University Of Missouri – Columbia

**Creating a Keylogger using Python**

IT3910 Advanced Cyber Security

Final Project

By: Alex Bohrer, Arbzdv

1. **Introduction**

For this project I decided to use python to create a keylogging program. The purpose of this is to capture all keystrokes inputted into the target machine that has my keylogging program on it and to save them to a .txt file. I wanted to put some of my coding skills to action by using the powerful language Python to create the keylogger. Because I have a good background with this language and the extensive documentation alongside the various libraries Python has, It surely was the right language for this project. Python’s easy to read syntax and structure will make it easy for others who are not as experienced in coding to follow along. I wanted to create this type of software to see how a keylogger works, and because I know that many companies use this type of tracking on their employees work computers to track what they do at work. I also want to learn the ways to detect keylogging and the ways to make it harder to detect so I can protect my machines from this type of spyware.

1. **Research**

Keylogging is a very real threat to security and personal privacy; it is important to recognize how this type of software works to better prepare users if they ever are targeted. Keylogging programs run in the background on your machine, they store every single key stroke, often into .txt files so they can be accessed for later use. If a keylogging program is on your machine an attacker could gain access to anything that you type in. This means any time you order something online and enter your card information, it is stolen. There are different types of keyloggers, some for mobile devices, some for desktop. They can access a wide array of data, whether that be only what is typed or as far as even stealing users’ clipboards when they copy and paste. Keyloggers also can be used by several groups, it could be a malicious attacker, the government, or it could be the IT department at your workplace. There is a use for keyloggers for good, like in the work setting where it is important that people do not search the wrong things on the internet. These types of attacks are quite dangerous because the longer they persist, the more information the attacker gains to use against the target. With enough time a keylogging program can gain an attacker access to virtually all passwords you use, and a flurry of other personal information that can be sold or used to steal your identity.

Keyloggers can store or send the data they receive in a few separate ways; It can upload the data to a database, website, server, they can store the data in hidden directories on the host’s machine that can then be accessed later through remote desktop, and some even email the data to a certain email periodically. In Metasploit’s Meterperter there also Is keylogging functionality that can be targeted to a log keystrokes from a specific program. Keylogging is done through software, but also can be in the physical state. There are wireless keyboard sniffers, that intercept the Bluetooth connection between a computer and the keyboard, that can intercept the messages that are meant to be sent to the machine. There are keyboard overlays which are a physical device that a thief plants on public machines like ATMs to steal user pin numbers. You also can load your keylogging program onto a flash drive and plant that on the back of a computer to collect data. Another common place where keylogging software is used is on public computers. On these public machines, users with malicious intent can plant the program on the machine or the easier solution is a hardware keylogger. It is best to avoid using public computers to log in to any important website or account, because of the risk of these keyloggers. There are several ways to implement this type of surveillance on machines, next we will continue to how to protect your machine from this type of attack, and how to detect and stop this on your own machine.

To discover this technology on your machine there are several approaches, and they all depend on the type of attack that is being done. To mitigate software that is capturing keyboard input, one could use an anti-keylogger like SpyShelter. These programs identify or stop the process that is behind the keylogging on your machine. There are two types of anti-keylogging programs, one is called signature based and the other is called heuristic analysis. Signature based is specific information that helps to uniquely identify a keylogger, these have a constantly updated list of popular keyloggers signatures, that is referenced to search machines for the malicious software. One drawback to the signature-based method, is the fact that hackers are always innovating, and sometimes new keylogging software signatures will not be on the updated list allowing them to run undetected. The other option, heuristic based, uses a checklist of disclosed features, attributes, and methods that modern keyloggers use and then blocks the use of those methods, features, or attributes to disable the keylogging. Heuristic based are more effective in detecting new keyloggers unlike the signature-based method, but also have a high chance of sending false alerts, breaking other applications that are legitimate.

1. **Applications**
2. Install Pynput via PIP
   1. Graphical user interface, text

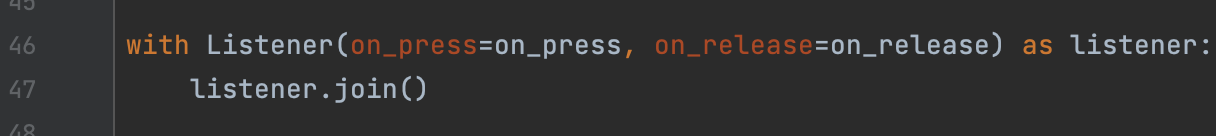
      Description automatically generated
   2. This is for downloading the library “pynput” which contains most of the processes needed for making a keylogging program in python.
3. In pycharm or any other text editor
   1. Create a new project
      1. Name the project and append the .py extension to make it a python program.
   2. Import pynput, and import “Key” and “Listener” from “pynput.keyboard”
      1. Graphical user interface, text

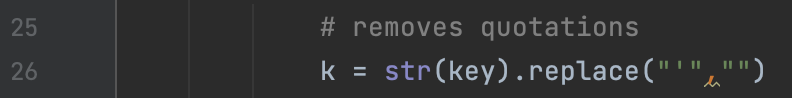
         Description automatically generated
      2. The “Listener” is the part that listens for key events, so it is important to remember to bring this into the project.
      3. Importing pynput is the module that allows the tracking of keystrokes and the mouse.
   3. Coding
      1. Start by making a function that is called every time a key is pressed.
         1. Graphical user interface, application

            Description automatically generated
            1. This function is called on keypress and prints out the key entered by the user inside the curly brackets, replacing the zero.
            2. The code beneath the formatting code line, is a loop that writes the keyboard input to the .txt file when it reaches the specified length.

This number should be can be changed based on how long you want it to collect input.

* + 1. Next make a function to run on key release,
       1. Text

          Description automatically generated with medium confidence
          1. Lines 16 and 17 allow the user to break out of the loop with the use of the escape key. Using the escape key will write the content to the file if it has reached the word count set previously.
    2. Finally use the listener to create a infinite loop that uses the functions with the listener.
       1. 
    3. Make function to create a file to save the keystrokes too.
       1. Text

          Description automatically generated
          1. Create a file called gotU.txt for the program to append to.
          2. This function opens a file named gotU.txt in append mode, meaning it will add the keystrokes to the file created.
    4. In the for key in keys loop
       1. The first line removes the quotations on the printed text to the file.
          1. 
       2. The first if statement adds a new line every time a user hits the space bar, to separate out the users input into a much more readable form
          1. A screenshot of a video game

             Description automatically generated with low confidence
       3. The first elif checks if there are multiple user inputs of space, and if there are it deletes the line, to keep the program from storing several spaces.
          1. Graphical user interface, text, application, chat or text message

             Description automatically generated
       4. Finally, the second elif, removes the printing of user input on the space bar, shift key, and all other keys that are not numbers or letters.
          1. Graphical user interface, application

             Description automatically generated
  1. Example Output
     1. Reading individual key press In IDE terminal.
        1. Text

           Description automatically generated
     2. Output to gotU.txt
        1. Graphical user interface, text, application

           Description automatically generated
  2. Final code screenshot
     1. Text

        Description automatically generated

1. **Sources**

How to write to a file in Python <https://www.w3schools.com/python/python_file_write.asp>

Pynput documentation <https://pynput.readthedocs.io/en/latest/index.html>

Keylogging: https://www.veracode.com/security/keylogger

What is a keylogger: <https://www.csoonline.com/article/3326304/what-is-a-keylogger-how-attackers-can-monitor-everything-you-type.html>

Python keylogger example code: <https://medium.com/@harsathAI/python-keylogger-advanced-with-autorun-usb-for-windows-a-modern-approach-with-clear-code-573e7b828cf2>

General keylogging info and how to mitigate <https://en.wikipedia.org/wiki/Keystroke_logging#Countermeasures>

Anti-Keyloggers: https://en.wikipedia.org/wiki/Anti-keylogger