**Keylogger Application using Python**

1. **Introduction:**

A keylogger, short for keystroke logger, is a type of surveillance technology used to monitor and record each keystroke typed on a computer keyboard. This can be done either through software or hardware mechanisms. Keyloggers are often employed for legitimate purposes, such as monitoring employee activities in a corporate environment, parental control, or conducting usability studies to improve software interfaces. However, keyloggers are also frequently associated with malicious activities, such as unauthorized access to sensitive information, cyber espionage, and other forms of data theft.

1. **Architecture Outline:**
2. **Key Generation and Management:**

**generate\_key**: Generates a new encryption key using the cryptography library and saves it to a file (key.key).

1. **Keystroke Capture:**

* Uses the **pynput** library to capture keyboard events.
* **on\_press**: Event handler that triggers on each key press, captures the keystroke, encrypts it, and stores it.

1. **Keystroke Encryption:**

* **encrypt\_keystroke**: Encrypts each captured keystroke using the Fernet symmetric encryption provided by the cryptography library.
* **load\_key**: Loads the encryption key from the file key.key.

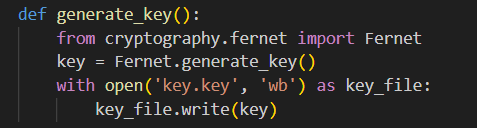
1. **Keystroke Storage:**

* **store\_keystroke**: Writes the encrypted keystroke to a file (encrypted\_keystrokes.txt).

1. **Decryption of Stored Keystrokes:**

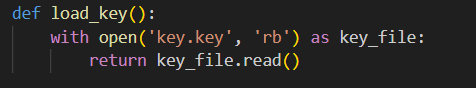
* **decrypt\_file**: Reads the encrypted keystrokes from the file, decrypts them, and writes the decrypted keystrokes to another file (decrypted\_keystrokes.txt).

1. **Detailed Methods:**
2. **generate\_key**:



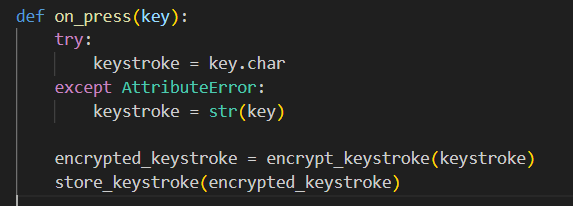
* Generates a new symmetric encryption key.
* Saves the key to a file named key.key.

1. **load\_key**:



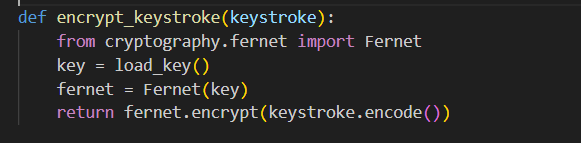
* Reads the encryption key from key.key and returns it.

1. **on\_press**:



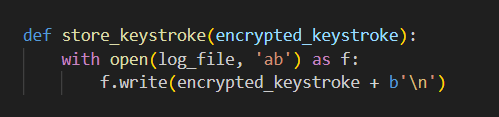
* Captures the keystroke and checks if it is a character or a special key.
* Encrypts the captured keystroke.
* Stores the encrypted keystroke.

1. **encrypt\_keystroke**:



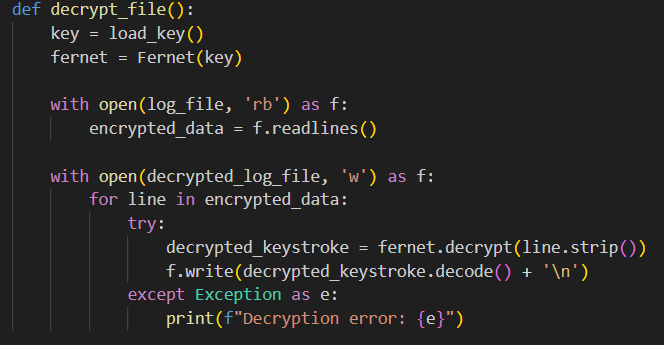
* Loads the encryption key.
* Encrypts the keystroke using the Fernet symmetric encryption scheme.
* Returns the encrypted keystroke.

1. **store\_keystroke**:



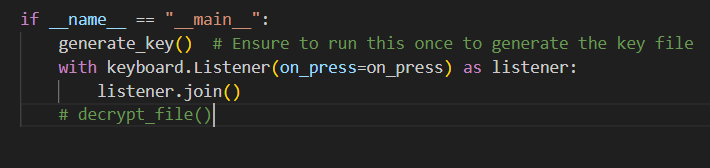
* Appends the encrypted keystroke to encrypted\_keystrokes.txt.

1. **decrypt\_file**:

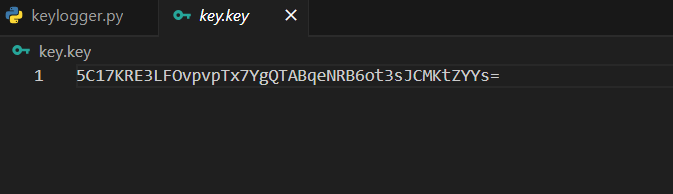


* Loads the encryption key.
* Reads the encrypted keystrokes from encrypted\_keystrokes.txt.
* Decrypts each line and writes the decrypted keystrokes to decrypted\_keystrokes.txt.

1. **Workflow:**
   1. Run the code, with the below code in the main.



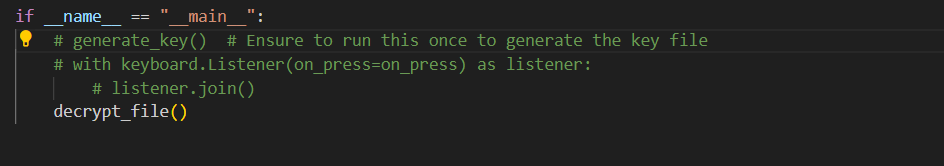
* 1. A file **key**.**key** will be generated, containing the encryption key.



* 1. When you start pressing the keys, it will be recorded in the ‘**encrypted\_keystrokes’** file, in the encrypted form.



* 1. Now, stop the keylogger and change the main to run the decrypt function.



* 1. There will be a new file created, named ‘**decrypted\_keystrokes’**, containing the decrypted text of the encrypted file.

