

## Case study 6

**Topic:** Implementation of a voice identification/authentication system

**Purpose of the work:** to develop students' skills in implementing voice-based identification/authentication processes.

### Theoretical part

Biometric authentication, unlike password- or token-based authentication, uses unique biological characteristics to verify an individual. It is more difficult to spoof and more convenient for users because they do not have to remember passwords or carry a physical token that can be easily lost or stolen. The authenticator is a part of the person. Voice recognition (also called speaker recognition or voice authentication) uses analysis of a person's voice to verify an individual. The airway and soft tissue cavities, as well as the shape and movement of the mouth and jaw, affect the voice patterns to create a unique "voiceprint."

Voice authentication systems are based on voice characteristics, such as pitch, modulation, and frequency, that are unique to each person. Voice recognition is different from speech recognition. While speech recognition technology can identify the words of a subscriber, voice recognition technology verifies the identity of the speaker. Speaker identification has some limitations. Different people can speak with similar voices, and the voice of any person can change over time depending on their mood, emotional state, and age. In addition, the variety of telephone devices and the quality of telephone connections make it difficult to identify the speaker. Therefore, voice recognition is considered to be useful in combination with other biometric systems, such as facial recognition or fingerprint recognition.

#### *Voice authentication :*

- *Attack* When your system stores voice data, it does so like any other data: on a server or in a database. If this point is not secured, hackers can steal the data and use it to create fake records that match it.
- *Unchangeable* . If biometric data is compromised, it cannot be changed like a password, because you can't just change your voice. Not 100 % accurate. No authentication method is foolproof. The same goes for speech recognition. However, advances in artificial intelligence and anti-fraud have made biometrics viable even for secure applications like payment processing. However, one form of identification should not be relied upon.
- *Not suitable for all environments* . Requires a very quiet environment for speech recognition. Sound artifacts other than the user's voice can interfere with authentication, which can cause problems if you use voice only.

#### *Risks for voice authentication:*

- *Voice changes can affect logins*. Users who rely on facial recognition to unlock their phones quickly realized how using one method during the pandemic (and requiring them to wear a mask) can limit logins. Noisy environments can

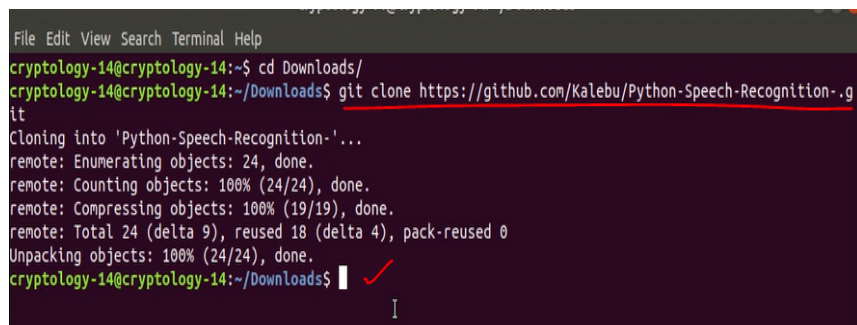
affect authentication, but common triggers include a cold, a sore throat, or minor changes in voice, accent, or speech pattern.

- *Technical voice masking.* New tools can modulate voices to resemble other voices and, in some cases, spoof biometrics. There is a constant back and forth as security systems outpace these technologies, as voice tampering becomes more common.

- *Determining survivability.* Since voice technology is somewhat easier to hack than other biometrics, it requires survivability tests to ensure that it is a real user and not a fraud machine.

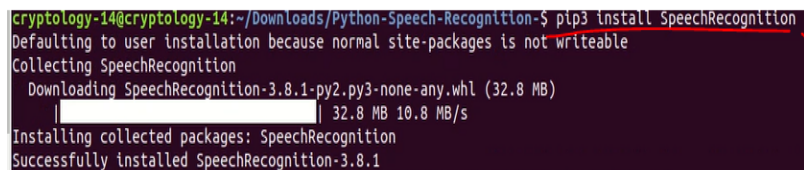
## Practical part

*In the first step, enter the command " cd Downloads " in the terminal, and if you have an audio file, it will take a long time to download and process it. Therefore, the best way is to divide the long audio source into small pieces from the file and then perform speech recognition on these pieces. The following window is where the command is entered for this case, the result is given in the following window.*

A terminal window with a dark background and light green text. The prompt is 'cryptology-14@cryptology-14:~\$'. The user enters 'cd Downloads/'. The prompt changes to 'cryptology-14@cryptology-14:~/Downloads\$'. The user enters 'git clone https://github.com/Kalebu/Python-Speech-Recognition.git'. The terminal shows the progress of cloning the repository, including enumerating, counting, and compressing objects, and finally unpacking them. A red checkmark is visible next to the final prompt.

```
File Edit View Search Terminal Help
cryptology-14@cryptology-14:~$ cd Downloads/
cryptology-14@cryptology-14:~/Downloads$ git clone https://github.com/Kalebu/Python-Speech-Recognition.git
Cloning into 'Python-Speech-Recognition'...
remote: Enumerating objects: 24, done.
remote: Counting objects: 100% (24/24), done.
remote: Compressing objects: 100% (19/19), done.
remote: Total 24 (delta 9), reused 18 (delta 4), pack-reused 0
Unpacking objects: 100% (24/24), done.
cryptology-14@cryptology-14:~/Downloads$
```

Figure 6.1. Selecting a file

A terminal window with a dark background and light green text. The prompt is 'cryptology-14@cryptology-14:~/Downloads/Python-Speech-Recognition\$'. The user enters 'pip3 install SpeechRecognition'. The terminal shows the progress of installing the package, including defaulting to user installation, collecting the package, downloading it, and finally installing it successfully. A red checkmark is visible next to the final prompt.

```
cryptology-14@cryptology-14:~/Downloads/Python-Speech-Recognition$ pip3 install SpeechRecognition
Defaulting to user installation because normal site-packages is not writeable
Collecting SpeechRecognition
  Downloading SpeechRecognition-3.8.1-py2.py3-none-any.whl (32.8 MB)
    | 32.8 MB 10.8 MB/s
Installing collected packages: SpeechRecognition
Successfully installed SpeechRecognition-3.8.1
```

Figure 6.2. The Pydub command window

*The example " app.py " is implemented with this code to detect speech in sound transmitted directly from a microphone:*

```
cryptology-14@cryptology-14:~/Downloads/Python-Speech-Recognition-$ python3 app.py
ALSA lib pcm_dsnoop.c:618:(snd_pcm_dsnoop_open) unable to open slave
ALSA lib pcm_dmix.c:1052:(snd_pcm_dmix_open) unable to open slave
ALSA lib pcm.c:2495:(snd_pcm_open_noupdate) Unknown PCM cards.pcm.rear
ALSA lib pcm.c:2495:(snd_pcm_open_noupdate) Unknown PCM cards.pcm.center_lfe
ALSA lib pcm.c:2495:(snd_pcm_open_noupdate) Unknown PCM cards.pcm.side
ALSA lib pcm_dmix.c:1052:(snd_pcm_dmix_open) unable to open slave
Adjusting noise
Recording for 4 seconds
Done recording
Recognizing the text
```

Figure 6.3. Speech recognition

*The second example uses this code to detect speech from audio loaded from a local audio file, "app\_audio.py":*

```
cryptology-14@cryptology-14:~/Downloads/Python-Speech-Recognition-$ python3 app_audio.py
Done recording
Recognizing the text
Decoded Text : Python Programming is the best of all by Jordan
```

Figure 6.4. Speech recognition code from audio loaded from an audio file

" app.py " is entered again , reason How is it now said in a sentence, for example, "hello how are you" 4 It will convert this word into text in a matter of seconds, and you can see the result in the window below.

```
cryptology-14@cryptology-14:~/Downloads/Python-Speech-Recognition-$ python3 app.py
ALSA lib pcm_dsnoop.c:618:(snd_pcm_dsnoop_open) unable to open slave
ALSA lib pcm_dmix.c:1052:(snd_pcm_dmix_open) unable to open slave
ALSA lib pcm.c:2495:(snd_pcm_open_noupdate) Unknown PCM cards.pcm.rear
ALSA lib pcm.c:2495:(snd_pcm_open_noupdate) Unknown PCM cards.pcm.center_lfe
ALSA lib pcm.c:2495:(snd_pcm_open_noupdate) Unknown PCM cards.pcm.side
ALSA lib pcm_dmix.c:1052:(snd_pcm_dmix_open) unable to open slave
Adjusting noise
Recording for 4 seconds
Done recording
Recognizing the text
Decoded Text : hello how are you
```

Figure 6.5. The word "Hello how are you" appears in the terminal as text

## **Assignment**

Convert words to text format on Windows and Linux OS using the above commands.

## **Control questions**

1. What is the function of a voice identification/authentication system?
2. voice transcription work ?