**GÉNIE INFORMATIQUE**

ÉTABLISSEMENT D’ENSEIGNEMENT SUPÉRIEUR TECHNIQUE PRIVÉ

SAS ANAPIJ - Lieu de l’établissement : 242 Rue du Faubourg Saint-Antoine, 75012 Paris

Siret : 531 458 669 00045 - APE : 8559B - Déclaration d’activité : 117 556 694 75 - UAI : 0754724C

Une image contenant Police, Graphique, capture d’écran, graphisme

Description générée automatiquement

PRESENTE PAR :

NGUEDIA TSAGUE ARTHUR

**CONCEPTION D’UNE APPLICATION ESPION**

# Introduction

la création d'un keylogger en Python qui permet l’espionnage de la machine d’une permet dans le déroulement de mon sujet je développerais premièrement l’étape de l’initialisation des Variables et Création des Fichiers ,la capture des frappes de touches au clavier, deuxièmement de prend des captures d'écran, et en fin l’envoie de ces données par email via SendGrid.

# Initialisation des Variables et Création des Fichiers

### 2. Initialisation des Variables et Création des Fichiers

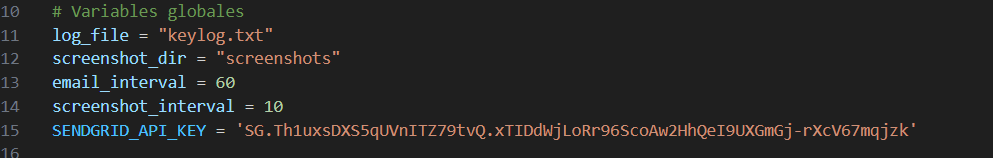
Cette partie du code prépare les fichiers et les dossiers nécessaires au bon fonctionnement du keylogger.

### 2.1 Variables Globales

* log\_file : Définit le fichier où seront enregistrées les frappes clavier.
* screenshot\_dir : Répertoire où seront stockées les captures d'écran.
* email\_interval : Intervalle de temps (en secondes) entre chaque envoi d'email.
* screenshot\_interval : Intervalle de temps entre chaque capture d'écran.
* SENDGRID\_API\_KEY : Clé API pour le service SendGrid, utilisé pour l'envoi des emails.
* L’email doit être envoyé toute les 60 seconde et les capture toute les 10 seconde

### 2.2 Vérification et Création des Fichiers

Le code vérifie si le fichier keylog.txt et le répertoire screenshots existent. S'ils n'existent pas, ils sont créés automatiquement.



# Capture des Frappes Clavier

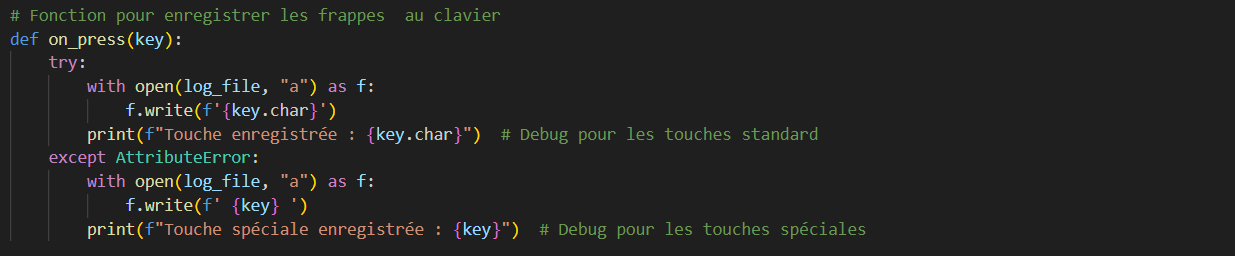
Le keylogger enregistre chaque touche pressée par l'utilisateur et les sauvegarde dans un fichier texte.

### 3.1 Fonction on\_press

Cette fonction capture les frappes de clavier à l'aide du module **pynput**.

Les touches standards (lettres, chiffres) sont ajoutées directement dans le fichier **keylog.txt**.

Les touches spéciales (comme **Shift** ou **Ctrl**) sont enregistrées sous forme d'étiquette.



### 3.2 Utilisation de pynput

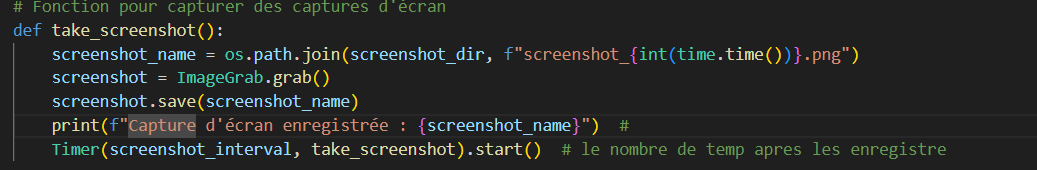
Le module **pynput** permet d'écouter les événements de clavier et d'exécuter des actions en fonction des touches pressées.

# IV Captures d'Écran

Le keylogger prend des captures d'écran à intervalle régulier et les stocke dans un répertoire.

### 4.1 Fonction take\_screenshot

* Utilise le module **Pillow (PIL)** pour capturer l'écran avec la méthode **ImageGrab.grab()**.
* Les captures d'écran sont nommées avec un timestamp pour éviter les conflits de noms et sont enregistrées dans le répertoire **screenshots**.

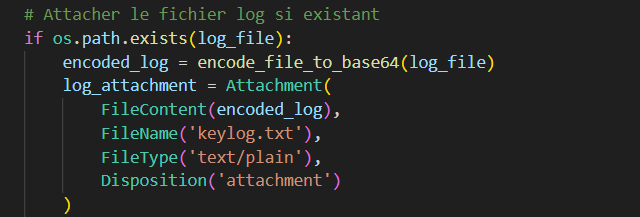


# V Envoi des Données par Email

Vu la reconnaissance en deux étapes instore par google nous avons choisi que Les données (fichier de logs et captures d'écran) sont envoyées périodiquement par email via le service **SendGrid**.

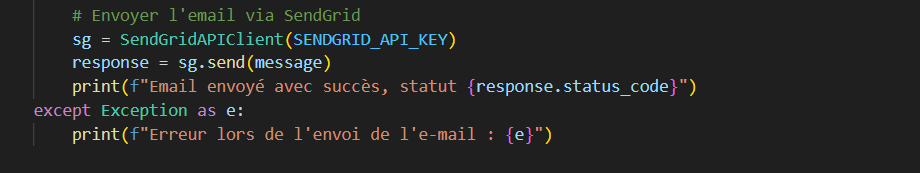
## 5.1 Fonction encode\_file\_to\_base64

* Cette fonction encode les fichiers en **Base64** pour qu'ils puissent être envoyés en tant que pièces jointes dans un email.



### 5.2 Fonction send\_email

* Crée un objet **Mail** avec l'API SendGrid.
* Attache le fichier de logs **keylog.txt** et les captures d'écran au mail.
* Envoie l'email via l'API **SendGrid**.



Une image contenant texte, capture d’écran, Police

Description générée automatiquement

# **VI Démarrage et Coordination des Fonctions**

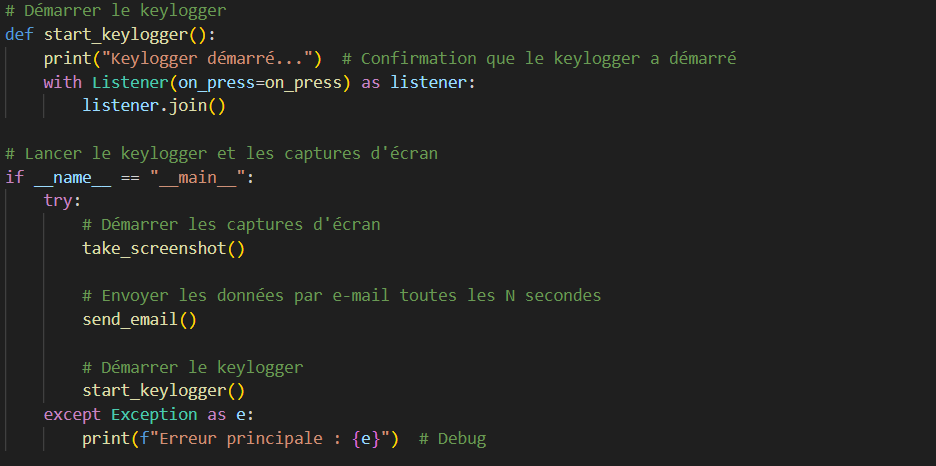
Cette partie coordonne les trois principales fonctionnalités : captures d'écran, keylogging, et envoi des emails.

### 6.1 Fonction start\_keylogger

* Cette fonction démarre l'écoute des frappes clavier et reste active tant que le programme est en cours d'exécution.

### 6.2 Bloc Principal

* **take\_screenshot()** démarre les captures d'écran périodiques.
* **send\_email()** démarre l'envoi des emails périodiques.
* **start\_keylogger()** lance l'écoute des touches du clavier.



# VII Conclusions

Ce projet de keylogger en Python permet de capturer les frappes clavier, de prendre des captures d'écran à intervalles réguliers et d'envoyer ces données par email via l'API SendGrid. Il fonctionne bien, mais plusieurs améliorations sont possibles. Parmi elles, on peut renforcer la sécurité en chiffrant les données, optimiser les performances en réduisant la fréquence des captures d'écran, et ajouter des fonctionnalités comme une interface utilisateur pour ajuster les paramètres, l'envoi à plusieurs destinataires ou la compatibilité multiplateforme. Ces améliorations rendraient le programme plus efficace, sécurisé et flexible tout en respectant les règles d'éthique et de légalité.