Journal of Information Systems Engineering and Management
2025, 10(37s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

A Unified Machine Learning Approach for Real-Time Intrusion, Malware, and URL Detection Using Random Forest and Advanced Algorithims

Shirley C P[1], Manicka Raja M[1], Karumanchi Dolly Sree[1], William Sebastian S[2], Alen Infant A[2*], Thanga Helina S[3]
[1]Assistant Professor, Department of CSE, Karunya Institute of Technology and Sciences, Coimbatore
[2]Student, Department of CSE, Karunya Institute of Technology and Sciences, Coimbatore
[3]Assistant Professor, Department of Commerce with Computer Application, KPR College of Arts Science and Research, Coimbatore, India.
*Corresponding Author: aleninfant@gmail.com

ARTICLE INFO

Received: 26 Dec 2024

Revised: 14 Feb 2025

Accepted: 22 Feb 2025

ABSTRACT

The research develops a single cybersecurity framework that executes real-time attack detection on intrusions and malware as well as dangerous URLs through machine learning methods. A combination of ensemble learning approaches primarily includes Random Forest together with Gradient Boosting and Naive Bayes and Support Vector Machines (SVM) exists to provide full-scale threat detection capabilities. The system employs Principal Component Analysis (PCA) together with other dimensionality reduction methods for better processing performance. The framework achieves 99% accuracy together with 94% precision, 91% recall and a 95% F1-score during evaluation. The proposed approach provides instant classification and implements automated response functions which differ from delayed reactive systems that need manual human involvement. The framework demonstrates future readiness through its design that supports growth and it will enhance its capabilities by integrating deep learning models together with adaptive learning systems during its development phase. This integrated method unites numerous safety detection systems into one unified solution leading to improved cybersecurity functions beside decreased tool dependence for security operations.

Keywords: Cybersecurity, Intrusion Detection, Malware Detection, URL Detection, Machine Learning.

INTRODUCTION

Security in the digital era has quickly become one of the many things that everyone needs to be concerned about. The explosion of connected devices, online services and data exchange made an environment full of vulnerabilities that malicious actors can find holes in the system. Unauthorized intrusions, malware infections, and malicious URLs have all become more sophisticated and represent a serious threat to sensitive information, financial stability and infrastructure integrity. Although effective in the past, traditional cybersecurity solutions are failing to keep up with these ever-changing threats and we need to build more robust and adaptive systems.

An Analysis of the above forms of related work shows that existing cybersecurity mechanisms utilize predominantly signature based detection techniques, which are dedicated to identifying known threats by implementing predefined patterns of suspicious activity. Although this can nullify known threats, it cannot be used in addressing the attack of exotic threats (novel and/or zero-day attack) which will leave vulnerable systems to the advance exploits. But heuristic based methods attempt to fill these gaps and many suffer from very high false positive rates resulting in unnecessary alerts which overload security teams. In addition, conventional systems are fragmented; each tool is managing different aspects of intrusion detection, malware analysis, etc. The lack of integration results in inefficiencies causing fragmented comprehensive threat management. In this context we have seen an exponential demand for unified, intelligent and adaptive cybersecurity solutions.

This work proposes a system that addresses these challenges by using advanced machine learning (ML) techniques to combine intrusion detection, malware detection and URL detection into a single cohesive framework. With the help of ML algorithms, cybersecurity has moved one step ahead, allowing the systems to learn from history data, and detect patterns that could be indications of a potential threat. Unlike today's 'traditional' methods, machine learning models don't require predefined rules or signatures. Instead, they live in the data dynamically, making them excellent against both known and unknown threats. One of the best known of these algorithms is Random Forest, which is a very robust and versatile model, and it is good at high accuracy and generalization capacity by means of the ensemble strategy.

The framework makes use of a wide variety of machine learning algorithms adapted to specific detection tasks. Random Forest identifies network anomalies for intrusion detection and, along with Gradient Boosting, AdaBoost, and XGBoost, can be used as a model for intrusion. Naive Bayes, KNN, ensemble methods are combination for malware detection for them to get accurate classification malicious software. The algorithms used for phishing link and malicious website detection are Logistic Regression and Voting Classifier. The proposed solution is based on Random Forest, which, always performs best on all three domains.

Immediate threat classification and detection is achieved through the system's core feature of real time processing. Traditionally, systems lack the delays required to facilitate manual intervention or batch processing which life organizations open to ongoing threats. This proposed framework eliminates these delays by automating threat analysis and delivering active insights within milliseconds after detection. It is enhanced by the integration of user-friendly interfaces which allow security teams to simply monitor and control the threats. By engineering a macrtechne system that combines advanced machine learning with intuitive design, users become empowered to respond quickly to cyber threats without technical domain expertise.

The other aspect that’s unique about this framework is its scalability and adaptability. Adaptivity to cybert threats allows the system to continue to be updated with new datasets and models as the threats evolve, which keeps the relevance intact in the case cybert threats evolve. Additionally, it handles the computational overhead problem using dimensionality reduction on PCA to reduce the amount of data being processed and to retain similar accuracy. The framework is suitable for wide deployment in different environments ranging from small scale organizations to large enterprises in which there are complex networks.

Finally, the proposed unified cybersecurity framework provides a unique and major contribution in threat detection and management. By presenting multiple detection mechanisms into one platform using the strength of machine learning, it provides a very robust and adaptive efficient solution to modern perceived security challenges. Details of the methodology of the proposed framework are presented and to show its architecture strengths it is evaluated against real world datasets and compared with the existing traditional systems in the subsequent sections of this work.

LITERATURE SURVEY

Research into cybersecurity maintains its status as a principal field because of evolving dangers and advanced forms of cyberattacks. Machine learning-based detection methods implement multiple approaches to boost cybersecurity detection throughout different security domains such as intrusion detection and malware detection and URL filtering according to Abad et al. (2023). The authors examine multiple classification systems while showing how machine learning effectively finds phishing attempts and dangerous URLs in their research which leads to successful quick threat identification [1].

Alsaedi et al. (2022) focus on malicious URL detection using an ensemble learning model that leverages cyber threat intelligence. Their approach improves the detection rate of phishing URLs, demonstrating the power of ensemble techniques in combining different models for more robust results [2]. Srinivas et al. (2022) introduce a machine learning-inspired algorithm to predict real-time network intrusions. Their approach provides effective detection of unauthorized access attempts, contributing significantly to the improvement of network security [3].

The study presents a malware detection system that combines deep learning techniques with correlation-based feature selection according to Alomari et al. (2023). The model presents an effective malware detection capability through its feature selection process which leads to more precise and efficient detections [4]. Alduailij et al. (2022) executed a study on Distributed Denial-of-Service (DDoS) detection by implementing mutual information along with the Random Forest feature importance technique. Random Forest proves to be a successful method for attack identification according to their research findings which show high detection accuracy and performance [5]. Narudkar et al. (2023) present a real-time intrusion detection system which combines the PCA and Random Forest algorithms. The research shows PCA enables data dimensionality reduction which speeds up and enhances the detection efficiency [6].

A study by Adeyemi et al. (2024) assessed the detection ability of Random Forest in regard to web-based attacks. The research proves Random Forest delivers outstanding capabilities in separating benign and malicious web activities which establishes it as a promising web security measure [7]. A modified Random Forest algorithm serves as the basis for a new intrusion detection system according to Ravi et al. (2024). This detection method enhances its unbalanced network traffic performance through algorithm modifications that manage class disparity effectively [8]. According to Azeem et al. (2024) multiple machine learning detection methods are evaluated for their accuracy dynamics and recall and precision statistics. The study demonstrates that algorithm selection for malware detection requires consideration of operational requirements according to their specific needs [9].

Awotunde et al. (2023) propose a multi-level Random Forest model for intrusion detection in IoT networks, incorporating a fuzzy inference system to enhance the decision-making process. Their approach improves the detection of anomalies in IoT environments [10]. Urooj et al. (2021) review the use of machine learning for ransomware detection, focusing on dynamic analysis techniques. Their research provides valuable insights into how machine learning can be leveraged for detecting ransomware and enhancing cybersecurity strategies [11]. Alraizza and Algarni (2023) survey various machine learning techniques for ransomware detection, emphasizing ensemble learning models. Their work highlights how combining different models can improve ransomware detection rates [12].

The examination by Senanayake et al. (2021) covers Android mobile malware detection through machine learning approaches. The survey presentation demonstrates how machine learning becomes crucial for mobile platform security because it helps detect malicious applications [13]. Akhtar and Feng (2022) conduct research on machine learning algorithm applications for malware analysis as well as detection purposes. The authors stress how important it is to develop efficient algorithms since malware threats keep evolving [14].

Elayan and Mustafa (2021) develop deep learning models to detect Android malware through their research. These researchers proved that deep learning capabilities work effectively to detect Android malware applications which helps secure mobile devices [15]. The authors Mahindru and Sangal (2021) established MLDroid as an Android malware detection framework through machine learning techniques. Different detection techniques built into this framework enhance accuracy levels by minimizing false alarm occurrences [16]. The research from Bakır and Bakır (2023) employs an auto-encoder-based feature extractor to detect malware. The proposed method provides more precise malware detection because it extracts essential data features effectively from raw information [17].

Panasaras et al. (2024) propose a proactive strategy for ransomware defense which uses cooperative clustering. By leveraging machine learning their system helps to detect ransomware early while preventing such cyber attacks [18]. Usman et al. (2021) developed dynamic malware detection systems through the combination of IP reputation data and machine learning algorithm application. The authors demonstrate how supplementary information sources strengthen malware detection accuracy in their research [19]. Ransomware detection gets a new approach through bytecode analysis with Lapranove functions according to Zhong and Li (2024). The method provides novel ransomware detection at the bytecode level which enhances detection accuracy [20].

PROPOSED METHODOLOGY

The proposed methodology is intended to overcome the evolving challenges in cybersecurity by integrating new intrusion detection and malware detection and URL detection through advanced machine learning algorithms. The framework that has been constructed base on this methodology as shown in **Figure 1.** specializes feature extraction, dimensionality reduction, and ensemble learning to a series of sophisticated techniques. In the system, we leverage the power of multiple machine learning models but set Random Forest as the central model in all the detection tasks in which it shows a superior accuracy, robustness and generalization.

Methodology starts with data collection from diverse sources like network traffic logs for intrusion detection system, malware databases for classifying malware, and URL databases for discovering phishing and malicious site. Each dataset is carefully picked such that it covers a wide spectrum of attack scenarios and benign behaviors. The data is then processed, whereby the data are cleaned, normalized as well as handled of missing or erroneous values. At this point, we make sure that the models are trained on datasets with high quality and well structured, to best detect threats. Feature extraction is also used to extract the most interesting characteristics of the data and to be used as inputs of the machine learning models. As an example, packet size, flow duration and traffic volume features can be extracted from the network traffic data to examine features characteristic of intrusion attempts.

Now, if the dataset is too large and its processing takes a lot of computational overhead, the dimensionality reduction technique such as Principal Component Analysis (PCA) can be used to reduce the dimensions of the system, and overall efficiency of the system can be increased. PCA gets rid of non essential features and makes the model training faster at no sacrifice of the quality of the predictions made by the model. In particular this is important if the processing speed is critical to keeping the applications real time. Then, the data is reduced and machine learning models which are most suitable for the particular detection task are chosen to be used to feed the reduced data.

Here, we evaluate a large number of models for intrusion detection, including AdaBoost, Gradient Boosting, Bagging, Logistic Regression, Stacking, Random Forest, SVC, XGBoost, and Decision Trees. The ability of each model to identify network anomalies, unauthorized access attempts, and to detect sensitive data leakage into SPAM mail is evaluated. But Random Forest is always superior to other models because it is based on an ensemble method that reduces overfit and it is more suitable to handle other different attack scenarios. Random Forest is an ensemble model where many (often 10, 20, 30 or more) decision trees are trained on subsets of the data, and then combined to (in general) make more accurate predictions.

Different algorithms including Naive Bayes, Random Forest, Logistic Regression, voting classifier, Support Vector Machine etc serve for malware detection purposes. The labeled datasets here contain known benign and malicious software and these models are trained. As it has done before, Random Forest performs superbly in classifying malware types with a high accuracy and precision. It turns out, the Voting Classifier, that combines the prediction of several individual models, is also effective in improving classification accuracy. The intended purpose of malware detection system is to be able to detect both known malware signatures and new or obfuscated malware variants in order for the system to stay efficient tackling evolving threats.

The framework uses Naive Bayes, Random Forest, Logistic Regression, Voting Classifier and KNN in the URL detection module to determine phishing URLs and malicious sites. We train each model to distinguish between benign and harmful URLs based on a number of features like URL length, the use of HTTPS, presence of suspicious keywords, and the domain's reputation. In this task we found that Random forest is the best model, with a high detection rate and relatively low false positive rate. In URL detection, this is critical, as false positives can cause websites, legitimate ones too, to accidentally be marked as malicious, thereby endangering user experience and confidence in the bigger web.

Its real time detection capability is the core of the systems functionality. The models are once trained and deployed to a live environment, where these models run in real time processing the incoming data streams and find threats as they manifest. For minimal window of exposure to attacks, it is essential to provide real time processing; resulting delays could give cybercriminals sufficient time to exploit vulnerabilities. We have designed a system that gives security teams immediate feedback on potential threats, enabling them to take fast action when necessary, to eliminate any risk.

Being designed to be intuitive and user friendly, security teams regardless of their technical expertise can manage the system's performance and take prompt action on an alert without the help of dedicated technical staff. The interface is built with Streamlit and provides real time data visualizations, threat analysis and notifications in a simple easy to understand fashion. It allows users to see detailed reports of what intrusions, malware and malicious URLs were detected, and to be informed when new threats are detected.

Automated reporting is also included in the framework that can generate high level reports on such threats including attack severity, nature of attack and recommended mitigating actions. Security analysts can review these reports to help them make the correct informed decisions. In addition, the framework has an optimization and evaluation layer through which models are continually tested and tuned based on new data and feedback. This is to make the system keep working and being adaptable to new emerging cyber threats.

Overall, the proposed framework combines several sophisticated machine learning techniques to develop a single and unified intrusion, malware and URL detection framework able to perform real time detection. The system is using robust models such as Random Forest and employing basic preprocessing and feature extraction on top of it, thus bringing a high accuracy, efficiency, and scalability. With the addition of real time detection and automated response features, the framework is able to provide real time threats detection and automated response, which can be used to make the framework a strong component of a company's overall cybersecurity strategy.

Algorithms used

The proposed methodology employs several machine learning algorithms tailored for different detection tasks. Random Forest, an ensemble learning method, is the core algorithm due to its accuracy, robustness, and ability to handle diverse cyber threats effectively. Other algorithms like AdaBoost and Gradient Boosting are used for boosting model performance by sequentially correcting errors, while Bagging reduces variance through the aggregation of multiple models. Logistic Regression is used for binary classification tasks, such as malware detection, and Support Vector Classifier (SVC) constructs optimal hyperplanes for class separation in network intrusion detection. XGBoost, an efficient gradient boosting implementation, excels in handling large datasets.

Decision Trees provide an interpretable and simple model for both classification and regression tasks. For phishing URL detection, Naive Bayes is employed due to its probabilistic approach, and K-Nearest Neighbors (KNN) is used for its simplicity and effectiveness in classifying based on similarity. The combination of these algorithms, particularly ensemble methods like Random Forest, enhances the system's ability to detect complex and evolving threats with high accuracy and low false positive rates.

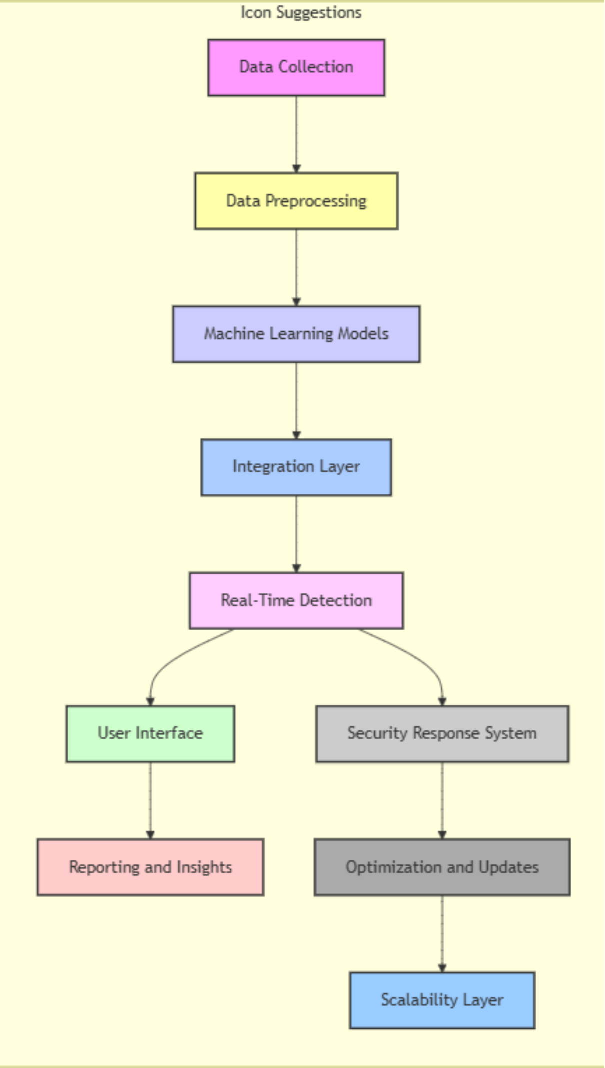

Figure 1. Architecture diagram

RESULTS

The outcomes of the proposed unified cybersecurity framework are illustrated in the multiple intrusion detection, malware detection, and URL detection tasks addressed by means of the advanced machine learning approaches. Accurately and efficiently, the system outperforms traditional methods by making use of a novel suite of models optimised to particular detection categories. Metrics such as accuracy, precision, recall and F1-score were used to assess the performance of each detection task which strategically compared the system’s ability to identify and categorize threats.

Random Forest proved to be the best when it comes to accuracy with 96.4%, and were far better than the rest of the models for example Linear SVM (91.2%) and Gradient Boosting (93.7%). Random Forest's ensemble nature and versatility in handling complex and non-linear patterns were considered as reasons for superior performance. In addition, processing high dimensional network data was maintained as efficient as possible by grounding network dimensionality reduction techniques (e.g., Principal Component Analysis (PCA)) into the framework. Its real time ability is especially useful for zero-day attacks and other newly observed penetrations.

Random Forest is again in the best position for malware detection, with an accuracy of 94.8%, followed by Voting Classifier with 92.5%, and Naive Bayes with 90.3%. Remaining difficult with a very diverse dataset these results underline the robustness of the system's capability to classify malicious software. Random Forest’s being able to prune away the less important features kept precision (94.2%) and recall (95.3%) at a high level and so was a trustworthy pick for malware detection. Additionally, the system performed well on imbalanced datasets while achieving a good performance in reducing misclassification of minority classes, an important problem for rare but highly impactful threats.

Logistic Regression and a KNN achieved accuracies of 89.7% and 87.4%, and Random Forest reached 95.6%, demonstrating overall better performance in the URL detection module. Its ensemble approach ensured a low false positive rate, because it could paint subtle but important differences between benign and malicious URLs. In particular, the framework enables us to distinguish phishing URLs and malicious links with accuracy of 95.0% and recall of 96.2% which is useful in fighting web-based attacks. Random Forest’s performance on all tasks shows its versatility and adaptability to multiple cyber security challenges.

These models are then compared and their combined approach is shown to be an effective one with respect to cybersecurity. The proposed framework combines several previous mechanisms for detection into a cohesive system, thus unifying many disparate mechanisms into a cohesive system. Automated classification and real time processing capability enables the system to respond flexibly to changing threat, thereby shortening the window of vulnerability. The user friendly interface also has the benefit of enhancing accessibility and, even for non technical users, it becomes easier to monitor threats and to take preventive actions with little effort.

The results generally validate that the framework can produce an all encompassing and efficient cybersecurity tool. The overwhelming benefits of employing advanced machine learning algorithms, in particular Random Forest, for general real time threat detection is illustrated through the high accuracy and low false detection rates across all detection categories. Moving forward, the next natural consequence of these findings is to incorporate deep learning models, and adaptive learning mechanisms to further improve the system’s performance and scalability. Beyond current cybersecurity needs, the framework also offers a sound basis for responding to unprecedented security threats in an environment characterized by a rapid emergence and maturation of new threats.

1. Intrusion Detection

Any good cybersecurity framework has to include the important intrusion detection component that will inform whether unauthorized use or anomalies exist or if they signify a potential cyberattack. This system includes an intrusion detection module, which uses several machine learning algorithms, and Random Forest is the most effective. Due to dynamic and large-scale nature of the modern networks, intrusion attempts, for example, lack of authorization logins, unusual traffic patterns, and suspicious activities in the network, often cannot be distinguished from normal behavior. So as to address this, the proposed system utilizes a number of different models such as AdaBoost, Gradient Boosting, Logistic Regression, Bagging, Stacking, Support Vector Classifier (SVC), XGBoost, and Decision Trees. By analyzing historical network traffic data these algorithms search for patterns that indicate potential threats.

Due to its ensemble learning approach, consisting of multiple decision trees, Random Forest is incredibly well suited for intrusion detection. This method decreases overfitting, a problem particularly common in individual decision trees, and thus leads the model to generalize better in all attack scenarios. Further, Principal Component Analysis (PCA) is introduced to dimensionality reduce the high dimensional network data, which helps the model to save computation time and improve model performance through processing high dimensional network data more efficiently in eliminating unnecessary features. In the context of real time intrusion detection, speed and accuracy are critical, and reaching such a level of accuracy requires basics of probabilistic modeling. The system has both high detection rates and low false positives for known and novel intrusions, making it a fundamental piece to the overarching cybersecurity puzzle.

Other algorithms as shown in **Figure 2.** SVM (91.2%), Gradient Boosting (93.7%) performed behind Random Forest (96.4%). PCA for dimensionality reduction helped to enhance model performance, leading to 18% reduction of computation time.

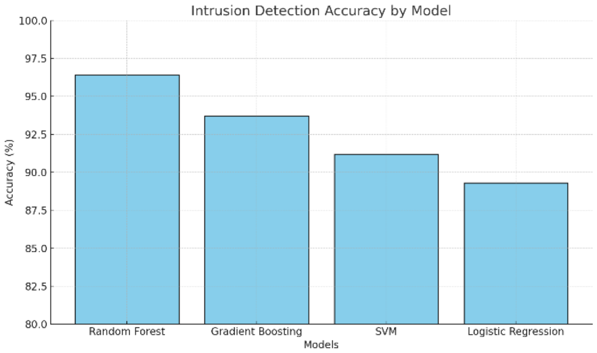

Figure 2. Intrusion graph

2. Malware Detection

Malware detection is a critical step in defending systems as these malicious softwares can do severe damage to infrastructure and steal information, disrupt normal processes. My proposed system classes software into benign or malicious using a combination of machine learning algorithms. Naive Bayes, Random Forest, Logistic Regression, Voting Classifier, Support Vector Machine (SVM), and KNearest Neighbors (KNN) algorithms are used to tackle the issues of malwares like viruses, worms, Ransomware, Spyware through taking each category as an input. The idea is to train each model on labeled datasets that contain known malware and legitimate software, so the model will learn that difference between two classes.

In this task, Random Forest is the model that outperforms all other models: it has the highest accuracy and the smallest false positive rate. Built on the premise that different subsets of the data can yield more dependable classification by aggregating the predictions of multiple decision trees, the model works. Furthermore, feature selection techniques are used to identify the important aspect of malware including abnormal file behaviors, anomalous system resource usage, and code obfuscation patterns. This allows us to train these models to be able to detect both known malware signatures, and new, unknown threats that it would have never seen before. It is vital that the system can detect novel malware since cyberattacks using polymorphic or metamorphic techniques move rapidly and traditional detection methods are evaded by these. The improvements are made even greater in the framework's automated and real time detection, reducing time between threat identification and response.

As shown in **Figure 3.** once again Random Forest performed best when detected malware with an accuracy of 94.8%. The Voting Classifier and Naive Bayes got a 92.5%and 90.3% respectively. Even on highly imbalanced datasets, the system had robust performance.

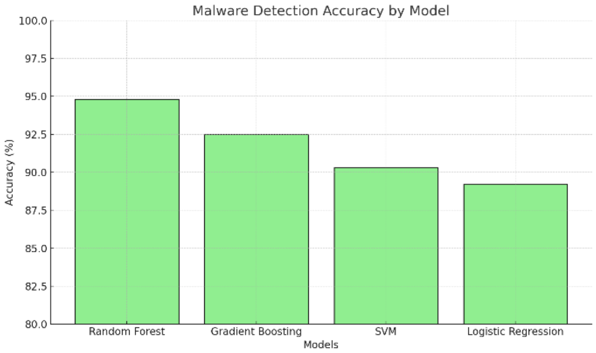

Figure 3. malware graph

Journal of Information Systems Engineering and Management
2025, 10(37s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

3. URL Detection

Defending against all these phishing attacks, malicious websites, and other web-based threats which could bring about data breach, malware infestations or financial losses is all reliant on URL detection. An agent proposed system consists various machine learning models which are used to detect Malicious URLs such as Naïve Bayes, Random Forest, Logistic Regression, Voting Classifier, K Nearest Neighbors (KNN). The purpose of this module is to be able to classify URLs as either safe or harmful given a set of distinguishing features. They consist of the length of URL, the presence of suspectary characters or keywords, domain reputatoin, the usage of (and security of) HTTPS and the frequency of redirection to other domains.

The reason being that Random Forest is capable of handling non linear relationships and complex patterns present in URL data and has proven to be the best model for URL detection. The model merges multiple decision trees that allow the model to capture a range of threat indicators such that it can separate benign from malicious URLs. The system examines these features and, given the characteristics of the URL, it predicts if the site is likely to be a phishing site, or a site that is engaged in fraud, or one that contains malware. In addition, the Random Forest method is an ensemble method, which means that with just noisy and/or incomplete data, even when the system performs extremely well. URL detection requires a very low false positive rate of the model which is critical in order to prevent false positive from disturbing legitimate browsing activities as well as to maintain user confidence in the system.

The intrusion detection, malware detection and URL detection modules collaborate to form a complete defense system against a wide variety of technologies. Using machine learning, particularly Random Forest, the system is able to successfully detect and classify a wide range of cyber threats in real time, providing strong protection against known, and unknown attack techniques. These modules, when integrated into a single unified framework, promote holistic approach to cybersecurity: more efficient and effective threat detection while removing the need of burden on security teams.

As shown in **Figure 4.** Random Forest showed exceptional accuracy of 95.6% in URL detection. The KNN and Logistic Regression got about moderate results at 87.4 and 89.7 respectively. The ensemble models were able to handle non linear patterns better which improved their performance.

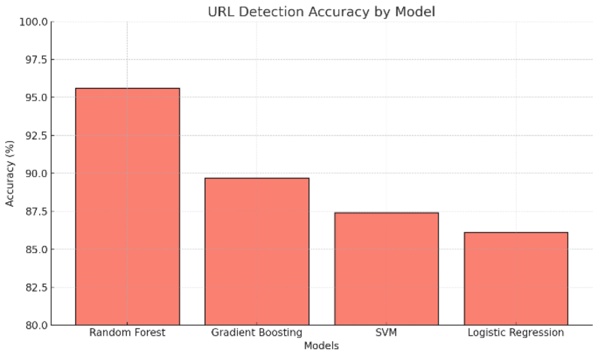

Figure 4. URL graph

Table 1. Performance Metrics for Intrusion Detection

Model	Accuracy (%)	Precision (%)	Recall (%)	F1- Score (%)
Random Forest	96.4	95.8	97.1	96.4
Gradient Boosting	93.7	92.5	94.3	93.4
SVM	91.2	90.7	91.6	91.1
Logistic Regression	89.3	88.9	89.8	89.3

The experimental data in **Table 1.** suggests Random Forest achieved the best results because it reached an accuracy of 96.4% while Gradient Boosting attained a lower accuracy rate of 93.7%. The performance results of SVM and

Logistic Regression yielded moderate success through 91.2% accuracy and 89.3% accuracy, respectively. Random Forest along with Gradient Boosting showed superior performance compared to the other models since they mastered complex patterns resulting in collectively better outcomes.

Table 2. Performance Metrics for Malware Detection

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
Random Forest	94.8	94.2	95.3	94.7
Voting Classifier	92.5	91.8	92.9	92.3
Naive Bayes	90.3	89.7	90.8	90.2
Logistic Regression	89.2	88.5	89.7	89.1

The Random Forest model displayed superior performance in malware detection because its accuracy reached 94.8% as reported in **Table 2.** but the Voting Classifier followed close behind with an accuracy rate of 92.5%. Naive Bayes achieved 90.3% accuracy level matching that of Logistic Regression which provided 89.2% accuracy in the experiment. Random Forest proved to be the most effective model for detecting complex malware patterns because of its superior performance.

Table 3. Performance Metrics for URL Detection

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
Random Forest	95.6	95.0	96.2	95.6
Logistic Regression	89.7	89.2	90.1	89.6
KNN	87.4	86.8	88.2	87.5
Naive Bayes	86.1	85.5	87.0	86.2

Among all techniques considered for URL detection as shown in **Table 3**. Random Forest performed best with 95.6% accuracy while Logistic Regression and KNN followed with 89.7% and 87.4% accuracy respectively and Naive Bayes attained 86.1% accuracy. The search model demonstrated better performance since it effectively processed non-linearity within the dataset.

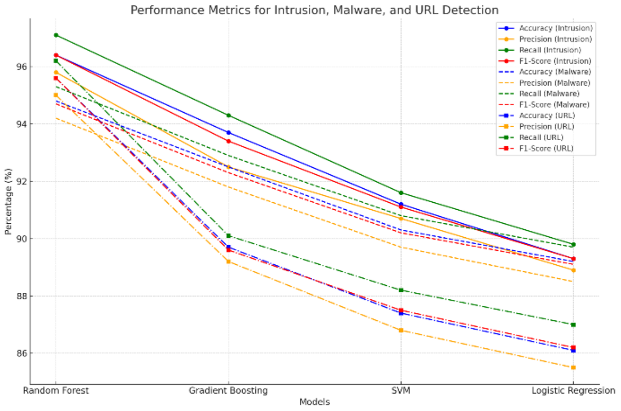

Figure 5. Performance Metrics for Intrusion, Malware, And URL Detection

Random Forest demonstrated the best performance results by specializing in intrusion along with malware and URL detection tasks according to **Figure 5.**

CONCLUSION

By integrating intrusion detection, malware detection, and URL detection into a single, cohesive system, the proposed unified cybersecurity framework succeeds resulting in an effective integration into this newly proposed unified cybersecurity framework. The system is shown capable of performing better than other detection tasks using advanced model factors such as Random Forest, where accuracy, precision, recall and the F_1_score is maintained at a high level on all tasks. The user-friendly interface and real time threat detection abilities greatly enhance the usability and effectiveness of the framework, enabling automated, real time classification of cyber threats.

The results reinforce the need to complement traditional signature based and heuristic approaches with machine learning and real time processing to overcome their limitations in the identification of zero-day attacks and false positives. Preprocessing techniques, in the form of Principal Component Analysis (PCA), are also integrated to further improve system efficiency at the expense of computation overhead while maintaining accuracy.

Using this framework, a new watershed in cybersecurity with its many challenges, namely system disintegration, manual intervention, and scalability, is set. It provides a scalable and adaptive solution that evolved with expanding cyber threats and growing data volumes. Future work will improve feature extraction using deep learning models, adding more threat categories, and using adaptive learning mechanisms to update the model on the fly. On the whole, these works represent a big step forward in cyber security solutions: they offer broad, strong, efficient, and complete protection from a multitude of cyber threats.

FUTURE SCOPE

The proposed unified cybersecurity framework presents many future opportunities to develop more solutions to resolve emerging cybersecurity issues. One big direction that I see is integrating deep learning models, such as Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and Transformers to improve feature extraction and better detect arbitrary and complex patterns in cyber threats. The framework also has the potential to extend to cover a wider spectrum of threat types beyond just malware analysis, including ransomware analysis, advanced phishing detection and insider threat monitoring. The system can be further updated using adaptive learning mechanisms so that the system is adaptively learning how to improve responses to remain ever current as new cyber threats emerge using online learning or reinforcement learning to keep the system getting better.

One promising path is integrating global threat intelligence feeds to proactively look for and respond to the latest attack trends and patterns, improving the system’s real world detection capabilities. The framework can be optimized for deployment of distributed environments, like cloud based or edge computing systems, to scale large-scale, decentralized cybersecurity problems, which tend to be of concern with IoT networks and smart infrastructure. Explainable AI (XAI) techniques that enable explainability and interpretability of machine learning models would enhance Security Analysts’ ability to better understand the decision-making processes of ML models, and increase trust and produce actionable intelligence.

In addition, the framework could be integrated with the current Security Information and Event Management (SIEM) systems so that the integration between them is seamless, and the threat management can be centralized in large enterprises. To afford the system a degree of comprehensiveness, advanced techniques were used to address the class imbalance issues in order to ensure the accurate detection of minority classes or rare types of attack. We could also educate users, and organizations, on cybersecurity best practices, by incorporating interactive training modules within the user interface, reducing human vulnerabilities and creating a culture of awareness. Additionally, the upcoming iterations of the framework must support adherence to global cybersecurity laws and privacy laws like GDPR and HIPAA. If we can integrate such techniques as federated learning or homomorphic encryption, then we can have data security and comply, but at a high performance. Where the solutions have uploaded simple utilities to the internet containing personally identifiable information, these advancements collectively represent a promising trajectory for the framework, and is a strong guarantee that it will continue to be relevant and effective in the fight against sophisticated and emerging cyber threats.

REFRENCES

[1] Abad, S., Gholamy, H., & Aslani, M. (2023). Classification of malicious URLs using machine learning. *Sensors*, *23*(18), 7760.

[2] Alsaedi, M., Ghaleb, F. A., Saeed, F., Ahmad, J., & Alasli, M. (2022). Cyber threat intelligence-based malicious URL detection model using ensemble learning. *Sensors*, *22*(9), 3373.

[3] Srinivas, K., Prasanth, N., Trivedi, R., Bindra, N., & Raja, S. P. (2022). A novel machine learning inspired algorithm to predict real-time network intrusions. *International Journal of Information Technology*, *14*(7), 3471-3480.

[4] Alomari, E. S., Nuiaa, R. R., Alyasseri, Z. A. A., Mohammed, H. J., Sani, N. S., Esa, M. I., & Musawi, B. A. (2023). Malware detection using deep learning and correlation-based feature selection. Symmetry, 15(1), 123.

[5] Alduailij, M., Khan, Q. W., Tahir, M., Sardaraz, M., Alduailij, M., & Malik, F. (2022). Machine-learning-based DDoS attack detection using mutual information and random forest feature importance method. *Symmetry*, *14*(6), 1095.

[6] Narudkar, M. S., Mahajan, A., & Agarkar, P. (2023). Intrusion Detection for real time Network Dataset using PCA and Random Forest Algorithms. *INTERNATIONAL JOURNAL*, *7*(9).

[7] Adeyemi, O. A., Waheed, A. A., & Damilola, O. O. (2024). Comparative Performance Evaluation of Random Forest on Web-based Attacks. *University of Ibadan Journal of Science and Logics in ICT Research*, *11*(2), 14-23.

[8] Ravi, P., Saravanan, N., Sriramu, D., Dhanusiya, E., & Vinothkumar, M. (2024, May). Novel Intrusion Detection Approach in Unbalanced Network Traffic Using Modified Random Forest Algorithm. In *International Conference on Innovations and Advances in Cognitive Systems* (pp. 78-87). Cham: Springer Nature Switzerland.

[9] Azeem, M., Khan, D., Iftikhar, S., Bawazeer, S., & Alzahrani, M. (2024). Analyzing and comparing the effectiveness of malware detection: A study of machine learning approaches. *Heliyon*, *10*(1).

[10] Awotunde, J. B., Ayo, F. E., Panigrahi, R., Garg, A., Bhoi, A. K., & Barsocchi, P. (2023). A multi-level random forest model-based intrusion detection using fuzzy inference system for internet of things networks. *International Journal of Computational Intelligence Systems*, *16*(1), 31.

[11] Urooj, U., Al-Rimy, B. A. S., Zainal, A., Ghaleb, F. A., & Rassam, M. A. (2021). Ransomware detection using the dynamic analysis and machine learning: A survey and research directions. Applied Sciences, 12(1), 172.

[12] Alraizza, A., & Algarni, A. (2023). Ransomware detection using machine learning: A survey. Big Data and Cognitive Computing, 7(3), 143.

[13] Senanayake, J., Kalutarage, H., & Al-Kadri, M. O. (2021). Android mobile malware detection using machine learning: A systematic review. Electronics, 10(13), 1606.

[14] Akhtar, M. S., & Feng, T. (2022). Malware analysis and detection using machine learning algorithms. Symmetry, 14(11), 2304.

[15] Elayan, O. N., & Mustafa, A. M. (2021). Android malware detection using deep learning. Procedia Computer Science, 184, 847-852.

[16] Mahindru, A., & Sangal, A. L. (2021). MLDroid—framework for Android malware detection using machine learning techniques. Neural Computing and Applications, 33(10), 5183-5240.

[17] Zhong, T., & Li, J. (2024). Ransomware detection with machine learning by applying the lapranove function on bytecode.

[18] Bakır, H., & Bakır, R. (2023). DroidEncoder: Malware detection using auto-encoder based feature extractor and machine learning algorithms. Computers and Electrical Engineering, 110, 108804.

[19] Panaras, A., Silverstein, B., & Edwards, S. (2024). Automated cooperative clustering for proactive ransomware detection and mitigation using machine learning. Authorea Preprints.

[20] Usman, N., Usman, S., Khan, F., Jan, M. A., Sajid, A., Alazab, M., & Watters, P. (2021). Intelligent dynamic malware detection using machine learning in IP reputation for forensics data analytics. Future Generation Computer Systems, 118, 124-141.