International Research Journal of Education and Technology

Peer Reviewed Journal, **ISSN 2581-7795**

Cyber Sentinel: Trojan and URL Detection Using Hybrid Machine Learning Model

[1]*Mrudula Sanjay Patil*
[2]*Krutika Vijay Patil*
[3]*Gauri Rajaram Patil*
[4]*Dhanashree Dhanaraj Patil*
[5]*V. O. Patil, Professor, Dept. of Computer Engineering, D.N.Patel College of Engineering, Shahada*

Abstract – *The project introduces a web-based hybrid detection system that has been created to enhance cyber security against contemporary threats like phishing, malware, ransomware, and Trojans. Conventional security mechanisms tend to fail when confronted with advanced cyberattacks, mainly because they rely on static rules. The system overcomes that drawback by combining both supervised and unsupervised machine learning algorithms. Random Forest and XGBoost are utilized for precise URL classification and identification of malicious activity, and K-Means and Isolation Forest are employed for powerful anomaly detection. Scalability and simplicity are built into the system, with a user-friendly dashboard appropriate for both technical and non-technical users. Real-time notifications and detailed log reports facilitate effective threat monitoring and response. Through its integration of both conventional and state-of-the-art detection techniques, the system offers an effective proactive cybersecurity solution that learns on its own, evolves to face new threats, mitigates false positives, and increases organizational resilience in an ever-changing digital world.*

Key Words: Hybrid detection, Random Forest, XGBoost, K-Means, Isolation Forest, URL classification, Trojan detection, anomaly detection, cybersecurity.

1. INTRODUCTION

In this era of interconnected digital networks, cybersecurity attacks are evolving at an unprecedented rate, causing great harm to individuals, businesses, and governments around the globe. Conventional security products, like antivirus software and URL blacklists, have grown ineffective in countering these dynamic and constantly changing threats.

Polymorphic malware, phishing attacks, ransomware, and zero-day exploits take advantage of the inherent inability of these static technologies, resulting in compromised systems, economic losses, and large-scale data breaches. The surge in phishing attacks, malware infections, and ransomware attacks underlines the need for smart, dynamic, and proactive defenses. Recent research has shown that phishing sites and malware have become more evasive and adaptive, using methods like hidden URLs and behavioural transformation to mask themselves. Attackers are increasingly using obfuscation techniques to evade conventional antivirus protection, which rely on pre-defined rules and established attack patterns, thereby revealing the weaknesses of static defences. To overcome these shortcomings, hybrid machine learning models have proven to be an attractive solution. These models use multiple algorithms to examine patterns, spot anomalies, and forecast emerging threats in real time. Machine learning-driven solutions provide a substantial benefit in continuously learning from data and evolving with new attack vectors, as

opposed to traditional security frameworks that still have exposure to unknown or emerging threats. This report exhibits the design, development, and implementation of a web based hybrid detection system to bridge the shortcomings of conventional cybersecurity solutions. The proposed system integrates supervised and unsupervised machine learning methods to classify URLs and identify malicious activities in real time. In particular, Random Forest and XGBoost algorithms were used due to high accuracy, robustness, and low-latency performance, with effective detection using minimal false positives. The hybrid mechanism allows phishing websites, malware processes, unusual system behavior, and malicious URLs to be identified, solving the changing nature of cyber threats. Aside from its sound technical functionalities, the solution is scaled and user-friendly. The system gives out detailed logs, visualization aids, and real-time notifications, so network administrators can track and manage threats efficiently. For non-technical users, the platform offers an intuitive dashboard for submitting URLs and files, making it accessible to users with varying levels of expertise. The broader objective of this project is to bridge the gap between traditional and modern cybersecurity solutions by integrating dynamic machine learning technologies. This hybrid system provides a scalable, proactive defence mechanism, reducing the need for constant human intervention while improving the security stance of organizations. The subsequent sections delve into the design, development, and implementation process in depth, along with an afterthought on challenges encountered, solutions deployed, and important lessons garnered during the course of the project.

2. LITERATURE SURVEY

This research is aimed at detecting malicious or bogus URLs, particularly phishing links, through machine learning. Rather than using classical blacklists that can only pick up known threats, the authors employed sophisticated

algorithms such as KNN, SVM, and XGBoost to identify new or unknown malicious links. These models were trained based on features of the URLs themselves—i.e., length of the link, special characters employed, and domain data from WHOIS records. This method enabled the system to attain more than 95% accuracy and proved it efficient in catching even zero-day phishing attacks. The research confirms that machine learning is an improved and more scalable option for safeguarding users against online threats.

1. Paper Name: Malicious URL Detection Using Machine Learning Techniques.
Author: R. Gupta, K. Roy, L. Chen
Abstract: The research introduces a robust deep learning architecture that marries Convolutional Neural Networks (CNN) with Quasi-Recurrent Neural Networks (QRNN) in order to identify cyber attacks like malware and phishing. CNNs are employed to find patterns within organized data, and QRNNs assist in comprehending how data shifts over time. By merging these two methods, the model is enhanced in its ability to identify threats that change or act differently over time. The researchers tested the model on real network traffic data, which means it was trained on actual data packets that are passed through a network. This method enables the system to learn and identify normal and abnormal behaviors. Since it detects patterns of events and timing (time-series analysis), the model is particularly effective at catching threats in real time. The results show that this hybrid method works very well in large networks, making it ideal for use in businesses and organizations that need to monitor cybersecurity threats continuously and accurately.

2. Paper Name: Deep Hybrid Model for Cyber Threat Detection in Network Traffic.
Author: P. Das, N. Verma, T. Yamada.
Abstract: The research introduces a robust deep learning architecture that marries Convolutional Neural Networks (CNN) with Quasi-Recurrent Neural Networks (QRNN) in order to identify cyber attacks like malware and phishing. CNNs are employed to find patterns within organized data, and QRNNs assist in comprehending how data shifts over time. By merging these two methods, the model is enhanced in its ability to identify threats that change or act differently over time. The researchers tested the model on real network traffic data, which means it was trained on actual data packets that are passed through a network. This method enables the system to learn and identify normal and abnormal behaviors. Since it detects patterns of events and timing (time-series analysis), the model is particularly effective at catching threats in real time. The results show that this hybrid method works very well in large networks, making it ideal for use in businesses and organizations that need to monitor cybersecurity threats continuously and accurately.

3. Paper Name: Detection of Malicious URLs Using Natural Language Processing and Machine Learning.
Author: A. Patel, S. Mehta, H. Lee
Abstract: This paper discusses how Natural Language Processing (NLP) and machine learning may be used together to identify dangerous or spurious URLs. The approach is to treat URLs as sentences composed of characters, and then process these "sentences" through NLP techniques. Authors employ models such as Naive Bayes, Random Forest, and XGBoost to classify a URL as safe or unsafe. Among the main methods applied is n-gram analysis, where the URL is split into smaller segments (such as 2 or 3 characters at once) in order to obtain patterns that are frequently encountered in malicious links. For instance, spurious URLs attempt to resemble valid ones such as banks or social media websites (e.g., "paypa1.com" instead of "paypal.com"). By examining these patterns, the models are able to learn how to identify URLs attempting to deceive users. The findings indicate that this method is highly precise in identifying these malicious links, and therefore represents a useful resource in warding off phishing attacks and enhancing security on the internet.

4. Paper Name: Trojan Detection through Network Behavior Analysis using Deep Learning
Author: L. Zhang, M. Oliveira, S. Joshi
Abstract: The paper introduces a deep learning-based method for detecting Trojan malware based on tracking unusual patterns of network activity. Instead of depending on static signatures as conventional antivirus software, the system looks at how devices use the network in real time. For instance, it searches for frequent communication with unknown or suspect IP addresses, data being transmitted in encrypted or obscured fashion, and using ports that are not usually active. All these activity patterns are usually an indication that a Trojan is running in the background. The model employs Convolutional Neural Networks (CNN) to identify spatial patterns in the data (such as packet organization) and Gated Recurrent Units (GRU) for the processing of time-based actions (such as habitual activity or timing trends). The integration of the two models makes it capable of identifying even well-concealed dangers that evolve dynamically to evade discovery. It was trained on extensive collections of network traffic data from secure systems as well as Trojan-infected computers. The findings revealed high accuracy and excellent real-time performance, which makes it ideal for real-time deployment in enterprise networks where active threat detection is of the essence.

5. Paper Name: URLNet -Learning URL Representations for Malicious URL Detection.
Author: H. Le, A. Hoang, C. Thach
Abstract: URLNet is a deep learning framework that can automatically identify malicious URLs. Contrary to conventional approaches that are heavily dependent on manual feature extraction, URLNet employs character-level and word-level embeddings to directly learn patterns from raw URL strings. It makes use of CNN layers in order to comprehend structural homologies between bad and good links. Detection of obfuscated and newly crafted phishing URLs is enhanced through this method. The model saw top-notch performance on public benchmark datasets, illustrating the potential of neural networks in URL security.

3. OBJECTIVE

1. To create a hybrid machine learning model that integrates mainstream ML classifiers with deep learning structures in order to detect Trojan malware and malicious URLs in real-time effectively.
2. To create an interactive, responsive web application with Next.js and React, wherein users can upload files and URLs for threat analysis, and get results in real time through a simple, intuitive interface.
3. To create and implement dashboards for administrators and users with real-time threat reports and graph based on visual insights using Chart.js, allowing for improved monitoring and system transparency.
4. To store and manage application data such as user activity, prediction outcomes, and system logs securely and effectively by utilizing MongoDB as the back-end database.
5. To facilitate flexibility and ongoing learning through the system design that enables it to learn from fresh threat intelligence over time, enhancing its detection potential with minimal human interaction.

4. METHODOLOGY

Trojan and URL Detection System uses a hybrid machine learning system that effectively identifies Trojan and URL-based threats. The development is carried out by Python for machine learning model development, Flask for RESTful API integration, and Next.js for the user interface. The backend database is handled using MongoDB, and the frontend interface is designed with Tailwind CSS.
1. Malicious URL Detection: URLs are inspected using structural attributes like length, special characters usage, domain name patterns, occurrence of HTTPS, and symbol frequency. A Random Forest classifier is trained using a labeled dataset of phishing, malware, defacement, and benign URLs. The model is trained and incorporated into a Flask API and hosted for real-time inference.
2. Trojan Detection: System-level attributes like CPU utilization, memory usage, process lists, and network usage are mined for the detection of Trojans. XGBoost and Isolation Forest are some of the models employed to examine behavioral patterns and identify anomalies that suggest Trojan activity. The system is capable of detecting known and zero-day threats by hybridizing supervised and unsupervised learning methods.
3. Web Interface and Dashboard: Web users have the ability to upload URLs or files via a Next.js interface, which communicates with the backend through Flask API calls. Real-time prediction results are returned, including a confidence rating and threat classification. A dashboard offers access to detection history, system logs, and threat analysis charts via Chart.js.
4. Admin Control and Backend Administration: Admins can securely log in through JWT-based authentication and access a dashboard with in-depth logs, user queries, and system health statistics. MongoDB persists all prediction history, user information, and system logs, which enables effective data management and retraining the future model.
5. Hybrid Model Approach: The model integrates Random Forest, CNN, KNN, XGBoost, and QRNN models to enhance accuracy and diminish false positives. Through this hybrid approach, robustness is improved and the system is able to detect an extensive range of threats with low latency. Continuous learning mechanisms allow the system to adapt by updating against new threats and hence make it adaptive and scalable.

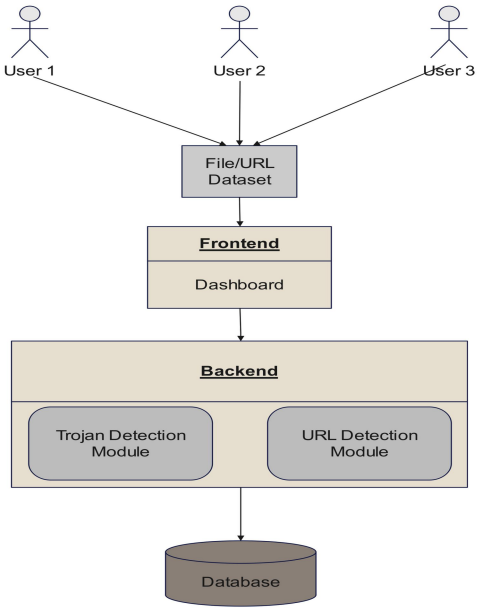

Fig -1: System Architecture

5. RESULT AND MODEL

The Cyber Sentinel platform is designed to cater to two types of users: Administrators and General Users. Each user has access to various modules of the platform, which cater to their requirements. The design is tailored to be user-friendly, responsive, and secure, providing easy access to non-technical users and controlling utilities for administrators.

- USER

Normal users are able to engage with the system via an intuitive and clean dashboard, enabling them to run basic cybersecurity scans without the need for technical knowledge.
1. Signup/Login: Users can securely sign up and log in through JWT-based authentication.
2. Malicious URL Detection: Users can paste or type in any suspicious URL to scan whether it is phishing, malware, defacement, or safe.
3. Trojan File Detection: Users are able to upload system activity files (e.g., logs) to scan for Trojan activity based on memory utilization, CPU spikes, etc.
4. Real-Time Prediction Results: Real-time analysis results are shown with a threat label (e.g., Safe, Malware Detected, Phishing).

5. Scan History: Users are able to see a detailed history of their past URL/file scans, including the result, date/time, and threat level.
6. Profile Management: Update profile information and reset passwords via the dashboard.
7.Responsive UI: Developed using Tailwind CSS and Next.js, making it responsive and interactive on mobile and desktop platforms.

- ADMIN

Administrators get a dedicated admin dashboard that gives an over-the-board overview of the system performance, user activity, and threat detection patterns.
1.Admin Login: Secure role-based access to the admin panel.
2.View and Manage Users: View all registered user information and activity logs.
3.Monitor System Health: Monitor API performance, prediction accuracy, and system uptime.
4.Threat Analytics Dashboard: Graphical reports through Chart.js to display daily scans, threat types identified, and system trends.
5. Manage Prediction Logs: Detailed logs of each scan uploaded on the platform with threat type and timestamp.
6. Role-Based Control: Separate permissions for user and admin to prevent misuse and maintain operational integrity.

5. CONCLUSIONS

The Malicious URL and Malware Detection System effectively tackled the fundamental issues related to conventional security systems by relying on machine learning models to offer real-time threat identification and precise predictions. By combining Random Forest, XG-Boost, K-Means, and Isolation Forest algorithms, the system proved to identify both known and unknown threats while reducing false positives. The merge of two web-based applications one utilizing Next.js and Flask, and the other utilizing HTML, PHP, and MySQL provided scalable and flexible solutions for different environments. The project achieved its goals by:
Offering timely threat detection using real-time analysis. Providing an intuitive interface that enabled technical as well as non-technical users. Limiting false positives through the use of hybrid models that balanced out each other's weaknesses. Being scalable with cloud-ready infrastructure and optimized data handling. This project emphasizes the role of machine learning in strengthening cybersecurity and illustrates how collective efforts can lead to a scalable, robust, and user-friendly system.

REFERENCES

[1] S. Karthic, S. Manoj Kumar, and P. N. Senthil Prakash, "A Hybrid Machine Learning Model for Detecting Cybersecurity Threats in IoT Applications," J. Rel. Intell. Environ., vol. 10, no. 3, pp. 75-85, 2023.
Available:https://link.springer.com/article/10.1007/s4187 0-022-01015-7.

[2] N. Al-Taleb and N. A. Saqib, "Towards a Hybrid Machine Learning Model for Intelligent Cyber Threat Identification in Smart City Environments," Appl. Sci., vol. 12, no. 4, pp. 1863, Feb. 2022.
Available:https://doi.org/10.3390/app12041863.

[3] S. Mahdavifar and A. A. Ghorbani, "Application of deep learning to cybersecurity: A survey,"Neurocomputing, vol. 347,pp. 149-176, May 2019.
Available:https://doi.org/10.1016/j.neucom.2019.0 2.056.

[4] P. Laplante and J. Voas, "Enhancing Intrusion Detection with Deep Learning: A Hybrid Approach for High-Traffic Networks," IEEE Comput., vol. 54, no. 12, pp. 48-57, Dec. 2022.
Available:https://ieeexplore.ieee.org/document/97 14079.

[5] R. A. A. Habeeb, F. Nasaruddin, A. Gani, I. A.T. Hashem, E. Ahmed, and M. Imran, "Real time big data processing for anomaly detection: A Survey," Int. J. Inf. Manage., vol. 45, pp. 289-307, Aug. 2018.
Available:https://doi.org/10.1016/j.ijinfomgt.2018. 08.006

[6] Garcia, M., & Weiss, B. (2022). Machine Learning for Cybersecurity: Principles and Practices. Springer. Available:https://link.springer.com/book/10.1007/ 978-3-030-87175-8

[7] Laplante, P., & Voas, J. (2022). Enhancing Intrusion Detection with Deep Learning: A Hybrid Approach for High-Traffic Networks. IEEE.
Available:https://doi.org/10.1109/ACCESS.2022.3 145692

[8] Scikit-learn Documentation. (n.d.). RandomForestClassifier API Reference. Retrieved October 21, 2024, from https://scikitlearn.org/stable/modules/generated/skl earn.ensemble.RandomForestClassifier.html

[9] Kumar, A., & Sharma, S. (2021). Introduction to Malicious URL Detection Techniques. Springer. https://link.springer.com/chapter/10.1007/978-981-15-6869-7_19

[10] MongoDB Documentation. (n.d.). Database Integration Best Practices. Retrieved October 21, 2024, from https://www.mongodb.com/docs/manual/best- practices