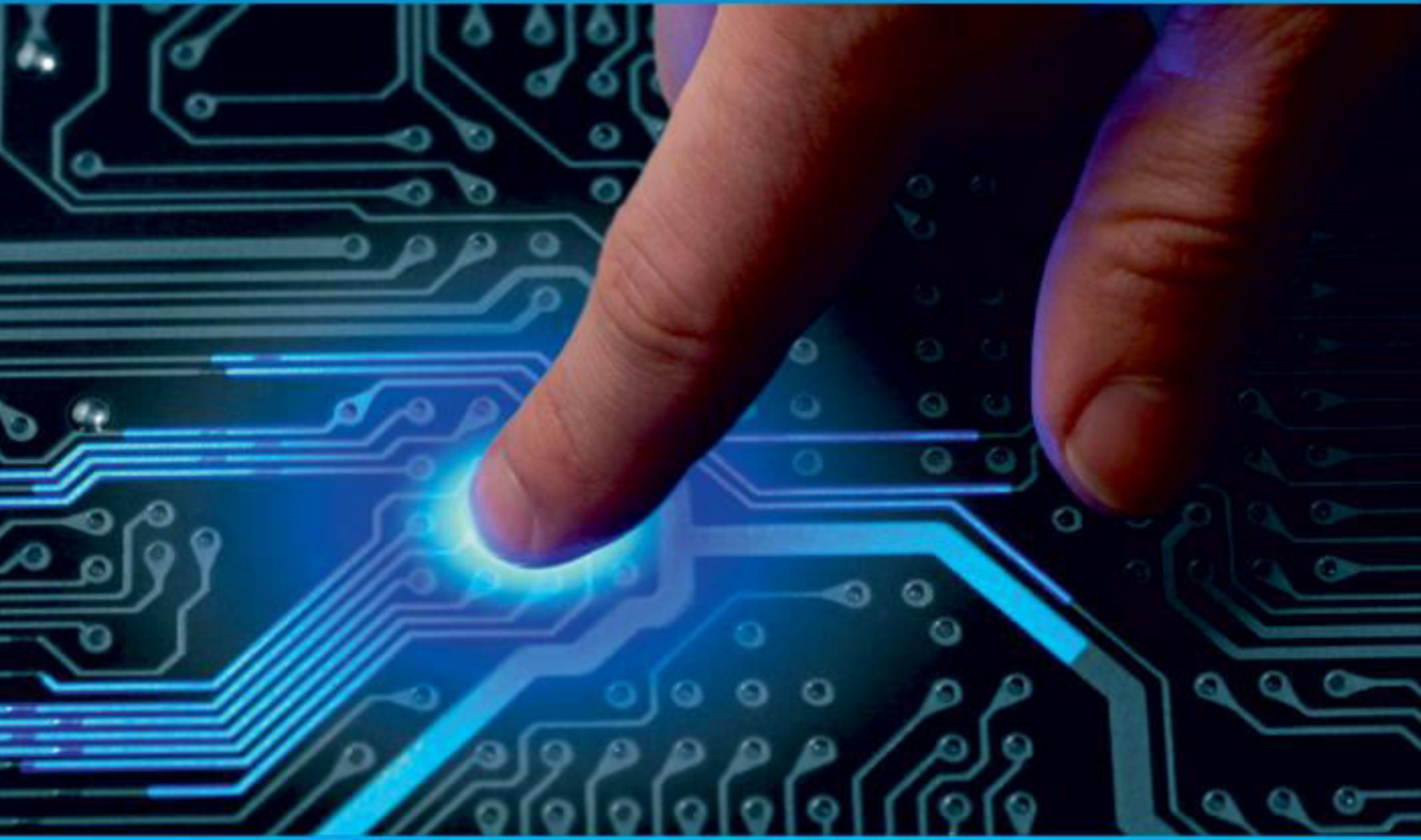
IJIRCCE

e-ISSN: 2320-9801 | p-ISSN: 2320-9798



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

Volume 10, Issue 4, April 2022

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 8.165

International Journal of Innovative Research in Computer and Communication Engineering

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.165 |

|| Volume 10, Issue 4, April 2022 ||

| DOI: 10.15680/IJIRCCE.2022.1004128|

Employee Monitoring System using Advanced Key-logger

Prof. Dr.Sanjeev Uppal[1], Sanchay Harjai[2], Chandan Sahu[3]

Professor, Department of Electronics & Communication, HMR Institute of Technology & Management, Delhi, India[1]

U.G. Student, Department of Electronics & Communication, HMR Institute of Technology & Management, Delhi, India[2]

U.G Student, Department of Electronics & Communication, HMR Institute of Technology & Management, Delhi, India[3]

ABSTRACT: Employee Monitoring System is an attempt to retrieve system information by capturing user input via keystroke monitoring, screenshot capturing, microphone recording, system information and then relaying this information with the help of a Log file attached via email. This paper presents a project on keylogger which is an Employee monitoring system using a keylogger. First, the paper provides an overview of keylogger programs, discusses keylogger design, implementation, and usage.

KEYWORDS: keylogging, rootkits, secure coding, cybersecurity education.

I. INTRODUCTION

Keylogging programs, commonly known as keyloggers, can be used as malware or can be used for monitoring purposes also, it is a software that maliciously or purposely put on a device to track user input from the keyboard to retrieve personal and private information. Increasing computer use for common business and personal activities using the Internet has made effective handling of keylogging urgent. Additionally, the Internet has not only become a major distraction for employees for placing and distributing different programs, so to overcome this challenge we will study about keylogger and its techniques

For a variety of reasons, keylogging applications and anti-keylogging strategies should be studied as part of cybersecurity education. First, keyloggers include a wide range of cybersecurity themes, including attacker intentions, malware types and implementation, malware's function in infecting and managing a system, and how compromised systems attain stealth. Second, students will learn about keylogger detection and prevention methods and procedures. Whether the detection is via active system monitoring for malware memory footprints or for keylogger-like behavior, a more dynamic approach to detecting keyloggers is needed. In fact, the degree of dynamism separates mediocre anti-malware programs from effective ones. Ensuring that a security practitioner learns about handling keylogging malware is thus important in cybersecurity education.

II. LITERATURE REVIEW

To recognize keyloggers all the more conceivably, it is significant for an individual to get a handle on top to bottom information about what keyloggers really are, how they are implemented, and understand different approaches to it. To response this kind of queries we will discuss about different kind of algorithm proposed so far to overcome the problem and also the drawbacks of those proposed systems. Key logging is a security trading off procedure which should be possible from multiple points of view. When an attackergainsphysical access to your computer devices they can wiretap the physical hardware like a keyboard to collect the valuable data of the user. This strategy is totally reliant on some actual properties, either the sound transmission created when a client is composing or the electromagnetic spread of a remote console (Martin Vuagnoux, 2009). External keyloggers or hardware keyloggers are small electronic devices which are placed in between keyboard and motherboard, this procedure requires the attackers to have physical access to the system which they are intended to compromise. Keyloggers are executed on the focused machine to record client's keystrokes logging movement, lastly giving over that private information to an outsider(Thorsten Holz, 2009). Keyloggers are utilized for both lawful and illicit purposes. Keyloggers are generally utilized by assailants to take private information of an individual or an association. In the past many credit card details have been compromised by attackers with the help of keyloggers. Henceforth, keyloggers are one of the most hazardous sorts of spyware till date, (Strahija, 2003).

Figure1.Basic components of Employee Monitoring System

III. ANALYZING KEYLOGGING TECHNIQUES

Previously existing keylogger monitoring systems have some drawbacks like it can only record keystrokes and create its log file and transfer that file on email or any other wireless medium. There are also some Keyloggers that come with some advanced features like taking screenshots but there is no complete solution for this problem. A keylogger is a program that record all key stick entered on the keyboard, in another word keyloggers are the sort of the spywares that take the information of the clients by following their keyboards. Detecting the key loggers is troublesome undertaking to perform because generally they hide their presence using technology like root-kit so they don’t get detected from antivirus and other system protections. The primary work of this program is that they will catch the keystrokes squeezed by the client and store them in a log file. Either this log file can bestored on the same system or sent to another system using the internet or other communication method. We all know how important it is to protect our password and other important data. Thiskeylogger had a difficult protection task so in this paper we are going to talk about various types of keylogger and their prevention methods.

Problem Statement

In many IT infrastructure organizations now-a-days, data security and data recovery are the most important factors which are basically deployed in Computer Forensics. Computer forensics consists of the art of examining digital media to preserve, recover and analyze the data in an effective manner. There are many cases where data recovery is required. So by using keylogger applicationsusers can retrieve data in the time of disaster and damaging of working files due to loss of power etc. Keyloggers are especially effective in monitoring ongoing crimes. This is a surveillance application used to track the users which keystrokes, uses log files to retrieve information, and captures a record of all typed keys. The collected information is saved on the system as a hidden file or emailed to the admin or the forensic analyst.

IV. DESIGN AND IMPLEMENTATION

1. Key logger design and implementation strategies are based upon several factors: the infecting medium, the type of target machine, the lifetime of the key logger, and the level of stealth and footprint left on the machine while active. Infection mechanisms depend on the form of the key logger. A software keylogger targets the user-mode of an operatingsystem and is injected remotely and a hardware keylogger via physical device placement. Software keyloggers require a well-crafted infection mechanism to ensure proper installation, for example, a web browser exploit. Most keyloggers share a common execution technique known as hooking, though each keylogger will implement it in a different way depending on the context for which the keylogger is needed The basic goal of hooking is to intercept the normal control flow and alter information returned by a target system routine. Hooks can be implemented in any level of the operating system for most functions. High-level key loggers executing in the user mode of an operating system are implemented using a variation of user mode hooks. Low-level kernel-mode key loggers are typically implemented as root ware, a combination of both rootkits and spyware that employ another variation of hooking.
2. All available user-monitoring products are essentially programs that report on (and in some cases constrain) how you use other programs. Having installed an user monitoring program, an organization can -- depending on the type of program -- see how much time users (individually and/or in aggregate) spend playing Solitaire, or what web sites they visit, or even read email messages thatthey typed but then deleted and didn't send. The organization may also be able to prevent users from visiting certain websites, or from sending or receiving certain emails One way to understand

these products is to consider where they are installed. There are basically two types: server-based monitors, designed to be installed on the organization's network; and client based monitors, designed to be installed right on the personal computer (PC) used by the user. First, we'll look at the network (server), then at the PC (client). To see the difference, let's imagine a typical user, whiling away the time playing Solitaire. Wes Cherry, the Microsoft programmer who wrote the Solitaire game included with Windows, has noted that he has single handedly "wasted more corporate time than any other developer" (though organizations might recall that many users first learned to use a mouse by playing Solitaire). The question is, Can the corporation tell (short of looking over his or her shoulder) whether an user is playing Solitaire? The answer is yes they can see everything.

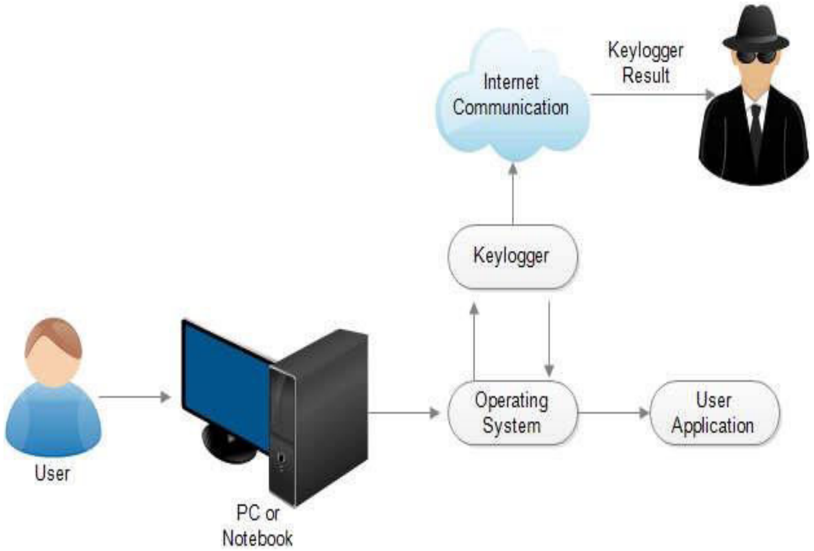
A list of Accountability Features

- Keystrokes Typed at any Place
- Programs Opened
- Title of Documents, videos, music, etc. left opened.
- Websites visited
- PC-wise and user analysis
- Record Audio
- Frequent Screenshot Capture
- Email Transfer of Log Files

Who May Need This?

- Hospitals
- Banks
- IT organizations
- Institutions & Universities
- Call Centers
- Government Regulatory
- Internet Business Organizations

System Flow Diagram



| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.165 |

|| Volume 10, Issue 4, April 2022 ||

| DOI: 10.15680/IJIRCCE.2022.1004128|

Flow Chart

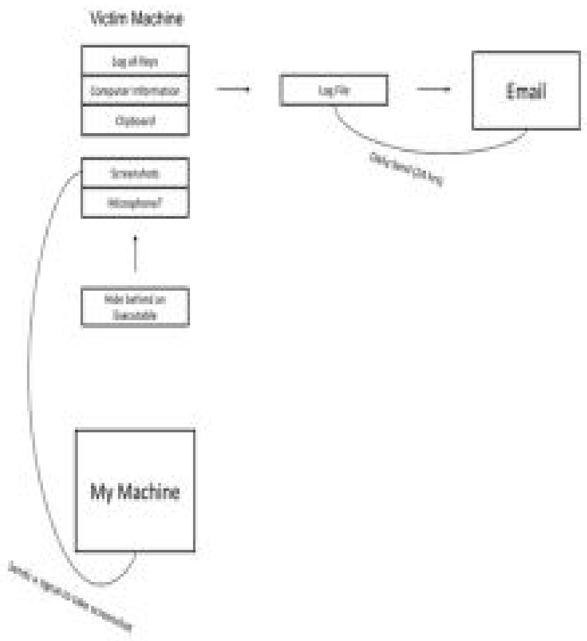


Figure2. Flow chart.

An Overview

The keyboard is the primary purpose from which keystroke loggers can obtain user input as it is the most commonly used user interface on a computer. However, hardware keyloggers are mentioned in this paragraph because they pose a significant security threat. A common example of a hardware keylogger is a "ghost" device that can be physically connected to a target computer to extract keystrokes and store them in permanent storage within the target computer. For example, inexpensive hardware keylogging devices such as the spy keylogger act as the medium between the USB adapter on the physical keyboard and the USB port on the computer; Like the "man in the middle", the device secretly records and stores All keystrokes that the user made in their memory. This document focuses on software keyloggers as they are the predominant form of keylogging. Both consumer-oriented inexpensive keylogger programs and keylogger programs. Properties Existing system Proposed System Features 1-Keystrokes capturing 2-Receiving Log file of email 1-Keystrokes capturing 2-Record audio 3- Clipboard & System Info 4-Capturing frequent screenshots 5-Removing duplicate screenshot images 2-Receiving attachments on email Challenges 1-Many different software are available in the market with different features 1-We tried to combine all the different features of available keyloggers which make it hard to implement Advantages 1-Easy to implement 2- Consumes less time to implement 1-Multiple features support Disadvantages 1-Limited amount of features 1-Very difficult to implement 2-Consumes more time to implement Specially developed keyloggers are available on the Internet. These keyloggers need to be tailored to each target operating system to ensure that the I / O is handled properly. Differences in the system inevitably lead to operating system-specific mechanisms implemented in software keyloggers: use of the keyboard status table, routine system bindings, and kernel-mode layer drivers. Further details on the techniques used in the development, distribution, execution, and detection of kernel and user-mode keyloggers are presented below, particularly in Microsoft Windows operating systems; Note throughout this article that a reference to Windows means a variant of Windows NT. A fundamental concept behind keyloggers and similar malware is their attack pattern. Most malware infections follow a fairly standard attack pattern that includes the sequential order of development, distribution, and infection. and implementation phases. The initial phase is critical to the process as undeployed malware cannot be used by an attacker. What is special about the development phase is that it places an emphasis on how the later phases are carried out. Implementation

Keylogger design and implementation strategies are based on several factors: the media infected, the type of target machine, the lifespan of the keylogger, and the level of camouflage and footprint that remains on the device while it is active. The mechanisms of infection depend on the shape of the keylogger. For example, a software keylogger that targets the user mode of an operating system is often injected remotely, and a hardware keylogger is injected via the physical location of the device. Software keyloggers require a sophisticated infection mechanism to ensure proper installation, such as a Web browser exploit. The attacker can identify and exploit existing security gaps in the browser used.A typical browser exploit uses a client-side language such as Javascript to create and execute an attack. These local attacks are generally aimed at creating a buffer overflow in the browser or a related component such as a control redirection plug-in .Target data stream. to enable the execution of malicious code.

Once the infection mechanism is implemented, the keylogger designer focuses on executing it. Most keyloggers use a common execution technique known as hooking, although each keylogger implements it differently depending on the context for which the keylogger is needed. The basic aim of the hook is to intercept the normal flow of control and to change the information returned by a routine on the target system. Hooks can be implemented at any level of the operating system for most functions, making them a common technique that keylogger developers can use.In this document, the term "hook" is used to describe any technique that intercepts data in an existing control flow for malicious purposes.

High-level keyloggers that run in user mode of an operating system are implemented using a variation of user mode hooks. to respond to the keystroke. This messaging mechanism can be wired so that an attacker could gain access to these key press events before they even reach the target application. Depending on the context, the keylogger under development can implement a global or local hook to retrieve key press events. Global hooks monitor system- wide messages, while local hooks monitor the application- specific message. With these hook messages, the attacker could read the messages, typed keystrokes, change the keystrokes, and even interrupt the flow of messages completely.Typically, however, keyloggers implemented in this way only read the keystroke data and forward the message to the next member in the chain. kernel mode low-level keyloggers are typically implemented as rootware, a combination of rootkits and spyware using a different variant of hooking. A rootkit is a small set of programs or tools that run covertly on an infected computer to provide long-term access without detection. to the root of a system for the attacker. Stealth is usually a high priority for a rootkit because it is supposed to be a "permanent" modification of the operating system kernel. Spyware is software that collects user data without the victim's consent. Using these two terms, rootware keyloggers are stealth software that connects to key system routines to collect and broadcast the user's keystrokes without the victim's knowledge or consent. The kernel remains an ideal target for a rootkit to achieve the desired level of camouflage and lifespan.With this approach, rootware developers can put their drivers at the top of the device driver stack and intercept the I / O requests that go between the keyboard device and the kernel to extract data from keystrokes in use.This layered driver approach as implemented on a Windows operating system is depicted in Figure 3.

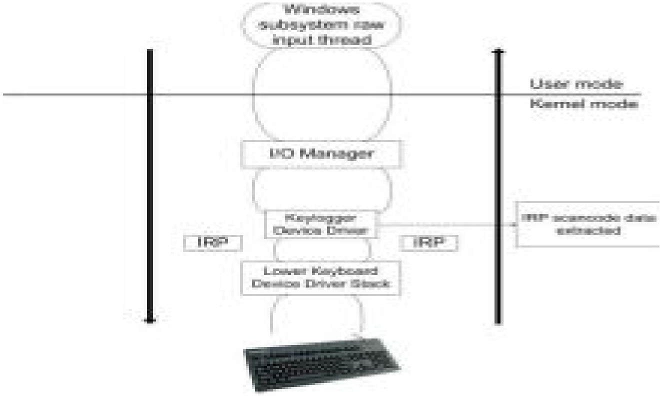

Windows subsystem raw input thread
User mode
Kernel mode
I/O Manager
Keylogger Device Driver
IRP
IRP
IRP scancode data extracted
Lower Keyboard Device Driver Stack

Figure 3. Layered Device Driver Interception of I/O data

| DOI: 10.15680/IJIRCCE.2022.1004128|

V. CONCLUSIONS

Software that can not only monitor every Keystroke and action performed at a Desktop, Laptop or any devices but also be used as legally binding evidence of wrongdoing has been unveiled. Worries about cyber- crime and sabotage have prompted many employers to consider monitoring employees.

They have joined forces to create a system which can monitor computer activity, store it and retrieve disputed files within minutes... "People need to recognize that you are using a PC as a representative of a company and that employers have a legal requirement to store data. Employee monitoring system can record keystrokes, can capture screenshots, recording the audio, fetching system information and great variety of check interval 4 hours to every one minute. Typically, most monitoring services test your server anywhere between once-per hour to once-per minute.

Features:

• Protect intellectual property and business secrets

• Prevent and stop sabotage and data theft

• Prevent Internet/email abuse • Reduce workplace slacker

REFERENCES

1. S. Sagiroglu and G. Canbek, “Keyloggers,” IEEE Technology and Society Magazine, vol. 28, no. 3, pp. 10 –17, fall 2009.
2. A. Emigh, “The crimeware landscape: Malware, phishing, identitytheft and beyond,” A Joint Report of the US Department of HomelandSecurity — SRI International Identity Theft Technology
3. P. Mell, K. Kent, and J. Nusbaum, “Guide to malware incidentprevention and handling,” National Institute of Standards andTechnology, Gaithersburg, MD, Tech. Rep. 800-83, November 2005.
4. T. Olzak, “Keystroke logging (keylogging),” Adventures in Security,April 2008 (accessed May 8, 2010), http://adventuresinsecurity.com/images/Keystroke_Logging.pdf.
5. N. Provos, D. McNamee, P. Mavrommatis, K. Wang, and N.Modadugu, “The ghost in the browser analysis of web-basedmalware,” in HotBots’07: Proceedings of the first conference on FirstWorkshop on Hot Topics in Understanding Botnets. Berkeley, CA,USA: USENIX Association, 2007, pp. 4–4.
6. T. Thornburgh, “Social engineering: the ’dark art’,” in InfoSecCD’04: Proceedings of the 1st annual conference on Information securitycurriculum development. Kennesaw, Georgia: ACM, 2004.
7. S. Shah, “Browser exploits - attacks and defense,” London, 2008(accessed May 8, 2010), http://eusecwest.com/esw08/esw08-shah.pdf. G. Hoglund and J. Butler, Rootkits: Subverting the WindowsKernel. Addison-Wesley Professional, 2005.
8. J. Butler, B. Arbaugh, and N. Petroni, “Rˆ2: The exponential growthof rootkit techniques,” in BlackHat USA 2006, 2006 (accessed May 8,2010), http://www.blackhat.com/presentations/ bh-usa-06/BH-US-06-Butler.pdf.

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

9940 572 462 6381 907 438 ijircce@gmail.com

www.ijircce.com

Scan to save the contact details