**MAC Keylogger Project**

**What is a keylogger?**

A **keylogger** (short for *keystroke logger*) is a type of surveillance tool or software that records the keys pressed on a keyboard in real time. In cybersecurity research and system monitoring, keyloggers can be used for legitimate purposes such as usability testing, employee activity auditing, or parental control. However, they are also commonly associated with malicious intent when used to capture sensitive data such as passwords, credit card numbers, or private messages without user consent

.

This project implements a **software-based keylogger** in Python that runs in the background and logs each keystroke made by the user.

Key features of the implementation include:

* **Real-time keystroke capture** using a keyboard listener.
* **Timestamped logging** for tracking when keys were pressed.
* **Stealth mode** to operate with minimal visibility to the user.
* **Optional file export** to store captured data for analysis.

**Legal and Ethical Notice:**

This tool must only be used on systems you own or have explicit permission to monitor. Unauthorized use of keyloggers is illegal and unethical. This is intended for educational purposes such as security research or learning how attackers operate so you can better defend systems.

**📁 Project Structure**

**Root Directory**

* main.py — Entry point; coordinates module execution and task scheduling.
* config.py — Centralized settings for email, encryption, and timing.
* requirements.txt — Lists all Python dependencies.
* README.md — Project overview, installation steps, and ethical guidelines.
* .gitignore — Prevents tracking of sensitive or generated files.

**modules/**

Modular components for different data collection and transmission tasks:

* logger.py — Logs all keystrokes using the pynput library.
* system\_info.py — Gathers hostname, IP, OS version, and user details.
* clipboard.py — Extracts the current clipboard content.
* microphone.py — Records audio from the default mic and saves as .wav.
* screenshot.py — Captures a screenshot of the desktop.
* emailer.py — Sends collected logs via SMTP email.
* encryptor.py — Encrypts files using cryptography.fernet.
* timer.py — Runs periodic tasks (e.g., email sending, screenshots).

**logs/**

Stores all captured and generated data:

* keylog.txt — Raw keystroke data.
* systeminfo.txt — Device and system metadata.
* clipboard.txt — Clipboard history logs.
* audio.wav — Microphone audio recordings.
* screenshot.png — Screenshots of desktop activity.
* encrypted\_data/ — Holds encrypted copies of logs before emailing.

IDE: Visual Studio Code

Pip : Standard package manager for Python. For any libraries that fall outside of the Python standard library, we will use pip to install and manage these libraries.

Pip Modules Needed:

pynput - used for controlling and monitoring input devices (ex, mouse and keyboard)

pyperclip – accesses clipboard data

pyautogui – takes screenshots

scipy – open source library for scientific and technical computing

cryptography - The standard library for cryptography in python

sounddevice- allows users to play and record audio using numpy arrays.

pip install pynput

pip install pyperclip

pip install sounddevice scipy

pip install pyautogui

pip install cryptography

config.py

app password

gmail:

1.sign in

2. make sure two-factor authentication is enable

3. search app password

4. add app name ex: Keylogger

5. copy and paste the 16 charcater password into “email\_password”