal and AES Cryptosystems, pp. 30-33.

[3] Ali S, Humaria A, Ramzan M, Khan I, Saqlain S, Ghani A, Zakia J and Alzahrani B, 2020, International Journal of Distributed Sensor Networks, Vol. 16, No. 6, An efficient cryptographic technique using modified Diffie–Hellman in wireless sensor networks, pp. 1-24.

[4] Yousif S, 2018, International Journal of Engineering Technologies and Management Research, Vol. 5, Issue 7, Encryption and Decryption of Audio Signal Based on RSA Algorithm, pp. 57-64.

[5] Elshamy E, El-Rabaie E, Faragallah O, Elshakankiry O, El-Samie F, El-sayed H and El-Zoghdy S, 2015, Int J Speech Technol, (Springer), Efficient audio cryptosystem based on chaotic maps and double random phase encoding.

[6] Yousif S, 2019, Journal of Engineering and Applied Sciences, Vol. 14, No. 17, Speech Encryption Based on Zaslavsky Map, pp. 6392-6399.

[7] Jung Y and Festijo E, 2014, Journal of Computer and System Sciences, (Elsevier), One-time packet key exchange scheme for secure real-time multimedia applications, pp. 1-13.

[8] Yousif S, 2018, International Journal of Engineering & Technology, Vol. 7, No. 4, A new speech cryptosystem using DNA encoding, genetic and RSA algorithms, pp. 4550-4557.

[9] Farsana F J and Gopakumar K, 2016, 6th International Conference on Advances In Computing & Communications, ICACC 2016, (India: Elsevier), A Novel Approach for Speech Encryption: Zaslavsky Map as Pseudo Random Number Generator, pp. 816 – 823.

[10] Abd Elzaher M F, Shalaby M and El Ramly S H, 2016, International Journal of Computer Applications, Vol. 135, No. 9, Securing Modern Voice Communication Systems using Multilevel Chaotic Approach, pp. 17-21.

[11] Sharma J and Rani J, 2017, International Journal of Computer Science and Mobile Computing, Vol. 6, Issue 1, An Efficient Hybrid Approach for Secure Speech Cryptography, pp. 23-29.

[12] Sathiyamurth P and Ramakrishnan S, 2017, EURASIP Journal on Audio, Speech, and Music Processing, (Springer), Speech encryption using chaotic shift keying for secured speech communication, pp. 1-11.

[13] Sheela S J, Suresh K V and Tandur D, 2017, Journal of Computer Networks and Communications, A Novel Audio Cryptosystem Using Chaotic Maps and DNA Encoding, pp. 1-12.

[14] Dissanayake W, 2018, International Journal of Computer Applications Technology and Research, Vol. 7, Issue 02, An Improvement of the Basic El-Gamal Public Key Cryptosystem, pp. 40-44.

[15] Narendren S, Yathish Y and Chandra B, 2018, International Journal of Pure and Applied Mathematics, Vol. 119, No. 7, A Cryptosystem using Two Layers of Security - DNA and RSA Cryptography, pp. 217-224.
[16] Agarwal A, Singh P and Katiyar S, 2019, International Journal of Computer Applications, Vol. 178, No. 22, Secured Audio Encryption using AES Algorithm, pp. 29-33.

[17] Kordov K, 2019, Journal of Electronics, Vol. 8, A Novel Audio Encryption Algorithm with Permutation-Substitution Architecture, pp. 1-15.

[18] Farsana F J, Devi V R and Gopakumar K, 2019, Applied Computing and Informatics, (Elsevier), An audio encryption scheme based on Fast Walsh Hadamard Transform and mixed chaotic keystreams, pp. 1-11.

[19] Farsana F J and Gopakumar K, 2020, Advances in Mathematical Physics, Speech Encryption Algorithm Based on Nonorthogonal Quantum State with Hyperchaotic Keystreams, pp. 1-12.

[20] Al-kateeb Z N and Mohammed S J, 2020, Multimedia Tools and Applications, (Springer), A novel approach for audio file encryption using hand geometry.

[21] Sathiayamurthi P and Ramakrishnan S, 2020, Multimedia Tools and Applications, (Springer), Speech encryption algorithm using FFT and 3D-Lorenz–logistic chaotic map.

[22] Zoghdy S, El-sayed H and Faragallah O, 2020, Wireless Personal Communications, (Springer), Transmission of Chaotic-based Encrypted Audio Through OFDM.

[23] El Abbadi N, Abaas S and Abd Alaziz A, 2016, International Journal of Advanced Research in Computer and Communication Engineering, Vol. 5, Issue 1, New Image Encryption Algorithm Based on Diffie-Hellman and Singular Value Decomposition, pp. 197-201.

[24] Loriya H, Kulshreshtha A and Keraliya D, 2017, International Journal of Advanced Research in Computer and Communication Engineering, Vol. 6, Issue 2, Security Analysis of Various Public Key Cryptosystems for Authentication and Key Agreement in Wireless Communication Network, pp. 267-274.

[25] Kaushik A and Satvika, 2013, Special Issue: Proceedings of 2nd International Conference on Emerging Trends in Engineering and Management, ICETEM 2013, International Journal of Advances in Engineering Sciences, Vol. 3, No. 3, Extended Diffie-Hellman Algorithm for Key Exchange and Management, pp. 67-70.

[26] Mosa E, Messiha N, Zahran O and El-Samie F, 2011, Int J Speech Technol, (Springer), Chaotic encryption of speech signals, pp. 285-296.

[27] Wahab H and Mahdi S, 2015, International Journal of Emerging Trends & Technology in Computer Science (IJETTCS), Vol. 4, Issue 2, Modify Speech Cryptosystem Based on Shuffling Overlapping Blocks Technique, pp. 70-75.

[28] Hasan F, 2016, Eng. & Tech. Journal, Vol. 34, Part (A), No.11, Speech Encryption using Fixed Point Chaos based Stream Cipher (FPC-SC), pp. 2152-2166.

[29] Al-saad S and Hashim E, 2013, Al- Mustansiriyah J. Sci., Vol. 24, No. 5, A Speech Scrambler Algorithm Based on chaotic system, pp. 357-372.

[30] Matalgah M, 2018, 2018 IEEE 88th Vehicular Technology Conference (VTC-Fall), (USA: IEEE), Error Performance of Cryptography Transmission in Wireless Fading Channels, pp. 1-4.

[31] Al Saad S and Hato E, 2014, International Journal of Computer Applications, Vol. 93, No. 4, A Speech Encryption based on Chaotic Maps, pp. 19-28.

[32] Krishnamurthy N and Hansen J, 2009, IEEE Transaction ON Audio, Speech, AND Language Processing, Vol. 17, No. 7, Babble Noise: Modeling, Analysis, and Applications, pp. 1394-1407.