iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** III **Month of publication:** March 2026

DOI: https://doi.org/10.22214/ijraset.2026.78798

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 14 Issue III Mar 2026- Available at www.ijraset.com

Malware Behavior Classification Using XGBoost with MITRE ATT&CK Technique Mapping

Dr. J.S. Kanchana[1], Monesh B.[2], Amirthaganesh S.[3], Visish B.[4]
Computer Science Engineering (Cyber Security), K.L.N College of Engineering, Madurai, Tamil Nadu, India

Abstract: ***This project proposes a web-based automated malware analysis system designed to identify adversarial behaviors using the MITRE ATT&CK framework. The system allows users to upload suspicious malware samples through a secure web interface, which are then executed in an isolated and sandboxed Windows environment to prevent host compromise. During execution, Windows event logs and enhanced telemetry data are collected to capture detailed runtime behavior, including process creation, command execution, file and registry modifications, and network activity. These logs are processed and transformed into structured behavioral features that represent the actions performed by the malware. A machine learning–based multi-label classification approach using a binary relevance model is employed to map the extracted features to corresponding MITRE ATT&CK techniques. Independent Extreme Gradient Boosting (XGBoost) models are trained to detect the presence of individual attack techniques, enabling the identification of multiple techniques from a single malware execution. In addition, the system extracts relevant Indicators of Compromise (IOCs) such as malicious file paths, process names, and network endpoints. The proposed framework enables automated, explainable malware behavior classification and provides a systematic method for classifying malware activities based on standardized adversary techniques.***

Keywords: Malware Analysis, XG Boost, Cybersecurity, Sandbox, Machine Learning, MITRE ATT&CK

I. INTRODUCTION

Malware continues to evolve in both complexity and sophistication, making its detection and analysis increasingly challenging for security professionals. Traditional signature-based methods often fail to keep pace with modern threats, especially those designed to evade detection through obfuscation or dynamic behavior. As a result, there is a growing need for intelligent, behavior-based systems that can analyze and interpret how malware operates in real-world environments rather than relying solely on known patterns. In this context, this research introduces a web-based automated malware analysis system that focuses on understanding malicious behavior through the lens of the MITRE ATT&CK framework—a widely accepted standard for categorizing adversarial techniques. The system is designed to provide a safe and user-friendly interface where analysts can upload potentially malicious files without risking their host systems. Once submitted, these samples are executed within a carefully controlled and isolated Windows sandbox environment, ensuring that any harmful activity remains contained. During execution, the system captures a wide range of runtime data, including process creation events, command-line activity, file system interactions, registry changes, and network communications.

This rich set of telemetry data provides deep visibility into how the malware behaves when it is actively running. Rather than presenting raw logs, the system processes and transforms this information into structured behavioral features that accurately represent the actions performed by the malware. To make sense of these behaviors, the system leverages machine learning techniques, specifically a multi-label classification approach. Since a single malware sample can exhibit multiple attack techniques, a binary relevance strategy is used, where independent models are trained to detect individual behaviors. Extreme Gradient Boosting (XGBoost) is employed for its efficiency and strong performance in handling complex datasets. This approach allows the system to map observed behaviors to multiple MITRE ATT&CK techniques simultaneously, offering a more comprehensive understanding of the threat. In addition to behavioral classification, the system also extracts key Indicators of Compromise (IOCs), such as suspicious file paths, process names, and network endpoints. These indicators can be used by security teams for further investigation, threat hunting, and incident response.

Overall, the proposed framework aims to bridge the gap between raw malware execution data and meaningful threat intelligence. By combining automated analysis, machine learning, and standardized threat classification, it provides a scalable and interpretable solution for understanding modern malware behavior in a structured and actionable way.

II. LITERATURE REVIEW

[1] G. Önal and M. Güven, in *"Enhancing Dynamic Malware Behavior Analysis Through Novel Windows Events With Machine Learning"* (2025), propose an improved dynamic analysis approach by incorporating newly identified Windows event logs into malware behavior detection. Their work demonstrates that enriching telemetry sources can significantly enhance the visibility of malicious activities and improve classification accuracy when combined with machine learning techniques. However, the study primarily focuses on feature enhancement and does not extensively address large-scale deployment or real-time analysis challenges.

[2] R. Elnaggar, L. Servadei, S. Mathur, R. Wille, W. Ecker, and K. Chakrabarty, in *"Accurate and Robust Malware Detection: Running XGBoost on Runtime Data From Performance Counters"* (2022), present a method that leverages hardware-level performance counters as input features for malware detection using XGBoost. This approach achieves high accuracy and robustness by utilizing low-level execution characteristics that are difficult for malware to evade. However, the dependency on specialized hardware data sources may limit its applicability across diverse environments.

[3] Z. Zhang, P. Qi, and W. Wang, in *"Dynamic Malware Analysis With Feature Engineering and Feature Learning"* (2020), explore a hybrid approach combining manual feature engineering with automated feature learning techniques. Their study highlights the importance of capturing meaningful behavioral patterns from runtime data to improve detection performance. While effective, the approach introduces additional complexity in balancing handcrafted and learned features, which may impact scalability.

[4] A. Pektaş and T. Acarman, in *"Classification of Malware Families Based on Runtime Behaviors"* (2017), investigate the use of behavioral features extracted during execution to classify malware into different families. Their results show that runtime characteristics can effectively distinguish between malware types. However, the study focuses on family-level classification and does not address fine-grained mapping to standardized attack techniques or multi-label scenarios.

[5] K. Berlin, D. Slater, and J. Saxe, in *"Malicious Behavior Detection Using Windows Audit Logs"* (2015), present an approach that utilizes Windows audit logs as a primary data source for detecting malicious activities. Their work demonstrates the practicality of leveraging system-level logs for behavior-based detection in enterprise environments. However, the reliance on traditional audit logs may limit the depth of behavioral insight compared to more advanced telemetry sources.

III. PROPOSED METHODOLOGY

A. Sandbox-Based Execution

Suspicious samples are executed within an isolated Windows sandbox environment to ensure safe observation of real-world malware behavior without risking host system compromise.

B. Behavioral Data Collection

Detailed runtime activities are captured using Windows event logs, including process execution, persistence techniques, file system changes, registry modifications, and network interactions.

C. Feature Engineering

Collected behavioral data is transformed into structured features, enabling systematic analysis of malware actions beyond traditional signature-based approaches.

D. Machine Learning–Based Classification

A multi-label classification approach is applied to map observed behaviors to techniques defined in the MITRE ATT&CK framework, improving the accuracy and interpretability of threat detection.

E. IOC Extraction

Relevant Indicators of Compromise (IOCs), such as suspicious file paths, process names, and network endpoints, are identified to support further investigation and incident response.

IV. SYSTEM ARCHITECTURE

A. Malicious File Preprocessing

This module handles the secure intake of malware samples through a web-based interface. Uploaded files are validated based on type and size to prevent accidental host-level risks. The samples are stored in isolated directories with restricted permissions, ensuring safe handling. Secure transfer mechanisms are used to move the files into the analysis environment without exposing the host system.

B. Malware Execution

In this module, malware samples are executed within an isolated Windows sandbox environment. Virtualization techniques ensure complete separation from the host system, preventing compromise. Execution time is controlled to limit resource usage and potential damage. After execution, the sandbox is restored to its original clean state using snapshot mechanisms, ensuring consistency for subsequent analyses.

C. Windows Event Log Collection

This module is responsible for collecting detailed runtime data during malware execution. Windows event logs are continuously monitored to capture system-level activities, including process creation, command execution, service operations, and network communication. All collected events are correlated with the executing malware sample to ensure accurate behavioral tracking.

D. Feature Extraction

The collected logs are processed to extract meaningful behavioral features. Key attributes such as Event IDs, process names, and activity patterns are identified and transformed into structured numerical and categorical data. Irrelevant or redundant information is removed to improve efficiency, resulting in a refined dataset that accurately represents malware behavior.

E. Mitre ATT&CK Technique Mapping

This module maps the extracted behavioral features to standardized adversarial techniques. Machine learning models analyze the patterns and associate them with corresponding MITRE ATT&CK techniques. The mapping provides a structured understanding of attack behavior and highlights important features, enabling explainable threat classification.

F. XGBoost-Based Classification

The classification module utilizes the XGBoost algorithm to learn patterns from malware behavior. It builds decision trees sequentially to improve prediction accuracy and applies regularization techniques to prevent overfitting. The model generates probability scores for each predicted technique, enabling confident multi-label classification and reducing false positives.

G. IOC Extraction And Output

This module identifies and extracts relevant Indicators of Compromise (IOCs) such as malicious file paths, process names, and network endpoints. The results are presented in a structured format, supporting further investigation, threat hunting, and incident response.

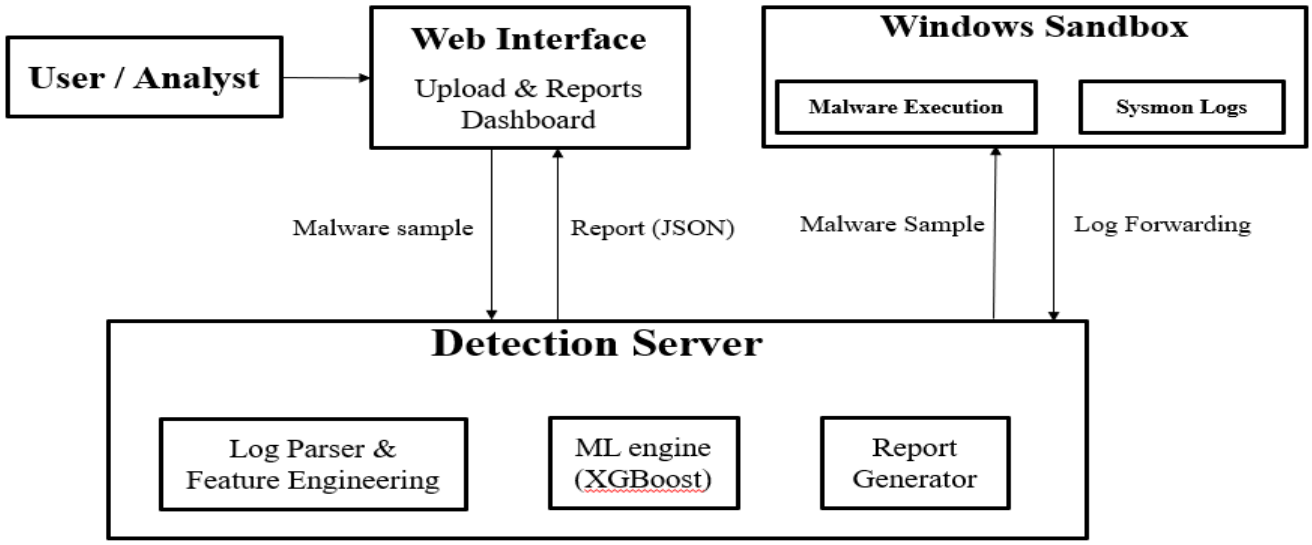


Fig. 1. Architecture Diagram

V. PERFORMANCE EVALUATION

This module evaluates the effectiveness of the proposed system by measuring detection accuracy and classification performance of malware behaviors mapped to MITRE ATT&CK techniques. It analyzes how accurately the system identifies malicious events from Windows Sysmon logs and classifies them into relevant attack techniques.

Accuracy = (True Positives + True Negatives) / (True Positives + True Negatives + False Positives + False Negatives)

Where:

True Positives represent correctly detected attack events and False Negatives represent malicious events missed by the system.

The system performance is further analyzed using detection metrics such as:

F1 Score = 2 × (Recall × True Positives / (True Positives + False Positives)) / (Recall + True Positives / (True Positives + False Positives))

Where:

The F1 Score provides a balanced evaluation of the detection performance by considering both missed detections and incorrect classifications, making it suitable for imbalanced malware event datasets.

TABLE I
PERFORMANCE COMPARISON

Metric	Logistic Regression	XGBoost
Accuracy	0.907	0.956
F1-Score	0.767	0.889

Fig. 2. Performance Comparison

VI. SYSTEM ANALYSIS

The proposed malware analysis and detection system is evaluated based on its capability to automatically analyze suspicious files, extract behavioral patterns, and accurately map them to adversarial techniques. The system integrates a web-based interface, a Windows sandbox environment, and a detection engine powered by machine learning models to ensure an end-to-end automated workflow.

Malware samples uploaded by users are executed in an isolated sandbox, where system activities are monitored and recorded through Windows Event Logs. These logs are processed to extract meaningful behavioral features such as process creation, file modifications, and network activities. The extracted features are then analyzed using trained models to classify potential threats and map them to corresponding MITRE ATT&CK techniques. This approach reduces the need for manual log inspection and improves the consistency of threat detection. The system enhances operational efficiency by automating critical steps including malware execution, log collection, feature extraction, and attack classification. It enables faster incident understanding by providing clear insights into attacker behavior and possible Indicators of Compromise (IOCs). Additionally, the modular architecture allows easy integration of new detection models and techniques, ensuring adaptability to evolving threats. Performance evaluation of the system is conducted using detection metrics such as recall to measure the system's ability to identify malicious activities. The system also considers incident response time by measuring the delay between malware execution and detection output. Overall, the architecture ensures scalability, flexibility, and efficient resource utilization, making it suitable for real-time malware analysis and cybersecurity research environments.

VII. EXPECTED RESULTS

The proposed malware analysis and detection system is designed to improve the efficiency of analyzing and identifying malicious files by integrating a web interface, Windows sandbox, and automated detection engine. The system processes malware samples submitted by users, where behavioral logs are generated during controlled execution in the sandbox environment. The generated logs are analyzed to extract relevant features, which are then used to detect malicious activities and map them to known attack techniques. The analysis results are stored and presented for further investigation. The generated outputs are visualized through the web interface, enabling clear understanding of malware behavior and supporting effective security analysis, as shown below.

VIII. CONCLUSION

The developed system provides an efficient method for analyzing malware using sandbox execution and behavior-based detection. It reduces manual effort and enables faster identification of malicious activities through automated log analysis and machine learning models. The system also helps in understanding attacker techniques by mapping behaviors to known attack patterns. Future improvements can focus on enhancing detection accuracy and supporting more advanced and evolving threats.

REFERENCES

[1] G. Önal and M. Güven, "Enhancing Dynamic Malware Behavior Analysis Through Novel Windows Events With Machine Learning," in IEEE Access, vol. 13, pp. 153937-153958, 2025, doi: 10.1109/ACCESS.2025.3604979.

[2] R. Elnaggar, L. Servadei, S. Mathur, R. Wille, W. Ecker, and K. Chakrabarty, ''Accurate and robust malware detection: Running XGBoost on runtime data from performance counters,'' IEEE Trans. Comput.-Aided Design Integr. Circuits Syst., vol. 41, no. 7, pp. 2066–2079, Jul. 2022, doi: 10.1109/TCAD.2021.3102007.

[3] Z. Zhang, P. Qi, and W. Wang, ''Dynamic malware analysis with feature engineering and feature learning,'' in Proc. AAAI Conf. Artif. Intell., Apr. 2020, vol. 34, no. 1, pp. 1210–1217, doi: 10.1609/aaai.v34i01.5474.

[4] A. Pektaş and T. Acarman, ''Classification of malware families based on runtime behaviors,'' J. Inf. Secur. Appl., vol. 37, pp. 91–100, Dec. 2017, doi: 10.1016/j.jisa.2017.10.005.

[5] K. Berlin, D. Slater, and J. Saxe, ''Malicious behavior detection using windows audit logs,'' in Proc. 8th ACM Workshop Artif. Intell. Secur., Oct. 2015, pp. 35–44, doi: 10.1145/2808769.2808773.

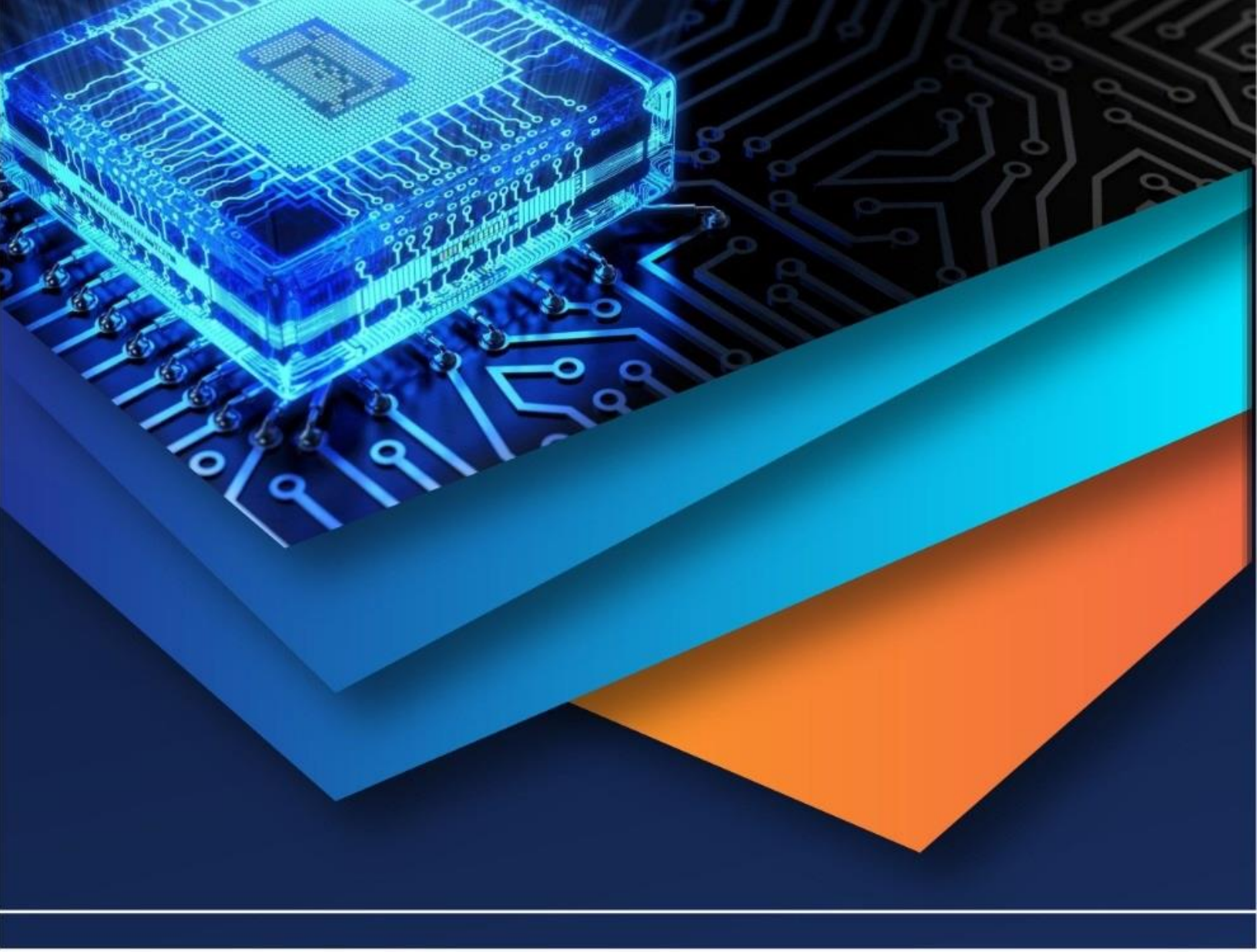
[6] R. M. S. Priya, P. K. R. Maddikunta, P. M., S. Koppu, T. R. Gadekallu, C. L. Chowdhary, and M. Alazab, ''An effective feature engineering for DNN using hybrid PCA-GWO for intrusion detection in IoMT architecture,'' Comput. Commun., vol. 160, pp. 139–149, Jul. 2020, doi: 10.1016/j.comcom.2020.05.048.

[7] M. A. Hossain, M. A. Haque, S. Ahmad, H. A. M. Abdeljaber, A. E. M. Eljialy, A. Alanazi, D. Sonal, K. Chaudhary, and J. Nazeer, ''AIenabled approach for enhancing obfuscated malware detection: A hybrid ensemble learning with combined feature selection techniques,'' Int. J. Syst. Assurance Eng. Manage., Mar. 2024, doi: 10.1007/s13198-024- 02294-y.

[8] P. S. Nguyen, T. N. Huy, T. A. Tuan, P. D. Trung, and H. V. Long, ''Hybrid feature extraction and integrated deep learning for cloud-based malware detection,'' Comput. Secur., vol. 150, Mar. 2025, Art. no. 104233, doi: 10.1016/j.cose.2024.104233.

[9] J. Busch, A. Kocheturov, V. Tresp, and T. Seidl, ''NF-GNN: Network flow graph neural networks for malware detection and classification,'' in Proc. 33rd Int. Conf. Sci. Stat. Database Manage., New York, NY, USA, Jul. 2021, pp. 121–132, doi: 10.1145/3468791.3468814.

[10] E. G. Onyedinma, A. Doris C, and I. E. Onyenwe, ''Towards resilient malware detection:Ahybrid framework leveraging static-dynamic features and ensemble models,'' World J. Adv. Eng. Technol. Sci., vol. 15, no. 3, pp. 634–639, Jun. 2025, doi: 10.30574/wjaets.2025.15.3.0901.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY