ETHICAL HACKING AND PREVENTION
PROJECT SYNOPSIS
B.Tech (CSE) Semester VII

Key Logger Creation and Detection

MADE BY:
LAVANYA BHATI - 22103007
BHAVYA MANTOO - 22103085
SHIVAPRASAD ARUNKUMAR FARALE - 22103012
AARUSH GUPTA - 22103030

Submitted To:
Dr. Shobit Tyagi Assistant Professor (Senior Grade)

PROBLEM STATEMENT

Keyloggers are harmful programs that track every key a user presses on their keyboard. This lets hackers get hold of important things like login details, money information, and private chats. Because they work secretly and can do a lot of damage, they are a big problem for computer security. But it's important to understand how they work so that we can make better tools to catch them, make systems safer, and teach people and companies how to stay secure. Doing this research in a responsible way helps us build protection without putting systems at risk.

This project will look at a basic educational keylogger in a completely safe virtual setup. The work will include looking at how it's built, finding signs that it might be bad, and making tools to find it. These tools will include YARA rules, a Python script that checks imports using pefile, and PowerShell scripts that find if the keylogger is staying on a system or acting strangely. The project will end with a detailed report and show how we found the keylogger, what we can do to stop it, and what steps people can take to keep their systems safe.

OBJECTIVE

The following points detail the scope, objectives, and deliverables:

- Examine the internal workings and overall structure of a Python based keylogger, including its modules, data flow, capture mechanisms, storage, and exfiltration logic.
- Determine and document indicators of compromise (IoCs) commonly associated with keylogger activity.
- Create detection scripts and rules using the following tools and approaches:
  - YARA for pattern based scanning
  - Python with the pefile library for binary import inspection and static analysis
  - PowerShell for detecting persistence mechanisms and monitoring processes
- Perform all analysis exclusively within an isolated, sandboxed virtual machine to ensure safety.
- Produce a comprehensive report and demonstration that present ethical detection methods, validated detection artifacts, and recommended defensive measures.

SCOPE

The goal of this project is to learn how a keylogger works and how it can be spotted, without ever running it on a real computer. All work will be done inside a sandboxed virtual machine (VM) that has no internet connection or personal data.

- **Controlled Testing** – The keylogger will be tested only within isolated virtual machines using safe, controlled methods such as static analysis and simulated keystroke inputs. No real user data will be captured or transmitted during the process.

- **Study the keylogger code** – The focus is on finding clues that a program is a keylogger, such as imports of pynput.keyboard, calls that write keystrokes to a file, or code that sends data via smtplib.
- **Detecting the keylogger with YARA** – YARA is a rule-based scanner that searches files for specific strings or patterns. We will write a small set of YARA rules that match the clues we found in step 1 (e.g., the words input "pynput", "keyboard", "log.txt", "smtplib").
- **Collect and document the results** – All YARA matches and psychnet alerts will be compiled into a simple report that includes screenshots of the command-line output, file hashes, and a short explanation of why each match is suspicious.
- **Ethical safeguards** – The VM is completely isolated, no real user data is used, and the keylogger is never run. Every document will contain a disclaimer: "All analysis was performed in a controlled lab environment for defensive research only."

TOOLS AND TECHNOLOGIES

- **Python 3.x** – for developing analysis scripts and detection utilities.
- **YARA** – to detect malicious patterns and suspicious code signatures.
- **PowerShell** – for identifying system-level indicators and persistence methods.
- **VirtualBox / VMware** – to create isolated virtual environments for safe testing.
- **Windows Sysinternals Suite** – for process and startup analysis.
- **Git & GitHub** – for version control and documentation.
- **Microsoft Word / PowerPoint** – for preparing reports and synopsis documentation.

ETHICAL CONSIDERATIONS

Isolation: All activities occur in a sandboxed VM with no network access, ensuring no real user data or systems are exposed.
Static-only analysis: The keylogger code is never executed; only source-code review and binary inspection are performed, complying with responsible-use policies for malware samples.
Transparency: Every document explicitly states that the work was conducted "ethically and in a controlled lab environment."
Legal compliance: The project uses publicly available open-source code and does not redistribute the malicious payload, avoiding copyright or anti-malware law violations.
Defensive intent: Detection methods are based on established research for keylogger identification, such as ML-based classifiers and memory forensic techniques, and are intended to improve defensive tooling rather than facilitate attacks.

By adhering to these constraints, the team demonstrates how to study malicious software responsibly while delivering a practical detection framework.

EXPECTED OUTCOMES

This project is expected to provide a thorough understanding of how keyloggers function and how their activities can be detected through ethical cybersecurity analysis.By studying a Python-based keylogger within a controlled virtual environment, the team will gain practical knowledge of malware behavior, system vulnerabilities, and detection techniques.

The outcome will include the development of a **detection framework** using YARA, Python, and PowerShell to identify suspicious code patterns, system processes, and persistence mechanisms associated with keyloggers. Additionally, the project will produce a detailed **detection and mitigation report** highlighting indicators of compromise, evidence collected, and strategies to prevent keylogger attacks in real-world systems.

Overall, the project aims to enhance the team's analytical, defensive, and ethical skills in cybersecurity.

SYSTEM ARCHITECTURE DIAGRAM

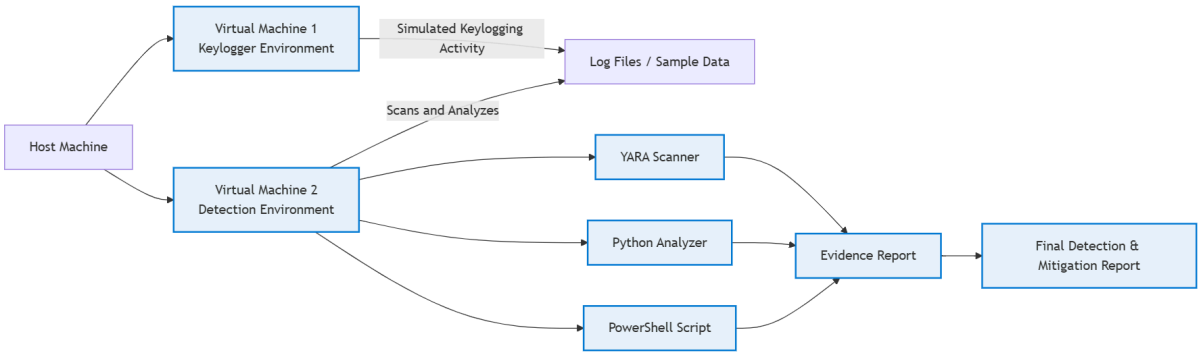

Fig 1: Shows how components interact — keylogger, detection tools, and environment