AI-Powered Intrusion Detection System for Malware: A Hybrid Approach to Detecting Spyware, Ransomware, and Trojans

A PROJECT REPORT

Submitted by,

SHESHA VENKAT GOPAL K	**20211CCS0053**
ARATHI SHREE V	**20211CCS0192**
AMRUTHA SINDHU A	**20211CCS0196**

Under the guidance of,

Dr. N Syed Siraj Ahmed

in partial fulfillment for the award of the degree of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING (CYBERSECURITY)

At

GAIN MORE KNOWLEDGE
REACH GREATER HEIGHTS

PRESIDENCY UNIVERSITY

BENGALURU

JANUARY 2025

PRESIDENCY UNIVERSITY

SCHOOL OF COMPUTER SCIENCE ENGINEERING

CERTIFICATE

This is to certify that the Project report **"AI-POWERED INTRUSION DETECTION SYSTEM FOR MALWARE: A HYBRID APPROACH TO DETECTING SPYWARE, RANSOMWARE, AND TROJANS"** being submitted by Shesha Venkat Gopal K, Arathi Shree V, Amrutha Sindhu A bearing roll number 20211CCS0053, 20211CCS0192, 20211CCS0196 in partial fulfillment of the requirement for the award of the degree of Bachelor of Technology in Computer Science and Engineering is a Bonafide work carried out under my supervision.

Dr. N SYED SIRAJ AHMED
Associate Professor (Selection Grade)
School of CSE & IS
Presidency University

20/1/25

Dr. ANANDARAJ S P
HoD
School of CSE&IS
Presidency University

Dr. L. SHAKKEERA
Associate Dean
School of CSE
Presidency University

Dr. MYDHILI NAIR
Associate Dean
School of CSE
Presidency University

Dr. SAMEERUDDIN KHAN
Pro-VC School of Engineering
Dean -School of CSE&IS
Presidency University

PRESIDENCY UNIVERSITY

SCHOOL OF COMPUTER SCIENCE ENGINEERING

DECLARATION

We hereby declare that the work, which is being presented in the project report entitled **AI-Powered Intrusion Detection System for Malware: A Hybrid Approach to Detecting Spyware, Ransomware, and Trojans** in partial fulfillment for the award of Degree of **Bachelor of Technology** in **Computer Science and Engineering**, **specialization in Cyber Security** is a record of our own investigations carried under the guidance of **N Syed Siraj Ahmed**, **Associate Professor, School of Computer Science Engineering , Presidency University, Bengaluru.**

We have not submitted the matter presented in this report anywhere for the award of any other Degree.

STUDENT NAME	ROLL NO	SIGNATURE
SHESHA VENKAT GOPAL K	20211CCS0053	
ARATHI SHREE V	20211CCS0192	
AMRUTHA SINDHU A	20211CCS0196	

ABSTRACT

The rapid evolution of malware, including spyware, ransomware, and trojans, presents significant challenges for conventional cybersecurity frameworks. Traditional systems often fail to address emerging threats, leaving critical infrastructures exposed to vulnerabilities. This project proposes a hybrid Intrusion Detection System (IDS) that combines anomaly-based and signature-based detection techniques to improve detection accuracy and adaptability. Leveraging the CIC-MalMem2022 dataset, the IDS utilizes five machine learning models—Logistic Regression, K-Nearest Neighbors (KNN), Support Vector Machines (SVM), Decision Tree, and Random Forest. Among these, tree-based models such as Random Forest and Decision Tree exhibited exceptional performance, demonstrating high precision in anomaly detection and effectiveness in multi-class classification for signature-based detection.

The hybrid IDS is structured into two tiers. The initial tier employs anomaly-based detection to identify unusual patterns and detect zero-day threats by analyzing deviations from normal behavior. The second tier focuses on signature-based detection to classify flagged anomalies into specific malware categories, including ransomware, spyware, and trojans. This dual-layered approach minimizes false positives and negatives, offering a more dependable detection mechanism.

Key features of the system include real-time monitoring and scalability. Tools like Pyshark and Python-based logging enable instantaneous threat alerts, ensuring smooth deployment across various environments such as IoT ecosystems, corporate networks, and cloud infrastructures. Advanced preprocessing methods, including feature selection and normalization, optimize the system's efficiency and adaptability.

This research highlights the potential of integrating machine learning techniques with hybrid detection strategies to address modern cybersecurity challenges. Future enhancements may include incorporating deep learning models, enabling incremental learning for continuous adaptation, and extending the system to address cross-domain cybersecurity applications. The proposed hybrid IDS serves as a scalable and adaptive defense mechanism, strengthening the ability to counter evolving malware threats and enhancing proactive security measures.

ACKNOWLEDGEMENT

First of all, we indebted to the **GOD ALMIGHTY** for giving me an opportunity to excel in our efforts to complete this project on time.

We express our sincere thanks to our respected dean **Dr. Md. Sameeruddin Khan**, Pro-VC, School of Engineering and Dean, School of Computer Science Engineering & Information Science, Presidency University for getting us permission to undergo the project.

We express our heartfelt gratitude to our beloved Associate Deans **Dr. Shakkeera L and Dr. Mydhili Nair,** School of Computer Science Engineering & Information Science, Presidency University, and **Dr. Anandaraj S P**, Head of the Department, School of Computer Science Engineering in Cyber Security, Presidency University, for rendering timely help in completing this project successfully.

We are greatly indebted to our guide **Dr. N Syed Siraj Ahmed**, Associate Professor and Reviewer **Mr. Praveen Giridhar Pawaskar**, Assistant Professor, School of Computer Science Engineering & Information Science, Presidency University for his inspirational guidance, and valuable suggestions and for providing us a chance to express our technical capabilities in every respect for the completion of the project work.

We would like to convey our gratitude and heartfelt thanks to the PIP2001 Capstone Project Coordinators **Dr. Sampath A K, Dr. Abdul Khadar and Mr. Md Zia Ur Rahman,** department Project Coordinators **Dr. Sharmasth Vali Y** and Git hub coordinator **Mr. Muthuraj.**

We thank our family and friends for the strong support and inspiration they have provided us in bringing out this project.

Shesha Venkat Gopal K
Arathi Shree V
Amrutha Sindhu A

LIST OF TABLES

LIST OF FIGURES

TABLE OF CONTENTS

CHAPTER-1

INTRODUCTION

1.1 Background

1.1.1 Digital Transformation and Cybersecurity Landscape

The digital revolution has fundamentally transformed the global landscape, creating unprecedented interconnectivity between individuals, organizations, and governments worldwide. This transformation has reshaped how businesses operate, how services are delivered, and how information is shared across global networks. With the widespread adoption of technologies such as cloud computing, Internet of Things (IoT), and artificial intelligence, critical infrastructure, financial systems, healthcare services, and government operations now heavily depend on digital technologies. This dependency has created a complex web of interconnected systems that must be protected to ensure functionality, confidentiality, and integrity. While the integration of digital technologies offers remarkable opportunities for innovation, economic growth, and societal advancement, it has also introduced significant vulnerabilities that require sophisticated and adaptive security measures. The interplay between digital innovation and security has become a focal point of contemporary technological discourse, highlighting the need for resilient cybersecurity frameworks.

1.1.2 Evolution of Cyber Threats

The cybersecurity landscape has witnessed a dramatic evolution in the sophistication and frequency of threats. Early cyberattacks primarily consisted of simple viruses and worms, which caused disruptions but were relatively easy to detect and mitigate. However, modern cyber threats have evolved into complex, multi-vector attacks that can simultaneously exploit multiple vulnerabilities within a system. Malicious actors now employ advanced techniques, leveraging technologies such as artificial intelligence and machine learning to enhance the precision and effectiveness of their attacks. For instance, ransomware attacks have become a prominent threat, encrypting critical data and demanding ransom payments for decryption keys. Similarly, spyware and advanced persistent threats (APTs) operate stealthily, often remaining undetected for extended periods to extract sensitive information or launch attacks at strategically opportune moments. These developments underscore the necessity for cybersecurity solutions capable of adapting to the ever-changing threat landscape and mitigating risks effectively.

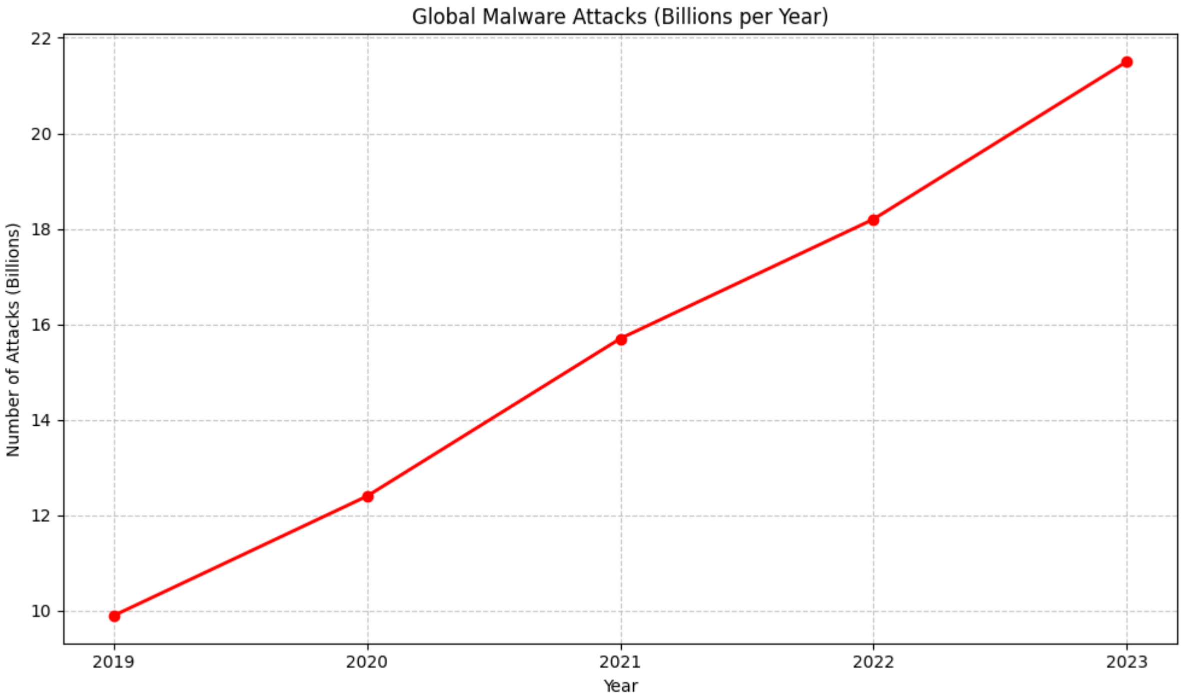


Figure 1. Trends in Malware Attacks Over the Years.

1.1.3 Traditional Security Approaches

Conventional Intrusion Detection Systems (IDS) have historically served as a primary defense against malicious activities by monitoring and analyzing network traffic. These systems operate using two core methodologies—signature-based detection and anomaly-based detection.

Signature-based detection works by comparing incoming data against a database of predefined patterns or signatures of known threats. This approach is effective in identifying established attack patterns, making it highly reliable for addressing previously documented threats. However, it suffers from significant limitations, such as its inability to detect zero-day attacks, where new malware variants exploit vulnerabilities that have yet to be documented. Signature-based systems also struggle against polymorphic and metamorphic malware, which modify their code to evade detection. Moreover, they require frequent updates to their signature databases, leading to delays in identifying emerging threats.

Anomaly-based detection, in contrast, establishes a baseline of normal behavior and flags deviations as potential threats. This approach is advantageous for identifying unknown attacks and zero-day exploits, as it does not rely on prior knowledge of malware patterns. Despite its flexibility, anomaly-based detection often generates high rates of false positives, mistakenly classifying benign activities as malicious. This issue, known as alert fatigue, can overwhelm security teams, reducing efficiency and increasing the likelihood of overlooking genuine threats.

Both methods, when deployed independently, face challenges in balancing accuracy and adaptability. Signature-based systems excel in precision but lack adaptability, whereas anomaly-based systems are adaptable but prone to inaccuracies. Additionally, traditional systems often lack scalability, making it difficult to process large volumes of data generated in modern networks, IoT environments, and cloud infrastructures.

Given these constraints, the integration of advanced methodologies, such as machine learning and hybrid detection frameworks, has become imperative. Hybrid approaches leverage the strengths of both techniques, combining signature-based detection for accuracy with anomaly-based detection for adaptability. This synergy reduces false positives and enhances threat detection, offering a more robust defense mechanism against evolving cyber threats. The development of such hybrid systems represents a significant step forward in addressing the limitations of traditional IDS models.

1.2 Problem Statement

1.2.1 Current Challenges in Cybersecurity

Organizations face an unprecedented volume of security incidents daily, with cyberattacks becoming increasingly sophisticated, targeted, and difficult to detect. The dynamic nature of the cybersecurity landscape—characterized by rapidly evolving threat vectors—compounds the difficulty of maintaining robust defense mechanisms. The financial impact of successful cyberattacks has grown exponentially, encompassing immediate monetary losses, operational disruptions, reputational damage, legal liabilities, and erosion of customer trust. Furthermore, as organizations continue to digitize their operations and adopt emerging technologies, the attack surface expands, providing malicious actors with more opportunities to exploit vulnerabilities. This growing complexity necessitates the development of adaptive, scalable, and efficient security measures capable of addressing both current and future challenges.

1.2.2 Limitations of Existing Detection Systems

Traditional intrusion detection systems are increasingly inadequate in combating modern cyber threats. Signature-based systems, while effective against known threats, rely on predefined patterns and thus cannot detect zero-day exploits or polymorphic malware. These systems are inherently reactive, requiring prior knowledge of threats to function effectively. Conversely, anomaly-based systems, which are better equipped to identify unknown threats, often generate excessive false positives. This phenomenon, known as alert fatigue, burdens security teams with managing large volumes of alerts, reducing their capacity to respond to genuine threats promptly. Moreover, the limitations of traditional systems are exacerbated by the growing sophistication of attackers, who employ evasion techniques designed to bypass conventional detection mechanisms.

1.2.3 Scale and Complexity Issues

The scale and complexity of modern cyber threats present additional challenges that exceed the capabilities of traditional security systems. The exponential growth in data volume and velocity within modern networks necessitates real-time processing and analysis to detect and respond to threats effectively. Traditional systems often lack the computational power and efficiency required to process vast amounts of data without introducing significant delays. This limitation is particularly pronounced in high-throughput environments, such as large enterprise networks, cloud computing platforms, and IoT ecosystems, where even minor delays in threat detection can have catastrophic consequences. Addressing these scale and complexity issues requires the development of advanced systems capable of handling large-scale data processing with high accuracy and efficiency.

1.3 Project Overview

1.3.1 Proposed Hybrid IDS Architecture

This project introduces an innovative hybrid Intrusion Detection System (IDS) that integrates signature-based and anomaly-based detection methodologies with advanced artificial intelligence techniques. The proposed system employs a two-tier framework designed to optimize threat detection capabilities while minimizing false positives. In the first tier, signature-based detection rapidly identifies known threats using a comprehensive and regularly updated database of threat signatures. The second tier leverages anomaly-based detection powered by machine learning algorithms to identify deviations from normal behavior, enabling the detection of novel or emerging threats. By combining these methodologies, the hybrid IDS architecture addresses the fundamental limitations of traditional approaches, offering a robust and adaptive solution for modern cybersecurity challenges.

1.3.2 System Features and Components

The proposed system incorporates several key features and components that enhance its effectiveness in detecting and mitigating cyber threats. At its core, the system utilizes the CIC-MalMem2022 dataset, which provides a diverse collection of malware and benign samples for training and testing machine learning models. Advanced machine learning algorithms, such as deep learning and ensemble techniques, are employed to improve pattern recognition and anomaly detection. These algorithms enable the system to identify complex attack vectors that might evade traditional detection methods. Additionally, the system includes real-time data processing capabilities, allowing it to analyze network traffic and detect threats with minimal latency. A user-friendly dashboard provides security teams with actionable insights, enabling rapid response to detected threats.

1.3.3 Project Scope

The scope of this project encompasses the development of a comprehensive security solution that addresses current cybersecurity challenges while maintaining adaptability to future threats. The system is designed to:

- Process and analyze network traffic in real-time, providing timely threat detection
- Identify and classify various types of cyber threats, including malware, ransomware, spyware and trojans.
- Adapt to new attack patterns through machine learning capabilities, ensuring resilience against evolving threats
- Provide scalable performance to accommodate the needs of organizations of varying sizes and sectors

- Maintain high accuracy while minimizing false positives, reducing alert fatigue and enhancing operational efficiency

1.3.4 Expected Contributions

This project aims to make several significant contributions to the field of cybersecurity. Key contributions include:

- The introduction of a novel hybrid detection approach that effectively combines traditional methodologies with advanced machine learning techniques
- The development of a scalable framework capable of handling large-scale data processing, meeting the demands of modern network environments
- The enhancement of detection accuracy, enabling the identification of sophisticated threats while maintaining low false positive rates
- The establishment of a foundation for future developments in intelligent cybersecurity systems, promoting ongoing innovation in the field
- The creation of a methodology for integrating and optimizing multiple detection approaches, providing a blueprint for the design of next-generation security solutions.

CHAPTER-2
LITERATURE SURVEY

[1] This particular study by **Saeed, Imtithal & Selamat, Ali & Abdelrahman, Ali. (2013)** provides an excellent and comprehensive review of the existing systems in use that detect malware, with emphasis on the more traditional signature-based approaches most commonly used. These are heavily dependent on predefined patterns or signatures to successfully identify known variants of malware and, as such, provide an extremely high degree of accuracy in identifying previously documented and cataloged threats. The paper, however, outlines a number of important limitations of these systems, notably their failure to address novel threats in an appropriate manner. Important issues among these include, but are not limited to, zero-day attacks-the attacks on previously unknown vulnerabilities-or polymorphic malware, which changes its code to avoid detection. To effectively bridge the existing gaps, this research undertakes the integration of a hybrid intrusion detection system that is designed to merge signature-based and anomaly-based methodologies into a cohesive framework. By strategically leveraging advanced machine learning algorithms within this new approach, the new system provides for significantly enhanced adaptability and improved accuracy. Thereby, it becomes capable enough to detect not only previously identified known threats but also completely new threats that have not been encountered before.

[2] This paper by **Ferdous, Jannatul & Islam, Md Rafiqul & Mahboubi, Arash & Islam, Md. (2024)** discusses the different AI-based approaches, especially targeting ransomware detection, focusing highly on the effectiveness and efficiency of the machine learning techniques used in recognizing and analyzing patterns of ransomware. It also focuses on the feature selection role and the optimization of datasets to significantly improve the accuracy of the mechanisms used in detection. Despite the fact that it portrays its strengths and advantages, the research is limited in the fact that it focuses solely on ransomware and does not take into consideration other types of malware that could be spyware or trojans. Therefore, in this particular work, the scope has been broadened in a thoughtful manner to include a wide variety of malware types beyond just ransomware. The proposed solution, by integrating advanced hybrid detection systems into the framework, not only ensures complete coverage across various scenarios but also enhances adaptability, thus effectively addressing the specific limitations identified in the original study. Moreover, this approach significantly improves scalability and robustness, allowing for better performance in diverse environments.

[3] In this enlightening research by **Ok, Emmanuel. (2024)**, a detailed comparison is made between traditional malware detection systems and those that are powered by artificial intelligence solutions. The findings compellingly demonstrate the clear superiority of AI in its capability to manage and respond to sophisticated threats that are becoming increasingly prevalent. This study highlights the remarkable ability of AI to analyze intricate patterns within data and its adaptability in responding to emerging forms of malware that pose risks to cybersecurity. However, it is important to note that despite its value, the study still lacks proper and comprehensive implementation of specific algorithms that are really very crucial to the research. In addition, it fails to adequately address the scalability issue concerning detection systems, which is a very essential aspect of their effectiveness in real-world applications. To overcome these identified shortcomings and limitations, this research strategically uses a variety of diverse machine learning models, including but not limited to Random Forest and K-Nearest Neighbors. These particular models are used together with the well-known CIC-MalMem2022 dataset. This thoughtful combination not only guarantees greater scalability for the system but also significantly enhances its ability to detect an extensive range of malware types, which leads to improved accuracy in the outcomes of detection.

[4] This study by **Saeed, Imtithal, and Selamat (2013)** provides an excellent and comprehensive review of the existing systems in use for detecting malware, with emphasis on the more traditional signature-based approaches most commonly used. These systems are heavily dependent on predefined patterns or signatures to successfully identify known variants of malware and, as such, provide an extremely high degree of accuracy in identifying previously documented and cataloged threats. The paper, however, outlines a number of important limitations of these systems, notably their failure to address novel threats in an appropriate manner. Important issues among these include, but are not limited to, zero-day attacks—the attacks on previously unknown vulnerabilities—or polymorphic malware, which changes its code to avoid detection. To effectively bridge the existing gaps, this research undertakes the integration of a hybrid intrusion detection system that is designed to merge signature-based and anomaly-based methodologies into a cohesive framework. Similarly, Wang, Chen, and Yu (2022) illustrate how such hybrid systems, by integrating signature-based and anomaly-based methods, leverage models like Random Forest to effectively handle complex datasets. They highlight the importance of feature engineering and preprocessing—particularly methods like feature selection and encoding—which significantly improve computational efficiency and detection accuracy. By strategically incorporating these advanced enhancements, the system achieves improved adaptability, enabling it to detect both known and novel threats with greater reliability.

[5] This extensive survey by **Wagh, S., Pachghare, V. K., & Kolhe, S. R. (2013)** exhaustively reviews different types of machine learning techniques especially devised for intrusion detection systems with heavy emphasis on the algorithms' proven efficacy, like decision trees and neural networks, in this crucial field. It provides valuable and meaningful insights into the promising possibilities of machine learning techniques applied exclusively to the challenging domain of malware detection. However, the study does not explore in considerable depth multi-class classification or the various hybrid methodologies. Based on this existing foundation, the current research innovates by implementing hybrid detection systems that effectively combine the specific strengths of both signature-based and anomaly-based detection approaches. Moreover, multi-class classification capabilities are incorporated in the systems, which enhances their functionality by allowing the detection and correct classification of multiple types of malware, including but not limited to spyware, ransomware, and trojans.

[6] This research work by **Kuriyal, Vivekanand et al. (2023)** focuses on various techniques of malware detection based on the development of machine learning in its methodology. In this study, the research emphasis is more on feature selection and model optimization processes, which are two essential components toward improving the overall performance of detection systems. However, though the paper does provide an excellent analysis of different classification methods available in the field, it notably lacks a strong focus on hybrid approaches that could perhaps integrate several types of detection methods to form more robust solutions. The current work adequately addresses this major gap in the literature by carefully incorporating both anomaly-based and signature-based approaches into the development of a robust hybrid intrusion detection system that efficiently leverages the complementary strengths of each methodology. This innovative system goes far beyond increasing accuracy regarding threat detection but is also capable of dynamically evolving to respond to changing conditions, addressing some of the main flaws found in the paper that first presented the approach.

[7] This insightful paper by **Schmitt, Marc. (2023)** covers deeply the use of artificial intelligence in securing smart infrastructures as well as digital industries against the rising menace of malware attacks. It focuses on the critical ways in which artificial intelligence improves the scalability and efficiency of detecting complex malware. However, the focus of the study is malware specific to certain infrastructures that, by its nature, limits its scope to other kinds of environments or classes of malware. To address this recognized limitation, the current research project is designing a scalable hybrid intrusion detection system that is capable of detecting a far greater number of classes of malware. This advanced system successfully integrates a range of signature-based methods that have been specifically designed for the detection of known threats along with new anomaly-based techniques especially developed for the identification of unknown threats, thereby guaranteeing adaptability and generalizability over a range of quite disparate scenarios that may arise.

[8] This highly in-depth study by **Djenna, A.; Bouridane, A.; Rubab, S.; Marou, I.M. (2023)** exhaustively investigates the role of artificial intelligence within such critical processes as malware detection, comprehensive analysis, and effective mitigation. Its focus is particularly placed on the increasingly important techniques for anomaly-based detection. In this paper, with explicit evidence, it effectively shows how anomaly detection can identify unseen threats by carefully examining what is seen to be deviant from the norm. However, the current approach does not contain signature-based techniques. Signature-based techniques are required and critical for proper identification of well-known malware. Therefore, to rectify this weakness, the current research study presents an innovation that integrates both anomaly-based techniques and signature-based methods into a comprehensive hybrid system. This strategic integration results in the system having significantly better accuracy and adaptability for proper detection of known as well as unknown threats that can threaten it.

[9] This research by **Maribana, Thabang; Chindipha, Stones; Brown, Dane. (2023)** very much focuses on the critical role that feature selection plays in the context of malware detection. Indeed, it reveals that proper feature selection contributes to considerable enhancement in the performance of models and to a significant decrease in the overhead in terms of computations. While the study provides a detailed exploration of the concept of feature selection, it does not advance to developing a hybrid detection system or solving the several challenges that could arise with regards to scalability. Still, the present research extends on these findings to build on those previous studies while using state-of-the-art feature selection techniques to construct a novel hybrid detection system in order to transcend those limitations. This ensures not only computational efficiency but also with a marked improvement in the accuracy of detection, thus overcoming the limitations and shortcomings that have been seen in the very initial study.

[10] This paper by **Feroz Khan, A.B.; Kamalakannan, K.; Ahmed, N. Syed. (2023)** presents the intricate integration of methodologies of machine learning with stochastic analysis techniques of patterns that have been designed for the detection of malware and underscores the relevant benefits that can be availed from the integration of these two techniques in enhancing the detection process precision. However, this research has not factored in complexity issues on multi-class classification of numerous types of malware or scale-related issues when applied to real-world environments. These are significant research gaps, and for this reason, the current work is conducted while implementing multi-class classification using tree-based approaches in combination with ensemble approaches. Besides, this hybrid system that has been developed in the course of this research has been structured with care for the sake of handling large datasets for enhancing its practicality and applicability in a wide range of diverse, changing environments.

[11] This research by **Ahmed, N.S.S. (2016)** significantly contributes to the application of rough set theory to the most critical area of intrusion detection, which would show how to effectively improve the classification processes of different malicious activities from the principles of formal concept analysis. Even though this research delivers novel ideas and methodologies, it is limited in its scope because it does not include contemporary machine learning techniques or hybrid detection approaches prominent in the field at the moment. With the results above and the gaps that these studies have, the present research adds another layer of integration by considering feature selection techniques inspired by rough set theory in a holistic hybrid detection framework. Thoughtful integration, therefore, will significantly advance the ability of the system to efficiently process complex data sets while maintaining an excellent level of high detection accuracy.

[12] This paper by **Madhu G. (2022)** introduces an intrusion detection and prevention model using COOT optimization with a hybrid LSTM-KNN classifier. The paper presents how hybrid classifiers improve detection rates in mobile ad hoc networks but is limited in its scope to a specific network environment. Here, the hybrid approach has been extended for the use of broader network environments and tree-based models combined with ensemble techniques that may allow for improved scalability and adaptability across a wider range of datasets and malware types.

[13] This research by the authors **Ahmed, N.S.S., Acharjya, D.P., and Sanyal, S,** takes up the challenge of providing a comprehensive framework for the identification of phishing attacks by using both rough set theory and formal concept analysis as the basis of its techniques. The study has been quite effective in demonstrating and showing how the implementation of these advanced analytical techniques can be highly efficient in increasing the accuracy of detection for social engineering-based attacks, which are very prevalent in the current digital environment. However, the study does not cover an important area that needs to be covered; this includes the integration of anomaly-based detection methods or signature-based detection techniques that will make it possible to have a more holistic approach to malware detection overall. To fill this gap that has been identified in the research, the current work takes up the challenge of integrating rough set-based feature selection into a sophisticated hybrid detection system. This thoughtful integration leads to a highly improved detection accuracy, which means providing better solutions in all categories of malware.

[14] This systematic review by **Maseno, E.M.; Wang, Z.; Xing, H. (2021)** carefully scrutinizes the domain of hybrid intrusion detection systems, focusing on the numerous benefits that result from the integration of the signature-based and anomaly-based approaches. While the paper is effective in highlighting and discussing the various advantages associated with hybrid systems, there are no specific details about the implementation processes or optimization techniques that may improve the understanding of practical applications. Throughout this research, a comprehensive hybrid detection system has been implemented using advanced preprocessing methods. Such methods include essential practices like feature selection and encoding, integral to the optimization of model performance as well as ensuring that the system scales effectively for real-world applications, taking into account the various challenges that may occur in real-world scenarios.

[15] This comprehensive review by **Shaikha, H.K.; Abduallah, W.M. (2017)** delves into the complex evolution of intrusion detection systems, where a significant emphasis has been put on their different architectural designs and the methodologies they utilize. It carefully looks into and highlights the numerous challenges traditional systems face in terms of modern malware threats that have been extremely sophisticated. This, however, does not imply that this review puts forth any concrete or specific solutions as far as the integration of artificial intelligence or machine learning technologies goes to overcome these challenges. Responding to these very serious concerns, the present work presents the advancement in developing a novel hybrid intrusion detection system that is powered by AI. This unique system will have the ability to integrate the signature-based detection method along with the anomaly-based techniques to create a more potent solution. In doing so, the new system developed has improved detection accuracy and also improved adaptability of the system to a great extent and consequently addresses the problems presented by the original study.

[16] This paper by **Öztürk, A. and Hızal, S** performs an in-depth analysis of the use of machine learning models solely for the purpose of malicious software detection while at the same time showing how feature selection can dramatically improve detection performance. However, it should be noted that this study does not explore hybrid approaches or ensemble methods that could potentially give more insights. In this research, different feature selection techniques are thoughtfully combined with ensemble models, such as Random Forest, within the context of a hybrid detection framework designed to maximize efficiency. The strategic integration not only improves the accuracy of detection but also enhances scalability, thus addressing and overcoming the identified limitations in the original study conducted earlier.

[17] In this research paper by **Dai, Z.; Por, L. Y.; Chen, L.; Yang, J.; Ku, C. S.; Alizadehsani, R.; Pławiak, P. (2024)**, the authors have proposed a sophisticated intrusion detection model that is specifically designed to recognize zero-day attacks within previously unseen data by using advanced machine learning techniques. The study effectively demonstrates how tree-based algorithms can play a vital role in detecting these novel threats while also emphasizing the need for optimizing feature engineering to achieve enhanced performance levels. However, it should be mentioned that the study does not use hybrid methodologies, which could potentially offer even greater improvements in terms of detection capabilities and overall effectiveness. To overcome this problem, the present study includes tree-based models in a sophisticated hybrid system. This hybrid system intelligently integrates both signature-based and anomaly-based detection techniques to enhance the detection rates for known as well as newly emerging threats while keeping the system adaptive to the ever-changing malware patterns.

[18] This is an extensive review by **Mahesh, B. (2020)**, where various machine learning algorithms are discussed in detail along with their diverse applications in the field of malware detection. The review provides a comprehensive comparative analysis of different models based on their strengths and weaknesses regarding the identification and mitigation of malware threats. Although it highlights the strength of ensemble methods such as Random Forest, the paper does not provide detailed information on hybrid systems and preprocessing techniques. This work extends the study by developing a hybrid system for malware detection using advanced preprocessing techniques such as feature selection and encoding for the optimization of the model performance. In this way, accuracy increases, and all types of malware are detected at a larger scale.

[19] This systematic literature review by **Matthew G. Gaber, Mohiuddin Ahmed, and Helge Janicke. (2024)** investigates the growing adoption of artificial intelligence in the malware detection area and further details the significant benefits that machine learning can provide in the detection of advanced, evasive strains of malware. Although this is a very robust and comprehensive piece of work, it does miss a few practical implementation details that are necessary for the effective integration of multiple methodologies into a single framework. The course of this research led to the careful development of a hybrid intrusion detection system wherein the inherent strengths of anomaly-based and signature-based approaches are strategically leveraged in order to augment overall effectiveness in security. This brings in machine learning models that are strategically integrated into the existing framework to significantly enhance the system's adaptability and accuracy. This integration specifically addresses and seeks to overcome the various limitations that were outlined in the comprehensive review conducted earlier.

[20] This paper by **Wolsey, A. (2022)** presents an exhaustive review of the most advanced and state-of-the-art techniques available for malware detection based on artificial intelligence. It specifically focuses on the adaptability and scalability offered by machine learning models in the sphere of cybersecurity. In the context of this discussion, this document emphasizes that AI systems present a large number of advantages related to the capabilities for malware detection. Nevertheless, the review does not touch on critical topics like feature engineering or potential benefits arising from hybrid approaches. Building upon these core findings, this work further enhances the present system using advanced sophisticated feature selection techniques in order to increase the performance of the overall system. In addition to this, it also includes both anomaly-based methods and signature-based methods in the design to provide a hybrid approach. Such thoughtful inclusion has provided significant improvement in the detection accuracy along with scalability in environments that change dynamically.

[21] This comprehensive study by **Frank, E. (2024)** compares the detailed traditional approaches used for the purpose of malware detection and the AI-driven systems that clearly dominate the former ones by a significant margin regarding handling and countering complex advanced cybersecurity threats. Yet it is worth mentioning that the study does not present different AI models or discuss an essential role of preprocessing techniques in enhancing the overall performance of these systems. Thus, to fill these serious lacunas in the contemporary literature, this study implements an extensive range of machine learning models, including popular algorithms such as Random Forest and Decision Trees, together with strong preprocessing techniques aiming at improving the effectiveness and accuracy of the detection process. The integration of hybrid detection methods significantly enhances the overall capability of the system, which can effectively detect a diverse and wide range of malware.

[22] This scholarly paper by **Alenezi, M.N.; Alabdulrazzaq, H.; Alshaher, A.A.; Alkharang, M.M. (2020)** critically reviews the current evolution of malware threats and the corresponding detection techniques that have been developed in response. It particularly emphasizes the various limitations that traditional detection systems face when it comes to effectively managing and addressing modern forms of malware. Furthermore, the paper emphasizes the urgent need for the integration of more advanced technologies, such as artificial intelligence, into these detection processes; however, it does not provide any specific strategies or guidelines for actual implementation. Based on these rich insights, this research work is focused on developing an innovative AI-powered hybrid intrusion detection system that combines and integrates the strengths of both signature-based and anomaly-based detection techniques. This advanced system will effectively use the capabilities of machine learning to ensure adaptability along with a high degree of accuracy, addressing specifically those limitations highlighted in the comprehensive review conducted earlier.

CHAPTER-3
RESEARCH GAPS OF EXISTING METHODS

3.1 Limitations of Signature-Based Approaches

Signature-based intrusion detection systems (IDS) work by comparing the incoming data against a database of pre-known malware patterns or "signatures." Though these techniques show high effectiveness in identifying known threats, their static nature basically puts them at a disadvantage in the face of modern cybersecurity challenges. An important point brought forward in the research paper is that one of the significant limitations of signature-based systems is their inability to detect zero-day attacks. Zero-day malware exploits previously unknown vulnerabilities, and without existing signatures, these attacks bypass the traditional detection methods entirely.

Besides polymorphism, polymorphic malware also employs metamorphism. In this type, a particular malware changes its code structure continuously, while the code of metamorphic malware changes its whole code in every iteration. As such, it is almost impossible to detect them by static signature-based systems. This technique of signature-based systems has an additional disadvantage of depending constantly on database updates, making it resource-intensive and hence often accompanied by a delay typically associated with the reaction time to emerging threats in an ever-changing cybersecurity landscape.

3.2 Limitations of Anomaly-Based Approaches

Anomaly-based IDS examine system behavior and look for patterns that don't conform to established baselines of normal activity. This can be effective for detecting novel threats, including zero-day malware, but it is not by any stretch of the imagination perfect. One of the key problems cited with anomaly-based systems is that they tend to produce high false positives. Normal, legitimate activities are frequently classified as malicious, so the security team gets an unnecessarily large number of alarms, which significantly lowers the operational efficiency of the system.

One of the major challenges with this approach is that it is pretty challenging to establish meaningful baselines of normal behavior in dynamic and complex environments. Organizational networks with different types of user interactions, therefore having changing traffic patterns, pose the problem of getting an accurate "normal" or baseline. Anomaly-based systems lack an accurate classification of threats in general. Even though they can sense malicious activities, they cannot specify what or who is the source, so their utility in targeted response crafting is very limited.

3.3 Opportunities for Improvement

Signature-based and anomaly-based IDSs are limited, and the space for developing more robust and adaptive solutions is open. The uploaded paper identifies HIDS as a promising alternative. HIDS combines the accuracy of signature-based methods with the adaptability of anomaly-based techniques, thus providing comprehensive protection against known and unknown threats.

Significant promise is held by machine learning algorithms such as Random Forest, Decision Tree, and Support Vector Machines (SVM), to enhance the accuracy of detection. These algorithms have the capability to learn from previous data, adapt to changing patterns of threats, and minimize both false positives and false negatives. In addition, advanced feature selection techniques can improve the detection process by identifying the most relevant attributes within datasets, thereby enhancing computational efficiency and accuracy in detection.

The paper also focuses on the necessity of using strong and diversified datasets, such as CIC-MalMem2022, which contain diverse malware types and benign samples. In this way, detection models will be trained in realistic scenarios, which will, therefore, prove effective in real-world applications. Real-time processing capabilities enabled by parallel processing and optimization algorithms may ensure scalability and efficiency in high-traffic environments. It can fill the existing research gaps, and the AI-powered hybrid IDS can be a powerful approach to enhance precision, scalability, and adaptability of cybersecurity defenses and be a proactive and reliable solution against evolving threats.

CHAPTER-4

PROPOSED METHODOLOGY

This chapter provides a detailed explanation of the methodology adopted for the design and implementation of the hybrid intrusion detection system. The methodology uses the CIC-MalMem2022 dataset, conducts rigorous preprocessing, and implements advanced machine learning algorithms in order to provide anomaly-based and signature-based malware detection.

4.1 Dataset and Preprocessing Dataset

The CIC-MalMem2022 dataset was selected because it very well covers the different types of malware, such as ransomware, spyware, trojans, and even non-malicious processes. It comprises system-level attributes derived from real-world environments, such as statistics for processes, memory usage, and information about service interactions. The dataset offers more than 30 features, which enable both binary classification for anomaly detection and multi-class classification to be used for malware-type identification.

1. **Data Cleaning and Transformation:** Systematic pre-processing was carried out to enhance the quality of raw data. The first step involved the identification of 559 redundant entries, which were removed to prevent data leakage and overfitting while training the model. Mean or median imputation is used for features such as pslist.threads and handles.nhandles, where missing values are present. Features with a very high percentage of missing values and having low relevance to the target variable were not considered for further analysis.

2. **Outliers:** were identified using the IQR method and then removed to enhance the uniformity of the data set so that the models could pick up important patterns without interference from anomalous values. Normalization Min-Max scaling was used, where all numerical attributes were normalized to a common range of 0 to 1, thus preventing features with large magnitudes from dominating the training process.

3. **Feature Selection:** Feature selection was performed based on a correlation matrix. A few features either had weak associations with the target variable or showed high multicollinearity with other features. The features excluded in this case were

pslist.nprocs64bit, handles.nport, and svcscan.interactive_process_services because their correlations are weak or nearly zero. The most important features retained were those corresponding to process counts, memory usage, and thread activity in order to make the model efficient and computable.

4. **Target Encoding:** There were two different target variables applied to the two stages of detection:
   Class (binary classification): In the anomaly-based detection, it appeared as 0 (benign) and 1 (malicious). Category (multi-class classification): Numerical for the malware type classifications:
   0: Benign
   1: Spyware
   2: Ransomware
   3: Trojan

   The transformations allowed alignment with the machine learning algorithms without loss in readability of the outcome.

4.2 Model Development

4.2.1 Anomaly-Based Detection

Anomaly-based detection was designed as the main approach to detect deviations from normal behavior. The target variable was encoded as the binary Class column. To ensure proper testing, the dataset was split into training (70%) and testing (30%) subsets.

The following machine learning models were used:

1. **Logistic Regression:** This was the baseline linear model for quick comparisons.
2. **K-Nearest Neighbors (KNN):** This model calculated the proximity of data points to detect anomalies.
3. **Support Vector Classifier (SVC):** Developed decision boundaries to distinguish between benign and malicious samples.

4. **Decision Tree:** Represented intricate relationships between features and the target variable.

5. **Random Forest:** Used multiple decision trees to minimize overfitting and improve generalization. Each of the models was trained on the training set and tested on the testing set with precision, recall, F1 score, and accuracy to measure its performance in detecting malicious activity.

4.2.2 Signature-Based Detection

A second layer of detection was established through a signature-based approach, which stressed multi-class classification to differentiate among distinct malware categories. The target variable was represented by the Category column. Consistency was maintained by subjecting the dataset to identical preprocessing procedures.

Tree-based model approaches, including Decision Tree and Random Forest, performed better in solving multi-class classification problems. These models were good at identifying complicated interrelations between features and identifying subtle differences between different types of malware. Ensemble methods, like Random Forest, enhanced the accuracy by combining decisions over multiple decision trees.

4.3 Hybrid Framework

The proposed hybrid framework integrates anomaly-based and signature-based detection, ensuring a comprehensive and scalable solution.

Hybrid Framework: Key Features

1. **Stage 1: Anomaly-Based Detection for Screening**
   This forms the first level of detection to identify malicious activities based on anomalies that occur outside normal behavior. It does a great job in finding new and unknown malware. Tree-based models, namely Random Forest and Decision Tree, were applied, where results revealed improved adaptability and accuracy.
2. **Stage 2: Signature-Based Detection for Classification**
   The threat is then further scanned to classify in detail and identify the malware categories in which it belongs. It ensures accurate identification of the types of known malware, such as spyware, ransomware, trojans. It allows targeted mitigation strategies for each type of malware.

3. **Model Synergy:** The anomaly-based module reduces the likelihood of missing new malware. The signature-based module enhances precision in the identification of known threats. This combination reduces false positives and negatives, improving detection performance.
4. **Efficiency and Scalability:** Tree-based models like Random Forest and Decision Tree were very effective because they can capture complex relationships and patterns in the dataset. The modular design of the framework ensures scalability and can easily be integrated into various cybersecurity systems.

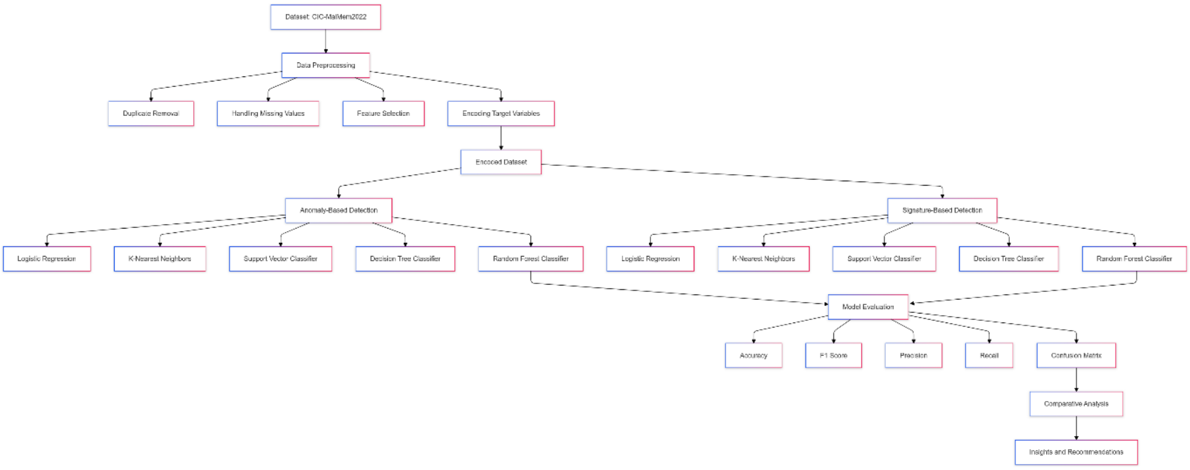


Figure 2. Detailed Workflow of the AI-Based Malware Detection Methodology

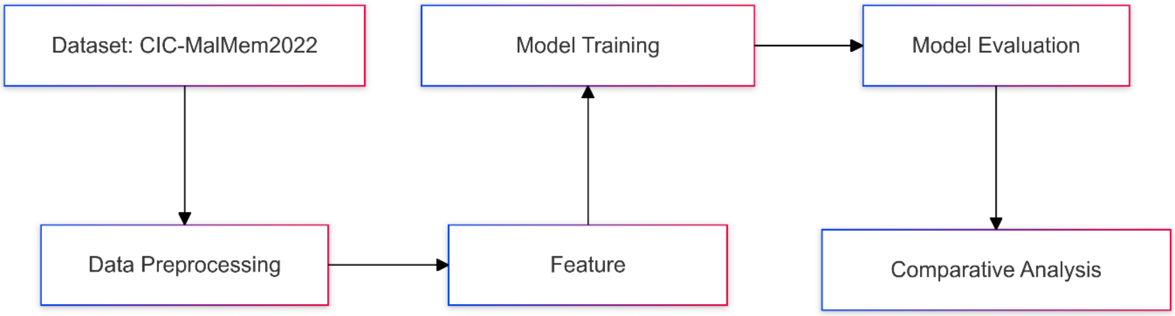


Figure 3. Simplified Workflow of the AI-Based Malware Detection Methodology

CHAPTER-5
OBJECTIVES

The major objectives of the project are as follows:

1. **Hybrid Intrusion Detection System Development:** Implementation of an IDS that incorporates signature-based detection to scan for known malware patterns along with anomaly-based detection capable of identifying unknown or recently emerging malware by monitoring deviations in system behavior. Thus, the hybrid system exploits the strengths of each for effectively detecting a wide cross-section of malware.

2. **Application of Machine Learning Towards Malware Detection:** Apply five machine learning algorithms: Logistic Regression, K-Nearest Neighbors (KNN), Support Vector Machine (SVM), Decision Tree, and Random Forest for the anomaly-based and signature-based detection approach. The models will be trained and evaluated on their capabilities to detect various types of malware such as spyware, ransomware, and trojans.

3. **Train and evaluate the models on the CIC-MalMem2022 dataset:** To use the CIC-MalMem2022 dataset, a rich dataset containing malware samples and benign instances, for training and evaluating the machine learning models. This dataset will provide a realistic basis for testing the system's detection capabilities across a wide range of malware types.

4. **Optimize the Detection System through Rigorous Preprocessing:** To preprocess the dataset by removing duplicates, handling missing values, and feature selection. These preprocessing steps are intended to improve model training efficiency and detection performance, so that only relevant features are used.

5. **Evaluate Model Performance Using Comprehensive Metrics:** Evaluate the IDS models based on multiple metrics such as Accuracy, F1 Score, Precision, Recall, and Confusion Matrix. These will provide detailed insights into how well the models are performing at detecting malware and how well they are distinguishing between benign and malicious activity.

6. **Accurate Malware Detection:** In order to achieve high accuracy of the detection of different kinds of malware, the scope would be narrowed down and targeted towards tree-based models, such as Random Forest and Decision Tree, famous for their excellent performance within anomaly-based and signature-based detection frameworks.

7. **Design a Scalable and Adaptive System:** Ensure the IDS system can grow and adapt; it should work in all places, such as business networks, cloud services, and personal computers. It should be able to locate new types of malware appearing and get better over time with little changes.

8. **To reduce false positives (flagging safe instances as malware) and false negatives (missing actual malware):** This hybrid system attempts to be a balance between precision and recall, providing a real-world, practical solution while being highly reliable for deployment in real-world cybersecurity.

9. **Compare detection methods as a basis for developing this hybrid approach:** Compare anomaly-based and signature-based methods of detection, point out the strengths and weaknesses associated with each, and based on this, guide towards developing a hybrid system capable of incorporating the best in both approaches. Finally discuss future optimization of non-tree-based models.

We want to find ways to improve non-tree-based models, like SVM and Logistic Regression, because they are not doing as well in multi-class classification (signature-based detection). In the future, research will look at making these models better or using advanced techniques like deep learning to increase detection accuracy.

CHAPTER-6

SYSTEM DESIGN & IMPLEMENTATION

6.1 System Architecture

The system is built on a modular, two-tier architecture designed to optimize threat detection accuracy and minimize false positives. The first tier is an **anomaly-based detection layer**, which acts as a filter for identifying deviations from normal behavior within the network or system. This layer focuses on capturing novel and emerging threats that traditional systems often miss. It relies on advanced machine learning models to flag suspicious activities, ensuring the system remains adaptive to new and evolving malware.

The second tier is a **signature-based detection layer**, responsible for classifying flagged anomalies into specific malware categories, such as spyware, ransomware, and trojans. This layer leverages predefined patterns and known signatures to ensure precise identification of threats. By combining these layers, the system balances the adaptability of anomaly detection with the precision of signature-based methods, creating a hybrid framework that enhances overall security.

The architecture also integrates real-time monitoring capabilities, enabling the IDS to process live data streams efficiently. This design ensures the system is versatile enough to handle diverse deployment scenarios, ranging from corporate networks to IoT environments and cloud-based infrastructures.

6.2 Dataset Overview and Preprocessing

The CIC-MalMem2022 dataset was chosen as the foundation for the system due to its comprehensive coverage of various malware types, including ransomware, spyware, and trojans, alongside benign instances. This dataset provides a rich source of features, such as memory usage statistics, process counts, and thread activity, which are critical for effective malware detection.

Data preprocessing was a crucial step to ensure the reliability and accuracy of the machine learning models. The process began with **data cleaning**, where redundant entries were identified and removed to prevent overfitting and data leakage. Missing values were handled using imputation techniques, such as mean or median substitution, to maintain dataset integrity.

Next, **outliers** were detected using the Interquartile Range (IQR) method and subsequently removed to eliminate noise that could interfere with model training. **Normalization** was performed using Min-Max scaling to ensure that features with larger magnitudes did not dominate the training process. This transformation scaled all numerical attributes to a uniform range of 0 to 1.

Feature selection was performed using a correlation matrix to identify and retain the most relevant attributes for malware detection. Features with weak correlations to the target variable or high multicollinearity were excluded to enhance computational efficiency. Finally, target variables were encoded to support both binary classification (benign vs. malicious) and multi-class classification (specific malware categories).

6.3 Model Development

The IDS employs a diverse set of machine learning models to leverage their unique strengths in anomaly-based and signature-based detection.

For **anomaly detection**, the following models were implemented:

- **Logistic Regression:** A baseline linear model used for comparative purposes.
- **K-Nearest Neighbors (KNN):** Detects anomalies by measuring the proximity of data points to normal patterns.
- **Support Vector Classifier (SVC):** Constructs decision boundaries to separate benign and malicious samples.
- **Decision Tree:** Captures complex feature interactions and relationships.
- **Random Forest:** Combines multiple decision trees to improve generalization and reduce overfitting.

For **signature-based detection**, the focus was on tree-based models:

- **Decision Tree:** Efficiently handles multi-class classification problems by mapping intricate feature relationships.

- **Random Forest:** An ensemble method that aggregates predictions from multiple decision trees to enhance classification accuracy.

These models were evaluated using metrics such as accuracy, precision, recall, F1 score, and confusion matrix. This multi-metric evaluation ensured a comprehensive understanding of model performance across different scenarios.

6.4 Hybrid Framework Implementation

The hybrid framework integrates the anomaly-based and signature-based layers into a cohesive system. The **first stage**, anomaly-based detection, acts as a screening mechanism, flagging activities that deviate from established baselines. This layer is particularly effective at identifying novel malware types, such as zero-day attacks.

The flagged anomalies are then passed to the **second stage**, the signature-based detection layer, where they are categorized into specific malware types. This dual-layered approach significantly reduces the likelihood of false positives and negatives, providing a balanced and reliable detection mechanism.

The synergy between these layers ensures the system is both adaptive and precise. Anomaly-based detection provides flexibility and scalability, while signature-based detection delivers targeted responses, making the hybrid IDS an effective tool for modern cybersecurity challenges.

6.5 Real-Time Monitoring and Scalability

To address the demands of dynamic and high-volume environments, the IDS incorporates real-time monitoring capabilities. Tools like Pyshark and Wireshark are used for live network traffic analysis, while Python logging and SMTP facilitate real-time alerts and notifications. This ensures that threats are detected and mitigated promptly, minimizing potential damage.

The system’s modular design supports scalability, allowing it to adapt to varying data loads in environments such as corporate networks, IoT systems, and cloud platforms. Parallel processing and optimized feature selection further enhance the system's efficiency, ensuring high performance even in resource-constrained settings.

6.6 System Testing and Deployment

The system was rigorously tested using a 70-30 train-test split on the CIC-MalMem2022 dataset. This ensured the models were exposed to diverse scenarios, including rare and complex malware types. During deployment, the system was designed for seamless integration with existing cybersecurity infrastructures, providing comprehensive threat reports and actionable insights to security analysts.

CHAPTER-7

TIMELINE FOR EXECUTION OF PROJECT

Month	Phase	Week	Key Activities

October 2024	Foundation Phase	Week 1-2 (Oct 1-13)	Complete literature review;
			Finalize requirement analysis;
			Initial system architecture design;

October 2024	Foundation Phase	Week 3-4 (Oct 14-27)	Complete system architecture design; Set up development environment; Initialize data collection framework; Begin data preprocessing; Establish project management tools and workflows
November 2024	Development Phase	Week 1-2 (Oct 28-Nov 10)	Complete feature extraction development; Implement initial machine learning models; Create training pipeline; Start data collection for training
November 2024	Development Phase	Week 3-4 (Nov 11-24)	Complete initial model training; First round of validation; Begin detection module implementation; Start response system development
November 2024	Development Phase	Week 5 (Nov 25-Dec 1)	Complete core system integration; Begin preliminary testing; Start documentation

December 2024	Integration and Finalization Phase	Week 1-2 (Dec 2-15)	Complete system integration;
			Perform comprehensive testing;
			Implement optimizations;
			Finalize core functionalities;
			Draft documentation

December 2024	Integration and Finalization Phase	Week 3 (Dec 16-22)	Conduct performance evaluation;
			Final system optimizations;
			Finalize documentation;

December 2024	Integration and Finalization Phase	Week 4 (Dec 23-31)	Final testing and quality assurance;
			System deployment;
			Project handover;
			Final documentation;
			Project closure

Table 1. Timeline For Execution of Project

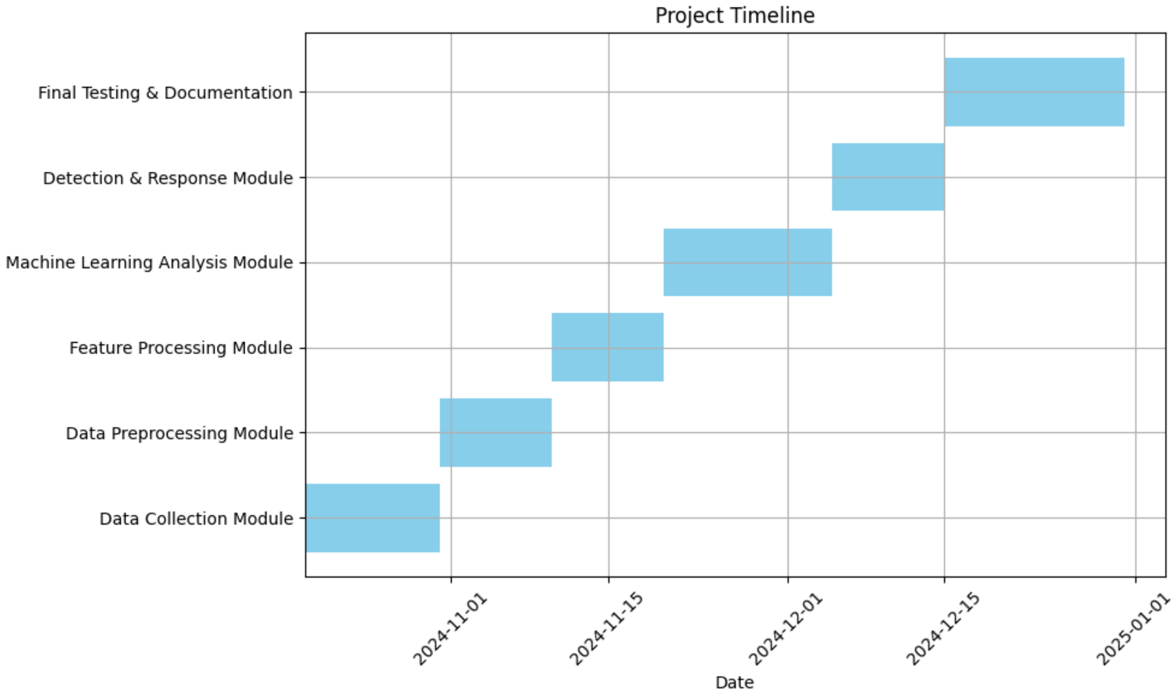


Figure 4. Project Timeline – Gantt Chart

CHAPTER-8
OUTCOMES

One of the primary achievements was the system's remarkable accuracy in detecting both known and unknown threats. This was made possible through the innovative integration of anomaly-based and signature-based detection methods, which complemented each other's strengths. Anomaly-based detection excelled in identifying novel and emerging threats, while signature-based detection provided precise identification of known malware types such as spyware, ransomware, and trojans. This hybrid approach not only reduced the overall detection errors but also improved the system's adaptability in dynamic environments.

Reducing false positives and false negatives was a critical focus of this project. Traditional anomaly detection systems often suffer from high false positive rates, overwhelming security teams with unnecessary alerts. The hybrid IDS addressed this issue by employing advanced machine learning algorithms like Random Forest and Decision Tree, which were optimized for feature selection and data preprocessing. This optimization led to a significant reduction in false positives, ensuring that legitimate activities were not mistakenly flagged as malicious. Simultaneously, the system minimized false negatives, thus reducing the risk of missing actual threats. These improvements enhanced the system's operational efficiency and reliability, making it suitable for deployment in real-world scenarios.

The system was designed with scalability and adaptability in mind, making it highly versatile for various environments. In corporate networks, the IDS handled large volumes of traffic effectively, identifying threats without delays. Its modular architecture ensured compatibility with IoT setups, where resource constraints often pose challenges. In cloud-based infrastructures, the system demonstrated its ability to manage high data loads while maintaining detection accuracy. Real-time monitoring capabilities were integrated into the design, enabling the system to analyze live network traffic, detect threats instantly, and generate alerts. This real-time functionality ensured that organizations could respond to incidents promptly, minimizing potential damage.

Another noteworthy outcome was the system's ability to classify malware types accurately. By employing tree-based models, the signature-based detection layer categorized flagged threats into specific malware groups. This granularity provided actionable insights, enabling organizations to implement targeted mitigation strategies tailored to each malware type. For instance, specific measures could be deployed for ransomware attacks while employing different strategies for spyware or trojan threats. This level of detailed classification significantly strengthened the system's role as a comprehensive cybersecurity tool.

The preprocessing of the CIC-MalMem2022 dataset played a crucial role in enhancing the system’s performance. The dataset, which included a diverse set of malware samples and benign instances, underwent rigorous preprocessing to ensure data quality. Steps such as removing redundant entries, handling missing values, and applying normalization techniques improved the dataset's usability. Feature selection based on correlation matrices further optimized the input for machine learning models, ensuring that only the most relevant features were considered. These efforts reduced computational overhead, enabling faster and more efficient processing, which is essential for real-time cybersecurity applications.

The outcomes of this project extend beyond its immediate implementation, contributing to the broader field of cybersecurity research. The hybrid IDS serves as a foundational framework for future innovations, including the integration of deep learning techniques and the exploration of alternative datasets. Its design demonstrates the effectiveness of combining anomaly-based and signature-based approaches, setting a precedent for the development of scalable and adaptive detection systems. By addressing the limitations of traditional intrusion detection systems, this project paves the way for more robust solutions capable of tackling the evolving threat landscape.

CHAPTER-9

RESULTS AND DISCUSSION

9.1 Overview of Experimental Results

The experimental results revealed significant insights into the effectiveness of both anomaly-based and signature-based detection approaches for malware identification. Through comprehensive evaluation using multiple machine learning models and metrics, our study demonstrates the robust potential of AI-powered approaches in cybersecurity applications.

9.2 Anomaly-Based Detection Performance

The experimental evaluation demonstrated exceptional performance across all implemented models, particularly in anomaly-based detection scenarios. Random Forest emerged as the leading performer, achieving near-perfect accuracy at 99.99% and optimal scores across all evaluation metrics. This remarkable performance underscores the model's robust capability in distinguishing between normal and malicious behavior patterns.

Model Used	Accuracy (%)	F1 Score	Precision	Recall
Random Forest	99.99	1.00	1.00	1.00
Logistic Regression	99.47	0.99	0.99	0.99
KNN	99.85	1.00	1.00	1.00
SVC	98.64	0.99	0.99	0.99
Decision Tree	99.98	1.00	1.00	1.00

Table 2. Performance Of Models in Anomaly-Based Detection

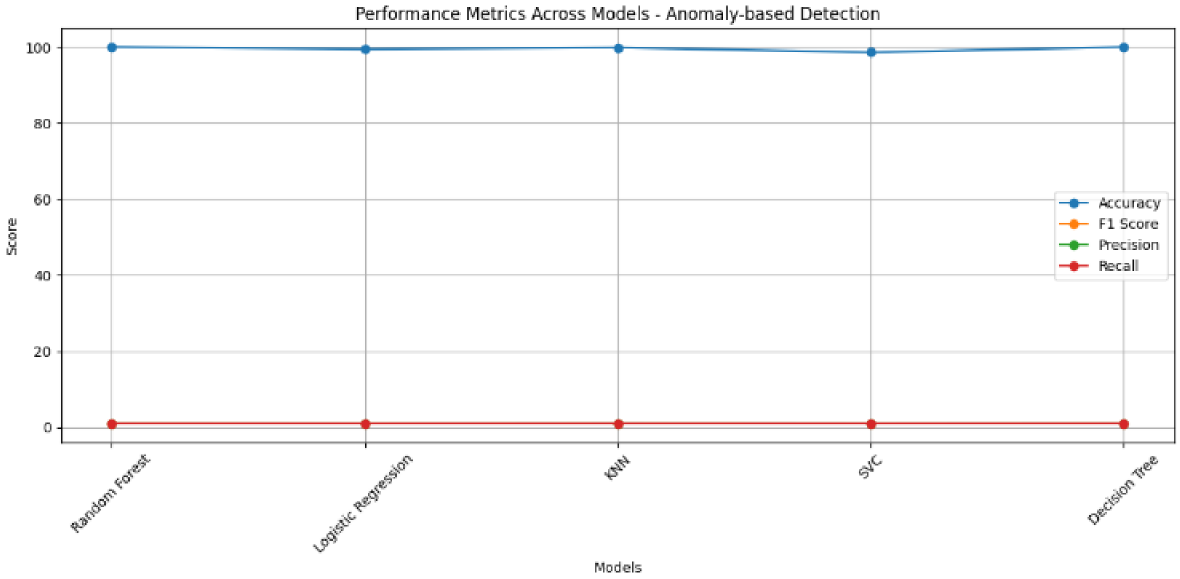


Figure 5. Performance Metrics Across Models – Anomaly-Based Detection

9.3 Signature-Based Detection Analysis

In the more complex domain of signature-based detection, our models demonstrated varying levels of effectiveness in distinguishing between different malware categories. Random Forest maintained its superior performance with 96.85% accuracy and 0.98 scores across F1, precision, and recall metrics, demonstrating its robust capability in multi-class classification tasks.

Model Used	Accuracy (%)	F1 Score	Precision	Recall
Random Forest	96.85	0.98	0.98	0.98
Logistic Regression	70.31	0.69	0.71	0.70
KNN	82.05	0.82	0.82	0.82
SVC	66.93	0.61	0.69	0.67
Decision Tree	93.64	0.94	0.94	0.94

Table 3. Performance Of Models In Signature-Based Detection

9.4 Comparative Performance Analysis

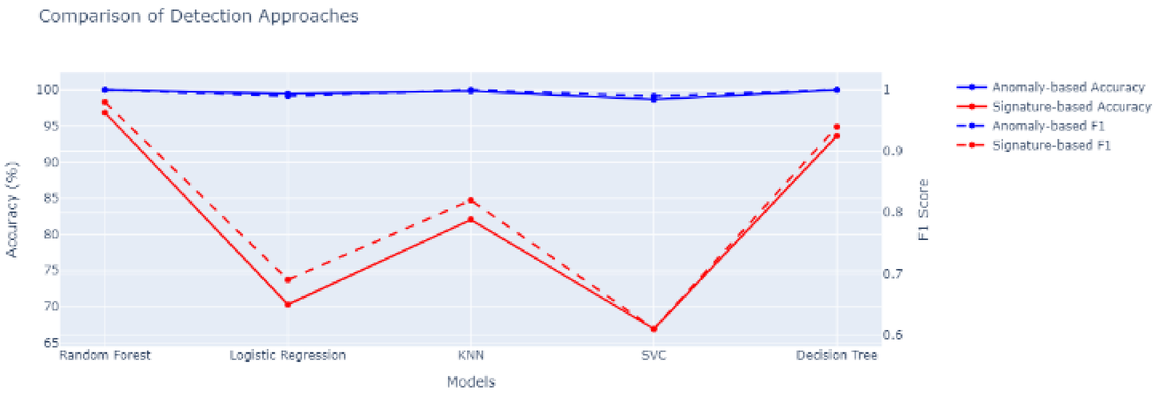


Figure 6. Comparion of Detection Approaches

The comparative analysis between detection approaches, visualized in Figure 6, reveals several intriguing patterns. The generally lower performance in signature-based detection compared to anomaly-based detection can be attributed to the increased complexity of differentiating between multiple malware categories rather than simple binary classification.

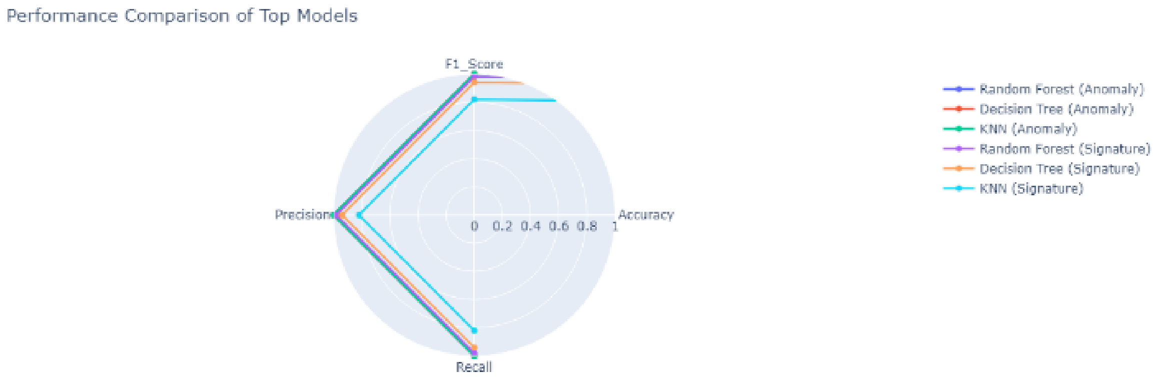


Figure 7. Performance Comparision Of Top Models

Figure 7 presents a performance comparison of the top models across both detection approaches, clearly illustrating the superiority of tree-based algorithms. This visualization helps understand the relative strengths and limitations of different models in handling various detection scenarios.

9.5 Impact of Preprocessing and Feature Selection

The preprocessing steps, including the removal of duplicate entries and careful feature selection, proved crucial for achieving these strong results. The selected feature set, after removing less correlated attributes, contributed significantly to the models' high performance. The 70:30 train-test split provided sufficient data for both training and evaluation, while the comprehensive evaluation metrics offered detailed insights into model behavior.

9.6 Model-Specific Performance Analysis

Random Forest's exceptional performance in both approaches (99.99% for anomaly-based and 96.85% for signature-based) demonstrates its robust capability in handling complex security data. The Decision Tree's strong performance (99.98% and 93.64% for anomaly and signature-based respectively) indicates that even single tree models can effectively capture the decision boundaries necessary for malware detection.

9.7 Practical Implications and Recommendations

Based on our findings, we recommend:

1. Implementing Random Forest as the primary algorithm for both detection approaches
2. Using anomaly-based detection as an initial screening mechanism
3. Considering computational resources when selecting between models
4. Implementing hybrid systems that leverage the strengths of both approaches

9.8 Limitations and Future Research Directions

Several promising research directions emerge from our findings, highlighting opportunities to enhance malware detection systems. Investigating advanced feature selection techniques could improve the identification of critical attributes, while exploring deep learning approaches may enable better handling of complex and evolving malware variants. Developing adaptive learning mechanisms that can dynamically respond to new threats is another key area for future work. Additionally, optimizing model architectures for real-time detection can enhance efficiency and scalability in practical applications. Further research into the impact of different preprocessing techniques may reveal methods to refine data preparation and improve model performance. Examining ensemble learning strategies could also provide insights into leveraging multiple models for improved accuracy and robustness. Moreover, integrating explainable AI methods may help increase transparency and trust in detection systems, while investigating adversarial resilience techniques could fortify models against sophisticated evasion attempts. Lastly, studying the scalability and deployment of these models in distributed and resource-constrained environments remains a vital direction for future exploration.

CHAPTER-10

CONCLUSION

The development of the AI-Powered Hybrid Intrusion Detection System (IDS) represents a critical advancement in the field of cybersecurity, addressing the limitations of traditional intrusion detection methods with a novel approach that combines anomaly-based and signature-based detection. This hybrid system leverages the adaptability of anomaly detection to identify novel threats and the precision of signature detection to classify known malware types. The integration of these methodologies creates a robust, scalable, and efficient framework that is well-suited to combat the increasingly sophisticated and dynamic nature of modern cyber threats.

The system achieved exceptional performance metrics during testing, with high accuracy in detecting both known and unknown threats. By employing advanced machine learning algorithms such as Random Forest and Decision Tree, the IDS demonstrated its ability to handle complex datasets and identify intricate patterns in network behavior. These capabilities were further enhanced by the rigorous preprocessing of the CIC-MalMem2022 dataset, which included data cleaning, outlier removal, normalization, and feature selection. These steps ensured high-quality input data, which is crucial for effective model training and evaluation.

A key strength of this system lies in its ability to significantly reduce false positives and negatives, which are major challenges in traditional detection systems. High false positive rates can overwhelm security teams with unnecessary alerts, while false negatives can leave systems vulnerable to undetected threats. The hybrid IDS successfully mitigates both issues, striking a balance between precision and recall that ensures reliable threat detection and minimal operational disruption. This improvement not only enhances the system's reliability but also makes it a practical tool for real-world deployment.

The system’s modular and scalable architecture allows it to adapt to various environments, including corporate networks, IoT ecosystems, and cloud infrastructures. Its real-time monitoring capabilities, supported by tools like Pyshark and Wireshark, enable live threat detection and instant response, making it highly effective in dynamic and high-traffic settings. This adaptability ensures that the IDS remains relevant in an evolving threat landscape, where attackers continually develop new methods to bypass traditional defenses.

The project also underscores the importance of comprehensive malware classification. By accurately categorizing threats into specific types, such as spyware, ransomware, and trojans, the system enables targeted mitigation strategies. This granularity enhances the ability of organizations to respond effectively to incidents, minimizing potential damage and ensuring continuity of operations. Additionally, the system's ability to process large datasets efficiently ensures that it can handle the demands of modern cybersecurity infrastructures.

Beyond its immediate application, the AI-Powered Hybrid IDS contributes to the broader field of cybersecurity research. The framework developed in this project serves as a foundation for future advancements, such as the integration of deep learning techniques, blockchain-based security, and natural language processing. These innovations could further enhance the system’s adaptability, scalability, and effectiveness, addressing even more complex and evolving threats. The research also highlights the potential for exploring additional datasets to expand the system's applicability across diverse domains.

The implications of this project extend to industries where cybersecurity is critical, such as finance, healthcare, government, and cloud services. The system’s ability to deliver high accuracy, reduced false alarms, and real-time monitoring makes it a valuable asset for these sectors. Its ease of integration with existing cybersecurity infrastructures further enhances its deployment potential, providing a practical and reliable solution for organizations facing the challenges of modern cyber threats.

REFERENCES

[1] Imtithal A. Saeed, Ali Selamat, Ali M. A. Abuagoub . A Survey on Malware and Malware Detection Systems. International Journal of Computer Applications. 67, 16 (April 2013), 25-31. DOI=10.5120/11480-7108

[2] Ferdous, Jannatul & Islam, Md Rafiqul & Mahboubi, Arash & Islam, Md. (2024). AI-Based Ransomware Detection: A Comprehensive Review. IEEE Access. 12. 136666-136695.

[3] Ok, Emmanuel. (2024). AI-Powered Malware Analysis: A Comparative Study of Traditional vs. AI-Based Approaches.

[4] B. -X. Wang, J. -L. Chen and C. -L. Yu, "An AI-Powered Network Threat Detection System," in IEEE Access, vol. 10, pp. 54029-54037, 2022, doi: 10.1109/ACCESS.2022.3175886.

[5] KishorWagh S, Pachghare VK, Kolhe SR (2013) Survey on intrusion detection system using machine learning techniques. Int J Comput Appl 78(16):30–37. https://doi.org/10.5120/13608-1412

[6] Vivekanand Kuriyal , Dibyahash Bordoloi , D.P.Singh , Vikas Tripathi. “A Comprehensive Study on Malware Detection Techniques Using Machine Learning”

[7] Marc Schmitt, Securing the digital world: Protecting smart infrastructures and digital industries with artificial intelligence (AI)-enabled malware and intrusion detection, Journal of Industrial Information Integration, Volume 36, 2023, 100520, ISSN 2452-414X, https://doi.org/10.1016/j.jii.2023.100520.

[8] Djenna, A.; Bouridane, A.; Rubab, S.; Marou, I.M. Artificial Intelligence-Based Malware Detection, Analysis, and Mitigation. Symmetry 2023, 15, 677. https://doi.org/10.3390/sym15030677

[9] Maribana, Thabang & Chindipha, Stones & Brown, Dane. (2023). Feature Selection in Malware Detection.

[10] Khan, A.B.F., Kamalakannan, K. & Ahmed, N.S.S. Integrating Machine Learning and Stochastic Pattern Analysis for the Forecasting of Time-Series Data. *SN COMPUT. SCI.* **4**, 484 (2023). https://doi.org/10.1007/s42979-023-01981-0

[11] Ahmed, N. S. S., (2016) An application of containing order rough set for analyzing data of intrusion detection. An Interdisciplinary Journal of Scientific Research & Education, Vol. 2, No. 5, pp. 52-57.

[12] G., M. (2022) “Design of Intrusion Detection and Prevention Model Using COOT Optimization and Hybrid LSTM-KNN Classifier for MANET”, *EAI Endorsed Transactions on Scalable Information Systems*, 10(3), p. e2. doi: 10.4108/eetsis.v10i3.2574.

[13] Ahmed, N. S. S., Acharjya, D. P., & Sanyal, S. (2017) A framework for phishing attack identification using rough set and formal concept analysis. International Journal of Communication Networks and Distributed Systems, Vol. 18, No. 2, pp. 186-212. https://doi.org/10.1504/IJCNDS.2017.082105

[14] Maseno, E. M., Wang, Z., & Xing, H. (2021). A Systematic Review on Hybrid Intrusion Detection System. Security and Communication Networks, 2022(1), 9663052. https://doi.org/10.1155/2022/9663052

[15] Shaikha, H.K. and Abduallah, W.M., 2017. A Review of Intrusion Detection Systems. Academic Journal of Nawroz University, 6(3), pp.101-105.

[16] A. Öztürk and S. Hızal, “Detection and Analysis of Malicious Software Using Machine Learning Models”, SAUCIS, vol. 7, no. 2, pp. 264–276, 2024, doi: 10.35377/saucis...1489237.

[17] Dai, Z., Por, L. Y., Chen, L., Yang, J., Ku, C. S., Alizadehsani, R., & Pławiak, P. (2024). An intrusion detection model to detect zero-day attacks in unseen data using

machine learning. PLOS ONE, 19(9), e0308469. https://doi.org/10.1371/journal.pone.0308469
[18] Mahesh, B., 2020. Machine learning algorithms-a review. International Journal of Science and Research (IJSR).[Internet], 9(1), pp.381-386. DOI:10.21275/ART20203995
[19] Matthew G. Gaber, Mohiuddin Ahmed, and Helge Janicke. 2024. Malware Detection with Artificial Intelligence: A Systematic Literature Review. ACM Comput. Surv. 56, 6, Article 148 (June 2024), 33 pages. https://doi.org/10.1145/3638552
[20] Wolsey, A. (2022). The State-of-the-Art in AI-Based Malware Detection Techniques: A Review. ArXiv. https://doi.org/10.48550/arXiv.2210.11239
[21] Frank, E., 2024. AI-Powered Malware Analysis: a Comparative Study of Traditional vs. AI-Based Approaches.
[22] Alenezi, M. N., Alabdulrazzaq, H. K., Alshaher, A. A., & Alkharang, M. M. (2022). Evolution of Malware Threats and Techniques: a Review. *International Journal of Communication Networks and Information Security (IJCNIS)*, *12*(3). https://doi.org/10.17762/ijcnis.v12i3.4723

APPENDIX-A

PSUEDOCODE

Data Preprocessing and Filtering

Purpose: Load dataset, filter categories, reduce dataset size, and prepare it for modeling.

```
1. IMPORT required libraries:
   - pandas for data handling.

2. LOAD dataset:
   a. READ dataset from the provided CSV file ("Obfuscated-MalMem2022.csv").
   b. DISPLAY dataset structure and basic information (e.g., number of rows, columns, data types).
   c. PRINT the first few rows for visual inspection.

3. DEFINE parameters:
   a. SPECIFY the column name containing malware categories (e.g., "Category").
   b. SET required sample sizes for each malware type (e.g., Benign: 2000, Ransomware: 1000).

4. INITIALIZE an empty DataFrame to store filtered data.

5. FILTER data based on categories:
   a. ITERATE through each category and its required sample size.
   b. IF the category represents specific malware types (e.g., Spyware):
      - FILTER rows containing the category keyword (case-insensitive).
   c. ELSE:
      - FILTER rows that exactly match the category name.

6. RANDOMLY SAMPLE the specified number of rows for each category:
   a. IF the number of available rows is less than the required count, TAKE all available rows.
   b. APPEND the sampled rows to the reduced dataset.

7. SAVE the reduced dataset as a new CSV file for further use.

8. PRINT summary statistics and distribution of categories in the reduced dataset.
```

Model Training and Evaluation

Purpose: Train and evaluate machine learning models for malware classification.

```
1. IMPORT required libraries:
   - pandas and numpy for data processing.
   - sklearn for machine learning algorithms and evaluation metrics.

2. LOAD the preprocessed dataset:
   a. READ dataset from CSV file.
   b. DISPLAY basic statistics (mean, standard deviation, etc.) and check for missing values.

3. HANDLE missing values:
   a. REPLACE missing values with mean/median for numerical columns.
   b. FILL missing values with mode or 'Unknown' for categorical columns.

4. ENCODE categorical variables using label encoding or one-hot encoding.

5. SPLIT dataset into training and testing sets:
   a. SET test size (e.g., 20%).
   b. RANDOMLY split the data into train and test subsets.

6. TRAIN multiple machine learning models:
   a. INITIALIZE Random Forest classifier.
   b. FIT the model using training data.
   c. TRAIN additional classifiers (e.g., Decision Trees, SVM) for comparison.

7. EVALUATE models:
   a. PREDICT labels for the test set.
   b. CALCULATE accuracy, precision, recall, F1-score, and confusion matrix.
   c. PLOT ROC curve and calculate the AUC score.

8. COMPARE performance metrics:
   a. SELECT the best model based on evaluation results.
   b. LOG results for each classifier in a table.

9. SAVE the trained model:
   a. EXPORT the best-performing model for deployment.
```

Signature-Based Detection

Purpose: Detect malware based on predefined signatures or patterns.

```
1. IMPORT required libraries:
   - pandas for data handling.
   - re (regular expressions) for pattern matching.

2. LOAD dataset:
   a. READ data from CSV or Excel file.
   b. DISPLAY the structure and check for any anomalies.

3. DEFINE signature patterns:
   a. SPECIFY patterns for each malware category (e.g., Trojan-like behavior).
   b. USE regular expressions to match strings representing malware behavior.

4. SCAN dataset for patterns:
   a. ITERATE through each row of the dataset.
   b. SEARCH each row for defined patterns using regex.
   c. FLAG rows where patterns are detected.

5. GENERATE detection results:
   a. COUNT the number of matches for each category.
   b. MARK rows as "Malicious" or "Benign" based on pattern detection.

6. EXPORT results:
   a. SAVE the flagged dataset to a new CSV file.
   b. PRINT summary statistics (e.g., number of flagged entries).
```

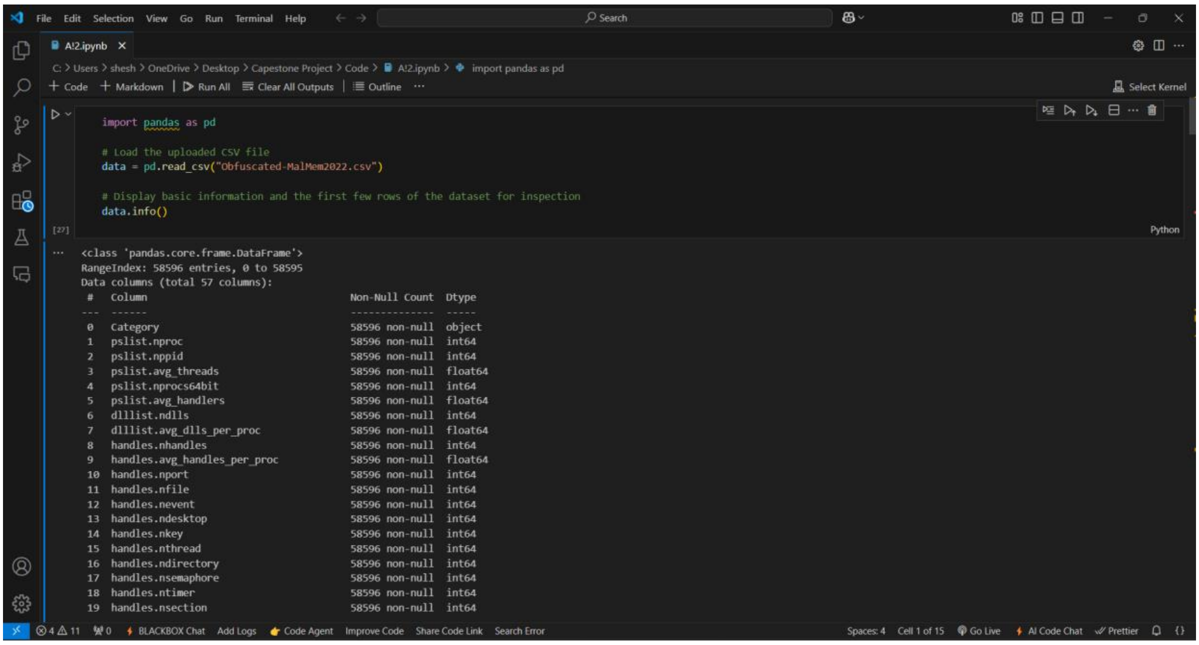
Data Visualization

Purpose: Visualize trends and patterns in the dataset to interpret results.

```
1. IMPORT required libraries:
   - pandas for data manipulation.
   - matplotlib and seaborn for visualization.

2. LOAD dataset:
   a. READ data from CSV or preprocessed output files.
   b. DISPLAY summary statistics and category distributions.

3. GENERATE visualizations:
   a. HISTOGRAMS:
      - PLOT frequency distributions of features (e.g., size, entropy).
   b. SCATTER PLOTS:
      - VISUALIZE relationships between key features (e.g., CPU usage vs Memory).
   c. BAR CHARTS:
      - SHOW distribution of malware categories and their counts.
   d. HEATMAPS:
      - CALCULATE correlation matrix.
      - PLOT heatmap to highlight feature correlations.

4. LABEL and format visualizations:
   a. ADD titles, axis labels, and legends for clarity.
   b. APPLY color schemes to highlight differences effectively.

5. EXPORT visualizations:
   a. SAVE plots as PNG or PDF files for reporting.
   b. INCLUDE visualizations in automated reports if needed.
```

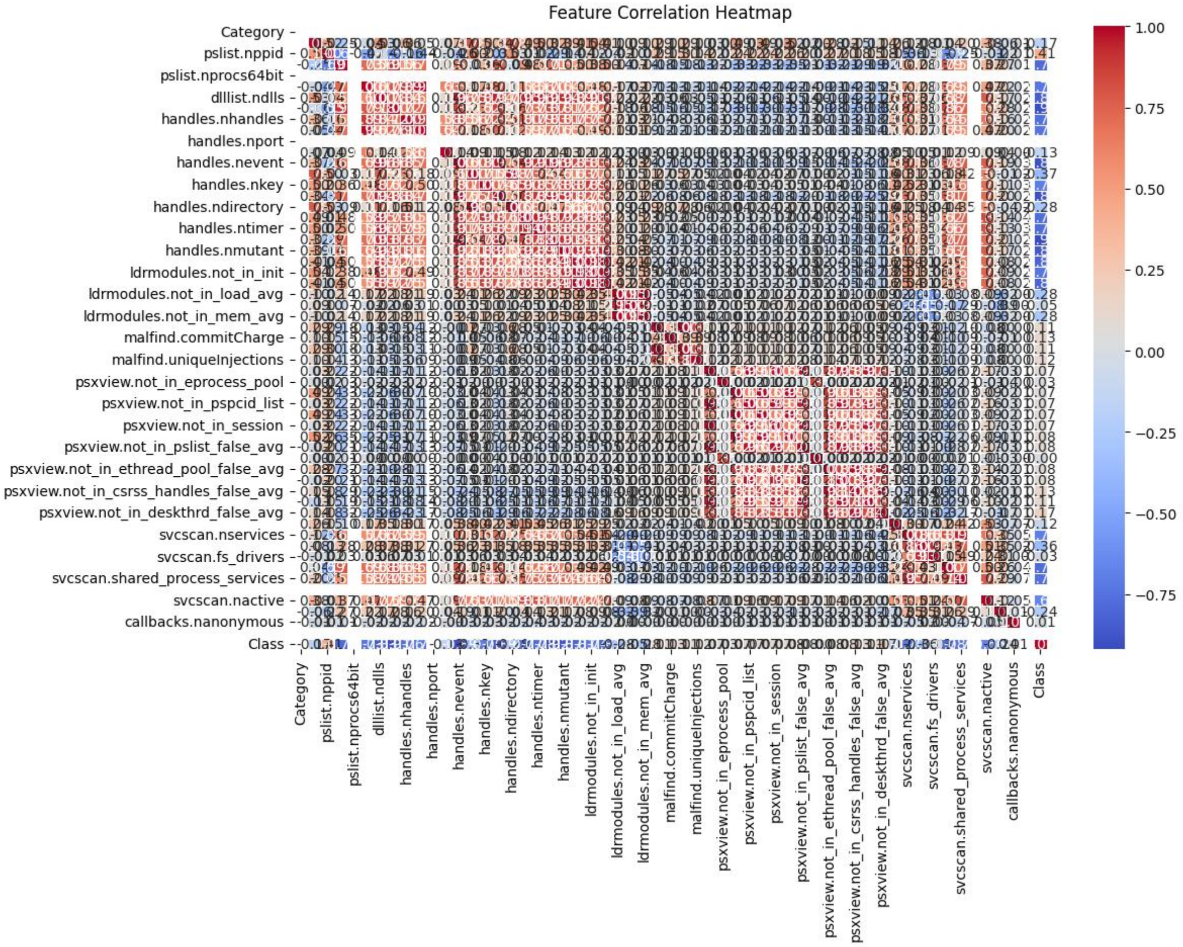
APPENDIX-B

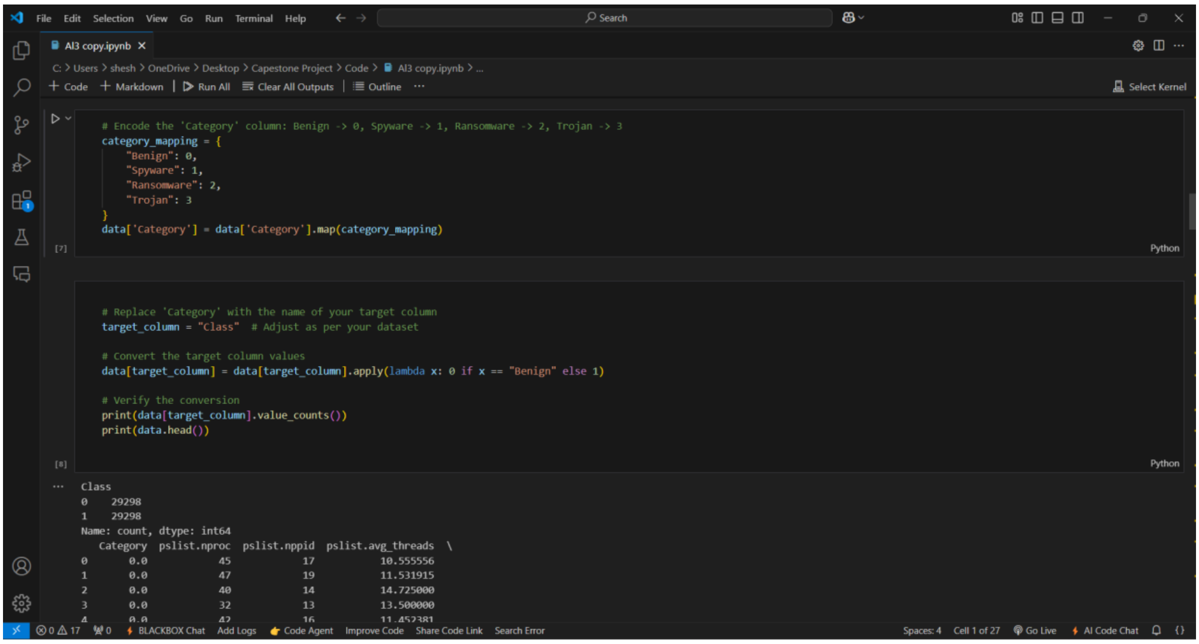
SCREENSHOTS


File Edit Selection View Go Run Terminal Help
Search
A!2.ipynb
C: > Users > shesh > OneDrive > Desktop > Capestone Project > Code > A!2.ipynb > import pandas as pd
+ Code + Markdown | Run All Clear All Outputs | Outline ...
Select Kernel
[27]
Python
⊗ 4 ⚠ 11 0 BLACKBOX Chat Add Logs Code Agent Improve Code Share Code Link Search Error
Spaces: 4 Cell 1 of 15 Go Live AI Code Chat Prettier

```
import pandas as pd

# Load the uploaded CSV file
data = pd.read_csv("Obfuscated-MalMem2022.csv")

# Display basic information and the first few rows of the dataset for inspection
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 58596 entries, 0 to 58595
Data columns (total 57 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0   Category                        58596 non-null  object
 1   pslist.nproc                    58596 non-null  int64
 2   pslist.nppid                    58596 non-null  int64
 3   pslist.avg_threads              58596 non-null  float64
 4   pslist.nprocs64bit              58596 non-null  int64
 5   pslist.avg_handlers             58596 non-null  float64
 6   dlllist.ndlls                   58596 non-null  int64
 7   dlllist.avg_dlls_per_proc       58596 non-null  float64
 8   handles.nhandles                58596 non-null  int64
 9   handles.avg_handles_per_proc    58596 non-null  float64
 10  handles.nport                   58596 non-null  int64
 11  handles.nfile                   58596 non-null  int64
 12  handles.nevent                  58596 non-null  int64
 13  handles.ndesktop                58596 non-null  int64
 14  handles.nkey                    58596 non-null  int64
 15  handles.nthread                 58596 non-null  int64
 16  handles.ndirectory              58596 non-null  int64
 17  handles.nsemaphore              58596 non-null  int64
 18  handles.ntimer                  58596 non-null  int64
 19  handles.nsection                58596 non-null  int64
```

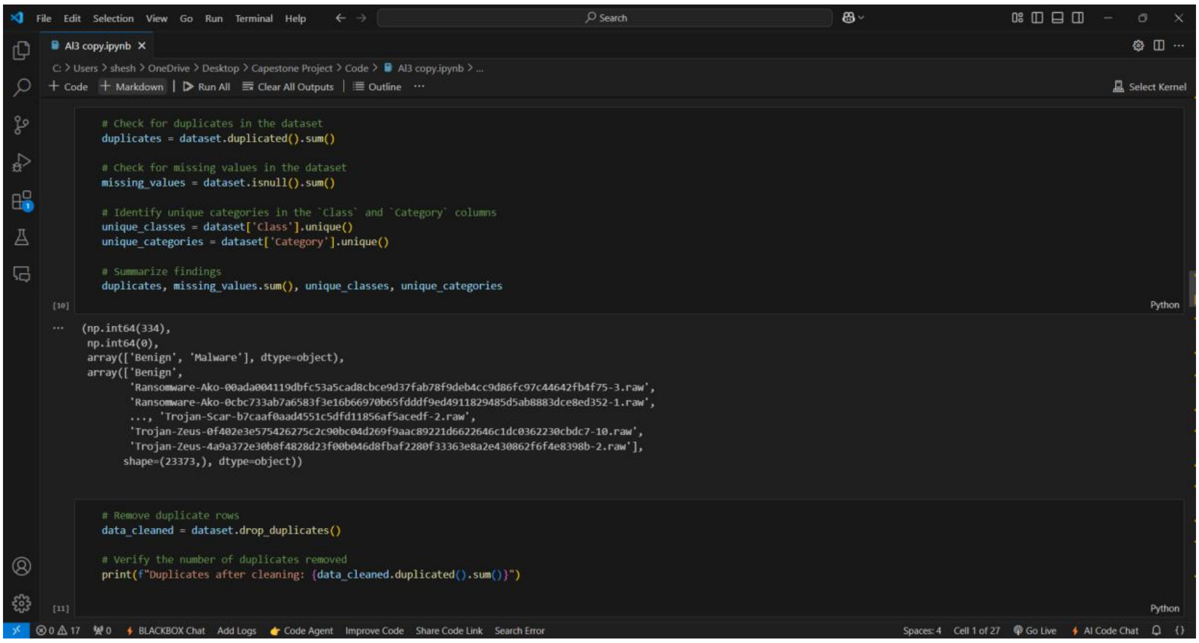




```
# Encode the 'Category' column: Benign -> 0, Spyware -> 1, Ransomware -> 2, Trojan -> 3
category_mapping = {
    "Benign": 0,
    "Spyware": 1,
    "Ransomware": 2,
    "Trojan": 3
}
data['Category'] = data['Category'].map(category_mapping)
```

```
# Replace 'Category' with the name of your target column
target_column = "Class"  # Adjust as per your dataset

# Convert the target column values
data[target_column] = data[target_column].apply(lambda x: 0 if x == "Benign" else 1)

# Verify the conversion
print(data[target_column].value_counts())
print(data.head())
```

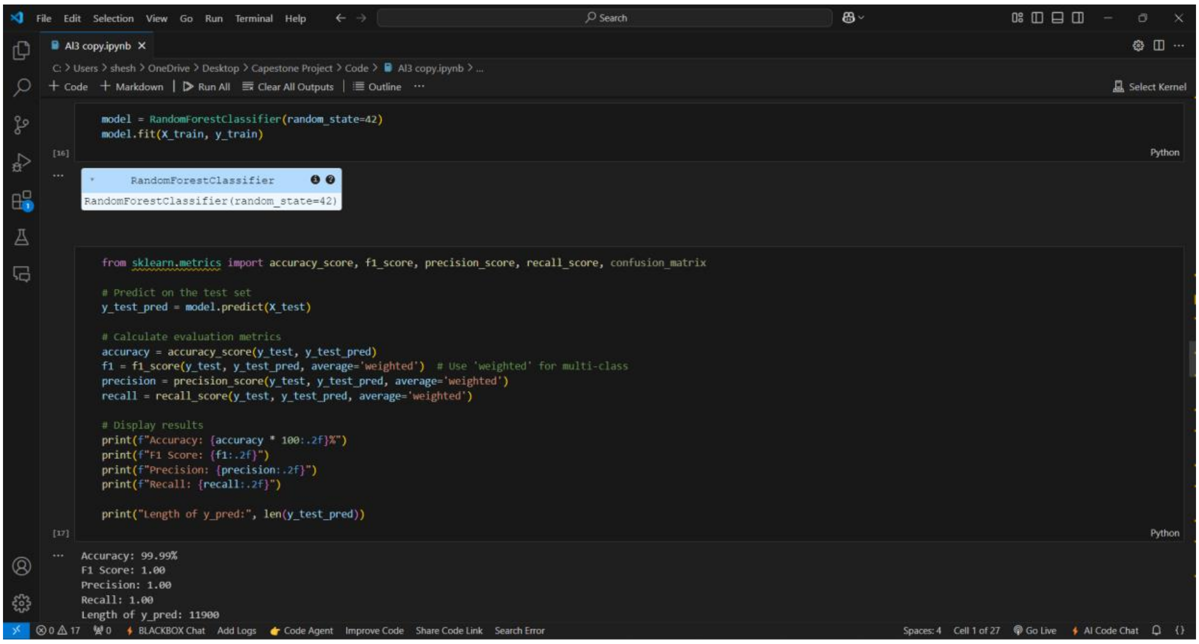
```
Class
0    29298
1    29298
Name: count, dtype: int64
   Category  pslist.nproc  pslist.nppid  pslist.avg_threads  \
0       0.0            45            17           10.555556
1       0.0            47            19           11.531915
2       0.0            40            14           14.725000
3       0.0            32            13           13.500000
4       0.0            42            16           11.452381
```



```
# Check for duplicates in the dataset
duplicates = dataset.duplicated().sum()

# Check for missing values in the dataset
missing_values = dataset.isnull().sum()

# Identify unique categories in the `Class` and `Category` columns
unique_classes = dataset['Class'].unique()
unique_categories = dataset['Category'].unique()

# Summarize findings
duplicates, missing_values.sum(), unique_classes, unique_categories
```

```
(np.int64(334),
 np.int64(0),
 array(['Benign', 'Malware'], dtype=object),
 array(['Benign',
        'Ransomware-Ako-00ada004119dbfc53a5cad8cbce9d37fab78f9deb4cc9d86fc97c44642fb4f75-3.raw',
        'Ransomware-Ako-0cbc733ab7a6583f3e16b66970b65fdddf9ed4911829485d5ab8883dce8ed352-1.raw',
        ..., 'Trojan-Scar-b7caaf0aad4551c5dfd11856af5acedf-2.raw',
        'Trojan-Zeus-0f402e3e575426275c2c90bc04d269f9aac89221d6622646c1dc0362230cbdc7-10.raw',
        'Trojan-Zeus-4a9a372e30b8f4828d23f00b046d8fbaf2280f33363e8a2e430862f6f4e8398b-2.raw'],
       shape=(23373,), dtype=object))
```

```
# Remove duplicate rows
data_cleaned = dataset.drop_duplicates()

# Verify the number of duplicates removed
print(f"Duplicates after cleaning: {data_cleaned.duplicated().sum()}")
```


File Edit Selection View Go Run Terminal Help
Search
AI3 copy.ipynb
C: > Users > shesh > OneDrive > Desktop > Capestone Project > Code > AI3 copy.ipynb > ...
+ Code + Markdown | Run All Clear All Outputs | Outline
Select Kernel
Python
BLACKBOX Chat Add Logs Code Agent Improve Code Share Code Link Search Error
Spaces: 4 Cell 1 of 27 Go Live AI Code Chat

```
model = RandomForestClassifier(random_state=42)
model.fit(X_train, y_train)
```

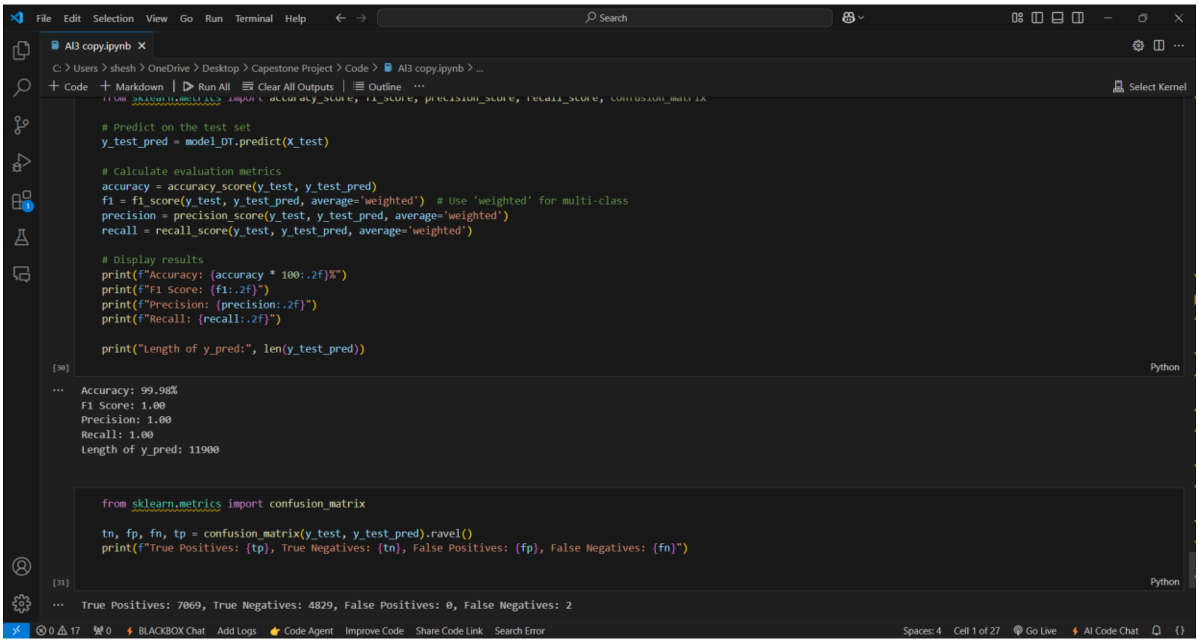
[16]

```
RandomForestClassifier
RandomForestClassifier(random_state=42)
```

```
from sklearn.metrics import accuracy_score, f1_score, precision_score, recall_score, confusion_matrix

# Predict on the test set
y_test_pred = model.predict(X_test)

# Calculate evaluation metrics
accuracy = accuracy_score(y_test, y_test_pred)
f1 = f1_score(y_test, y_test_pred, average='weighted')  # Use 'weighted' for multi-class
precision = precision_score(y_test, y_test_pred, average='weighted')
recall = recall_score(y_test, y_test_pred, average='weighted')

# Display results
print(f"Accuracy: {accuracy * 100:.2f}%")
print(f"F1 Score: {f1:.2f}")
print(f"Precision: {precision:.2f}")
print(f"Recall: {recall:.2f}")

print("Length of y_pred:", len(y_test_pred))
```

[17]

```
Accuracy: 99.99%
F1 Score: 1.00
Precision: 1.00
Recall: 1.00
Length of y_pred: 11900
```


File Edit Selection View Go Run Terminal Help
Search
AI3 copy.ipynb
C: > Users > shesh > OneDrive > Desktop > Capestone Project > Code > AI3 copy.ipynb > ...
+ Code + Markdown | Run All Clear All Outputs | Outline
Select Kernel
Python
BLACKBOX Chat Add Logs Code Agent Improve Code Share Code Link Search Error
Spaces: 4 Cell 1 of 27 Go Live AI Code Chat

```
from sklearn.metrics import accuracy_score, f1_score, precision_score, recall_score, confusion_matrix

# Predict on the test set
y_test_pred = model_DT.predict(X_test)

# Calculate evaluation metrics
accuracy = accuracy_score(y_test, y_test_pred)
f1 = f1_score(y_test, y_test_pred, average='weighted')  # Use 'weighted' for multi-class
precision = precision_score(y_test, y_test_pred, average='weighted')
recall = recall_score(y_test, y_test_pred, average='weighted')

# Display results
print(f"Accuracy: {accuracy * 100:.2f}%")
print(f"F1 Score: {f1:.2f}")
print(f"Precision: {precision:.2f}")
print(f"Recall: {recall:.2f}")

print("Length of y_pred:", len(y_test_pred))
```

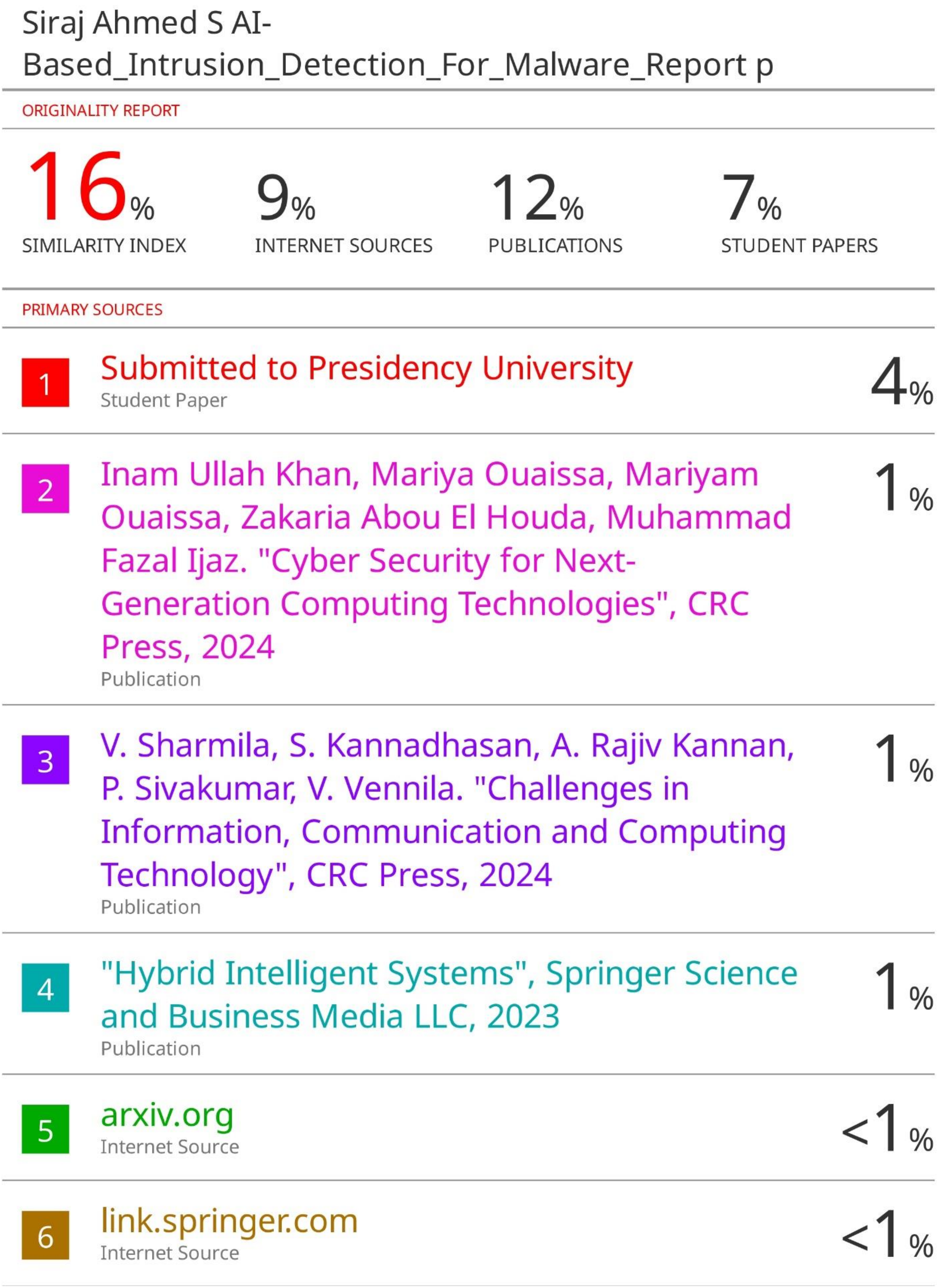
[30]

```
Accuracy: 99.98%
F1 Score: 1.00
Precision: 1.00
Recall: 1.00
Length of y_pred: 11900
```

```
from sklearn.metrics import confusion_matrix

tn, fp, fn, tp = confusion_matrix(y_test, y_test_pred).ravel()
print(f"True Positives: {tp}, True Negatives: {tn}, False Positives: {fp}, False Negatives: {fn}")
```

[31]

```
True Positives: 7069, True Negatives: 4829, False Positives: 0, False Negatives: 2
```



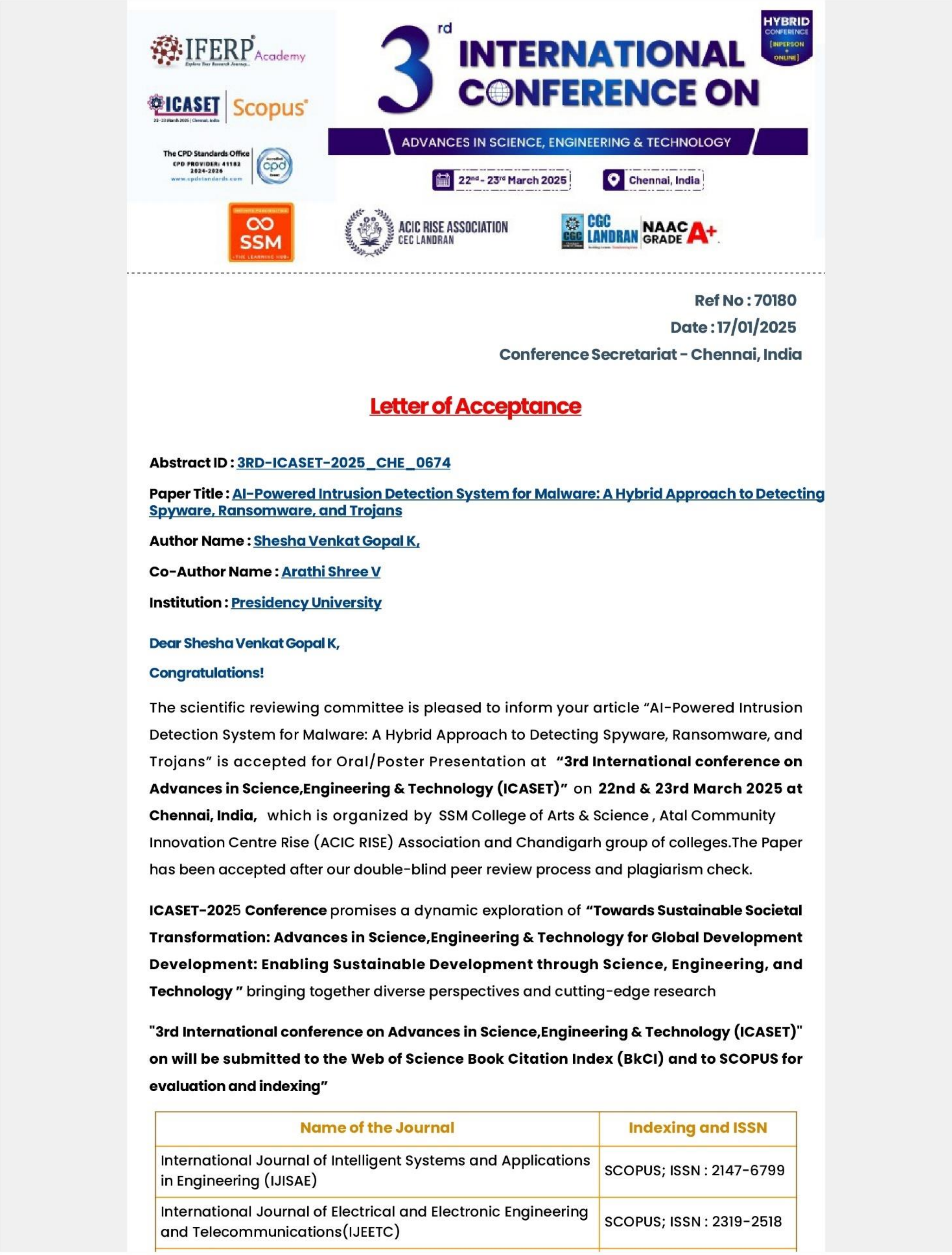
Siraj Ahmed S AI-Based_Intrusion_Detection_For_Malware_Report p

ORIGINALITY REPORT

16%	9%	12%	7%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Presidency University Student Paper	4%
2	Inam Ullah Khan, Mariya Ouaissa, Mariyam Ouaissa, Zakaria Abou El Houda, Muhammad Fazal Ijaz. "Cyber Security for Next-Generation Computing Technologies", CRC Press, 2024 Publication	1 %
3	V. Sharmila, S. Kannadhasan, A. Rajiv Kannan, P. Sivakumar, V. Vennila. "Challenges in Information, Communication and Computing Technology", CRC Press, 2024 Publication	1 %
4	"Hybrid Intelligent Systems", Springer Science and Business Media LLC, 2023 Publication	1 %
5	arxiv.org Internet Source	<1 %
6	link.springer.com Internet Source	<1 %



3rd INTERNATIONAL CONFERENCE ON

ADVANCES IN SCIENCE, ENGINEERING & TECHNOLOGY

HYBRID CONFERENCE [INPERSON + ONLINE]

22nd - 23rd March 2025 | Chennai, India

Ref No : 70180
Date : 17/01/2025
Conference Secretariat - Chennai, India

Letter of Acceptance

Abstract ID : 3RD-ICASET-2025_CHE_0674

Paper Title : AI-Powered Intrusion Detection System for Malware: A Hybrid Approach to Detecting Spyware, Ransomware, and Trojans

Author Name : Shesha Venkat Gopal K,

Co-Author Name : Arathi Shree V

Institution : Presidency University

Dear Shesha Venkat Gopal K,

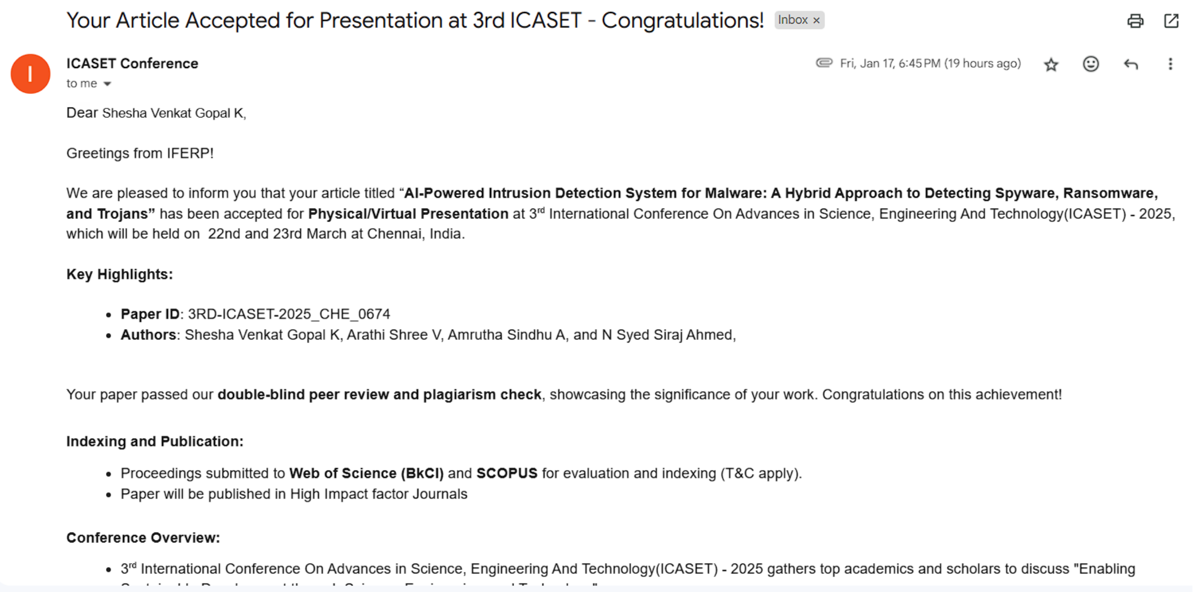
Congratulations!

The scientific reviewing committee is pleased to inform your article "AI-Powered Intrusion Detection System for Malware: A Hybrid Approach to Detecting Spyware, Ransomware, and Trojans" is accepted for Oral/Poster Presentation at **"3rd International conference on Advances in Science,Engineering & Technology (ICASET)"** on **22nd & 23rd March 2025 at Chennai, India,** which is organized by SSM College of Arts & Science , Atal Community Innovation Centre Rise (ACIC RISE) Association and Chandigarh group of colleges.The Paper has been accepted after our double-blind peer review process and plagiarism check.

ICASET-2025 **Conference** promises a dynamic exploration of **"Towards Sustainable Societal Transformation: Advances in Science,Engineering & Technology for Global Development Development: Enabling Sustainable Development through Science, Engineering, and Technology "** bringing together diverse perspectives and cutting-edge research

"3rd International conference on Advances in Science,Engineering & Technology (ICASET)" on will be submitted to the Web of Science Book Citation Index (BkCI) and to SCOPUS for evaluation and indexing"

Name of the Journal	Indexing and ISSN
International Journal of Intelligent Systems and Applications in Engineering (IJISAE)	SCOPUS; ISSN : 2147-6799
International Journal of Electrical and Electronic Engineering and Telecommunications(IJEETC)	SCOPUS; ISSN : 2319-2518



Your Article Accepted for Presentation at 3rd ICASET - Congratulations!

ICASET Conference

to me

Fri, Jan 17, 6:45 PM (19 hours ago)

Dear Shesha Venkat Gopal K,

Greetings from IFERP!

We are pleased to inform you that your article titled “**AI-Powered Intrusion Detection System for Malware: A Hybrid Approach to Detecting Spyware, Ransomware, and Trojans”** has been accepted for **Physical/Virtual Presentation** at 3rd International Conference On Advances in Science, Engineering And Technology(ICASET) - 2025, which will be held on 22nd and 23rd March at Chennai, India.

Key Highlights:

- **Paper ID**: 3RD-ICASET-2025_CHE_0674
- **Authors**: Shesha Venkat Gopal K, Arathi Shree V, Amrutha Sindhu A, and N Syed Siraj Ahmed,

Your paper passed our **double-blind peer review and plagiarism check**, showcasing the significance of your work. Congratulations on this achievement!

Indexing and Publication:

- Proceedings submitted to **Web of Science (BkCI)** and **SCOPUS** for evaluation and indexing (T&C apply).
- Paper will be published in High Impact factor Journals

Conference Overview:

- 3rd International Conference On Advances in Science, Engineering And Technology(ICASET) - 2025 gathers top academics and scholars to discuss "Enabling Sustainable Development through Science, Engineering and Technology."

MAPPING OF THE PROJECT WITH THE SUSTAINABLE DEVELOPMENT GOALS (SDGS).

SUSTAINABLE DEVELOPMENT GOALS
1 NO POVERTY
2 ZERO HUNGER
3 GOOD HEALTH AND WELL-BEING
4 QUALITY EDUCATION
5 GENDER EQUALITY
6 CLEAN WATER AND SANITATION
7 AFFORDABLE AND CLEAN ENERGY
8 DECENT WORK AND ECONOMIC GROWTH
9 INDUSTRY, INNOVATION AND INFRASTRUCTURE
10 REDUCED INEQUALITIES
11 SUSTAINABLE CITIES AND COMMUNITIES
12 RESPONSIBLE CONSUMPTION AND PRODUCTION
13 CLIMATE ACTION
14 LIFE BELOW WATER
15 LIFE ON LAND
16 PEACE, JUSTICE AND STRONG INSTITUTIONS
17 PARTNERSHIPS FOR THE GOALS

The project "AI-Powered Intrusion Detection System for Malware," can be mapped to relevant SDGs are:

1. **SDG-9: Industry, Innovation, and Infrastructure**
    - **The project promotes innovation through the development of AI-based security solutions, which contribute to building resilient infrastructure and fostering sustainable industrialization.**
2. **SDG-16: Peace, Justice, and Strong Institutions**
    - **By enhancing cybersecurity and protecting systems from malware, the project supports strong and secure institutions, contributing to justice and peace by safeguarding sensitive information and reducing vulnerabilities.**
3. **SDG-3: Good Health and Well-Being (Indirectly related)**
    - **While the project primarily focuses on cybersecurity, secure systems are crucial for protecting healthcare data and ensuring reliable digital health infrastructure, indirectly supporting this goal.**