



Two-Factor Challenge Response Based Authentication

CS416M Project Report - The SHIELD

Author: Vedant Satav
170070012

Ojas Thakur
170070017

Nitish Tongia
170010042

Rishabh Dahale
17D070008

Institute: IIT Bombay

Date: April 25, 2021

Contents

Demo Video

Problem Statement

- Aim and introduction
- Learning Outcomes
- Deliverables

Solution Software Architecture

- GUI
- Voice to text
- Speaker recognition
- Password verification

Architecture Design

- Sign-up Process
- Login Process

Progress Report

- Finished work
 - speech-to-text module (self made)
 - speech and speaker recognition
 - GUI screenshots
- Future work
 - Web application for the integrated system

Contributions

- Work distribution by team members

Final Deliverable

- Speech Processing
- User Interface

Future Projects

- Embellishments with more secure principles
- Protection against different and more complex attacks

References

- Annotated bibliography/URLs of resources

Problem Statement

Aim and Introduction

As the world progresses in the fields of science and technology, traditional security methods shall soon be phased out. Although the idea of voice-activated security has been around since the Arabian Night's Alibaba and the forty thieves, its real world applications have been limited to bank vaults in movies. Conventional login password methods have been replaced by more inventive methods which take into account the uniqueness of certain parameters. Fingerprint and Retinal-scanner based identification systems have been installed in many places. However, we can agree that just speaking out the password is generally more convenient than positioning the eyeball for a retinal scan or placing a greased finger over the scanner. It is due to this reason that more and more applications are being fitted with voice controls. We aim to implement a voice-activated unlocking system which will verify the user-uttered password as well as the authenticity of the visitor.

Learning Outcomes

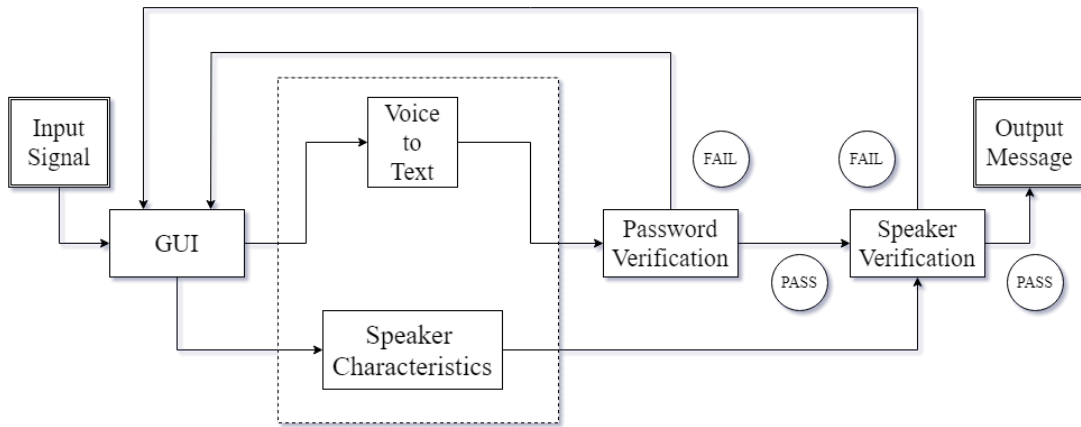
The project implementation would involve a deep study of speech processing algorithms for word and speaker detection, a basic understanding of the use of UI/UX for the user-interface, and knowing the relevance and importance of the use of two-factor authentication with challenge response for security purposes. Through the course of this project, our team will learn how to

- Understanding the security aspects regarding audio based authentication systems
- Develop an end-to-end Speech-to-Text converter system
- Understand import features required for speaker recognition
- Implement a robust Speaker identification system using python libraries
- Combine the two independent blocks to develop a secure system
- A user friendly GUI to access the system

Deliverables

Confusion matrix for the self developed(from scratch), fully operational speech-to-text (STT) conversion system for a finite word vocabulary with a pretty good accuracy is presented in the sections below. A secure two-factor challenge response based audio authentication system is developed and a working demo is attached with the submitted files along with the necessary instructions for running the code.

System Software Architecture



The figure above shows the basic structure of our developed system. Each block separately performs its task and then these individual blocks are combined together to build a secure 2-factor authentication system. Before looking at the system structure in detail, let's look at the task performed by each block briefly:

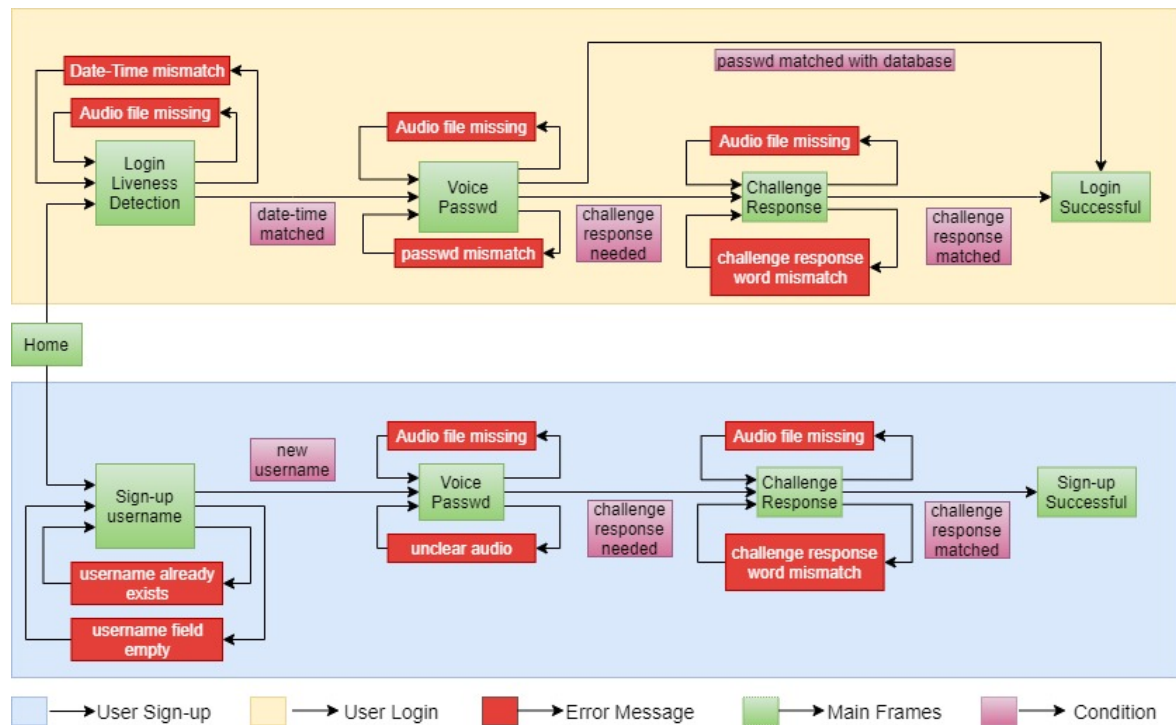
GUI- A user friendly interface, which combines together the speech-to-text block with the speaker verification block and allows the user to securely sign up/login into the system.

Voice to text- This block process the input audio signal provided by the user and converts it into plaintext english words. This text is then compared with the password set by the user which is fetched from the database.

Speaker Characteristics- This block extracts the characteristic features of the speaker from the input audio signal and compares it with all the users from the database to identify the speaker. This block uses the fact that some characteristics like pitch are fundamental to the speaker.

Password and Speaker Verification- These blocks use the standard comparison metrics with thresholds to decide whether the input audio signals corresponds to a genuine user-password combination and not some malicious user trying to access someone else account.

Architecture Design



The figure above shows the detailed system architecture design of our 2-factor audio authentication system with integrated challenge response. The color coding provides the information about the type of frames and in which domain they lie in. The two major processes along with the involved subtleties are explained in the following sections

Sign-up process

First-time users of the product will have to sign-up with the system. The GUI on the first page prompts a button for Sign-up, clicking on which requests the user to provide its username. If the username already exists in the database, it is notified to the user and requests for another username. The next step seeks to take in the intended password for the user, which he/she has to record and submit. This password is converted into plaintext English and stored as a textfile. Next, if it is required, the challenge response for the user is taken as the input. The user has to record reading out a randomly generated word, which is then converted into plaintext English. If the challenge response is matched correctly, the sign-up is successful.

If at any point the input voice signal is weak or if there is any error in storing the audio file, the user is prompted to submit another sample. This concludes the sign-up process.

Log-in process

The log-in process is for those users who have already signed-up in the system. The system prompts the user to read out the time and the date as a means for liveness detection. This is done so that the product is safe from replay attacks. Only once the details are matched than the user is taken forward to the password authentication section- the user would have to read-out his/her password. If the password is directly matched with sufficient confidence, the log-in process is successful. If not, the user is taken to a challenge response.

If at any instance the audio file storage faces issues or if the audio input is unclear, the user is prompted to submit another sample.

Progress report

As mentioned in the midterm report, we first went on to improve the accuracy of the speech-to-text module as well as doubling up the password vocabulary. But as per the feedback obtained from the professor and the teaching assistants, we shifted our focus towards the security aspects of our system. To blend in protective measures in our system like replay attacks, a much wider password vocabulary and much more robust speech and speaker recognition systems were required. Hence we move on to using built-in python libraries for these tasks which works well for a much more general domain. Then we combined these two modules through a user friendly graphical user interface to develop a fully operational 2-factor audio authentication challenge response based system.

Finished work

Speech to Text Module (Self-made)

Building up on the previous work, the first step that we undertook for improving the accuracy of the model was to pre-process the audio signal using a energy based end-pointing block before extracting the features. This block removes the unnecessary low energy silences at the start and the end of the signal which ensures that the extracted features represent the uttered audio much more efficiently. To further improve the accuracy, we implemented a LSTM model in place a phone-hmm model for each word. This also helped in doubling up the password vocabulary while maintaining a much better accuracy. The performance of this model can be judged on the basis of confusion matrices obtained for the clean and noisy test data which are shown below.

Words	Down	Go	Left	No	Off	on	right	stop	up	yes
Down	229	10	1	4	0	0	0	1	0	2
Go	2	231	0	4	1	0	0	0	0	0
Left	0	1	258	0	0	0	0	0	0	2
No	1	10	0	231	0	0	0	1	1	1
Off	0	1	1	0	253	0	0	0	2	0
On	1	0	1	0	0	236	0	0	2	0
right	0	0	1	0	1	0	248	0	1	0
stop	1	3	1	0	1	0	0	238	1	0
up	0	4	1	0	3	0	0	0	263	0
yes	0	1	0	0	0	0	0	0	0	248

Table 1: Confusion matrix for Clean test data (Accuracy = 0.972)

Words	Down	Go	Left	No	Off	on	right	stop	up	yes
Down	181	34	1	6	2	0	1	7	1	14
Go	2	212	2	7	6	2	1	1	0	5
Left	0	5	214	0	9	0	6	2	7	18
No	6	54	4	161	5	0	0	3	3	9
Off	1	1	0	0	248	1	0	1	5	0
On	0	4	3	0	44	177	3	1	7	0
right	1	3	5	1	4	0	227	0	4	6
stop	3	17	2	0	16	0	0	194	10	3
up	0	2	2	0	37	0	0	2	225	0
yes	0	3	7	0	2	0	0	2	1	234

Table 2: Confusion matrix for Noisy test data (Accuracy = 0.829)

Speech and Speaker recognition - As mentioned earlier, to bring in the security aspects, a much more robust STT system with a much larger vocabulary was required. Hence we shifted to using built in libraries for speech and speaker recognition system. The necessary details regarding these libraries are provided in the references section.

GUI screenshots

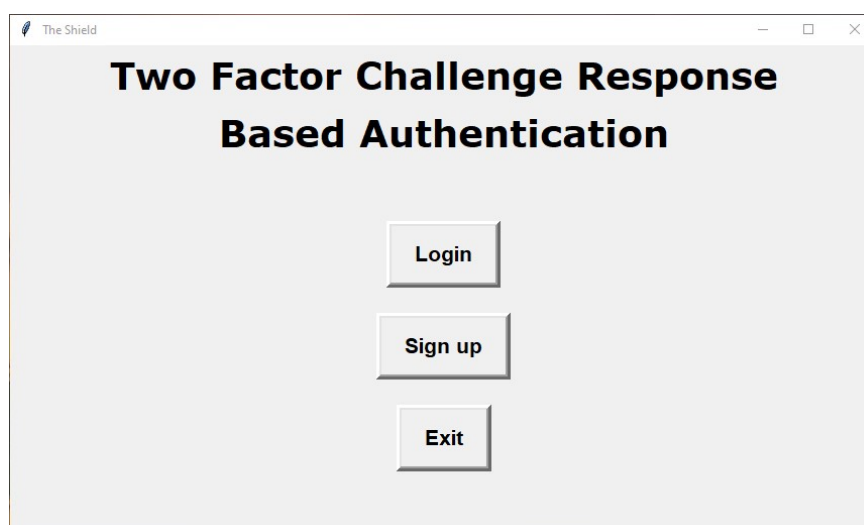
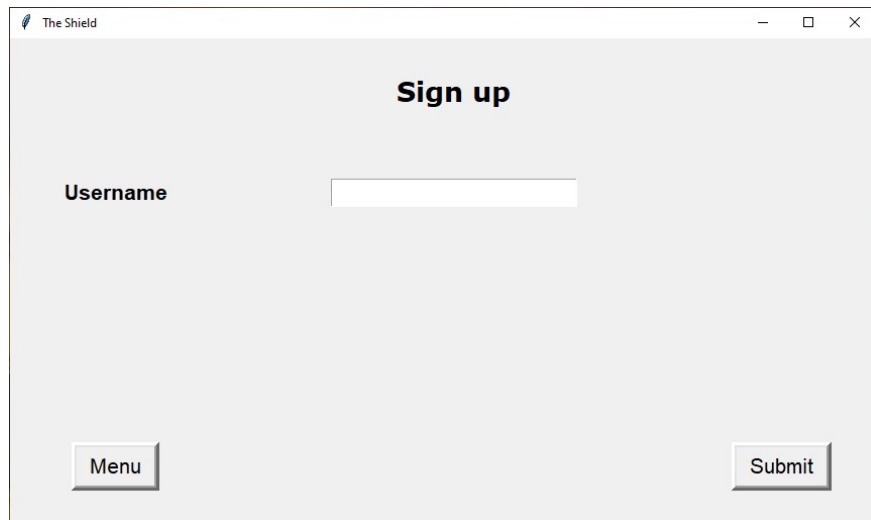


Figure 1: Home page



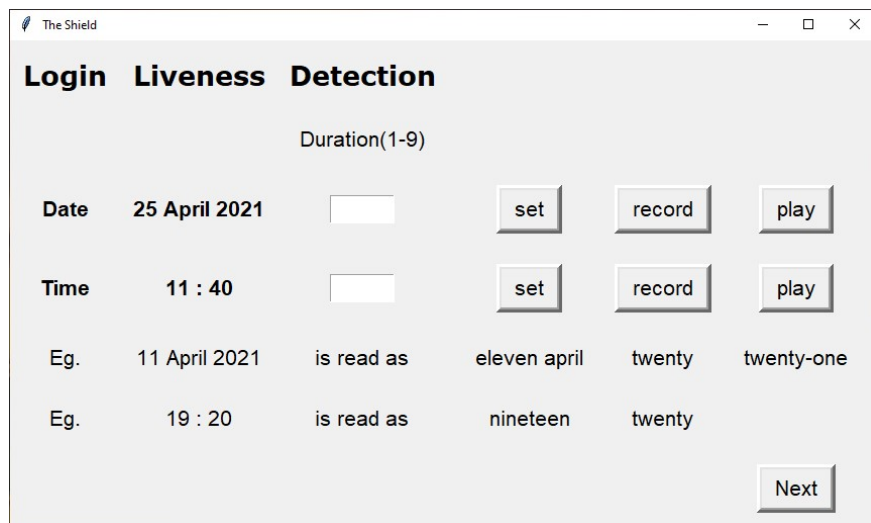
The Shield

Sign up

Username

Menu Submit

Figure 2: Signup Page



The Shield

Login Liveness Detection

Duration(1-9)

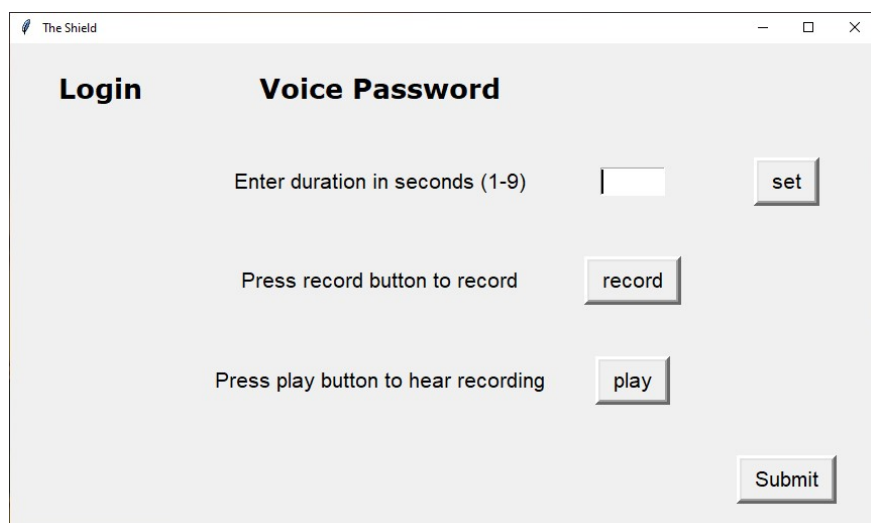
Date	25 April 2021	<input type="text"/>	set	record	play
Time	11 : 40	<input type="text"/>	set	record	play

Eg. 11 April 2021 is read as eleven april twenty twenty-one

Eg. 19 : 20 is read as nineteen twenty

Next

Figure 3: Liveness Detection



The Shield

Login Voice Password

Enter duration in seconds (1-9) set

Press record button to record record

Press play button to hear recording play

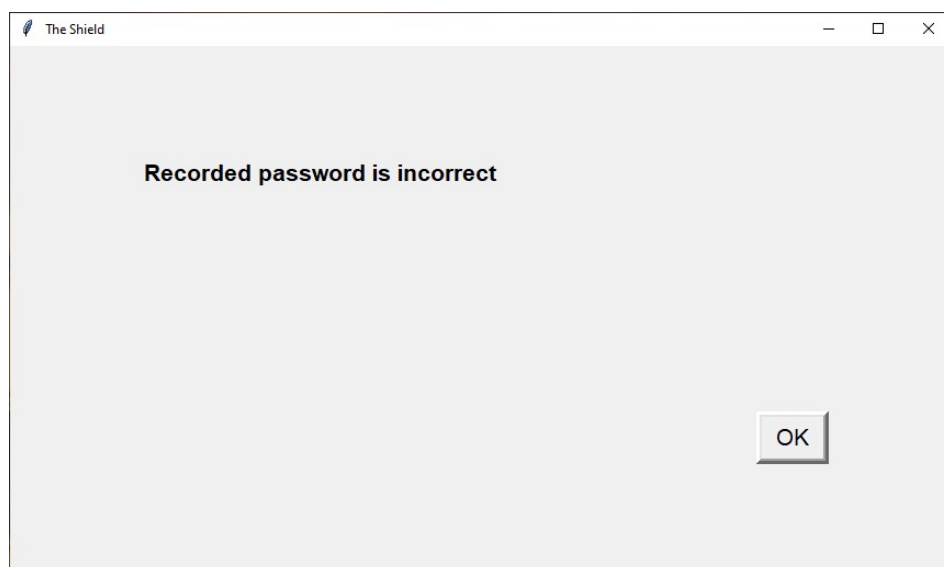
Submit

Figure 4: Voice input



The screenshot shows a window titled "The Shield" with a standard Windows-style title bar (minimize, maximize, close buttons). The main content area has a light gray background. At the top, the text "Challenge Response" is displayed in a bold, black font. Below this, there is a label "Enter duration in seconds (1-9)" followed by a small, empty text input field. To the right of the input field is a button labeled "set". Further down, there is a label "Press record and read out :" followed by two buttons: "record" and "play". Below these buttons, the word "Left" is centered. At the bottom right of the window is a button labeled "Submit".

Figure 5: Challenge Response



The screenshot shows the same "The Shield" window, but the content area now displays the message "Recorded password is incorrect" in a bold, black font. At the bottom right of the window, there is a button labeled "OK".

Figure 6: Error Page

Future work

The immediate step forward would be to develop a web-application for accessing the system which operates on the front-end and maintains a database of sensitive information like passwords and user identity on the back-end. This would ensure that the data is protected and stored at a safe place and is less vulnerable to be compromised. But this comes with added requirements of protection of data while being sent to and from back and front end like encrypting and integrity protecting the fetched password while being sent to the back end server over a insecure communication network. The system itself can be further improved in several aspects like to be robust in presence of background noise or correctly identifying the speaker even when he/she has cough/cold etc.

Contributions

As mentioned in the midterm report, our team worked together on more or less every aspect of the 2-factor authentication system or the project itself. But, for the sake of completeness, here is a broad distribution of major contributions provided by each team member individually-

Vedant Satav (170070012) - Worked primarily on the self developed speech-to-text block and the literature review regarding the characteristic features for the speaker recogniser block. Came up with a majority of aspects regarding improving the model's accuracy, increasing the password vocabulary and increasing the robustness of the model

Ojas Thakur (170070017) - Worked on developing the user friendly graphical user interface (GUI), combining the speech-to-text and the speaker recogniser modules to build a secure system and build the platform from where the user can access the system. Worked on the architecture of the system to provide protection against security attacks.

Nitish Tongia (170010042) - Worked on developing the speech-to-text block and on implementing the required improving aspects for obtaining a better accuracy. Implemented the necessary amendments in the GUI architecture for liveliness detection and protection against replay attacks.

Rishabh Dahale (17D070008) - Worked on reviewing the literature for robust speaker recognition and implementing and integrating the libraries used for both the independent modules with the user accessible platform. Integrated the challenge response based authentication to the system.

Final Deliverable

The project itself can be said to compose of two different sections:

1. Speech processing for word and speaker recognition
2. User Interface for wrapping the speech processing components

Speech Processing

The speech processing component gets its audio from the user interface and will carry out word and speaker recognition and will deliver a boolean true signal when the password and speaker are found to be a match, or a request to the UI to generate an alternate audio request to the user because the input audio failed to cross the threshold to be recognised with confidence.

User Interface

This component of the project again has two parts :

SIGN UP :

1. A new user will first enter the "username". If the username is already used, then system will prompt for another username.
2. After successful username entry, system will ask for a spoken password. Inside the database, the new username and password will be stored.
3. Then the user will be prompted to speak out another word as challenge response, to have more samples of the user's voice.
4. After this, the sign-up process is complete.

LOGIN :

1. The user will first be prompted to record the current date and time to prevent replay attacks and for liveliness detection.
2. If the recorded date and time match, user is prompted to the voice password frame. Speaker is also recognized using these recorded audio files and we extract corresponding password from database.
3. The extracted password is compared with the recorded password. Speaker is once more determined using recorded files. When both the determined speakers match and the password is correct, then login procedure is complete.
4. If the threshold for speaker recognition is not crossed with any existing speaker, we opt for challenge response and once again try to determine the speaker and match it with previously determined speakers. We also check if the challenge response word is spoken correctly.
5. If the challenge response matches, login procedure is complete.

Future Projects

Interesting variations and extensions of our project:

A. Embellishments with more secure principles

The proposed voice-activated access system with two-factor authentication and challenge response can be embellished with more secure principles like:

1. Keeping the hardware components like, speaker and display separate from the intelligent part of the product. This will ensure lesser duplicate copies of the software ensuring lesser security threat. This method would require some secure communication channel between the hardware and the brain that would also need to be energy efficient
2. Implementing better speech and speaker recognition algorithms to increase the accuracy of the detection

The first part would require in-depth practical knowledge about transfer protocols, encryption methods and their energy efficiencies- it would be a separate project in itself and would require significant time management.

B. Protection against different and more complex attacks

The previous idea can be extrapolated to cover principles of network security by connecting the different hardware units connected in a network, sharing information among themselves, in a vast array of access doors. Such a system would also require knowledge about cryptography to ensure security of the network. This project off-shoot would require a lot of team work among the members but not much complexity. The students can begin with learning the basic network systems, communication protocols and their security.

References

GUI

For developing the Graphical User Interface (GUI), **tkinter** library of python was used.

A couple of really good references for the same are :

1. [Basic GUI programming](#)
2. [Tkinter Tutorial](#)

Speech to text

The speech processing aspects of the project were sourced from the EE679 - Speech processing course undertaken by prof. Preeti Rao from the Electrical Engineering department. Some of the useful links that complete the understanding:

1. [MFCC Feature extraction](#)
2. [GMM-HMM](#)
3. [Spectral Analysis of Speech by Linear Prediction](#)

Speaker Recognition

In the extensive review we performed to get an idea about performing robust speaker recognition, we found a couple of references which were really good for understanding the overall challenges in this task:

1. [Features and Techniques in speaker recognition](#)
2. [Overview of different techniques for speaker recognition](#)
3. [Mean Hilbert Envelope Coefficients \(MHEC\)](#)

Python Libraries

To incorporate security aspects into our authentication system, we used built in python libraries for speech-to-text and speaker recognition modules and focused more on protection mechanisms in the final product. The formal documentation of the libraries can be found here:

1. [Speaker Verification Toolkit](#)
2. [Speech Recognition Toolkit](#)