**CAPSTONE PROJECT**

**KEYLOGGER DETECTION**

** KEY-GUARD**

**Report Paper Submitted**

**By**

**Team**

**Nathanael A Thomas**

**E23MCAG0068**

**Austin Devodass**

**E23MCAG0059**

**Shobit Jain**

**E23MCAG0089**



**Department of computer Science Engineering, Bennett University**

**Abstract:**

The main objective of the keylogger detection project is to create efficient techniques to identify and counteract the risk posed by keyloggers these harmful programs are specifically design to find the keystroke made on a user computer.

We plan to do a few things with this project. Studying the common characterises of keyloggers first. With that works to create algorithms that quickly recognize these harmful programs.

The main discovery shows that detection works well against known keyloggers but detection methods are better at spotting new threats. Its also important to take proactive steps like update software regularly educating users to lower the chances of keylogger infections.

Our keylogger detection system project has proven to be highly successful as shown by the result of our experiments. We designed a strong and efficient tool that can detect it is also be a perfect success rate.

**Keyword:**

User activity monitoring, access control system, user behaviour, security.

**Table of Contents:**

* Introduction
* Background and Context of the Project
* Importance and Scope of the Project
* Keylogger
* Keylogger Detection**:**
* Methodology
* System Design
* Implementation
* Results
* Conclusion
* References

**1) Introduction:**

**Background content:**

Keylogger a form of malicious software is designed to find every keystroke made on a computer. This program makes a significant risk to computer security as they will easily find the most sensitive information etc login details, credit card numbers, and private messages. To protect our system, it is important to employ effective tools that can detect.

**Important of the project:**

The main thing of this project is to create a software application called Key-Giard it is designed for identify keylogger on a system. Our software will continuously monitor network traffic to capture the unusual behaviour that will shows the presence of a keylogger in system.

This project is mainly for the Windows system. The main objective of the project is to find keyloggers that communicate through TCP port 587,465 and 2525 which is mostly used for email transmission.

**Keylogger:**

A keylogger is a malicious software that secretly records every keystroke you press on your keyboard it works in the background it will take the all-secret information you type in your keyboard include password, credit card number, emails, texts, and other private info also.

**Keylogger Detection:**

Keylogger detection is simply trying to identify if a software is capturing your key movement.

**Methodology:**

Key-Guard uses network traffic analyses to identify keyloggers. Through analyse the network traffic the software can detect any suspicious activity it will inform to the user the keylogger is activated in the system. If the key-guard is finding any suspicious activity it will give flag.

**Tools and Technologies**

1. Language: Python

2. Library: psutil

3. Platform: Windows

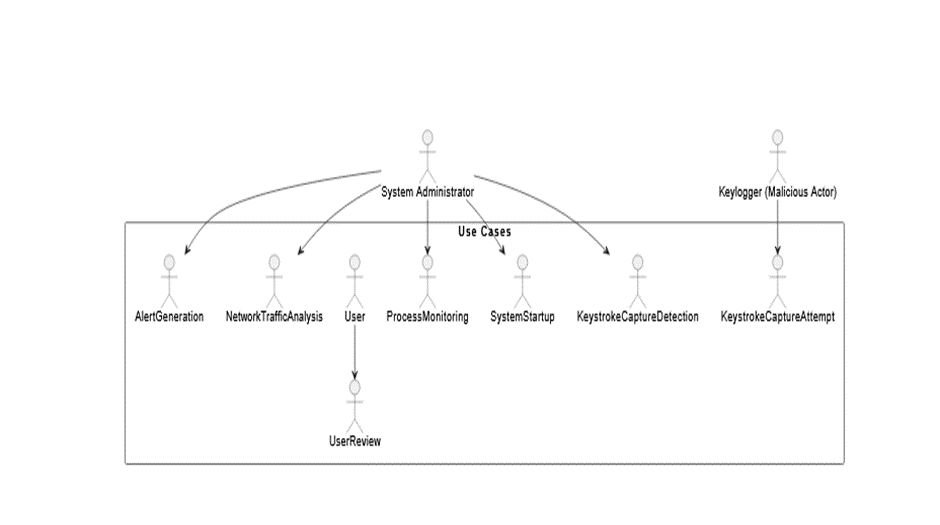
4. Tool: auto\_py\_to\_exe

Psutil:

psutil is a very powerful library that activate key-guard can easily track the network traffic generated by various programs. We make a module for data analysis and keylogger detection we monitoring the network traffic data using differ method. We also make the module for actions upon detection which determine what actions to take if an unusual keylogger if identified.

**Design and Implementation Process:**

We will code the main functions of key-guard using python and the chosen libraries is psutil and potentially scapy. If we want more advance network traffic analysis, we can consider using scapy This library use to capture packets provide well insights into patterns. This system is specially for windows the environment of the project is suitable for the windows system.

**Use case diagram:**

**System design:**

The GUI means the user which is used to interact with the system. And there are some buttons each one having their own functions. Some are here to check information, show output, manage blacklist and whitelist, start and stop and other tasks.

Network monitoring function it will operating in the background all time it monitoring the network traffic by program running on the system. It using the libraries such as psutil to monitor network data.

The major duty of the data analysis and the detection system is to examine the gathered network traffic data. It is using the technique such as algorithm to find any pattern that could similar to the existed keylogger.

Action chapter are working like the software key-guard is detect any keylogger in the system this action function will respond based on the algo and defined rules we applied in the software it will only act like that.

**Possible action:**

Showing a alert message on the interface to alert the user to aware the keylogger in detect in the system.

Halting if the software finds any keylogger or suspicious activity in the system it will halt the all process

Ending the program suspected of being a keylogger

Record is the harmful process or a keylogger is identified in this software it will record for future analysis.

The software we build it also have one more feature called blacklist and whitelist management. It will make a two separate lists. One for the blacklist and another for the whitelist process if any thing find as an error the system will ask it wat to add In blacklist or whitelist. If it will add on the blacklist, it will permanently block but at the same time it will add on the whitelist it will never block from the system. The user has the full control to manage the lists if the user wants to add it in blacklist or whitelist it is possible with the GUI.

**Flowchart:**

Start: This is the beginning of the project and it is the starting point of the program.

Add/Remove: we can add or remove the list of programs

Blacklist/Whitelist: The list consists of programs that the scanning function will take permission.

The whitelist consists program will consider as a safe one, and the blacklist consist program will consider as an unsafe one.

Scan: This function scans the all system for finding the keyloggers are in the system or not.

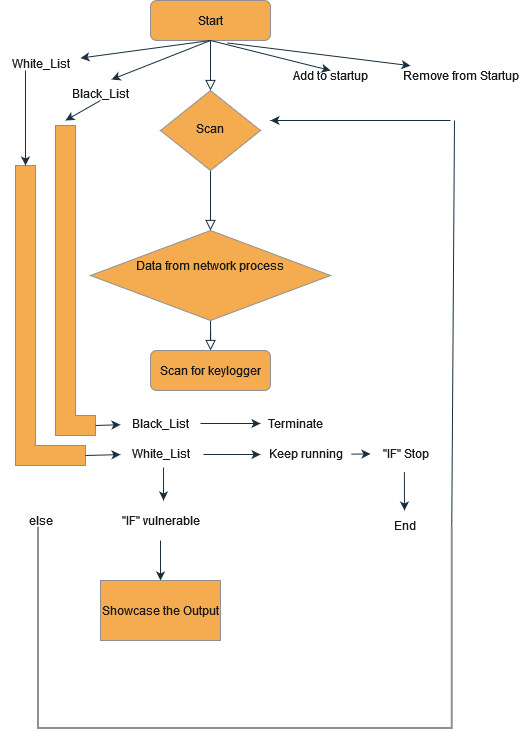
Blacklist: if the program if find something wrong during the scan it will check the blacklist to see if the error is matched with the list or not.

Whitelist: if the program if find something wrong during the scan it will check the blacklist to see if the error is matched with the list or not.

Terminate: if the program will find the error during the scan and it will check the white list also in the same time it is not in the whitelist it will terminate.

Keep running: if the program finds any error but it is in the whitelist or nothing error it will keep run.

Showcase the output: This function displays the output of the scan.

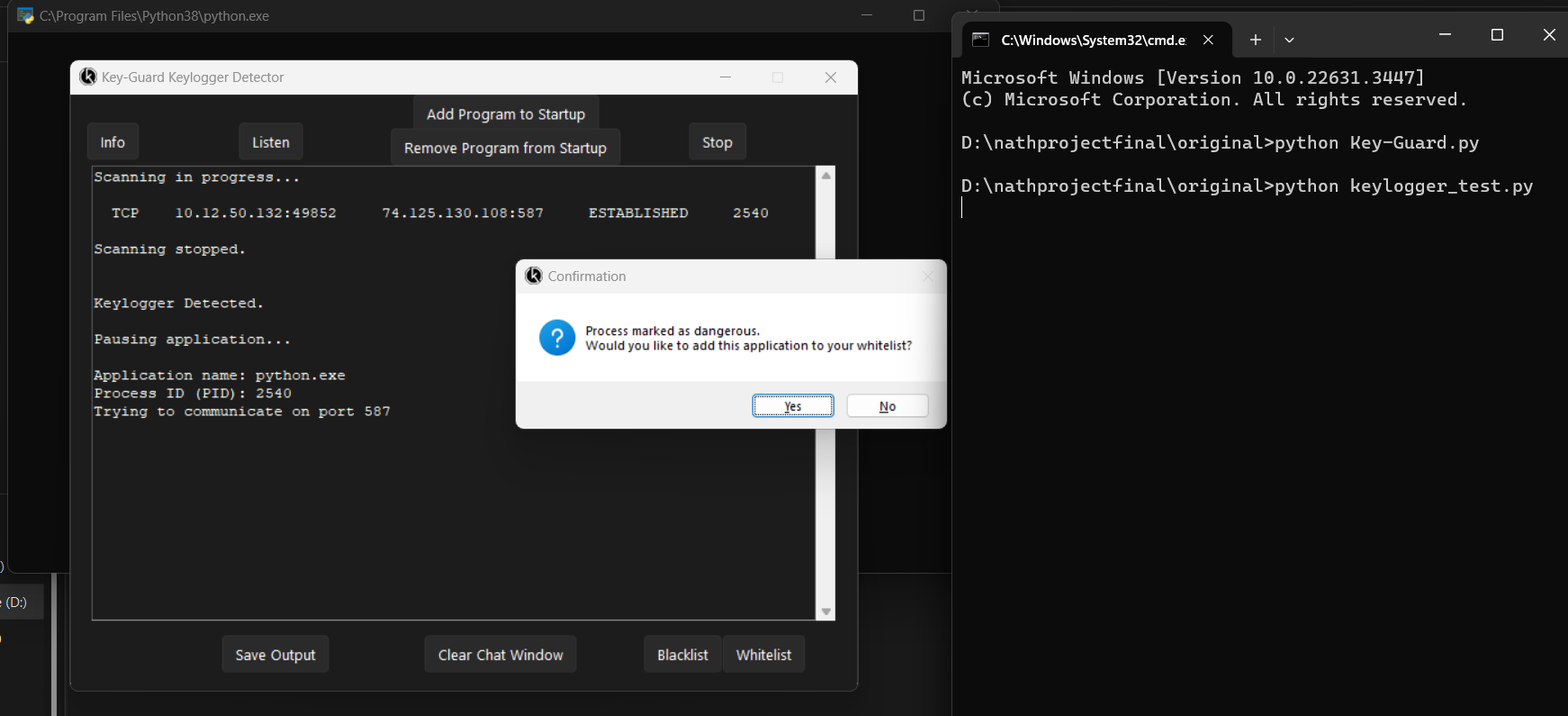


**Implementation:**

This document presents a system designed to detect keyloggers with monitoring the network traffic in a device. The system focuses on application that is for SMPT ports. It is compatible to both windows and Linux. When an application tries to access a monitored port, the system pauses the application and asks the user for input. Users can choose to approve the application and asks the user for input. Users can choose to approve the application by adding it to a whitelist or terminate it adding it to the blacklist. If we put an application in to a blacklist it will automatically terminate by the software. The system faces some challenges such as limited detection capability on specific port and reliance on user decisions for whitelisting or blacklisting applications which is mostly introduce human error and reduce reliability.

**Result:**

We test an experimental valuations of our project keylogger detection system using a actual test keylogger. We run the actual keylogger in the system first and turn on the key-guard it will successfully detect the keylogger in the background we repeat this same test differ conditions in all time it shows 100% accurate value results and it working very finely and its didn’t shows any errors and we test it with out internet connection and at that time it didn’t work because if the internet is not connected the data transfer didn’t applicable. Otherwise, it will perform well.



**Conclusion:**

To summarise the project keylogger detection system has been a high successful project. The keylogger detection is working properly and detect all the errors and detect the keyloggers. This result demonstrates how well our solution protect against unauthorized keyboard logging activity and its improving cybersecurity for every individual’s system.

Key-Guard has the potential to become a powerful tool in the fight against keyloggers by overcoming challenges and incorporating user friendly feature for selecting action.

**References:**

<https://www.researchgate.net/profile/Ana-Maria-Ciobotaru/publication/266743342_An_Introduction_to_Undetectable_Keyloggers_with_Experimental_Testing/links/57bc297008ae51eef1f61559/An-Introduction-to-Undetectable-Keyloggers-with-Experimental-Testing.pdf>

<https://ieeexplore.ieee.org/abstract/document/7726880>

<https://www.researchgate.net/profile/Yahye-Abukar/publication/309230926_Survey_of_Keylogger_Technologies/links/59a00619aca27237edba3c12/Survey-of-Keylogger-Technologies.pdf>

<https://link.springer.com/chapter/10.1007/978-3-642-40576-1_26>