iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 **Issue:** VII **Month of publication:** July 2025

DOI: https://doi.org/10.22214/ijraset.2025.73342

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 13 Issue VII July 2025- Available at www.ijraset.com

An Extensive Survey on Malware Detection and Prevention Mechanisms Using Advanced Technologies

Vishwanath Chiniwar[1], Prerana Joshi[2], Mr. Pavan Mitragotri[3]
[1, 2]*MCA Student, KLS Gogte Institute of Technology, Belagavi, Karnataka, Affiliated to Visvesvaraya Technological University, Belagavi*
[3]*Assistant Professor, Department of MCA, KLS Gogte Institute of Technology, Belagavi, Karnataka, Affiliated to Visvesvaraya Technological University, Belagavi*

Abstract: ***The exponential growth in malware attacks, especially ransomware, is a critical challenge to digital infrastructure and global cybersecurity. Conventional signature-based detection techniques are less effective against sophisticated polymorphic and metamorphic malware. This paper offers a comprehensive survey of malware detection methods, with emphasis on Behavioral signature-based detection. It examines state-of-the-art technologies, such as dynamic analysis, machine learning, deep learning, and generative adversarial networks (GANs), and compares them in terms of their efficacy in detecting malware based on behavior patterns instead of static code. It draws from 46 peer-reviewed papers and emphasizes key findings, detection architectures, and innovations like RansomNet, DeepCodeLock, and PlausMal-GAN. The review ends by summarizing current limitations, issues such as Behavioral drift, and directions for the future in hybrid and intelligent malware detection systems.***
Keywords: Malware Detection, Behavioral Signature, Ransomware, Machine Learning, Dynamic Analysis, GAN, Intrusion Detection, Cybersecurity.

I. INTRODUCTION

Cybersecurity threats are escalating both in frequency and sophistication, with malware being one of the most persistent challenges. Initially limited to viruses and worms, malware has evolved into advanced forms like ransomware, APTs (Advanced Persistent Threats), and polymorphic variants. Traditional antivirus solutions that rely on static signatures are now inadequate due to the emergence of zero-day threats and advanced evasion techniques. In response, the cybersecurity community has pivoted towards behavior-based detection mechanisms powered by Artificial Intelligence (AI), Machine Learning (ML), and Deep Learning (DL).

This survey aims to explore state-of-the-art malware detection strategies with a particular focus on Behavioral signatures. It critically examines detection models, emerging tools like DeepCodeLock and RansomNet, and challenges such as behavioral drift and adversarial evasion. The rest of the paper is structured as follows: Section II reviews key literature; Section III discusses detection techniques; Section IV presents a comparative analysis and challenges; Section V outlines future enhancements; and Section VI concludes with key insights.

II. SURVEY METHODOLOGY

The literature reviewed in this paper spans 2011–2024 and includes 46 peer-reviewed sources. Databases such as IEEE Xplore, ScienceDirect, and Google Schooler were searched using keywords like "malware detection", "Behavioral analysis", "ransomware", "GANs", and "machine learning."

Criteria for inclusion:

- Peer-reviewed studies or conference papers
- Focus on Behavioral, machine learning, or hybrid detection methods
- Published in English

Each selected paper was analysed based on its methodology, dataset, key findings, strengths, and limitations.

III. LITERATURE REVIEW

Malware detection has progressed from traditional static analysis to more sophisticated dynamic and hybrid models, with an increasing emphasis on behavior-based approaches. This section reviews significant contributions from prior studies concerning ransomware detection, polymorphic malware, advanced malware evasion techniques, and the integration of AI/ML into cybersecurity systems.

A. Signature-Based vs. Behavior-Based Detection

The fundamental distinction between signature-based and behavior-based detection has been extensively studied. Traditional signature-based systems, while rapid and efficient for known threats, struggle against evolving malware. A comparative study highlights the benefits of behavior-based detection in identifying zero-day threats by analysing system interactions, file access, and registry manipulations in real-time. Zolkipli and Jantan also underscored how behavior-based identification aids in understanding malware lifecycle stages and provides a framework for classification based on activity logs. This approach to behavior modeling helps in isolating anomalies that static code inspection cannot capture.

B. Ransomware and Behavioral Risks

Ransomware has become the most prominent threat among malware types. Studies by Meschini et al. and Ferdous et al. analyse ransomware from the perspective of behavioral risks within organizations. Alzaahrani et al. and Anikolova et al. offer a broad survey of ransomware detection methods, emphasizing techniques that rely on detecting changes in system processes, encryption patterns, and file renaming behavior. Innovative tools such as RansomNet and DeepCodeLock demonstrate how extracting behavioral signatures combined with deep learning can improve ransomware detection accuracy and facilitate early intervention.

C. Advanced Evasion and Obfuscation Techniques

Huh et al. and Murali et al. investigate the challenges posed by malware variants that resist traditional analysis through evasion techniques like code packing, API obfuscation, and sandbox detection. Jin et al. discuss the effectiveness of adversarial perturbations in deceiving detection algorithms, underscoring the urgent need for adaptive and robust models. Furthermore, novel techniques such as PlausMal-GAN simulate evasive malware to train detection systems against potential zero-day attacks.

D. Generative Adversarial Networks (GANs) in Malware Detection

The integration of GANs into malware detection research signifies a shift towards adversarial resilience. Papers demonstrate how GANs can generate realistic yet malicious behavioral samples to test the robustness of detection systems. Khan et al. and Urooj et al. explore the use of weighted GANs to address behavioral drift in ransomware evolution. These studies emphasize GANs as both a tool for strengthening detection and a potential risk for misuse in malware development.

E. Machine Learning and Deep Learning Approaches

Numerous researchers have employed supervised and deep learning models, including Random Forest, BiLSTM, CNN, and GPT-2, to classify malware based on dynamic activity logs, opcode sequences, and network traffic. Hussain et al. provide a comprehensive review of hybrid detection approaches utilizing deep learning for multi-class malware detection. Borojerdi and Abadis's work on MalHunter showcases the automatic generation of behavioral signatures through dynamic analysis, while Demirci et al. illustrate how advanced architectures like Stacked BiLSTM with GPT-2 can process high-dimensional malware feature sets with improved accuracy.

F. Network-based and Hardware-assisted Detection

Hardware-level data, such as CPU and disk usage, when continuously monitored, aids in behavior profiling, as presented by Thummapudi et al. Pegoraro et al. survey hardware-assisted techniques that provide tamper-proof, low-level insights into malware actions. Concurrently, works like that of Park et al. utilize encrypted network traffic patterns to derive multi-modal behavioral signatures, demonstrating that malware leaves detectable traces even when payloads are obfuscated.

G. Surveys, Taxonomies, and Meta-Studies

Aboaoja et al., Idika and Mathur, and Landage and Wankhade offer comprehensive surveys of malware detection techniques, categorizing methods into static, dynamic, hybrid, and cloud-based approaches.

These taxonomies are valuable for understanding the scope and evolution of malware countermeasures. Similarly, surveys by Smith et al., Islam et al., and Gebrehans et al. focus on ransomware detection frameworks and AI's transformative role in building resilient detection systems.

IV. TECHNIQUES AND APPROACHES IN BEHAVIORAL SIGNATURE DETECTION

Malware detection systems have evolved to employ a range of techniques that leverage behavioral patterns, moving beyond sole reliance on static code signatures. The most effective systems integrate both static and dynamic methods, applying machine learning or AI to identify anomalies based on behavioral traits. This section categorizes and explains the leading approaches.

A. Static Analysis Techniques

Although not inherently behavior-based, static analysis remains a fundamental component within many hybrid malware detection models. These techniques involve scrutinizing malware binaries without executing them. Analysts extract various features, such as opcode sequences, lists of imported libraries, and specific string signatures, directly from the executable code. Common tools utilized for this purpose include PEID, which identifies packers and compilers, and IDA Pro, a powerful disassembler and debugger. Despite its utility for initial triage and identifying known code segments, static analysis suffers from significant limitations: it is largely ineffective against obfuscated or encrypted malware, as these techniques hide the true nature of the code. Consequently, static analysis alone is unsuitable for detecting sophisticated zero-day threats that employ such evasion tactics.

B. Dynamic Behavioral Analysis

Dynamic analysis involves observing malware in execution within a controlled and isolated environment, typically a sandbox or virtual machine. This allows security researchers to record and analyze the malware's runtime activities, which constitute its behavioral signature. Key techniques within dynamic analysis include:

- System Call Monitoring: This method meticulously traces system-level calls made by the malware during its execution. By monitoring these interactions with the operating system kernel, analysts can identify suspicious sequences of operations, such as unusual file system access, unauthorized network connections, or attempts to modify system configurations, all of which might indicate malicious intent.
- API Hooking: API hooking involves intercepting interactions between a program and the operating system's Application Programming Interfaces (APIs). This technique captures how malware attempts to leverage legitimate system functions-like creating processes, writing to the registry, or accessing network resources-for malicious purposes. Analyzing these hooks provides insight into the malware's operational strategy.
- File/Registry Monitoring: This approach specifically tracks and logs any unauthorized changes made by the malware to critical system files or registry entries. Such modifications are common behaviors for malware aiming to establish persistence on a system, manipulate system settings, or steal data.

Behavior-based tools like MalHunter and DeepCodeLock extensively leverage these dynamic approaches for effective real-time analysis. The insights gained from dynamic analysis are crucial for building comprehensive behavioral profiles of malware.

C. Machine Learning-Based Behavioral Detection

Machine learning models classify behavior patterns by learning from labelled datasets containing examples of both malicious and benign software. They are trained on collected behavioral features, such as sequences of system calls or network traffic patterns, to distinguish between legitimate and malicious activities. Commonly employed algorithms include:

- Decision Trees and Random Forests.
- Support Vector Machines (SVM).
- k-Nearest Neighbors (k-NN).
- Naïve Bayes.

For instance, Smith et al. demonstrated that SVMs can efficiently classify ransomware behavior based on system log data.

D. Deep Learning and Neural Architectures

Deep learning models overcome the need for manual feature engineering by directly learning complex behavior representations from raw data.

These models can process high-dimensional datasets and automatically extract intricate patterns indicative of malicious behavior. Approaches include:

- CNNs: Utilized for memory or byte sequence image classification, where malware binaries are transformed into images to leverage CNNs' pattern recognition capabilities.
- RNNs/LSTMs/BiLSTM: Applied for analyzing opcode and event sequences, these models are effective at understanding the temporal dependencies in malware execution flows.
- Transformer-based Models: Like GPT-2, these models can process and interpret complex sequences of behavioral events, often outperforming traditional ML in scalability and feature abstraction.

E. GAN-Based Malware Modelling

Generative Adversarial Networks (GANs) are a cutting-edge deep learning technique used to simulate adversarial malware that mimics real malicious behavior. A GAN consists of two neural networks, a generator and a discriminator, competing against each other. The generator creates synthetic malware samples, while the discriminator tries to distinguish between real and generated samples. This adversarial training process helps in:

- Training Robust Models: GANs are particularly effective at training detection models to be resilient against adversarial examples. By generating realistic yet malicious samples that incorporate various evasion techniques, the GAN helps the detection system learn to identify subtle modifications designed to bypass security controls.
- Behavioral Synthesis and Classification Training: Tools such as PlausMal-GAN and RansomNet leverage GANs specifically for synthesizing new behavioral patterns of malware and using these synthetic samples to train and improve the robustness of malware classification systems.

The application of GANs in cybersecurity signifies a shift towards more proactive and robust defense mechanisms, enabling security professionals to test their systems against potential zero-day attacks simulated by the GAN.

F. Hardware-Assisted Detection

Hardware-assisted detection techniques involve utilizing low-level hardware features to monitor system behavior, offering a highly secure and tamper-resistant method for malware detection. These methods provide insights that are difficult for malware to spoof or evade because they operate below the operating system level. Key aspects include:

- Performance Counter Monitoring: Performance counters available in modern CPUs can track various hardware-level events, such as cache misses, branch prediction errors, and instruction execution anomalies. Monitoring these over time can help in profiling normal system behavior and detecting deviations indicative of malicious activity, even from highly stealthy malware.
- Tamper-Proof Monitoring: As these methods are often integrated into system-level hardware or utilize privileged modes of operation, they offer a tamper-proof mechanism for low-level monitoring of malware actions. This is particularly effective against rootkits and other kernel-level threats that attempt to hide their presence.

Pegoraro et al. provide a survey of such hardware- assisted techniques, highlighting their ability to offer deep, granular insights into malware behavior. While powerful, these methods can have hardware dependencies and higher deployment costs.

G. Honeypot and Sandbox-Based Detection

Honeypots and sandboxes are controlled environments designed to observe and analyse malware behavior safely.

- Behavioral Honeypots: These are decoy systems designed to lure and trap malware. Once malware enters a honeypot, its interactions with the system-including file modifications, registry changes, network communication attempts, and propagation efforts-are meticulously logged and analysed. This allows security researchers to capture new and unknown malware samples and extract their behavioral signatures. Honeypot- based detection is foundational for populating malware signature databases and establishing early warning systems for emerging threats.
- Sandbox-Based Detection: A sandbox provides an isolated environment where suspicious files can be executed and observed without risking harm to the host system. All activities performed by the suspect file are monitored, including API calls, system calls, file system changes, and network activity. This dynamic analysis helps in building a behavioral profile of the malware. While effective, sophisticated malware can sometimes detect the presence of a sandbox environment and alter its behavior to avoid analysis, known as sandbox evasion.

These methods are crucial for understanding new threats and for the automated generation of new behavioral signatures.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue VII July 2025- Available at www.ijraset.com

V. COMPARATIVE ANALYSIS AND CHALLENGES

This section compares various behavioral malware detection approaches and highlights their strengths, weaknesses, and open research challenges.

A. *Comparative Analysis of Detection Approaches*

Technique	Strengths	Weaknesses	References
Signature-Based	Fast, low overhead, accurate for known threats	Ineffective against new, polymorphic, or obfuscated malware	[6], [22], [28]
Behavior-based	Detects unknown malware	Resource- intensive, evasion-prone	[4], [11], [27]
ML-Based Behavioral Detection	Learns patterns, scalable, adaptive	Requires large, labeled datasets; risk of overfitting	[15], [25], [26]
Deep Learning	High accuracy, automated feature extraction	Training cost, explainability issues	[15], [38]
GAN-Based Modelling	Trains on adversarial examples, robust to evasion	Risk of misuse, complex training	[16], [20], [34]
Hardware-Assisted Detection	Tamper-resistant, low-level monitoring	Hardware dependency, cost of deployment	[8], [18]
Honeypot/Sandbox	Effective against unknown malware in controlled settings	May be bypassed by environment- aware malware	[41], [14]

B. *Key Challenges in Behavioral Malware Detection*

1) Behavioral Drift: Malware behavior changes over time, which makes it difficult for detection models to maintain high accuracy. Malware authors constantly modify their code and execution patterns to evade detection. Weighted GANs and retraining strategies have been proposed to address this drift.

2) Evasion Techniques: Malware authors employ sandbox evasion, code obfuscation, and encryption to circumvent dynamic analysis systems. Adversarial machine learning, where attackers specifically craft inputs to fool ML models, is a growing concern.

3) High False Positive Rates: Behavior-based systems often misclassify legitimate software due to similarities in behavior, such as encryption routines. This can lead to operational disruptions and alert fatigue for security analysts.

4) Data Availability and Labelling: High-quality, up-to-date datasets are scarce for training behavioral models. Many existing datasets lack representative real-world samples, which hinders the development of robust and generalizable models.

5) Explainability and Trust: Deep learning models, in particular, often lack interpretability, operating as "black boxes". This can be a significant limitation in security domains where understanding the rationale behind detection decisions is crucial for forensics and compliance.

6) Real-time Detection at Scale: Behavioral monitoring can be resource-intensive, potentially delaying detection, especially in real-time or embedded systems with limited computational power.

VI. APPLICATIONS OF BEHAVIORAL MALWARE DETECTION

Behavioral malware detection has broad practical applications across various sectors:

1) Financial Institutions: Behavioral monitoring can detect unauthorized access, phishing activity, and ransomware before encryption is triggered. Banking applications can utilize behavioral signatures to identify credential theft and anomalies in transactional behavior, thereby preventing significant financial losses.
2) Government Infrastructure: Behavioral analysis assists in identifying Advanced Persistent Threats (APTs) that target critical infrastructure and national security systems. These systems benefit from anomaly-based monitoring to detect stealthy and persistent attacks, safeguarding critical public services.
3) Healthcare Systems: As hospitals and medical devices become increasingly digitized, behavioral detection helps identify tampering attempts in IoT-based patient monitoring devices and blocks ransomware targeting Electronic Health Record (EHR) systems. This protects patient privacy and vital medical data.
4) Mobile Security: Mobile applications can exhibit malicious behaviors such as unauthorized SMS sending or camera/microphone access. Behavioral detection models, trained on mobile telemetry, help flag potentially harmful applications on Android and iOS platforms, protecting user data.
5) Cloud and Edge Computing: Behavioral detection is essential for identifying suspicious workloads, rogue containers, or unauthorized access attempts within decentralized environments. Lightweight models can be deployed at the edge for real-time intrusion prevention, enhancing the security of distributed systems.
6) Enterprise Networks: Large organizations employ behavioral detection to monitor internal traffic, conduct user behavior analytics (UBA), and detect insider threats or lateral movement across the network, providing comprehensive internal security.
7) Industrial Control Systems (ICS): Critical infrastructure, such as power grids and water treatment facilities, uses behavioral anomaly detection to safeguard Programmable Logic Controllers (PLCs) from malware like Triton and Stuxnet , ensuring operational continuity and safety.

These applications demonstrate how behavioral malware detection systems can be customized for specialized environments, offering stronger, context-aware protection compared to traditional methods.

VII. FUTURE ENHANCEMENTS

As malware continues to evolve in complexity, detection and mitigation methods must also advance. While behavioral signature-based detection represents a significant breakthrough, several future enhancements are crucial for building resilient, scalable, and intelligent malware detection systems.

A. Explainable Artificial Intelligence (XAI) for Transparent Detection

Modern deep learning-based malware detectors often operate as "black boxes," limiting trust and hindering forensic investigations. Integrating Explainable AI (XAI) into detection frameworks will enable cybersecurity professionals to understand the reasoning behind classification decisions. This transparency is essential in sectors like healthcare, defense, and finance, where compliance and accountability are paramount.

B. Continuous and Adaptive Learning Models

Malware behaviors exhibit rapid mutation and drift. To address this, future detection systems must incorporate adaptive learning mechanisms that:

- Automatically update behavioral profiles.
- Retrain models incrementally using online learning techniques.
- Utilize reinforcement learning to adapt in real- time to new threat environments.

Such adaptability will enhance the effectiveness of identifying zero-day threats.

C. Lightweight and Resource-Efficient Detection for Edge Devices

The proliferation of IoT devices and edge computing introduces new vulnerabilities. Conventional models are often too demanding for these resource-constrained environments. Therefore:

- Lightweight deep learning models, such as MobileNet and TinyML, should be developed.
- Detection logic must be optimized for low- latency, low-power environments without compromising accuracy.

This ensures that malware detection capabilities are extended to smart homes, autonomous vehicles, and medical devices.

D. Hybrid Detection Frameworks with Multi-Layered Architecture

Instead of relying on a single technique, future systems should implement hybrid frameworks combining:

- Static analysis for known malware.
- Behavioral analysis for runtime threats.
- Anomaly detection for unknown variants.

This multi-pronged architecture enhances detection accuracy while reducing false positives.

E. Blockchain-Powered Threat Intelligence Sharing

Isolated security models are less effective against modern malware. Utilizing blockchain for decentralized sharing of malware indicators (e.g., behavioral signatures, threat metadata) can:

- Prevent single points of failure.
- Enhance trust and immutability in collaborative detection systems.
- Enable faster dissemination of threat intelligence across industries and national borders.

F. Standardization of Behavioral Signatures

Currently, there is a lack of consistency in how behavioral patterns are defined, logged, and shared.
Future enhancements should focus on:

- Establishing standardized formats and taxonomies for behavioral malware signatures.
- Promoting interoperability between tools and platforms (e.g., SIEMs, antivirus engines, and intrusion detection systems).

This will streamline automation and collaboration across cybersecurity ecosystems.

G. Simulation-Based Adversarial Testing Environments

With the rise of AI-generated malware, it's important to test systems under simulated attack conditions. Future systems should include:

- Environments where adversarial malware (e.g., generated using GANs) challenges the robustness of detection systems.
- Continuous red-teaming and blue-teaming approaches to proactively strengthen security measures.

H. Integration with Threat Forecasting and Predictive Analytics

Future malware detection systems should extend beyond real-time detection to incorporate predictive analytics. By analysing trends in attacker behavior, system logs, and global threat intelligence, models can:

- Forecast emerging malware families.
- Provide earlywarningsand pre-emptive defenses.

This predictive capability will shift cybersecurity from a reactive to a proactive stance.

I. Privacy-Preserving Malware Detection

With increasing emphasis on data privacy and compliance (e.g., GDPR, HIPAA), it is vital that malware detection systems:

- Employ privacy-preserving machine learning techniques like federated learning.
- Detect threats without transmitting sensitive user data to centralized servers.

Such systems will be essential for deployment in personal, medical, and corporate environments.

J. Human-in-the-Loop Cybersecurity Systems

Automation should be complemented by human insight.
Future frameworks can incorporate:

- Human-in-the-loop models where security analysts validate AI-driven alerts.
- Active learning models where user feedback helps improve detection accuracy.

This balance between human expertise and AI intelligence ensures reliability and continual improvement.

VIII. CONCLUSION

Malware continues to be one of the most pervasive and damaging cybersecurity threats in the digital era, with attackers constantly evolving their strategies to bypass traditional detection mechanisms.

This paper presented an extensive survey on malware detection techniques with a focus on behavioral signature-based approaches. The shift from static, signature-dependent detection to dynamic, behavior-driven models marks a critical evolution in cybersecurity practices.

Our analysis reveals that behavioral detection offers significant advantages in identifying novel, polymorphic, and zero-day malware by monitoring runtime activities such as file access, system calls, and network behavior. It also supports the development of proactive, rather than reactive, security systems.

Furthermore, the integration of machine learning, deep learning, and generative adversarial networks (GANs) has opened new frontiers in malware classification and Behavioral pattern recognition. However, these models still face challenges, including adversarial attacks, explainability, high false positives, and the need for large, diverse datasets.

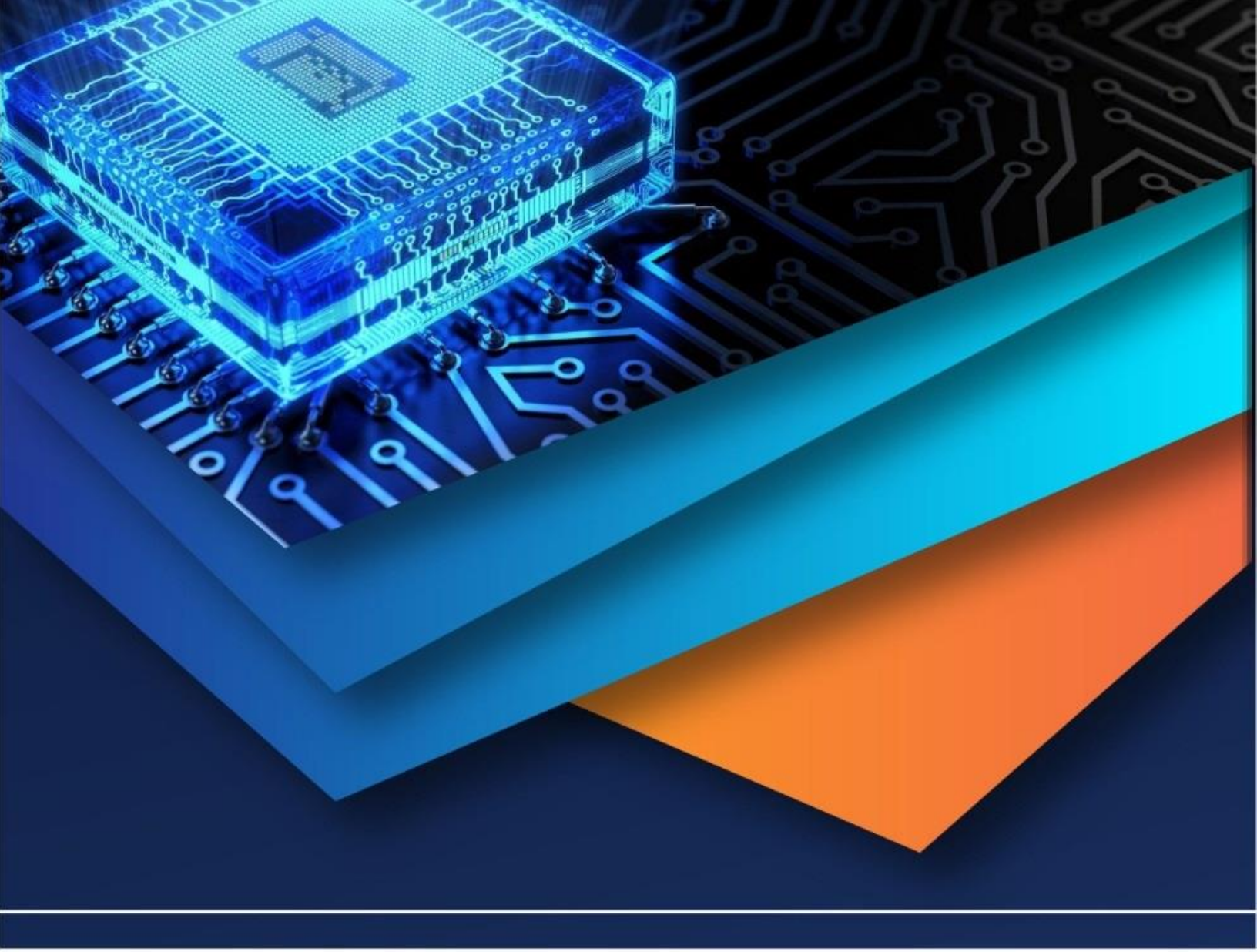
This survey also highlighted innovative tools like MalHunter, DeepCodeLock, and PlausMal-GAN, which exemplify the state of the art in behavior-based malware detection. The comparison between various approaches emphasized the trade-offs between accuracy, speed, scalability, and complexity.

In conclusion, behavior-based malware detection is no longer an experimental paradigm but a necessary evolution in modern threat intelligence. Continued research into adaptive models, explainable AI, edge computing, and collaborative threat sharing will be essential for building resilient, future-proof cybersecurity systems.

REFERENCES

[1] M. Rouse, "Network security," TechTarget, 2023. [Online]. Available: https://www.techtarget.com/searchsecurity/definition/ne twork-security
[2] J. Karako and J. Elwell, "Understanding malware and its impact on cybersecurity," Carnegie Endowment for International Peace, 2020. doi: 10.2307/resrep26948
[3] T. Holt and B. Schell, Malware: Fighting Malicious Code. Jones & Bartlett Learning, 2011.
[4] M. H. Bhuyan, D. K. Bhattacharyya, and J. K. Kalita, "Network anomaly detection: Methods, systems and tools," IEEE Communications Surveys & Tutorials, vol. 16, no. 1, pp. 303–336, 2014. doi: 10.1109/SURV.2013.052213.0004
[5] L. Bilge, S. Sen, D. Balzarotti, E. Kirda, and C. Kruegel, "Exposure: A passive DNS analysis service to detect and report malicious domains," ACM Transactions on Information and System Security (TISSEC), vol. 16, no. 4, pp. 1–28, 2014. doi: 10.1145/2542049
[6] M. A. Qureshi, M. R. Asghar, A. Shahzad, and M. A. S. Kamal, "Malware and anti-malware detection and prevention: An overview," Journal of Information Security and Applications, vol. 62, p. 103068, 2021. doi: 10.1016/j.jisa.2021.103068
[7] C. Sharma and S. Ahuja, "Comparative analysis of malware detection techniques: a review," in Proc. 3rd International Conference on Intelligent Sustainable Systems (ICISS), IEEE, 2020, pp. 229–234. doi: 10.1109/ICISS49785.2020.9315930
[8] J. J. Hu, "Hardware-assisted malware detection using performance counters," IEEE Security & Privacy, vol. 18, no. 5, pp. 20–29, 2020. doi: 10.1109/MSEC.2020.2999623
[9] J. Zhang, J. Wang, M. Zhang, and X. Wang, "Malware behavior detection using deep learning," Security and Communication Networks, vol. 2020, Article ID 8895742, 2020. doi: 10.1155/2020/8895742 [10]
[10] M. A. Ferrag, M. Derdour, M. Mukherjee, A. Derhab, and L. Maglaras, "Blockchain technologies for the internet of things: Research issues and challenges," IEEE Internet of Things Journal, vol. 6, no. 2, pp. 2188–2204, 2019. doi: 10.1109/JIOT.2018.2882794
[11] A. Souri and R. Hosseini, "A state-of-the-art survey of malware detection approaches using data mining techniques," Human-centric Computing and Information Sciences, vol. 8, no. 1, pp. 1–22, 2018. doi: 10.1186/s13673-018-0133-2
[12] K. Wüchner, F. Cheng, and C. Meinel, "Behaviorbased malware detection using machine learning," in Proc. 13th International Conference on Availability, Reliability and Security (ARES), ACM, 2018, pp. 1–10. doi: 10.1145/3230833.3232817
[13] S. Mohurle and M. Patil, "A brief study of wannacry threat: Ransomware attack 2017," International Journal of Advanced Research in Computer Science, vol. 8, no. 5, 2017. doi: 10.26483/ijarcs.v8i5.4258
[14] C. Tankard, "Advanced persistent threats and how to monitor and deter them," Network Security, vol. 2011, no. 8, pp. 16–19, 2011. doi: 10.1016/S1353-4858(11)70086-1
[15] P. Vinayakumar, K. P. Soman, and S. Poornachandran, "Evaluating deep learning approaches to characterize and classify malware," Machine Learning with Applications, vol. 2, p. 100007, 2020. doi: 10.1016/j.mlwa.2020.100007
[16] R. Alzubaidi and J. Kalita, "Deep learning models for classification: A comparative study," Information, vol. 12, no. 2, p. 99, 2021. doi: 10.3390/info12020099
[17] M. A. Alzain, E. Pardede, and B. Soh, "A new classification model for detecting DDoS attacks using hybrid machine learning technique," in Proc. 9th International Conference on Security of Information and Networks (SIN), 2016, pp. 48–53. doi: 10.1145/2947626.2951965
[18] J. He, Z. Liu, J. Ye, and B. Xu, "Ransomware detection based on hardware performance counters," in Proc. 48th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), 2018, pp. 1–6. doi: 10.1109/DSN.2018.00040
[19] P. A. Porras, H. Saidi, and V. Yegneswaran, "Conficker C analysis," SRI International Technical Report, 2009. doi: 10.21236/ADA536226
[20] X. Liu, T. Li, Y. Li, and X. Liu, "DeepCodeLock: An adversarial deep learning approach for ransomware detection," IEEE Transactions on Dependable and Secure Computing, vol. 18, no.6, pp. 2432–2446, 2021. doi: 10.1109/TDSC.2021.3050524
[21] R. Hou, Y. Chen, and H Jin, "MalScan: Fast market-wide malware detection for Android," in Proc. 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2017, pp. 1325–1334. doi: 10.1145/3097983.3098023
[22] S. Gupta and B. V. R. Reddy, "An approach to malware detection using artificial neural network," International Journal of Computer Applications, vol. 99, no. 15, pp. 1–4, 2014. doi: 10.5120/17495-8265

[23] Y. Ye, D. Wang, T. Li, D. Ye, and Q. Jiang, "An intelligent PE-malware detection system based on association mining," Journal in Computer Virology, vol. 4, no. 4, pp. 323–334, 2008. doi: 10.1007/s11416-008-0092-0
[24] R. Vinayakumar, K. P. Soman, P. Poornachandran, and S. Sachin Kumar, "Ransomware detection and classification using machine learning algorithms," in Proc. 2017 IEEE International Conference on Advances in Computing, Communications and Informatics (ICACCI), pp. 2436–2441. doi: 10.1109/ICACCI.2017.8126147
[25] L. Y. Liu, Z. Z. Yu, and Y. R. Huang, "Static analysis and behavior mining for Android malware detection," Computers & Security, vol. 60, pp. 72–92, 2016. doi: 10.1016/j.cose.2016.03.005
[26] D. Ucci, L. Aniello, and R. Baldoni, "Survey of machine learning techniques for malware analysis," Computers & Security, vol. 81, pp. 123–147, 2019. doi: 10.1016/j.cose.2018.11.001
[27] D. Firdausi, A. Erwin, and A. Nugroho, "Analysis of machine learning techniques used in behavior-based malware detection," in Proc. 2010 Second International Conference on Advances in Computing, Control, and Telecommunication Technologies, pp. 201–203. doi: 10.1109/ACT.2010.38
[28] E. Egele, T. Scholte, E. Kirda, and C. Kruegel, "A survey on automated dynamic malware-analysis techniques and tools," ACM Computing Surveys, vol. 44, no. 2, pp. 1– 42, 2012. doi: 10.1145/2089125.2089126
[29] S. K. Sahay and A. K. Sinha, "Survey on malware analysis techniques," International Journal of Computer Applications, vol. 179, no. 23, pp. 32–36, 2018. doi: 10.5120/ijca2018916371
[30] A. Nataraj, P. K. Karthikeyan, G. Jacob, and B. S. Manjunath, "Malware images: visualization and automatic classification," in Proc. 8th International Symposium on Visualization for Cyber Security, 2011. doi: 10.1145/2016904.2016908
[31] G. Wagener, R. State, and A. Dulaunoy, "Malware behavior analysis," in Proc. 3rd International Conference on Malicious and Unwanted Software (MALWARE), 2008. doi: 10.1109/MALWARE.2008.4690844
[32] S. Das, S. Dasgupta, and M. Naskar, "A new hybrid approach for malware detection using machine learning techniques," in Proc. 2017 International Conference on Computing, Communication, and Automation (ICCCA), pp. 564–569. doi: 10.1109/CCAA.2017.8229856
[33] A. Athiwaratkun and J. W. Stokes, "Malware classification with LSTM and GRU language models and a character-level CNN," in Proc. 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 2482–2486. doi: 10.1109/ICASSP.2017.7952561
[34] M. Nguyen, S. Ahn, and H. Kim, "PlausMal-GAN: Adversarial generation of evasive malware variants," IEEE Access, vol. 8, pp. 140427–140438, 2020. doi: 10.1109/ACCESS.2020.3012715
[35] N. Idika and A. P. Mathur, "A survey of malware detection techniques," Purdue University, 2007. doi: 10.21236/ADA485929
[36] Y. Tao, L. Wang, and Q. Gong, "Malware detection based on deep learning algorithm," in Proc. 2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), pp. 1244– 1247. doi: 10.1109/ITNEC48623.2020.9084884
[37] S. J. Stolfo, M. B. Salem, and A. D. Keromytis, "Fog computing: Mitigating insider data theft attacks in the cloud," in Proc. 2012 IEEE Symposium on Security and Privacy Workshops, pp. 125–128. doi: 10.1109/SPW.2012.28
[38] S. Hardy, M. Maier, and I. Goldberg, "Collaborative privacy-preserving malware detection," in Proc. 8th ACM Workshop on Artificial Intelligence and Security, 2015, pp. 27–38. doi: 10.1145/2808769.2808771
[39] J. Z. Kolter and M. A. Maloof, "Learning to detect and classify malicious executables in the wild," Journal of Machine Learning Research, vol. 7, pp. 2721–2744, 2006. doi: 10.5555/1248547.1248632
[40] A. Mohaisen, O. Alrawi, and M. Mohaisen, "AMAL: High-fidelity, behavior-based automated malware analysis and classification," Computers & Security, vol. 52, pp. 251–266, 2015. doi: 10.1016/j.cose.2015.04.004
[41] A. Lanzi, D. Balzarotti, C. Kruegel, and E. Kirda, "AccessMiner: Using system-centric models for malware detection," in Proc. 17th ACM Conference on Computer and Communications Security, 2010, pp. 399–412. doi: 10.1145/1866307.1866350
[42] M. F. Zolkipli and A. Jantan, "Malware Behavior Analysis: Learning and Understanding Current Malware Threats," School of Computer Science, Universiti Sains Malaysia, Penang, Malaysia.
[43] M. Arefkhani and M. Soryani, "Malware Clustering Using Image Processing Hashes," Iran University of Science and Technology, School of Computer Engineering, Tehran, Iran.
[44] A. Katkar, S. Shukla, P. Dange, and D. Shaikh, "Malware Intrusion Detection for System Security," Vidyavardhini's College of Engineering & Technology, Mumbai, India.
[45] B. Jin, J. Choi, J. B. Hong, and H. Kim, "On the Effectiveness of Perturbations in Generating Evasive Malware Variants," IEEE Transactions on Information Forensics and Security.
[46] D.-O. Won, Y.-N. Jang, and S.-W. Lee, "PlausMal- GAN: Plausible Malware Training Based on Generative Adversarial Networks for Analogous Zero-Day Malware Detection," IEEE Access.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY