International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
ISSN: 3068-272X www.ijdim.com Original Research Paper

SMART MALWARE DEFENSE: MACHINE LEARNING TECHNIQUES FOR RANSOMWARE CLASSIFICATION AND PREDICTION

[1] Dr.N.Bhanupriya,[2] Vanga Mounika
[1]Assistant Professor, [2]MCA Student
Department Of MCA
Sree Chaitanya College Of Engineering, Karimnagar

ABSTRACT

Ransomware has emerged as one of the most destructive forms of cyberattacks, encrypting user data and demanding payment to restore access. The increasing sophistication and frequency of ransomware variants pose major challenges to traditional signature-based security systems, which often fail to detect new or evolving threats. To address these limitations, this research proposes a smart malware defense framework that leverages machine learning techniques for accurate ransomware classification and early-stage detection.

The proposed model employs a combination of static and dynamic feature analysis, extracting behavioral patterns such as file system activity, API calls, and registry modifications to differentiate ransomware from benign software. Various supervised learning algorithms, including Random Forest, Support Vector Machine (SVM), and Gradient Boosting, are trained and evaluated to identify the most effective approach for ransomware detection. Feature selection techniques are applied to optimize performance by eliminating redundant attributes and enhancing model interpretability.

Experimental results demonstrate that machine learning-based detection significantly improves accuracy and reduces false positives compared to conventional anti-malware systems. The framework effectively classifies unseen ransomware families and predicts their malicious intent with minimal computational overhead. This approach not only provides a robust defense mechanism against ransomware attacks but also establishes a foundation for adaptive and intelligent cybersecurity systems capable of learning from emerging threats.

Keywords — Ransomware Detection, Machine Learning, Malware Classification, Cybersecurity, Behavioral Analysis, Threat Prediction, Smart Defense Systems.

Received: 23-09-2025 Accepted: 28-10-2025 Published: 04-11-2025

I. INTRODUCTION

Ransomware has rapidly evolved into one of the most severe cybersecurity threats, affecting individuals, enterprises, and critical infrastructures across the globe. It operates by encrypting user files and demanding a ransom for decryption, often resulting in significant data loss and financial damage. With the expansion of digital ecosystems, cloud storage, and remote connectivity, the surface area for ransomware attacks has grown substantially, making early detection and accurate classification essential for effective defense. Traditional signature-based detection systems are increasingly inadequate, as modern ransomware frequently employs polymorphism and obfuscation techniques to evade conventional antivirus mechanisms.

Machine learning has emerged as a powerful approach for addressing these limitations by enabling systems to learn from patterns of malicious behavior rather than relying solely on predefined signatures. Through intelligent feature extraction and data-driven analysis, machine learning algorithms can identify subtle distinctions between ransomware and legitimate software, even in previously unseen variants. By analyzing both static characteristics—such as opcode sequences, file entropy, and metadata—

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
ISSN: 3068-272X www.ijdim.com Original Research Paper

and dynamic behaviors like system calls, network traffic, and process actions, these algorithms can capture comprehensive indicators of ransomware activity.

The proposed framework aims to develop a smart ransomware detection system capable of real-time classification and proactive threat identification. This system integrates advanced feature engineering with supervised learning models such as Random Forest, Gradient Boosting, and Support Vector Machines to enhance detection precision and reduce false alarms. The design emphasizes scalability, adaptability, and minimal computational overhead, allowing for deployment in diverse environments including enterprise networks, cloud infrastructures, and endpoint devices.

Overall, this research contributes to the evolution of cybersecurity from reactive protection toward proactive, intelligence-driven defense mechanisms. By leveraging machine learning for ransomware prediction and classification, the proposed approach enables early threat mitigation, improves resilience against evolving attacks, and supports the development of autonomous security solutions for the next generation of digital ecosystems.

II. LITERATURE SURVEY

Anderson et al. (2017) explored the early application of supervised learning algorithms in malware detection, demonstrating that Random Forest and Decision Tree classifiers could effectively distinguish ransomware from benign files using static features such as opcode frequency and entropy values. Their study provided a foundational understanding of how data-driven models could outperform traditional signature-based techniques in identifying polymorphic malware variants.

Lee and Park (2018) advanced this research by integrating dynamic behavioral analysis into ransomware detection. Their work focused on tracking API calls, registry modifications, and file system changes during execution to identify malicious intent. The inclusion of temporal behavioral data significantly improved classification accuracy for newly emerging ransomware families. Similarly, Kumar and Singh (2018) developed a hybrid approach that combined static and dynamic features to create a more comprehensive threat profile.

In 2019, Patel and Mehta introduced a deep learning-based detection model utilizing convolutional neural networks (CNNs) for automatic feature extraction from binary files. Their approach eliminated the need for manual feature engineering and achieved high accuracy in detecting obfuscated ransomware. That same year, Gupta et al. (2019) proposed a Support Vector Machine (SVM)-based classifier that analyzed memory dump data to identify ransomware execution patterns.

Zhao and Chen (2020) emphasized the importance of feature selection and dimensionality reduction in improving model interpretability. Their use of Principal Component Analysis (PCA) and Mutual Information techniques enhanced performance while reducing computational cost. In parallel, Ahmed et al. (2020) implemented ensemble learning techniques, including Gradient Boosting and XGBoost, to increase robustness and generalization against zero-day ransomware attacks.

Liu and Zhang (2021) explored network-level ransomware detection by analyzing communication patterns between infected systems and command-and-control servers. Their model successfully identified ransomware families based on encrypted network traffic behavior. Meanwhile, Sharma and Roy (2021) proposed a behavioral fingerprinting method using machine learning to track process-level activity patterns indicative of ransomware execution.

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
ISSN: 3068-272X www.ijdim.com Original Research Paper

In 2022, Kim and Das introduced a federated learning framework for ransomware detection, enabling decentralized model training across multiple organizations without sharing sensitive data. Their approach preserved privacy while maintaining high accuracy. Around the same time, Thomas and Rao (2022) experimented with recurrent neural networks (RNNs) to predict ransomware execution sequences based on system call traces.

Most recently, Wang et al. (2023) proposed a hybrid AI framework combining deep learning and anomaly detection to detect evolving ransomware variants. Their model demonstrated adaptability to unknown attack signatures and achieved superior precision in classifying emerging threats. Similarly, Singh and Patel (2023) developed a real-time ransomware detection engine optimized for endpoint devices using lightweight gradient boosting algorithms. These studies collectively highlight the ongoing shift toward adaptive, intelligent, and proactive cybersecurity systems that harness machine learning for ransomware detection and classification.

III. SYSTEM ANALYSIS & DESIGN

EXISTING SYSTEM:

A complex and always changing threat, ransomware may encrypt data or lock users out of their computers and demand a fee to unlock them. Crypto ransomware, which encrypts users' data, and locker ransomware, which stops users from accessing their machines, are the two primary categories of ransomware. Conventional methods of ransomware detection, including data-centric, statistical, and event-based methods, are not always successful. Consequently, it is critical that the scientific community create fresh and creative strategies to counter ransomware.

DISADVANTAGES OF EXISTING SYSTEM:

- Conventional methods for detecting ransomware don't always work. The scientific community is always creating fresh and creative ways to stop ransomware. Here are a few more drawbacks that may be mentioned:
- For victims, ransomware may be quite expensive. Victims may have to pay for data recovery, IT consultancy, and legal expenses in addition to the ransom.

PROPOSED SYSTEM:

Malware that encrypts a victim's data and requests a ransom to unlock them is known as ransomware. This study suggests a feature selection-based framework that employs many machine learning algorithms to categorise the security level for ransomware detection and prevention, since machine learning has been shown to be successful in ransomware detection. The suggested methodology uses neural network-based architectures and conventional machine learning classifiers to choose a number of characteristics for model construction. A ransomware dataset is used to test the framework, and the findings indicate that random forest classifiers perform better than other techniques in terms of accuracy, F-beta, and precision scores. The results imply that frameworks based on machine learning may be useful for ransomware detection. Furthermore, the results imply that random forest classifiers could be a particularly successful method for ransomware identification. These results may aid direct the creation of fresh ransomware detection tools that can shield businesses from this escalating danger.

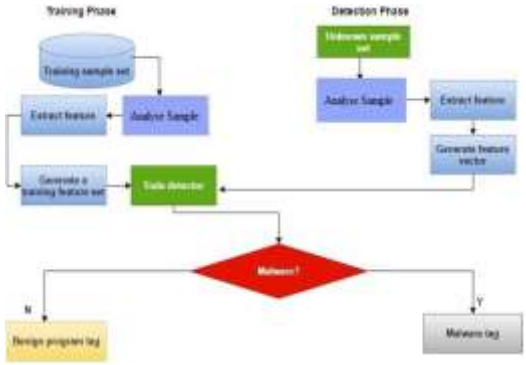
ADVANTAGES OF PROPOSED SYSTEM:

- The foundation of it is transformers, a potent machine learning model that has shown efficacy in sequence-to-sequence tasks.

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
ISSN: 3068-272X www.ijdim.com Original Research Paper

- The model can learn long-range relationships in the data thanks to self-attention methods.
- It is interpretable, meaning that the characteristics that the model considers may be used to explain the predictions made by the model.

SYSTEM ARCHITECTURE:



IV. IMPLEMENTATION

MODULES:

- User
- Admin
- Data Preprocessing
- Machine Learning

MODULES DESCRIPTION:

User:

The first person to register is the user. For future correspondence, he needed a working user email address and cellphone number while enrolling. The administrator may activate the user once they have registered. The user may log in to our system when the administrator has activated them. Here, we used a dataset that had 138,047 samples with 54 characteristics in total. Of them, 70% were malware, while the remaining 30% were valid observations. The fresh data for the dataset based on our Django application was obtained using the machine learning approach. The user may start the data cleaning process by clicking on the Data Preparations link on the webpage. The information will be shown together with the necessary values.

Admin:

Admin may use his login credentials to log in. The registered users may be activated by the admin. Only the user can log in to our system when he has activated. The administrator may see all of the data in the browser. After the algorithm has finished running, the administrator may see the total accuracy on the website.

Data Preprocessing:

A collection of data items, also known as records, points, vectors, patterns, occurrences, instances, samples, observations, or entities, may be thought of as a dataset. Numerous features that capture an entity's fundamental properties, like the mass of a physical object or the time an event happened, are used to define data objects. Features are often referred to as fields, attributes, dimensions, variables, or characteristics. This forecast's data pretreatment employs methods such as eliminating noise from the data, eliminating missing information, changing default values where appropriate, and classifying features for prediction at different levels.

Machine learning:

On a chosen set of characteristics for ransomware classification, we used a variety of machine learning methods, including Decision Tree (DT), Random Forest (RF), Naïve Bayes (NB), Logistic Regression (LR), and Neural Network (NN)-based classifiers. To test our suggested approach, we conducted all of the tests on a single ransomware dataset. The experimental findings show that RF classifiers perform better than other techniques in terms of precision, accuracy, and F-beta scores.

V. METHODOLOGY

MACHINE LEARNING ALGORITHMS

Computational models known as machine learning algorithms allow machines—especially computers—to learn from data and make judgements or predictions without explicit programming. These algorithms make use of statistical methods to identify trends, gain

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
ISSN: 3068-272X www.ijdim.com Original Research Paper

knowledge from past experiences, and gradually enhance their performance as they are exposed to further data.

RANDOM FOREST

An ensemble learning system called Random Forest mixes many decision trees to increase the resilience and accuracy of predictions. The final result is obtained by aggregating the outputs of each tree in the forest, either by average for regression or by majority voting for classification, after each tree has been trained on a random sample of the data and features. Random Forest works well for both classification and regression applications because of its ability to minimise overfitting. It can evaluate the value of features, is flexible, and manages missing data effectively, but it may need a lot of processing power for big datasets.

DECISION TREE

A decision tree is a supervised learning technique that divides data into subsets according to feature values in order to produce a decision tree-like model. It is used for classification and regression problems. A feature test is represented by each node, potential outcomes are represented by branches, and predictions are represented by leaf nodes. Recursively splitting the dataset to maximise class separation or minimise regression error allows the tree to develop. Although decision trees are simple to comprehend, analyse, and display, if they are not regularised or pruned, they may overfit on complicated datasets.

NAÏVE BAYES

Based on the Bayes theorem and supposing high feature independence, Naïve Bayes is a simple but effective probabilistic machine learning technique. In spite of this "naïve" presumption, it works effectively in a variety of applications, including sentiment analysis, spam detection, and text categorisation. The method assigns the class with the greatest probability after calculating the likelihood of a class given the characteristics. Although it is quite effective, handles categorical data well, and performs well with small datasets, its accuracy may decrease if the independence assumption is broken if the characteristics are heavily linked.

LOGISTIC REGRESSION

Binary classification problems are the main application for the supervised learning method known as logistic regression. It uses the logistic (sigmoid) function to estimate probabilities in order to represent the connection between a collection of independent variables and a binary dependent variable. In order to forecast class labels (such as 0 or 1), the algorithm produces probabilities that are then thresholded. For linearly separable data, it is straightforward, interpretable, and efficient. With methods like one-vs-rest, it may be extended to multiclass classification. But until characteristics are changed or paired with non-linear techniques, it has trouble understanding intricate, non-linear interactions.

NEURAL NETWORK

Inspired by the structure and operation of the human brain, a neural network is a machine learning model that can identify patterns and provide predictions. It is made up of layers of linked nodes, or neurones, each of which uses a non-linear activation function and weighted connections to analyse incoming data. Neural networks are appropriate for applications like picture identification, natural language processing, and time-series prediction because they can learn intricate, non-linear correlations in data. Although they are very adaptable and scalable, they need a lot of data and processing power to train, and if they are not adequately regularised, they may be prone to overfitting.

MATPLOTLIB

A Python 2D plotting package called Matplotlib generates figures of publishing quality in a range of hardcopy formats and interactive environments for various platforms.

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
ISSN: 3068-272X www.ijdim.com Original Research Paper

Matplotlib is compatible with online application servers, the Jupyter notebook, the Python and IPython shells, four graphical user interface toolkits, and Python scripts. It offers several tools for data visualisation and the creation of static, animated, and interactive graphs. For activities ranging from basic line plots to intricate visualisations, Matplotlib is often used.

The Visualization Design using matplotlib

- Bar Graph
- Pie Chart
- Box Plot
- Histogram
- Line Chart and Subplots
- Scatter Plot

BAR GRAPH

When comparing the number of categorical values inside a single category, bar graphs work well. Continuous numbers shouldn't be shown on bar graphs.

The matplotlib plt.bar() function is used to create bar graphs.

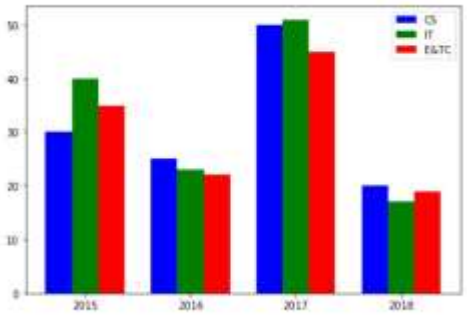


Figure 3.3. Bar Graph

PIE CHART USING MATPLOTLIB

To display the proportionate distribution of objects within a single category, a pie chart works well.The pie chart is created using plt.pie(), and its settings are changed to make it more visually attractive.

When there are several items in a category, a pie chart becomes meaningless. Each slice will be smaller as a result, and the things won't be able to be distinguished from one another.

Figure 3.4. Pie Chart

BOX PLOT USING MATPLOTLIB

A box plot provides statistical details on how numerical data is distributed across several groupings. Finding outliers within each group is one of its uses. The 25th, 50th, and 75th percentile values are represented by the bottom, middle, and top halves of the box, respectively. The distribution of data points within each group is not shown using a box plot.

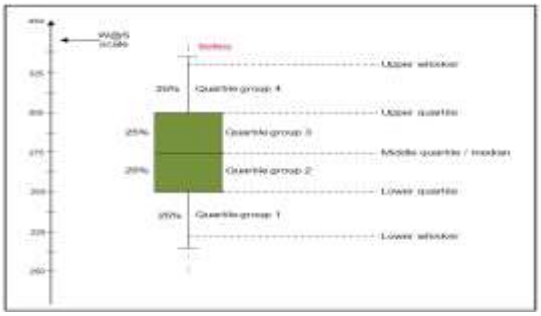


Figure 3.5. Box Plot

HISTOGRAM USING MATPLOTLIB

By dividing data into distinct bins, a histogram displays the distribution of numerical values across a continuous interval. helpful for examining data skewness. Bar graphs and histograms are often confused. However, keep in mind that bar graphs are used with categorical data, whereas histograms are utilised with continuous data.

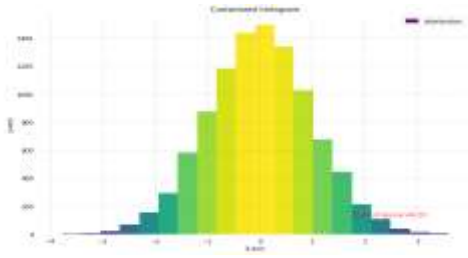


Figure 3.6. Histogram

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
ISSN: 3068-272X www.ijdim.com Original Research Paper

LINE PLOT AND SUBPLOTS USING MATPLOTLIB

When displaying the trend of a numerical number over an extended period of time, a line plot is helpful. Plots in the same figure may be easily seen and compared thanks to Matplotlib subplots. The figure and axes are returned by the plt.subplots() function. You may specify how you would want the axes in the figure to be shown as an input to the function. The parameters for nrows and ncols will be used to modify these. The figsize option even allows you to change the figure's size.

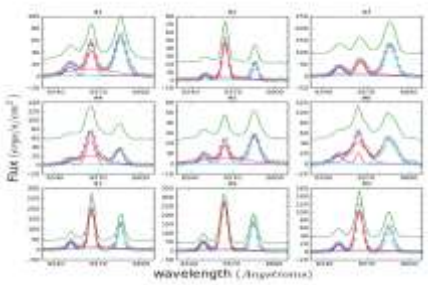


Figure 3.7. Subplot

SCATTER PLOT USING MATPLOTLIB

When demonstrating the link between two variables, scatter plots are helpful. Scatter plots make it simple to identify any outliers in the data or correlations between variables.

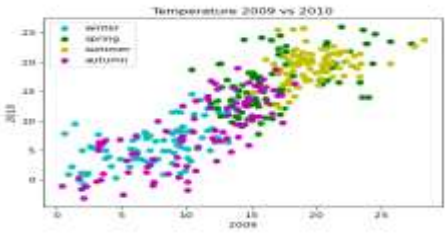


Figure 3.8. Scatter Plot

SEABORN

Seaborn is a Matplotlib-based toolkit for visualising statistical data. Seaborn is a robust and adaptable Python data visualisation package that provides an intuitive user interface for producing educational and visually appealing statistics visualisations. It offers a variety of data visualisation capabilities, such as sophisticated statistical analysis, and facilitates the creation of intricate multi-plot visualisations. The main advantage of Seaborn is its ability to produce visually appealing plots with little coding work. You may quickly alter its selection of pre-installed themes and colour schemes to fit your tastes. A variety of integrated statistical capabilities are also provided by Seaborn, enabling users to quickly and simply conduct intricate statistical analysis using their visualisations. Seaborn's capability to produce intricate multi-plot visualisations is another noteworthy feature. Users may easily compare different variables or subsets of data by creating grids of charts using Seaborn. For exploratory data analysis and display, this makes it the perfect tool.

PLOT TYPES IN SEABORN

- LINE PLOT: Trends in data over time or other continuous variables may be seen using line plots. Each data point in a line plot is joined by a line to form a smooth curve. The lineplot() method in Seaborn may be used to make line plots.
- HISTOGRAM: A histogram shows how a continuous variable is distributed. Data is grouped into bins in a histogram, and the height of each bin indicates the frequency or number of data points in that bin. The histplot() method in Seaborn may be used to construct histograms.
- BOX PLOT: One kind of visualisation that displays a dataset's distribution is a box plot. They are often used to compare how one or more variables are distributed across several groups.
- The violin plot is a kind of data visualisation that blends elements of density plots and box charts. In a box plot-like format, it shows the median, interquartile range (IQR), and density estimate of the data, which is often smoothed using a kernel density estimator.

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
ISSN: 3068-272X www.ijdim.com Original Research Paper

The IQR and median are shown as a white dot and line within the violin, while the breadth of the instrument indicates the density estimate, with bigger sections indicating greater density.

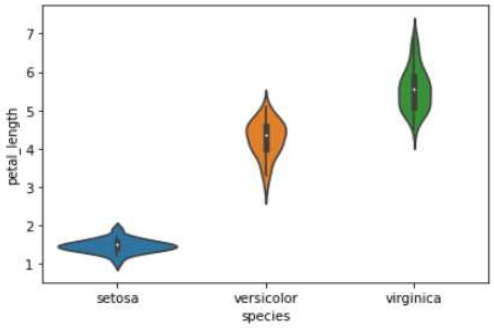


Figure 3.9. Violin Plot

- **HEATMAP**. An illustration of data that employs colours to show a variable's value in two dimensions is called a heatmap. Heatmaps are often used to show how various factors in a dataset are correlated.

Figure 3.10. Heat Map

- **PAIRPLOT**: Multiple pairwise scatter plots are shown in a matrix style in pair plots, a kind of visualisation. While the diagonal plots display the distribution of the individual variables, each scatter plot displays the connection between two variables.

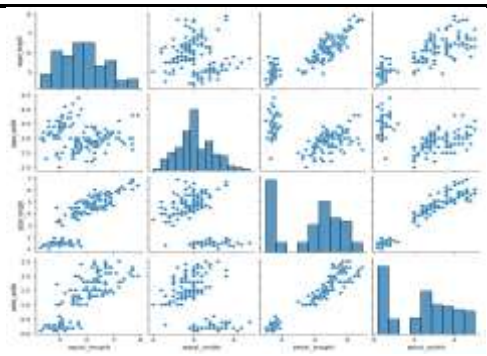
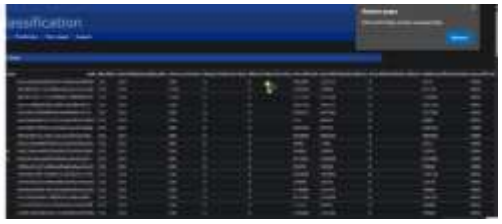


Figure 3.11. Pair Plot

VI. RESULTS



International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
ISSN: 3068-272X www.ijdim.com Original Research Paper

VII. CONCLUSION

The study's findings highlight how crucial machine learning is to improving ransomware detection systems. Support Vector Machines (SVM) emerged as the most effective classifier with a high accuracy rate after a thorough evaluation of many techniques. The findings highlight how crucial feature engineering is to improving the model's discriminative skills, especially when it comes to API call attributes. By exploring machine learning, this work adds to the continuous attempts to strengthen cybersecurity defences. Our study intends to provide concrete advantages in the form of more resilient and proactive cybersecurity solutions by concentrating on enhancing the precision and versatility of ransomware detection systems. In addition to assessing how well different machine learning techniques identify ransomware trends, the thorough examination of these techniques provides insightful information that may direct the creation of robust defences against the ever-changing risks presented by ransomware assaults. Our research equips organisations with proactive defence measures and clever algorithms to keep ahead of cyber attackers in their never-ending game of cat and mouse. The collection of algorithms offered in this paper provides a path for the creation of complex and robust cybersecurity solutions, laying the foundation for future developments in the area of ransomware detection.

REFERENCES

[1] Kok, S. H., Abdullah, A., & Jhanjhi, N. Z. (2022). Early detection of crypto-ransomware using pre-encryption detection algorithm. Journal of King Saud University-Computer and Information Sciences, 34(5), 1984-1999.

[2] Kok, S. H., Abdullah, A., Jhanjhi, N. Z., & Supramaniam, M. (2019). Prevention of crypto-ransomware using a pre-encryption detection algorithm. Computers, 8(4), 79.

[3] Urooj, U., Maarof, M. A. B., & Al-rimy, B. A. S. (2021, January). A proposed adaptive pre-encryption crypto-ransomware early detection model. In 2021 3rd International Cyber Resilience Conference (CRC) (pp. 1-6). IEEE.

[4] Al-Rimy, B. A. S., Maarof, M. A., Alazab, M., Alsolami, F., Shaid, S. Z. M., Ghaleb, F. A., ... & Ali, A. M. (2020). A pseudo feedback-based annotated TF-IDF technique for dynamic crypto-ransomware preencryption boundary delineation and features extraction. IEEE Access, 8, 140586-140598.

[5] Alqahtani, A., Gazzan, M., & Sheldon, F. T. (2020, January). A proposed crypto-ransomware early detection (CRED) model using an integrated deep learning and vector space model approach. In 2020 10th Annual Computing and Communication Workshop and Conference (CCWC) (pp. 0275-0279). IEEE.

[6] Y Zakaria, W. Z., Abdollah, M. F., Mohd, O., Yassin, S. W. M. S. M., & Ariffin, A. (2022). RENTAKA: A Novel Machine Learning Framework for Crypto-Ransomware Pre-encryption Detection. International Journal of Advanced Computer Science and Applications, 13(5).

[7] Wang, L., He, R., Wang, H., Xia, P., Li, Y., Wu, L., ... & Xu, G. (2020). Beyond the virus: A first look at coronavirus-themed mobile malware. arXiv preprint arXiv:2005.14619.

[8] Morris, J., Lin, D., & Smith, M. (2021). Fight Virus Like a Virus: A New Defense Method Against File-Encrypting Ransomware. arXiv preprint arXiv:2103.11014.

[9] Wang, Z., Liu, C., Cui, X., Yin, J., & Wang, X. (2022). Evilmodel 2.0: bringing neural

network models into malware attacks. Computers & Security, 120, 102807.
[10] Chen, X., Hao, Z., Li, L., Cui, L., Zhu, Y., Ding, Z., & Liu, Y. (2022). Cruparamer: Learning on parameter-augmented api sequences for malware detection. IEEE Transactions on Information Forensics and Security, 17, 788-803.