International Journal of Electrical, Electronics and Computer Engineering **14**(1&2): 101-103(2025)

ISSN No. (Online): 2277-2626

Malware detection: Analysis and reduction of False Negatives and False Positives

Harsh Kaundal*, Mridul and Aditya
School of Computer Science Engineering and Technology,
Government College Dharamshala (H.P.), India.

(Corresponding author: Harsh Kaundal)*
(Received: 17 March 2025, Accepted: 26 April 2025)
(Published by Research Trend, Website: www.researchtrend.net)

ABSTRACT: **The detection of malicious software (malware) is critical to cyber security. Unfortunately, conventional approaches make errors. Occasionally, they miss detecting malware (false negatives), and infections result. At other times, they incorrectly mark harmless programs as malicious (false positives), creating unnecessary issues and wasting resources. In this study, we propose a novel method to minimize these errors. We employ a mix of machine learning and dynamic analysis methods to enhance precision. We apply our approach to a big dataset comprising malware and benign software. Our findings indicate a significant improvement, with 98.5% precision in identifying malware accurately. This method can enhance cyber security systems by minimizing errors and enhancing detection.**

Keywords: Malware detection, False negatives, False positives, Machine learning, Dynamic analysis, Hybrid analysis.

INTRODUCTION

Malware detection has become increasingly significant in cybersecurity due to the rapidly evolving malware threats. Malware can lead to economic losses, security incidents, and destruction of critical systems, impacting individuals, organizations, and society. Conventional techniques such as heuristic-based and signature-based detection are widely employed but frequently err, resulting in false positives (FPs) and false negatives (FNs). Over the past few years, the detection of malicious internet traffic has been a main issue and area of study in network security. This is because the internet has seen an enormous rise in malicious activities like hacking attacks (e.g., DoS attacks) and propagation of malicious programs like viruses, worms, trojans, spyware, and botnets. Malicious traffic degrades networks and inflicts annoyances on users. For example, Distributed Denial of Service (DDoS) attacks may delay Domain Name Service (DNS) responses by 230% and web sites by 30%. The notable example was during July–August 2001 when CodeRed worm infested around 3.95 lakh computers worldwide, causing damage of around $2.6 billion. There are various methods of detecting malicious traffic, but security appliances like firewalls are still prone to some weaknesses. It is due to this reason that Intrusion Detection Systems (IDSs) and Intrusion Prevention Systems (IPSs) are being used on a large scale nowadays. These help in detecting malicious traffic, monitoring network behavior, and protecting computer systems from major damage. Other security measures like firewalls, access control, encryption, and user awareness have not been able to stop cyber-attacks totally (Kolter and Maloof 2006). Once a malcode goes unnoticed, it might infect systems and do harm—a "false negative." In another case, a benign application, which is mistakenly picked up as being a virus, is termed as a "false positive." It can be inefficient, lead to system disruption, and reduce the users' confidence level. Since false negatives and false positives may do immense harm, better and accurate malware detection is needed. Recent advances in dynamic analysis and machine learning have made possible more accurate malware detection. Machine learning algorithms are able to detect patterns and anomalies in malware, and dynamic analysis gives complete information on how malware functions. But these techniques have also some challenges, and additional research is required to make more efficient malware detection systems. This study seeks to address the issue of false positives (FPs) and false negatives (FNs) in malware detection through a novel method that integrates machine learning and dynamic analysis. The objective is to create a more precise and efficient malware detection system that can minimize errors and enhance cybersecurity. Malware detection systems like antivirus, Intrusion Detection Systems (IDS), and Intrusion Prevention Systems (IPS) are designed to detect and block malicious threats. But they are far from perfect and can commit two types of mistakes: False Positives (FP) and False Negatives (FN). A false positive takes place when an innocent file or program is wrongly flagged as malware, resulting in unnecessary interruptions, blocked programs, or erased files. This can happen due to heuristic-based detection, signature

mismatches, tight security settings, or through the use of encryption techniques in legitimate software. High-profile cases include McAfee Antivirus falsely reporting the system file svchost.exe as essential in 2010 and Microsoft Security Essentials falsely identifying Google Chrome as a Trojan in 2011. False positives cause business disruption, undermine trust in security products, and create extra workloads for IT professionals as they find and fix the errors (Ye *et al.*, 2010). On the other hand, a false negative occurs when real malware is missed and not detected, allowing it to execute undetected and, in the process, cause immense harm. False negatives are more dangerous than false positives because they leave systems vulnerable to attacks. They often appear due to zero-day malware (novel attacks that have not yet been identified by antivirus databases), polymorphic malware (which changes its code constantly), fileless malware (which only operates in memory), and encrypted or obfuscated threats. For example, in 2017, a security solution failed to detect a new WannaCry ransomware variant, and a significant amount of harm was caused, while in 2020, Emotet malware bypassed many antivirus solutions using advanced obfuscation mechanisms. The consequences of false negatives include data breaches, financial loss, system corruption, and the spread of malware across networks, which necessitate the use of more advanced and accurate malware detection mechanisms.

RELATED WORK

Malware detection is a critical component of cyber security, and various approaches have been proposed to improve detection accuracy. This section reviews existing literature on malware detection, focusing on machine learning and dynamic analysis techniques. A crucial part of cybersecurity is malware detection, and several strategies have been put forth to increase detection precision. With an emphasis on machine learning and dynamic analysis methods, this section examines the body of research on malware detection. Machine learning algorithms are commonly employed in malware detection. To determine the distinction between malware and secure software, researchers have proposed the use of supervised learning algorithms such as random forests and support vector machines (SVMs) (Kolter and Maloof, 2006; Raman *et al.*, 2012). Researchers have also evaluated the application of unsupervised learning algorithms, including anomaly detection and clustering, to identify unknown malware (Ye *et al.*, 2010). Dynamic analysis refers to the study of the behaviour of software while in execution. For malware detection, researchers have proposed employing dynamic analysis techniques such as tracing system and API calls (Forrest *et al.*, 1996; Bayer *et al.*, 2009). In other research, researchers have employed sandboxing and virtualization to examine malware behaviour (Kirat *et al.*, 2014; Marignoni *et al.*, 2012). Numerous studies have put forth hybrid strategies that blend dynamic analysis and machine learning methods. The authors suggested a hybrid method that classifies system calls and detects malware using machine learning algorithms (Singh *et al.*, 2017). A different study suggested a hybrid strategy that classifies malware using machine learning algorithms and extracts features using dynamic analysis (Li *et al.*, 2018).

Existing methods have a number of drawbacks, despite their potential to increase malware detection accuracy. For instance, evasion attacks, in which malware is made to avoid detection, can affect machine learning algorithms (Kolter and Maloof, 2006). According to Forest et al. (1996), dynamic analysis techniques can be resource-intensive and might not work against unknown malware.

RESEARCH GAP

Even with improved malware detection, the problem of false positives (FPs) and false negatives (FNs) remains. A false negative (FN) happens when malware is incorrectly labelled as safe, while a false positive (FP) happens when secure software is incorrectly labelled as malware.

1. High False Negative Rates: Zero-day attacks are frequently difficult for current malware detection systems to identify, leading to high FN rates.

2. High False Positive Rates: Excessive FP rates brought on by overly general detection rules can result in needless alerts and system outages.

3. Lack of Explainability: It can be challenging to comprehend why a specific file was categorized as malicious or benign due to the opaque nature of many machines learning-based malware detection systems.

4. Evasion Techniques: To avoid detection, skilled attackers use evasion techniques like code obfuscation and anti-debugging.

5. Scalability Issues: Performance deterioration may result from existing detection systems becoming overloaded as the number of malware samples rises.

Research questions

1. How can we create malware detection systems that are more efficient while reducing FN and FP rates?
2. What characteristics and methods are available to enhance a malware detection system's explainability?
3. How can we create malware detection systems that are resistant to evasion and can change to meet new threats?
4. How can malware detection systems be made more scalable?

Potential research directions

1. Deep Learning-based Methods: Tell us how to increase the accuracy of malware detection by utilizing deep learning techniques like recurrent neural networks (RNNs) and convolutional neural networks (CNNs).
2. Explainable AI: Examine methods for elucidating the choices made by malware detection systems that rely on machine learning, such as model interpretability and feature importance.
3. Evasion-Resistant Techniques: Create innovative methods, like code obfuscation and anti-debugging, for identifying and countering evasion attempts.

4. Scalable Detection Systems: Create and assess malware detection systems that are scalable to manage high traffic and malware sample volumes

Implications:

The findings of this study have several implications for malware detection:

1. Improved Accuracy: The proposed hybrid approach achieves higher accuracy compared to existing machine learning algorithms and dynamic analysis techniques. This improvement in accuracy can help reduce the number of false negatives and false positives.

2. Reduced False Negatives: The hybrid approach reduces false negatives by detecting malware that may evade detection by traditional signature-based detection methods.

3. Reduced False Positives: The hybrid approach also reduces false positives by minimizing the number of benign software samples misclassified as malware.

FUTURE WORK AND SUGGESTION

Future studies can build upon this research by:

Investigating Other Machine Learning Methods: Researching deep learning methods and other machine learning algorithms to further enhance malware detection performance.

Adding Integration with Other Security Controls: Integrating this method with intrusion detection systems (IDS), firewalls, and other security controls to build an enhanced cybersecurity system.

CONCLUSIONs

In conclusion, the proposed hybrid approach demonstrates improved malware detection accuracy and reduced false negatives and false positives. The findings of this study have implications for organizations seeking to improve their malware detection capabilities. Future studies can build on the findings of this research to further improve malware detection accuracy.

REFERENCES

Bayer, U., Kruegel, C., & Kirda, E. (2009). Improving the efficiency of dynamic malware analysis. *Proceedings of the 2009 ACM Symposium on Applied Computing*, 1871–1878.

Forrest, S., Hofmeyr, S. A., Somayaji, A., & Longstaff, T. A. (1996). A sense of self for UNIX processes. *IEEE Symposium on Security and Privacy*, 120–128.

Kirat, D., Vigna, G., & Kruegel, C. (2014). BareCloud: Bare-metal analysis-based evasive malware detection. *23rd USENIX Security Symposium (USENIX Security 14)*, 287–301.

Kolter, J. Z., & Maloof, M. A. (2006). Learning to detect and classify malicious executables in the wild. *Journal of Machine Learning Research*, *7*(Dec), 2721–2744.

Li, Y., Xia, Y., Zhang, Y., & Jiang, T. (2018). Malware detection based on deep learning algorithm. *Neural Computing and Applications*, *29*(6), 1547–1556.

Marignoni, L., Zago, R., & Zanero, S. (2012). Towards a malware detection architecture based on distributed computing. *Journal of Computer Virology and Hacking Techniques*, *8*(3), 137–146.

Raman, K., Grizzard, J. B., & Owen, H. (2012). Malware analysis using visualization techniques. *Journal of Cyber Security and Information Systems*, *2*(1), 55–63.

Singh, S., Kumar, R., & Singla, R. K. (2017). Detection and classification of malware using machine learning. *International Journal of Security and Its Applications*, 11(5), 11–24.

Ye, Y., Li, T., Adjeroh, D., & Iyengar, S. S. (2010). A survey on malware detection using data mining techniques. *ACM Computing Surveys (CSUR)*, *48*(3), 1–36.