SECURE CLOUD-BASED AUDIO STORAGE AND PROCESSING

ABUKARI MOHAMMED YAKUBU

A THESIS SUBMITTED

FOR THE DEGREE OF MASTER OF SCIENCE

DEPARTMENT OF APPLIED COMPUTER SCIENCE

UNIVERSITY OF WINNIPEG

2015

Thesis Committee

- Dr. Pradeep K. Atrey (Thesis advisor)

  (On leave from) Department of Applied Computer Science

  The University of Winnipeg, MB, Canada

  (Presently at) Department of Computer Science

  State University of New York, Albany, NY, USA

- Dr. Sergio Camorlinga (Internal examiner)

  Department of Applied Computer Science

  The University of Winnipeg, MB, Canada

- Dr. Vivek K. Singh (External examiner)

  School of Communication and Information

  Rutgers, The State University of New Jersey, NJ, USA

Acknowledgments

First and foremost, I would like to offer my most profound gratitude to my thesis supervisor, Dr. Pradeep K. Atrey, for accepting me as a student and for mentoring me throughout these years. I am grateful for his continued assistance and support throughout every step of this thesis work. I appreciate his kind and inspiring words which motivated me through this work. His in-depth knowledge of research (security and privacy, multimedia computing etc.) that he shared with me helped me to adapt to thinking outside the box in my research. I am also thankful for the excellent example he has provided as a successful and accomplished professor in his field, which I hope to achieve myself some day.

I would also like express my gratitude to our research collaborator, Dr. Namunu C Maddage, for his immense support and input into the work from day one. I am grateful for his patience and the knowledge that he shared on the subject of audio processing and cloud computing.

I would also like to thank Dr. Sergio Camorlinga and Dr. Simon Liao for granting me the opportunity to take their graduate courses, as well as the Department of Applied Computer Science at the University of Winnipeg for their support and assistance.

Pursuing my master's degree in Canada has also provided me with the

opportunity to make great friends who will forever be like family. I am grateful for the ideas and foundation that Ankita Lathey laid for me in secure multimedia processing which paved the way for my research topic. I would like to thank Ashish Tripathi for always looking out for me and motivating me to study; Qianjia Huang who always invited us over for his special homemade chinese noodles, and Bakul Trehan for his long, funny speeches as well as his tales from India. Furthermore, I would like to offer thanks to my friends Orlando Simpson, Ashmeet Singh, Kanwarpreet Kaur, Manish Sharma, Abayomi Awe, Ajay Ramganesh, Gagandeep Singh, Vinay, Oluyemi Badmus, Fatima Akhmedova, Kasun Senevirathna, Shreelatha Bhadravati Sredhara, Arshia Ulaganathan , Aditya Bharadwaj, Cenker Sengoz, Parth Brahmbhatt, Waarengeye Vikram, Syed Aqeel Awais and Guoquing Xu for supporting me throughout this journey.

I would also like to thank my parents and family for all their love, support and encouragement throughout these years. Finally, I owe everything to the almighty God for giving me life and blessings.

Contents

Abstract

The outsourcing of multimedia content such as audio and speech data to Cloud Data Centers (CDCs) for storage and computation is becoming increasingly common due to their high storage and computational needs. Companies constrained in resources tend to benefit from the storage, high-end processing, elasticity, scalability and cost effectiveness of CDCs. However, the use of third party servers such as CDCs raises security concerns due to the sensitive nature of audio and speech data. Data encryption is commonly practiced to improve security. However, to process the data at CDCs, data must often be decrypted, which also raises issues in security. Thus, there is a need to protect the security of audio and speech records for storage and computation at CDCs, such that the CDC cannot learn anything from the confidential data. We propose secure methods based on (K, N) Shamir's Secret Sharing (SSS) method for audio/speech storage and computation (audio reverberation and speech noise reduction) over cloud. Our secure computation techniques are based on digital convolution reverberation and digital filtering (low pass filter, comb filter and high pass filter) for the addition of reverberation effect to an audio file and speech noise reduction, respectively. We show that our proposed schemes for storage and computation are information theoretically secure and meet the security requirements of efficiency,

accuracy and checkability for both semi-honest and malicious adversarial models. Experimental results for our proposed computation schemes (audio reverberation and speech noise reduction) in Encrypted Domain (ED) produce similar results as compared to their Plaintext Domain (PD) implementation versions whilst maintaining security and privacy with minimal overheads.

List of Tables

List of Figures

List of Abbreviations and Symbols

Abbreviation	Meaning
AES	Advanced Encryption Standard
ANOVA	Analysis of Variance
ASS	Audio Secret Sharing
CDC	Cloud Data Center
CF	Comb Filter
ED	Encrypted Domain
HAS	Human Auditory System
HPF	High Pass Filter
LPF	Low Pass Filter
LTI	Linear Time Invariant
MOS	Mean Opinion Score
PD	Plaintext Domain
PESQ	Perceptual Evaluation of Speech Quality
SSS	Shamir Secret Sharing
segSNR	Segmental Signal to Noise Ratio
SPED	Signal Processing in Encrypted Domain

Symbol	Definition
A	Original secret audio signal
A'	Preprocessed secret audio signal
A''	Scaled signal of A' to multiples of size M of low pass filter (Equation (5.9))
a_o	Original secret audio/speech sample ($a_0 \in A$)
a'_o	Preprocessed secret audio/speech sample ($a'_0 \in A'$)
a''_o	Preprocessed secret audio/speech sample ($a''_0 \in A''$)
b	Number of bits to represent the prime q
b'	Number of bits to represent the prime q'
d	Rounding precision
$D(.)$	Decryption function
$E(.)$	Encryption function
$\mathcal{E}_r$	Round-off error
$\mathcal{E}_l$	Error between ED and PD processing
$\mathcal{N}_U$	Total number of operations performed on client
$\mathcal{N}_{CDC}$	Total number of operations performed on CDC
$Pr(.)$	Probability function
$GF(.)$	Finite field

q	Prime number greater than the maximum original secret audio sample (a_o)
q'	Prime number greater than the maximum preprocessed secret audio sample (a'_o)
N	Total number of CDCs or shares
K	Threshold number of CDCs or shares required for secret reconstruction
S	Share audio/speech
U	Client or user
α	Efficiency ratio
β	Checkabilty probability
γ	Percentage loss in accuracy
$\mathbb{Z}$	Set of integers
$\mathbb{R}$	Set of real-valued numbers
fs	Sampling frequency of signal
$\delta[n]$	Unit impulse
$x[n]$	Input sample of an FIR system
$y[n]$	Output sample of an FIR system
L	Number of sample of audio/speech signal
$\varphi, \lambda, \lambda', \vartheta, \vartheta'$	Constant additive shifts to the signal to avoid negative numbers
S'_{REV}	Processed share audio with added reverb effect over cloud
$r_{(ED,PD)}$	Pearson's similarity coefficient between ED and PD processing

$\bar{N}$	Number of honest CDCs or untampered shares
$\hat{N}$	Number of malicious CDCs or tampered shares
w	Total number of ways of reconstructing audio/speech secret
$\bar{w}$	Total number of ways of reconstructing untampered audio/speech secret
$\hat{w}$	Total number of ways of reconstructing tampered audio/speech secret
$\bar{A}$	Set of untampered reconstructed audio/speech secret
$\hat{A}$	Set of tampered reconstructed audio/speech secret
$\mathcal{D}(.,.)$	Distance function
M	Size of low pass filter
D	Delay component of comb filter
g	Gain or feedforward coefficient of comb filter
g'	Scaled gain of comb filter from real-valued to integer
f_o	Fundamental frequency of humming noise
p	p-value for statistical test
q_{LPF}	Modulus prime for creating shares for low pass filtering (Diff. eqn. scheme)
q'_{LPF}	Modulus prime for creating shares for low pass filtering (Conv. scheme)
q_{CF}	Modulus prime for creating shares for comb filtering (Diff. eqn. schemeh)
q'_{CF}	Modulus prime for creating shares for comb filtering (Conv. scheme)
q_{HPF}	Modulus prime for creating shares for high pass filtering (Diff. eqn. scheme)
q'_{HPF}	Modulus prime for creating shares for high pass filtering (Conv. scheme)
b_{LPF}	Number of bits representing q_{LPF}

b'_{LPF} Number of bits representing q'_{LPF}

b_{CF} Number of bits representing q_{CF}

b'_{CF} Number of bits representing q'_{CF}

b_{HPF} Number of bits representing q_{HPF}

b'_{HPF} Number of bits representing q'_{HPF}

S'_{LPF} Processed low pass filtered (Difference eqn. approach) share over cloud

S''_{LPF} Processed low pass filtered (Convolution approach) share over cloud

S'_{CF} Processed comb filtered (Difference eqn. approach) share over cloud

S''_{CF} Processed comb filtered (Convolution approach) share over cloud

S'_{HPF} Processed high pass filtered (Difference eqn. approach) share over cloud

S''_{HPF} Processed high pass filtered (Convolution approach) share over cloud

$h[n]$ Impulse response of an FIR system

h_{REV} Impulse response of an acoustic space for convolution reverb

h'_{REV} Scale impulse response to convert h_{REV} to integer samples

h_{LPF} Impulse response of low pass filter

h_{CF} Impulse response of comb filter

h'_{CF} Scale impulse response to convert h_{CF} to integer samples

h_{HPF} Impulse response of high pass filter

I_{REV} Number of sample of the impulse response h_{REV}

I_{LPF} Number of sample of the impulse response h_{LPF}

I_{CF} Number of sample of the impulse response h_{CF}

I_{HPF} Number of sample of the impulse response h_{HPF}

Chapter 1

Introduction

1.1 Motivation

Multimedia applications such as call centers, surveillance applications, telecommunication systems and emergency calling systems (911 emergency calls) produce a large amount of audio/speech content on a daily basis, most of which contains sensitive information such as names, addresses, social security numbers, credit card numbers, evidence to be used in a court of law by a jury, information with national security implications, etc.

Most of this data is outsourced to CDC for storage and high-end computing. Over the years, cloud computing has provided a framework of elastic and scalable services for data storage, high-end computing and online access to computer resources, and companies, governments and individuals are utilizing this to save costs on operations and to avoid investment in on-premise IT infrastructure, expertise and resources. Since CDCs are physically located in a different jurisdiction, and are managed by external third parties, data

security and privacy is a growing concern. For instance, a rogue or malicious employee within the CDC may use this confidential information to their own benefit.

In real world scenarios, companies generally encrypt sensitive multimedia content before uploading it to a CDC in order to protect privacy and confidentiality. In such cases, encryption schemes like Advanced Encryption Standard (AES) are used, which suffers from single point vulnerability, meaning that the security of the method lies in securing the encryption keys, which are usually entrusted to the sender and receiver. Thus, an adversary with access to the encryption key can obtain the confidential data.

Apart from storage services provided by CDCs, most clients also make use of computing services. Most importantly, when the need arises for some processing to be done on this encrypted data, the third-party server will first have to decrypt the cipher text, which will expose the confidential information. This makes the confidential data vulnerable to exploitation by an adversary. Hence, secure storage and processing of such confidential data is of utmost importance.

1.2 Thesis Goal, Challenges and Contribution

The goal of this thesis is to investigate how to protect the privacy, security and confidentiality of audio/speech records outsourced to a cloud environment for storage and processing (reverberation and noise reduction). For

this purpose we employ Signal Processing in Encrypted Domain (SPED), which is the application of cryptographic primitives and signal processing techniques to perform operations directly on encrypted signals. This way the untrusted CDC performs operations on the encrypted signal and returns the encrypted computation results to the client without having access to the sensitive information. Our proposed schemes achieve the following goals:

- Minimal computational overhead on client: The main purpose of outsourcing to a CDC is to relieve a client who is constrained in storage capacity and high-end processing capability. That is, data storage and majority of the computation should be performed on the CDC. We have shown that the efficiency requirement of our schemes delegates most of the computation to the CDC.

- Minimal transmission overhead between client and CDC: Homomorphic encryptions such as Shamir's Secret Sharing (SSS), Paillier, etc. cause data expansion in the cipher space due to the dynamic nature of multimedia content. For instance, the Paillier cryptosystem uses very large primes and exponentiation operations, which result in large message space (e.g. 1024 bits), whereas SSS can use any prime greater than the maximum sample of a signal. This is one of the reasons we choose SSS as our cryptosystem over others such as Paillier. Our proposed schemes yield minimal transmission overhead, which is bounded by the modular prime used by SSS.

- Minimal loss in accuracy: Our schemes have been shown to yield minimal loss (near zero) between ED and PD processing. That is, the

error incurred as a result of preprocessing and round-offs of real-valued signals to integers should be minimal.

- Information theoretic security: SSS is information theoretically secure, which means that an adversary with unlimited computational power cannot obtain the secret information. Our schemes leverage this feature to provide data security. We have shown that preprocessing prior to encryption has no effect on the information theoretic security of our schemes.

Contribution of this thesis is in three-fold:

- We propose a secure cloud-based storage method for audio records using SSS. Previous Audio Secret Sharing (ASS) techniques have at least one of these limitations: 1) It does not extend to the (K, N) threshold scheme, 2) Information theoretic security is not proven and 3) It has the limitations of Human Auditory System (HAS) decryption, as will be discussed in Chapter 2 . To the best of our knowledge, this is the first ASS scheme based on SSS which addresses all three limitations mentioned above, i.e. (K, N) threshold, information theoretically secure and computationally efficient decryption.

- We propose a secure implementation of convolution reverberation to artificially add reverberation effects to an encrypted audio secret over cloud. As far as we know, this is the first work to propose the application of audio effects (reverberation effect) to an audio signal in ED.

- We propose secure noise reduction schemes for speech secrets contaminated with white noise, humming noise and wind noise over cloud. To our knowledge this is the first work on speech noise reduction in ED.

1.3 Thesis Organization

The rest of the thesis is organized as follows: In Chapter 2, we present literature review on previous work in audio secret sharing for secure storage and processing of audio/speech records in ED, and background knowledge on SPED and the cryptosystem, (K, N) SSS, that we have chosen for this thesis. We present our proposed work for secure audio storage over cloud in Chapter 3. In Chapter 4, the work to securely add reverberation effects to an audio secret outsourced to cloud for storage and computation is presented. The data storage technique for this work is based on the proposed storage method in Chapter 3. In Chapter 5, we extend our proposed cloud storage technique to propose secure noise reduction schemes (low pass filtering, comb filtering and high pass filtering) to enhance the quality of speech records (contaminated with white noise, humming noise and wind noise) outsourced to cloud for storage and computation. Finally, the conclusion and future work is presented in Chapter 6.

Chapter 2

Literature Review and Background Knowledge

This chapter reviews previous work in secure audio storage with secret sharing techniques and processing of audio/speech records in ED. We also discuss the limitations of these works and how our proposed methods address them. The organization of this chapter is as follows. Section 2.1 provides a review of the previous secure audio sharing techniques and their drawbacks, followed by a review of ED processing of audio/speech data in Section 2.2. Section 2.3 and Section 2.4 discuss the cryptosystem that we use for this thesis and the security model, respectively. Finally, the chapter is summarized in Section 2.5.

2.1 Secure Audio Storage

Audio data is typically encrypted prior to storage or transmission in order to protect it from an adversary due to the fact that it might contain sensitive information. Several cryptographic techniques used to secure audio data in real world scenarios include both private key and public key cryptosystems. One of the most widely used methods is to encrypt the data is using 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. This technique is called an Audio Secret Sharing (ASS) scheme.

Some of the existing ASS schemes [33], [11] 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 a $(2, N)$ threshold and never extended to the general (K, N) threshold. The ciphertext was decrypted by the Human Auditory System (HAS) by simultaneously playing authorized shares, which is analogous to the 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 the 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 [11]. While the schemes proposed in [33], [11] encrypt a binary secret message, the schemes in [12], [68], [62] encrypt an audio secret. However, decryption still requires the human auditory system.

The scheme proposed in [12] 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 theoretic point of view and is highlighted in [68], [62]. The authors in [68], [62] propose schemes whose security is evaluated in terms of the mutual information between the secret and the shares from an information theoretic perspective. The scheme in [62] is an improvement to [68], where the encryption function uses normal distribution over a bounded domain in order to create bounded shares. However, both schemes do not extend to the (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 must be performed on a computer. The scheme in [19] achieved decryption computationally, but it 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 theoretic point of view. In summary, each one of the previous schemes has at least one of these limitations: 1) It does not extend to the (K, N) threshold scheme, 2) Information

Table 2.1: A comparison of the proposed scheme with previous schemes

Scheme	Threshold	Information theoretic security	Decryption
Desmedt et al. [11]	$(2, N)$	Yes	HAS
Lin et al. [33]	$(2, N)$	Yes	HAS
Nishimura et al. [19]	(N, N)	Not proven	Computer
Ehdaie et al. [12]	(K, N)	Not proven	HAS
Yoshida and Watanabe [68]	(N, N)	Yes	HAS
Washio and Watanabe [62]	(N, N)	Yes	HAS
Proposed scheme [65]	(K, N)	Yes	Computer

theoretic security is not proven and 3) It has the limitations of HAS decryption.

In this thesis, 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, is information theoretically secure and 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 [1]. Another work [7] 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 2.1 compares the limitations of previous techniques and highlights that the proposed scheme for secure audio storage does not have such limitations.

2.2 Audio and Speech Processing in ED

Multimedia data processing in ED has employed fields in signal processing and cryptography to make computation on encrypted signals possible. This merger of signal processing and cryptography techniques is a totally new interdisciplinary framework called SPED [22]. Work done so far is this area has applied cryptographic primitives - Secure Multiparty Computation[1] (SMC) [66], Commitment Schemes [10], [18], Zero-knowledge Protocols (ZKP) [49], Private Information Retrieval [8] and homormophic encryption [46], [14] to develop schemes based on the security requirements of the application scenario to make secure signal processing possible. SPED has been applied in applications such as secure processing of medical data (MRI, ECG, DNA) [4], secure digital watermarking [48], Data mining on private databases [36], [39], Protecting Privacy in video surveillance systems [54]. etc.

In our literature review, we found that SPED research has been focused mainly on image and video data, yet audio and speech data have been explored far less. Research on speech processing in ED has been limited to speech classification tasks. Work in [28] and [3] proposed techniques for speaker recognition and verification over encrypted voice over IP (VoIP) conversations. Both works were based on speaker dependent packet-length information extracted from encrypted VoIP signals to build models for speaker identification and verification. Encoding of VoIP traffic to narrow band prior to encryption is a common practice to save transmission bandwidth. Encod-

[1]Protocols that allow multiple parties to jointly compute a public function over their inputs in a secure way such that their inputs are kept private. Secret sharing, Yao's protocol, secure-two-party computation and Oblivious Transfer (OT) are examples of primitives of SMC

ing techniques such as variable bit rate (VBR) and voice activity detection (VAD) used in real life scenarios results in variable length VoIP packets. There is a relationship between this length and a speaker's identity which remains unchanged even after encryption with secure real time transport protocol (SRTP) based on AES. The basic idea behind the works [28] and [3] stems from observing the relationship between the speaker's identity and the length of the packet carrying their VoIP speech contents. Hence, by utilizing discrete hidden Markov models (HMMs) and GMM to create models for each speaker based on the sequence of the packet-length extracted from encrypted VoIP conversations, speaker identification and verification from encrypted VoIP packets can be achieved. Work [28] utilized VBR, while [3] employed VAD.

Works [47] and [53] also present a framework for speaker verification/ identification and sound recognition/classification respectively using Gaussian Mixture Models (GMM) and likelihood ratio test in ED. Both methods are based on SMC and homomorphic encryption (Paillier and Boneh-Goh-Nissim (BGN) cryptosystem), which enables computation and classification to be performed in secure way. Work [47] proposes a client-server two party setting where the client has a speech sample and the server stores the encrypted model (GMM parameters of the speech) after the training pro-

[2]Knowledge of the ciphertext (and length) of some unknown message does not reveal any additional information about the message that can be feasibly extracted [63].

[3]The cryptosystem is unbreakable even when the adversary is computationally unbounded. This means that a ciphertext reveals nothing about the underlying plaintext, and thus an adversary who intercepts a ciphertext learns nothing about the plaintext that was encrypted [27]. While Information theoretically secure means that the ciphertext does not reveal any information about the plaintext, semantic security implies that any information revealed cannot be feasibly extracted.

Table 2.2: Comparison of work on audio and speech processing in ED

Scheme	Task in ED	Cryptographic primitive	Techniques used	Security achieved
[28]	speaker identification and verification over encrypted voice over IP (VoIP) conversations	AES	VBR and HMM	Security lies in protecting the private keys (one key for encryption and decryption which when not protected leads to single point vulnerability)
[3]	speaker identification and verification over encrypted voice over IP (VoIP) conversations	AES	VAD	Same as [28]
[47]	speaker verification and identification	SMC (Yao's protocols) and homomorphic encryption (Paillier cryptosystem)	GMM	Security lies in: (1)The difficulty in solving very complicated and complex mathematics with large primes. This is mostly computationally expensive to encrypt and decrypt data which means more overhead for resource constrained clients. (2)protecting the private keys (3)Semantically secure [2]
[53]	sound recognition and classification	SMC (secure two-party) and homomorphic encryption based on [21]	GMM	Same as [47]
Proposed scheme [64]	addition of reverb effect to an audio file	homomorphic encryption (SSS)	digital convolution reverb	Security lies in: (1)The uniqueness of each polynomial used to compute a ciphertext and does not rely on solving very complicated mathematics as with asymmetric cryptosystems (Pallier etc.). It is lightweight for encryption and decryption, and resource constrained clients will require less computation overhead. (2)Collective control and it is not vulnerable to single point failure as with AES (3)Information theoretically secure [3]
Proposed scheme	speech noise reduction	homomorphic encryption (SSS)	Digital linear filters	Same as [64]

cess. For secure speaker recognition/verification to be performed, the client sends the encrypted feature vectors (mel frequency cepstral coefficient) of the speech sample to the server which then computes the inner products between the encrypted feature vectors and the encrypted GMM models using

the homomorphic properties of the Paillier and BGN cryptosystem to obtain a score. The score is then compared with a threshold in order to make a decision as to whether there is a match or not. The authors utilize secure comparison protocols (secure maximum index protocol and Yao millionaire protocol) for the matching process. Work [53] applies the same approach for secure classification of sound.

The security of previous work on audio and speech processing in ED lies in protecting the encryption and decryption keys, which means that an adversary with access to the keys can obtain the plaintext data. Moreover, these methods are computationally expensive as a result of the large message space (1024 bits) and exponentiation operation of the Paillier cryptosystem. Our proposed work for secure audio reverberation and secure speech noise reduction over cloud based on (K, N) SSS is light-weight in terms of the modular prime, gives collective control to decryption and is not vulnerable to single point failure. These are the main reasons why we choose (K, N) SSS as our cryptosystem over others such as Paillier and BGN homomorphic cryptosystems.

2.3 The Cryptosystem

2.3.1 SSS scheme

Shamir introduced his scheme in 1979 [52]. His scheme 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 secret data among N participants, a polynomial function $f(x)$ is constructed with a degree of $K-1$ using K random coefficients $a_1, a_2 \ldots a_{k-1}$ in a finite field $GF(q)$ where a_0 is the secret, and q is a prime number $> a_0$.

$$f(x) = (a_0 + a_1 x + \cdots + a_{K-1} x^{K-1}) mod\ q \tag{2.1}$$

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. $f(0) = a_0$

$$f(x) = \sum_{j=1}^{K} \left(y_j \prod_{i=1, i \neq j}^{K} \left(\frac{x - x_i}{x_j - x_i} \right) \right) mod\ q \tag{2.2}$$

2.3.2 Homomorphic encryption

A cryptosystem is homomorphic if computation on its ciphertext yields an encrypted result, which when decrypted, will match the result of some computation on its plaintext. Homomorphic encryption is expected to play an important part in cloud computing, allowing companies to store encrypted data in a public cloud and take advantage of the cloud provider's analytic services. As depicted in Figure 2.1, let m_1 and m_2 belong to the plaintext

space of some cryptosystem, and $E(.)$ and $D(.)$ denote the encryption and decryption functions respectively. If $\oslash_P$ and $\oslash_E$ denote operations in PD and ED respectively, then the cryptosystem is homomorphic on the operator $\oslash_P$ if it satisfies:

$$E(m_1) \oslash_E E(m_2) = E(m_1 \oslash_P m_2) \tag{2.3}$$

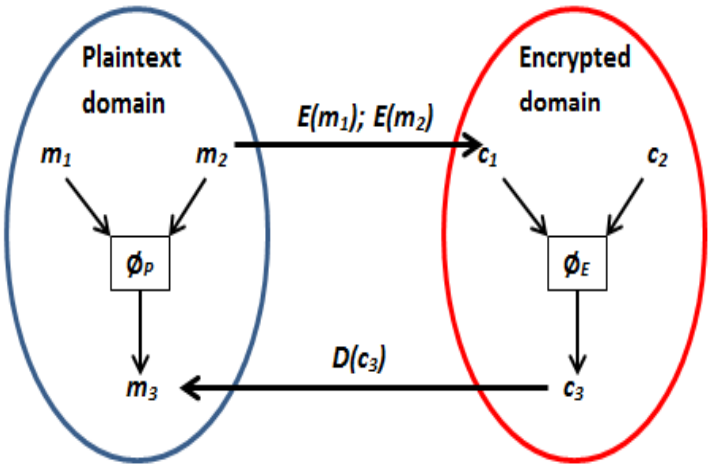


Figure 2.1: Homomorphism

The cryptosystem is additive homomorphic if $\oslash_P$ is the addition operator $(+)$ and multiplicative homomorphic if $\oslash_P$ is the multiplication operator $(\times)$. SSS scheme is homomorphic on addition and multiplication, meaning that addition and multiplication operations can be implemented in ED. However, subtraction and division operations are not possible unless some intelligent preprocessing techniques are done before encryption.

1. Subtraction- we add an additive constant to the plaintext prior to encryption. This constant should be large enough to avoid negative numbers for all ciphertext within encrypted domain.

2. Division- we scale the plaintext by the divisor (the divisor to be used in ED) before encryption.

Thus, a linear system which can be decomposed into addition, multiplication, subtraction and division is feasible in ED. In this work, we take advantage of the homomorphic property of SSS to do linear computation on encrypted audio/speech data.

2.4 Security Model and Requirements

A security model outlines requirements in terms of confidentiality, integrity, non-repudiation, authentication, efficiency, etc., that a system has to meet in order to achieve security. These requirements are dependent on the security needs of an application scenario. In our application scenario, a resource-constrained client U wants to outsource storage and computation of an audio/speech secret in a secure manner to a CDC such that the CDC learns nothing about the secret. We formulate our security model in terms of (i) data confidentiality- no information about the secret data is learned, (ii) integrity checks- identification of tampered encrypted data and (iii) secret recovery- recovering the secret without using tampered data. To achieve this, we make assumptions that (1) the client is reliable (honest) and does not upload tampered shares to the CDCs and (2) Any $K \leq N$ CDCs are non-colluding meaning that they do not come together to reconstruct the secret. When designing security models, it is important to take into account the possible behavior of parties involved in the protocol. There are two common adversarial models used to categorize such behaviors, and our security

model addresses them both:

1. Semi-honest (passive adversary): The CDC follows the computation protocol semi-faithfully, meaning that it can be unintentionally faulty in its computations, but most importantly it attempts to infer sensitive/confidential information from its hosted share.

2. Malicious (active adversary): this behavior presents a more realistic case where a CDC deviates from the protocol. Here the CDC can dishonestly tamper with shares by injecting false data or returning false computation results to the client.

Transmission security between the client and CDC is provided by Transport Layer Security (TLS) and Secure Sockets Layer (SSL) protocol from Internet protocol suite or IPSec if the client is using a VPN tunnel connection to the CDC.

In addition to the above, our secure outsourcing scheme should satisfy the following efficiency, checkability and accuracy requirements:

- α-efficiency [24]: A secure outsource computation model is efficient if the server (CDC) relieves the client (U) of majority of its computational load. That is, the computation performed locally by U should be substantially less than that performed at the CDCs. This is represented by α which is the ratio of the total number of operations performed by U to the total number of operations performed by CDC. α should be as less as possible.

- β-checkability [24]: This requirement allows U to be able to check the

faulty computations done by malicious CDCs with a probability no less than β . The value of β should be non-negligible.

- γ-accuracy: Multimedia signal processing may involve floating point ($\mathbb{R}$) operations whereas homomorphic cryptographic techniques operate in some finite ring on integer values ($\mathbb{Z}$). Quantization is often performed to round or scale $\mathbb{R}$ to $\mathbb{Z}$, which may introduce some loss in accuracy. This requirement allows U to perform the operation securely with at most γ % loss in accuracy. The value of γ should be negligible.

2.5 Chapter Summary

This chapter discussed previous work in the area of secure audio/speech storage and processing in ED. It also provided a description of the homomorphic properties of SSS and the requirements that our security model should meet for outsourcing storage and computation to an untrusted CDC.

Chapter 3

Secure Cloud-based Audio Storage Scheme

A client/customer who is constrained in storage wants to securely outsource storage of a confidential audio data to a CDC. An important aspect of our outsourcing model for storage is that the CDC does not learn any information about the confidential audio data. We apply (K, N) SSS to encrypt the audio secret into N shares that can be distributed among N CDCs ($K \leq N$ non-colluding) such that $K \leq N$ number of shares can be retrieved by an authorized user to reconstruct the secret; individual shares are of no use on their own. The contribution of this chapter is to present how to apply (K, N) SSS to protect the security of audio data over cloud in order to address the drawbacks of previous ASS schemes as already discussed in Section 2.1.

The rest of this chapter is organized as follows. In Section 3.1, we present the proposed method for secure audio storage, the data overhead and the security analysis. Section 3.2 discusses the experimental results and we sum-

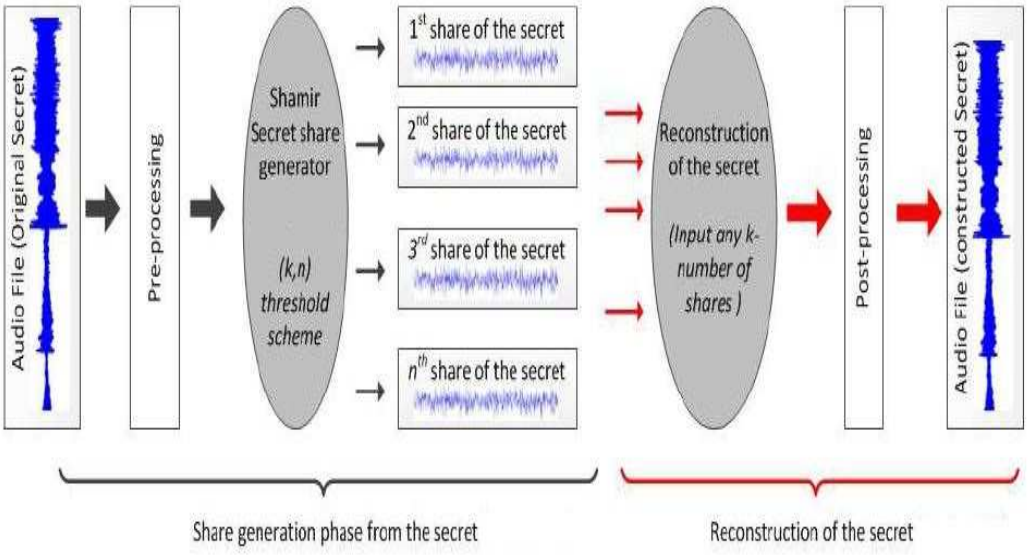


Figure 3.1: Audio secret sharing framework

marize this chapter in Section 3.3.

3.1 Proposed Method for Secure Audio Storage over Cloud

We apply the SSS scheme to create an audio secret sharing method as depicted in Figure 3.1. In our method, we create shares of amplitude samples since they contain the information of an audio signal. The following section explains share generation and reconstruction of the secret audio.

3.1.1 Audio preprocessing and share generation

Using the (K, N) SSS 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. Round-off error is bounded by:

$$-\frac{1}{2} \times 10^{1-d} \leq \mathcal{E}_r \leq \frac{1}{2} \times 10^{1-d} \tag{3.1}$$

where $\mathcal{E}_r$ 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 φ to obtain positive sample values within $\mathbb{Z}_p$. Shifting the signal to first the quadrant does not distort the waveform, as illustrated in Figure 3.2.

$$a_0' = \big((a_0 + \mathcal{E}_r) \times 10^d\big) + \varphi \tag{3.2}$$

Using Equation (2.1) from Section 2.3, N shares are created and distributed to N participants. The algorithm is shown below.

Algorithm 1: Audio share generation

Input: Secret audio $A = \{A_1, A_2 \dots A_m\}$; where A_m is the amplitude at the m^{th} time interval

Output: Secret Shares $S_1, S_2 \dots S_n$

Description:

1. Read wav file i.e. $[A, fs]$ = wavread('*wavfile*')
2. $A = \text{round}((A + \mathcal{E}_r) \times 10^d)$
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 of 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** $S_1, S_2 \ldots S_n$;

3.1.2 Audio secret reconstruction and post-processing

To reconstruct the secret audio we need at least K out of the N shares. Referring to Figure 3.1, there are two blocks at the secret reconstruction phase: 1) reconstruct the secret by using Equation (2.2) to solve the polynomial

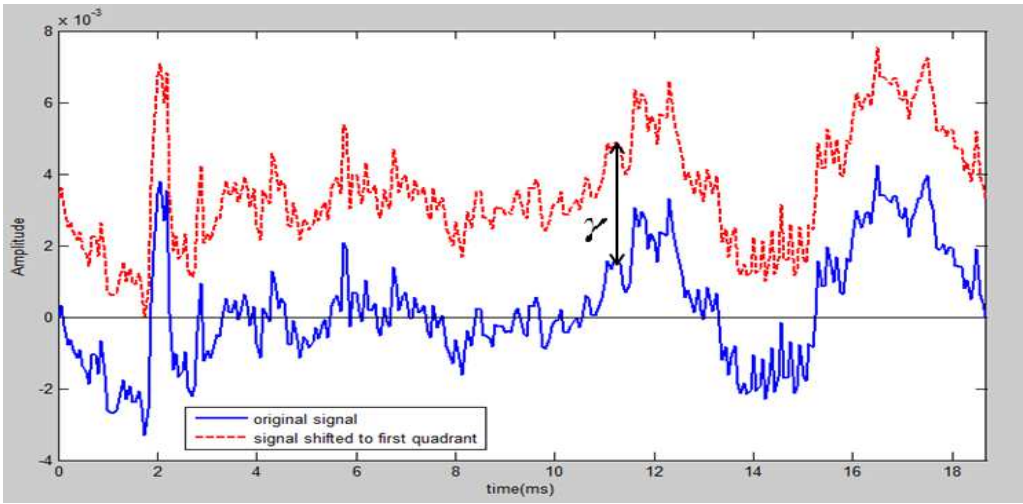


Figure 3.2: Shifting signal to first quadrant

function in Equation (2.1) 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: Audio 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 Equation (2.2) 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;

3.1.3 Data overhead

Our proposed scheme introduces some data overhead to transmit a share to a CDC. This is 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 $maximum[(a_0 + \mathcal{E}_r) \times 10^d + \varphi])$ 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:

$$b' = \log_2(q') \tag{3.3}$$

Due to the dynamic range of audio signals, q' will always vary for different audio signals depending on the quantization level (8 bit, 16 bit etc.) of the ADC converter used during quantization. From Equation (3.1), 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.

3.1.4 Security analysis

The proposed method is based on the (K, N) SSS threshold scheme which is proven to be information theoretically secure [56]. 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 theoretic security. We evaluate the security of our proposed method using the below corollaries based on theorems in [30].

Corollary 1 *Information theoretic 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 $a'_0 = (a_0 \times 10^d) + \mu$ shared under $GF(q')$ (where $\mu = \left(\mathcal{E}_r \times 10^d\right) + \varphi$ from Equation (3.2) and q' is a prime number greater than $(q \times 10^d) + \mu$)*

Proof 1 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:

$$Pr(a_0)_{0 \leq a_0 \leq q-1} = \frac{1}{q} \tag{3.4}$$

Similarly, for each plaintext a'_0 of the preprocessed audio secret A' where $a'_0 = (a_0 \times 10^d) + \mu$ there is also an equal probability of being any value from

Table 3.1: 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

the set $0 \leq a'_0 \leq q' - 1$ of q values with probability given as:

$$Pr(a'_0)_{0 \leq a'_0 \leq q'-1} = \frac{1}{q} \tag{3.5}$$

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 theoretic security after preprocessing the original audio secret. An adversary in both cases will have to guess the secret with a probability of $\frac{1}{q}$.

3.2 Experimental Results

Table 3.1 details the 6 audio files obtained from [57] 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.

We implemented the audio secret sharing method using MATLAB14 on a 2.53GHz i5 CPU with 4GB RAM. Table 3.2 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 that of creating secret shares. Since the proposed method is applied at an audio sample level, the processing time

Table 3.2: 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

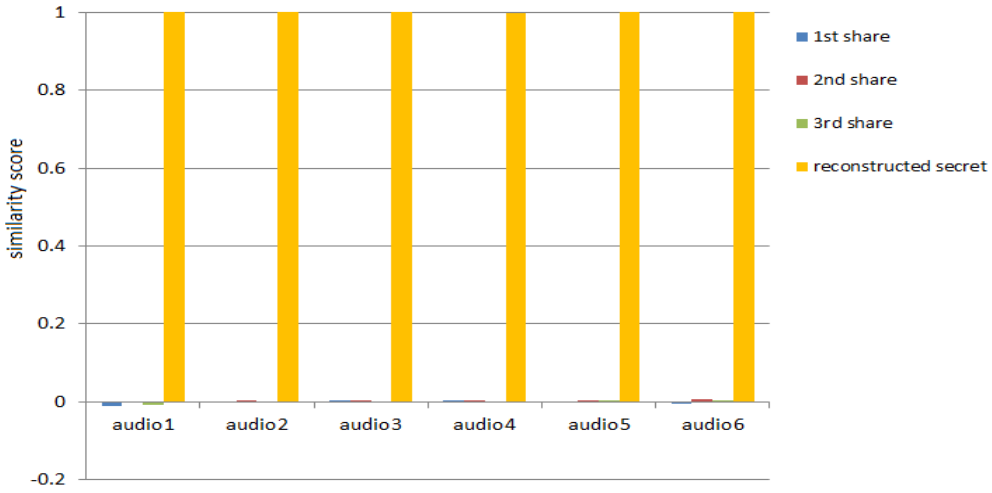
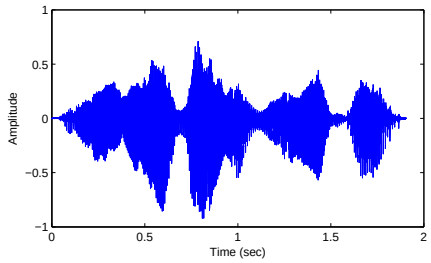


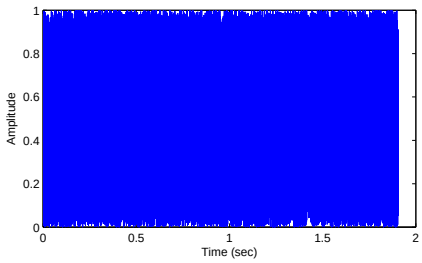
Figure 3.3: Similarity score for 6 audio clips

is directly proportional to the audio bit rate, which is associated with the sampling frequency and number of bits per sample.

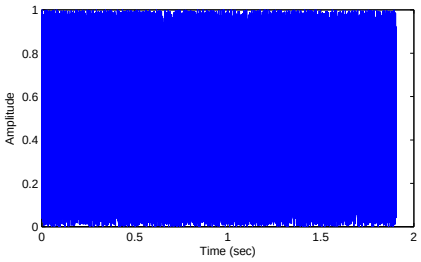
Audio signals by nature have correlating adjacent samples and the use of random coefficients as a blinding factor in Equation (2.1) to generate shares eliminates this correlation. Thus, individual shares do not reveal information about the secret audio. The time domain plots of one of our test audio files (audio1) in Figure 3.4 illustrate: 1) the difference between the audio secret and its noisy shares and 2) the similarity between the reconstructed and original secret audio. Figure 3.3 shows the similarity scores between the



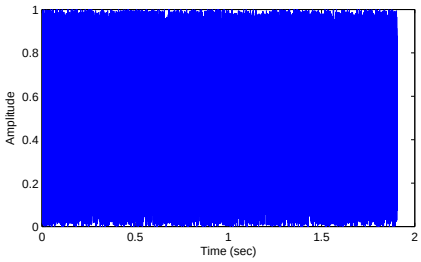
(a) Original secret audio



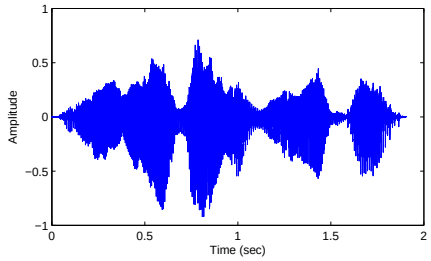
(b) 1st share



(c) 2nd share



(d) 3rd share



(e) Reconstructed secret audio

Figure 3.4: audio1, its shares and reconstructed secret

original secret audio, and 1st share, 2nd share, 3rd 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 loss due to rounding error in the preprocessing step.

We also performed a listening study to evaluate perceptual security, which

Table 3.3: 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

was conducted online [1]. User scores are summarized in Table 3.3. 20 subjects in the age range of $20-40$ years old 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 the audio secret are completely dissimilar in terms of content. However, about a 92% average similarity score was achieved for content similarity between the original audio secret and the reconstructed audio secret which confirms that our proposed scheme is perceptually secure. However, as depicted in Figure 3.3, using Pearson's correlation analysis, we were able to establish about a 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 3.3) vs machine evaluation (Figure 3.3) of similarity.

3.3 Chapter Summary and Conclusion

In this chapter we proposed a secure audio storage scheme using the SSS scheme. Compared to existing techniques, the proposed technique meets

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

the (K, N) threshold requirement, is information theoretically secure and has computationally efficient decryption which does not rely on HAS. Our experimental results also support the fact that the reconstructed secret is identical to the original audio secret with minimal losses.

Chapter 4

Secure Cloud-based Audio Reverberation

This chapter focuses on the addition of reverb effects to an audio recording in ED over cloud. This is one of the most widely used delay effects, among others such as flanging, phasing, chorus effects etc., for audio recording, reproduction and editing. This effect adds an acoustic environment to an audio recording to make it sound realistic. The resulting reverb effected audio inherits characteristics from that acoustic environment and sounds as if the recording was created in that environment. Reverberation is a series of delayed and attenuated sound waves reflected within an acoustic environment which is perceived by the human ear in less than 0.1 seconds after the original sound wave. The human auditory system is unable to perceive the 0.1 second delay and interprets the original sound wave and delayed reflections as one prolonged sound. This effect is different from echo where delays are more than 0.1 seconds and the delayed sounds are perceived distinctly as decaying

copies of the original sound. Key areas of applications of reverb effects in audio are:

1. the recording industry for audio editing and reproduction.

2. audio forensics for analyzing/simulating an audio recording in different acoustic environments.

3. simulation of acoustic reverberation for dereverberation algorithms, etc.

In this chapter, we propose the implementation of convolution reverb to artificially add reverb effects to an encrypted audio secret over cloud to ensure information assurance of a client's data from a privacy and security perspective. To the best of our knowledge, this is the first work that applies reverb effect to an audio signal in ED.

The rest of this chapter is organized as follows. Section 4.1 provides an overview of artificial reverberation techniques. In Section 4.2 we present the proposed work for secure reverberation over cloud along with the data overhead and security analysis. Experimental results are presented in Section 4.3, followed by the conclusion and summary of this chapter in Section 4.4.

4.1 Artificial Reverberation Techniques

There are many techniques for applying reverb effects to audio. The most common techniques for artificial addition of reverb are:
1) Filter banks and delay line: this approach involves connecting filters (comb, all-pass, lowpass filters) in parallel and series and adjusting their

parameters to produce the desired reverb effect, e.g. Schroeder's Reverberator [51], Moorer's Reverberator [42] etc.

2) Convolution reverb: acoustic space is a Linear Time-Invariant (LTI) system, and like any LTI system its impulse response can be modeled and convolved with an audio signal to produce the effects of that space. Modeling of the impulse response depends on the application scenario and the needs of the designer, which are beyond the scope of this work. Some examples of modeled impulse responses are room, concert hall, cathedral, bottle hall, conic long echo hall, deep space, etc. In this work, we apply this approach to artificially add reverb effect to an audio secret in ED. Let $h[n]$ be the modeled impulse response and $x[n]$ be the audio signal. Then, the below discrete convolution will yield the reverb effected signal $y[n]$.

$$y[n] = x[n] * h[n] = \sum_{k=-\infty}^{\infty} x[k]h[n-k] \tag{4.1}$$

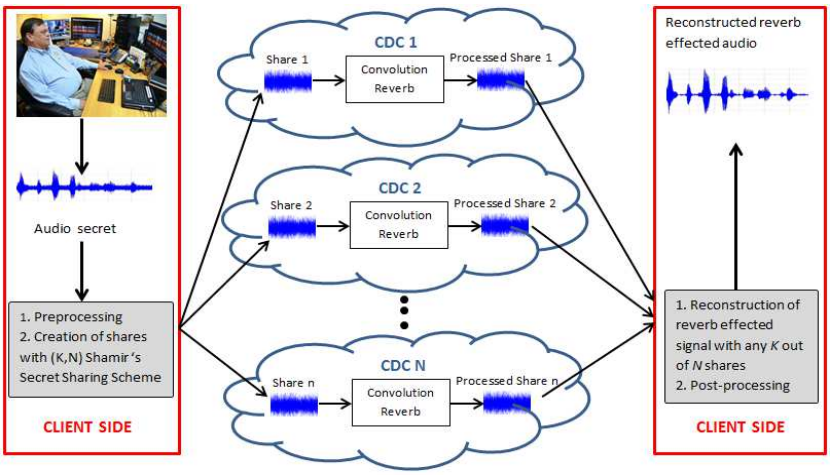


Figure 4.1: The proposed work: Addition of reverberation effect in ED

4.2 Proposed Method for Secure Convolution Reverb

The audio secret to be reverb effected is encrypted by creating shares with the (K, N) SSS threshold scheme on the client's system. The client then uploads each of the N shares to N different non-colluding CDCs. The CDC then applies the reverberation effect to their hosted shares by convolving with the desired impulse response. Reverb impulse responses are public signals in plaintext. The client does not need to transmit or upload impulse responses to the CDC since the CDC can obtain a library of all impulse responses. An authorized user then reconstructs the reverb effected secret by combining at least K out of N processed shares. Figure 4.1 represents our proposed scheme for the secure addition of reverberation effect to an audio secret over cloud.

Signal processing operations often involve real-valued signal amplitudes; however, 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 the real valued samples of the audio secret to positive integer values. Steps 1 and 2 below preprocess the original signal before creating shares. The below steps detail our method in ED.

Step 1: Scale real-valued signal amplitudes with constant factor 10^d where d is an integer value. Round-off error is bounded by:

$$-\frac{1}{2} \times 10^{1-d} \leq \mathcal{E}_r \leq \frac{1}{2} \times 10^{1-d} \tag{4.2}$$

where $\mathcal{E}_r$ is the rounding error and d is the rounding precision.

Step 2: Add a constant additive shift φ to the signal obtained from Equation (4.2) in order to avoid negative numbers.

$$A' = \left((A + \mathcal{E}_r) \times 10^d\right) + \varphi \tag{4.3}$$

Step 3: Create N shares $(S_1, S_2 \dots S_N \in S)$ of preprocessed signal A' using Equation (2.1) of SSS in the finite field of q' and upload them to N non-colluding CDCs. In order to avoid the cipher blow-up problem as a result of convolution as discussed in Section (5.2.5), the modulus prime should be the product of the maximum sample of A' and the sum of absolute values of the impulse response sequence i.e. $q' > max(A') \times \sum_{n=0}^{I-1} |h_{REV}[n]|$, where $h_{REV}[n]$ is the impulse response.

Step 4: Convolution of each share on the CDC. The modeled impulse response is real-valued and there are instances where some samples are negative. This might result in errors while performing convolution reverberation. In order to avoid errors, perform the following steps:

1) Scale impulse response h_{REV} by 10^t that is $h'_{REV} = h_{REV} \times 10^t$; where t is an integer value.

2) Modify Equation (4.1) by adding a constant additive shift ϑ' to avoid negative samples. $\vartheta' = \lceil \frac{q'}{2} \rceil$ if h'_{REV} has a negative sample, else $\vartheta' = 0$, where $\lceil . \rceil$ is a ceiling function.

Convolve each share $(S_1, S_2 \dots S_N \in S)$ with h'_{REV} using Equation (4.4) to obtain $(S'_{1^{REV}}, S'_{2^{REV}} \dots S'_{N^{REV}} \in S'_{REV})$.

$$
\begin{aligned}
S'_{i^{REV}}[n] &= \left(\left(S_i[n] * h'_{REV}[n]\right) + \vartheta'\right) mod q' \\
&= \left(\left(\sum_{k=0}^{L-1} S_i[k] h'_{REV}[n-k]\right) + \vartheta'\right) mod q' \quad , i \in \{1, 2 \ldots N\}
\end{aligned}
\tag{4.4}
$$

Where L is the number of samples of each share.

Step 5: An authorized user can reconstruct the reverb effected secret by putting together K processed shares from any of the N CDCs using Lagrange interpolation from Equation (2.2).

Step 6: Postprocess the reconstructed secret to reverse-engineer the preprocessing done in the above steps. 1) Subtract the additive shift ϑ' and divide by the scale factor of h'_{REV} which is 10^t. 2)Subtract the additive shift φ and divide by the scale factor 10^d from Equations (4.3) and (4.2) respectively.

4.2.1 Data overhead

Preprocessing prior to creating shares introduces some data overhead for our secure reverberation scheme. This data overhead per sample to transmit a share from client to CDC is upper bounded by the modulus prime q' used for share creation. If b' is the number of bits to represent q', then data overhead in bits is bounded by:

$$b' = \log_2(q') \tag{4.5}$$

4.2.2 Security analysis

The proposed method is based on the (K, N) SSS threshold scheme which is proven to be information theoretically secure [56]. The following corollary proves the information theoretic security property of our proposed work for secure reverberation over cloud.

Corollary 2 *Information theoretic 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 $a'_0 = (a_0 \times 10^d) + \mu$ shared under $GF(q')$ (where $\mu = \left(\mathcal{E}_r \times 10^d\right) + \varphi$ from Equation (4.3) and q' is a prime number greater than $(q \times 10^d) + \mu$)*

Proof 2 Proof follows the same lines as Corollary 1.

Since the generation of shares is bounded by q', it follows that samples of each share are within the set $\{0, 1, 2, ..., q' - 1\}$ and, referring to Equation (2.1) of SSS, each sample of a share is a unique polynomial. In this case, an adversary will have to guess with a probability of $\frac{1}{q}$, making it highly unlikely to infer a secret sample from its share. Audio signals, by nature, have correlating adjacent samples. This correlation reduces the entropy (degree of uncertainty) of the entire signal; i.e. a sample can be predicted from its adjacent samples as in Linear Predictive Coding (LPC). However, the use of random coefficients as a blinding factor in Equation (2.1) to generate shares

eliminates this correlation. Thus, individual shares do not reveal information about the secret audio.

Theorem 1 *Our proposed scheme in ED for the addition of reverberation effect is* $100(1 - r_{(ED,PD)})$*-accurate implementation of its plaintext version with negligible loss in accuracy, where* $0 \leq r_{(ED,PD)} \leq 1$ *is the similarity coefficient between ED and PD processing.*

Proof 3 From preprocessing in Equation (4.2) rounding error $\mathcal{E}_r$ is bounded by:

$$\mathcal{E}_r \leq \left| \frac{1}{2} \times 10^{1-d} \right| \tag{4.6}$$

$\mathcal{E}_r$ propagates to the output of the computation. We can say that $\mathcal{E}_r$ is proportional to the output error or loss in accuracy ($\mathcal{E}_l$) between ED and PD processing i.e.

$$\mathcal{E}_r = c_f \mathcal{E}_l \tag{4.7}$$

where c_f is a magnification factor which determines the growth of the error from the input to the output. Substituting (4.7) into (4.6) gives the bound of the loss in accuracy between ED and PD processing ($\mathcal{E}_l$):

$$\mathcal{E}_l \leq \frac{1}{c_f} \times \left| \frac{1}{2} \times 10^{1-d} \right| \tag{4.8}$$

From (4.8) $\mathcal{E}_l$ will always be negligible for increasing d. $\mathcal{E}_l$ can be computed from the similarity coefficient between ED and PD processed signals, i.e. $\mathcal{E}_l = (1 - r_{(ED,PD)})$, and in terms of percentage:

$$\mathcal{E}_l = 100(1 - r_{(ED,PD)}) \tag{4.9}$$

Theorem 2 *In the malicious model, our proposed scheme in ED for the addition of reverberation effect is a 1-checkable implementation of the corresponding computations over cloud.*

Proof 4 For (K, N) SSS, there are $\binom{N}{K}$ ways of reconstructing the secret, hence the client will have $w = \binom{N}{K}$ ways of reconstructing the processed signal. It should be noted that:

- Untampered reconstructed secret: All shares for reconstruction are untampered and all untampered reconstructed secrets from w will always be the same.
- Tampered reconstructed secret: At least one share is tampered and all reconstructed secrets from w that use the tampered share will always be different.

Now suppose that $\hat{N}$ (where $\hat{N} < N$) CDCs deviate from the computation protocol by tampering with shares or returning false computation results to the client. There will then be $\bar{N} = N - \hat{N}$ (where $\bar{N} > K$) CDCs with untampered shares. Consequently, this will result in $\bar{w} = \binom{\bar{N}}{K}$ and $\hat{w} = \left(\binom{N}{K} - \binom{\bar{N}}{K}\right)$ ways of reconstructing untampered and tampered secrets respectively (i.e. $w = \bar{w} + \hat{w}$).

Now let $(\bar{A}_1, \bar{A}_2 \ldots \bar{A}_{\bar{w}} \in \bar{A})$ be a set of all untampered reconstructed secrets and $(\hat{A}_1, \hat{A}_2 \ldots \hat{A}_{\hat{w}} \in \hat{A})$ be a set of all tampered reconstructed secrets, then the below holds true for comparison in pairs between untampered and tampered reconstructed secrets where $\mathcal{D}(.,.)$ is some distance metric:

$$\mathcal{D}(\bar{A}_i, \bar{A}_j)_{i \neq j} = 0; \;\; i \in \{1, 2 \ldots \bar{w}\}, j \in \{1, 2 \ldots \bar{w}\} \tag{4.10}$$

$$\mathcal{D}(\bar{A}_i, \hat{A}_j) > 0; \quad i \in \{1, 2 \dots \bar{w}\}, j \in \{1, 2 \dots \hat{w}\} \tag{4.11}$$

$$\mathcal{D}(\hat{A}_i, \hat{A}_j)_{i \neq j} > 0 \;\; i \in \{1, 2 \dots \hat{w}\}, j \in \{1, 2 \dots \hat{w}\} \tag{4.12}$$

Thus, with a distance of 0 as in Equation (4.10), the client can be certain that both reconstructed secrets are untampered with a probabilty of 1:

$$Pr\big[\mathcal{D}(\bar{A}_i, \bar{A_i})_{i \neq j} = 0\big] = 1 \tag{4.13}$$

And with a distance > 0, from Equations (4.11) and (4.12), the client can be certain that shares of either of the reconstructed secrets are tampered with a probability of 1 and hence can discard them.

$$Pr\big[\mathcal{D}(\bar{A}_i, \hat{A}_j) > 0\big] = 1, \quad Pr\big[\mathcal{D}(\hat{A}_i, \hat{A}_j)_{i \neq j} > 0\big] = 1 \tag{4.14}$$

Remark 1 The computational complexity for checkability by the client is linear to the size L of the reconstructed processed signal, i.e $O(L)$. This might cause additional overhead for the client and hence can be performed offline.

4.3 Experimental Results

Table 4.1 details the test audio files obtained from [57] that we used for experimenting with our proposed method. We test our method with a modeled impulse response [43] from the acoustic environment of a living room. That is, we add the effects of a living room to our audio test files in ED. In the (K, N) SSS scheme, we set $K = 2$ and $N = 3$, implying that we created

Table 4.1: Data set

Test file (.wav)	length(secs)	Bits/sample	sampling frequency (Hz)
audio1	2	16	16000
audio2	43	16	8000
audio3	13	16	11025
audio4	14	8	44100
audio5	4	8	8000
audio6	2	32	8000

Table 4.2: Average processing time (ms)

Test file	Share creation (offline)	ED processing	reverb effected secret reconstruction
audio1	205.18	97.77	17.81
audio2	1620.27	631.44	80.32
audio3	747.46	276.57	39.58
audio4	2802.95	1068.85	115.98
audio5	219.06	73.21	18.64
audio6	149.20	44.26	15.02

three shares of the audio secret such that at least two shares will be required to reconstruct the reverb effected audio secret.

We implemented the proposed method using MATLAB14 on a 2.53GHz i5 CPU with 4GB RAM. Table 4.2 details the processing time for creating secret shares, applying the reverberation effect in ED and reconstructing the audio secret. The table suggests that the complexity of reconstructing the secret is relatively lower than that of creating shares and ED processing. This is as a result of the additional complexity of preprocessing prior to share creation. The creation of shares can be performed offline in order to reduce complexity for the client. Our method is applied on a sample basis. As a result, the processing time is directly proportional to the audio bit rate, which is associated with the sampling frequency and number of bits per sample. Therefore, the greater the length of the signal, the greater the complexity. This is evident for audio2.wav and audio4.wav with the greatest complexities.

Table 4.3: Number of operations on cloud and client side

ED(Cloud processing)	Client side
I_{REV} multiplications $2\Big((L+I_{REV})-1\Big)$ modular additions $L \times I_{REV}$ modular multiplications 1 division	$2L$ subtractions $2L$ divisions
Note: L and I_{REV} are the number of samples of the audio signal and the impulse response h_{REV} respectively.	

Theorem 3 *Our proposed scheme for the addition of reverberation effect in ED is* $O\big(\frac{\mathcal{N}_U}{\mathcal{N}_{CDC}}\big)$*-efficient if the total number of operations* $\mathcal{N}_U$ *performed by the client is less than the total number of operations* $\mathcal{N}_{CDC}$ *performed by the CDC (where* $O(\mathcal{N}_U)$ *and* $O(\mathcal{N}_{CDC})$ *are the asymptotic complexities of the client and the CDC respectively).*

Proof 5 Proof follows from Table 4.3 which represents the number of operations performed by the client and the CDC. It is evident that for the proposed scheme in ED from Table 4.3 that: $\mathcal{N}_U < \mathcal{N}_{CDC}$ and consequently $O(\mathcal{N}_U) < O(\mathcal{N}_{CDC})$, i.e.

$$O\bigg(\frac{4L}{I_{REV} + 2((L+I_{REV})-1) + (L \times I_{REV}) + 1}\bigg) \tag{4.15}$$

Figure 4.2 shows the similarity scores between the PD and ED reverb effected signals. We computed the similarities using Pearson's correlation method. Results suggest the reverb effected signal, after applying our method, correlates about 99.99% with normal PD processing. Thus, our method yields identical results to PD processing while maintaining security and privacy. The 0.01% difference can be accounted for by the rounding-off during the preprocessing steps for both original audio secret A and impulse response h_{REV}. We hope to optimize the solution in the future to further minimize

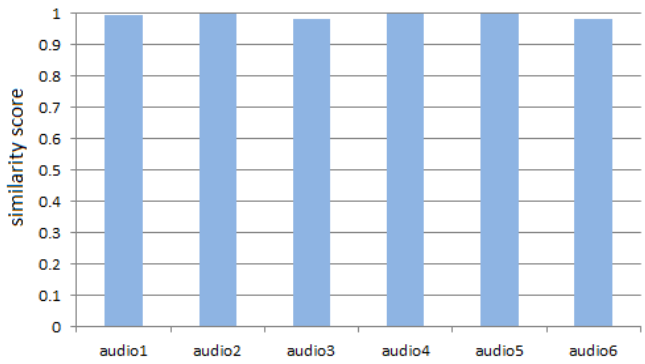


Figure 4.2: Similarity score between PD and ED processing

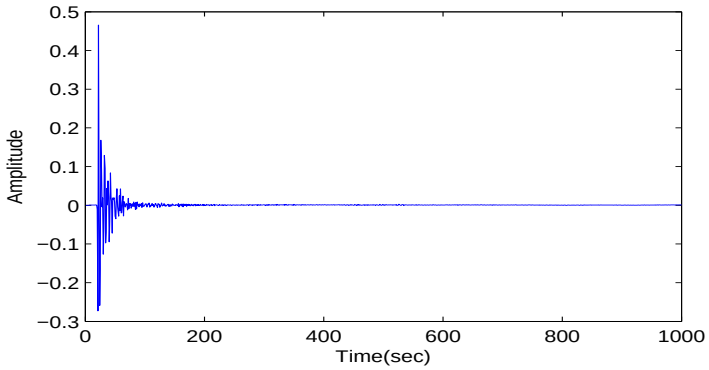
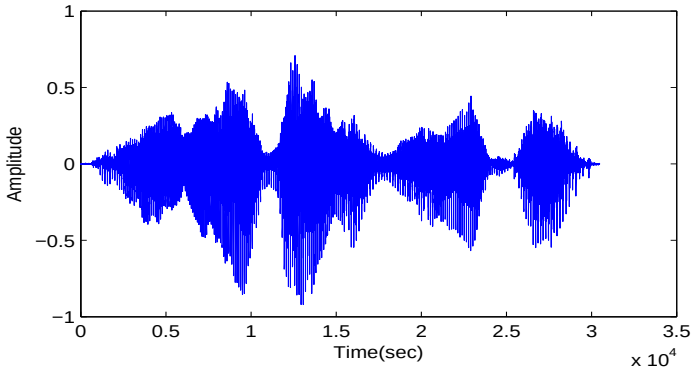
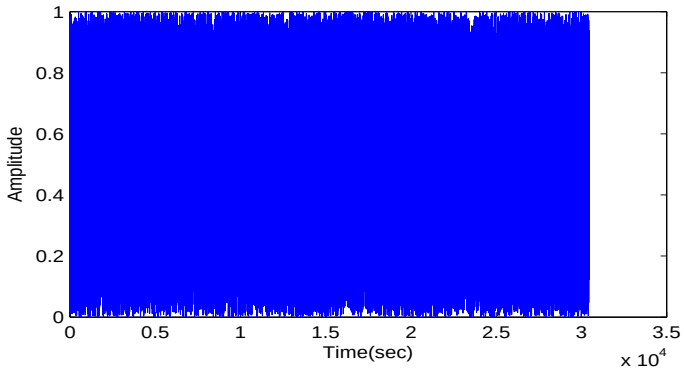


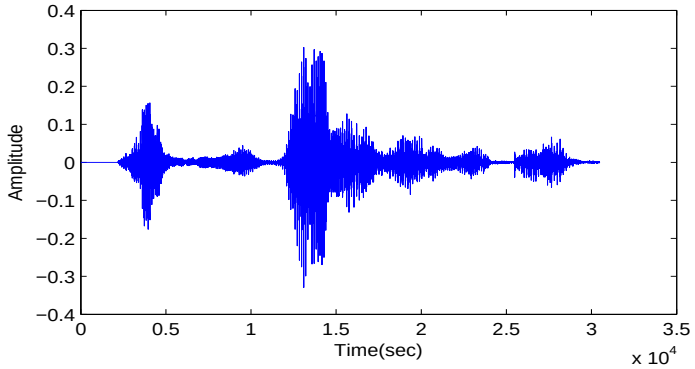
Figure 4.3: Modeled room impulse response



(a) audio secret



(b) 1*st* share



(c) reconstructed reverb effected audio

Figure 4.4: Time domain plots of audio1.wav

round-off errors.

Time domain plots of audio1.wav are represented in Figure 4.4 showing the audio secret, one of its shares and the reverb effected reconstructed secret. The time series reveals that 1) the share is noise and likely to have equal power across all frequencies and 2) the amplitude series of the reverb effected reconstructed secret shows some low amplitude regions as compared to the audio secret. This results from the delay and decay effect of the impulse response shown in Figure 4.3, which verifies that the audio secret has been reverb effected.

4.4 Chapter Summary and Conclusion

The addition of reverberation effects, among others such as flanging and chorus effect, is a vital aspect of audio editing and production in big industries such as music and film production, etc. Such operations are expensive, especially for a client that is constrained in resources. We have proposed in this chapter a secure addition of reverberation effect to an audio secret over cloud by using the (K, N) SSS scheme as our cryptosystem. Our method implements convolution reverb and can be applied to any reverberation impulse response and an audio secret in ED over cloud. Experimental results reveal that our proposed method is efficient and yields similar results to PD with minimal loss.

Chapter 5

Secure Cloud-based Speech Noise Reduction

Apart from storage services provided by CDCs, most clients also make use of computing services. Most importantly, when the need arises for some processing to be done on this encrypted data, the third-party server will first have to decrypt the cipher text which will expose the confidential information. This makes the confidential data vulnerable to exploitation by an adversary. Hence, secure processing of such confidential data is of utmost importance.

A quality-degraded speech secret outsourced to a CDC for storage and processing could be contaminated with different types of noise. The noise could be random, from the circuitry of the recording system or even from power line interference. In this work we propose the implementation of secure digital linear filters using difference equations and impulse responses to enhance the quality of an encrypted speech secret over cloud. We filter out noise from an encrypted quality-degraded speech secret such that decrypting

the ciphertext will produce an enhanced version of the speech secret. We use SSS as the cryptosystem in order to achieve this.

The rest of this chapter is organized as follows. In Section 5.1, we discuss the feasibility of noise reduction techniques and digital filters in ED. The proposed method for secure noise reduction over cloud is presented in Section 5.2 and experimental results are discussed in Section 5.3. We conclude and summarize this chapter in Section 5.4.

5.1 Preliminaries of and Challenges in Noise Reduction ED

5.1.1 White, wind and humming noise

Noise is an unwanted sound which may contaminate a speech signal through the acquisition process (e.g. cockpit voice recorder, hidden recorders of surveillance applications, power line interference, etc.), transmission channel, signal quantization, etc. As a result, the quality and/or intelligibility of the signal is degraded. In general, the main goals of speech enhancement are:

1. to make it pleasant for human perception and to reduce listener fatigue. Noise is disturbing, irritating, annoying and in the case of low frequency noise (infrasound) it might be hazardous to human health [45], [9].

2. to be used as a preprocessing step for speech processing applications such as speech transcription systems, speech/speaker recognition sys-

tems, speech coders, etc., in order to increase accuracy.

There are different types of noise, and the removal technique depends on the nature and frequency characteristics of the noise. In this work, we focus on the attenuation of white, wind and humming noise. We choose these types of noise because: 1) Their frequency characteristics cut across the general spectral nature of noise. White noise consists of high frequencies, wind noise of lower frequencies and humming noise of harmonic frequencies. Our technique can be applied to any noise of similar frequency characteristics. 2) Noise with these characteristics can be attenuated with linear filters (Finite Impulse Response). In our case, linear filters can be implemented in ED with the homomorphic property of SSS (i.e. addition and multiplication). Subtraction and division operations can also be performed in the ED of SSS with intelligent preprocessing techniques. Table 5.1 gives a brief description of the noise types that our proposed scheme attenuates in ED.

Types of noise presented in Table 5.1 are additive, meaning that the noisy signal is a sum of the clean signal and the noise process. That is:

$$x(n) = y(n) + v(n) \tag{5.1}$$

where x is the noisy signal, y is the clean signal and v is the noise. Proposed noise removal/attenuation algorithms in literature either operate in temporal domain (time domain) or some transform domain (Fast Fourier Transform (FFT), Discrete Cosine Transform (DCT), wavelet tranform). They can be classified into: 1) Wiener filtering approach [55], [13]; 2) Spectral subtraction [6], [37]; 3) Statistical model approaches [34]; 4) Subspace techniques [25],

Table 5.1: Noise types

Noise	Characteristics	Reduction technique			
		PD	ED		
				Proposed schemes	
				Difference eqn.	Convolution
White noise	Wide-band frequency characteristics, flat spectral density, statistically uncorrelated with a zero mean normal distribution, overlaps with speech spectral components, random in nature. [***Noise source:*** random process]	Low-pass filter (TD,FD) Wiener filter (TD,FD) Kalman filter (TD,FD) Spectral subtraction (FD) Statistical techniques (TD,FD) Subspace techniques (EGD) etc.	Low pass filter (LPF)	LPF-scheme1	LPF-scheme2
Humming noise	narrow-band frequency characteristics, characterized by a fundamental frequency and its harmonics, tonal in nature. [***Noise source:*** Interference from AC power line (50hz or 60hz)]	Notch filter (TD,FD), Comb filter (TD,FD), Wiener filter (TD,FD), Kalman filter (TD,FD), Spectral subtraction (TD,FD), Statistical techniques (TD,FD), Subspace techniques (EGD) etc.	Comb filter (CF)	CF-scheme1	CF-scheme2
Wind noise	Wide-band frequency characteristics, has higher strengths in selected frequency bands; mostly in the lower frequency band ($< 500Hz$). [***Noise source:*** air fluctuations]	High-pass filter (TD,FD), Wiener filter (TD,FD), Kalman filter (TD,FD), Spectral subtraction(FD), Statistical techniques (TD,FD), Subspace techniques (EGD) etc.	High pass filter (HPF)	HPF-scheme1	HPF-scheme2
Note: TD and FD means that the reduction technique can be performed in Time Domain and Frequency Domain respectively. However, all reduction techniques under Encrypted Domain are performed in time domain. Subspace techniques are performed in Eigen Domain (EGD)					

[15]; and 5) Pass-stop filters (low-pass, high-pass, etc.) [32], [67]. A detailed survey can be obtained from [44], [32], [31], [20]. Noise reduction techniques are not meant to attenuate noise 100%, as significant attenuation results in distortion of the speech signal. Instead, they reduce the noise to a level which makes perception of the actual signal easy and pleasant. After processing, most of these algorithms leave uncorrelated magnitude peaks in the spectrum of the processed signal, which is called residual noise. For instance, spectral

subtraction leaves behind musical tones, so there is often a tradeoff between noise reduction and the quality and/or intelligibility of the signal.

5.1.2 Digital filters

In this work, we propose the implementation of three Finite Impulse Response (FIR) digital filters in ED for the attenuation of the three types of noise stated above: 1) LPF for the attenuation of white noise, 2) CF for the reduction of humming noise and 3) HPF (differentiator) for the reduction of wind noise. We implement these filters in time domain using difference equations and convolution with their impulse responses, as converting to other transform domains such as FFT is complex-valued and not feasible in ED of SSS. Despite the fact that there are implementation techniques for FFT in ED [5], the estimation of the noise spectrum during speech inactive periods is not feasible solely on homomorphism in ED (eg. spectral subtraction and its variants - Power Spectral Subtraction, Magnitude Spectral Subtraction, etc.)

Prior knowledge of the signal and noise statistics makes speech enhancement algorithms easy to implement. However, in real life scenarios, these parameters are not known a priori [20]. Hence, most enhancement algorithms estimate these parameters by employing modeling techniques (Hidden Markov Models (HMM), autoregression (AR), LPC, etc.), Voice Activity detectors (VAD) to learn noise parameters during non-voiced parts of the speech signal, and adaptive techniques to adjust filter coefficients as and when noise parameters are detected or changed. The major challenge here

is that these implementations are nonlinear, involving operations which are not feasible in ED by using homomorphism. For instance, the well known Wiener and Kalman filters, which many of these techniques revolve around, are based on optimization algorithms which converge to a minimal error coefficient and are recursive and iterative. Recursive, adaptive and iterative filtering solely based on homomorphic encryption is not feasible as discussed in [58], [60]. For these reasons, we focus on time domain techniques which can be decomposed into the four basic arithmetic operations (addition, subtraction, multiplication and division) which are feasible in ED based on SSS homomorphic encryption. Table 5.2 illustrates the operations involved in the techniques for the reduction of the noise types stated earlier and their feasibility in ED.

In addition to the feasibility of LPF, CF and HPF in time domain in ED, below are key reasons for the selection of these three filters:

(1) they are linear in nature (Linear Time-Invariant) and can be disintegrated into the four basic arithmetic operations (addition, subtraction, multiplication and division) as illustrated in Table 5.2. These operations are supported by homomorphism as discussed in Section 2.3.2 and hence can be implemented in ED.

(2) they are time domain filters, i.e., the Moving Average (MA) filter and the differentiator are time domain implementations of a simple LPF and HPF respectively.

(3) FIR filters have a finite impulse duration which makes it possible to apply convolution to its impulse response and the noisy signal, unlike Infinite Impulse Response (IIR) filters with infinite impulse duration. Furthermore,

Table 5.2: Filtering and operations involved

Filter	Operations	ED feasible (Yes/No)
Time Domain		
Low-pass filter (MA filter)	Addition, Multiplication, Division	Yes
High-pass filter (Differentiator)	Subtraction	Yes
Notch filter	Addition, Subtraction, Multiplication, Trigonometric Functions	No
Comb filter	Subtraction, Multiplication	Yes
Wiener filter	Addition, subtraction, multiplication, division, comparison, iterative filtering, MSE	No
Kalman filter	Addition, subtraction, multiplication, division, comparison, estimates the state space (AR) model parameters from the noisy speech, MSE	No
Statistical approach techniques	modeling of noise which requires non-linear operations	No
Frequency Domain		
Low-pass filter, High-pass filter, Notch filter, Comb filter, Wiener filter, Kalman filter, Spectral subtraction, Statistical approach techniques	FFT transform (Complex arithmetic)	No
Eigen Domain		
Subspace model-based techniques	Addition, subtraction, multiplication, division, eigen vector decomposition (EVD), optimization criteria, thresholding	No

FIR filter operations have a linear phase property, i.e., the time shifts (delay) performed per sample in each of the filtering operations (LPF, CF and HPF) discussed below are by a constant amount. This results in a linear phase response and there is no phase distortion of the resulting enhanced signal. In the future we hope to examine the feasibility of the other FIR filtering techniques for speech enhancement.

LPF - MA filter

MA filter, also called anti-hiss filter, is a low pass filter which passes low frequencies and attenuates higher frequencies. This filter is mostly used to smoothen the higher frequency portions of a signal in temporal domain. The difference equation for an M-point MA filter is represented below, where $\hat{y}$

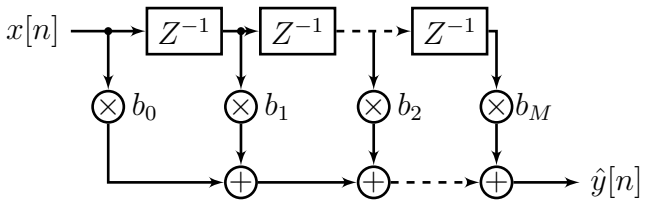


Figure 5.1: MA filter

is the denoised signal sequence, x is the noisy signal sequence and M is the size of the filter:

$$\begin{aligned} \hat{y}[n] &= \sum_{k=0}^{M-1} b_k x[n-k] \quad , b_0 = \cdots = b_{M-1} = \frac{1}{M} \\ &= \frac{1}{M} \sum_{k=0}^{M-1} x[n-k] \end{aligned} \tag{5.2}$$

The signal x is partitioned into frames of size M samples; with $M-1$ overlap and the average of each frame is computed. As can be seen from Figure 5.1, the order of the filter is equal to the number of samples delayed, which is $M-1$. Like any LPF, increasing the order of the MA filter results in a steeper roll-off in the transition band, which produces a sharper cutoff of higher frequencies. It is important to note that larger values of M not only reduce noise but also affect the crispness and distort the speech signal. As a result, the size M of the filter should be chosen in such a way to maintain a balance between noise reduction and signal distortion.

Comb filter

The comb filter attenuates harmonic-like noise (e.g noise from 50Hz or 60Hz power mains). Figure 5.2. shows its digital signal processing (DSP) diagram whose input is the noisy signal x, output is enhanced signal $\hat{y}$, a delay component D and a feedforward coefficient g. The difference equation is:

$$\hat{y}[n] = x[n] - g \times x[n - D] \tag{5.3}$$

where $D = \frac{f_s}{f_o}$, f_s is the sampling frequency of the noisy signal, f_o is the fundamental frequency of humming noise (mostly 60Hz) and g is within $0 < g \leq 1$, which controls the level of noise attenuation. The magnitude response of this filter has the comb effect which results from phase cancellation and reinforcement between the delayed and undelayed signal. Adding a signal with a delayed version of itself where the delay is $\frac{f_s}{f_o}$ results in phase cancellations in time domain which corresponds to the harmonics of f_o in frequency domain. Humming noise not only degrades speech signals but also contaminates Electrocardiography (ECG) signals which might mislead patient diagnosis. Our proposed scheme for secure humming noise removal can also be applied for the reduction of humming noise of encrypted ECG signals in order to protect patient record confidentiality and privacy for delivering cloud based telemedicine services or cloud based patient record storage.

HPF - differentiator

The HPF attenuates lower frequencies and passes higher frequencies. This filter is applicable for attenuating wind noise, and eliminating DC offsets and

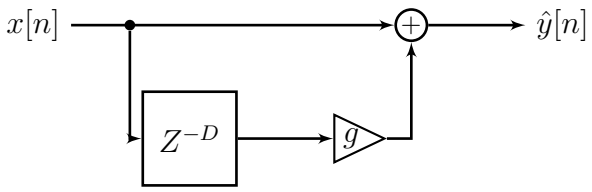


Figure 5.2: Comb filter

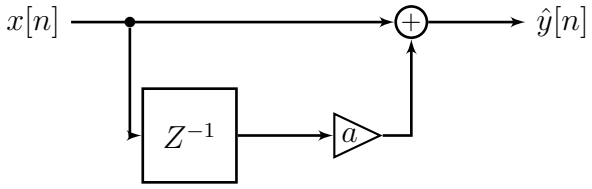


Figure 5.3: HPF filter

microphone pops. We implement a simple HPF, also called a differentiator, as shown in Figure 5.3. Its input is the quality-degraded signal x, and its output is the denoised signal $\hat{y}$ and a delay component of one sample. Its difference equation is shown below:

$$\hat{y}[n] = x[n] - x[n-1] \tag{5.4}$$

Our proposed scheme for HPF of wind noise in ED can also be applied as a pre-emphasis filter in ED to increase the energy of a signal at higher frequencies prior to performing many speech processing applications like speech recognition, Linear Predictive Coding (LPC), etc.

Convolution with impulse response

Another approach to perform the filtering discussed above is to compute the impulse response $h[n]$ of the filters as they are LTI systems. The impulse response is then convolved with the quality-degraded signal to produce the

denoised signal. Impulse response represents the behavior of a system H in response to a unit impulse $\delta[n]$ and is computed by:

$$h[n] = H\{\delta[n]\}; \quad where \quad \delta[n] = \begin{cases} 1, & \text{if } n = 0. \\ 0, & \text{otherwise.} \end{cases} \tag{5.5}$$

The convolution sum to produce the enhanced signal is given by:

$$\hat{y}[n] = x[n] * h[n] = \sum_{k=-\infty}^{\infty} x[k]h[n-k] \tag{5.6}$$

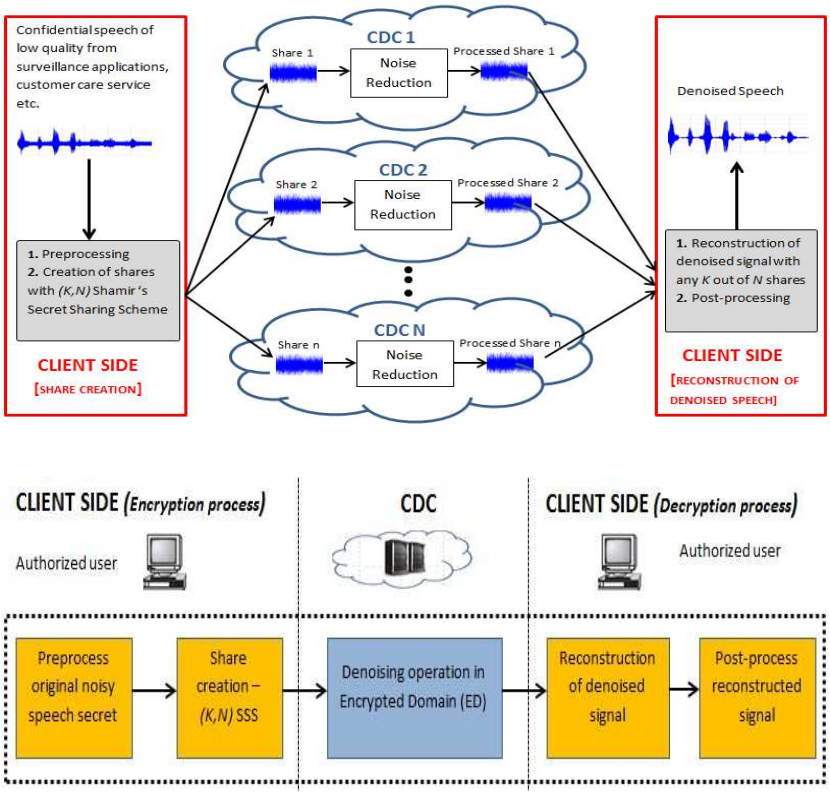


Figure 5.4: The proposed work: Speech noise reduction in ED

5.2 Proposed Method for Speech Noise Reduction in ED

Figure 5.4 represents the proposed work. Shares of the quality-degraded speech signal (contaminated with noise) are created with the (K, N) SSS threshold scheme on the client system. The client then uploads each of N shares to N different non-colluding CDCs. The CDC then performs the noise reduction operation on their hosted shares (that is, processing the encrypted speech signal without knowing the secret). The authorized user then reconstructs the enhanced (denoised) secret by putting at least K out of N processed shares together. We propose the implementation of the difference equation approach and convolution approach (i.e convolution with the impulse response of the digital FIR filter) for each of the filters (LPF, CF and HPF) discussed in Section 5.1.2 in ED. We implement two schemes for each filter.

5.2.1 Preprocessing

We preprocess the speech signal to convert it from real values to positive integer values. Below, we describe the preprocessing steps performed on the original quality-degraded speech signal A prior to creating shares for all three filtering techniques (LPF, CF and HPF).

Step 1: Scale real-valued signal amplitudes with constant factor 10^d where

d is some integer value. Round-off error is bounded by:

$$-\frac{1}{2} \times 10^{1-d} \leq \mathcal{E}_r \leq \frac{1}{2} \times 10^{1-d} \tag{5.7}$$

where $\mathcal{E}_r$ is the rounding error and d is the rounding precision.

Step 2: Shift the signal to the first quadrant by a constant additive shift φ to avoid negative numbers.

$$A' = \big((A + \mathcal{E}_r) \times 10^d\big) + \varphi \tag{5.8}$$

5.2.2 LPF in ED

The steps below describe the operations involved in performing ED Low pass filtering (MA filtering) on shares of the secret speech signal contaminated with white noise.

LPF-scheme 1: Difference equation approach.

Step 1: Preprocess signal A' from Equation (5.8) to multiples of the size of the MA filter M from Equation (5.2). This is done to make division as a result of the averaging operation in Equation (5.2) possible in ED of the cryptosystem (SSS).

$$A'' = A' \times M \tag{5.9}$$

Step 2: Create N shares $(S_1, S_2 \ldots S_N \in S)$ of A'' under $GF(q_{LPF})$ where $q_{LPF} > max(A'')$ using Equation (2.1) of SSS and upload shares to N CDCs.

Step 3: Low pass filter each share over CDC. Apply Equation (5.10) to each share $(S_1, S_2 \dots S_N)$ to obtain processed shares $(S'_{1^{LPF}}, S'_{2^{LPF}} \dots S'_{N^{LPF}} \in S'_{LPF})$

$$S'_{i^{LPF}}[n] = \left(\frac{1}{M} \sum_{k=0}^{M-1} S_i[n-k] \right) mod q_{LPF} \quad , i \in \{1, 2 \dots N\} \tag{5.10}$$

Step 4: The authorized user downloads K processed shares from any K out of N CDCs, and uses Lagrange interpolation from Equation (2.2) to reconstruct the denoised version of the secret.

Step 5: Postprocess to reverse the preprocessing done in the above steps. First, divide the reconstructed denoised secret by the size of the MA filter M, then subtract the additive shift φ and finally divide by the scaling factor 10^d from Section (5.2.1) step 1.

LPF-scheme 2: Convolution approach.

Step 1: Preprocess signal A' to obtain A'' using Equation (5.9) as performed in step 1 of LPF-scheme 1.

Step 2: Create N shares $(S_1, S_2 \dots S_N \in S)$ of A'' using Equation (2.1) of SSS in the finite field of q'_{LPF} and upload shares to N CDCs. As discussed in Section (5.2.5), the modular prime chosen should be more than the maximum sample of A'' multiplied by the sum of absolute values of the impulse response sequence. From Equation (5.17), $q'_{LPF} > max(A'') \times \sum_{n=0}^{I_{LPF}-1} |h_{LPF}[n]|$, where $h_{LPF}[n]$ is the impulse response of the MA filter computed using Equation (5.5).

Step 3: Convolve the impulse response $h_{LPF}[n]$ with each share on the CDC to produce processed shares $(S''_{1^{LPF}}, S''_{2^{LPF}} \dots S''_{N^{LPF}} \in S''_{LPF})$

$$\begin{aligned} S''_{i^{LPF}} &= \Big(S_i[n] * h_{LPF\_filter}[n] \Big) mod q'_{LPF} \\ &= \Big(\sum_{k=0}^{L-1} S_i[k] h_{LPF\_filter}[n-k] \Big) mod q'_{LPF} \\ & i \in \{1, 2 \dots N\} \end{aligned} \tag{5.11}$$

where L is the number of samples of each share.

Step 4: The authorized user puts together K processed shares from any K out of N CDCs, then uses Lagrange interpolation from Equation (2.2) to reconstruct the denoised version of the secret.

Step 5: Postprocess to reverse the preprocessing done in the above steps. First, divide the reconstructed denoised secret by the size of the MA filter M, then subtract the additive shift φ and finally divide by the scaling factor 10^d from Section (5.2.1) step 1.

5.2.3 CF in ED

Comb filtering in ED is detailed below for the enhancement of a speech secret degraded with humming noise.

CF-scheme 1: Difference equation approach.

Step 1: Create N shares $(S_1, S_2 \dots S_N \in S)$ of preprocessed signal A' from Equation (5.8) using Equation (2.1) of SSS under $GF(q_{CF})$, where q_{CF} is the

first prime more than $max(A')$.

Step 2: Comb filter each share. Comb filtering in ED with Equation (5.3) might result in errors due to the negative real-valued feedforward gain g. In order to avoid errors, we modify Equation (5.3) by performing the below scaling and additive shift in ED.

1. Scaling real-valued g: The real-valued feedforward coefficient $0 < g \leq 1$ might result in real numbers during processing on encrypted shares. As g is a one decimal place number between 0 and 1, scaling it with 10 will eliminate real numbers, i.e., $g' = 10g$, where $0 < g' \leq 10$, which is also the same as multiplying Equation (5.3) by 10. The value of g' determines the level of noise attenuation by the filter (i.e. 1 means 10% attenuation, 2 means 20% attenuation, 3 means 30% attenuation ... 10 means 100% attenuation).

$$\begin{aligned} S'_{CF}[n] &= 10\Big(S[n] - g \times S[n-D]\Big) \\ S'_{CF}[n] &= 10 \times S[n] - g' \times S[n-D] \end{aligned} \tag{5.12}$$

2. Avoiding negative numbers: Add a constant additive shift ϑ to Equation (5.12) which is equal to 10^d (signal scaling factor from step 1 Section (5.2.1) multiplied by 10 (scaling factor from above), i.e. $\vartheta = 10^{d+1}$

$$S'_{i^{CF}}[n] = \left(\left(10 \times S_i[n] - g' \times S_i[n-D]\right) + \vartheta\right) mod q_{CF} \qquad (5.13)$$

$$i \in \{1, 2 \ldots N\}$$

Apply Equation (5.13) to comb filter each share to obtain processed shares $(S'_{1^{CF}}, S'_{2^{CF}} \ldots S'_{N^{CF}} \in S'_{CF})$

Step 3: The authorized user can reconstruct the enhanced secret by putting together K processed shares from any of the N CDCs using Lagrange interpolation from Equation (2.2).

Step 4: Postprocess the reconstructed enhanced secret to reverse-engineer preprocessing done in the above steps.

1. Subtract additive shift ϑ and divide by the scale factor of 10 from Section (5.2.3) step 2.

2. Subtract additive shift φ and divide by the scale factor of 10^d from Section (5.2.1) step 2 and 1 respectively.

CF-scheme 2: Convolution approach.

Step 1: Create N shares $(S_1, S_2 \ldots S_N \in S)$ of the preprocessed signal A' in finite field of q'_{CF} using Equation (2.1) of SSS. In order to address the cipher blow-up problem as a result of convolution in ED as discussed in Section 5.2.5, we determine the modular prime using Equation (5.17), i.e. $q'_{CF} > max(A') \times \sum_{n=0}^{I_{CF}-1} |h'_{CF}[n]|$, where h'_{CF} is the scaled impulse response h_{CF} of the comb filter, which will be explained in the next step.

Step 2: Convolution of each share on CDC. Due to the negative real-valued feedforward gain g of the comb filter, $h_{CF} = [1, \ldots, -g]$ has a negative real-valued last sample. We perform the following in ED to avoid negative and real-valued samples. 1) Since g is a one decimal place number between 0 and 1, we scale h_{CF} by 10, i.e. $h'_{CF} = 10 \times h_{CF}$ and 2) we add a constant additive shift ϑ' to avoid negative samples, i.e. $\vartheta' = \lceil \frac{q'_{CF}}{2} \rceil$, where $\lceil . \rceil$ is a ceiling function. Convolve each share $(S_1, S_2 \ldots S_N \in S)$ with h'_{CF} using Equation (5.14) to obtain $(S''_{1^{CF}}, S''_{2^{CF}} \ldots S''_{N^{CF}} \in S''_{CF})$

$$\begin{aligned} S''_{i^{CF}}[n] &= \left(\left(S_i[n] * h'_{CF}[n] \right) + \vartheta' \right) mod q'_{CF} \\ &= \left(\left(\sum_{k=0}^{L-1} S_i[k] h'_{CF}[n-k] \right) + \vartheta' \right) mod q'_{CF} \\ & i \in \{1, 2 \ldots N\} \end{aligned} \tag{5.14}$$

Step 3: The authorized user can reconstruct the enhanced secret by putting together K processed shares from any of the N CDCs using Lagrange interpolation from Equation (2.2).

Step 4: Postprocess the reconstructed enhanced secret to reverse-engineer the preprocessing done in the above steps. 1) Subtract additive shift ϑ' and divide by the scale factor of h'_{CF}, which is 10. 2)Subtract additive shift φ and divide by the scale factor of 10^d from Section (5.2.1) step 2 and 1 respectively.

5.2.4 HPF in ED

Removal of wind noise in ED with HPF is detailed below.

HPF-scheme 1: Difference equation approach.

Step 1: Create N shares $(S_1, S_2 \dots S_N \in S)$ of A' using Equation (2.1) of SSS. Shares are created under $GF(q_{HPF})$; where q_{HPF} is the first prime more than the maximum sample a'_{max} of A'.

Step 2: High pass filter each share. Before high pass filtering, add a constant additive shift λ to Equation (5.4) to avoid negative numbers. Additive shift is equal to the scaling factor 10^d from step 1 Section (5.2.1), i.e. $\lambda = 10^d$. Filter each share with Equation (5.15) to obtain processed shares $(S'_{1^{HPF}}, S'_{2^{HPF}} \dots S'_{N^{HPF}} \in S'_{HPF})$

$$S'_{i^{HPF}}[n] = \left(\left(S_i[n] - S_i[n-1]\right) + \lambda\right) mod q_{HPF} \qquad i \in \{1, 2 \dots N\} \tag{5.15}$$

Step 3: Reconstruction of the denoised secret by the authorized user with any K processed shares using Lagrange interpolation from Equation (2.2).

Step 4: Postprocess the reconstructed denoised secret. Subtract the sum of additive shifts (i.e. $\lambda + \varphi$) and divide by the scale factor of 10^d from Section (5.2.1) step 1.

HPF-scheme 2: Convolution approach.

Step 1: Create N shares $(S_1, S_2 \dots S_N \in S)$ of A' using Equation (2.1)

of SSS under $GF(q'_{HPF})$ and upload them to N CDCs. $q'_{HPF} > a'_{max} \times \sum_{n=0}^{I_{HPF}-1} |h_{HPF}[n]|$, where $h_{HPF}[n]$ is the impulse response of HPF computed from Equation (5.5) and a'_{max} is the maximum sample of A'.

Step 2: On the CDC, convolve each share with $h_{HPF}[n]$ to obtain processed shares $(S''_{1^{HPF}}, S''_{2^{HPF}} \ldots S''_{N^{HPF}} \in S''_{HPF})$. The computed impulse response of HPF is $h_{HPF} = [1, -1]$. Due to the negative second sample of h_{HPF}, convolution in ED might result in negative values, so we add an additive shift λ' to avoid using negative integers in modular domain. $\lambda' = \lceil \frac{q'_{HPF}}{2} \rceil$, where $\lceil . \rceil$ is a ceiling function.

$$
\begin{aligned}
S''_{i^{HPF}}[n] &= \left(\left(S_i[n] * h_{HPF}[n] \right) + \lambda' \right) mod q'_{HPF} \\
&= \left(\left(\sum_{k=0}^{L-1} S_i[k] h_{HPF}[n-k] \right) + \lambda' \right) mod q'_{HPF} \\
& \quad i \in \{1, 2 \ldots N\}
\end{aligned}
\tag{5.16}
$$

Step 3: Reconstruction of the denoised secret by the authorized user with any K processed shares using Lagrange interpolation from Equation (2.2).

Step 4: Postprocess the reconstructed denoised secret by 1) subtracting the sum of additive shifts $(\lambda' + \varphi)$ and 2) dividing by the scale factor of 10^d from Section (5.2.1) step 1.

5.2.5 Implementation challenges of convolution in ED

The implementation of convolution in ED requires us to address the problem of cipher blow-up with homomorphic computation[60],[59],[16]. Homomor-

phic operations performed in ED are in some finite field (modular domain). However, signal processing operations on plaintext are not in modular domain. This means that computations in ED, when applied to PD, should not overflow the message space as a result of the modular operation. Discrete convolution sum is comprised of a series of multiplication and addition operations and sometimes results in cipher blow-ups depending on the numbers and weights of the samples of the impulse response. This causes operations to overflow (wrap-around) the modular space several times which disconnects the homomorphic mapping between ED and PD.

To simplify this explanation, assume that the preprocessed samples of some 8 bit real-valued signal $X = \{x_1, x_2, x_3\}$ after scaling to integer $\hat{X} = \{56, 162, 98\}$, are processed in both PD and ED with an FIR system $y[n] = x[n] + x[n-1]$ whose impulse response is $h = [1, 1]$. Creating shares with the $(2, 3)$ SSS scheme under modulus prime 257 ($q = 257$; i.e. the first prime greater than the maximum sample of an 8 bit signal $\hat{X}$) produces $S_1 = \{155, 4, 197\}, S_2 = \{254, 103, 39\}$ and $S_3 = \{96, 202, 138\}$ as share 1, 2 and 3 respectively.

i. PD convolution: convolution of $\hat{X}$ with $h = [1, 1]$ will give $\{218, 260, 98\}$

ii. ED convolution under $GF(257)$: Convolving each share (S_1, S_2 and S_3) with $h = [1, 1]$ and reconstructing will produce $\{218, 3, 98\}$

It is evident that the results of *i* and *ii* above are not the same. The mapping between the result of *ii* and its real-valued version with a scale factor is lost. This is due to the fact that computation overflows the modular space $GF(257)$. To solve this problem, the modulus prime should be chosen

large enough to accommodate overflows within the modular domain. For the convolution operation, we propose that the modular prime should be greater than the multiple of the maximum sample of the preprocessed signal and the sum of absolute values of the impulse response. That is:

$$q > x_{max} \times \sum_{n=0}^{I-1} |h[n]| \tag{5.17}$$

where x_{max} is the maximum sample of the scaled signal $\hat{X}$, and I is the number of samples of the impulse response. With this in mind, we compute the prime with Equation (5.17) to get 331 and repeat the above example under $GF(331)$ to obtain $\{218, 260, 98\}$ in PD and $\{218, 260, 98\}$ in ED after reconstruction, which are identical. Though this will produce large primes with increasing numbers of non-zero samples of the impulse response, it is practical for FIR systems with less overhead since they are non-recursive with finite impulse duration. For instance, the above example incurred an overhead of less than 1 bit (i.e. $\log_2(331) - \log_2(257)$)

5.2.6 Data overhead

In our proposed ED noise reduction schemes, there are some data overheads incurred to transmit a share from the client to the CDC as a result of data expansion caused by the preprocessing steps. This data overhead is the number of bits used to represent the maximum preprocessed speech sample. Since shares are generated under a finite field bounded by a modulo prime number, we can conclude that the data overhead is also bounded by the number of bits used to represent this prime number.

Overhead for LPF-scheme 1

Preprocessing before creating shares for low pass filtering (scheme 1) involves Equations (5.7),(5.8) and (5.9). Shares of preprocessed signal A'' are created bounded by q_{LPF}. If b_{LPF} is the number of bits representing q_{LPF} then the overhead is:

$$b_{LPF} = \log_2(q_{LPF}) \tag{5.18}$$

Now we examine how the expansion of the message space of the original signal A due to preprocessing relates to data overhead and transmission. From Equations (5.7),(5.8) and (5.9), the relationship between a_{max} (maximum sample of the original secret signal A) and a''_{max} (maximum sample of preprocessed signal A'') can be expressed as:

$a''_{max} = \big((a_{max} \times 10^d) + \mu\big) \times M$; where $\mu = \big(\mathcal{E}_r \times 10^d\big) + \varphi$ and consequently, $q_{LPF} > a''_{max}$. Rewriting Equation (5.18) in terms of amplitude gives:

$$b_{LPF} > \log_2\left(\left((a_{max} \times 10^d) + \mu\right) \times M\right) \tag{5.19}$$

Overhead for LPF-scheme 2

Convolution in ED requires the selection of a modular prime such that the cipher blow-up problem highlighted in Section 5.2.5 does not occur. The prime q'_{LPF} chosen bounds the message space. If b'_{LPF} is the overhead in bits then:

$$b'_{LPF} = \log_2(q'_{LPF}) \tag{5.20}$$

From LPF-scheme 2 step 2,
$q'_{LPF} > a''_{max} \times \sum_{n=0}^{I_{LPF}-1} |h_{LPF}[n]|$ so:

$$\begin{aligned} b'_{LPF} &> \log_2\left(a''_{max} \times \sum_{n=0}^{I_{LPF}-1} |h_{LPF}[n]|\right) \\ b'_{LPF} &> \log_2\left[\left(\left((a_{max} \times 10^d) + \mu\right) \times M\right) \times \sum_{n=0}^{I_{LPF}-1} |h_{LPF}[n]|\right] \end{aligned} \tag{5.21}$$

Overhead for CF-scheme 1

Shares of preprocessed signal A' are created for this scheme under $GF(q_{CF})$. Let b_{CF} be the number of bits of q_{CF} then overhead is:

$$b_{CF} = \log_2(q_{CF}) \tag{5.22}$$

From Equation (5.7) and (5.8), $q_{CF} > (a_{max} \times 10^d) + \mu$ where $\mu = \left(\mathcal{E}_r \times 10^d\right) + \varphi$ and consequently:

$$b_{CF} > \log_2\left((a_{max} \times 10^d) + \mu\right) \tag{5.23}$$

Overhead for CF-scheme 2

Let b'_{CF} be the data overhead in bits for comb filtering scheme 2. Shares of the preprocessed signal A' are created in the finite field of q'_{CF}. Then, the overhead is represented by:

$$b'_{CF} = \log_2(q'_{CF}) \tag{5.24}$$

$q'_{CF} > a'_{max} \times \sum_{n=0}^{I_{CF}-1} |h'_{CF}[n]|$ from CF-scheme 2 step 1. Expressing b'_{CF} in terms of amplitude will give:

$$b'_{CF} > \log_2\left[\left((a_{max} \times 10^d) + \mu\right) \times \sum_{n=0}^{I_{CF}-1} |h'_{CF}[n]|\right] \tag{5.25}$$

Overhead for HPF-scheme 1

For HPF, preprocessing involving Equations (5.7) and (5.8) produces A' and shares are bounded by q_{HPF}. If b_{HPF} is the overhead for this scheme then it is given by:

$$b_{HPF} = \log_2(q_{HPF}) \tag{5.26}$$

$q_{HPF} > (a_{max} \times 10^d) + \mu$ so:

$$b_{HPF} > \log_2\left((a_{max} \times 10^d) + \mu\right) \tag{5.27}$$

Overhead for HPF-scheme 2

The convolution approach of HPF is bounded by q'_{HPF}. If the bit representation of q'_{HPF} is b'_{HPF}, then the overhead is:

$$b'_{HPF} = \log_2(q'_{HPF}) \tag{5.28}$$

From step 1 of HPF-scheme 2, $q'_{HPF} > a'_{max} \times \sum_{n=0}^{I_{HPF}-1} |h_{HPF}[n]|$, and rewriting equation (5.28) gives:

$$b'_{CF} > \log_2\left[\left((a_{max} \times 10^d) + \mu\right) \times \sum_{n=0}^{I_{HPF}-1} |h_{HPF}[n]|\right] \tag{5.29}$$

For all the difference equation approaches of all the filters (LPF-scheme 1, CF-scheme 1 and HPF-scheme 1), it is evident that from Equations (5.19), (5.23) and (5.27), 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. From Equation (5.19), it is clear that LPF-scheme 1 and 2 data overhead also depends on the size M of the filter. The greater the size, the greater the overhead to transmit a share to CDC. The overhead for the convolution approaches (LPF-scheme 2, CF-scheme 2 and HPF-scheme 2), as shown in Equations (5.21), (5.25) and (5.29), depends on the maximum sample of the preprocessed signal and the weights and number of non-zero samples of the impulse response sequence. The higher these values are, the higher the bandwidth resources required to transmit each share to CDC. Thus, an impulse response with higher weights and many non-zero elements will require a larger message space (modulus prime) in order for convolution in ED to maintain mapping to their plaintext versions with a scaled factor, without the cipher blow-up problems stated in Section 5.2.5. FIR filters have finite impulse duration and the responses of the FIR filters implemented in this work have small weights and finitely few samples. For example, the impulse response of an M point MA filter has weights of $\frac{1}{M}$ and a length of M, and the comb filter and the HPF have two non-zero samples with weights of 1.

5.2.7 Security analysis

The proposed method is based on the (K, N) SSS threshold scheme which is proven to be information theoretically secure [56]. However, our scheme preprocesses the speech signal before generating shares so it is imperative to examine the impact on information theoretic security. We evaluate the security of our proposed method using the below corollaries based on the theorems in [30].

Corollary 3 *LPF-scheme 1 is information theoretically secure if the probability of revealing a speech secret sample a_0 shared under $GF(q)$ is the same as the probability of determining $a_0'' = \big((a_0 \times 10^d) + \mu\big) \times M$ shared under $GF(q_{LPF})$ (where $\mu = \big(\mathcal{E}_r \times 10^d\big) + \varphi$ from Equation (5.8) and q_{LPF} is a prime number greater than $\big((q \times 10^d) + \mu\big) \times M$).*

Proof 6 For each plaintext of speech 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:

$$Pr(a_0)_{0 \leq a_0 \leq q-1} = \frac{1}{q} \tag{5.30}$$

Similarly, for each plaintext a_0'' of the preprocessed speech secret A'' where $a_0'' = \big((a_0 \times 10^d) + \mu\big) \times M$, there is also an equal probability of being any value from the set $\{0, \big(10^d + \mu\big) \times M, \big((2 \times 10^d) + \mu\big) \times M, \big((3 \times 10^d) + \mu\big) \times M, ..., q_{LPF} - 1\}$, that is $0 \leq a_0'' \leq q_{LPF} - 1$ of q values with the probability given as:

$$Pr(a_0'')_{0 \leq a_0'' \leq q_{LPF}-1} = \frac{1}{q} \tag{5.31}$$

Corollary 4 *LPF-scheme 2 is information theoretically secure if the probabilities of revealing a speech secret sample and a preprocessed sample are the same.*

Proof 7 Proof is the same as Corollary 3 where the preprocessed sample is shared under $GF(q'_{LPF})$ with $q'_{LPF} > \big[\,\big((q \times 10^d) + \mu\big) \times M\big] \times \sum_{n=0}^{I_{LPF}-1} |h_{LPF}[n]|$.

Corollary 5 *CF-scheme 1 is information theoretically secure.*

Proof 8 Proof is the same as Corollary 3 with preprocessed speech secret sample $a'_0 = (a_0 \times 10^d) + \mu$ shared under $GF(q_{CF})$; $q_{CF} > (q \times 10^d) + \mu$.

Corollary 6 *CF-scheme 2 is information theoretically secure.*

Proof 9 Proof is the same as Corollary 3 with preprocessed speech secret sample $a'_0 = (a_0 \times 10^d) + \mu$ shared under $GF(q'_{CF})$; $q'_{CF} > \big[(q \times 10^d) + \mu\big] \times \sum_{n=0}^{I_{CF}-1} |h'_{CF}[n]|$.

Corollary 7 *HPF-scheme 1 is information theoretically secure.*

Proof 10 Proof is the same as Corollary 3. Preprocessed speech secret sample shared under $GF(q_{HPF})$; $q_{HPF} > (q \times 10^d) + \mu$.

Corollary 8 *HPF-scheme 2 is information theoretically secure.*

Proof 11 Proof is the same as Corollary 3 where preprocessed speech secret sample is shared under $GF(q'_{HPF})$; $q'_{HPF} > \big[(q \times 10^d) + \mu\big] \times \sum_{n=0}^{I_{HPF}-1} |h_{HPF}[n]|$.

In all cases above, the probability of revealing the secret sample a_0 and the preprocessed samples a'_0 or a''_0 is $\frac{1}{q}$. Thus, our scheme preserves information theoretic security after preprocessing the original secret speech signal. An adversary in both cases will have to guess the secret with a probability of $\frac{1}{q}$.

Corollary 9 *Our proposed schemes in ED (LPF-scheme 1 and 2, CF-scheme 1 and 2, and HPF-scheme 1 and 2) are* $100(1 - r_{(ED,PD)})$*-accurate implementations of their plaintext versions with negligible loss in accuracy, where* $0 \leq r_{(ED,PD)} \leq 1$ *is the similarity coefficient between ED and PD processing.*

Proof 12 Proof follows the same lines as theorem 1.

Corollary 10 *In the malicious model, our proposed schemes in ED (LPF-scheme 1 and 2, CF-scheme 1 and 2, and HPF-scheme 1 and 2) are 1-checkable implementations of their corresponding computations over cloud.*

Proof 13 Proof follows the same lines as theorem 2.

Remark 2 *The computational complexity for checkability by the client is linear to the size* L *of the reconstructed denoised signal, i.e.* $O(L)$*. This might cause additional overhead for the client and can therefore be performed offline.*

5.3 Experimental Results

We selected a set of 35 clean speech paragraphs from the Language Technologies Institute at CMU (Carnegie Mellon University) database [61]. The speech files are sampled at 16kHz 16bit. We digitally added white noise, humming noise (60Hz harmonics) [2] and wind noise [17] to the 35 clean speech files at 7 global SNR levels of −15dB,−10dB,−5dB, 0dB, 5dB, 10dB and 15dB to obtain 35 noisy speech secret files per noise type (ie. a total of 105 noise corrupted speech signals). We used these 105 noisy speech signals

(35 white noise, 35 humming noise and 35 wind noise corrupted speech signals) to test and evaluate our proposed method. We perform the denoising operations in ED for our methods and then in PD. We later evaluate the ED denoised signal with respect to the PD denoised signal and the clean reference signal. Figure 5.5 illustrates our experimental setup.

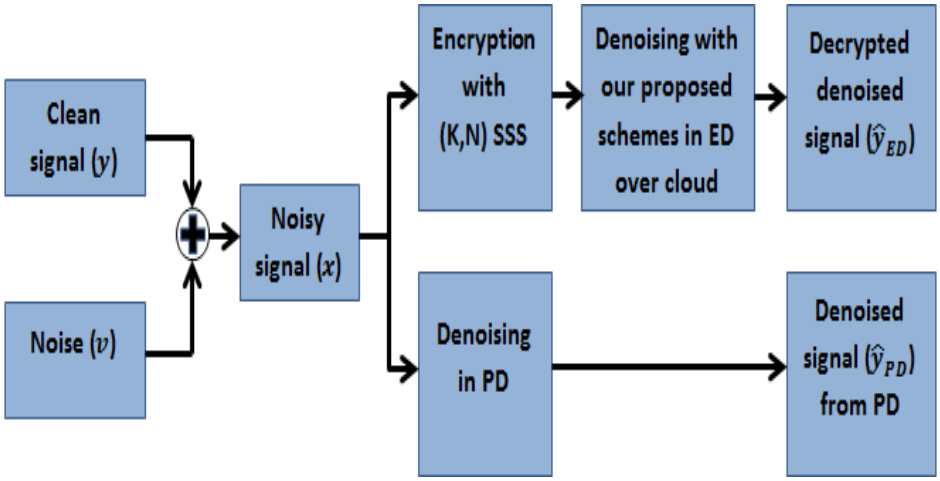


Figure 5.5: Experimental setup

We implemented the proposed method using MATLAB14 on a 2.53GHz i5 CPU with 4GB RAM. In the (K, N) threshold SSS scheme, we set $K = 2$ and $N = 3$, implying that we created three shares of the quality-degraded speech secret such that at least two shares will be required to reconstruct the enhanced speech secret.

The experiments performed in this section and their goals are:

(i) Objective quality measurement:

(a) Segmental signal-to-noise ratio (segSNR): We employ segSNR to evaluate the quality of the denoised speech signal in ED with respect to the clean reference signal. The same is done for the PD

denoised signal. segSNR provides a higher correlation with subjective ratings than the classical SNR and it is one of the most effective metrics in evaluating speech noise reduction algorithms [29],[35],[23].

(b) Perceptual Evaluation of Speech Quality (PESQ): We evaluate the distortion measure of the denoised signal in ED with reference to the clean signal. PESQ [50] is an international standard widely used for estimating speech quality. It was developed by the International Telecommunication Union (ITU).

(c) Pearson's correlation: We evaluate the similarity between the denoised speech signal in ED and in PD. Alternatively, this test is similar to computing the Mean Square Error(MSE) between the denoised speech signal in ED and in PD.

(ii) Subjective quality measurement: A listening test survey was conducted to evaluate the quality similarity of the denoised speech signal in ED and in PD.

(iii) Computational complexity: We evaluate how many computational operations are performed on both the client and the CDC side.

5.3.1 Objective quality measurement

Segmental signal-to-noise ratio (segSNR)

We use segSNR to evaluate the quality of the denoised reconstructed speech from our proposed method in ED and the percentage loss in segSNR with

respect to PD filtering operations. segSNR was measured by averaging the frame level SNR per signal using Equation (5.32).

$$segSNR = \frac{10}{J}\sum_{j=0}^{J-1}\log_{10}\frac{\sum_{n=Tj}^{Tj+T-1} y[n]^2}{\sum_{n=Tj}^{Tj+T-1}(y[n]-\hat{y}[n])^2} \tag{5.32}$$

where y is the clean signal (reference signal), $\hat{y}$ is the enhanced signal, T is the frame length (we choose $20ms$) and J is the number of frames in the signal. Signal energy during silent intervals of a speech signal may cause biasing to the measurement of segSNR [29], [35], [23]. We remedy this by using a P.56-based VAD (voice activity detector) to exclude the silent frames of the signal from the computation of segSNR (5.32). We obtained the implementation source code of the segSNR measurement from [40].

Table 5.3: segSNR for white, wind noise reduction

White noise							
SNR(dB) noisy speech	segSNR(dB) noisy speech	LPF-scheme1			LPF-scheme2		
		denoised in ED	denoised in PD	% Loss in segSNR	denoised in ED	denoised in PD	% Loss in segSNR
-15	-26.603074	-17.287079	-17.287012	0.000387	-17.286468	-17.287012	0.00315
-10	-19.883201	-10.982527	-10.982503	0.000215	-10.982241	-10.982503	0.002391
-5	-14.333825	-6.361027	-6.360992	0.000543	-6.360759	-6.360992	0.003663
0	-10.067725	-3.255279	-3.255331	0.001587	-3.255185	-3.255331	0.004483
5	-4.078657	0.629071	0.628893	0.028254	0.628833	0.628893	0.009446
10	-0.518699	2.462141	2.462831	0.028009	2.46114	2.462831	0.068667
15	5.067411	4.669561	4.669828	0.005732	4.665243	4.669828	0.098188
Wind noise							
SNR(dB) noisy speech	segSNR(dB) noisy speech	HPF-scheme1			HPF-scheme2		
		denoised in ED	denoised in PD	% Loss in segSNR	denoised in ED	denoised in PD	% Loss in segSNR
-15	-24.284081	-4.651359	-4.651354	0.000111	-4.654222	-4.643519	0.230485
-10	-17.46624	-2.161524	-2.161478	0.002165	-2.166288	-2.15826	0.371955
-5	-12.129808	-0.914804	-0.914705	0.010843	-0.920766	-0.912698	0.883978
0	-7.703269	-0.137905	-0.137799	0.076895	-0.13617	-0.136017	0.112288
5	-1.831293	0.178706	0.178841	0.075663	0.175428	0.178967	1.977176
10	1.941045	0.18719	0.187331	0.075486	0.183301	0.187298	2.133943
15	7.533529	0.32759	0.327743	0.046702	0.31184	0.327734	4.849494

Tables 5.3 and 5.4 present the segSNR averaged over signals per global

Table 5.4: segSNR for humming noise reduction

g=0.2							
SNR(dB) noisy speech	segSNR(dB) noisy speech	CF-scheme1			CF-scheme2		
		denoised in ED	denoised in PD	% Loss in segSNR	denoised in ED	denoised in PD	% Loss in segSNR
-15	-26.789155	-24.891482	-24.891496	0.000057	-24.892079	-24.892094	0.000058
-10	-19.900499	-18.004854	-18.004874	0.000114	-18.005024	-18.005045	0.000114
-5	-14.344659	-12.424354	-12.424385	0.000254	-12.424816	-12.424848	0.000255
0	-10.055748	-8.363831	-8.363574	0.003077	-8.363858	-8.3636	0.003082
5	-4.081147	-2.772173	-2.772037	0.004913	-2.771655	-2.771523	0.004769
10	-0.534559	-0.105765	-0.105028	0.702441	-0.105102	-0.104357	0.713137
15	5.052778	4.048975	4.051667	0.066431	4.048216	4.050895	0.066127
g=0.5							
SNR(dB) noisy speech	segSNR(dB) noisy speech	CF-scheme1			CF-scheme2		
		denoised in ED	denoised in PD	% Loss in segSNR	denoised in ED	denoised in PD	% Loss in segSNR
-15	-26.789155	-20.969356	-20.969364	0.000037	-20.969971	-20.969979	0.000039
-10	-19.900499	-14.290356	-14.290355	0.000007	-14.290599	-14.290596	0.000021
-5	-14.344659	-9.253574	-9.253547	0.000287	-9.254081	-9.254058	0.00025
0	-10.055748	-6.076147	-6.075712	0.007162	-6.076219	-6.075781	0.00721
5	-4.081147	-1.989504	-1.989222	0.014186	-1.989012	-1.988735	0.013926
10	-0.534559	-0.556309	-0.554837	0.265251	-0.555794	-0.554317	0.266441
15	5.052778	2.105467	2.109487	0.190577	2.10437	2.1084	0.191157
g=0.8							
SNR(dB) noisy speech	segSNR(dB) noisy speech	CF-scheme1			CF-scheme2		
		denoised in ED	denoised in PD	% Loss in segSNR	denoised in ED	denoised in PD	% Loss in segSNR
-15	-26.789155	-14.515005	-14.514963	0.000285	-14.515667	-14.515624	0.000294
-10	-19.900499	-8.979682	-8.979531	0.001685	-8.980078	-8.979923	0.001724
-5	-14.344659	-5.474085	-5.473757	0.00599	-5.474839	-5.474518	0.005856
0	-10.055748	-3.452074	-3.450484	0.046104	-3.452613	-3.451017	0.046252
5	-4.081147	-1.103423	-1.101352	0.188093	-1.103358	-1.101282	0.188466
10	-0.534559	-0.129311	-0.123949	4.325916	-0.128999	-0.123626	4.346415
15	5.052778	1.102041	1.111948	0.890938	1.100711	1.110647	0.894576
g=1							
SNR(dB) noisy speech	segSNR(dB) noisy speech	CF-scheme1			CF-scheme2		
		denoised in ED	denoised in PD	% Loss in segSNR	denoised in ED	denoised in PD	% Loss in segSNR
-15	-26.789155	-9.334196	-9.333807	0.004172	-9.33494	-9.334552	0.004149
-10	-19.900499	-5.472685	-5.471761	0.016869	-5.472775	-5.471851	0.016882
-5	-14.344659	-3.149341	-3.147692	0.052401	-3.152706	-3.151098	0.051006
0	-10.055748	-1.85155	-1.845075	0.350943	-1.852237	-1.845771	0.350286
5	-4.081147	-0.807828	-0.800418	0.925796	-0.806939	-0.799499	0.930626
10	-0.534559	-0.355181	-0.345201	2.891109	-0.35443	-0.344465	2.8931
15	5.052778	-0.081436	-0.081464	0.034719	-0.08299	-0.082119	1.05962

SNR level for ED schemes and their corresponding PD filtering operations. Tables 5.3 and 5.4 reveal that:

(i) For all ED proposed schemes, it is evident that: (1) the segSNR decreases as the global SNR increases from -15dB to 15dB. This is as a result of increasing signal distortion with respect to noise suppression as the SNR value of the noisy speech signal increases, and (2) at lower SNRs (-15dB to 5dB), there is a significant increase in segSNR. However, for relatively higher SNRs (10dB and 15dB), segSNR is minimal as there is much less noise to suppress, i.e., the algorithm imposes more distortion to the signal than it would suppress the noise at higher SNR values. It is important to note that these observations are similar for all speech enhancement algorithms ([38], [41]) and not just our proposed methods in ED.

(ii) For white noise reduction in ED, there is an average segSNR increase of 5.7561dB and 5.7554dB for LPF-scheme 1 and LPF-scheme 2 respectively over all global SNR levels, whereas wind noise reduction in ED yields an average segSNR increase of 6.6811dB and 6.6712dB for HPF-scheme 1 and HPF-scheme 2 respectively. However, it can be observed from Table 5.4 for humming noise reduction in ED (CF-scheme 1 and CF-scheme 2) that segSNR increases with an increasing gain value g of the comb filter. The greater the value of g, the greater the attenuation in the stop band of the filter's frequency response.

(iii) Most importantly, segSNR observations in ED are similar to their corresponding PD filtering operations with no significant differences. The

%loss in segSNR between ED and PD values is minimal as can be seen from Tables 5.3 and 5.4. This is because roundoff errors are introduced by the preprocessing steps prior to the creation of shares for ED schemes and artifacts are introduced into the speech signal as a result of attenuation effect for each scheme. This is common for speech enhancement algorithms, as discussed in Section 5.1. This supports the fact that our proposed scheme improves quality in ED with minimal losses when compared to their PD implementation versions.

Perceptual evaluation of speech Quality (PESQ)

Tables 5.5 and 5.6 represent the perceptual evaluation of speech quality (PESQ) scores of the noisy speech, the denoised signal in ED and in PD and the %loss in PESQ between ED and PD. PESQ is one of the most effective objective measures for estimating the quality of noise-corrupted speech processed by noise suppression algorithms. It has shown high correlations ($r > 0.92$) with subjective listening tests [26]. PESQ evaluates the quality of speech by obtaining the loudness spectra of the clean reference signal and the enhanced signal through an auditory transform (a model of the human auditory system). The difference between both spectra is then computed to estimate the quality of the enhanced signal on a 5 point mean-opinion score (MOS) scale from 1(bad) to 5(excellent). Higher scores represent good speech quality. The implementation of this measure is obtained from [50].

Tables 5.5 and 5.6 reveal that at lower SNRs where background noise is higher, there is improvement in the quality. However, this improvement reduces with higher SNRs. This is because at higher SNRs, the noise power

Table 5.5: PESQ for white, wind noise reduction

White noise							
SNR(dB) noisy speech	PESQ noisy speech	LPF-scheme1			LPF-scheme2		
		denoised in ED	denoised in PD	% Loss in PESQ	denoised in ED	denoised in PD	% Loss in PESQ
-15	1.559797	1.761232	1.761192	0.002268	1.761231	1.744328	0.969022
-10	1.460326	1.777213	1.777222	0.000481	1.777211	1.777224	0.000747
-5	1.624654	1.922633	1.922628	0.000274	1.922635	1.922742	0.005547
0	1.861036	2.157773	2.157729	0.002081	2.157745	2.157882	0.006327
5	1.990906	2.306961	2.307005	0.001926	2.306943	2.306946	0.000125
10	2.417972	2.632079	2.632279	0.007633	2.63206	2.632238	0.006767
15	2.679818	2.896017	2.896386	0.012753	2.89601	2.896394	0.01324

Wind noise							
SNR(dB) noisy speech	PESQ noisy speech	HPF-scheme1			HPF-scheme2		
		denoised in ED	denoised in PD	% Loss in PESQ	denoised in ED	denoised in PD	% Loss in PESQ
-15	2.12794	2.363889	2.363904	0.000628	2.363982	2.36421	0.009672
-10	2.3692	2.69919	2.699187	0.000114	2.699132	2.699265	0.004924
-5	2.70672	3.004558	3.004537	0.000692	3.005193	3.004641	0.018369
0	3.0294	3.239933	3.239868	0.002019	3.244064	3.23989	0.128812
5	3.32316	3.505822	3.505643	0.005128	3.515403	3.505644	0.278388
10	3.65638	3.788975	3.789028	0.001377	3.799723	3.789028	0.282253
15	3.85828	3.976098	3.976029	0.001748	3.989873	3.976028	0.348202

is minimal as compared to the signal, and the algorithm introduces little distortion to the signal. This can be observed for white and wind noise reduction in Table 5.5, as the PESQ scores for the noisy signal and the denoised signal decrease as SNR values increase. Humming noise reduction presented in Table 5.6 also shows similar characteristics for gain values of $g = 0.2$ and 0.5. However, for SNR values of 5dB and higher, the quality for $g = 0.8$ and 1 drops as there is less noise present and the filter imposes more attenuation on the signal at higher gain values, thus causing more signal distortion at higher SNR and g values.

It is important to note that the %loss in PESQ for all schemes in ED from Tables 5.5 and 5.6 is minimal. This further supports the fact that our scheme in ED yields identical results as their PD versions with minimal losses while providing security and privacy of data.

Table 5.6: PESQ for humming noise reduction

g=0.2							
SNR(dB) noisy speech	PESQ noisy speech	CF-scheme1			CF-scheme2		
		denoised in ED	denoised in PD	% Loss in PESQ	denoised in ED	denoised in PD	% Loss in PESQ
-15	1.846326	1.857057	1.857079	0.001206	1.898395	1.898579	0.009738
-10	1.810163	1.885748	1.885754	0.000309	1.877674	1.887513	0.521283
-5	2.040983	2.125112	2.125187	0.003509	2.12919	2.1298	0.028642
0	2.347948	2.441143	2.44117	0.001124	2.459677	2.452621	0.287709
5	2.563545	2.677533	2.677753	0.008234	2.673487	2.679049	0.207584
10	3.021158	3.121028	3.12124	0.006808	3.123496	3.123119	0.012087
15	3.261622	3.372775	3.373371	0.017683	3.375616	3.375677	0.00179
g=0.5							
SNR(dB) noisy speech	PESQ noisy speech	CF-scheme1			CF-scheme2		
		denoised in ED	denoised in PD	% Loss in PESQ	denoised in ED	denoised in PD	% Loss in PESQ
-15	1.846326	1.973547	1.973577	0.001491	2.014597	2.016042	0.071672
-10	1.810163	2.061267	2.061258	0.000422	2.051797	2.062942	0.540288
-5	2.040983	2.316765	2.316809	0.001905	2.319042	2.320255	0.052287
0	2.347948	2.625317	2.625272	0.001721	2.644772	2.638542	0.236092
5	2.563545	2.799452	2.799647	0.006979	2.794744	2.799963	0.1864
10	3.021158	3.150586	3.151013	0.01354	3.146897	3.146427	0.014953
15	3.261622	3.237025	3.237661	0.019657	3.242378	3.242322	0.001735
g=0.8							
SNR(dB) noisy speech	PESQ noisy speech	CF-scheme1			CF-scheme2		
		denoised in ED	denoised in PD	% Loss in PESQ	denoised in ED	denoised in PD	% Loss in PESQ
-15	1.846326	2.136331	2.136361	0.001397	2.165529	2.168196	0.123011
-10	1.810163	2.299348	2.299326	0.000966	2.289894	2.301155	0.489362
-5	2.040983	2.539561	2.53973	0.006686	2.541054	2.543268	0.087021
0	2.347948	2.74083	2.741173	0.012527	2.761715	2.756688	0.18233
5	2.563545	2.752785	2.753687	0.032742	2.741107	2.746802	0.207326
10	3.021158	2.961089	2.962238	0.038795	2.952738	2.952613	0.004229
15	3.261622	2.947032	2.947951	0.03119	2.951306	2.951925	0.02099
g=1							
SNR(dB) noisy speech	PESQ noisy speech	CF-scheme1			CF-scheme2		
		denoised in ED	denoised in PD	% Loss in PESQ	denoised in ED	denoised in PD	% Loss in PESQ
-15	1.846326	2.127502	2.128384	0.041428	2.147793	2.150311	0.117068
-10	1.810163	2.276821	2.279339	0.110486	2.27321	2.24232	1.377592
-5	2.040983	2.487225	2.487807	0.023395	2.461804	2.484055	0.895769
0	2.347948	2.636943	2.637247	0.011554	2.660013	2.633787	0.995739
5	2.563545	2.600074	2.600783	0.027269	2.58934	2.587343	0.077187
10	3.021158	2.799851	2.817672	0.632458	2.78111	2.786666	0.199397
15	3.261622	2.767098	2.762191	0.177645	2.817171	2.812651	0.160714

Pearson's correlation (similarity score)

Table 5.7 represents (1) the similarity scores for white, wind and humming noise reduction between the denoised signal in ED and in PD and (2) the %loss in accuracy between ED denoising and PD denoising. The similarities were computed using Pearson's correlation method. Pearson's correlation coefficient $r_{(X,Y)}$ between two data series X and Y is given by:

$$r_{(X,Y)} = \frac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^{n}(X_i - \bar{X})^2}\sqrt{\sum_{i=1}^{n}(Y_i - \bar{Y})^2}} \tag{5.33}$$

Where $\bar{X}$ and $\bar{Y}$ are the mean of X and Y respectively. $r_{(X,Y)}$ of 0 means no correlation and 1 means total correlation. The correlation coefficients from the results from Table 5.7 suggest that for all proposed schemes in ED, the denoised signals in ED correlate to the denoised signals in PD with coefficients in the range of 0.99 to 1. This represents a strong correlation ($0.99 \leq r \leq 1$), which translates to an accuracy in the range of 99% to 100%. Loss in accuracy between ED and PD denoising is in the order of 10^{-5} to 10^{-9}, which is not significant. This accuracy loss can be accounted for by the roundoff errors introduced as a result of scaling the real-valued signal to positive integers during the preprocessing steps. In the future, we hope to further optimize the solution in order to avoid such errors. This observation suggests that our methods in ED yield identical results to normal PD filtering operations with near zero loss in accuracy while maintaining security and privacy.

Table 5.7: Similarity scores with Pearson's correlation
(Comparison of denoised signal in ED and PD)

	White noise				Wind noise			
	LPF-scheme 1		LPF-scheme 2		HPF-scheme 1		HPF-scheme 2	
SNR (dB) noisy speech	ED vs. PD	% Loss in accuracy	ED vs. PD	% Loss in accuracy	ED vs. PD	% Loss in accuracy	ED vs. PD	% Loss in accuracy
-15	0.99999978	2.15E-09	0.9999997	2.20E-09	0.9999987	1.27E-08	0.99869569	1.30E-05
-10	0.99999959	4.01E-09	0.99999957	4.23E-09	0.99999869	1.31E-08	0.99923438	7.66E-06
-5	0.99999944	5.59E-09	0.99999938	6.13E-09	0.99999866	1.33E-08	0.99922232	7.78E-06
0	0.99999918	8.19E-09	0.99999907	9.21E-09	0.99999797	2.03E-08	0.99966522	3.35E-06
5	0.99999917	8.28E-09	0.99999906	9.42E-09	0.99999816	1.84E-08	0.99980772	1.92E-06
10	0.99999929	7.01E-09	0.99999919	8.08E-09	0.99999781	2.19E-08	0.99977371	2.26E-06
15	0.99999928	7.19E-09	0.99999918	8.24E-09	0.99999863	1.37E-08	0.99986181	1.38E-06

Humming noise CF-scheme 1								
	g=0.2		g=0.5		g=0.8		g=1	
SNR (dB) noisy speech	ED vs. PD	% Loss in accuracy	ED vs. PD	% Loss in accuracy	ED vs. PD	% Loss in accuracy	ED vs. PD	% Loss in accuracy
-15	0.99999959	4.08E-09	0.99999887	1.13E-08	0.99999558	4.42E-08	0.99999254	7.45E-08
-10	0.99999901	9.94E-09	0.99999764	2.36E-08	0.99999439	5.62E-08	0.99999305	6.95E-08
-5	0.9999979	2.12E-08	0.99999619	3.81E-08	0.99999424	5.76E-08	0.99999379	6.21E-08
0	0.99999525	4.75E-08	0.99999346	6.55E-08	0.99999221	7.79E-08	0.99999199	8.01E-08
5	0.99999406	5.94E-08	0.99999338	6.62E-08	0.99999304	6.96E-08	0.99999299	7.01E-08
10	0.9999941	5.90E-08	0.99999369	6.31E-08	0.9999935	6.50E-08	0.99999348	6.52E-08
15	0.99999413	5.87E-08	0.99999408	5.92E-08	0.99999406	5.94E-08	0.99999406	5.95E-08

Humming noise CF-scheme 2								
	g=0.2		g=0.5		g=0.8		g=1	
SNR (dB) noisy speech	ED vs. PD	% Loss in accuracy	ED vs. PD	% Loss in accuracy	ED vs. PD	% Loss in accuracy	ED vs. PD	% Loss in accuracy
-15	0.99999959	4.08E-09	0.99999887	1.13E-08	0.99999558	4.42E-08	0.99999255	7.45E-08
-10	0.99999901	9.94E-09	0.99999764	2.36E-08	0.99999439	5.62E-08	0.99999305	6.95E-08
-5	0.9999979	2.12E-08	0.99999619	3.81E-08	0.99999424	5.76E-08	0.99999379	6.21E-08
0	0.99999525	4.75E-08	0.99999346	6.55E-08	0.99999221	7.79E-08	0.99999199	8.01E-08
5	0.99999406	5.94E-08	0.99999338	6.62E-08	0.99999304	6.96E-08	0.99999299	7.01E-08
10	0.9999941	5.90E-08	0.99999369	6.31E-08	0.9999935	6.50E-08	0.99999348	6.52E-08
15	0.99999413	5.87E-08	0.99999408	5.92E-08	0.99999406	5.94E-08	0.99999406	5.95E-08

5.3.2 Subjective quality measurement

We evaluated the subjective quality similarity between the denoised signals from our proposed schemes in ED and their PD denoised versions by conducting an online user survey [1]. 20 users in the age group of 18-34 years participated in the survey. We randomly selected a speech file from each noise group (i.e. white noise, humming noise and wind noise) and denoised it in ED and in PD. Users listened to each pair (ED denoised and PD denoised speech files) and rated the quality similarity in terms of clarity and pleasantness. The score is captured on a 5 point Mean Opinion Score (MOS) scale from 1(bad) to 5(excellent). Figure 5.6 shows the bar graph of the MOS taken for the 20 users. The bar graph reveals that the scores for the similarity comparison between the ED denoised signals for our proposed schemes (LPF-scheme 1 and 2, CF-scheme 1 and 2, and HPF-scheme 1 and 2) and their PD denoised versions are in the range of [4 - 5]. This shows a similarity score between very good and excellent with slight variations between schemes. This can be attributed to the fact that speech listening tests are influenced by the highly subjective nature of quality. This means that participants have different standards of what they consider to be good or poor quality, thereby introducing slight variations in the ratings among listeners [35]. We subjected scores from all 20 users to statistical analysis in order to assess their significant differences. Analysis of variance (ANOVA) indicated no significant difference ($F(5, 114) = 0.312, p = 0.904$), which means that all users agreed that the perceptual quality of denoised signals by our proposed

[1] http://myabukari.polldaddy.com/s/speech-quality-similarity-survey

schemes in ED is similar to the quality of their PD implementation versions with minimal data loss, which corroborates our findings for segSNR, PESQ and similarity scores.

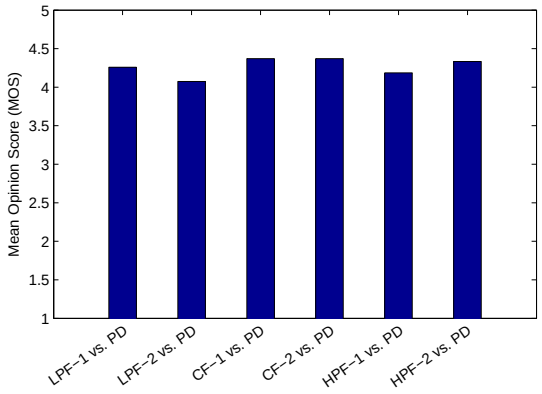
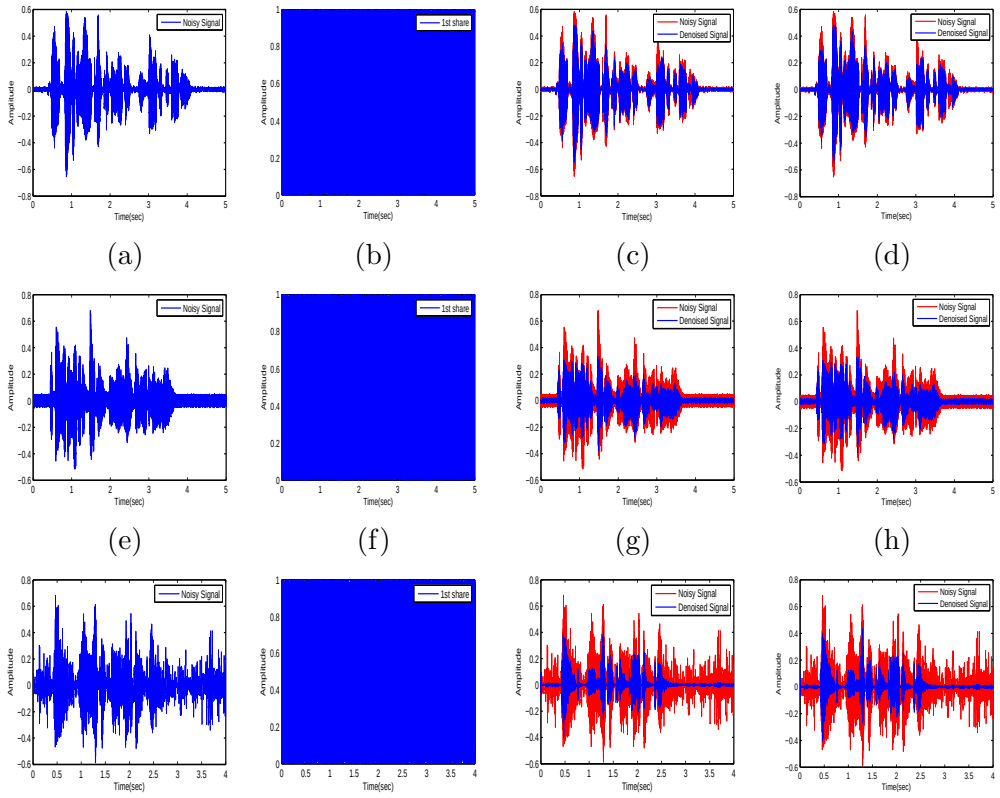


Figure 5.6: User study for comparison of quality similarity between ED denoised signals (LPF-scheme 1 and 2, CF-scheme 1 and 2, and HPF-scheme 1 and 2) and their PD denoised versions.

5.3.3 Time domain analysis

Figures 5.7a - 5.7d, 5.7e - 5.7h and 5.7i - 5.7l show the time domain plots for low pass filtering white noise, comb filtering humming noise and high pass filtering wind noise, respectively. Each plot scenario contains the noise quality degraded speech secret, one of its shares, the denoised signal in ED and the denoised signal in PD. In all cases, observing the amplitude distribution of the quality degraded speech secret and denoised reconstructed secret, it is evident that: (1) our proposed methods reduce noise, as part of the amplitude portions of the signals, which likely represent noise, are absent in the denoised signals, (2) the encrypted share signals are completely noise

(arbitrary data) as each share sample is a unique polynomial generated with a random blinding factor in Equation (2.1) and (3) the waveforms of the denoised signals in ED and in PD are similar which supports our finding from Pearson's similarity score that ED denoising correlates highly with their PD versions with minimal loss in accuracy.



(a) (b) (c) (d)

(e) (f) (g) (h)

(i) Noisy speech secret (j) $1st$ share (k) denoised in ED (l) denoised in PD

Figure 5.7: Time domain plots of noisy speech, one of its shares, denoised signals in ED and PD: (a) white noise corrupted speech signal (b) its $1st$ share (c) denoised in ED and (d) denoised in PD. Humming noise reduction with comb filtering: (e) humming noise corrupted speech signal (f) its $1st$ share (g) denoised in ED and (h) denoised in PD. Wind noise reduction with HPF (High pass filtering): (i) noisy speech signal corrupted with wind noise (j) its $1st$ share (k) denoised in ED and (l) denoised in PD.

5.3.4 Magnitude spectrum analysis (frequency domain)

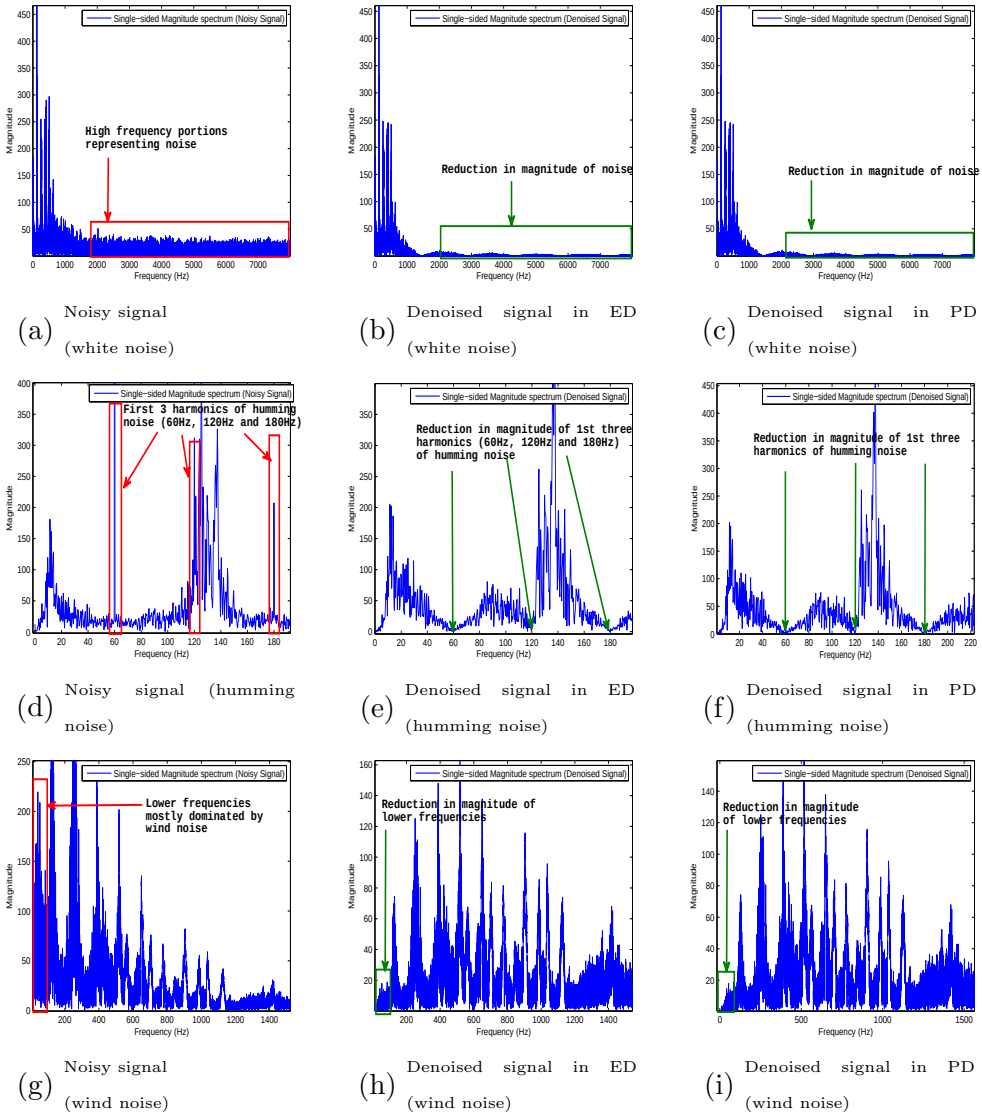
Figure 5.8 represents the magnitude spectrum (frequency domain) of the noisy speech secret, the denoised signal in ED and the denoised signal in PD for the attenuation of white noise (fig. 5.8a - 5.8c), humming noise (fig. 5.8d - 5.8f) and wind noise (fig. 5.8g - 5.8i). Based on these plots, the following observations can be made:

1. Figures 5.8a - 5.8c show that white noise, which is characterized by higher frequencies (from about 1.5kHz to 8kHz) in the noisy signal, is reduced in magnitude in the denoised signal in both ED and PD.

2. Figures 5.8d - 5.8f show a zoomed-in version of the magnitude response of the signals to highlight the $1st$, $2nd$ and $3rd$ harmonics (60Hz, 120Hz and 180Hz) of the humming noise. These plots reveal spikes in magnitudes at 60Hz, 120Hz and 180Hz of the noisy signal (Figure 5.8d) and a reduction in magnitude in the denoised signals in ED and in PD.

3. The magnitude spectrum for the wind noise contaminated signal (Figure 5.8g) shows higher magnitudes concentrated at lower frequencies (wind noise is characterized by lower frequencies mostly $<$ 500Hz depending on the wind speed). The denoised signals in ED (Figure 5.8b) and in PD (Figure 5.8c) reveal an attenuation in the magnitude of these lower frequencies.

The above observations support the fact that our methods in ED attenuate the noise for each case and also produce similar magnitude responses as those in PD processing.



(a) Noisy signal (white noise)

(b) Denoised signal in ED (white noise)

(c) Denoised signal in PD (white noise)

(d) Noisy signal (humming noise)

(e) Denoised signal in ED (humming noise)

(f) Denoised signal in PD (humming noise)

(g) Noisy signal (wind noise)

(h) Denoised signal in ED (wind noise)

(i) Denoised signal in PD (wind noise)

Figure 5.8: Frequency plots (magnitude spectrum) of noisy speech signal and denoised signals in ED and in PD for the reduction of white, humming and wind noise.

5.3.5 Computational complexity analysis

Table 5.8 details the average processing time per scheme for creating secret shares, performing filtering operations in ED and reconstructing the enhanced speech secret for all 35 quality-degraded speech secrets for each noise type. The time information in the table suggests that the complexity of reconstructing the enhanced secret is relatively lower than that of creating secret shares and filtering in ED. Creating shares has the highest time complexities because of the preprocessing (Equations (5.7), (5.8) and (5.9)) involved to convert the real-valued signal to positive integer domain and the polynomial computation in modular domain.

Table 5.8: Average processing time per signal point (ms)

Encrypted Domain				
Scheme	Noise type	Share creation (offline)	ED processing	Denoised secret reconstruction
LPF-scheme 1	white noise	337.85	575.29	7.36
LPF-scheme 2	white noise	333.49	29.11	9.01
CF-scheme 1	humming noise	357.55	20.93	9.02
CF-scheme 2	humming noise	343.88	11.29	7.38
HPF-scheme 1	wind noise	357.64	13.36	6.91
HPF-scheme 2	wind noise	347.07	14.17	5.77
Plaintext Domain				
Scheme	Noise type	Share creation	PD processing	Denoised secret reconstruction
LPF-difference eqn.	white noise	n/a	75.40	n/a
LPF-convolution	white noise	n/a	0.86	n/a
CF-difference eqn.	humming noise	n/a	1.96	n/a
CF-convolution	humming noise	n/a	0.87	n/a
HPF-difference eqn.	wind noise	n/a	1.90	n/a
HPF-convolution	wind noise	n/a	0.75	n/a
	wind noise	0.75		

Note: LPF-difference eqn., LPF-convolution, CF-difference eqn., CF-convolution, HPF-difference eqn. and HPF-convolution are the PD filtering versions of LPF-scheme 1, LPF-scheme 2, CF-scheme 1, CF-scheme 2, HPF-scheme 1 and HPF-scheme 2 respectively.

Share creation can be performed offline in order to reduce complexity on the client side. Outsourcing storage and high-end computing to the CDC means that the majority of computation (operations) should be performed

on the CDC. To evaluate the number of operations performed on the CDC and on the client side, we present Table 5.9. This table shows the number of operations for ED processing over cloud and on the client side. Creating shares is not included as it is performed offline. The table shows that the majority of operations are performed on the CDC, most of which are modular operations. It should also be noted that modular operations are more expensive than basic arithmetic operations (e.g. a modular addition operation on the CDC has a higher complexity than a basic addition operation on the client side).

Table 5.9: Number of operations on cloud and client side

Scheme	ED(Cloud processing)	Client side
LPF-scheme1	$L(M-1)$ modular additions L modular inverse	L subtractions L divisions
LPF-scheme2	$(L+I_{LPF})-1$ modular additions L modular inverse	L subtractions L divisions
CF-scheme1	L modular additions L modular subtractions $2L$ modular multiplications	$2L$ subtractions $2L$ divisions
CF-scheme2	I_{CF} multiplications $2\Big((L+I_{CF})-1\Big)$ modular additions $L \times I_{CF}$ modular multiplications 1 division	$2L$ subtractions $2L$ divisions
HPF-scheme1	L modular additions L modular subtractions	L subtractions L divisions
HPF-scheme2	$2\Big((L+I_{HPF})-1\Big)$ modular additions $L \times I_{HPF}$ modular multiplications 1 division	L subtractions L divisions
Note: L is the number of samples of the speech signal, M is the size of the MA lowpass filter, I_{LPF}, I_{CF} and I_{HPF} are sizes of the impulse responses h_{LPF}, h_{CF} and h_{HPF} respectively.		

Theorem 4 *Our proposed schemes in ED (LPF-scheme 1 and 2, CF-scheme 1 and 2, and HPF-scheme 1 and 2) are $O\big(\frac{\mathcal{N}_U}{\mathcal{N}_{CDC}}\big)$-efficient if the total number of operations $\mathcal{N}_U$ performed by the client is less than the total number of operations $\mathcal{N}_{CDC}$ performed by the CDC (where $O(\mathcal{N}_U)$ and $O(\mathcal{N}_{CDC})$ are the asymptotic complexities of the client and the CDC respectively).*

Proof 14 Proof follows from table 5.9 which represents the number of operations performed by the client and the CDC. It is evident for all proposed schemes in ED (LPF-scheme 1 and 2, CF-scheme 1 and 2, and HPF-scheme 1 and 2) from table 5.9 that: $\mathcal{N}_U < \mathcal{N}_{CDC}$ and consequently $O(\mathcal{N}_U) < O(\mathcal{N}_{CDC})$.

5.3.6 Statistical analysis

We used statistical analysis to assess the significant differences between our results obtained from the objective evaluation. The goal of our statistical analysis are: (i) to evaluate the significant effect between ED and PD processing in order to test the hypothesis that our proposed schemes in ED produce similar results as their PD versions with minimal loss in accuracy, (ii) to compare the performance of our proposed schemes in ED and their PD versions with reference to the noisy signals, (This is to evaluate statistically whether the quality of the denoised signals has been improved) and (iii) to compare the performance of the various g values of the comb filter in order to ascertain which range of g produces the best enhancement for humming noise attenuation.

We subjected the results obtained from objective measures (segSNR and PESQ) from ED and PD denoising to statistical analysis. The analysis was performed on results from distributions of: (i) ED difference eqn. schemes: LPF-scheme 1, CF-scheme 1 and HPF-scheme 1, (ii) PD implementation versions of difference eqn. schemes, (iii) ED convolution schemes: LPF-scheme 2, CF-scheme 2 and HPF-scheme 2, and (iv) PD implementation versions

of convolution schemes. As depicted in Table 5.10, ANOVA indicated no significant difference ($0.89 < p \leq 1$) across all SNR levels and noise types between ED and PD denoised signals, which supports the fact that our proposed schemes in ED yield similar results to their PD implementations with minimal or near negligible data losses.

Further ANOVA analysis of results from distributions of ED denoised signals (difference eqn. schemes and convolution schemes) and noisy signals indicated a significant effect of ($F(2, 102) = 4.86, p < 0.009$), ($F(2, 102) = 3.48, p < 0.04$) and ($F(2, 102) = 4.61, p < 0.0014$) for white noise, wind noise and humming noise respectively. It is important to note that the same significant effect will result from comparing PD denoised signals and noisy signals since we have already established that there is no significant difference between ED and PD denoising. In order to assess where the difference lies, we conducted a pairwise multiple comparison (post-hoc test) using Tukey's HSD test. The results for Tukey's HSD test are shown in Table 5.11 for both segSNR and PESQ. From the table, an indication of: (i) "E" means that there is no significant difference between the ED denoised signal and the noisy signal, (ii) "Y" means scores obtained from ED denoised signals are significantly higher than that of the noisy signal, and (iii) "N" means that scores obtained from ED denoised signals are significantly less than that of the noisy signal. Table 5.11 shows that at lower SNRs (-15dB to 5dB) for both white and wind noise, the quality (segSNR) of the denoised signlas in ED is significantly higher ($p < 0.05$) than that of the noisy signals. However, there is no significant effect at higher SNRs of 15dB. This is because there is more signal than noise power at higher SNRs, hence the noise reduction algorithms

impose more strain on the signal than the noise, thereby causing distortions. For PESQ scores at -15dB, the results from Table 5.5 show improvement in denoised signals for both white and wind noise reduction, however the difference in scores was not found to be statistically significant. From -5dB to 15dB, PESQ scores for enhanced signals were significantly higher ($p < 0.05$) in quality than noisy signals. Analysis results from Table 5.11 also showed that humming noise reduction with $g = 0.8$ produced significantly better quality (segSNR and PESQ) than the other g values ($0.2, 0.5$ and 1) across all SNR levels. The suppression effect of the comb filter is minimal at lower values of g and increases towards 1. g can be any value between 0.1 and 1. Lower values of g cause minimal noise attenuation with less distortions while higher values suppress more noise but introduce more distortions to the signal. The significance score from our statistical analysis does not mean that $g = 0.8$ is the best gain value, but rather a value between 0.1 and 1 should be chosen in order to balance noise attenuation and signal distortion. The significant difference between the denoised signals in ED and noisy signals for $g = 0.8$ follows the same lines as that of white and wind noise.

5.4 Chapter Summary and Conclusion

This chapter presented the denoising of quality-degraded speech secret files outsourced to cloud. LPF, CF and HPF were proposed, based on their linearity and feasibility with homomorphic computation, for the attenuation of white, humming and wind noise respectively. Utilizing the principles of LTI systems, each of these filters were implemented using (1) difference equation

and (2) convolution with filter impulse response.

Our objective and subjective (listening test) evaluation revealed that our proposed schemes in ED improved the quality of the degraded speech which yields similar results to PD denoising with minimal accuracy losses due to preprocessing techniques. Further statistical analysis on the experimental results (segSNR, PESQ and survey scores) showed no significant difference between ED and PD denoising which corroborates our findings from both the objective and subjective evaluation.

Table 5.10: ANOVA comparison of results between ED and PD denoising

		segSNR			PESQ		
noise type	SNR(dB)	F	$F_{0.05}$	p val	F	$F_{0.05}$	p val
white	-15	0.307	3.24	0.82	0	3.24	1
	-10	0	3.24	1	0	3.24	1
	-5	0	3.24	1	0	3.24	1
	0	0	3.24	1	0	3.24	1
	5	0	3.24	1	0	3.24	1
	10	0	3.24	1	0	3.24	1
	15	0	3.24	1	0.0001	3.24	1
wind	-15	0	3.24	1	0.0001	3.24	1
	-10	0.2139	3.24	0.8853	0.0002	3.24	1
	-5	0.0001	3.24	1	0.0001	3.24	1
	0	0.0037	3.24	0.9997	0.0056	3.24	0.9994
	5	0.0434	3.24	0.9875	0.0061	3.24	0.9993
	10	0.0362	3.24	0.9904	0.0114	3.24	0.9983
	15	0.1	3.24	0.9589	0.0587	3.24	0.9807
humming g=0.2	-15	0	3.24	1	0.0006	3.24	1
	-10	0.0026	3.24	0.9998	0.001	3.24	1
	-5	0	3.24	1	0.0013	3.24	0.9999
	0	0.0019	3.24	0.9999	0.0005	3.24	1
	5	0.0033	3.24	0.9997	0	3.24	1
	10	0	3.24	1	0.0003	3.24	1
	15	0	3.24	1	0.001	3.24	1
humming g=0.5	-15	0.0001	3.24	1	0.0006	3.24	1
	-10	0.0038	3.24	0.9997	0.001	3.24	1
	-5	0.0001	3.24	1	0.0014	3.24	0.9999
	0	0.0017	3.24	0.9999	0.0004	3.24	1
	5	0.0031	3.24	0.9998	0.0002	3.24	1
	10	0	3.24	1	0.0008	3.24	1
	15	0.0003	3.24	1	0.0013	3.24	0.9999
humming g=0.8	-15	0.0002	3.24	1	0.0006	3.24	1
	-10	0.0062	3.24	0.9993	0.0013	3.24	0.9999
	-5	0.0002	3.24	1	0.0013	3.24	0.9999
	0	0.0016	3.24	0.9999	0.0001	3.24	1
	5	0.0078	3.24	0.999	0.0107	3.24	0.9984
	10	0	3.24	1	0.0253	3.24	0.9943
	15	0.0011	3.24	0.9999	0.0114	3.24	0.9983
humming g=1	-15	0.0009	3.24	1	0.0001	3.24	1
	-10	0.0971	3.24	0.9605	0.0062	3.24	0.9993
	-5	0.028	3.24	0.9934	0.0014	3.24	0.9999
	0	0.05	3.24	0.9847	0.0034	3.24	0.9997
	5	0.0005	3.24	1	0.0218	3.24	0.9954
	10	0.0009	3.24	1	0.0238	3.24	0.9948
	15	0.0452	3.24	0.9868	0.0239	3.24	0.9948

Table 5.11: Tukey's HSD test of multiple comparisons between results obtained from ED denoised signals and noisy signals

		segSNR						PESQ					
		ED (diff eqn. schemes) denoised vs. noisy			ED (conv. schemes) denoised vs. noisy			ED (diff eqn. schemes) denoised vs. noisy			ED (conv. schemes) denoised vs. noisy		
noise type	SNR (dB)	mean diff.	$Q_{0.05}$		mean diff.	$Q_{0.05}$		mean diff.	$Q_{0.05}$		mean diff.	$Q_{0.05}$	
white	-15	9.32	4.23	Y	9.32	4.23	Y	0.21	0.56	E	0.21	0.56	E
	-10	8.9	2.47	Y	8.9	2.47	Y	0.36	0.35	Y	0.36	0.35	Y
	-5	7.97	2.89	Y	7.97	2.89	Y	0.32	0.45	E	0.32	0.45	E
	0	6.81	2.66	Y	6.81	2.66	Y	0.33	0.28	Y	0.33	0.28	Y
	5	4.71	1.55	Y	4.71	1.55	Y	0.35	0.19	Y	0.35	0.19	Y
	10	2.98	1.96	Y	2.98	1.96	Y	0.25	0.21	Y	0.25	0.21	Y
	15	0.4	1.37	E	0.4	1.37	E	0.24	0.17	Y	0.24	0.17	Y
wind	-15	19.63	3.02	Y	19.63	3.02	Y	0.24	0.47	E	0.24	0.47	E
	-10	15.3	1.34	Y	15.3	1.34	Y	0.31	0.31	Y	0.37	0.31	Y
	-5	11.22	1.84	Y	11.21	1.84	Y	0.3	0.2	Y	0.3	0.2	Y
	0	7.57	1.65	Y	7.55	1.65	Y	0.21	0.14	Y	0.21	0.14	Y
	5	2.01	1.04	Y	2	1.04	Y	0.18	0.14	Y	0.19	0.14	Y
	10	1.75	1.74	Y	1.77	1.74	Y	0.13	0.12	Y	0.14	0.12	Y
	15	7.21	0.88	N	7.22	0.88	N	0.12	0.09	Y	0.13	0.09	Y
humming g=0.2	-15	1.9	4.38	E	1.9	4.38	E	0.05	0.54	E	0.05	0.54	E
	-10	1.9	2.49	E	1.9	2.49	E	0.07	0.41	E	0.07	0.41	E
	-5	1.92	2.81	E	1.92	2.81	E	0.09	0.43	E	0.09	0.43	E
	0	1.69	2.98	E	1.69	2.98	E	0.11	0.41	E	0.11	0.41	E
	5	1.31	2.16	E	1.31	2.16	E	0.11	0.23	E	0.11	0.23	E
	10	0.43	2.93	E	0.43	2.93	E	0.1	0.31	E	0.1	0.31	E
	15	1	1.66	E	1	1.66	E	0.11	0.12	E	0.11	0.12	E
humming g=0.5	-15	5.82	4.38	Y	5.82	4.38	Y	0.17	0.51	E	0.17	0.51	E
	-10	5.61	2.48	Y	5.61	2.48	Y	0.24	0.39	E	0.24	0.39	E
	-5	5.09	2.83	Y	5.09	2.83	Y	0.28	0.41	E	0.28	0.41	E
	0	3.98	2.81	Y	3.98	2.81	Y	0.3	0.4	E	0.3	0.4	E
	5	2.09	1.89	Y	2.09	1.89	Y	0.23	0.23	Y	0.23	0.23	Y
	10	0.02	2.14	E	0.02	2.14	E	0.13	0.32	E	0.13	0.32	E
	15	2.95	1.32	N	2.95	1.32	N	0.02	0.07	E	0.02	0.07	E
humming g=0.8	-15	12.27	4.3	Y	12.27	4.3	Y	0.32	0.46	E	0.32	0.46	E
	-10	10.92	2.29	Y	10.92	2.29	Y	0.48	0.32	Y	0.48	0.32	Y
	-5	8.87	2.63	Y	8.87	2.63	Y	0.5	0.38	Y	0.5	0.38	Y
	0	6.6	2.05	Y	6.6	2.05	Y	0.41	0.36	Y	0.41	0.36	Y
	5	2.98	1.43	Y	2.98	1.43	Y	0.18	0.17	Y	0.18	0.17	Y
	10	0.41	1.52	E	0.41	1.52	E	0.07	0.35	E	0.07	0.35	E
	15	3.95	0.96	N	3.95	0.96	N	0	0.23	E	0	0.46	E
humming g=1	-15	17.45	3.81	Y	17.45	3.81	Y	0.3	0.38	E	0.3	0.38	E
	-10	14.43	1.79	Y	14.43	1.79	Y	0.46	0.26	Y	0.46	0.26	Y
	-5	11.19	1.94	Y	11.19	1.94	Y	0.42	0.3	Y	0.42	0.3	Y
	0	8.2	1.48	Y	8.2	1.48	Y	0.31	0.32	E	0.31	0.32	E
	5	3.27	1.06	Y	3.27	1.06	Y	0.03	0.22	E	0.03	0.22	E
	10	0.18	1.35	E	0.18	1.35	E	0.24	0.4	E	0.24	0.4	E
	15	5.14	0.79	N	5.14	0.79	N	0.44	0.07	Y	0.44	0.07	N

Note: Comparison between the scores of denoised signals in ED with that of noisy signal indicates (1) "E" if there no significant difference between ED denoised signal and noisy signal (2) "Y" if scores obtained from ED denoised signals are significantly higher than that of noisy signal and (3) "N" if scores obtained from ED denoised signals are significantly lesser than that of noisy signal

Chapter 6

Conclusion and Future Work

There is a current trend in IT moving toward cloud computing where online access to resources (storage, computation and network) is available on a pay-per-use basis. In order for individuals and companies to entrust storage and computation of their data to a CDC, security issues should be given great importance. Though there are policies and service level agreements (SLAs) governing the operations of CDCs, this is not enough to guarantee the security and privacy of data. Researchers in the fields of mathematics, computer science and engineering are continuously developing encryption protocols and computational tasks possible in ED. Only then will CDC services be fully embraced without fear of security or privacy issues. As a contribution to realize this, this thesis has presented methods for secure storage and processing (addition of reverberation effect and noise reduction) of audio/speech data over cloud with minimal overheads in terms of data losses, transmission bandwidth and computational complexity. With the proposed methods in this thesis, a client constrained in resources (storage, computation or IT exper-

tise) can offload data to a CDC and leverage the benefits of the CDC such as elasticity, scalability, availability, etc., to reduce capital expenditure and optimize operational cost. The information theoretic security of our methods protects the privacy and confidentiality of sensitive information over CDC (e.g. Amazon Web Services (AWS) and Microsoft Azure) where it is the responsibility of the client to secure their data.

Audio and speech processing in ED is still scarcely explored as compared to other multimedia content such as text and image, and there are many open topics that still need to be addressed:

1. Collusion avoidance among CDCs: CDCs in this work are assumed to be non-colluding which means that they do not come together to reconstruct the secret. In the future we hope to explore the possibility of designing a protocol to avoid collusion amongst CDCs.

2. Non-linear speech noise reduction algorithms: In this thesis we focused on linear noise reduction algorithms that are feasible with only homomorphism. Implementation of non-linear algorithms (e.g. with maxima, minima or comparison) such as subspace techniques, log-MMSE (log-Minimum Mean Square Error), etc. cannot be achieved with only homomorphic encryption techniques. However, incorporating other SPED primitives such as SMC (Yao's protocol) with homomorphic cryptosystems has been used to solve the problem of non-linearity in ED computation, which can be applied in the case of speech noise reduction.

List of Related Publications

Below is a list of publications related to this thesis:

Published

1. M Abukari Yakubu, Namunu C Maddage, and Pradeep K Atrey. Audio secret management scheme using Shamir's secret sharing. In *Proceedings of 21st International Conference on MultiMedia Modeling*, pages 396-407, Sydney, Australia, 2015.

2. M Abukari Yakubu, Pradeep K Atrey, and Namunu C Maddage. Secure audio reverberation over cloud. In *Proceeding of 10th Annual Symposium on Information Assurance (ASIA15)*, page 39, New York, USA, 2015.

To be submitted

M Abukari Yakubu, Namunu C Maddage, and Pradeep K Atrey. Speech Noise Reduction in Encrypted Domain. *To be submitted to: IEEE Transactions on Information Forensics and Security.*

Bibliography

[1] Pradeep K Atrey, Saeed Alharthi, M Anwar Hossain, Abdullah Al-Ghamdi, and Abdulmotaleb El Saadik. Collective control over sensitive video data using secret sharing. *Multimedia Tools and Applications*, 73(3):1459–1486, 2014.

[2] auphonic. auphonic, 2014. https://auphonic.com.

[3] Michael Backes, Goran Doychev, Markus Dürmuth, and Boris Köpf. Speaker recognition in encrypted voice streams. In *Proceedings of 15th European Symposium on Research in Computer Security*, pages 508–523, Athens, Greece, 2010.

[4] Mauro Barni, Pierluigi Failla, Vladimir Kolesnikov, Riccardo Lazzeretti, Ahmad-Reza Sadeghi, and Thomas Schneider. Secure evaluation of private linear branching programs with medical applications. In *Proceedings of the 14th European Symposium on Research in Computer Security*, pages 424–439. Springer, Saint-Malo, France, 2009.

[5] Tiziano Bianchi, Alessandro Piva, and Mauro Barni. On the implementation of the discrete fourier transform in the encrypted domain. *IEEE Transactions on Information Forensics and Security*, 4(1):86–97, 2009.

[6] Steven F Boll. Suppression of acoustic noise in speech using spectral subtraction. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 27(2):113–120, 1979.

[7] Ka Fai Peter Chan. Secret sharing in audio steganography. In *Proceedings of Information Security for South Africa*, Johannesburg, South Africa, 2011.

[8] Benny Chor, Eyal Kushilevitz, Oded Goldreich, and Madhu Sudan. Private information retrieval. *Journal of the ACM (JACM)*, 45(6):965–981, 1998.

[9] Industrial Noise Control. Noise control concepts, 2010. http://www.industrialnoisecontrol.com/qa.htm.

[10] Ivan Damgård and Eiichiro Fujisaki. A statistically-hiding integer commitment scheme based on groups with hidden order. In *Proceeding of the 8th International Conference on the Theory and Application of Cryptology and Information Security*, pages 125–142, Queenstown, New Zealand, 2002.

[11] Yvo Desmedt, Shuang Hou, and Jean-Jacques Quisquater. Audio and optical cryptography. In *Proceedings of International Conference on the Theory and Application of Cryptology and Information Security*, pages 392–404, Beijing, China, 1998.

[12] Mohammad Ehdaie, Taraneh Eghlidos, and Mohammad Reza Aref. A novel secret sharing scheme from audio perspective. In *Proceedings of*

International Symposium on Telecommunications, pages 13–18, Tehran, Iran, 2008.

[13] Marwa A Abd El-Fattah, Moawad I Dessouky, Alaa M Abbas, Salaheldin M Diab, El-Sayed M El-Rabaie, Waleed Al-Nuaimy, Saleh A Alshebeili, and Fathi E Abd El-Samie. Speech enhancement with an adaptive wiener filter. *International Journal of Speech Technology*, 17(1):53–64, 2014.

[14] Taher ElGamal. A public key cryptosystem and a signature scheme based on discrete logarithms. In *Proceedings of 4th Annual International Cryptology Conference*, pages 10–18, Santa Barbara, USA, 1985.

[15] Yariv Ephraim and Hany L Van Trees. A signal subspace approach for speech enhancement. *IEEE Transactions on Speech and Audio Processing*, 3(4):251–266, 1995.

[16] Zekeriya Erkin, Alessandro Piva, Stefan Katzenbeisser, Reginald L Lagendijk, Jamshid Shokrollahi, Gregory Neven, and Mauro Barni. Protection and retrieval of encrypted multimedia content: When cryptography meets signal processing. *EURASIP Journal on Information Security*, 2007:17, 2007.

[17] freesound. wind noise, 2014. https://www.freesound.org/search/?q=wind+noise.

[18] Eiichiro Fujisaki and Tatsuaki Okamoto. A practical and provably secure scheme for publicly verifiable secret sharing and its applications. In

Proceedings of International Conference on the Theory and Application of Cryptographic Techniques, pages 32–46, Espoo, Finland, 1998.

[19] Norihiro Fujita, Ryouichi Nishimura, and Yôiti Suzuki. Audio secret sharing for 1-bit audio. *Acoustical Science and Technology*, 27(3):171–173, 2006.

[20] Sharon Gannot. Speech enhancement: Application of the kalman filter in the estimate-maximize (em) framework. In *Speech Enhancement*, pages 161–198. Springer, 2005.

[21] Bart Goethals, Sven Laur, Helger Lipmaa, and Taneli Mielikäinen. On private scalar product computation for privacy-preserving data mining. In *Proceedings of 7th International Conference on Information Security and Cryptology*, pages 104–120. Springer, Seoul, Korea, 2005.

[22] universida de Vigo GPSC. Signal processing encrypted domain, 2014. http://webs.uvigo.es/gpscuvigo/?q=content/signal-processing-encrypted-domain.

[23] John HL Hansen and Bryan L Pellom. An effective quality evaluation protocol for speech enhancement algorithms. In *Proceedings of the 5th International Conference on Spoken Language Processing*, volume 7, pages 2819–2822, Sydney, Australia, 1998.

[24] Susan Hohenberger and Anna Lysyanskaya. How to securely outsource cryptographic computations. In *Theory of Cryptography, Lecture Notes in Computer Science*, volume 3378, pages 264–282. Springer, Cambridge, MA, USA, 2005.

[25] Yi Hu and Philipos C Loizou. A generalized subspace approach for enhancing speech corrupted by colored noise. *IEEE Transactions on Speech and Audio Processing*, 11(4):334–341, 2003.

[26] Yi Hu and Philipos C Loizou. Evaluation of objective quality measures for speech enhancement. *IEEE Transactions on Audio, Speech, and Language Processing*, 16(1):229–238, 2008.

[27] Jonathan Katz and Yehuda Lindell. *Introduction to Modern Cryptography*. Chapman & Hall/CRC Press, 1st edition, 2007.

[28] Liaqat Ali Khan, Muhammad Shamim Baig, and Amr M Youssef. Speaker recognition from encrypted voip communications. *Digital Investigation*, 7(1):65–73, 2010.

[29] Kazuhiro Kondo. *Subjective quality measurement of speech: its evaluation, estimation and applications*. Springer Science & Business Media, 2012.

[30] Ankita Lathey and Pradeep K Atrey. Image enhancement in encrypted domain over cloud. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 11(3):38, 2015.

[31] Hanoch Lev-Ari and Yariv Ephraim. Extension of the signal subspace speech enhancement approach to colored noise. *IEEE Signal Processing Letters*, 10(4):104–106, 2003.

[32] Jae S Lim and Alan V Oppenheim. Enhancement and bandwidth compression of noisy speech. *Proceedings of the IEEE*, 67(12):1586–1604, 1979.

[33] Chen-Chi Lin, Chi-Sung Laih, and Ching-Nung Yang. New audio secret sharing schemes with time division technique. *Journal of Information Science and Engineering*, 19(4):605–614, 2003.

[34] Philipos C Loizou. Speech enhancement based on perceptually motivated bayesian estimators of the magnitude spectrum. *IEEE Transactions on Speech and Audio Processing*, 13(5):857–869, 2005.

[35] Philipos C Loizou. Speech quality assessment. In *Multimedia Analysis, Processing and Communications*, volume 346, pages 623–654. Springer, 2011.

[36] Wenjun Lu, Ashwin Swaminathan, Avinash L Varna, and Min Wu. Enabling search over encrypted multimedia databases. In *Proceedings of IS&T/SPIE Electronic Imaging*, pages 725418–725418, San Jose, CA , USA, 2009.

[37] Yang Lu and Philipos C Loizou. A geometric approach to spectral subtraction. *Speech Communication*, 50(6):453–466, 2008.

[38] Daniel PK Lun, Tak-Wai Shen, and KC Ho. A novel expectation-maximization framework for speech enhancement in non-stationary noise environments. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 22(2):335–346, 2014.

[39] Mohammad Bilal Malik, M Asger Ghazi, and Raian Ali. Privacy preserving data mining techniques: current scenario and future prospects. In *Proceedings of 3rd International Conference on Computer and Communication Technology*, pages 26–32, Allahabad, India, 2012.

[40] Imperial College London Mike Brookes. Voicebox, 2011. http://www.ee.ic.ac.uk/hp/staff/dmb/voicebox/voicebox.html.

[41] Nasser Mohammadiha, Paris Smaragdis, and Arne Leijon. Simultaneous noise classification and reduction using a priori learned models. In *Proceedings of IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, pages 1–6, Southampton, UK, 2013.

[42] James A Moorer. About this reverberation business. *Computer Music Journal*, pages 13–28, 1979.

[43] DREAMS Initial Training Network. Room impulse responses, 2013. http://www.dreams-itn.eu/index.php/dissemination/science-blogs/24-rir-databases.

[44] Stavros Ntalampiras, Todor Ganchev, Ilyas Potamitis, and Nikos Fakotakis. Objective comparison of speech enhancement algorithms under real world conditions. In *Proceedings of the 1st ACM International Conference on PErvasive Technologies Related to Assistive Environments*, page 34, Athens, Greece, 2008.

[45] The Society of Wind Vigilance. Low frequency noise, infrasound and wind turbines, 2012. http://www.windvigilance.com/about-adverse-health-effects/low-frequency-noise-infrasound-and-wind-turbines.

[46] Pascal Paillier. Public-key cryptosystems based on composite degree residuosity classes. In *Proceedings of International Conference on the Theory and Application of Cryptographic Techniques*, pages 223–238, Prague, Czech Republic, 1999.

[47] Manas Pathak and Bhiksha Raj. Privacy-preserving speaker verification and identification using gaussian mixture models. *IEEE Transactions on Audio, Speech, and Language Processing*, 21(2):397–406, 2013.

[48] Alessandro Piva, Tiziano Bianchi, and Alessia De Rosa. Secure client-side st-dm watermark embedding. *IEEE Transactions on Information Forensics and Security*, 5(1):13–26, 2010.

[49] Alessandro Piva, Vito Cappellini, D Corazzi, Alessia De Rosa, Claudio Orlandi, and Mauro Barni. Zero-knowledge st-dm watermarking. In *Proceedings of IS&T/SPIE Electronic Imaging*, pages 60720R–60720R. International Society for Optics and Photonics, 2006.

[50] International Telecommunication Union Telecommunications section (ITU-T) Recommendation. Perceptual evaluation of speech quality (pesq), an objective method for end-to-end speech quality assessment of narrowband telephone networks and speech codecs. *ITU-T Recommendation*, page 862, 2001.

[51] Manfred R Schroeder. Natural sounding artificial reverberation. *Journal of the Audio Engineering Society*, 10(3):219–223, 1962.

[52] Adi Shamir. How to share a secret. *Communications of the ACM*, 22:612–613, November 1979.

[53] Madhusudana VS Shashanka and Paris Smaragdis. Secure sound classification: Gaussian mixture models. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, volume 3, pages III–III, Toulouse, France, 2006.

[54] Hosik Sohn, Konstantinos N Plataniotis, and Yong Man Ro. Privacy-preserving watch list screening in video surveillance system. In *Proceedings of the 11th Pacific Rim Conference on Multimedia*, pages 622–632, Shanghai, China, 2010. Springer.

[55] TV Sreenivas and Pradeep Kirnapure. Codebook constrained wiener filtering for speech enhancement. *IEEE Transactions on Speech and Audio Processing*, 4(5):383–389, 1996.

[56] William Stallings. *Cryptography and Network Security Principles and Practice*. Prentice Hall, New York, NY, 5th. edition, 2010.

[57] SV&V. Sv&v media audio database, 2008. http://download.wavetlan.com/SVV/Media/HTTP/http-wav.htm.

[58] Juan Ramón Troncoso-Pastoriza and Fernando Pérez-González. Secure adaptive filtering. *IEEE Transactions on Information Forensics and Security*, 6(2):469–485, 2011.

[59] Juan Ramón Troncoso-Pastoriza and Fernando Pérez-González. Secure adaptive filtering. *IEEE Transactions on Information Forensics and Security*, 6(2):469–485, 2011.

[60] Juan Ramón Troncoso-Pastoriza and Fernando Perez-Gonzalez. Secure signal processing in the cloud: Enabling technologies for privacy-preserving multimedia cloud processing. *IEEE Signal Processing Magazine*, 30(2):29–41, 2013.

[61] Carnegie Mellon University. Cmu speech database, 2007. http://festvox.org/cmu_faf/index.html.

[62] Shinya Washio and Yoshihiro Watanabe. Security of audio secret sharing scheme encrypting audio secrets with bounded shares. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 7396–7400, Florence, Italy, 2014.

[63] Wikipedia. Semantic security, 2015. http://en.wikipedia.org/wiki/Semantic_security.

[64] M Abukari Yakubu, Pradeep K Atrey, and Namunu C Maddage. Secure audio reverberation over cloud. In *Proceeding of 10th Annual Symposium on Information Assurance (ASIA15)*, page 39, New York, USA, 2015.

[65] M Abukari Yakubu, Namunu C Maddage, and Pradeep K Atrey. Audio secret management scheme using shamirs secret sharing. In *Proceedings of 21st International Conference on MultiMedia Modeling*, pages 396–407, Sydney, Australia, 2015.

[66] Andrew Chi-Chih Yao. Protocols for secure computations. In *Proceedings of 23rd IEEE Annual Symposium on Foundations of Computer Science*, volume 82, pages 160–164, Chicago, IL, USA, 1982.

[67] Sungyub D Yoo, J Robert Boston, Amro El-Jaroudi, Ching-Chung Li, John D Durrant, Kristie Kovacyk, and Susan Shaiman. Speech signal modification to increase intelligibility in noisy environments. *The Journal of the Acoustical Society of America*, 122(2):1138–1149, 2007.

[68] Kenta Yoshida and Yoshihiro Watanabe. Security of audio secret sharing scheme encrypting audio secrets. In *Proceedings of 7th International Conference for Internet Technology And Secured Transactions*, pages 294–295, London, UK, 2012.