Multimed Tools Appl (2018) 77:25629–25657
https://doi.org/10.1007/s11042-018-5810-2

A novel approach for verifiable (n, n) audio secret sharing scheme

Shambhu Shankar Bharti[1] · **Manish Gupta**[1] · **Suneeta Agarwal**[1]

Received: 18 July 2017 / Revised: 3 January 2018 / Accepted: 15 February 2018 /
Published online: 28 February 2018

Abstract Audio is a natural way of communication among persons. Nowadays, many cases of Phone-tapping, hacking of E-mail having some audio files have been reported. Hence, for sending a secret message in audio file over the internet, security and confidentiality must be assured. A secret message is commonly transmitted in encrypted form to assure confidentiality/security using a single key. This does not serve the purpose completely as knowing the key, the whole secret can be revealed. To enhance the security of the secret, sending it through multiple units is preferred. Here, a novel approach is proposed for the same. In the proposed approach an audio secret is divided into n audio shares such that information contained in any proper subset of share/shares (unit) is insignificant. With all the shares only secret can be revealed after performing some computation. Authenticity of the revealed secret can also be confirmed (if necessary) by checking the integrity of individual shares received. If any share is lost during transmission, the proposed scheme has the facility to determine the share number of the lost one, so that request may be sent for resending the same. The proposed scheme is suitable for real-time audio communication as the construction of shares is based upon the available bandwidth. Also, it does not need any cover audio to transmit the share. Experimentally, Mean Opinion Score (MOS) and correlation coefficient ($r(S, S_r)$) between input secret and revealed secret are found 5 and 0.99 respectively when there is no attack during the transmission. In case the total (either in one or more shares) attack is less than 50% of the size of a single share, MOS of the revealed secret lies between 3.8 to 4.

✉ Shambhu Shankar Bharti
shambhu4u08@gmail.com

Manish Gupta
manishymca2007@gmail.com

Suneeta Agarwal
suneeta@mnnit.ac.in

[1] National Institute of Technology Allahabad, Allahabad, India

Keywords Secret share · Audio security · Internet telephony · Secret audio storage · Secret audio communication

1 Introduction

Audio is one of the major component of the multimedia being the most natural way of communication. Persons communicate with call centers to resolve their issues, who may record the conversations. These conversations may also contain important personal information like name, mobile number, account number, credit card number, Aadhaar number etc. which may later be illegally used. Similarly, if any prior information of a terror attack or crime is received by some intelligence agency then this information is to be shared secretly. Data received from sensors for various purposes such as activity recognition [7, 8, 10, 11], must also be securely transmitted. Several times social media data [9] stored on the server must also be secured as it contains personal information. Encryption is not sufficient to secure these sensitive informations. Hence, to ensure the security of these data, secret sharing scheme was introduced. Secret sharing scheme was first proposed by Shamir et al. [14] and Blakely et al. [2] in 1979. In the secret sharing scheme, secret information is divided into multiple units called shares before storing or transmitting it. In a (k, n) secret sharing scheme, secret is divided into n shares and at least k shares are required to regenerate the secret. Secret sharing can be categorized into two categories with respect to the computation required for revealing the secret. In the first category, some computation is necessary while in the second no computation is required. In the second category, only with the help of human perception system like ear, eye, etc., the secret information can be revealed. For example, in visual cryptographic system human can reveal the secret without any computation just by superimposing the visual shares in an order. Similarly, a human can reveal the audio secret by playing the audio shares simultaneously. Revealing audio secret without computation is affected by hearing impairment, location of audio players, effect of noise during transmission etc. Thus, first category (revealing secret with some computation) is a better option in case of audio secret sharing.

Security of audio data is a major concern among researchers working in the field of multimedia security. Number of researchers have proposed different approaches to secure audio data but still it is an open research problem. To secure audio data, here we are proposing verifiable (n,n) audio secret sharing scheme by separating the amplitudes and signs of audio secret. Integrity of each share received can also be checked at the receiver end. This scheme is also suitable for real-time audio communication as the available bandwidth is also taken into consideration for the construction of shares. Total size of all the shares together is same as the size of secret itself as it does not require any cover audio to transmit the shares. During transmission of the shares, if some attacks are made even then secret may be revealed. Here, two types of attacks are considered.

- Addition of noise/meaningful audio may happen in some part of the share/s.
- Replacement of some part of the share/s by noise/meaningful audio.

The remaining part of the paper is structured as follows:

Literature Survey for audio secret sharing is presented in Section 2. Proposed approach for verifiable (n, n) ASS scheme is described in Section 3. Effectiveness of the proposed approach is shown through experimental results in Section 4. Comparison of proposed approach with state of the art approaches is discussed in Section 5. Paper is concluded in Section 6 followed by references.

2 Literature survey for Audio Secret Sharing (ASS)

Number of researchers have been working in the field of ASS scheme which is a more secure way to transmit/store secret data through audio. In this section, a brief review of state of the art approaches for ASS scheme is presented.

Y. Desmedt et al. [3] proposed a (2, 2) audio secret sharing scheme to hide a binary secret message through cover audio. Two audio shares each of same duration as of original cover audio are generated using interference property of audio. To hide a bit of binary secret, part of the cover audio of fixed duration (assume T) is used. For example, if secret binary bit is '1' then phase difference between both shares for time 'T' are taken to be 0 otherwise Π. Playing both shares simultaneously, binary secret will be revealed as follows: Higher amplitude (constructive interference) of audio is decoded as '1' otherwise (destructive interference) '0'. This scheme is good for binary secret but it cannot be used for audio secret. Here, total size of all shares is very large as compared to secret size. Also, it may not be used to hide secret in real time audio transmission. This scheme [3] is further extended to (2,n) audio secret sharing scheme which requires $\lceil \log_2 n \rceil$ number of cover audios. Ching-Nung Yang [18] and Chen-chi-Lin et al. [6] also proposed (2,n) ASS scheme with only one cover audio but other limitations still exist continue. Above mentioned schemes may not be generalized for (k, n) ASS scheme. Later, Md. Ehdaie et al. [4] also proposed $(2, n)$ ASS scheme that can be generalized for (k, n) ASS. In this scheme, secret message is an audio file. They generate shares from the secret itself. First share is generated randomly and remaining shares depend on it. Here also, secret can be revealed by playing audio shares simultaneously. Total size of shares is larger as compared to secret size but lesser than the above mentioned approaches. This scheme is also not applicable in real time audio transmission. Md Ehadaie et al. [12] introduced a new generalization method to extend every (2, n) secret sharing scheme. Daniel Socek et al. [15] proposed a scheme for sharing audio secret by using Morse code which has two types of sounds: short beep and long beep. Here, audio secret is first converted into Prefix Binary code signal then, it is transformed into Morse-code structures. In this approach there is no computation required to reveal the secret.

Jing-Zhang et al. [17] proposed an audio secret sharing scheme based on fractal encoding to reduce the size of the original secret by 40%. Fractal is divided into n parts which are embedded into n distinct audio covers using Least Significant Bit (LSB) technique in the frequency domain. Hence, total size of the shares is larger than the original secret. Computation time is high so, it can not be applicable in real time audio communication. S. Vyavahare et al. [16] proposed an audio secret sharing scheme using matrix projection technique which does not require any cover audio. Shares are generated from the secret itself, and total size of shares generated is approximately same as secret size. During the construction of shares a remainder matrix is generated which is later used for revealing the secret. If any change occurs in the remainder matrix then secret cannot be revealed. It is the major drawback of this approach. Ryouichi Nishimurce et al. [13] proposed audio secret sharing scheme for 1-bit audio. 1-bit audio is a high quality digital audio with sampling frequency of 2,822.4 kHz and a resolution of 1-bit. Here, share is constructed using sharing table. This scheme cannot be generalized for other formats of audio.

Apart from the above mentioned limitations no authors has provided a method for checking the integrity of received shares if any attack occur during the transmission of shares.

In this paper, authenticity of revealed secret can be guaranteed by checking the integrity of each shares received. Other limitations found in the literature are reduced in the proposed scheme of this paper.

3 Proposed approach for Verifiable (n, n) Audio Secret Sharing (*VASS*) scheme

Flow chart of the proposed approach for generation of verifiable shares and revealing of secret is shown in Fig. 1. Details of construction of verifiable shares and revealing of secret are shown in Fig. 2 and in Fig. 7 respectively. There are three phases in the proposed approach to generate verifiable audio shares. First phase converts input audio secret into two different streams: Stream of scaled amplitudes (S_p) and Stream of signs (S_s), using Algorithm 1. Second phase generates n primary shares. These primary shares are generated by changing the location of scaled amplitudes and relative ordering of signs using Algorithm 2. Primary shares when played give the impression of noise. In the primary shares, share number and scaling factor are also inserted which are required to reveal the secret later. In third phase, n authentication keys are generated and embedded them into respective primary shares which will be used to check the integrity of shares later. The shares thus generated are verifiable. Once, all the shares are received then by performing some computation, secret is revealed through Algorithm 4. At the receiver end, integrity of share/s can be verified (if required) by using Algorithm 5. Time complexity of the proposed approach in this paper is $O(N)$, where N is the total number of samples in audio secret. Experimentally, it is also found that time required to generate the shares and revealing the secret is low which is also clear from the Table 6. Hence, to increase the security of audio conversation in real time over Internet this approach may be used.

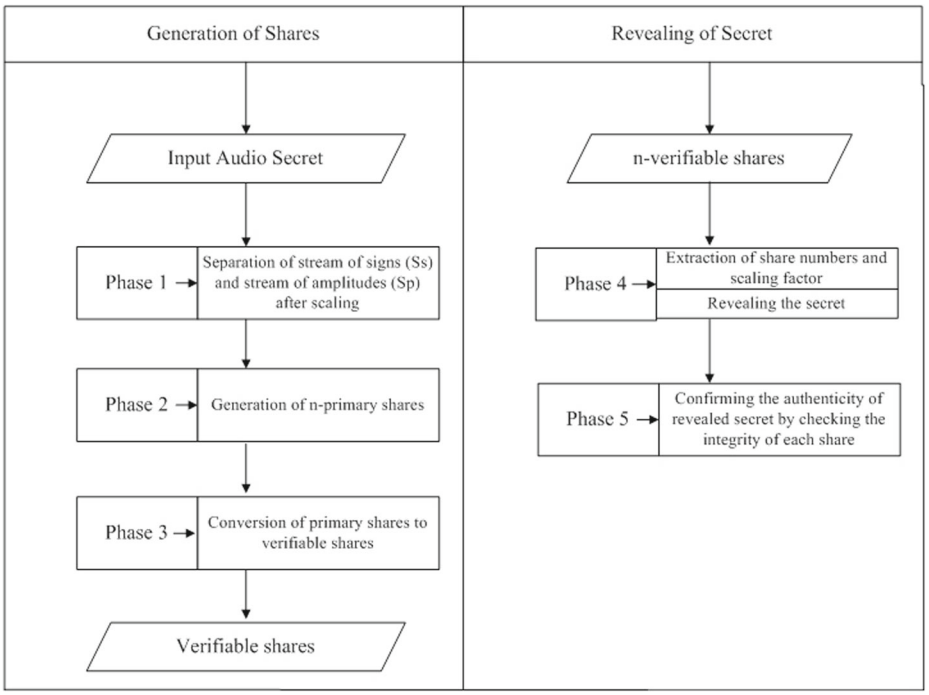


Fig. 1 Flow chart of proposed approach for verifiable (n, n) ASS scheme

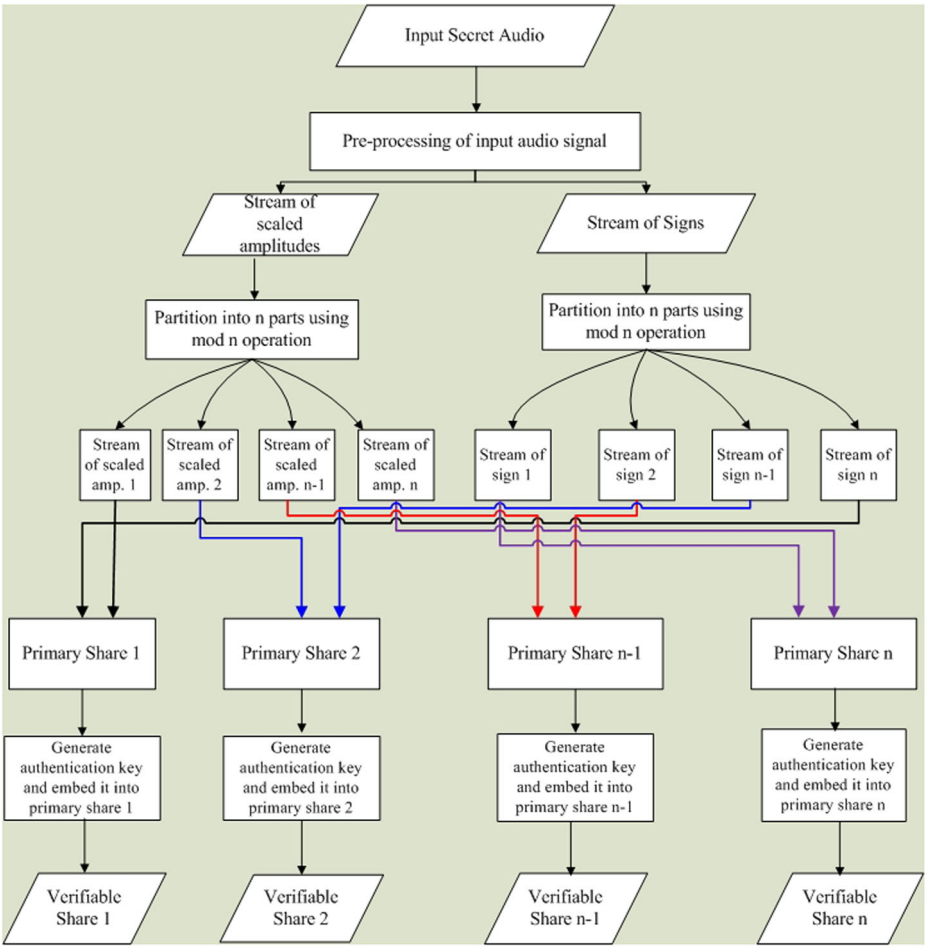


Fig. 2 Flow chart of proposed approach for share generation

As the quality of audio signal decreases with changes in relative ordering of sign, to provide more security in audio shares they are generated by separately processing on stream of signs (S_s) and stream of amplitudes (S_p) of original audio secret (S). Due to it, no cover audio is required for transmitting/storing the shares. Even by playing all the shares no secret can be revealed. Secret can only be revealed after applying the Algorithm 5.

3.1 Phase 1 (pre-processing of audio secret): separation of stream of signs and stream of scaled amplitudes

First of all scaling of input audio secret (S) is done by dividing it with scaling factor (sf). Scaling is done to reduce the transmission time of the shares. Scaling factor (sf) is taken as the ratio of largest amplitude of S and L (Default value $= 255$, it can be changed depending upon the largest allowed amplitude for transmission). Due to scaling of S, musical noise

may be introduced. To minimize it, numbers with zero amplitudes in the scaled audio secret are replaced by ones. Quality of audio secret is inversely proportional to the scaling factor. Hence, upper limit of the scaling factor also depends upon sampling frequency of the audio secret.

Algorithm 1 Preprocessing of audio secret

INPUT: S : Secret audio signal of size $N \times 1$, fs : sampling frequency used in input audio secret, A_m : maximum amplitude which is allowed to be transmitted.
OUTPUT: S_p : Stream of scaled amplitude of size $N \times 1$, S_s : Stream of sign of size $N \times 1$.
Ensure:
(1) $\lceil . \rceil$ is ceil function.
(2) $|S|$, represents the magnitude of audio signal S.
(3) $max(.)$ represents a function that returns maximum value of input vector.
(4) sf is scaling factor.

$S_{mag} = |S|$
$m = max(S_{mag})$ ▷ Find the maximum amplitude of the input secret audio.
$sf = \lceil (m/A_m) \rceil$ ▷ Evaluate scaling factor.
for $i \leftarrow 1$ to N **do**
    $S(i) = \lceil (S(i)/sf) \rceil$ ▷ Scaling input audio secret
    if $S(i) = 0$ **then**
        $S(i) = 1$
    end if
    if $S(i) > 0$ **then** ▷ S_s stores the sign bit of audio sample.
        $S_s(i) = 1$ ▷ 1 is used for positive sign.
        $S_p(i) = S(i)$
    else
        $S_s(i) = -1$ ▷ -1 is used for negative sign.
        $S_p(i) = S(i) \times -1$
    end if
end for
return$(Sp[], Ss[], sf)$

Here, 1 and -1 are used to represent positive and negative signs respectively. After scaling, audio secret is separated into two streams: stream of scaled amplitudes (S_p) and stream of signs (S_s) as shown in Fig. 3.

Primary shares of audio secret are generated in phase 2.

3.2 Phase 2: generation of primary shares

Stream of amplitudes: S_p and stream of signs: S_s, each one is divided into $n (n \geq 2)$ vectors of size $\lceil \frac{N}{n} \rceil \times 1$ using modulus n operator applied on indices. If $(\frac{N}{n})$ is not an integer then S_p and S_s both are appended by required number of ones. To generate n primary shares, n vectors $Avec_1$, $Avec_2$, ... ,$Avec_n$ are generated from S_p and n vectors $Svec_1$, $Svec_2$, ...,$Svec_n$ are generated from S_s. Value at i^{th} index of S_p and (S_s) are copied at $(\frac{i-1}{n} + 1)^{th}$ index of $Avec_t$ and ($Svec_t$) vectors respectively, where $t = ((i - 1) \mod n + 1)$.

Algorithm 2 Algorithm for primary share generation

INPUT: S_p :Stream of scaled amplitudes of size $N \times 1$, S_s : Stream of sign of size $N \times 1$, n : number of shares to be generated, sf : scaling factor.

OUTPUT: $Sh_{pr}^{(1\ldots n)}$: n Primary shares each of size $\left(\frac{N}{n}+2\right) \times 1$.

Ensure:

(1) $data$: is a matrix of size $\left(\frac{N}{n}\right) \times n$

(2) $count$ is used as temporary variables.

(3) $data_f$ is a matrix of size $\left(\frac{N}{n}+2\right) \times n$.

(4) Sh_{pr}^{k} represents k^{th} primary share of size $\left(\frac{N}{n}+2\right) \times 1$

1: count=1;
2: **for** $i \leftarrow 1$ to $\frac{N}{n}$ **do**
3: **for** $j \leftarrow 1$ to n **do**
4: data$[i][j] = S_p(count)$ ▷ Divide the stream of amplitude of secret audio into n equal vectors using mod n operator applied on indices.
5: $sign[i][j] = S_s(count)$ ▷ Divide the stream of sign of secret audio into n equal sign vectors using mod n operator applied on indices.
6: $count = count + 1$
7: **end for**
8: **end for**
9: **for** $j \leftarrow 1$ to n **do**
10: **for** $i \leftarrow 1$ to $\frac{N}{n}$ **do**
11: $data_f[i][j] = data[i][j] \times sign\left[\frac{N}{n} - i + 1\right][n - j + 1]$ ▷ Multiply sign-bit of n^{th} column in reverse order to the first column of data and sign bit of $(n-1)^{th}$ column in reverse order to the second column of data and so on to generate primary share.
12: **end for**
13: $data_f\left[\frac{N}{n}+1\right][j] = sf + j$ ▷ Inserting scaling factor and sequence number of share.
14: $data_f\left[\frac{N}{n}+2\right][j] = sf - j$
15: **end for**
16: **for** $j \leftarrow 1$ to n **do**
17: $sh_{pr}[j] = data_f[][j]$
18: **end for**
19: **for** $j \leftarrow 1$ to n **do**
20: **Distribute / send** $sh_{pr}(j)$ to j^{th} *participants/recipents*
21: **end for**

For example, Let n = 4 shares be constructed for the input audio secret S. S:

Index	1	2	3	4	5	6	7	8	9	10	11	12
Data	128	−150	−255	−185	158	140	−80	106	208	190	−170	−35

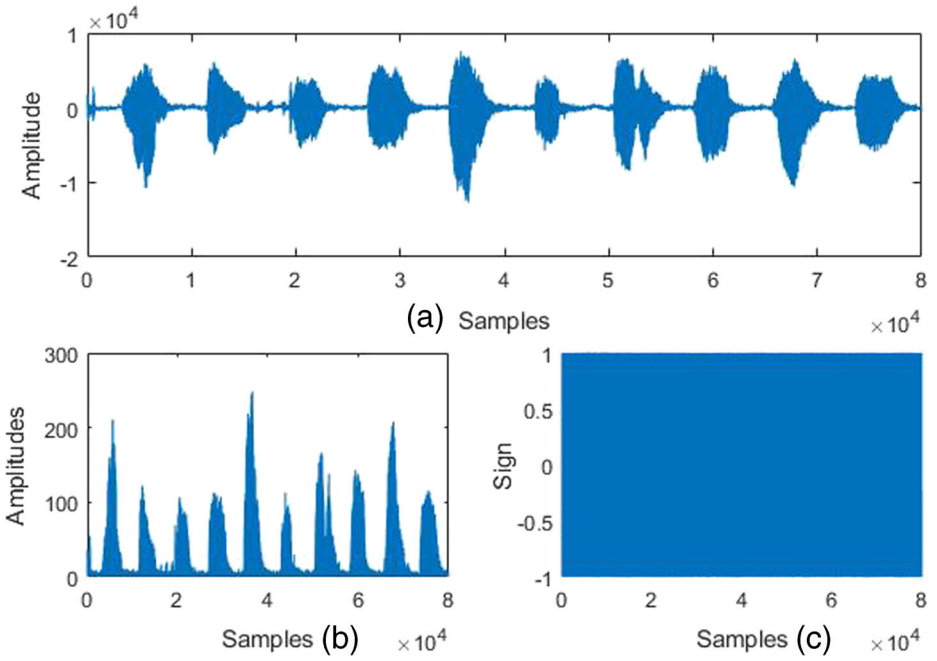


Fig. 3 **a** S: Input audio secret **b** S_p: Stream of scaled amplitude of input audio secret and **c** S_s: Stream of signs

Here, maximum amplitude of S is 255 at index 3. If maximum allowed amplitude that can be transmitted is 26 then, $sf = \lceil \frac{255}{26} \rceil = 10$. Now, stream of scaled amplitudes (S_p) are as follows:

Index	1	2	3	4	5	6	7	8	9	10	11	12
Data	13	15	25	18	16	14	8	11	21	19	17	3

Similarly, stream of signs S_s is:

Index	1	2	3	4	5	6	7	8	9	10	11	12
Data	1	−1	−1	−1	1	1	−1	1	1	1	−1	−1

To generate 4 primary shares, following four vectors $Avec_1$, $Avec_2$, $Avec_3$ and $Avec_4$ are generated from S_p using mod 4 as explained in Algorithm 2.

$Avec_1$	$Avec_2$	$Avec_3$	$Avec_4$
13	15	25	18
16	14	8	11
21	19	17	3

Similarly, following four vectors $Svec_1$, $Svec_2$, $Svec_3$ and $Svec_4$ are generated from S_s using *mod* 4 as explained in Algorithm 2.

$Svec_1$	$Svec_2$	$Svec_3$	$Svec_4$
1	−1	−1	−1
1	1	−1	1
1	1	−1	−1

For the generation of i^{th} primary share, vec_i is generated first using $Avec_i$ and $Svec_{n-i+1}$ as follows:

$$vec_i(j) = Avec_i(j) \times Svec_{n-i+1}\left(\frac{N}{n} - j + 1\right)$$

vec_1	vec_2	vec_3	vec_4
−13	−15	25	18
16	−14	8	11
−21	−19	−17	3

Next, share number and scaling factor used are appended in shares: $sf + i$ is appended in the $(\frac{N}{n} + 1)^{th}$ index and $sf - i$ is appended in the $(\frac{N}{n} + 2)^{th}$ index of the i^{th} vec_i to generate i^{th} primary share ($sh_{pr}(i)$). These extra values appended in a share will be used to determine share number and scaling factor during revealing of the secret. Following are the four primary shares thus generated by Algorithm 2 for $sf = 10$.

$sh_{pr}1$	$sh_{pr}2$	$sh_{pr}3$	$sh_{pr}4$
−13	−15	25	18
16	−14	8	11
−21	−19	−17	3
11	12	13	14
9	8	7	6

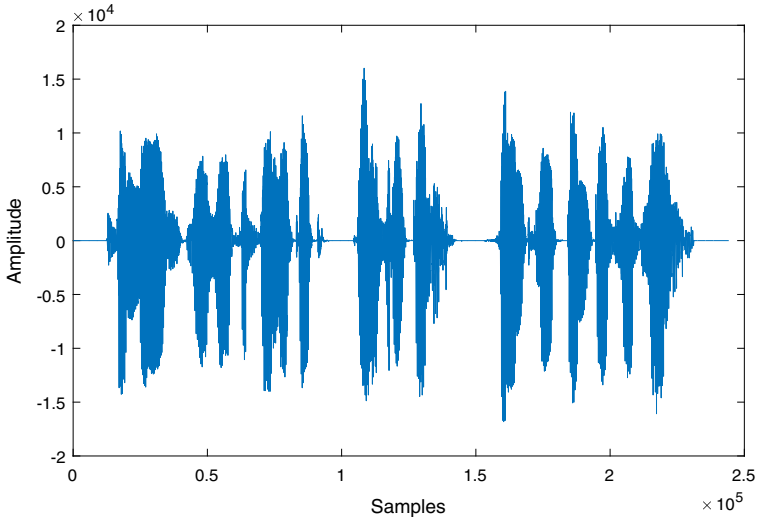
For primary share1: $sf(10) + i(1) = 11$ and $sf(10) - i(1) = 9$ have been appended at the 4^{th} and 5^{th} locations respectively. Similarly, for primary share2: $sf(10) + i(2) = 12$ and $sf(10) - i(2) = 8$, for primary share3: $sf(10) + i(3) = 13$ and $sf(10) - i(3) = 7$, and for primary share4: $sf(10) + i(4) = 14$ and $sf(10) - i(4) = 6$ have been appended at the 4^{th} and 5^{th} locations respectively.

Figure 4 shows the four primary shares generated from the secret S.

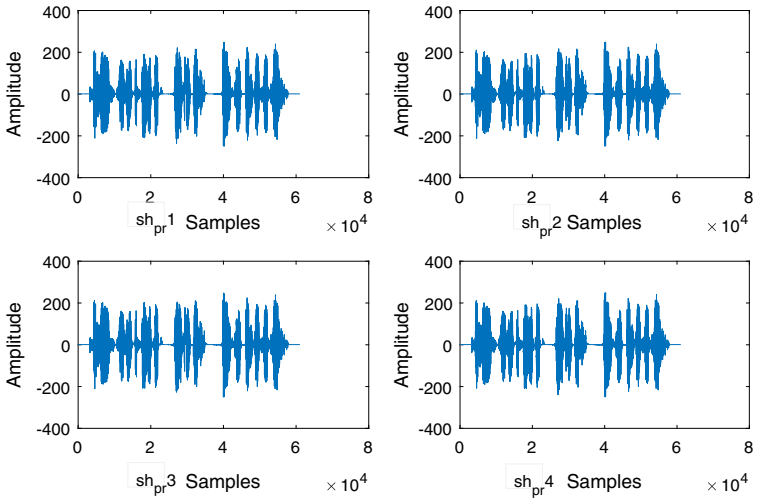
Authenticity of the secret revealed from primary shares can not be confirmed. So, our next step would be to make the shares verifiable to ensure the authenticity of secret revealed.

3.3 Phase 3: process to make primary shares verifiable

To make primary shares (obtained by Algorithm 2) verifiable, first an authentication key for each share is generated and thereafter embedded there itself. Authentication



(a) S: Input audio secret (Male spoken in English language)



(b) Primary shares

Fig. 4 Generated primary shares from S for n = 4

key is generated by performing repetitive bitwise XOR operation among the first $\lceil \frac{N}{n} \rceil$ elements of the primary share. To make the primary share verifiable, maximum amplitude of a primary share is replaced by the value obtained through bitwise xoring of respective authentication key and maximum amplitude of that

share. Let, ak_i denote the authentication key of i^{th} primary share (generated in Section 3.2).

$$\begin{aligned} \text{ak}_1 &= -13 \oplus 16 \oplus -21 = 8 \\ \text{ak}_2 &= -15 \oplus -14 \oplus -19 = -18 \\ \text{ak}_3 &= 25 \oplus 8 \oplus -17 = -2 \\ \text{ak}_4 &= 18 \oplus 11 \oplus 3 = 26 \end{aligned}$$

Algorithm 3 Algorithm for conversion of primary shares into verifiable

INPUT: sh_{pr} :Primary share of size $\left(\frac{N}{n}+2\right) \times 1$, n : number of shares to be generated.

OUTPUT: $Sh_v(1...n)$: n Verifiable share each of size $\left(\frac{N}{n}+2\right) \times 1$.

Ensure:

(1) $count1$ and k are used as temporary variables.
(2) $share_i$ represents i^{th} share.
(3) $location(.)$: It is a function that returns the indices of a number in vector.
(4) $max(.)$: is a function that returns first maximum value of input vector.

1: $share_i \leftarrow \dfrac{sh\left(\frac{N}{n}+1\right) - sh\left(\frac{N}{n}+2\right)}{2}$
2: **for** sh_{pr} : 1 to n **do**
3: count1=0
4: **for** $i \leftarrow 1$ to $\frac{N}{n}$ **do**
5: $count1 = count1 \oplus sh_{pr}(i)$ ▷ Generation of authentication key
6: **end for**
7: $k = location(max(sh_{pr}))$
8: $sh_{pr}[k] = count1 \oplus sh_{pr}[k]$ ▷ Embedding of authentication key
9: $sh_v(share_i) = sh_{pr}$
10: **end for**

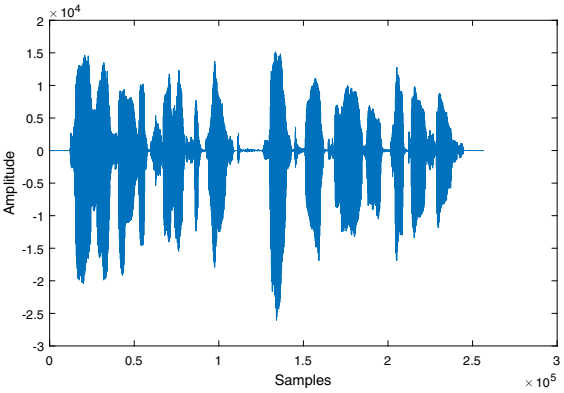
Highest amplitude value (-21) of primary share1 will be replaced by xoring it with ak_1 ($-21 \oplus 8 = -29$). Similarly, highest amplitude value (-19) of primary share2 will be replaced by xoring it with ak_2 ($-19 \oplus -18 = 3$), highest amplitude value (25) of primary share3 will be replaced by xoring it with ak_3 ($25 \oplus -2 = -25$) and highest amplitude value (18) of primary share4 will be replaced by xoring it with ak_4 ($18 \oplus 26 = 8$).

Now, following are the generated four shares which are verifiable.

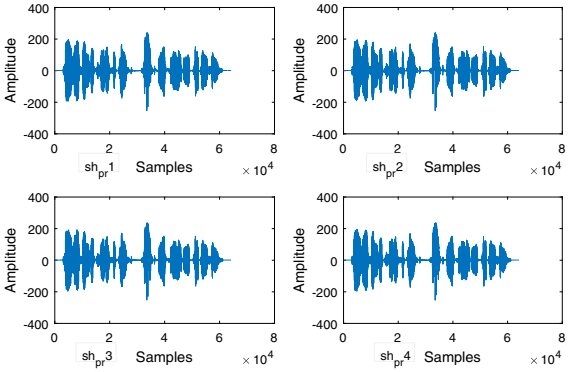
sh_v1	sh_v2	sh_v3	sh_v4
−13	−15	−25	8
16	−14	8	11
−29	3	−17	3
11	12	13	14
9	8	7	6

This embedding helps in verifying the integrity of received shares without any explicit use of its authentication key.

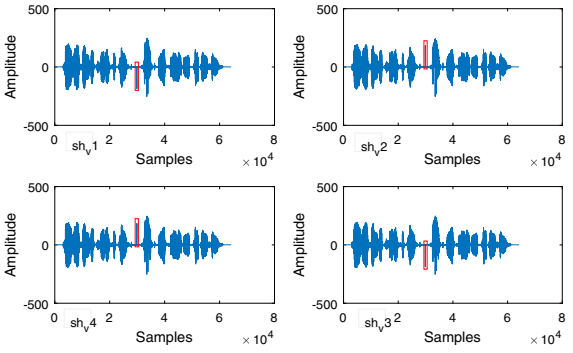
For example, let a primary share consists of four elements A, B, C and D. So, authentication key $(E) = A \oplus B \oplus C \oplus D$. Suppose, B is the largest. So, $F = B \oplus E$ will be embedded at the position of B. Hence, verifiable share will contain A, F, C and D instead



(a) S: Input audio secret (Male spoken in English language)



(b) Primary shares generated from S for n=4



(c) Verifiable shares obtained from primary shares by embedding authentication key at mid-position of the respective share. It shows suspected behavior which is shown in rectangular portion

Fig. 5 Verifiable shares generated from primary shares for n = 4

of A, B, C and D. Bitwise xoring of elements of the received share is: $A \oplus F \oplus C \oplus D = A \oplus B \oplus E \oplus C \oplus D = E \oplus E = 0$.

Hence, if no tampering is made during transmission, bitwise xoring of the elements of each share should be 0. In other words, non-zero xoring value confirms tampering.

To maintain the normal representation of shares, the authentication key is embedded at the location of maximum amplitude value of the primary share. For example, if the authentication key is embedded within the share at the silence part or the part where amplitude is very low then it can easily be identified by applying cryptanalysis techniques. This may defeat the goal of embedding (Abrupt hike may create a suspense sometimes as shown in Fig. 5c when authentication key is embedded at the mid location). Hence, it is to be embedded at the location of maximum amplitude value of the respective primary share sh_{pr} (Fig. 6).

Running time complexity of the proposed approach for verifiable share generation is O(N), where N is the total number of samples in the audio secret. Revealing of secret from received shares is discussed next.

3.4 Phase 4: revealing of secret from the shares

Algorithm 4 is used to reveal the secret from all the n shares. Number of shares (n), scaling factor used (sf) and share number (sn) are required for revealing the secret audio from the received shares. Information about number of shares (n) for a secret is publicly available. Scaling factor (sf) used can be determined from any share received as follows: $sf = \frac{sh_v(\frac{N}{n}+1)+sh_v(\frac{N}{n}+2)}{2}$.

Share number (sn) of a received share is determined as: $sn = \frac{sh_v(\frac{N}{n}+1)-sh_v(\frac{N}{n}+2)}{2}$. All the share numbers of the received shares must be distinct and lying between 1 to n for revealing the secret.

For example, received shares are sh_v1, sh_v2, sh_v3 and sh_v4. For sh_v1: $sn = \frac{11-9}{2} = 1$ and $sf = \frac{11+9}{2} = 10$. Similarly, for sh_v2: $sn = \frac{12-8}{2} = 2$ and $sf = \frac{12+8}{2} = 10$, for sh_v3:

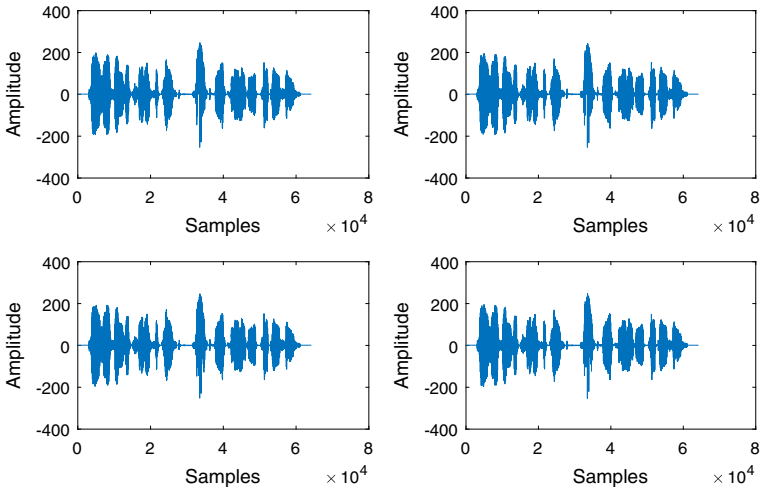


Fig. 6 Verifiable shares are obtained using Algorithm 3. No point of suspense is found

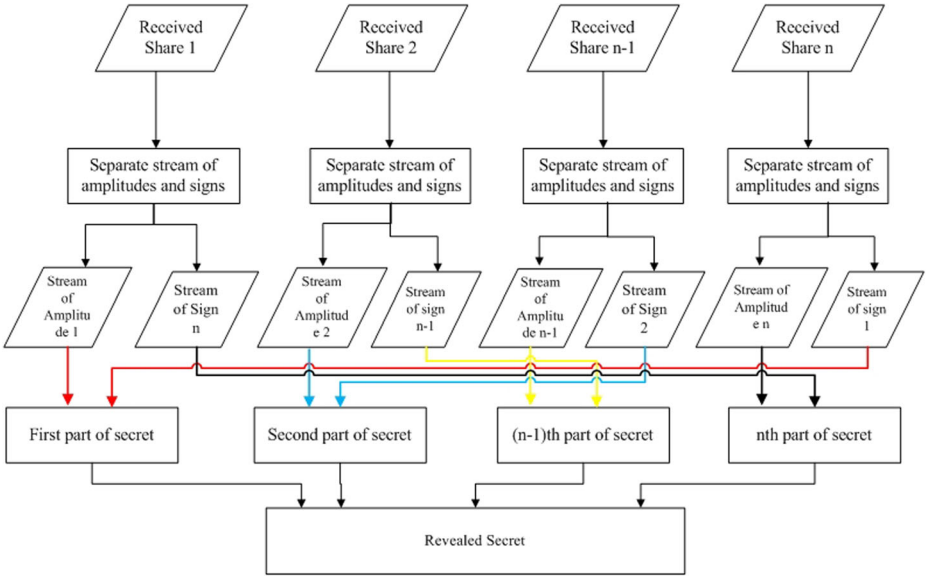


Fig. 7 Flow chart for revealing the secret

$sn = \frac{13-7}{2} = 3$ and $sf = \frac{13+7}{2} = 10$ and for sh_v4: $sn = \frac{14-6}{2} = 4$ and $sf = \frac{14+6}{2} = 10$. Share numbers are distinct and lie between 1 to 4. Hence, secret can be revealed from them. Flow chart for revealing the secret is shown in Fig. 7. Now, secret can be revealed using Algorithm 4 which is the reverse process of generation of shares. When the Algorithm 4 is applied on received shares (Section 3.3) following secret will be revealed as shown in Fig. 8.

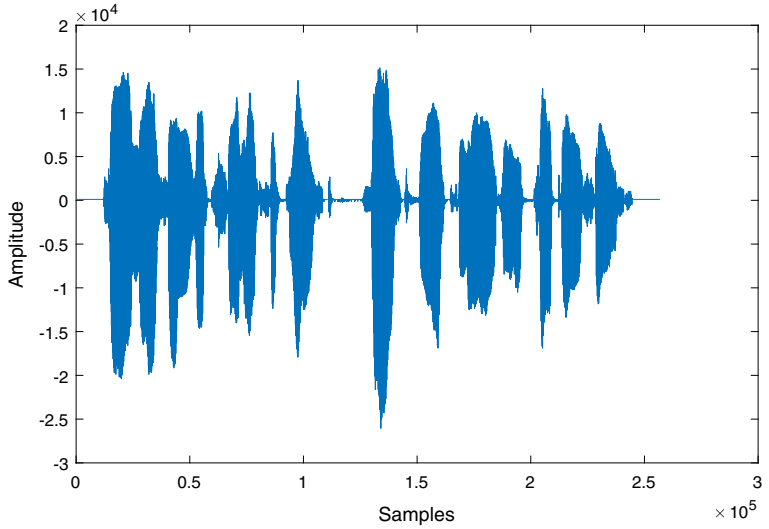


Fig. 8 Secret audio is reconstructed from verifiable share using Algorithm 4

Revealed Secret:

Index	1	2	3	4	5	6	7	8	9	10	11	12
Data	13	−15	−25	−8	16	14	−8	11	29	3	−17	−3

The running time complexity of this algorithm is O(N).

Algorithm 4 Algorithm for secret audio reconstruction

INPUT: $Sh(1...n)$:n shares each of size $\left(\frac{N}{n}+2\right)\times 1, n$: total number of shares, $share_n$: Sequence number of share.

OUTPUT: S_r : reconstructed secret speech signal of size $N \times 1$.

Ensure:

(1) $length(.)$ is a function that returns the length of a vector.

$t2 \leftarrow \dfrac{sh\left(\frac{N}{n}+1\right) - sh\left(\frac{N}{n}+2\right)}{2}$

for $j \leftarrow 1$ to n **do**

$t1 \leftarrow \dfrac{sh\left(\frac{N}{n}+1\right) + sh\left(\frac{N}{n}+2\right)}{2}$

  if $(t1 > n$ or $t1 < 1)$ **then**

    Wrong share

  else

    for $i \leftarrow 1$ to $length(share) - 2$ **do**

      if $sh[i][t1] > 0$ **then**

        $sign[i][t1] = 1$

      else

        $sign[i][t1] \leftarrow -1$

        $sh[i][t1] \leftarrow sh[i][t1] \times -1$

      end if

    end for

  end if

end for

for $j \leftarrow 1$ to n **do**

  for $i \leftarrow 1$ to $length(share) - 2$ **do**

    $data_r[i][j] \leftarrow sh[i][j] \times sign\left[\frac{N}{n} - i + 1\right][n - j + 1]$

  end for

end for

$k \leftarrow 1$

for $i \leftarrow 1$ to $length(share)$ **do**

  for $j \leftarrow 1$ to n **do**

    $recovered_s(k) \leftarrow data_r[i][j] \times t2$

    $k \leftarrow k + 1$

  end for

end for

return $recovered_s$

3.5 Phase 5: Integrity Verification of Share (IVS)

To confirm the authenticity of the revealed secret, each received share must pass the integrity test. Integrity of a share is confirmed by performing repetitive bitwise XOR operation among the first $\lceil \frac{N}{n} \rceil$ elements of that share. Non-zero xoring result alarms tampering. In that case sender is asked to retransmit the particular share.

It is worth to mention that zero result of bitwise xoring does not guarantee 100% integrity of the share. However, this can be ignored as it is a rare case as:

Probability of declaring a tampered share to be non-tampered is calculated as follows: Let k be the total number of samples in a share. Each sample is represented with b-bits.

Total possible sets of samples can be generated $= 2^{b \times k}$.

Here, it is assumed that if xoring of all the samples in a share is zero it means, share is correct. Output of bitwise XOR operation will be zero then one of the following conditions must be satisfied:

- All bits are zero at same bit position of all samples in a share.
- Even number of ones at same bit position of all samples in a share.

There is only one $({}^{k}C_0)$way to have all bits zero at same position of all samples together.

Total number of ways to have '1' in two places at same bit position of all samples simultaneously $= ({}^{k}C_2)$.

Total number of ways to have '1' in four places at same position of all samples simultaneously $= ({}^{k}C_4)$.

Total number of ways to get '1' in six places at same bit position of all samples simultaneously $= ({}^{k}C_6)$.

Total number of ways to get even numbers of '1' in k {if k is even number} places at same bit position of all samples simultaneously $= ({}^{k}C_k)$

Total number of ways to get even numbers of '1' in k {if k is odd number} places at same bit position of all samples simultaneously $= ({}^{k}C_{k-1})$.

Hence, for one bit location total number of favorable cases (to have zero output)

$$\begin{aligned} &= ({}^{k}C_0) + ({}^{k}C_2) + ({}^{k}C_4) + ({}^{k}C_6) + \cdots + ({}^{k}C_k) \quad \cdots\cdots\cdots\cdots \text{ if k is even} \\ &= ({}^{k}C_0) + ({}^{k}C_2) + ({}^{k}C_4) + ({}^{k}C_6) + \cdots + ({}^{k}C_{k-1}) \quad \cdots\cdots\cdots\cdots \text{ if k is odd} \end{aligned}$$

Since, it can happen for b bit locations Hence, total number of favorable cases (to have zero output)

$$\begin{aligned} &= (({}^{k}C_0) + ({}^{k}C_2) + ({}^{k}C_4) + ({}^{k}C_6) + \cdots + ({}^{k}C_k))^b \quad \cdots\cdots\cdots\cdots \text{ if k is even} \\ &= (({}^{k}C_0) + ({}^{k}C_2) + ({}^{k}C_4) + ({}^{k}C_6) + \cdots + ({}^{k}C_{k-1}))^b \quad \cdots\cdots\cdots\cdots \text{ if k is odd} \end{aligned}$$

Probability of declaring tampered share as non-tampered is

$$\begin{aligned} &\approx \frac{(({}^{k}C_0)+({}^{k}C_2)+({}^{k}C_4) + ({}^{k}C_6) + \cdots + ({}^{k}C_k))^b}{(2^{b \times k})} \approx \frac{1}{2^b} \quad \cdots\cdots\cdots\cdots \text{ if k is even} \\ &\approx \frac{(({}^{k}C_0)+({}^{k}C_2)+({}^{k}C_4)+({}^{k}C_6) + \cdots + ({}^{k}C_{k-1}))^b}{(2^{b \times k})} \approx \frac{1}{2^b} \quad \cdots\cdots\cdots\cdots \text{ if k is odd} \end{aligned}$$

Algorithm 5 Algorithm for IVS

INPUT: Sh_v^k: k^{th} verifiable share, n: total number of generated verifiable share.
OUTPUT: Share is tampered or not.
Ensure:
(1) $count$ is used as temporary variable.
(2) $Sh_v(k)$ represents k^{th} verifiable shares of size $(\frac{N}{n} + 2) \times 1$.
(3) $Share_i$: It represents i^{th} share number.

```
count=0
for i ← 1 to length(share)-2 do
    count = count ⊕ sh[i]
end for
if (count == 0) then
    No tampering in share
else
    Tampering in share
end if
```

Hence, the probability of declaring a tampered share as non-tampered share $\approx \frac{1}{2^b}$. So, probability of detecting tampered share as of non-tampered depends on the number of bits used to represent a sample (maximum allowed amplitude that is to be transmitted) in a share. It means if a sample is represented with 8-bits then probability of declaring a tampered share as non tampered is $\approx \frac{1}{2^8} = \frac{1}{256} \approx 0.0039$. Experimental results are analyzed in the next section.

4 Analysis of experimental results

To analyze the effectiveness of the proposed approach, experiments have been conducted with audio secrets of various size having different sampling frequencies for n=4 shares. Parameters: SNR, MOS and correlation coefficient (r) being the most popular objective/subjective parameters have been used to measure the quality of the audio signal.

Shares generation time and secret reconstruction time are also recorded. Each experiments are conducted in three cases. Clean speech is taken as secret and Algorithm 4 is used to reveal the secret from received shares.

- Case 1:- no distortion is applied on any share during transmission.
- Case 2:- any c^{th} $(0 < c \leq 1)$ fraction of a share is replaced by different real world noise or meaningful audio at different SNR values. Same thing is repeated for p $(1 \leq p \leq n)$ number of shares.
- Case 3:- distortion created is similar to the previous case but in place of replacement, addition is made.

4.1 Database used for the experiment

Audio secrets of various durations (1.5 sec to 5 sec) are taken from IndicTTS [1] database and TIMIT [5] database. Both IndicTTS and TIMIT database contain clean sentences spoken by male and female. In IndicTTS database, spoken sentences are of 13 Indian languages having sampling frequency $fs = 48000$. In TIMIT database, spoken sentences are recorded

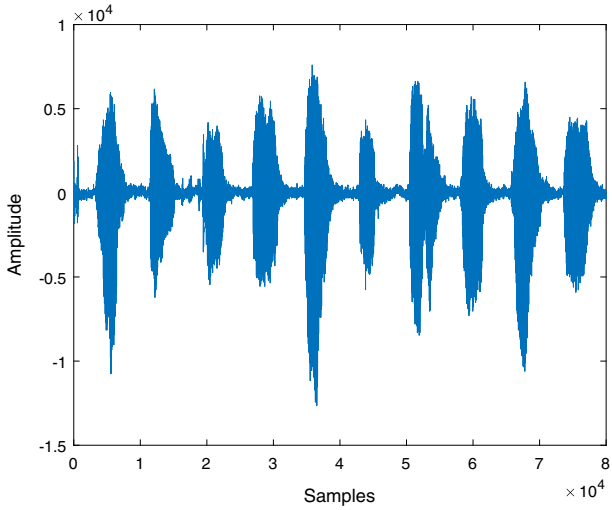


Fig. 9 Input audio secret

in English by speakers belong to 8 distinct dialect regions of the United states having sampling frequency $fs = 16000$. Some more audio secrets spoken by male and female in English language of duration (8 to 10 seconds) are also recorded in the lab used for the experiment. Various types of noise taken from NOIZEUS database are used for the experiment.

4.2 Parameters used for performance measurment

To calculate subjective parameter MOS, 20 persons are divided into two equal groups. In the group 1, original secret audio is played first thereafter revealed secret is played. Persons are asked to provide rating according to the defined labels. Average of ratings is calculated. In the group 2, only the revealed secret is played and asked to provide ratings. Finally, average ratings of group 1 and group 2 is taken as final MOS rating.

Rating	*Label*
5	Excellent quality
4	Content of audio is clear with acceptable quality
3	Less than 50% content of audio is clear with fair quality
2	Less than 25% content of audio is clear with bad quality
1	Only noise like audio is audible

Objective parameter SNR is :-

$$SNR_{db} = 10 \times \log_{10} \frac{\sum_{i=1}^{N} S^2(i)}{\sum_{i=1}^{N} (S(i) - S_r(i))^2} \tag{1}$$

4.3 Experimental results

For input share shown in Fig. 9, four primary shares (Fig. 10) are generated using Algorithm 2. Algorithm 3 is used to make them verifiable (Fig. 11). In the case 1 of the experiment when

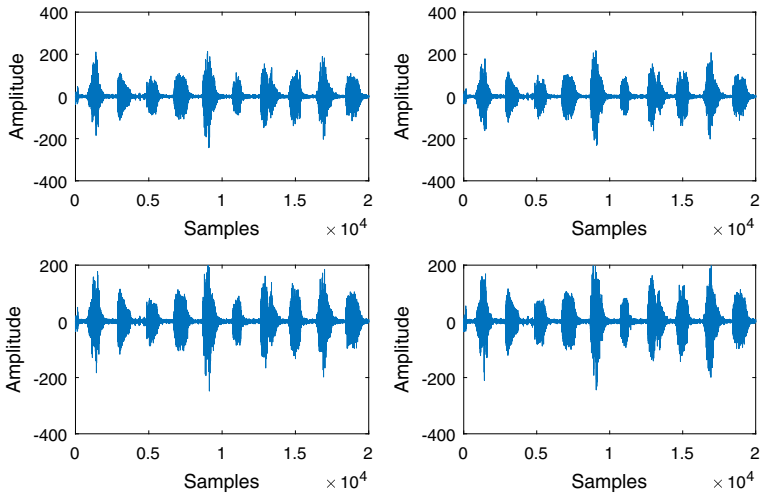


Fig. 10 Four primary shares generated using Algorithm 2

no modification is assumed during transmission the revealed secret (Fig. 12) is found to be same as original secret (Fig. 9).

Figure 13 shows the shares when attacked under case 2 for c = 1 and p = 1 where share sh_v1 is completely replaced by Babble noise at 0 dB during transmission.

Figure 14 shows the secret revealed from the attacked shares (Fig. 13). By comparing original audio secret and the revealed secret (Figs. 9 and 14), it is clear that noise added in one share has propagated throughout the secret. In this case, revealed secret could not be recovered. It shows that it is not possible to reveal the secret even if one complete share is replaced by noise. For other values of p and c, experimental results are shown in Tables 2 and 3.

Figure 15 shows the shares when attacked under case 3 for c=1 and p=1 where babble noise is added in existing share sh_v1 at 0 dB during transmission.

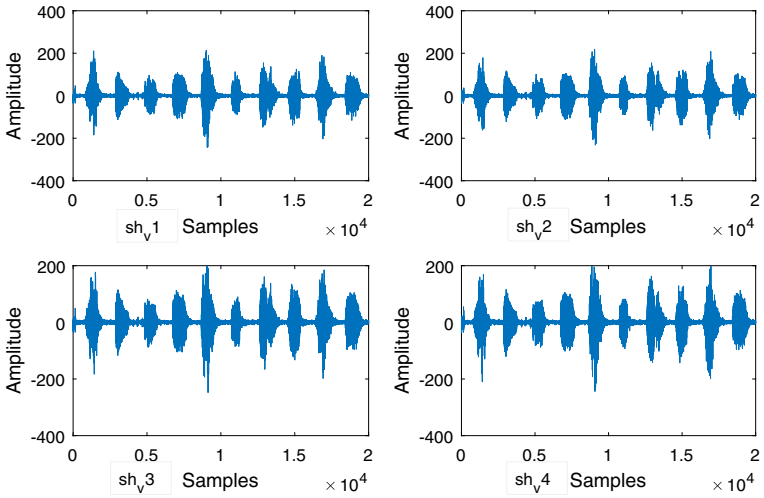


Fig. 11 Four verifiable shares generated using Algorithm 3

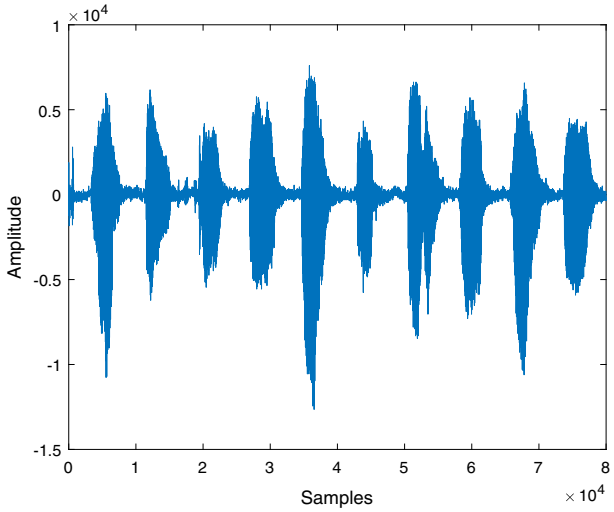


Fig. 12 Revealed secret using algorithm for phase1

Figure 16 shows the secret revealed from the above attacked shares (Fig. 15) using Algorithm 4. By comparing original audio secret and revealed secret (Figs. 9 and 16), it is clear that revealed secret S_r is completely different from the original one. It again confirms that even if one complete share is effected by noise, secret is not audible. For other values of p and c, experimental results are shown in Tables 4 and 5.

In Table 1: S, S_n and S_r represent input secret, transmitted secret and revealed secret respectively. Here, correlation coefficient (r) between $(S\&S_r)$ and $(S_n\&S_r)$ both are obtained as 0.99 instead of 1. Scaling done at preprocessing step is the cause of it. It is observed through Table 1 that MOS value of revealed secret is consistently 5 for case 1.

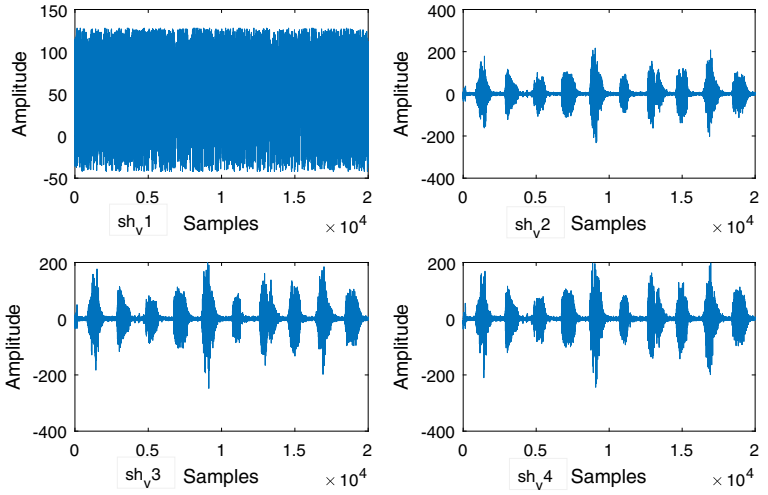


Fig. 13 Four verifiable shares generated using Algorithm 3. Share sh_v1 has been completely replaced by noise

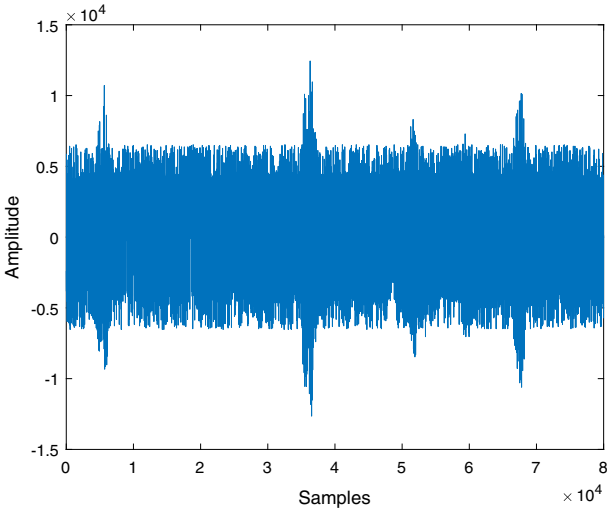


Fig. 14 Reconstructed secret using Algorithm 5 for phase2

Correlation coefficient in each case (column 6 & 7) is ≈ 1. These two observations show that revealed secret is of same quality as of original secret when there is no attack during transmission. This confirms the effectiveness of the proposed approach through secure transmission. For the experiment in case 2, p represents number of attacked shares and c represents fraction of one share attacked.

Through Table 2, it can be concluded that if single share is corrupted up to 50% then secret is audible otherwise not. Through Tables 2 and 3, it is observed that if more than one shares are attacked such that total effected part is more than 50% size of a single share then also original secret is not audible. Similar results are seen in case 3 of the experiment shown in Tables 4 and 5.

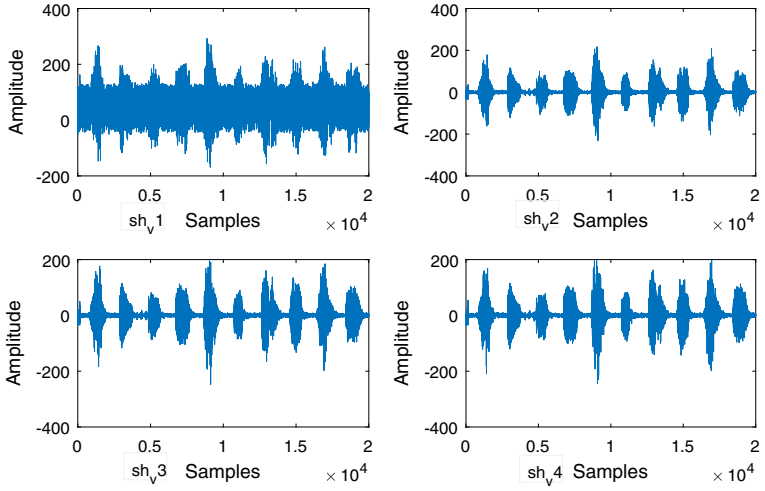


Fig. 15 Four verifiable shares sent. Babble noise is added in share sh_v1 at 0 dB

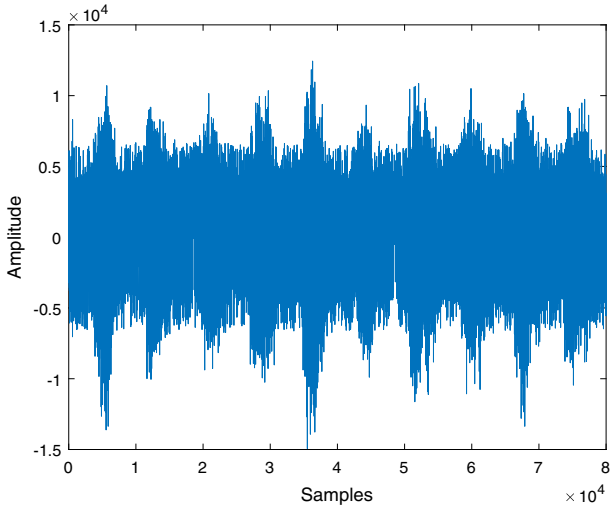


Fig. 16 Reconstructed secret using Algorithm 5 for phase3

Share generation time and secret revealing time for n = 4 shares are shown in Table 6. Here, it is observed that overall time for both generating the shares and revealing the secret is $\approx \frac{1}{70}^{th}$ size (w.r.t. time) of original secret for sampling frequency of 8000. It takes $\approx \frac{1}{38}^{th}$ size (w.r.t. time) of original secret with sampling frequency of 16000. Share generation time and secret revealing time are also recorded for n = 2 and n = 6 shares. It is similar as the result obtained for n = 4 shares. It can be concluded from here that share generation time/secret revealing time is not significantly effected by number of shares generated. For secure transmission with least possibility of attack primary shares themselves instead of verifiable shares may be transmitted. Due to lesser time consumed it can also be used for real time audio/video communication via internet.

From the Figs. 17 and 18 following observations are made:

1. Addition of some noise in more than one share is more harmful than addition of same amount of noise in one share. In other words, quality of revealed secret is better if same amount of noise is confined in single share instead of more shares.
2. Similar results are seen for case 3 which can be confirmed through Fig. 18.

Table 1 SNR, MOS and $correlation$ coefficient (r) for reconstructed secret for case 1 of the experiment

Gender	Language	Sampling frequency	SNR(dB)	MOS	r(S, S_r)	r(S_r, S_n)
Male and Female both	Hindi music	8000	22	5	0.9988	0.9988
Male	Hindi	8000	16	5	0.9994	0.9996
Female	Hindi	8000	16	5	0.9999	0.9999
Female	English	8000	14	5	0.9999	0.9999
Female	English	48000	20	5	0.9999	0.9999
Male	Bengali	48000	13	5	0.9999	0.9999
Female	Bengali	48000	12	5	0.9999	0.9999

Table 2 SNR and MOS value of reconstructed speech for case 2 of the experiment

	Noise is taken as of 0 dB		Noise is taken as of 5 dB		Noise is taken as of 10 dB	
Value of c and p	SNR in case 2	MOS in case 2	SNR in case 2	MOS in case 2	SNR in case 2	MOS in case 2
P = 1, c = 0.25	12.33	4	15.80	4	15.9	4.2
P = 1, c = 0.50	8.68	3.8	8.83	3.9	8.87	4.1
P = 1, c = 0.75	7.73	3.3	7.86	3.3	7.91	3.5
P = 1, c = 1.0	3.90	2	3.90	2	4.00	2
P = 2, c = 0.25	4.55	3.5	4.67	3.62	4.73	3.62
P = 2, c = 0.50	3.94	2.8	3.97	2.8	3.99	2.8
P = 2, c = 0.75	1.13	2.6	1.13	2.6	1.15	2.6
P = 2, c = 1.0	0	1	0	1	0	1
P = 3, c = 0.25	3.50	3	3.60	3.1	3.60	3.1
P = 3, c = 0.50	1.29	2.8	1.31	2.8	1.32	2.8
P = 3, c = 0.75	0.50	1.5	0.50	1.5	0.50	1.6
P = 3, c = 1.0	0	1	0	1	0	1
P = 4, c = 0.25	2.43	2.5	2.43	2.6	2.45	2.6
P = 4, c = 0.50	0	2.2	0	2.2	0	2.3
P = 4, c = 0.75	− 0.53	1	− 0.53	1	−0.51	1
P = 4, c = 1.0	− 01.23	1	−1.23	1	−1.19	1

Table 3 $Correlation\ coefficient(r)$ for reconstructed speech for case 2 of the experiment

	Noise is taken as of 0 dB		Noise is taken as of 5 dB		Noise is taken as of 10 dB	
Value of c and p	$r(S, S_r)$	$r(S_r, S_n)$	$r(S, S_r)$	$r(S_r, S_n)$	$r(S, S_r)$	$r(S_r, S_n)$
P = 1, c = 0.25	0.8942	0.8942	0.0.8248	0.8248	0.7643	0.7644
P = 1, c = 0.50	0.7936	0.7938	0.7006	0.7006	0.6335	0.6336
P = 1, c = 0.75	0.6941	0.6942	0.6002	0.6003	0.5430	0.5431
P = 1, c = 1.0	0.5988	0.5989	0.5193	0.5194	0.4766	0.4767
P = 2, c = 0.25	0.7910	0.7911	0.6949	0.6950	0.6279	0.6280
P = 2, c = 0.50	0.5955	0.5955	0.5248	0.5250	0.4722	0.4723
P = 2, c = 0.75	0.4060	0.4061	0.3840	0.3842	0.3762	0.3764
P = 2, c = 1.0	0.2261	0.2262	0.2908	0.2910	0.3082	0.3084
P = 3, c = 0.25	0.6910	0.6910	0.5928	0.5929	0.5352	0.5353
P = 3, c = 0.50	0.4055	0.4056	0.3814	0.3815	0.3730	0.3731
P = 3, c = 0.75	0.2504	0.2505	0.2310	0.2311	0.2287	0.2288
P = 3, c = 1.0	0.1098	0.1098	0.1304	0.1305	0.1297	0.1297
P = 4, c = 0.25	0.5932	0.5933	0.5088	0.5089	0.4661	0.4662
P = 4, c = 0.50	0.2217	0.2218	0.2767	0.2768	0.3022	0.3023
P = 4, c = 0.75	0.1033	0.1034	0.1148	0.1148	0.1229	0.1230
P = 4, c = 1.0	0.0013	0.0013	0.0014	0.0014	0.0018	0.0018

Table 4 SNR and MOS of reconstructed speech for case 3 of the experiment

	Noise is taken as of 0 dB		Noise is taken as of 5 dB		Noise is taken as of 10 dB	
Value of c and p	SNR of reconstructed secret in case 3	MOS of reconstructed secret in case 3	SNR of reconstructed secret in case 3	MOS of reconstructed secret in case 3	SNR of reconstructed secret in case 3	MOS of reconstructed secret in case 3
P = 1, c = 0.25	13.56	4.1	13.68	4.2	13.79	4.2
P = 1, c = 0.50	9.29	4	9.34	4	9.37	4
P = 1, c = 0.75	8.12	3.4	8.35	3.4	8.57	3.5
p = 1, c = 1.0	5.35	2.6	5.45	2.8	5.63	2.8
P = 2, c = 0.25	6.65	3.8	6.78	3.8	6.80	3.9
P = 2, c = 0.50	4.37	3.3	4.37	3.3	4.39	3.5
P = 2, c = 0.75	2.33	2.2	2.34	2.3	2.34	2.3
P = 2, c = 1.0	1.0	1.6	1.0	1.6	1.0	1.7
P = 3, c = 0.25	3.80	3.2	3.86	3.2	3.88	3.3
P = 3, c = 0.50	1.35	2.9	1.35	2.9	1.35	2.9
P = 3, c = 0.75	1	1.7	1	1.7	1.15	1.7
P = 3, c = 1.0	0	1	0	1	0	1
P = 4, c = 0.25	2.62	2.8	2.67	2.8	2.68	2.9
P = 4, c = 0.50	0	2.3	0	2.4	0	2.4
P = 4, c = 0.75	−0.33	1.6	−0.32	1.6	−0.29	1.6
P = 4, c = 1.0	−1.11	1	−1.11	1	−1.09	1

Table 5 $Correlation\ coefficient(r)$ value of reconstructed speech for case 3 of the experiment

	Noise is taken as of 0 dB		Noise is taken as of 5 dB		Noise is taken as of 10 dB	
Value of c and p	$r(S, S_r)$	$r(S_r, S_n)$	$r(S, S_r)$	$r(S_r, S_n)$	$r(S, S_r)$	$r(S_r, S_n)$
P = 1, c = 0.25	0.6043	0.6044	0.0.7039	0.7040	0.8715	0.8716
P = 1, c = 0.50	0.5036	0.5037	0.6796	0.6797	0.7797	0.7797
P = 1, c = 0.75	0.4583	0.4585	0.5180	0.5181	0.7080	0.7081
P = 1, c = 1.0	0.4279	0.4280	0.4414	0.4416	0.6506	0.6508
P = 2, c = 0.25	0.4995	0.4996	0.7071	0.7072	0.7755	0.7756
P = 2, c = 0.50	0.4279	0.4280	0.5793	0.5794	0.6436	0.6437
P = 2, c = 0.75	0.4052	0.4054	0.5041	0.5042	0.5518	0.5520
P = 2, c = 1.0	0.3946	0.3948	0.4055	0.4057	0.4167	0.4169
P = 3, c = 0.25	0.4502	0.4503	0.4571	0.4573	0.4365	0.4367
P = 3, c = 0.50	0.4010	0.4012	0.4126	0.4128	0.4095	0.4097
P = 3, c = 0.75	0.2832	0.2833	0.3688	0.3690	0.3175	0.3177
P = 3, c = 1.0	0.2096	0.2097	0.2478	0.2479	0.2610	0.2611
P = 4, c = 0.25	0.4222	0.4224	0.4286	0.4288	0.4170	0.4172
P = 4, c = 0.50	0.3903	0.3905	0.3777	0.3778	0.4094	0.4096
P = 4, c = 0.75	0.2020	0.2021	0.2451	0.2452	0.2581	0.2581
P = 4, c = 1.0	0.0835	0.0836	0.1147	0.1149	0.1540	0.1540

Table 6 Share generation and reconstruction time for n = 4 shares

Duration of audio in sec	Sampling frequency	Share generation time	Secret revealing time (Sec)
108 (Music Tune)	8000	0.902	0.657
279.805 (Hindi Male)	8000	2.087	1.735
285.202 (Hindi Female)	8000	2.132	1.738
106.435 (English Female)	8000	0.82	0.68
156.704 (Female English Bengali)	16000	2.22	1.98
186.13 (bengali female)	16000	2.67	2.23
106.43 (female English)	16000	1.63	1.47

5 Comparison of proposed approach with existing state of the art approaches

In order to compare with the proposed approach in this paper to the various state of the art approaches, different criteria have been taken into account as shown in Table 7.

- Threshold:- Threshold is the minimum number of shares that are required to regenerate the secret. Threshold should be high for better security of shares.
- Decryption:- Decryption of secret in audio secret sharing can be performed either without computation or with some computation. In ASS scheme, decryption with some computation is preferred because decryption without computation is affected by hearing impairment, location of audio players, effect of noise during transmission of shares etc.
- Secret type:- The types of secret which can be transmitted through audio shares.

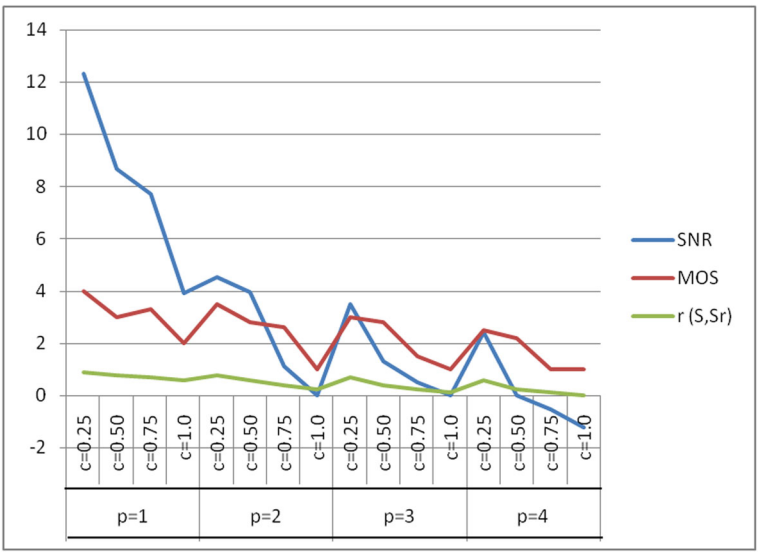


Fig. 17 SNR in dB, MOS and $r(S, S_r)$ obtained for case 2 of the experiment when babble noise is replaced at 0 dB

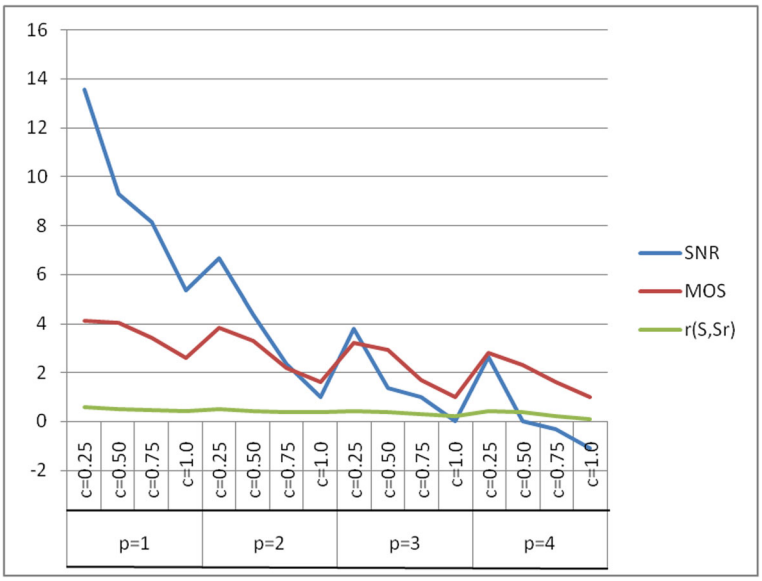


Fig. 18 SNR in dB, MOS and $r(S, S_r)$ obtained for case 3 of the experiment when babble noise is added at 0 dB

– Integrity assurance:- Provision of verifying the integrity of received shares.
– Space efficiency (M):- Ideally, for space efficient approach M should be near to 1. where, M is defined as shown in (2).

$$M = \frac{\sum_{i=1}^{n} size\ of\ i^{th}\ share}{size\ of\ secret} \tag{2}$$

where, n is the total number of constructed shares.
– Number of cover audio:- Cover audio is defined as an audio in which secret is embedded. It may be 0 or more.

Comparison results are shown in Table 7.

Table 7 Comparison of state of the art approaches for ASS

Scheme	Threshold	Decryption with or without computation	Secret type	Integrity assurance	Space efficiency	Number of cover audio required
Y. Desmedt et al. [3]	2	Without computation	binary	No	No	$\lceil \log_2 n \rceil$
Ching-Nung Yang [18]	2	Without computation	binary	No	No	1
Chen-chi-Lin et al. [6]	2	Without computation	binary	No	No	1
Md. Ehdaie et al. [4]	2 & k	Without computation	Audio	No	No	0
Jing-Zhang et al. [17]	n	With computation	Audio	No	No	n
S. Vyavahare et al. [16]	k	With computation	Audio	No	Yes	0
Proposed approach	n	With computation	Audio	Yes	Yes	0

6 Conclusion

In this paper a novel approach for verifiable (n, n) audio secret sharing scheme is proposed. It reduces the drawbacks of existing work: requirement of cover audios, requirement of more than one audio players to reveal the secret etc. Proposed approach provides the extra facility to confirm the authenticity of the revealed secret by checking the integrity of individual shares received. Experimentally it is shown that the quality of revealed secret is as good as original. Experimentally it is also confirmed that the performance of this approach is better than the state-of-art approaches in terms of space efficiency, integrity verification etc. Proposed approach is more secure in the sense that even by hacking all the shares no clue of the secret can be obtained without applying the algorithm for it. The proposed approach takes little time both for generating the shares and revealing the secret from the received shares. Hence, it can also be used to provide security for real time applications like Video chatting, Internet telephony etc.

References

1. Baby A, Thomas AL, Nishanthi NL, Consortium T Resources for Indian languages. In: CBBLR – Community Based Building of Language Resources, Sep 2016, pp 37–43, Brno, Czech Republic: Tribun EU. [Online]. Available: https://www.iitm.ac.in/donlab/tts/index.php
2. Blakley GR (1979) Safeguarding cryptographic keys. In: Proceedings of americian federation of information processing societies national computer conference, (AFIPS'79), California, pp 313–317
3. Desmedt Y, Hou S, Quisquater J-J (1998) Audio and optical cryptography. In: ASIACRYPT'98, LNCS, vol 1514, pp 392–404
4. Ehdaie M, Eghlidos T, Aref MR (2008) A novel secret sharing scheme from audio perspective. In: Proceedings of international symposium on telecommunications (IST2008). IEEE, pp 13–18
5. Garofolo JS, Lamel LF, Fisher WM, Fiscus JG, Pallett DS, Dahlgren NL, Zue V (1993) TIMIT acoustic-phonetic continuous speech corpus LDC93s1. Web download, Philadelphia: Linguistic Data Consortium
6. Lin C-C, Laih C-S, Yang C-N (2003) New audio secret sharing schemes with time division technique. J Inf Sci Eng 19(4):605–614
7. Liu Y (2015) Action2activity: recognizing complex activities from sensor data. In: Proceedings of the twenty-fourth international joint conference on artificial intelligence IJCAI, pp 1617–1623
8. Liu L (2016) Recognizing complex activities by a probabilistic interval-based model. In: Proceedings of the thirtieth AAAI conference on artificial intelligence, vol 30, pp 1266–1272
9. Liu Y (2016) Fortune teller: predicting your career path. In: Proceedings of the thirtieth AAAI conference on artificial intelligence, pp 201–207
10. Liu Y (2016) From action to activity: sensor-based activity recognition. Neurocomputing 181:108–115
11. Lu Y (2017) Towards unsupervised physical activity recognition using smartphone accelerometers. Multimed Tools Appl 76(8):10701–10719
12. Mohammad E, Taraneh E, Reza AM (2008) Some new issues on secret sharing schemes. In: Int'l conference on telecommunications (ICT' 08), June 16–19, St. Petersburg, Russia
13. NishimuraNorihiro R, Suzuki F (2005) Y suzuki audio secret sharing for 1-Bit audio. Lect Notes Comput Sci 3682:1152–1158
14. Shamir A (1979) How to share a secret. Commun ACM 22(11):612–613
15. Socek D, Magliveras SS (2005) General access structures in audio cryptography. In: Proceedings of IEEE international conference on electro information technology, p 6
16. Vyavahare S, Patil S (2016) Analysing secret sharing schemes for audio sharing. Int J Comput Appl 137(11):39–42
17. Wang JZ, Wu TX, Sun TY (2015) An audio secret sharing system based on fractal encoding. In: Proceedings of 49^{th}, international Carnahan conference on security technology (ICCST), pp 211–216
18. Yang CN (2002) Improvements on audio and optical cryptography. J Inf Sci Eng 18(3):381–391

Shambhu Shankar Bharti has received the B.Tech. degree in computer Engineering from Aligarh Muslim University, Aligarh in 2011, after that he has completed M.Tech. in computer science & Engineering from Aligarh Muslim University, India in 2013. Currently he is pursuing Ph.D. from National Institute of Technology Allahabad, India with Audio signal processing as an area of interest. His current research interest includes Audio security, Audio secret sharing, Audio enhancement, Voice Activity Detection.

Manish Gupta has received the B.Tech. degree in Information Technology from MIET, Meerut in 2002, after that he has completed M.Tech. in computer science & Engineering from YMCA Faridabad, India in 2009. Currently he is pursuing Ph.D. from National Institute of Technology Allahabad, India with Audio signal processing as an area of interest. His current research interest includes Audio watermarking, Speaker recognition, Gender classification and Emotion recognition. He is having 10 years of teaching experience.

Suneeta Agarwal received B.Sc. degree in 1973 from University of Allahabad, M.Sc. degree in 1975 from University of Allahabad, Ph.D. in 1980 from Indian Institute of Technology Kanpur and M.Tech. degree in 2007 from AAIDU. She is having 31 years of Teaching Experience and currently Professor in the Computer Science and Engineering Department, National Institute of Technology Allahabad, India. Her current research interest includes Audio security, Audio watermarking, speaker recognition, speech enhancement, voice activity detection, Pattern Recognition, Computer Vision, Theory of Computation Science, Algorithms, Automata Theory, Compression, Patten matching, Finger print recognition.