📝 Python Encrypted Keylogger — Setup & Execution Report

🗒️ Introduction

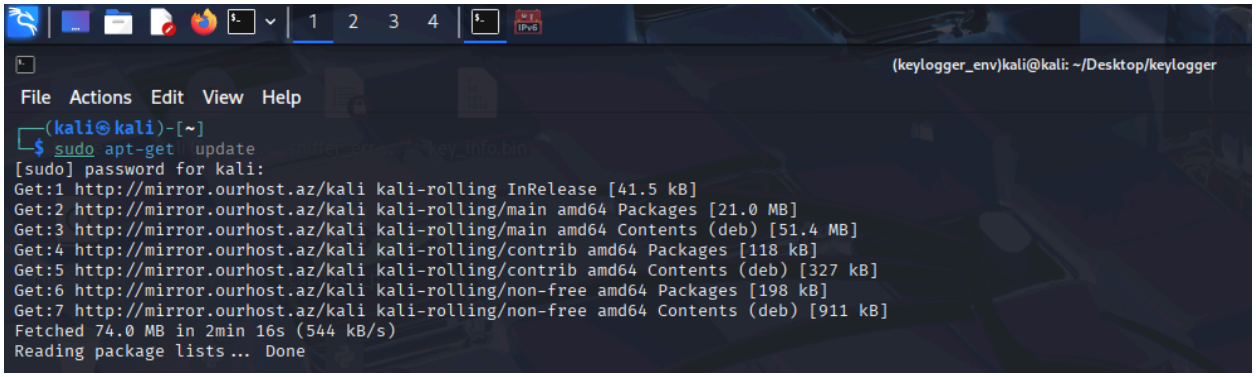
This document outlines the setup, execution, and testing of an advanced encrypted Python keylogger with GUI support. All steps were performed in a **Kali Linux environment** using a virtual Python environment (`venv`).

🔧 Environment Setup

1. **System Update**

bash

```
sudo apt-get update
```



2. **Navigate to Desktop & Create Project Folder**

bash

```
cd ~/Desktop
mkdir keylogger
cd keylogger
```

```
┌──(kali㉿kali)-[~]
└─$ cd Desktop

┌──(kali㉿kali)-[~/Desktop]
└─$ mkdir keylogger

┌──(kali㉿kali)-[~/Desktop]
└─$ cd keylogger
```

3. **Create Files**

bash

```
nano encrypted_keylogger.py
nano decrypt_logs.py
nano keylog_viewer_gui.py
```

```
┌──(kali㉿kali)-[~/Desktop/keylogger]
└─$ nano encrypted_keylogger.py

┌──(kali㉿kali)-[~/Desktop/keylogger]
└─$ nano decrypt_logs.py

┌──(kali㉿kali)-[~/Desktop/keylogger]
└─$ ls
decrypt_logs.py  encrypted_keylogger.py

┌──(kali㉿kali)-[~/Desktop/keylogger]
└─$ nano keylog_viewer_gui.py
```

4. **Create Virtual Environment**

bash

```
python3 -m venv keylogger_env
source keylogger_env/bin/activate
```

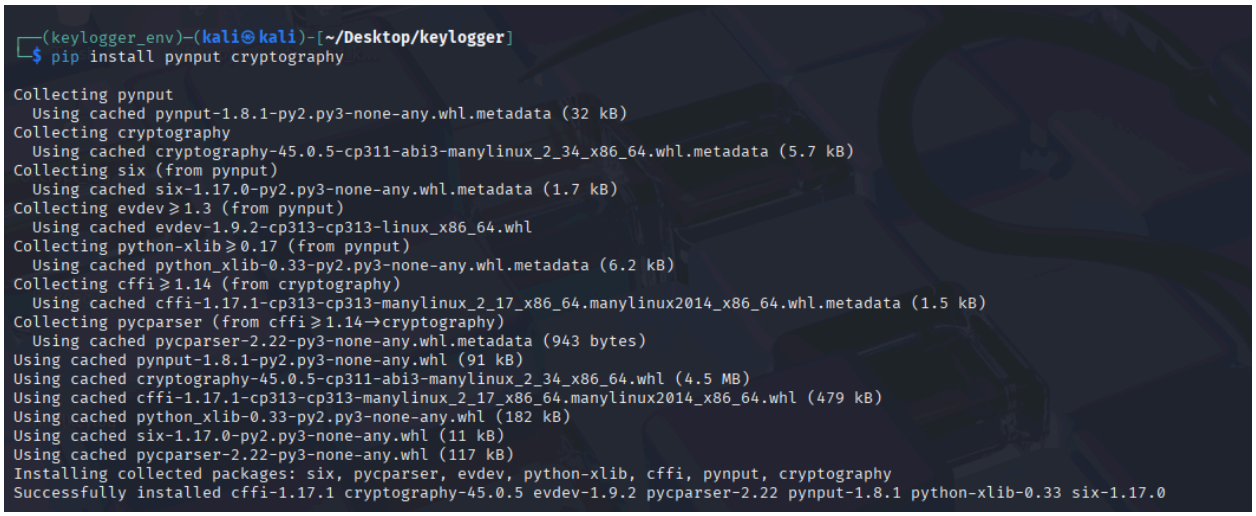
```
┌──(kali㉿kali)-[~/Desktop/keylogger]
└─$ python3 -m venv keylogger_env
source keylogger_env/bin/activate
```

📦 Dependency Installation

Inside the virtual environment:

bash

```
pip install pynput cryptography
```



Modules Installed:

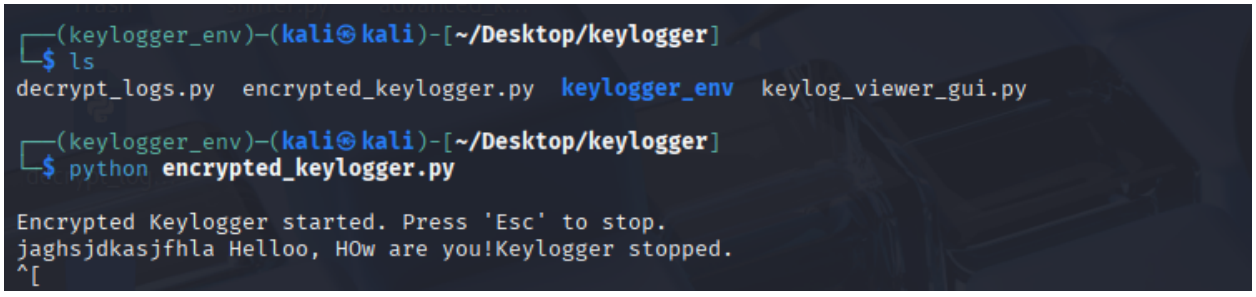
- `pynput` – for listening to keyboard inputs.
- `cryptography` – for AES encryption.
- `python-xlib`, `evdev`, `six`, `cffi`, `pycparser` (auto-handled dependencies)

🧪 Script Execution & Testing

✅ Run Encrypted Keylogger

bash

```
python encrypted_keylogger.py
```

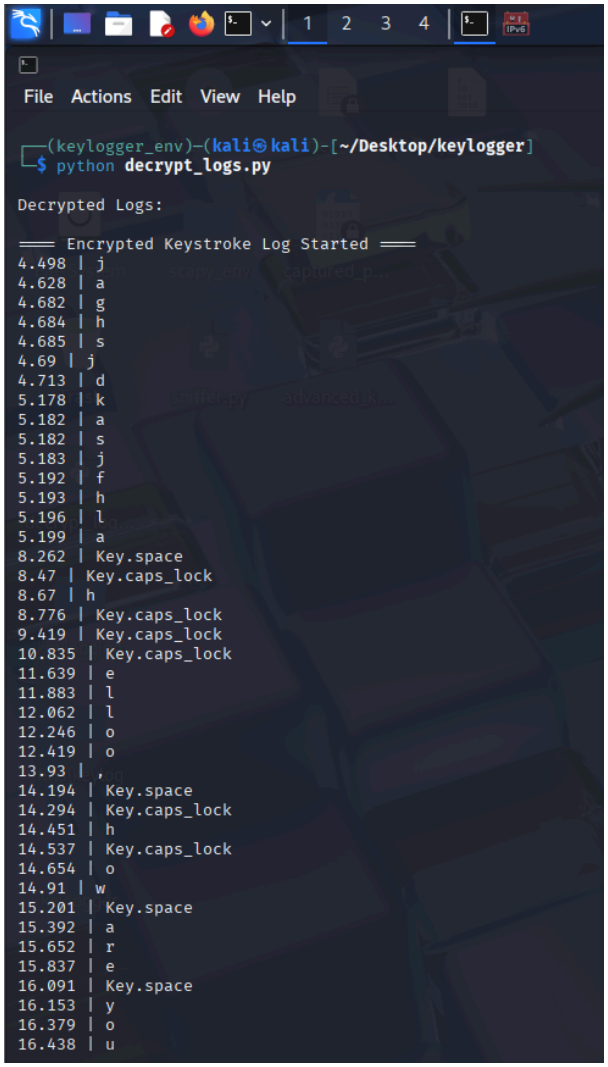


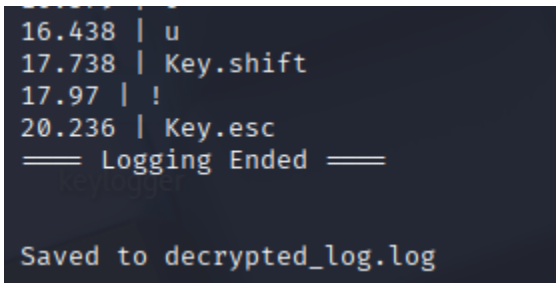
📌 *Keystrokes captured in encrypted format and saved as* `keystrokes.enc` *& key metadata in* `key_info.bin`*.*

✅ Run Decryptor

bash

```
python decrypt_logs.py
```

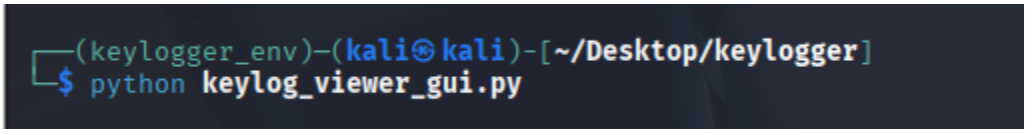




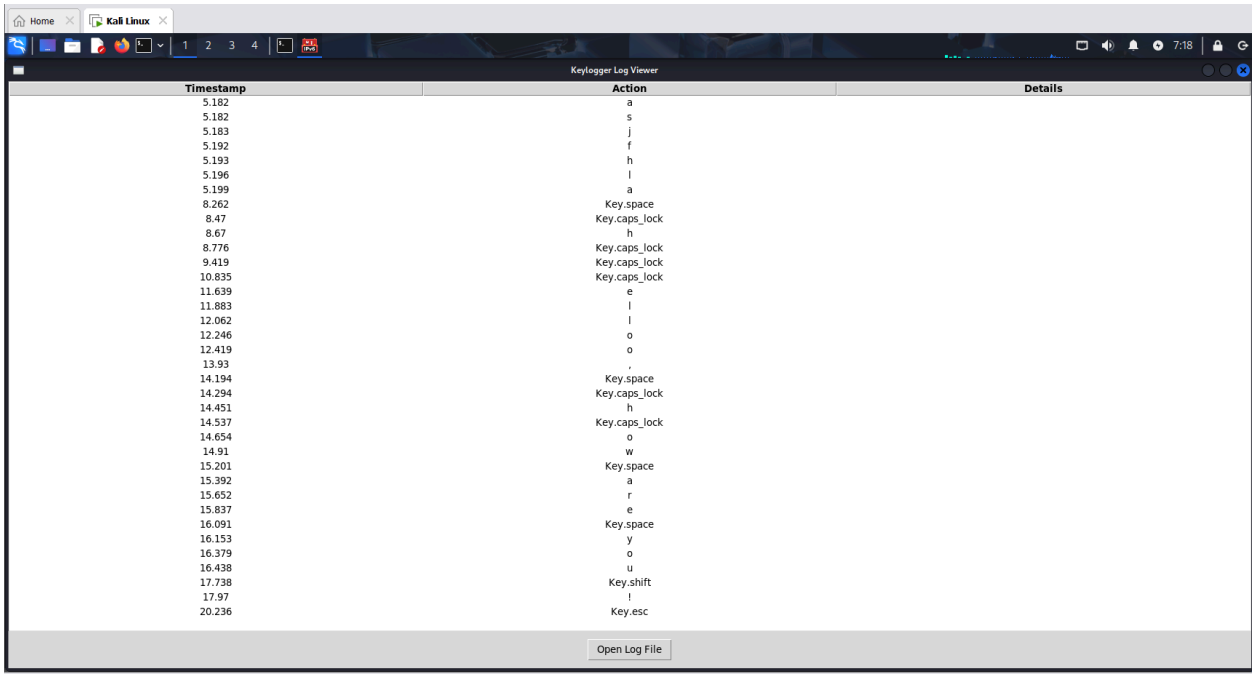
✅ Run GUI Viewer

bash

```
python keylog_viewer_gui.py
```



- Opens a GUI window.
- Click on **“Open Log File”** → Select `decrypted_log.log` → View output in clean GUI format.



📁 File Structure

```
keylogger/
├── encrypted_keylogger.py
├── decrypt_logs.py
├── keylog_viewer_gui.py
├── keylogger_env/              # Virtual environment folder
├── keystrokes.enc              # Encrypted log (auto-created)
├── key_info.bin                # Encrypted AES key (auto-created)
├── decrypted_log.log           # Output of decryption
```

🧩 Observations

- All scripts work seamlessly in **Kali Linux on X11**.
- Encrypted logs are secure and accurate.
- GUI viewer simplifies post-analysis of keystrokes.
- Virtual environment isolates dependencies and keeps the host clean.

✅ Conclusion

The keylogger project successfully demonstrates:

- Secure keystroke logging using AES encryption.
- Modular script separation for better maintainability.
- GUI-based visualization for improved usability.
- Compatibility with Linux and portable to Windows/macOS with minor changes.