iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 11 **Issue:** XII **Month of publication:** December 2023

DOI: https://doi.org/10.22214/ijraset.2023.57386

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue XII Dec 2023- Available at www.ijraset.com

Behaviour Based Malware Detection usingMachine Learning

V. Jyoshita Reddy[1], R. Jyoshna[2], B. Jyothi[3], Rithika. J. Poojari[4], N. Kartheek[5], B. Karthik Goud[6], R. Karthik[7]
[1, 2, 3, 4, 5, 6]*B.Tech, School of Engineering, Hyderabad, India*
[7]*Assistant Professor School of Engineering, Mallareddy University*

Abstract*: Considering all the researches done, it appears that over last decade, malware has been growing exponentially and also has been causing significant financial losses to different organizations. Thus, it becomes important to detect if a file contains any malware or not. The malwares can cause a lot of damage to the system such as slowing downthe system and also stealing sensitive information from the system. malware is a executable program specifically to destroy a genuine user's computer by spreading harmful virusin different ways .In the current times, one of the most important assets of the people is their data and information which needs to be protected. Hence, in order to protect the data and information, there is a need for software which couldperform this task and help in ensuring the integrity of our system. Our method for malware detection uses different machine learning algorithms such as decision tree, random forest etc. The algorithm which has the maximum accuracy gets selected which provides a great detection ratio for the system. Furthermore, the performance of the system is detected by calculating the false positive and false negative rates using the confusion matrix.

I. INTRODUCTION

Cyberattacks are currently the most pressing concern in the realm of modern technology. The word implies exploiting a system's flaws for malicious purposes, such as stealing from it,changing it, or destroying it. Malware is an example of cyberattack. Malware is any program or set of instructions that is designed to harm a computer, user, business, computer system. The problem to be examined involves the high spreading rate of computer malware (viruses, worms, Trojan horses, rootkits, botnets, backdoors, and other malicious software) and conventional signature matching-based antivirus systems fail to detect polymorphic and new, previously unseen malicious executables. Malware are spreading all over the worldthrough the Internet and are increasing day by day, thus becoming a serious threat. The manual heuristic inspection of static malware analysis is no longer considered effective and efficient compared against the high spreading rate of malware. Nevertheless, researches are trying to develop various alternative approaches in combating and detecting malware. One proposed approach (solution) is by using automatic dynamic (behavior) malware analysis combined with data mining tasks, such as, machine learning (classification) techniques to achieve effectiveness and efficiency in detecting malware. Rieck et al. [1] aim to exploit specific shared patterns for classification of malware. The authors said that variants of malware families share typical behavioral patterns reflecting its origin and purpose. Their method proceeds in three stages: (a) behavior of collected malware is monitored in a sandbox environment, (b) based on a corpus of malware labeled by an anti-virus scanner a malware behavior classifier is trained using learning techniques and (c) discriminative features of the behavior models are ranked for explanation of classification decisions. Rieck et al. [2] propose a framework for automatic analysis of malware behavior using machine learning. The framework allowsfor automatically identifying novel classes of malware with similar behavior (clustering) and assigning unknown malware to these discovered classes (classification). Christodorescu et al. [3] propose a technique by comparing the execution behavior of a known malware against the execution behaviors of a set of benign programs. The authors mine the malicious behavior present in a known malware that is not present in a set of benign programs. The output of the authors' algorithm can be used by malware detectors to detect malware variants.

II. PROBLEM STATEMENT

The rapid increase in complex and varied malware is seriously endangering the safety of computer systems and sensitive information. Traditional ways of identifying malware using fixedsignatures are struggling to keep up with these new and evolving threats. This creates a pressing need for smarter, more adaptable systems that can use machine learning to recognize and stop these threats effectively. These new systems must be accurate, able to handle different types of malware, and importantly, should avoid mistaking harmless files for threats (false positives) or missing actual threats (false negatives). Building a system that can handle these challenges while working quickly and effectively in real- time is a significant obstacle in keeping our systems safe from the ever-growing array of malware.

III. LITERATURE REVIEW

The proliferation of computers, smartphones, and other Internet- enabled gadgets leaves the world vulnerable to cyber assaults. A plethora of malware detection methods have arisen in response to the explosion in malware activity. When trying to identify malicious code, researchers use a variety of big data tools and machine learning techniques. Traditional machine learning-based malware detection approaches have a considerable processing time, but may effectively identify newly emerging malware. [19]

Armaan (2021) illustrated and tested the accuracy of various models. Without data, no application built for a digital platform can perform its function [4]. There are several cyberrisks, so it is essential that precautions be taken to safeguard data. Although feature selection is difficult when developing a model of any sort, machine learning is a cutting-edge approach that paves the way for precise prediction. The approach needs a workaround that is adaptable enough to handle non-standard data. To effectively manage and preventfuture assaults, we must analyse malware and create new rules and patterns in the form of creation of malware type [5]. To find patterns, IT security professionals may use malware analysis tools. The availability of technologies that analyse malware samples and determine their level of malignancy significantly benefit the cybersecurity sector. These tools helpmonitor security alerts and prevent malware attacks. If malware is dangerous, we must eliminate it before it transmits its infection any further. Malware analysis is becoming increasingly popular as it helps businesses lessen the effects of the growing number of malware threats and the increasing complexity of the ways malware can be used to attack [6].CAST-128, IDEA and RC2. Blowfish consumes the least power. Eventually, it turns out that Blowfish is the best, in terms of time, throughput and power. Saini [7] sums up that the superior algorithms are prominent for their popularity. An efficient cryptography achieves two parts of an equation, possibility and acceptability. Malicious programs and their threats, or “malware,” became increasingly common and sophisticated as the Internet developed. Their rapid dispersionover the Internet has provided malware authors with access toa wide variety of malware generation tools [8].Every day, the reach and sophistication of malware grows. This study focused on analysing and measuring classifier performance tobetter understand how machine learning works. Latent analysis extracted features from the recovered PE file and library information; six classifiers based on ML techniques were evaluated. It was recommended that ML systems be trained and tested to determine whether or not a file is harmful. Experimental outcomes verified that the random forest method is preferable for data categorization, with 99.4percent accuracy. These results showed that the PE library was compatible with static analysis and that focusing on onlya few properties could improve malware detection and characterization. The main benefit is that it is less likely that malicious software will be installed by accident, as users cancheck a file’s validity before opening it [9].

IV. METHODOLOGY

Like how big the file is or what type it is. Then, we dive deeper, peeking into the file's inner code to find any strange patterns or commands it might have. We also watch how the file behaves when we run it in a safe environment—sort of like a test zone— to see if it does anything suspicious.

This includes checking the kinds of calls it makes to other computer functions. Additionally, we look at extra details about the file, like when it was made or if it seems related to other files. Sometimes, we even measure how jumbled or compressed the information inside is because badsoftware often tries to hide itself that way. Once we gather all thisinformation, we pick out the most important parts and convert them into a format that computers can understand really well. This helps special computer programs learn to tell the differencebetween normal and harmful files.

Features Selection: When we're looking at lots of information tofigure out if something is good or bad—like checking if a file is safe or not for our computers—we sometimes have way too muchdata. Feature selection is a bit like picking the most important clues from a big detective case. Imagine having a thousand cluesabout a mystery, but not all of them are equally helpful. Some clues might be really similar or don't really help solve the case. So, feature selection is like sorting through these clues to choosethe ones that tell us the most important stuff without making things too complicated. We try to select the details that are the most helpful in making decisions.

Maybe the size of the file or aparticular way the file behaves is more important than other details. By picking out these important pieces of information, wemake it easier for our computer programs to learn and understandwhat's good and what's not, without getting confused by too manyunnecessary details. This way, we can figure out if a file is safe or if it might be hiding something harmful. The research methodology process will be explained in this section. The general overview of the research methodology isshown in Fig.4.1.

1) *Data set:* The data set consists of malware data set and benign instance data set. Both malware and benign instance data sets are in the format of Windows Portable Executable (PE) file binaries. Datasets of benign ad malware were collected from www.kaggle.com.

2) *Pre-Processing:* Data were stored in the file system as binary code, and the files themselves were unprocessed executables. We prepared them in advance of our research. Unpacking the executables required a protected environment, or virtual. machine (VM). PEiD software automated unpacking of compressed executables
3) *Features Extraction:* When we're trying to spot bad software (malware) on computers, we use something called feature extraction. This process involves looking at different parts of a file to figure out if it's safe or not. First, we check basic things

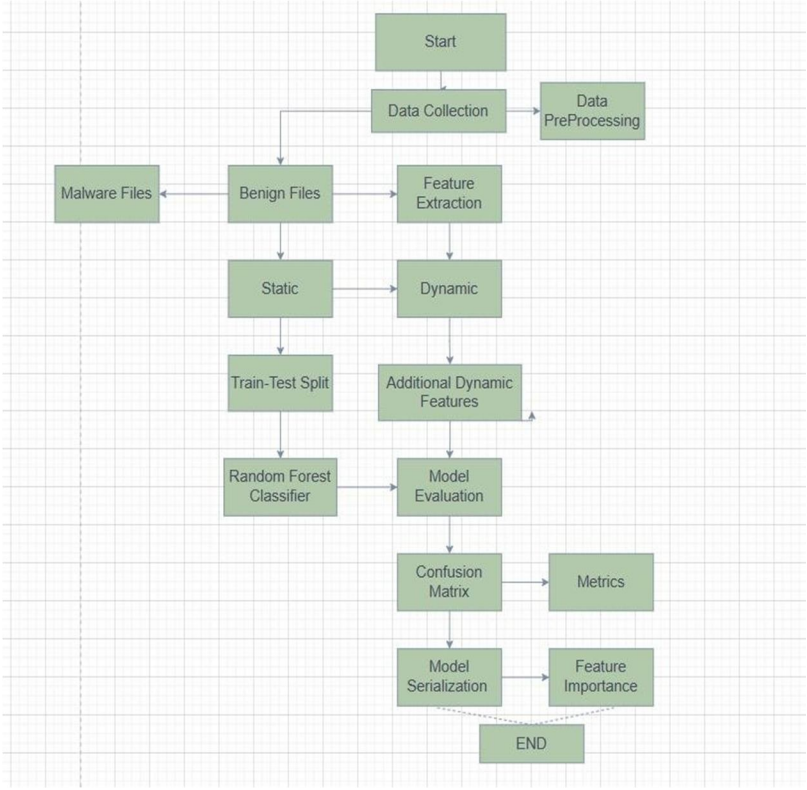


Fig 4.1 Architecture

V. EXPERIMENT RESULTS

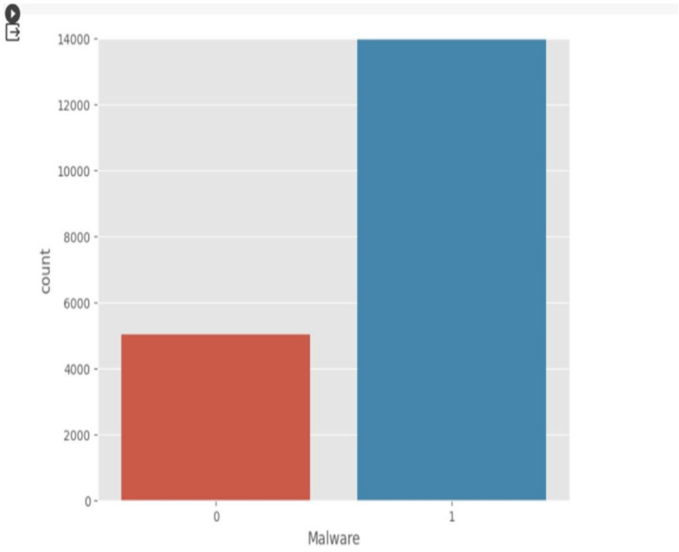


Fig 5.1 Classes Distribution

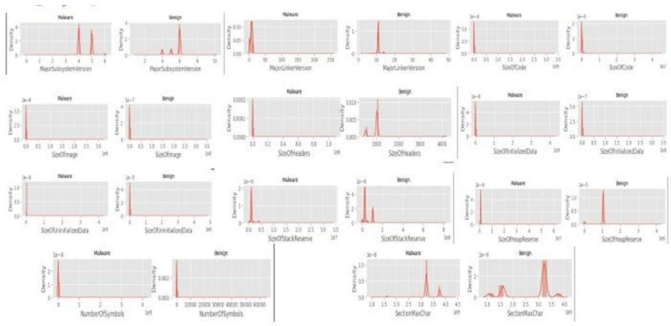


Fig 5.2 Features visualization

	precision	recall	f1-score	support
Benign	0.99	0.96	0.97	1004
Malware	0.99	1.00	0.99	2919
accuracy			0.99	3923
macro avg	0.99	0.98	0.98	3923
weighted avg	0.99	0.99	0.99	3923

Fig 5.3 Classification Report

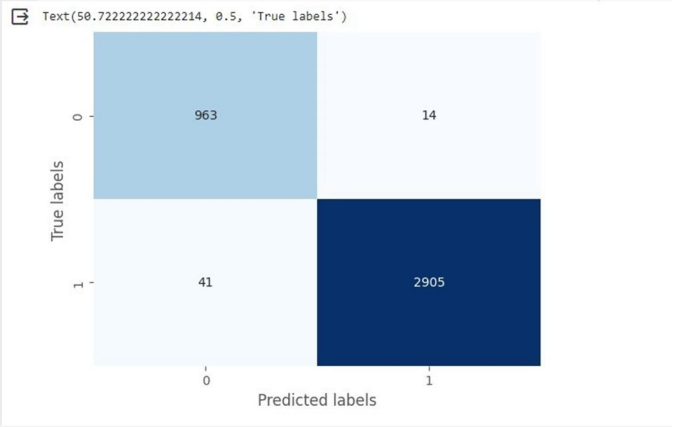


Fig 5.4 Confusion matrix

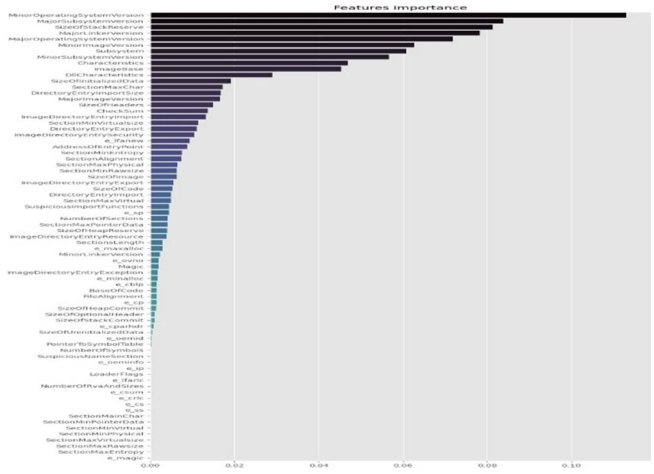


Fig 5.5 Features Importance

VI. CONCLUSION

In the realm of cybersecurity, the challenges posed by rapidlyevolving malware necessitate innovative solutions. This studyfocused on leveraging machine learning algorithms for malware detection, aiming to address the persistent threat landscape that traditional signature-based methods struggle toconfront. Through a comprehensive exploration of feature extraction, encompassing both static attributes and dynamic behaviors of files, this research showcased the significance of identifying and utilizing pertinent characteristics in distinguishing between benign software and malicious code. The examination of various machine learning techniques, including decision trees, random forests, and neural networks,underscored the potential of these algorithms in effectively identifying and classifying diverse malware types.

The process of feature selection emerged as a pivotal step, elucidating the importance of choosing discriminative features while reducing dimensionality, thereby enhancing the accuracy and efficiency of detection models. Moreover, the evaluation metrics, including accuracy, precision, recall, and false positive/negative rates obtained through theconfusion matrix, provided insights into the performance of the detection system. These metrics highlighted the trade-offs between correctly identifying threats and minimizing false alarms, contributing to the ongoing discourse on optimizing detection systems. In conclusion, while machine learning- based approaches exhibit promise in augmenting malware detection, there is a need for ongoing research and development. Future endeavors should focus on enhancing the robustness and adaptability of detection systems, exploring novel feature extraction techniques, and investigating the amalgamation of multiple algorithms for improved accuracy and resilience against sophisticated malware threats. Ultimately, the pursuit of more effective andproactive detection mechanisms remains essential in safeguarding systems and preserving data integrity in an ever-evolving cybersecurity landscape.

VII. FUTURE ENHANCEMENT

Looking ahead, the evolution of malware detection using machine learning presents several avenues for improvement. Adaptive systems capable of dynamically learning and adapting to new malware patterns in real-time stand as a critical future enhancement. These self-learning models could utilize reinforcement learning techniques, continuously interacting withevolving threats to refine their detection strategies. Furthermore, advancements in feature engineering are anticipated, especially in exploring deep learning-based approaches for automatic extraction of intricate patterns within files. Hybrid feature selection methods that merge human expertise with automated algorithms might also unlock more nuanced and relevant features. Additionally, the potential of ensemble models, combining multiple algorithms, or hybrid models merging rule- based and machine learning approaches, holds promise in enhancing accuracy and interpretability. Scalability remains a key focus, with strides expected in parallel computing for real- time analysis and lightweight models catering to resource- constrained environments like IoT devices. Future efforts may also emphasize adversarial robustness by fortifying models against evasion attempts and enhancing explainability for trust and comprehension. Collaborative initiatives promoting data sharing, standardized evaluation protocols, and industry- academia partnerships are pivotal for collective advancements in the field.

VIII. ACKNOWLEDGEMENT

I would like to express my heartfelt gratitude to my guide R.Karthik, and head of department, Dr .Thayyaba Khatoon, for their invaluable guidance and unwavering support throughout the development of this project. Their insightful feedback helped me to refine my ideasand develop a comprehensive understanding of the subject matter.

Their mentorship was instrumental in shaping my approach towards the project, and I am grateful for the knowledge and experience they shared with me. Without their encouragementandsupport, this project would not have been possible. Once again, I extend my sincere thanks to my guides and head of department fortheir unwavering support and guidance.

Our sincere thanks to all the teaching and non-teaching staff of Department of Computer Science and Engineering (AI&ML) for their support throughout our project work.

REFERENCES

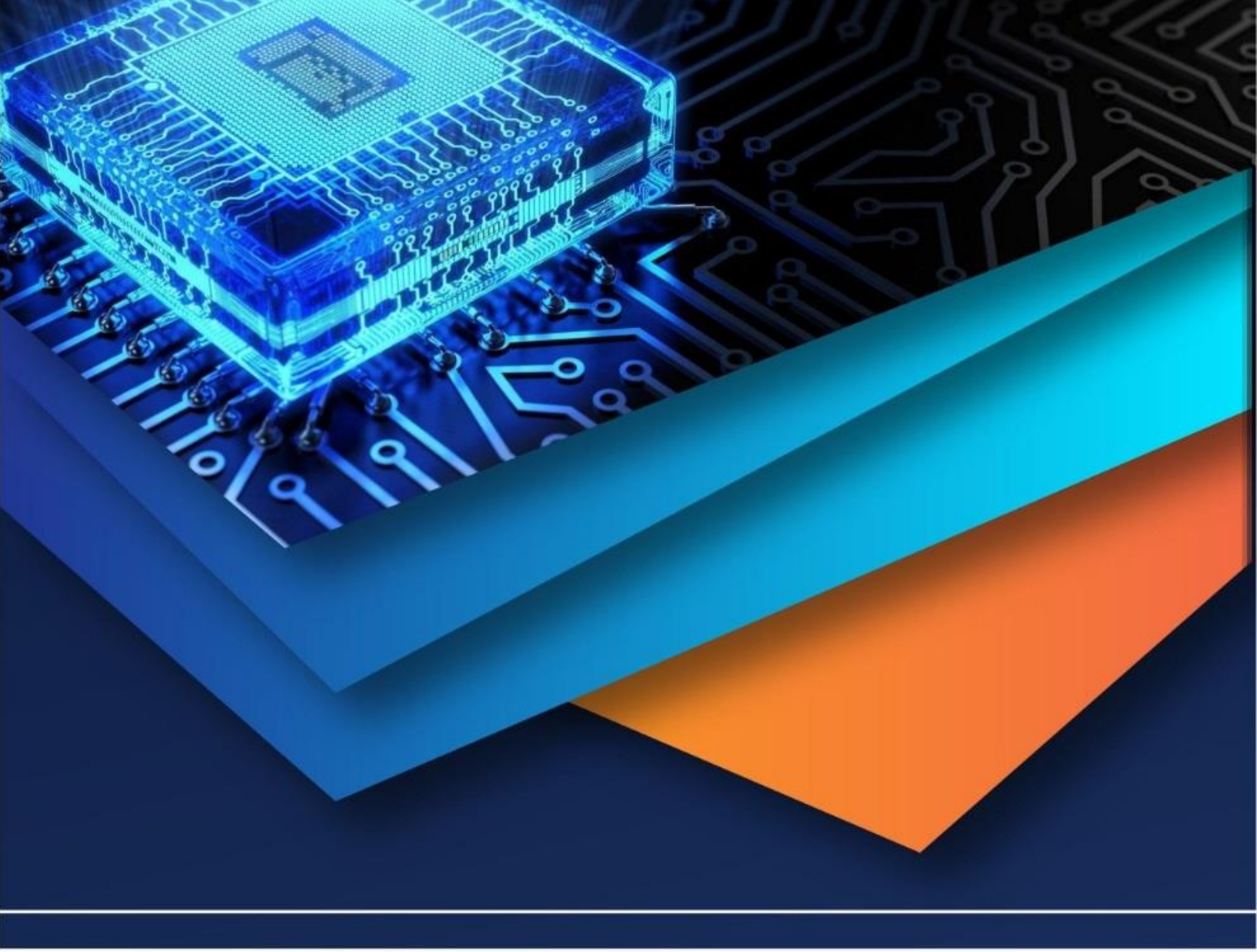
[1] Rieck, T. Holz, C. Willems, P. Duessel, and P. Laskov, K “Learning and Classification of Malware Behavior”, DIMVA, LNCS 5137, pp. 108–125, Berlin Heidelberg: Springer-Verlag, 2008.K. Rieck, P. Trinius, C. Willems, and T. Holz, “Automatic Analysis of Malware Behavior using Machine Learning”, 2009.

[2] M. Christodorescu, S. Jha, and C. Kruegel, “Mining Specifications of Malicious Behavior”, Proceedings of the 6th joint meeting of the ESEC and the ACM SIGSOFT Symposium on the FSE, September 3–7, Dubrovnik, Croatia, ACM,

[3] Tahtaci, B.; Canbay, B. Android Malware Detection Using Machine Learning. In Proceedings of the 2020 Innovations in Intelligent Systems and Applications Conference (ASYU), Istanbul, Turkey, 15–17 October 2020; pp. 1–6.

[4] Baset, M. Machine Learning for Malware Detection. Master’s Dissertation, Heriot Watt University, Edinburg,Scotland, December 2016.

[5] Akhtar, M.S.; Feng, T. Deep learning-based frameworkfor the detection of cyberattack using feature engineering. Secur.Commun.Netw. 2021, 2021, 6129210.
[6] Altaher, A. Classification of android malware applications using feature selection and classification algorithms. VAWKUM Trans.Comput. Sci. 2016, 10,1.
[7] Gavrilu ţ , D.; Cimpoesu, M.; Anton, D.; Ciortuz, L. Malware detection using machine learning. In Proceedingsof the 2009 International Multiconference on Computer Science and Information Technology, Mragowo, Poland, 12–14 October 2009; pp. 735–741.
[8] Pavithra, J.; Josephin, F.J.S. Analyzing various machine learning algorithms for the classification ofmalwares. IOP Conf. Ser.Mater. Sc



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY