Secure Audio Conversation by Steganography Based on Diffie–Hellman and One-time pad

Dadmehr Rahbari, Yaghoub Farjami, Faranak Fotouhi Ghazvini

Abstract

Different applications same personal, business, military, healthcare and many others need secure audio transferring of signals. Attackers are in ambush and access to these voices will be dangerous for owners. Cryptography caused to high changes and noise on audio. Steganography is a good method for set crypto audio in cover voice to attacker can’t doubt to it. In our proposed method Diffie–Hellman exchange key and one-time pad algorithms used for secure session and wavelet transformation and exclusive-or help us for steganography of encrypted audio in cover audio. We implement and analysis our algorithm by different frequencies and bit rates then compare it with other steganography methods. High transparency in cover audio, fast cryptography in steganography and speed up transferring with vast hidden bit counts convert our proposed method to the novel and optimized idea for secure voice signals.

Keywords: Steganography, Cryptography, One-time pad, Diffie–Hellman.

1. Introduction

The conversation between two humans can be by face to face, phone, network or the internet and so on. Also, machine to machine communications has many applications in ubiquitous computing and internet of things. Security in communications is the important problem and it provided with different methods. Cryptography is a good technique for data hiding that it use a secret key and encrypt data. The stream ciphers work on the input stream and encrypt it. The Audio cipher is a combination of linear feedback shift registers with irregular clocking and a non-linear combiner. These methods are suitable for audio data in GSM mobile phone but are attacked at a few seconds (Güneysu et al 2008). When data is audio then encryption make so noise on voice and the third person recognize it and try to decrypt audio. The high level secure applications need to use another method.

The Steganography technique is the art and science of data hiding. This method hides the private data in cover data after the encryption, the cover audio make an audio secret is called host audio. The content of host or cover audio when modified is called audio steganography. Audio hiding in another audio is different by image and video. The hiding process is more difficult than embedding data in other media. In an audio, the minimum changes are detectable for the third person but image and video have many positions for data hiding without high detectability. Audio steganography can use for maintaining the transmitted information in insecure channels. There are many methods for audio hiding and each one of those has advantages and disadvantages. Some methods often used for low length data as author copyright or hidden text to audio and don’t use for audio steganography in another audio.
Performance parameters of audio steganography are security, imperceptibility, high capacity, resistance and accurate extraction (Meligy et al 2005). The graphical presentation of audio signals, SNR (signal to noise ratio), the PSNR (peak signal to ratio) and runtime are our performance test for proposed proof.

In the following, section 2 summarizes related work on audio steganography. Exchange key technique discussed in section 3. In section 4, we exactly explain the proposed method for steganography also present the simulation results and analysis.

Related Work

In the least significant bit (LSB) method, the lowest bit of each 8 bits audio stream replace with other audio bits. This work doesn’t have noticeable changes in original audio. The LSB method can’t store large data in the audio stream and only use for small data as digital signature or text. The LSB data is easily detectable so that attacker can restore hiding bits. In (Meligy et al 2005) an audio steganography algorithm used for hiding of text, audio or image by wavelet transform, the least significant bit method and three random keys for robustness increasing of the LSB, according the result of SNR and PSNR values, algorithm works good for data with low length also caused by use of CA rather CD coefficients, slow processing is another problem of it. The LSB method has problems as easy to extract, noise addition, data destroy caused by compression (Tayel et al 2016). Another technique for steganography is the use of the higher bit in samples (Gopalan, 2016). According to this method, hidden audios with higher bits are so fragile via attacks or changes. Some low level noise was added to the hidden audio in each case and as a result SNR value was good for TIMIT database. Authors in (Paira and Chandra, 2016) presented the LSB method by cryptography based on symmetric key for text hiding in audio. They have used shared key for encryption and decryption process of a text file to an audio file without change of audio file header. Phase encoding is based on an original audio shifting to forward and real audio set instead. Detectability of hiding bits is hard but the audio length will be increased. Phase encoding via LSB has length increment and security increment. The human hearing ability of audio frequencies is the method for data hiding in audio signal (Paul and Raychoudhury, 2011). They have divided the cover audio to the segment with lower 0.1 second distance and set the hidden data in it. Human hear can’t recognize this low distance. This work don’t have any cryptography method also the length of the output signal is large and it is detectable for the attacker.

Echoes insertion on audio with initial amplitude, decay rate, and short delay parameters is other method for data hiding with minimum effect on original voice. Human ear can’t listen audio with amplitude and decay rate less than special values. So the voice data could mix on echoes of original audio without noticeable effect (Farmer and Johnson, 2014). With spread spectrum method, audio data can hide on masked frequencies of original audio. The encoded message combined with a random signal then it interleaved with cover signal and send to another side. This process is reversed for audio extraction (Singh, 2016). In (Ghanwat and Rajan, 2013) spread spectrum is used for copyright hiding of an audio. By this method, the cover audio stream is transformed using FFT analysis then the information added to frequencies components using spread spectrum technique and transform back the audio stream into time domain audible range. This method is high secure for audio transfer but when an audio hide on a cover audio then the noisy voice have noticeable effects, therefor man in the middle doubt to voice and try to discover hiding data (Djebar et al 2012). The spread spectrum has problems as high bandwidth consumption and low security for time scale modification (Tayel et al 2016).

Lower power tone placement in the middle of higher tones is another technique. This technique selects two known frequencies with high power and set low power data on cover audio (Gopalan and Wenndt 2004). The main problem of this method, this is that an attacker can use low pass filter and bit truncation for extract hiding data. In another work, private messages encrypted by advanced encryption standard (AES) 128bits and key distribution and then bits embedded in cover audio. At the receiver also reverse operation (Tang et al 2014). Authors could transfer audio steganography on the internet but symmetric AES cryptography is an olden method and breaks easily by attackers. The silent regions in the speech on VOIP are used for steganography (PS and Vinayagam 2012), authors calculate the entropy of audio frame then set the data in frames with low energy. They added a header to each message and use it for data decoding. This method hides more data messages in audio but have much overhead.

Another method for steganography is the use of RSA cryptography. In (Saurabh and Ambhaikar, 2012), Authors encrypt the message by RSA and add noise to message. They use the random LSB method for steganography by...
genetic algorithm. This method has high run time. Also, they transferred the small messages. The RSA algorithm isn’t suitable for audio steganography. So our proposed method based on the audio cipher. In Internet of things (IOT), machines transfer data together on the insecure channel. In (Singh et al 2016), authors proposed a method for audio steganography in the 5G platform but not implemented it. They presented their methods for the audio passwords to protect the database that it is low length data and IOT devices could steganography the audio with low capacity power. An audio stream cipher with the symmetric key, a plain audio is encrypted one at a time with the corresponding bits of the key stream, to give a bit of the cipher audio stream. Encryption of bits is dependent on the current state of the cipher, it is also known as state cipher. For fast encryption of stream, exclusive-or is used. RC4, A5/1, A5/2 are used for audio stream cipher but that aren’t sufficient security (Güneysu et al 2008). RC4 is an olden method that hacked. A5/1 and A5/2 are the newer technique but easily hacked on GSM phone (Barkam et al 2008).

**Diffie–Hellman with One-time pad**

Diffie–Hellman key exchange is a method of securely exchanging cryptographic keys over a public channel. This protocol is an asymmetric cryptography technique, therefore the sides of relation don’t need before the agreement and share key is built on the start of a session. This method has capability expansion of use for multi side communications. The implementation of the protocol uses the multiplicative group of integers modulo \( p \), where \( p \) is prime, and \( g \) is a primitive root modulo \( p \). These two values are chosen in this way to ensure that the resulting shared secret can take on any value from 1 to \( p-1 \). Alice and Bob agree to use a modulus prime number of \( P \) and base \( g \) generate by one-time pad method. The value of \( g \) is different in each session, therefore attacker can’t use those in after communications. Alice and Bob have secret key by \( a \) and \( b \). Alice send Bob \( A = g^a \mod p \) and Bob sends Alice \( B = g^b \mod p \). Alice computes \( s = B^a \mod p \). Bob computes \( s = A^b \mod p \). Alice and Bob reach a shared key. Our idea for more security at Diffie–Hellman is one-time pad on base value.

A one-time pad uses a key stream of completely random digits that independently generated of the plain audio and cipher audio stream. The key stream is combined with the plain audio digits one at a time to form the cipher audio. The key stream must be generated completely at random with at least the same length as the plain audio and cannot be used more than once. This makes the system cumbersome to implement in many practical applications. Key generation, distribution and management are critical for those applications (Paar and Pelzl 2009). The random digit of one-time pad method only use for one session and it builds again for after sessions. So an attacker can’t analysis the before audio transferred stream for key finding. Binary stream ciphers are often constructed using linear feedback shift registers (LFSRs) because they can be easily implemented in the hardware and can be readily analyzed mathematically. The use of LFSRs on their own, however, is insufficient to provide good security. Various schemes have been proposed to increase the security of LFSRs (Geremia 2016). If Alice or Bob reuse keys then attack probability is high. This meant that there was a high likelihood that it be used twice if more than a few thousand packets were sent with the same master key (see birthday attack), subjecting the packets with duplicated key reuse attack. Suppose an adversary knows the exact content of all or part of one of our messages. As a part of a man in the middle attack or replay attack, he can alter the content of the message without knowing the key. Bit-flipping attacks can be prevented by including message authentication code to increase the likelihood that tampering will be detected. Our solution for two above problems is one-time pad by current system time that it is different at the sessions.

**Proposed Method**

The auditory system of the human is sensitive to small amplitude audios. Since LSB method can’t hide large data in cover audio, therefore it needs to the extension. In our technique more audio data placed in cover audio. This method use of 17 low level bits of each 32 bit covers audio for data hiding instead of one bit on LSB method. In a result, our extended method not only is suitable for audio with normal volume but also it can steganography the high percent of input audio stream. Audio compression by wavelet transform, decrease the original private voice that twice uses of it can audio volume decrease to quadrant. Before the start of steganography, we exchange keys. According to the proposed algorithm, Alice runs program on pc1 then communication will initialize. Diffie–Hellman exchange key
operation begins and Generate One-time pad value (g), Large Prime number (p), Alice secret key (a) that will send to pc2. In this time Bob by pc3 read share directory from pc2 and receive A, g, p then calculate $B = g^b \mod p$ and send B to PC2 of course share Key is $A^b \mod p$. The share Key on pc1 is equal $B^a \mod p$. When exchange key finished then plain audio streams from pc1 and pc3 read and save in pc2. Pc3 and pc1 read those and next step wavelet transform run on plain audio twice. Plain bits exclusive-or with share key then in the final step, all plain bits set on cover audio and send to share directory on pc2.

Fig 1. Steganography on Voice Communication

In steganography process, we have multi steps to doing. In first plain audio bit reduced, it filtered by wavelet transform twice. This transform reduced input volume to half. In second each 17 low level bits of 32 bit cover for the hiding of plain audio bits. So plain audio volume convert to quadrant. The algorithm encrypts plain bits and decrypts cipher bits by exclusive-or with share key on two way. These operations caused fasting execution and reduced time delay for the listener. The pseudo code of proposed algorithm is following:

Algorithm 1. Steganography

1. PC1 Start Communication
2. Diffie–Hellman Exchange Key
   2-1 Generate One-time pad value (g), Large Prime number (p), Alice secret key (a)
   2-2 Send g, p, $A = g^a \mod p$ to PC2
2-3 Bob receive A, g, p from PC2, calculate $B = g^b \mod p$ and send B to PC2
2-4 PC3 calculate share Key = $A^b \mod p$
3. Read plain Audio Stream and buffering from microphone
4. Twice Wavelet Transform of plain Audio
5. AudioEncrypted = XOR (plain Audio Stream, Key)
6. For each 32 bits of Audio cover
   6-1 Replace 17 lowest bits of Audio cover by AudioEncrypted bits
7. Send new Audio Cover to PC2

Algorithm 2. Desteganography

1. PC3 receive Audio
2. For each 32 bits of Audio cover
   2-1 Read 17 lowest bits of Audio cover that is AudioEncrypted
3. PlainAudio= XOR (new Audio Stream, Key)
4. Play PlainAudio from Speaker

In Fig 1, DWT is discrete wavelet transform. A, g, p, B values exchange at begin of communication. The voices from PCs would transfer after key exchange operation. PC1 and PC3 prepared by the microphone for voice receive.
and speaker for voice play. PC2 is a shared place for both PC1 and PC3. PCs are symbols of different devices in the real environment as computer or phone or tablet or other devices with processor and memory. The cover voice is an audio that read and saved previously. If each person eavesdrops audio not only listened voice is fully natural conversation but also it is not recognizable. So Alice and Bob dialogue with together but eavesdropper can’t understand the real dialogue between there.

**Simulation and Results**

We used Matlab software for Simulation environment. This software has libraries for audio analysis (Giannakopoulos and Pikrakis, 2014) and it has a good runtime. Our system for execution of Alice, Bob and share PC have properties included Intel core i5 CPU 2.67 Giga Hertz, 3 Gigabyte RAM with Microsoft windows10 32-bit. Cover voice read with frequency equal 16000 sample, bit count equal 16 bits and save in a wave file then it used in algorithm1 for steganography. Alice and Bob’s voices read with 16000, 11000, 8000 sample and bit count equal 8 bits. By fewer bits in plain audio, we reduced time delay. For play voice for Alice and Bob, frequencies are equal.

![Fig. 2. Cover Audio before steganography](image)

The audio cover can be before the conversation between Alice and Bob that is not important and don’t need security. Varied audios used for covering to the attacker will not doubt to steganography audio. Fig. 2 is frequencies of audio cover via time.

![Fig. 3. Cover Audio after twice wavelet transform](image)

In Fig. 3, we show the result of twice wavelet transform on cover audio. Audio Tests don’t exist high difference between audios at after and before transform for the human. Fig. 4 show that the Cover Audio After steganography. Whereas we set 17 bits of each 32 bits in cover audio so it doesn’t have extra changes. The cover audio in pc1, pc2, pc3 is fully transparence and attacker will not doubt.
Fig. 4. Cover Audio After steganography

Fig. 1 show the Input plain audio from Alice or Bob PC. We hid this audio in cover audio after twice wavelet transform and cryptography operations.

Fig. 5. Input audio from Alice or Bob pc

The analysis of simulation results shows that transparency in cover audio is high and eavesdropper will not listen to noise or time delay in audio. Also, if the man in the middle receives audio between Alice and Bob, he will not capable listen the real audio. Steganography for hiding real audio in other audio and encryption for cipher audio build grand obstacle via attacker but with minimum signs of cryptography.

Table 1. Runtime comparison

<table>
<thead>
<tr>
<th>Step</th>
<th>AES</th>
<th>Proposed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cryptography Step</td>
<td>9.67 s</td>
<td>0.2971 s</td>
</tr>
<tr>
<td>Full Steganography</td>
<td>66.6310 s</td>
<td>10.22 s</td>
</tr>
</tbody>
</table>

Cryptography of the proposed method is Diffie–Hellman with one-time pad key and exclusive-or operator that it run with minimum time delay. Also, we execute AES cryptography. In comparison with our method, AES have the very long runtime. Table1 show that our proposed method for steganography is much better than AES mode. We exchange key between Alice and Bob at the start of the conversation with time spending. In AES, Alice and Bob had the public key before the conversation. The SNR (signal to noise ratio), the PSNR (peak signal to ratio) and MES (Mean Square Error) between cover and steganography audios is calculated by the following formula:

$$\text{SNR} = 10 \times \log \left( \frac{\sum_{n=1}^{N} \text{Cover}^2(n)}{\sum_{n=1}^{N} (\text{Cover}^2(n) - \text{Stego}^2(n))^2} \right) \quad (1)$$

$$\text{MSE} = \frac{\sum_{n=1}^{N} (\text{Cover} - \text{Stegano})^2}{\text{Size(Cover)}} \quad (2)$$
PSNR = 10 \cdot \log\left(\frac{\max(\text{Cover})}{\text{MSE}}\right) \quad (3)

In the above equations, the cover is the original cover audio, stego is the output of steganography and N is the numbers of samples in the input cover signals and max is the maximum value of the samples. We gathered audio from 10 men and woman by a microphone that frequency value equal 16000 and bit rate was 8.

![SNR & PSNR vs Sampling Rate](image)

**Fig. 6.** SNR and PSNR values for different input audio from microphone with same cover audio

Fig. 6 show that SNR and PSNR value for 10 human audios with same cover audio that values aren’t very different and all those close to mean value. The last Column is the average of each row in the chart. The SNR and PSNR value increment with fewer sampling rate. In out method, each 17 bit of 32 bit frame of cover audio is replaced also cryptography and wavelet transform run on the signal. So in proposed method, the sample numbers and noise level in the audio signal have a direct relation. We reach to better results in SNR values and data hiding length than (Devi and Pugazhenthi, 2016) paper that it used of LSB method.

**Conclusion**

The audio steganography doesn’t have any sign for attacker, therefore he will not doubt to audio. If attacker discover steganography with minimum probability but cryptography is the after obstacle. Attacker doesn’t have share key of Alice and Bob for audio decryption. Also, bit counts for steganography is to hide. The one-time pad method caused to generating of the random number on different communication. If the attacker breaks the one-time pad on one session but on after session the pad is new and not detectable. High speed steganography is a good method for audio sending. According to results of SNR and PSNR analysis, the sample numbers and noise level in the audio signal have direct relation. These results are very good for fast processing and transfer of signal. Our proposed method is usable in GSM communications and VOIP and applications with audio transfer.

**References**


Gopalan, K. and Wenndt, S. (2004), Audio Steganography for Covert Data Transmission by Imperceptible Tone Insertion, *WOC, (Banff, Canada)*.


