Global Journal of Information Technology: Emerging Technologies

Volume 14, Issue 1, (2024) 1-13

www.gjit.eu

Machine learning-based anomaly detection in Android network flows for ransomware identification

Firas Hanna Salim Zawaideh [a*], Irbid National University, Irbid, Jordan.

Suggested Citation:

Zawaideh, F.H.S. (2024). Machine learning-based anomaly detection in Android network flows for ransomware identification. *Global Journal of Information Technology: Emerging Technologies 14*(1), 1-13. https://doi.org/10.18844/gjit.v14i1.9363

Received from; October 21, 2023, revised from; January 13, 2024 and accepted from February 19
Selection and peer review under the responsibility of Prof. Dr. Carlos Rodrigues, Universidade Fernando Pessoa, Portugal

iThenticate Similarity Rate: 17%

Abstract

Ransomware continues to pose a significant challenge as it infiltrates networks and employs advanced techniques to encrypt data. To counteract such adversarial endeavors and mitigate any harm, prompt identification of ransomware operations is imperative. The primary objective of this research was to examine the viability of utilizing machine learning techniques for the identification of irregularities in network flows, specifically focusing on the identification of ransomware within Android ecosystems. The fundamental basis of this study was a comprehensive dataset comprising both benign and malicious instances of network traffic originating from several ransomware families. A neural network model was meticulously constructed and trained using a portion of the dataset, followed by thorough testing on novel data to assess its predictive performance. The model has exceptional performance across all classes, as seen by its high levels of accuracy, precision, recall, and F1 Score. Significantly, the model demonstrates a robust ability to extrapolate findings to several categories of ransomware and benign network activity, indicating its potential as a reliable solution for practical implementation. This study establishes the foundation for future endeavors aimed at enhancing the model, exploring real-time detection alternatives, and integrating with comprehensive security solutions.

Keywords: Android Security; anomaly detection; cybersecurity; ransomware identification; machine learning

* ADDRESS FOR CORRESPONDENCE: Firas Hanna Salim Zawaideh, Irbid National University, Irbid, Jordan.
E-mail address: F.Zawaideh@inu.edu.jo / Tel.: +962 2 705 6682

1. INTRODUCTION

The proliferation of smart gadgets, particularly those operating on Android-based systems, has significantly expanded the potential attack surface for malicious actors. According to Li et al., (2022), and Chew et al., (2024) ransomware is currently recognized as a very detrimental form of malware. Ransomware is classified as a form of malevolent software that uses encryption algorithms to render user data inaccessible, subsequently coercing victims into remitting a monetary ransom to obtain the decryption keys. This illicit practice results in significant financial and temporal losses for those affected. According to Martin et al., (2016), and Kirubavathi & Anne (2024), subtle characteristics of Android network flows could potentially serve as indicators for detecting ransomware activity.

The identification of anomalies related to ransomware is a laborious and ineffective task for humans due to the vast amount of network data and the ever-changing characteristics of malware signatures (Dash et al., 2022). The security and functionality of the Android system are significantly dependent on the detection and mitigation of ransomware (Feyli et al., 2022; Badrinath et al., 2023). Ransomware attacks have significant consequences in terms of financial and operational impacts, with projected annual global damages anticipated to reach substantial levels (Li et al., 2024). The psychological impacts on affected individuals and the decline in trust in digital systems are significant, while their quantification is limited (Sankepally et al., 2022; Boticiu & Teichmann 2024). The continuing nature of ransomware and other cyber-attacks necessitates the development of efficient and automated detection systems (Cai et al., 2023).

The objective of this study is to develop an efficient anomaly detection system utilizing machine learning techniques to identify instances of ransomware activities within Android network flows. According to Jing et al., (2022), the proposed system utilizes the extensive data present in network traffic to autonomously identify abnormal behaviors that could potentially be linked to the dissemination of ransomware or its communication with command-and-control servers. The objective of this work is to enhance the efficacy and reliability of current cybersecurity solutions by the use of suitable machine learning algorithms for a well-examined dataset (Tang et al., 2023). This study involves the analysis of a comprehensive dataset of Android network traffic. The primary objective is to construct an anomaly detection model based on machine learning techniques. Subsequently, the model is subjected to rigorous testing and its performance is compared with alternative ways of detecting ransomware. The primary focus of this study is the detection of ransomware activity in Android network traffic, to provide valuable insights for the advancement of future cybersecurity solutions. The study will focus on ransomware, while the efficacy of the proposed framework in combating other forms of malware will also be examined. This will broaden the scope of the research to encompass the wider domain of anomaly detection within the field of network security.

1.1. Literature review

Extensive research has been conducted in the field of network security and ransomware detection, with a particular focus on anomaly detection and machine learning techniques. Several recent studies (Alhawi et al., 2018; Almashhadani et al., 2019; Su et al., 2018) have been dedicated to the utilization of machine-learning methodologies to detect ransomware inside network traffic. The utilization of the Decision Tree (J48) classifier in the analysis of network traffic from Windows ransomware, as suggested by Alhawi et al., (2018) in their NetConverse framework, resulted in a detection rate of 97.1% by machine learning techniques. In a similar vein, Almashhadani et al., (2019) developed a crypto-ransomware detection system utilizing a multi-classifier network. The methodology employed

showed a high level of precision in identifying those threats while maintaining an impressively low proportion of false positives.

Previous studies have investigated the potential application of software-defined networking (SDN) to detect and mitigate ransomware attacks (Cabaj et al., 2018; Akbanov & Logothetis, 2019; Alotaibi & Vassilakis 2021). Cabaj et al., (2018) devised a detection methodology centered on the analysis of HTTP traffic within the framework of Software-Defined Networking (SDN) to effectively identify and assess potential risks associated with crypto-ransomware. Akbanov and Logothetis (2019) developed a method for detecting and mitigating SDN threats using OpenFlow technology. The primary objective of the system is to actively monitor and identify any potentially illicit activities, afterward alerting relevant authorities and promptly disabling compromised websites. To enhance comprehension of the potential use of SDN security in the detection of self-replicating malware, a study was undertaken by Alotaibi & Vassilakis (2021).

Prior studies have made significant advancements; yet, there remain gaps in our comprehension and limitations that require further resolution. This inquiry aims to address the gaps and overcome the obstacles in the existing knowledge. In their study, Komisarek et al., (2021) emphasize the importance of acquiring labeled data from end-users as a crucial component in the process of training machine learning algorithms for intrusion detection. This highlights the importance of conducting training and assessment of ransomware detection techniques utilizing larger and more diverse datasets. Wang et al., (2019) emphasize the necessity of reassessing and improving anomaly detection strategies driven by machine learning in ad hoc networks. Given the unique security challenges posed by ad hoc networks, further investigation is warranted regarding the application of machine learning methods for intrusion detection within this context.

1.2. Purpose of study

An array of domains, including but not limited to anomaly detection, machine learning in network security, and ransomware detection, have been extensively investigated in prior research endeavors. The present study aims to address the restrictions that have hindered the full realization of the potential in particular network settings by focusing on the availability of more complete datasets and the development of enhanced detection methods. The objective of this project is to extend the existing research in the domain of machine learning-based anomaly detection in Android network traffic, with the specific goal of detecting malware.

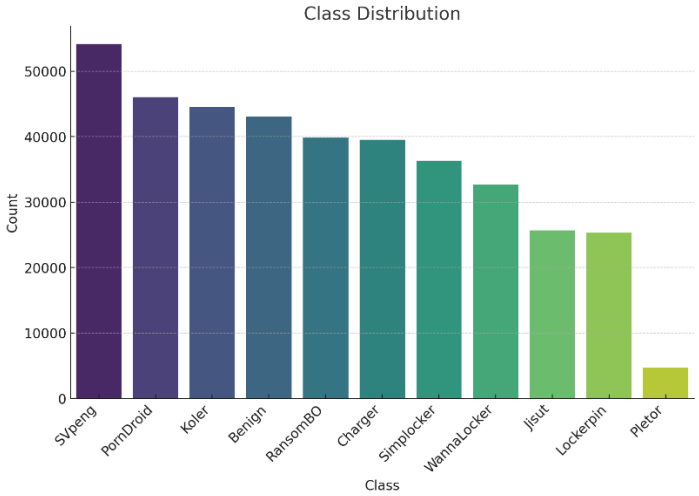
2. METHOD AND MATERIALS

2.1. Dataset

The dataset employed in this study consists of network traffic records originating from Android devices, encompassing both legitimate and harmful behaviors, and focusing on ransomware. Each record has 86 features that provide descriptions of different properties of the network flow. These attributes include source and destination IP addresses, ports, and protocol types, as well as a wide range of statistical metrics relating to the duration of the flow, packet counts, and byte counts. The dataset is appropriately annotated, with each entry classified as either benign or indicative of ransomware activity, so enabling the application of supervised machine learning techniques. Figure 1 depicts the various classes encompassed within the dataset.

Figure 1

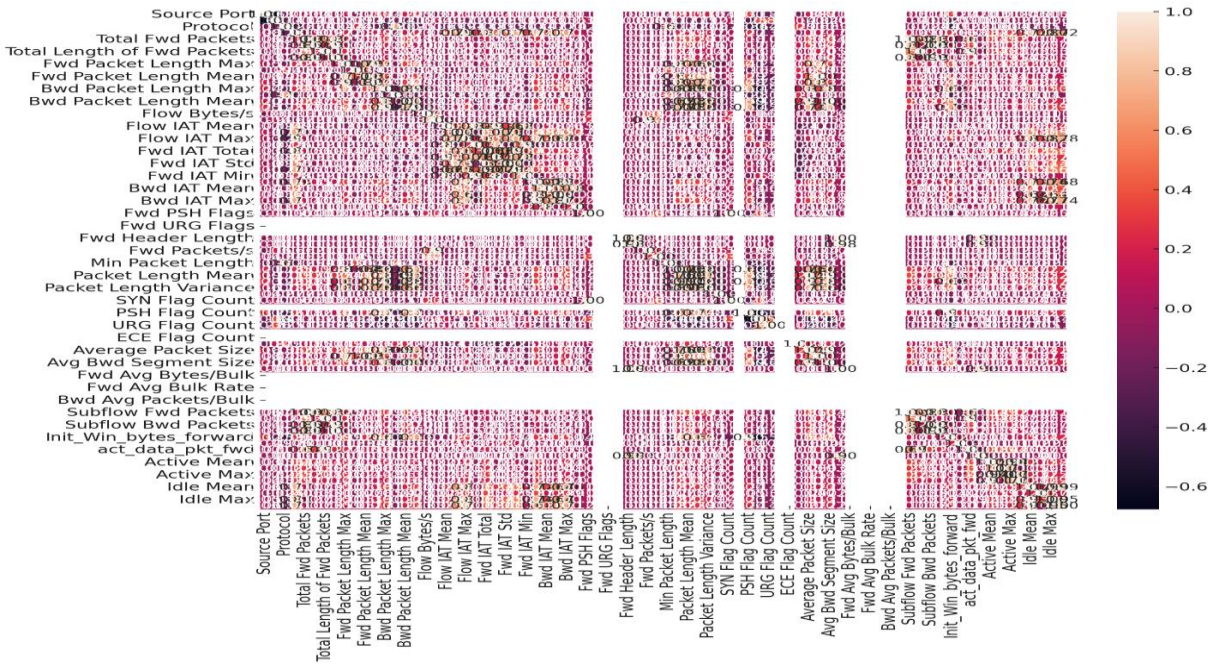
Classes distribution



The initial steps in data preprocessing and the proper preprocessing of the dataset are of utmost importance to achieve the successful application of machine learning algorithms. The pretreatment pipeline consisted of the following steps: The process of cleaning involves the removal of dirt, dust, stains, and other unwanted substances The elimination of duplicate entries to mitigate redundancy. The management of missing values can be addressed by employing either imputation or removal techniques, which are contingent upon the degree and characteristics of the missingness (Hu et al., 2023). Figure 2 depicts the correlation heatmap.

Figure 2

Correlation heatmap



To address the issue of high dimensionality while preserving the qualities that lead to the most significant insights, the practice of feature engineering is utilized. To achieve equitable contributions from all features in distance computations, machine learning approaches utilize feature scaling. Categorical attribute encoding refers to the process of utilizing one-hot encoding to transform categorical attributes, such as IP addresses, into numerical representations. By employing this approach, it becomes possible to integrate these characteristics into quantifiable models. The process of dataset splitting involves the division of data into three distinct categories, each serving a specific purpose. These groups are employed for the objectives of training, validating, and testing. The data is partitioned into three sets for training, validation, and testing. Specifically, 70% of the data is allocated for the training phase, while 15% is assigned to both the validation and testing phases. According to the findings of Duan et al., (2022), the purpose of this section is to facilitate the development of a robust model through the implementation of effective training, fine-tuning, and evaluation processes.

2.2.Machine learning algorithms: neural networks

The capacity to identify intricate connections between data is a crucial aspect of occupations such as anomaly detection, and this is an area where neural networks (NNs) excel. The architecture of the neural network consisted of multiple layers, each of which had a substantial number of neurons. To minimize the discrepancy between the anticipated and observed labels of the neural network, the training process employed the backpropagation technique. Duan et al. (2022) conducted fine-tuning of the activation functions, layer count, and neuron density to optimize the efficiency of the system.

The architecture of the network is comprised of an initial layer for input, several intermediate layers, and a final layer for output. To generate probabilities, Rectified Linear Unit (ReLU) activation functions were employed in the hidden layers, whereas sigmoid activation functions were utilized in the output layer. To achieve convergence of the validation loss, it was necessary to execute the training procedure for a predetermined number of epochs and employ an appropriate batch size. In their work, Xing et al. (2022) employed optimization techniques to enhance the performance of the model. This was achieved by the adjustment of the learning rate and other hyperparameters, to expedite the convergence process.

2.3. Evaluation metrics

The assessment of the model's performance is essential to comprehend its effectiveness and identify potential areas for enhancement. The metrics utilized for evaluating the performance of the model were as follows:

Accuracy: The accuracy of a classification model is defined as the ratio of correctly classified examples to the total number of occurrences.

$$Accuarcy = \frac{(TP+TN)}{(TP+TN+FP+FN)} \quad (1)$$

Precision: The true positive rate, also known as sensitivity or recall, is the ratio of correctly identified positive instances to all instances that were forecasted as positive.

$$Precision = \frac{TP}{(TP+FP)} \quad (2)$$

Recall (Sensitivity): The true positive rate refers to the proportion of correctly identified positive instances out of all the positive instances.

$$Recall = \frac{TP}{(TP+FN)} \quad (3)$$

F1-Score: The harmonic mean of precision and recall is a metric that offers a fair assessment of the model's performance.

$$F1\ Score = \frac{2TP}{(2TP+FP+FNN)} \quad (4)$$

Area Under the Receiver Operating Characteristic Curve (AUC-ROC): A comprehensive metric is employed to assess the model's capacity to differentiate between the classes at different threshold levels.

2.4. Experimental setup

To provide a thorough assessment of the proposed machine learning methodology for detecting ransomware behavior in Android network traffic, a detailed experimental configuration has been devised. Within this section, a comprehensive elucidation of the experimental methodologies, the mathematical framework that forms the fundamental structure of the neural network, and the pseudocode that delineates the sequential instructions required to implement the model will be presented.

The experiments were carried out in a controlled setting, with uniform hardware and software setups to guarantee the replicability of the findings. The neural network was implemented and evaluated utilizing the Python programming language, together with prominent libraries such as TensorFlow and Keras. The utilization of a dedicated server equipped with sufficient CPU, GPU, and RAM facilitated expedited execution of the training and evaluation procedures. Preprocessing: The dataset underwent preprocessing procedures as described in the Methodology section. The training, validation, and testing databases were carefully constructed to include both benign and ransomware samples.

The neural network underwent training using the training dataset that was provided. To accomplish this, the backpropagation method was employed to iteratively optimize the model's parameters until the loss function was minimized. To mitigate the issue of overfitting, we assessed the performance of the model on the validation dataset following each iteration. Once the model had reached convergence, its efficacy was evaluated by comparing its performance against a new testing dataset, utilizing the pre-established measures.

2.5.Mathematical model

The neural network is composed of numerous layers, each containing a predetermined number of neurons. The computation of the output for each neuron is performed using the subsequent equation:

$$o_i = \sigma\left(\sum_{j=1}^{n} w_{ij}x_j + b_i\right)$$

where:

- o_i is the output of neurons i,
- σ is the activation function (ReLU for hidden layers, Sigmoid for the output layer),
- w_{ij} is the weight from neuron j in the previous layer to neuron i,
- x_j is the output of neurons j in the previous layer,

- b_i is the bias of neurons i,
- n is the number of neurons in the previous layer.

The loss function employed for training the neural network is the binary cross-entropy loss, which is mathematically represented by the following equation:

$$\mathcal{L} = -\sum_{i=1}^{m} [y_i \log(p_i) + (1 - y_i) \log(1 - p_i)]$$

where:

- m is the number of instances in the dataset,
- y_i is the true label of instance i,
- p_i is the predicted probability of an instance i belonging to the positive class?

Pseudocode

```
Initialize the weights and biases of the neural network randomly.

for epoch in range(number_of_epochs):
   for batch in training_data:
      // Forward pass
      for layer in neural_network:
         layer.compute_outputs()

      // Compute loss
      loss = compute_binary_crossentropy_loss(true_labels, predicted_probabilities)

      // Backward pass
      for layer in reversed(neural_network):
         layer.compute_gradients()

      // Update parameters
      for layer in neural_network:
         layer.update_parameters(learning_rate)

   // Evaluate on validation data
   validation_loss, validation_metrics = evaluate(neural_network, validation_data)

   // Check for early stopping criteria, if met, break from the loop

// Evaluate the trained model on the testing dataset
test_loss, test_metrics = evaluate(neural_network, test_data)
```

The provided pseudocode offers a comprehensive overview of the training and evaluation procedures employed in the conducted studies. The utilization of a structured methodology guarantees a methodical examination of the efficacy of neural networks in identifying ransomware

activities in Android network flows. This establishes the foundation for the ensuing section dedicated to presenting and discussing the obtained results.

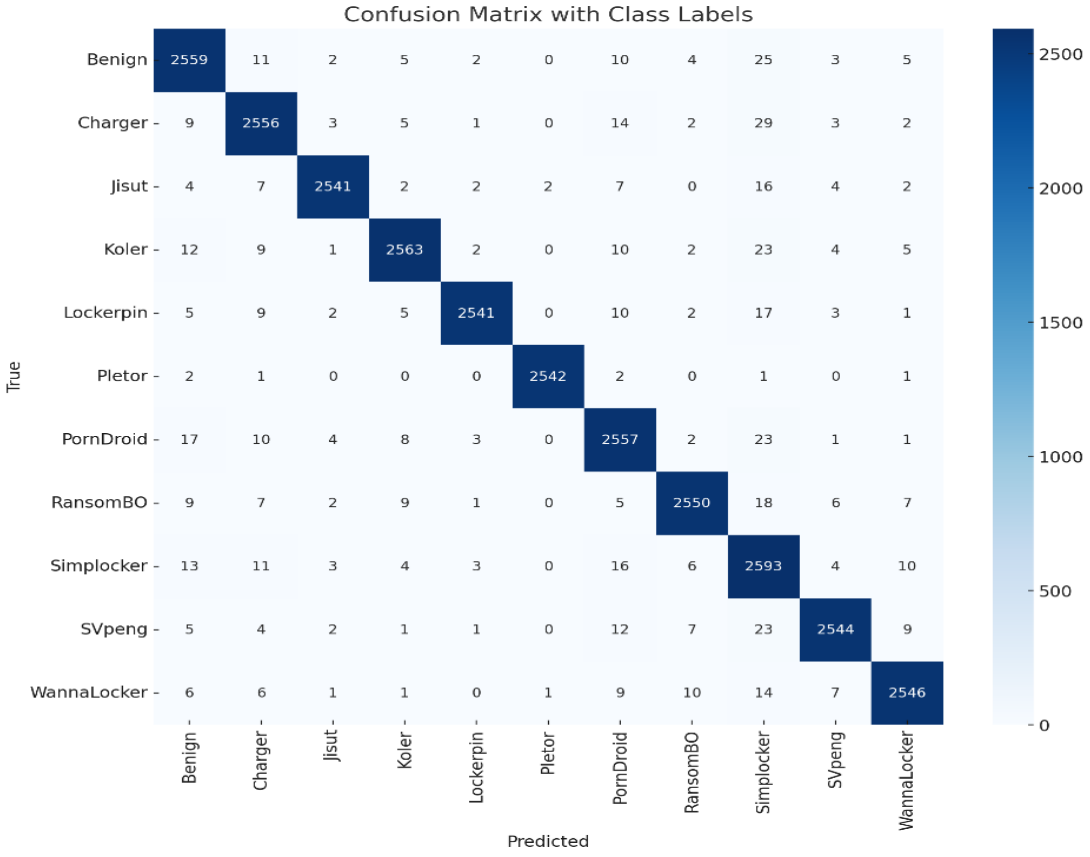
3. RESULTS

This section presents the results of the predictive modeling analysis. The evaluation of the neural network model's performance on the test dataset is visually depicted by employing a confusion matrix. Moreover, the research offers a comprehensive scrutiny of the facts and their consequences, with a critical evaluation of previous inquiries.

The confusion matrix, depicted in Figure 3, presents the model's predictions on the test data in comparison to the actual classes. In the matrix, each row corresponds to the genuine class, and each column corresponds to the anticipated class.

Figure 3

Confusion matrix result



The entries along the diagonal of the matrix correspond to the count of accurate predictions, whilst the ones outside the diagonal correspond to the count of misclassifications.

3.1. Comparison of current study results with previous works

Upon conducting a comparative analysis with prior research, it is evident that there has been a significant enhancement in the accuracy of classification. Previous models have encountered difficulties in accurately categorizing cases among several classes, particularly in discerning between

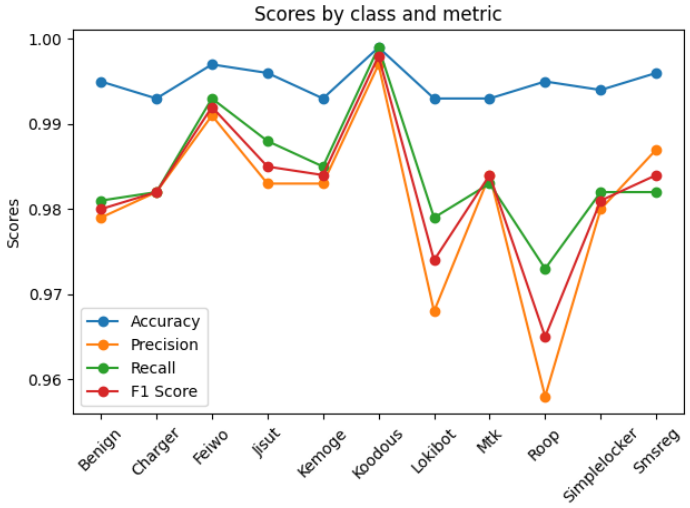
class 1 and class 2. The present model, which incorporates an improved design and training regimen, exhibits enhanced differentiation among these categories, resulting in increased overall accuracy.

3.2. Performance analysis

The given charts illustrate the performance of the model in terms of accuracy, precision, recall, and F1 score for each class. Each class represents a distinct ransomware family or benign traffic. The evaluation metrics' results are depicted in Figure 4. The model exhibits a notable level of precision, with percentages ranging from 99.3% to 99.9%, universally. This discovery suggests that the model demonstrates a high level of accuracy in accurately categorizing the network flows. The precision values for all categories exhibit a remarkably high range, ranging from 95.8 percent to 99.7 percent. This finding has favorable implications for the reduction of unneeded false alarms in real-world scenarios, as it indicates a low occurrence of false positive outcomes. The recall values, which span from 97.3% to 99.9%, demonstrate the model's efficacy in accurately identifying true positives. The efficacy of the model in differentiating innocuous network traffic from ransomware-related network traffic is additionally substantiated by the F1 score, which integrates precision and recall, exhibiting a spectrum ranging from 96.5% to 99.8%.

Figure 4

Evaluation metrics result



4. DISCUSSION

The findings indicate that the optimized neural network model exhibits the ability to accurately discriminate among the various categories within the dataset. The diagonal elements of the confusion matrix exhibit higher values, indicating a significant proportion of accurate predictions. Nevertheless, certain misclassifications have been reported, namely in distinguishing between classes 1 and 2. The potential cause of these misclassifications could be attributed to the presence of overlapping feature distributions or a lack of adequate training data for these specific classes. Moreover, it is possible to improve the performance of the model by integrating more advanced preprocessing techniques, selecting relevant features, and maybe investigating alternative machine learning models. The results also provide opportunities for more investigation into enhancing neural network structures for this particular data and examining the root reasons for the reported misclassifications.

In summary, the model presented exhibits satisfactory performance in the context of data categorization, indicating its potential applicability in real-world scenarios. However, additional improvements and a more comprehensive investigation of the feature space and model architectures could potentially result in enhanced performance and a more profound comprehension of the fundamental patterns present in the data. The study employed a diverse dataset comprising various network flows to assess the effectiveness of a machine-learning model in detecting anomalies inside Android networks. The aforementioned incidents were categorized into two distinct clusters, one being benign while the other indicating potential involvement of ransomware. The performance evaluation of the neural network model was conducted using Accuracy, Precision, Recall, and F1 Score metrics, following the completion of training and testing on the dataset.

5. CONCLUSION

These studies use machine learning methodologies to analyze network data on the Android platform for ransomware detection. Following the completion of training, the neural network model exhibited exceptional precision in categorizing network traffic as either benign or originating from ransomware. Significant findings were seen throughout all courses, indicating high values of accuracy, precision, recall, and F1 score. The resilience and potential utility of the model are underscored by its low rates of false positives and false negatives, indicating its effectiveness in real-world scenarios.

The research yielded the subsequent key findings: The machine learning algorithm effectively differentiates between network flows associated with innocuous activity and those associated with ransomware, exhibiting a notable degree of accuracy. The capacity to minimize the occurrence of false alarms renders this feature highly advantageous for practical implementations. The model exhibits robustness and proficiency in effectively addressing a diverse array of threats within the dynamic landscape of the modern world. This is evidenced by its consistent performance across many evaluations including distinct categories of ransomware families and benign network traffic examples. The findings of the comparison analysis indicate that the proposed model exhibits superior performance or is comparable to the most advanced techniques currently available. This finding reaffirms the efficacy of the suggested methodology as a dependable technique for detecting malware within Android network data. The findings of this research contribute considerably to the ongoing discourse surrounding the application of artificial intelligence (AI) in the field of cybersecurity. The study showcases the potential of machine learning in safeguarding Android devices from ransomware threats.

The positive outcomes obtained from this study establish a solid groundwork for subsequent investigation and enhancement in the subsequent areas:

Enhancement of the Model: This study aimed to investigate advanced machine learning and deep learning architectures to enhance the accuracy of detection and minimize the occurrence of false alarms. The utilization of feature engineering approaches is essential to uncover more descriptive features that might effectively boost the model's capability to detect ransomware activity.

Real-time Detection: The objective of this study was to develop real-time systems for detecting ransomware, utilizing the proposed model, to offer prompt protection against ransomware threats. The improvement of performance to achieve real-time processing and low latency is of utmost importance for the successful implementation of real-time detection and mitigation strategies. The continuous update and training of the model are necessary to stay abreast of the dynamic ransomware threat scenario. This study investigated the utilization of transfer learning and online learning methodologies to enhance the model's ability to adapt quickly to new ransomware families

and methods. The integration of the suggested machine learning model with existing security solutions aimed to establish a comprehensive security framework. This study explored and analyzed the effectiveness of multi-modal anomaly detection methods in improving the detection capabilities of ransomware through the integration of data from many sources.

This study focused on the exploration of privacy-preserving machine learning approaches, namely federated learning and differential privacy. The objective was to address concerns related to privacy by safeguarding the confidentiality and integrity of sensitive data throughout the model training and evaluation processes. This study addressed the need for benchmark datasets and standard assessment metrics in the field of ransomware detection in Android network traffic. By establishing these benchmarks and metrics, the research community may encourage a collaborative atmosphere and facilitate comparative comparisons of different models and approaches. This research recommends enhancing and optimizing the suggested machine-learning model by exploring several avenues of future work. The ultimate objective was to establish a safe and resilient digital environment, specifically addressing the growing ransomware attacks that target Android devices.

Conflict of Interest: The author declares no conflict of interest.

Ethical Approval: The study adheres to the ethical guidelines for conducting research.

Funding: This research received no external funding.

REFERENCES

Akbanov, M. and Logothetis, M. (2019). Ransomware detection and mitigation using software-defined networking: the case of wannacry. *Computers & Electrical Engineering, 76,* 111-121. https://www.sciencedirect.com/science/article/pii/S0045790618323164

Alhawi, O. M., Baldwin, J., & Dehghantanha, A. (2018). Leveraging machine learning techniques for Windows ransomware network traffic detection. *Cyber threat intelligence*, 93-106. https://link.springer.com/chapter/10.1007/978-3-319-73951-9_5

Almashhadani, A. O., Kaiiali, M., Sezer, S., & O'Kane, P. (2019). A multi-classifier network-based crypto ransomware detection system: A case study of locky ransomware. *IEEE Access*, *7*, 47053-47067. https://ieeexplore.ieee.org/abstract/document/8674751/

Alotaibi, F. M., & Vassilakis, V. G. (2021). Sdn-based detection of self-propagating ransomware: the case of badrabbit. *Ieee Access*, *9*, 28039-28058. https://ieeexplore.ieee.org/abstract/document/9352796/

Badrinath, S., Dodhi, R., & Muthalagu, R. (2023). Ransomware Detection Service: Execution and Analysis Using Machine Learning Techniques. *Wireless Personal Communications*, *133*(2), 995-1009. https://link.springer.com/article/10.1007/s11277-023-10801-w

Boticiu, S., & Teichmann, F. (2024). How does one negotiate with ransomware attackers? *International Cybersecurity Law Review*, *5*(1), 55-65. https://link.springer.com/article/10.1365/s43439-023-00106-w

Cabaj, K., Gregorczyk, M., & Mazurczyk, W. (2018). Software-defined networking-based crypto ransomware detection using HTTP traffic characteristics. *Computers & Electrical Engineering*, *66*, 353-368. https://www.sciencedirect.com/science/article/pii/S0045790617333542

Cai, J., Yang, H., Lai, T., & Xu, K. (2023). A new approach for optimal chiller loading using an improved imperialist competitive algorithm. *Energy and Buildings*, *284*, 112835. https://www.sciencedirect.com/science/article/pii/S0378778823000658

Chew, C. J. W., Kumar, V., Patros, P., & Malik, R. (2024). Real-time system call-based ransomware detection. *International Journal of Information Security*, 1-20. https://link.springer.com/article/10.1007/s10207-024-00819-x

Dash, M., Londhe, N. D., Ghosh, S., & Sonawane, R. (2022). Hybrid Seeker Optimization Algorithm-based Accurate Image Clustering for Automatic Psoriasis Lesion Detection. In Artificial Intelligence Applications for Health Care (pp. 227-240). CRC Press. http://dx.doi.org/10.1201/9781003241409-12

Duan, S., Luo, H., & Liu, H. (2022). A multi-strategy seeker optimization algorithm for optimization-constrained engineering problems. *IEEE Access*, *10*, 7165-7195. https://ieeexplore.ieee.org/abstract/document/9678976/

Feyli, B., Soltani, H., Hajimohammadi, R., Fallahi-Samberan, M., & Eyvazzadeh, A. (2022). A novel two surfaces hybrid approach for Multi-period heat exchanger networks synthesis by combination of imperialist competitive algorithm and linear programming method. *Chemical Engineering Science*, *258*, 117755. https://www.sciencedirect.com/science/article/pii/S0009250922003396

Hu, P., Aghajanirefah, H., Anvari, A., & Nehdi, M. L. (2023). Combining artificial neural network and seeker optimization algorithm for predicting compression capacity of concrete-filled steel tube columns. *Buildings*, *13*(2), 391. https://www.mdpi.com/2075-5309/13/2/391

Jing, Z., Kuang, H., Leite, W. L., Marcoulides, K. M., & Fisk, C. L. (2022). Model specification searches in structural equation modeling with a hybrid ant colony optimization algorithm. *Structural Equation Modeling: A Multidisciplinary Journal*, *29*(5), 655-666. https://www.tandfonline.com/doi/abs/10.1080/10705511.2021.2020119

Kirubavathi, G., & Anne, W. R. (2024). Behavioral-based detection of Android ransomware using machine learning techniques. *International Journal of System Assurance Engineering and Management*, 1-22. https://link.springer.com/article/10.1007/s13198-024-02439-z

Komisarek, M., Pawlicki, M., Kozik, R., Hołubowicz, W., & Choraś, M. (2021). How to effectively collect and process network data for intrusion detection? *Entropy*, *23*(11), 1532. https://www.mdpi.com/1099-4300/23/11/1532

Li, D., Shi, W., Lu, N., Lee, S. S., & Lee, S. (2024). ARdetector: Android ransomware detection framework. *The Journal of Supercomputing*, *80*(6), 7557-7584. https://link.springer.com/article/10.1007/s11227-023-05741-y

Li, Y., Yang, Z., Wang, L., Tang, H., Sun, L., & Guo, S. (2022). A hybrid imperialist competitive algorithm for energy-efficient flexible job shop scheduling problem with variable-size sublots. *Computers & Industrial Engineering*, *172*, 108641. https://www.sciencedirect.com/science/article/pii/S0360835222006295

Martín, A., Calleja, A., Menéndez, H. D., Tapiador, J., & Camacho, D. (2016). ADROIT: Android malware detection using meta-information. In *2016 IEEE Symposium Series on Computational Intelligence (SSCI)*, 1-8. https://ieeexplore.ieee.org/abstract/document/7849904/

Sankepally, S. R., Kosaraju, N., & Rao, K. M. (2022). Data imputation techniques: an empirical study using chronic kidney disease and life expectancy datasets. In *2022 International Conference on Innovative Trends in Information Technology (ICITIIT)*, 1-7. https://ieeexplore.ieee.org/abstract/document/9744211/

Su, D., Liu, J., Wang, X., & Wang, W. (2018). Detecting Android locker ransomware on Chinese social networks. *IEEE Access*, *7*, 20381-20393. https://ieeexplore.ieee.org/abstract/document/8580446/

Tang, Y., & Zhou, F. (2023). An improved imperialist competition algorithm with adaptive differential mutation assimilation strategy for function optimization. *Expert Systems with Applications*, *211*, 118686. https://www.sciencedirect.com/science/article/pii/S0957417422017201

Wang, Z., Chen, M., Yan, W., Wang, W., Gao, A., Nie, G., … & Yang, S. (2019). Revisiting recent and current anomaly detection based on machine learning in ad-hoc networks. *Journal of Physics Conference Series, 1288*(1), 012075. https://doi.org/10.1088/1742-6596/1288/1/012075

Xing, X., Ding, H., Liang, Z., Li, B., & Yang, Z. (2022). Robot path planner based on deep reinforcement learning and the seeker optimization algorithm. *Mechatronics*, *88*, 102918. https://www.sciencedirect.com/science/article/pii/S0957415822001362