

SecureAuth

SecureAuth este o aplicație de securitate care utilizează recunoașterea facială și vocală pentru autentificarea utilizatorilor. Aceasta permite utilizatorilor să se înregistreze cu informațiile lor personale, să înregistreze un videoclip cu fața lor și un fișier audio cu vocea lor, și să se autentifice folosind aceste înregistrări. SecureAuth oferă un nivel ridicat de securitate, asigurându-se că doar utilizatorii autorizați pot accesa serverul.

Funcționalități

1. Înregistrare Utilizator:

- Utilizatorul introduce numele, prenumele și parola.
- Informațiile sunt salvate într-un fișier JSON (`users.json`).
- Taburile pentru înregistrarea video și audio sunt activate doar după ce un utilizator este înregistrat.

2. Înregistrare Video Utilizator:

- Utilizatorul înregistrează un videoclip de 3 secunde cu fața sa.
- Videoclipul este salvat local cu un nume unic care include numele și prenumele utilizatorului (`user_video_nume_prenume.avi`).

3. Înregistrare Audio Utilizator:

- Utilizatorul înregistrează un fișier audio de 5 secunde cu vocea sa.
- Audio este convertit în text folosind Google Speech Recognition.
- Fișierul audio și textul recunoscut sunt salvate local (`user_audio_nume_prenume.wav` și `user_audio_text_nume_prenume.txt`).

4. Autentificare Utilizator:

- Utilizatorul este autentificat prin recunoaștere facială și vocală.
- Camera captează un cadru pentru recunoașterea facială și modelul antrenat este utilizat pentru a confirma identitatea.
- Microfonul captează vocea și textul recunoscut este comparat cu parola stocată.

5. Conectarea la Server:

- Dacă autentificarea este reușită, un mesaj este trimis la server indicând că utilizatorul s-a conectat.
- Serverul confirmă primirea mesajului.

Detalii Tehnice

Antrenarea Modelului de Recunoaștere Facială

- Utilizăm modelul `LBPHFaceRecognizer` din OpenCV pentru recunoașterea facială.
- Modelul este antrenat cu cadrele extrase din videoclipul înregistrat al utilizatorului.
- Fiecare cadru este convertit în grayscale înainte de antrenare pentru a simplifica procesarea și a îmbunătăți acuratețea recunoașterii.
- Antrenarea se face folosind cadrele și o etichetă comună pentru toate cadrele (0 în acest caz), deoarece fiecare videoclip este asociat cu un singur utilizator.

Recunoașterea Facială

- Modelul antrenat este folosit pentru a prezice ID-ul și nivelul de încredere pentru fața capturată în timpul autentificării.
- Dacă nivelul de încredere este sub un prag prestabilit (ajustat la 70), autentificarea video este considerată reușită.

Recunoașterea Vocală

- Folosesc biblioteca `speech_recognition` pentru a capta și converti audio în text.
- Google Speech Recognition este utilizat pentru transcrierea audio în text.
- Textul recunoscut este comparat cu parola introdusă inițial de utilizator și cu parola stocată în fișierul `parole.txt`.

Gestionarea Fișierelor

- `users.json`: Stochează informațiile utilizatorilor în format JSON.
- `parole.txt`: Stochează parolele utilizatorilor într-un format text simplu (nume, prenume, parola).
- Fișiere video și audio: Salvează videoclipurile și fișierele audio cu nume unice care includ numele și prenumele utilizatorului (`user_video_nume_prenume.avi`, `user_audio_nume_prenume.wav`, `user_audio_text_num_prenume.txt`).

Cum să Rulați Aplicația

1. Instalare Dependințe:

Asigurați-vă că aveți instalate următoarele biblioteci:

```
pip install numpy opencv-python-headless tkinter speechrecognition
```

2. Rularea Aplicației:

- Salvați codul aplicației într-un fișier `client.ipynb` și rulați-l folosind Jupyter Notebook sau Jupyter Lab.

3. Serverul de Autentificare:

Rulați serverul de autentificare pe mașina locală. Serverul trebuie să fie configurat pentru a asculta pe portul 65432. Codul serverului arată astfel:

```
import socket

server_ip = '127.0.0.1'
server_port = 65432

with socket.socket(socket.AF_INET, socket.SOCK_STREAM) as s:
    s.bind((server_ip, server_port))
    s.listen()
    print(f"Serverul ascultă pe {server_ip}:{server_port}...")

    while True:
        conn, addr = s.accept()
        with conn:
            print('Conexiune de la', addr)
            while True:
                data = conn.recv(1024)
                if not data:
                    break
                print('Received', data.decode('utf-8'))
                conn.sendall(b'Mesajul a fost primit')
```

4. Interacțiune cu Aplicația:

- **Înregistrare Utilizator:** Introduceți numele, prenumele și parola în tabul "Înregistrare Utilizator".
- **Înregistrare Video:** Mergeți la tabul "Înregistrare Video Utilizator" și înregistrați un videoclip de 3 secunde.
- **Înregistrare Audio:** Mergeți la tabul "Înregistrare Audio Utilizator" și înregistrați un fișier audio de 5 secunde.
- **Autentificare:** Mergeți la tabul "Logare Server" și autentificați-vă folosind recunoașterea facială și vocală. Dacă autentificarea este reușită, un mesaj va fi trimis la server.