A

Project Report

On

HOST INTRUSION DETECTION SYSTEM

Submitted in partial fulfillment of the requirement for the degree of

Bachelor of Technology

In

Computer Science and Engineering

By

Divyanshu Tewari	**2261198**
Anuj Bhatt	**2261108**
Sagar Joshi	**2261498**
Gokul Chopra	**2261233**

Under the Guidance of

Mr. Prince Kumar

ASSISTANT PROFESSOR

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

GRAPHIC ERA HILL UNIVERSITY, BHIMTAL CAMPUS

SATTAL ROAD, P.O. BHOWALI,DISTRICT- NAINITAL-263132

2024-2025

STUDENT'S DECLARATION

We, **Divyanshu Tewari**, **Anuj Bhatt, Sagar Joshi**, **Gokul Chopra** hereby declare the work, which is being presented in the project, entitled ' **Host Intrusion Detection System**' in partial fulfillment of the requirement for the award of the degree **Bachelor of Technology (B.Tech.)** in the session **2024-2025**, is an authentic record of my work carried out under the supervision of **Mr. Prince Kumar, Assistant Professor**.

The matter embodied in this project has not been submitted by me for the award of any other degree.

Date:

Divyanshu Tewari

Anuj Bhatt

Sagar Joshi

Gokul Chopra

CERTIFICATE

The project report entitled “Host Intrusion Detection System ” being submitted by Divyanshu Tewari S/o Mr. Pramod Chandra Tewari (2261198) , Anuj Bhatt s/o Mr. G.P Bhatt (2261108) , Sagar Joshi s/o Mr. Dinesh Chandra Joshi (2261498) , Gokul Chopra s/o Mr. Kailash Chandra Chopra (2261233) of B.Tech.(CSE) to Graphic Era Hill University Bhimtal Campus for the award of bonafide work carried out by them. They have worked under my guidance and supervision and fulfilled the requirement for the submission of a report.

Mr. Prince Kumar
(Project Guide)

Dr. Ankur Singh Bisht
(Head, CSE)

ACKNOWLEDGEMENT

We take immense pleasure in thanking the Honorable Director **'Prof. (Col.) Anil Nair (Retd.)',** GEHU Bhimtal Campus to permit me and carry out this project work with his excellent and optimistic supervision. This has all been possible due to his novel inspiration, able guidance, and useful suggestions that helped me to develop as a creative researcher and complete the research work, in time.

Words are inadequate in offering my thanks to GOD for providing me with everything that we need. We again want to extend thanks to our president **'Prof. (Dr.) Kamal Ghanshala'** for providing us with all infrastructure and facilities to work in need without which this work could not be possible.

Many thanks to **'Dr. Ankur Singh Bisht'** (Head, Department of Computer Science and Engineering, GEHU Bhimtal Campus), our project guide **'Mr. Prince Kumar'** (Assistant Professor, Department of Computer Science and Engineering, GEHU Bhimtal Campus) and other faculties for their insightful comments, constructive suggestions, valuable advice, and time in reviewing this report.

Finally, yet importantly, We would like to express my heartiest thanks to our beloved parents, for their moral support, affection, and blessings. We would also like to pay our sincere thanks to all my friends and well-wishers for their help and wishes for the successful completion of this project.

Divyanshu Tewari , 2261198

Anuj Bhatt , 2261108

Sagar Joshi , 2261498

Gokul Chopra , 2261233

Abstract

In the modern digital era, computer systems and networks are highly susceptible to various security threats such as malware, unauthorized access, privilege escalation, and zero-day attacks. With the growing complexity and interconnectivity of systems, traditional defence mechanisms like firewalls and antivirus software are often insufficient to combat sophisticated attacks. Therefore, there is an increasing need for more intelligent and adaptive security solutions that can detect both known and unknown threats effectively.

An **Intrusion Detection System (IDS)** is a crucial component of modern cybersecurity frameworks. It monitors system activities and detects potential security breaches by analysing patterns of behaviour or known attack signatures. Among various types of IDS, a **Host-Based Intrusion Detection System (HIDS)** focuses specifically on monitoring and analysing activities within a single host (e.g., a computer or server). It leverages logs, file integrity, system calls, and user behaviour to detect anomalies.

This project proposes the development of a **Hybrid HIDS** that combines **signature-based detection** (for known attacks) and **anomaly-based detection** (for unknown or zero-day attacks). The system is designed using fundamental **operating system concepts** such as process management, system calls, and user activity logging. It also integrates **machine learning algorithms** to detect abnormal patterns in system behaviour, thereby improving detection accuracy and reducing false positives.

The objective of this project is to provide a lightweight, efficient, and extensible HIDS solution that can be integrated into existing systems for enhanced security monitoring. The hybrid approach not only enables rapid detection of well-known threats but also equips the system with adaptive intelligence to identify new and emerging security risks.

.

TABLE OF CONTENTS

CHAPTER 1: INTRODUCTION

1.1 Prologue

An **Intrusion Detection System (IDS)** is a crucial component of modern cybersecurity frameworks. It monitors system activities and detects potential security breaches by analysing patterns of behaviour or known attack signatures. Among various types of IDS, a **Host-Based Intrusion Detection System (HIDS)** focuses specifically on monitoring and analysing activities within a single host (e.g., a computer or server). It leverages logs, file integrity, system calls, and user behaviour to detect anomalies.

This project proposes the development of a **Hybrid HIDS** that combines **signature-based detection** (for known attacks) and **anomaly-based detection** (for unknown or zero-day attacks). The system is designed using fundamental **operating system concepts** such as process management, system calls, and user activity logging. It also integrates **machine learning algorithms** to detect abnormal patterns in system behaviour, thereby improving detection accuracy and reducing false positives.

The objective of this project is to provide a lightweight, efficient, and extensible HIDS solution that can be integrated into existing systems for enhanced security monitoring. The hybrid approach not only enables rapid detection of well-known threats but also equips the system with adaptive intelligence to identify new and emerging security risks.

1.2 Background and Motivations

The motivation behind this project stems from the increasing frequency and sophistication of cyberattacks that bypass traditional antivirus solutions. Many attackers leverage subtle, host-level activities—such as unauthorized file creation, privilege escalation, and registry modifications—to gain persistent access or execute harmful payloads.

HIDS tools inspect system logs, file integrity, process activity, and user behavior to detect signs of compromise. Signature-based detection relies on known threat patterns, while anomaly-based detection leverages baseline behavior to identify deviations. With the growing complexity of threats like ransomware, rootkits, and zero-day exploits, modern HIDS solutions are evolving to incorporate machine learning, behavioral analysis, and threat intelligence.

As a student of Computer Science and Engineering with an interest in cybersecurity, I was driven to explore how a proactive, host-centric defense mechanism could detect such threats in real time.

.

.

1.3 **Problem Statement**

Despite the availability of several digital whiteboarding applications, current offerings are plagued by notable limitations:

1. With the rapid growth of cyber threats such as malware, ransomware, and unauthorized access, traditional antivirus software often fails to detect sophisticated intrusions.
2. Most intrusion detection systems focus on network-level activity, neglecting host-level anomalies and system behavior.
3. Manual monitoring of system logs and file changes is impractical and inefficient for users and administrators.
4. Existing HIDS solutions are often expensive, complex, or not open-source, making them inaccessible for educational and small-scale use

1.4 **Objectives and Research Methodology**

Project Objectives:

- To design and implement a Host-based Intrusion Detection System using Python and open-source tools.
- To integrate signature-based detection using YARA rules for identifying known threats.
- To implement anomaly detection by learning system behavior and flagging unusual activities.
- To provide a user-friendly web interface for viewing real-time alerts, logs, and system status.
- To monitor key components of the host such as file changes, running processes, and system logs.
- To ensure portability, configurability, and scalability for use on multiple machines

Research Methodology:

To achieve the above objectives, we followed a structured research and development process:

1. Literature Review Studied existing IDS/HIDS models, techniques (signature-based, anomaly-based), and open-source projects.
2. Requirement Gathering: Identified system-level events that are commonly exploited by attackers (e.g., file access, registry edits).
3. Design & Prototyping: We designed UI wireframes and backend architecture.
4. Implementation: Developed the core logic in Python for monitoring file system events and processes.
5. Testing & Optimization: Simulated attack scenarios (malicious file creation, suspicious scripts) to test detection accuracy.
6. Documentation & Deployment: Maintained comprehensive documentation of the project structure, codebase, and user instructions.

1.5 **Project Organization**

The report is structured into several chapters to cover all phases of the project development life cycle:

- Chapter 1: Introduction – Describes the motivation, background, and objectives of the project.
- Chapter 2: Hardware and Software Requirements – Lists the technical specifications needed for both development and deployment.
- Chapter 3: Coding of Functions – Explains key modules, implementation logic, and core functionalities of the system.
- Chapter 4: Snapshots – Displays screenshots of the application's user interface, features, and sample sessions.
- Chapter 5: Limitations – Discusses the current challenges or boundaries faced by the system.
- Chapter 6: Enhancements – Lists the future improvements or new features planned for the system.
- Chapter 7: Conclusion – Provides a closing summary and the overall impact of the project.
- References : Official Documentation , Technical Articles and Papers , Websites.

CHAPTER 2 : PHASES OF SOFTWARE DEVELOPMENT CYCLE

HARDWARE AND SOFTWARE REQUIREMENTS

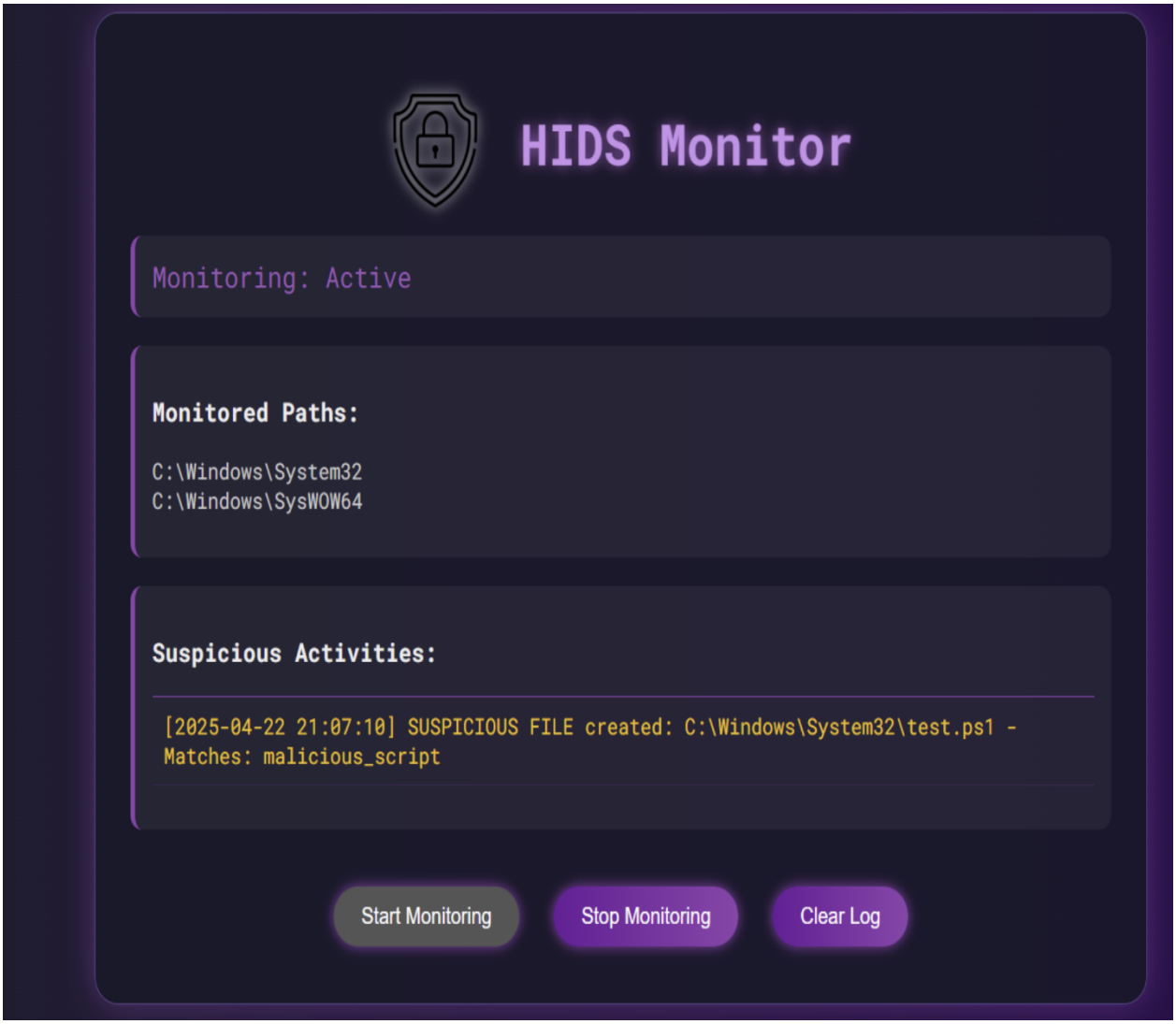
2.1Hardware Requirement

Sl. No	Name of the Hardware	Specification
1.	Processor	Intel Core i5 / AMD Ryzen 5 or higher
2.	RAM	Minimum 8GB (Recommended: 16GB)
3.	Storage	Minimum 256GB SSD (Recommended: 512GB)
4.	Network Interface Card (NIC)	Gigabit Ethernet / Wi-Fi Adapter

2.2 Software Requirements

Sl. No	Name of the Software	Specification
1.	Operating System	Linux (Ubuntu 20.04+ / CentOS 7+) or Windows 10/11
2.	Programming Language	Python 3.8+ / C++ / Java (for system monitoring)
3.	IDS Framework	OSSEC, Snort, Suricata (for reference & integration)
4.	Database	MySQL / PostgreSQL (for logging intrusion data)
5.	Machine Learning Libraries	Scikit-learn, TensorFlow, PyTorch (for anomaly detection)
6.	IDE / Code Editor	PyCharm, VS Code, or Eclipse (for development)

SNAPSHOT


HIDS Monitor
Monitoring: Active
Monitored Paths:
C:\Windows\System32
C:\Windows\SysWOW64
Suspicious Activities:
[2025-04-22 21:07:10] SUSPICIOUS FILE created: C:\Windows\System32\test.ps1 -
Matches: malicious_script
Start Monitoring
Stop Monitoring
Clear Log

LIMITATIONS :

Despite implementing a wide range of features, the current version of the Real-Time Collaborative Whiteboard has some limitations, as outlined below:

- Limited to Host-Level Monitoring: The system is designed to monitor a single machine and does not provide network-wide intrusion detection capabilities..

- Basic Anomaly Detection: The anomaly detection mechanism in this version uses simple behavioral baselines and may not effectively identify sophisticated behavioral deviations without machine learning.

- Static Signature Dependency: Signature-based detection relies on predefined YARA rules; it may not detect new or unknown threats (zero-day attacks) unless continuously updated.

- Platform Specificity:The system currently supports only Windows (due to use of Windows-specific APIs and paths); cross-platform compatibility is not yet implemented.

- Performance Overhead: Continuous monitoring of files and processes can introduce slight performance overhead, especially on resource-constrained systems.

ENHANCEMENTS:

To further improve the application and meet broader user needs, the following

enhancements are planned for future development:

1. **Machine Learning Integration:** Introduce advanced anomaly detection using supervised/unsupervised machine learning models to detect unknown threats more effectively.

2. **Cross-Platform Support:**Extend compatibility to Linux and macOS systems to make the HIDS more versatile and widely usable.

3. **Centralized Management Console:** Develop a central dashboard for monitoring multiple hosts from a single interface, suitable for enterprise environments.

4. **Email/SMS Alert System:**Add real-time alert mechanisms to notify administrators through email, SMS, or mobile apps when suspicious activity is detected.

5. **Process Control and Isolation:**Incorporate external threat intelligence feeds and online malware databases to keep YARA signatures up-to-date automatically.

6. **Log Export and Analytics:**Enable exporting of logs in standard formats (e.g., JSON, CSV) and integrate with SIEM (Security Information and Event Management) tools for deeper analysis.
7. **Threat Intelligence Integration:**Incorporate external threat intelligence feeds and online malware databases to keep YARA signatures up-to-date automatically.

CONCLUSION :

The development of this Host-based Intrusion Detection System (HIDS) marks a significant step toward understanding and implementing effective endpoint security measures. Through a combination of signature-based detection using YARA rules and behavioral monitoring of system activities, the project successfully demonstrates how real-time intrusion detection can be achieved at the host level.

The system provides a lightweight and extensible solution for monitoring file changes, process creation, and potential malicious behavior, all while offering an accessible web-based interface for user interaction. Although limited in scope and reliant on predefined signatures, it lays the groundwork for further development and integration of advanced detection techniques.

This project has not only enhanced technical skills in Python programming, file system handling, and web development, but also deepened the understanding of cybersecurity principles and real-world threat detection strategies. With the potential for future enhancements such as machine learning-based anomaly detection, cross-platform support, and centralized monitoring, this HIDS project holds promise for both educational and practical applications in the field of cyber defense.

REFERENCES :

1. Debar, H., Dacier, M., & Wespi, A. (1999). Towards a taxonomy of intrusion-detection systems. *Computer Networks*, *31*(8), 805–822. https://doi.org/10.1016/s1389-1286(98)00017-6
2. Scarfone, K. A., & Mell, P. M. (2007). *Guide to Intrusion Detection and Prevention Systems (IDPS)*. https://doi.org/10.6028/nist.sp.800-94
3. Denning, D. (1987). An Intrusion-Detection model. *IEEE Transactions on Software Engineering*, *SE-13*(2), 222–232. https://doi.org/10.1109/tse.1987.232894
4. Baker, A. R., Caswell, B., & Poor, M. (2004). Snort 2.1 intrusion detection. In *Syngress Publishing eBooks*. http://ci.nii.ac.jp/ncid/BB01942289
5. OSSEC – Open Source HIDS . https://www.ossec.net
6. Scikit-learn Documentation (for ML algorithms). https://scikit-learn.org/stable/
7. Ghosh, A. K., Schwartzbard, A., & Schatz, M. (1999). Learning program behavior profiles for intrusion detection. *Proceedings of the 1st USENIX Workshop on Intrusion Detection and Network Monitoring*.
8. Bace, R., & Mell, P**.** *Intrusion Detection Systems*. National Institute of Standards and Technology (NIST), 2001. https://csrc.nist.gov/publications/detail/nist-ir/7298/rev-3/final