Audio Secret Management Scheme Using Shamir's Secret Sharing

M. Abukari Yakubu[1], Namunu C. Maddage[2], and Pradeep K. Atrey[1,3]

[1] Department of Applied Computer Science, University of Winnipeg, Winnipeg, MB, Canada
[2] NextGmultimedia, Melbourne, Australia
[3] Department of Computer Science, University at Albany - State University of New York, Albany, NY, USA

Abstract. Audio Secret Sharing (ASS) is a technique used to protect audio data from tampering and disclosure by dividing it into shares such that qualified shares can reconstruct the original audio data. Existing ASS schemes encrypt binary secret messages[1] and rely on the human auditory system for decryption by simultaneously playing authorized shares. This decryption approach tends to overburden the human auditory system when the number of shares used to reconstruct the secret increases [3]. Furthermore, it does not create room for further analysis or computation to be performed on the reconstructed secret since decryption ends at the human auditory system. Additionally, schemes in [2], [3], [4], [6] do not extend to the general (k, n) threshold. In this paper we propose an ASS scheme based on Shamir's secret sharing, which is (k, n) threshold, ideal[2], and information theoretically secure and it provides computationally efficient decryption.

Keywords: Shamir's secret sharing scheme (SSS), threshold schemes, information theoretically secure.

1 Introduction

Audio is one of the key types of multimedia content which may contain confidential information such as names, addresses, social security numbers, credit card numbers, evidence to be used in a court of law by a jury, and information with national security implications. Such sensitive information might be misused when it falls into the wrong hands. For instance, call centers record several hours of customer calls, most of which contain confidential information. In order to save cost, call centers often store data on Cloud Data Centers (CDCs). A rogue or malicious employee within the call center or CDC may use this confidential information to their own benefit. Therefore the security of such sensitive audio records is of utmost importance.

[1] A binary representation of a secret plaintext message.
[2] In a perfect secret sharing scheme, any unauthorized subset of participants cannot obtain any information about the secret. We will say that a perfect sharing scheme is ideal if all of the shares are from the same domain as the secret.

X. He et al. (Eds.): MMM 2015, Part I, LNCS 8935, pp. 396–407, 2015.

One way to secure the audio could be to encrypt it using Advanced Encryption Standard (AES). However, AES suffers from single point vulnerability meaning that the security of the method lies in securing the encryption key which is usually entrusted to the sender and receiver. This problem can be overcome by employing a secret sharing scheme to divide the audio secret into a number of shares and distribute them among a number of participants such that only more than a certain number of participants can reconstruct the secret by putting their shares together; individual shares are of no use on their own. Thus, a group of participants collectively protect and control access to the secret. In this case, the audio shares are distributed amongst multiple CDCs. Unless the required number of CDCs are compromised, an adversary cannot get the secret audio.

Some of the existing Audio Secret Sharing (ASS) schemes [2], [3] are designed to encrypt text secrets. In these schemes a binary representation of the text secret is embedded into an audio cover and shares of the cover signal are created. This approach combines cryptography to encrypt the plaintext and steganography to hide the existence of the ciphertext. Such schemes only had $(2, n)$ threshold and never extended to the general (k, n). The ciphertext was decrypted by the Human Auditory System (HAS) by simultaneously playing authorized shares which is analogous to Visual Cryptographic System (VCS) where the human visual system is used for decryption in image secret sharing. There is no computational cost to decrypt with HAS, however it has following limitations: 1) People with hearing impairments cannot participate in the decryption process 2) It requires manpower to decrypt the secret and also overburdens the human ear with increasing numbers of shares required to reconstruct the secret [3]. While schemes proposed in [2], [3] encrypt a binary secret message, schemes in [5], [6], [7] encrypt an audio secret. However, decryption still requires the human auditory system.

The scheme proposed in [5] is (k, n) threshold secret sharing scheme, where k out of n generated secret shares are required to reconstruct the secret audio. The security of this scheme is not proven from an information theoretical point of view and is highlighted in [6], [7]. Authors in [6], [7] propose schemes whose security is evaluated in terms of the mutual information between secret and shares from an information theoretical perspective. The scheme in [7] is an improvement to [6] where the encryption function uses normal distribution over a bounded domain in order to create bounded shares. However, both schemes do not extend to (k, n) threshold.

In practical applications of secret sharing schemes to an audio secret and to address the limitations of HAS decryption, there are instances where decryption is required to be performed on a computer. The scheme in [4] achieved decryption computationally, but is limited to binary audio and does not extend to the general (k, n) threshold scheme. Moreover, the security of this scheme is not proven from an information theoretical point of view. In summary, each one of previous schemes has at least one of these limitations: 1) It does not extend to (k, n) threshold scheme, 2) Information theoretical security is not proven and 3) It has the limitations of HAS decryption.

In this paper we propose a method to protect audio secrets using Shamir's secret sharing (SSS) scheme to address the above limitations. To the best of our knowledge, this is the first ASS scheme based on SSS which is (k, n) threshold and information theoretically secure and it offers a computationally efficient decryption. SSS in general does not have the above limitations described in points 1 and 2. Because of the proven security properties of the SSS scheme, many researchers have applied it to protect secret text, images, video, digital signatures and encryption/decryption keys [10]. Another work [11] uses SSS to protect an image and PDF secret by creating shares and applying steganography to hide each share in an MP3 cover. Such an approach is different from our method since we are protecting an audio secret. Table 1 compares the limitations of previous techniques and highlights that the proposed scheme does not have such limitations.

The rest of this paper is organized as follows. In Section 2, we discuss Shamir's secret sharing scheme. The proposed method for managing audio secrets is detailed in Section 3 and Section 4 discusses the experimental results. We conclude the paper in Section 5.

Table 1. A comparison of the proposed scheme with previous schemes

Scheme	Threshold	Information theoretically secure	Decryption
Desmedt et al. in 1998 [3]	$(2, n)$	Yes	Human auditory system
Lin et al. in 2003 [2]	$(2, n)$	Yes	Human auditory system
Nishimura et al. in 2005 [4]	(n, n)	Not proven	Computer
Ehdaie et al. in 2008 [5]	(k, n)	Not proven	Human auditory system
Yoshida and Watanabe in 2012 [6]	(n, n)	Yes	Human auditory system
Washio and Watanabe in 2014 [7]	(n, n)	Yes	Human auditory system
Proposed scheme	(k, n)	Yes	Computer

2 SSS Scheme

Shamir introduced his scheme in 1979 [1], which is based on polynomial interpolation. The goal of this scheme is to divide data into n shares such that:

1. Any k or more shares can reconstruct the secret.
2. $k-1$ or fewer shares cannot reconstruct the secret.

Such a scheme is called a (k, n) threshold scheme where $2 \leq k \leq n$, n is the number of shares and k is the least number of shares required to reconstruct the secret.

To share a secret S among n participants, a polynomial function $f(x)$ is constructed of degree $k-1$ using k random coefficients $a_1, a_2 \ldots a_{k-1}$ in a finite field $GF(q)$ where a_0 is S, and q is prime number $> a_0$.

$$f(x) = (a_0 + a_1 x + \ldots + a_{k-1} x^{k-1}) mod\ q \tag{1}$$

Any k out of n shares can reconstruct the secret using Lagrange interpolation to reconstruct the polynomial $f(x)$; the secret can be obtained at $f(0)$ i.e. $f(0) = a_0 = S$

$$f(x) = \sum_{j=1}^{k} \left(y_j \prod_{i=1, i \neq j}^{k} \left(\frac{x - x_i}{x_j - x_i} \right) \right) mod\ q \tag{2}$$

3 Proposed Method

As described in Section 2, we apply the SSS scheme to create an audio secret sharing method as depicted in Fig. 1. In our method we create shares of amplitude samples since they contain information of an audio signal. The following section details share generation and reconstruction of the secret audio. Notations of variables, symbols and functions used throughout this section are summarized in Table 2.

Table 2. Notation of variables

Variable	Description
A	Original secret audio signal
A'	Preprocessed secret audio signal
a_o	Original secret audio sample ($a_0 \in A$)
a'_o	Preprocessed secret audio sample ($a'_0 \in A'$)
ϵ	Round-off error
d	Rounding precision
γ	DC shift of signal to first quadrant
$Pr(.)$	Probability function
$GF(.)$	Finite field
q	First prime number greater than the maximum original secret audio sample (a_o)
q'	First prime number greater than the maximum preprocessed secret audio sample (a'_o)
b	Number of bits to represent q
b'	Number of bits to represent q'

3.1 Preprocessing and Share Generation

Using the SSS (k, n) threshold, we generate n shares such that at least k shares can reconstruct the secret. Using real numbers in a cryptosystem means excluding the modular prime operation which in the case of SSS degrades security. Therefore, we have to preprocess amplitude samples of the secret audio from real to positive integer values. During preprocessing, we first round-off the real amplitude samples by multiplying by 10^d where d is some integer value. Roundoff error is bounded by:

$$-\frac{1}{2} \times 10^{1-d} \leq \epsilon \leq \frac{1}{2} \times 10^{1-d} \tag{3}$$

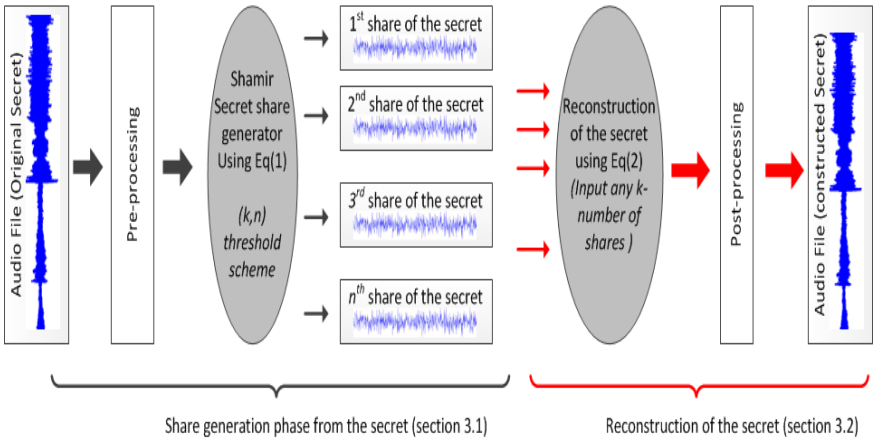


Fig. 1. Audio secret sharing framework

where ϵ is the rounding error and d is the rounding precision. Each amplitude secret a_0 is converted to an integer and shifted to the first quadrant by a threshold γ to obtain positive sample values within $\mathbb{Z}_p$. Shifting the signal to first quadrant does not distort the waveform as illustrated in Fig. 2.

$$a'_0 = \left((a_0 + \epsilon) \times 10^d\right) + \gamma \tag{4}$$

Using Equation (1) from Section 2, n shares are created and distributed to n participants. The algorithm is shown below.

Algorithm 1: Share Generation
Input: Secret audio $A = \{A_1, A_2 \ldots A_m\}$; where A_m is the amplitude at the m^{th} time interval
Output: Secret Shares $S_1, S_2 \ldots S_n$
Description:

1. Read wav file i.e. $[A, fs]$ = wavread('*wavfile*')
2. $A = \text{round}((A + \epsilon) \times 10^d)$
3. $A' = A$+absolute of the minimum value of A
4. Compute the first prime number q' greater the than maximum value of A'
5. **for** $i = 1$ to length of A' **do**
   amplitude value at the i^{th} time interval is the secret i.e. $a'_0 = A'_i$ and randomly choose coefficients $a_1, a_2 \ldots a_{k-1}$ from a set a positive integer field $\mathbb{Z}_p$
6. **for** $j = 1$ to n; number of shares to create **do**
   Compute $share(i, j)$ from the polynomial obtained in 5. $share(i, j)$ is the j^{th} share for the i^{th} amplitude value
7. **end for**

8. **end for**
9. **for** $j = 1$ to n **do**
10. S_j = combine all amplitude share values for each share index
11. **end for**
12. **return** $S_1, S_2 \ldots S_n$;

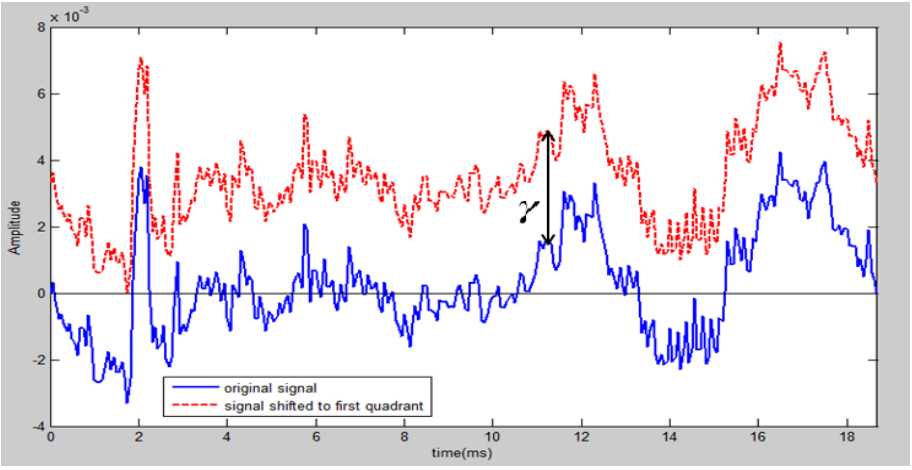


Fig. 2. Shifting signal to first quadrant

3.2 Secret Reconstruction and Post-processing

To reconstruct the secret audio we need at least k out of the n shares. Referring to Fig. 1, there are two blocks at the secret reconstruction phase: 1) reconstruct the secret by using Equation (2) to solve the polynomial function in Equation (1) and obtain the secret sample at evaluation point $x = 0$, (this is done for all samples) and 2) post-process to reverse engineer the preprocessing done during share generation. We first subtract the signal shift threshold from the obtained signal in step 1 and then divide by 10^d to get the secret audio signal. The algorithm is shown below.

Algorithm 2: Secret Reconstruction
Input: Any $k \leq n$ audio shares $S_1, S_2 \ldots S_k$
Output: Secret Audio $A = \{A_1, A_2 \ldots A_m\}$
Description:

1. Reconstruct the polynomial $f(x)$ from shares $S_1, S_2 \ldots S_k$ using Lagrange interpolation in (2) in a finite field $GF(q')$

2. **for** $i = 1$ to length of share **do**
   Obtain a'_0 coefficient at evaluation point $f(0)$ i.e. a'_0 is the reconstructed amplitude secret at the i^{th} time interval
   $A'(i) = a'_0$
3. **end for**
4. $A = (A'$ - absolute of the minimum value of A from Algorithm 1, step 2)$/10^d$
5. **return** A;

3.3 Security Analysis

The proposed method is based on the SSS (k, n) threshold scheme which is proven to be information theoretically secure [8]. SSS has perfect secrecy when applied to independent input sequences, however our scheme preprocesses the audio signal before generating shares so it is imperative to examine the impact on information theoretically security.

Theorem 1. *Information theoretical security of SSS is preserved if the probability of revealing an audio secret sample a_0 shared under $GF(q)$ is the same as the probability of determining $a'_0 = (a_0 \times 10^d) + \alpha$ shared under $GF(q')$ (where $\alpha = \left(\epsilon \times 10^d\right) + \gamma$ from Equation (4) and q' is a prime number greater than $(q \times 10^d) + \alpha$)*

Proof. For each plaintext of audio secret $a_0 \in A$ there is an equal probability that it can be any value from the set $0 \leq a_0 \leq q - 1$ of q values since SSS encryption is upper bounded by q. This probability is given by:

$$Pr(a_0)_{0 \leq a_0 \leq q-1} = \frac{1}{q} \tag{5}$$

Similarly, for each plaintext a'_0 of the preprocessed audio secret A' where $a'_0 = (a_0 \times 10^d) + \alpha$ there is also an equal probability of being any value from the set $0 \leq a'_0 \leq q' - 1$ of q values with probability given as:

$$Pr(a'_0)_{0 \leq a'_0 \leq q'-1} = \frac{1}{q} \tag{6}$$

The probability of revealing the secret a_0 and a'_0 in the above cases is the same $\frac{1}{q}$. Thus, our scheme preserves information theoretical security after preprocessing the original audio secret. An adversary in both cases will have to guess the secret with a probability of $\frac{1}{q}$.

3.4 Data Overhead

Our proposed scheme introduces some data overhead due to the preprocessing step. This data overhead is the number of bits used to represent the maximum preprocessed audio sample. Since the generation of shares under a finite field $GF(q')$ is upper bounded by q' (where q' is the first prime number greater than

$maximum[(a_0 + \epsilon) \times 10^d + \gamma])$ we can conclude that the data overhead is also upper bounded by the number of bits used to represent q'. If b is the number of bits to represent this value then:

$$b = log_2(q') \tag{7}$$

Due to the dynamic range of audio signals, q' will always vary for different audio signals depending on the quantization level ($8bit, 16bit$ etc.) of the ADC converter used during quantization. From Equation (3), it can be seen that increasing d during preprocessing will yield minimal round-off error but higher data overhead so d should be chosen to maintain a balance between the two.

4 Experimental Results

Table 3 details the 6 audio files obtained from [9] that we use to test the proposed audio secret sharing method. In the (k, n) threshold scheme, we set $k = 2$ and $n = 3$ implying that 2 out of 3 created secret shares are required to reconstruct the secret audio.

Table 3. Data set

Test file (.wav)	length(secs)	Bits/sample	sampling frequency (Hz)
audio1	2	16	16000
audio2	43	16	8000
audio3	8	8	22050
audio4	14	8	44100
audio5	4	8	8000
audio6	2	32	8000

We implemented the audio secret sharing method using MATLAB14 on a 2.53GHz i5 CPU with 4GB RAM. Table 4 details the processing time for creating secret shares and reconstructing the original audio secret. The time information in the table suggests that the complexity of reconstructing the secret is relatively lower than creating secret shares. Since the proposed method is applied at an audio sample level, the processing time is directly proportional to the audio bit rate, which is associated with the sampling frequency and number of bits per sample.

Audio signals by nature have correlating adjacent samples and the use of random coefficients as a blinding factor in Equation (1) to generate shares eliminates this correlation. Thus, individual shares do not reveal information about the secret audio. Time domain plots of one of our test audio files (audio1) in Fig. 4 illustrates: 1) the difference between the audio secret and its noisy shares and 2) the similarity between the reconstructed and original secret audio. Fig. 3 shows the similarity scores between original secret audio, and $1st$ share, $2nd$ share,

Table 4. Average processing time to create shares and reconstruct the secret

Test file	length(secs)	Share creation (ms)	Secret reconstruction (ms)
audio1	2	152	7
audio2	43	1614	50
audio3	8	929	29
audio4	14	2770	80
audio5	4	150	12
audio6	2	83	5

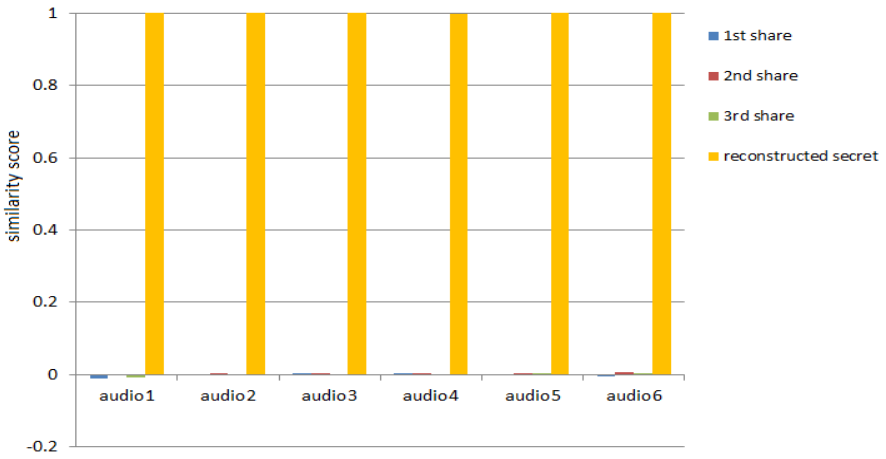


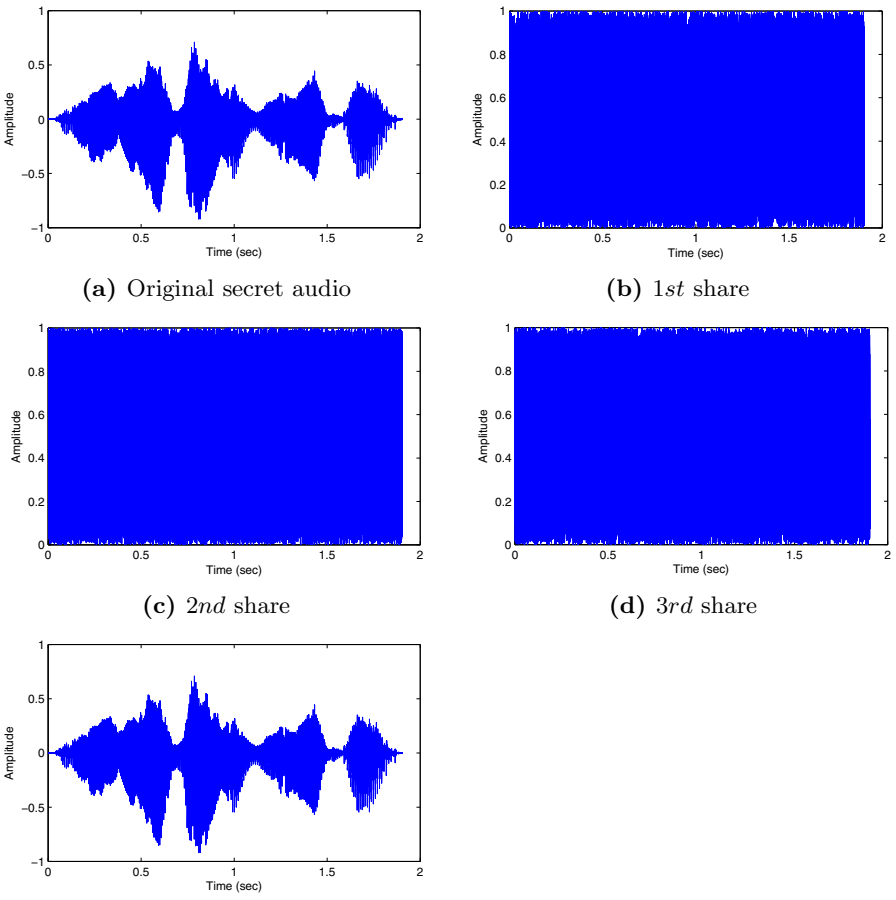
Fig. 3. Similarity Score

3*rd* share and reconstructed secret audio. The similarities were computed using pearson's correlation method. Results suggest less than 1% correlation between the original secret audio and its shares.

It is also evident that the reconstructed secret audio is about 100% correlated with the original secret audio; suggesting minimal information losses due to rounding error in the preprocessing step.

We also performed a listening study to evaluate perceptual security which was conducted online [3]. User scores are summarized in Table 5. 20 subjects in the age range of $20-40$ years participated in the survey. The similarity score is captured in a 4 point scale where the value 3 is given when two audio files are exactly the same content wise and the value 0 is given when two audio files are not similar at all content wise. As expected, all the participants agreed that both the share and audio secret are completely dissimilar in terms of content. However, about 92% of average similarity score was achieved for content similarity between the

[3] `https://az1.qualtrics.com/SE/?SID=SV_0AHmNAbzvekkweN&Preview=Survey&BrandID=qtrial2014`



(a) Original secret audio

(b) 1*st* share

(c) 2*nd* share

(d) 3*rd* share

(e) Reconstructed secret audio

Fig. 4. audio1, its shares and reconstructed secret

original audio secret and the reconstructed audio secret which confirms that our proposed scheme is perceptually secure. However, as depicted in Fig. 3, using pearson's correlation analysis, we were able to establish about 100% similarity score between the original audio secret and the reconstructed audio secret. In the future we would like to investigate the disparity of human judgment (Table 5) vs machine evaluation (Fig. 3) of similarity.

5 Conclusion

In this paper we propose an audio secret sharing technique using the Shamir's secret sharing (SSS) scheme. Compared to existing techniques, the proposed

Table 5. User study

	share	reconstructed secret
audio1	0	2.75
audio2	0	2.67
audio3	0.08	2.75
audio4	0.08	2.67
audio5	0	2.92
audio6	0	2.83

technique is (k, n) threshold, information theoretically secure and computationally efficient decryption which does not rely on Human Auditory System (HAS). Our security analysis and experimental results also show that our scheme is information theoretically secure, perceptually secure and computationally efficient. Short time framing can be combined with the proposed scheme to share an audio secret of several hours long. In the future, we will experiment the capabilities of the proposed scheme for modeling network architecture to encrypt voice signals over multiple communication channels to address security breaches in voice over IP and wiretapping.

Acknowledgement. This research was supported in parts by the Natural Sciences and Engineering Research Council (NSERC) of Canada, Grant No. 371714 and University at Albany - State University of New York Grant No. 640075.

References

1. Shamir, A.: How to share a secret. Communications of the ACM 22, 612–613 (1979)
2. Lin, C.C., Laih, C.S., Yang, C.N.: New Audio Secret Sharing Schemes With Time Division Technique. Journal of Information Science and Engineering 19, 605–614 (2003)
3. Desmedt, Y.G., Hou, S., Quisquater, J.-J.: Audio and optical cryptography. In: Ohta, K., Pei, D. (eds.) ASIACRYPT 1998. LNCS, vol. 1514, pp. 392–404. Springer, Heidelberg (1998)
4. Nishimura, R., Fujita, N., Suzuki, Y.: Audio Secret Sharing for 1-Bit Audio. In: Khosla, R., Howlett, R.J., Jain, L.C. (eds.) KES 2005. LNCS (LNAI), vol. 3682, pp. 1152–1158. Springer, Heidelberg (2005)
5. Ehdaie, M., Eghlidos, T., Aref, M.R.: A novel secret sharing scheme from audio perspective. In: International Symposium on Telecommunications, pp. 13–18. IEEE, Tehran (2008)
6. Yoshida, K., Watanabe, Y.: Security of audio secret sharing scheme encrypting audio secrets. In: International Conference for Internet Technology and Secured Transactions, pp. 294–295. IEEE, London (2012)
7. Washio, S., Watanabe, Y.: Security of audio secret sharing scheme encrypting audio secrets with bounded shares. In: International Conference on Acoustics, Speech and Signal Processing (ICASSP 2014), pp. 7396–7400. IEEE, Italy (2014)
8. Stallings, W.: Cryptography and Network Security Principles and Practice, 5th edn. Prentice-Hall (2010)

9. SVV media audio database,
http://download.wavetlan.com/SVV/Media/HTTP/http-wav.htm
10. Atrey Pradeep, K., Alharthi, S., Hossain, M.A., AlGhamdi, A., El-Saadik, A.: Collective control over sensitive video data using secret sharing. Multimedia Tools and Applications (2013), doi:10.1007/s11042-013-1644-0
11. Chan, K.F.P.: Secret Sharing in Audio Steganography. In: Information Security South Africa, ISSA. South Africa (2011)