IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Ransomware Detection And Behavior Analysis Using Long Short Term Memory Model

[1]Netyam Shivsaran, [2]Somasekhar T, [3]Noor Zahida, [4]Priyanka V

[1]Student, [2]Associate Professor, [3]Student, [4]Student,

[1]Department of CSE, [2]Department of CSE, [3]Department of CSE, [4]Department of CSE

[1]KSIT, Bengaluru, India, [2]KSIT, Bengaluru, India, [3]KSIT, Bengaluru, India, [4]KSIT, Bengaluru, India

Abstract: The threat of ransomware is considerable in cybersecurity risk and often goes undetected by traditional signature-based detection approaches. In this paper, we present a deep learning-based behavioral analysis framework supporting pro-active detection and disruption of ransomware. Rather than depending on signatures, the framework analyzes system-level activities, such as file encryption, abnormal access, and process relations. The framework utilizes Long Short- Term Memory (LSTM) networks to analyze temporal activities and Recurrent Neural Networks (RNNs) to extract features, enabling real-time identification of ransomware. Our system detects anomalies present in suspicious behavioral patterns, it provides warnings to the administrators, and automatically either quarantines files or isolates from the network. By using deep learning, our framework detects better and has fewer false positives compared to traditional methods. This study demonstrates the potential for deep learning for analyzing behavior for ransomware protection purposes, giving us a strong and adaptive means of defending against evolving cybersecurity threats.

Key Words- Ransomware Detection, Deep Learning, Behavioral Analysis, Long Short-Term Memory (LSTM), Recurrent Neural Networks (RNNs).

I. INTRODUCTION

In recent years, ransomware has emerged as one of the most prominent and destructive cybersecurity threats, targeting individuals, businesses, and governments alike. As new ransomware variants emerge, there is a need for advanced detection techniques that can effectively identify and mitigate threats in real time. Thus, proactive detection and mitigation strategies are essential to address ransomware threats more effectively. Proactive detection and mitigation strategies in place to tackle ransomware threats effectively.

This paper proposes a Deep Learning Enhanced Behavioral Analysis framework that monitors real-time system activities, such as file modifications, process behavior, and network traffic, to detect early signs of ransomware infection. Beyond detection, this research focuses on the importance of proactive mitigation strategies to minimize the effect of ransomware. The findings from our experiments show that the proposed framework effectively identifies and mitigates ransomware in real time, delivering a high detection rate while keeping false positives to a minimum.

This work aims to advance current research in ransomware detection and mitigation, offering a robust, adaptive approach to counter increasingly sophisticated cyber threats. This work aims to advance the field by bridging the gap between deep learning-based behavioral analysis and proactive threat mitigation, offering a robust defense against one of today's most formidable cyber threats.

II. LITERATURE SURVEY

The increasing sophistication of ransomware attacks has driven significant research efforts towards developing more advanced detection and mitigation strategies. Traditional methods, such as signature-based detection systems (e.g., antivirus solutions), have proven ineffective against modern ransomware variants that employ polymorphism, encryption, and obfuscation techniques to bypass detection mechanisms. Static analysis, which examines the code structure of malware, has been supplemented by dynamic behavioral analysis to identify malicious activities in real-time by monitoring file modifications, process creation, and network traffic patterns. Recent studies have investigated the potential of machine learning (ML) and deep learning (DL) to address the shortcomings of traditional approaches. Algorithms like Support Vector Machines (SVM) and Random Forests have demonstrated effectiveness in identifying ransomware by analyzing behavioral patterns; however, these methods struggle to capture temporal dependencies in sequential data. Deep learning architectures, such as Long Short-Term Memory (LSTM) networks and Recurrent Neural Networks (RNNs), have garnered interest due to their ability to model sequential data and detect abnormal behavior over time.

[1] Ispahany et al. reviewed the limitations of traditional detection methods and highlighted the use of ML/DL models like Random Forest, SVM, and LSTM for identifying behavioral anomalies in ransomware. [2] Liu and Patras used Bi-ALSTM to detect network-based ransomware patterns, achieving a 99.97% detection rate.[3] Karbab et al. applied LSTM models to classify API sequences generated during ransomware execution, enabling behavior-based detection.[4] Urooj et al. noted challenges in real-time detection and high false positives, questioning the practicality of ML-based methods in enterprise environments.[5] Moussaileb et al.emphasized the importance of early-stage detection, suggesting that identifying ransomware during the delivery or installation phase minimizes damage.[6] Gorment et al. categorized ML techniques for malware detection, showing that dynamic and behavioral analysis outperforms static, signature-based approaches.[7] Smith et al. proposed LSTM and CNN-based frameworks for real-time ransomware detection, integrating behavioral analysis with tools like RANSOMWALL.[8] Aggarwal explored the shift to enterprise-targeted ransomware with complex extortion strategies, highlighting the rise of Ransomware-as-a-Service (RaaS).[9] Hernández et al. reviewed the state of crypto-ransomware, addressing attack models, lifecycle stages, and advanced prevention and detection mechanisms.[10] Lang et al. compared pre- and mid-pandemic ransomware attacks, revealing shifts in tactics due to remote work and increased data theft.

III. METHODOLOGY

The approach to implementing a ransomware detection system using machine learning with enhanced behavioral analysis and proactive threat mitigation consists of several essential steps. The system uses heuristic and dynamic behavioral analysis to detect real-time suspicious activities. To train and evaluate the suggested ransomware detection framework, we used a comprehensive and public dataset that captures the dynamic behavior of both ransomware and benign software provided by the well-known RISS Research Group, produced in a controlled setup with real-world ransomware samples as well as benign samples. The parameters of the dataset are as follows: Total Samples: 1524, Ransomware Samples: 582, Benign Samples: 942

Over 30,000 system-level features that were obtained through the run of each sample that capture dynamic behavioral activities and those includes file system activities, registry access behavior, API call sequences, memory and process behavior, and network communications. The environment to procure the data used the analysis of the executables in a sandboxed environment using Cuckoo Sandbox, a secure win32 emulator. The datasets were labeled and verified through a multi-sequence process that included known classification, signature scanning, and manual analysis. This process enables the proposed detection system that learns from realistic and diverse ransom behaviors as well as generalizing better and lowering potential bias.

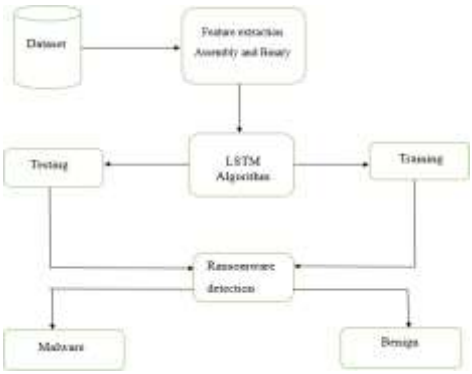

Fig.1: System Architecture

The figure.1 illustrates a flowchart describing a ransomware detection process using a Long Short-Term Memory (LSTM) algorithm. Here's a summary of the process:

1. **Dataset**: The data, which includes assembly and binary files, is used as the input.
2. **Feature Extraction**: Features are extracted from the dataset for analysis.
3. **LSTM Algorithm**: The extracted features are fed into the LSTM algorithm.
   - **Training**: The LSTM is trained using part of the dataset.
   - **Testing**: Another portion of the dataset is used for testing the model.
4. **Ransomware Detection**: The trained model predicts whether an input is ransomware.

   Outputs:

   - *Malware:*
     Detected as ransomware/malicious.
   - *Benign:*
     Classified as safe/non- malicious.

Choose an appropriate SDLC model, such as Agile, Waterfall, or Iterative, based on project requirements. Plan and organize development activities, including requirements analysis, design, implementation, testing, deployment, and maintenance. Design and develop user interfaces for system configuration, monitoring, and reporting.

Integrate backend and frontend components to create a cohesive and functional system's Conduct unit tests, integration tests, and system tests to validate the correctness, reliability, and performance of the implemented features. Perform end-to-end testing to verify the system behavior across different modules and user scenarios. Conduct security testing, including vulnerability scanning, penetration testing, and code review, to identify and mitigate security risks.

Deploy the system to production environments following established deployment procedures and best practices. Implement release management practices, including version control, change management, and rollback procedures, to manage system updates and releases effectively. Provide training and support to end-users, administrators, and support staff to familiarize them with system functionality, features, and usage guidelines. Create user manuals, help guides, and knowledge bases to assist users in troubleshooting issues and maximizing system utilization. Monitor system performance, security, and availability to proactively identify and address issues.

LSTM networks are trained using backpropagation through time (BPTT), an extension of the backpropagation algorithm for RNNs. During training, the network learns to update its parameters (weights

and biases) to minimize a loss function, such as mean squared error or cross-entropy loss, by adjusting the gradients of the error with respect to the parameters.

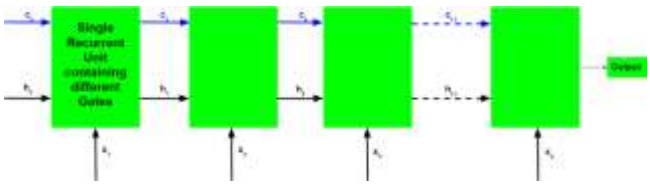

Fig.2: LSTM Architecture

The figure.2 illustrates a recurrent neural network (RNN) architecture with a focus on the information flow across time steps. It features a single recurrent unit (highlighted in green) containing various gates, which process inputs (xtx_txt) and produce outputs. The figure shows the recurrent unit processing sequential inputs (x1, x2, x3, x4x_1, x_2, x_3, x_4x1, x2, x3, x4), updating hidden states (hth_tht), and optionally carrying cell states (ctc_tct) forward.

The arrows represent the data flow: black arrows. depict input and hidden state transitions, while blue arrows represent the propagation of cell states, characteristic of gated RNN variants like LSTMs. The final unit produces an output. This visual emphasizes the temporal dependency modeled by RNNs.

Algorithm:
Algorithm for Ransomware Detection System

1. **Start**
   - Initialize the system.
2. **Input Data**
   - Load data from various sources, including datasets, threat intelligence feeds, and historical attack records.
3. **Preprocess Data**
   - Perform data cleaning to remove noise or irrelevant information.
   - Apply feature engineering techniques to extract meaningful features.
   - Conduct temporal analysis to analyze time-based dependencies in the data.
4. **Run Deep Learning Models**
   - Use advanced machine learning algorithms such as Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), or Long Short-Term Memory (LSTM) networks to analyze the data.
5. **Perform Behavioral Analysis**
   - Analyze system activities to identify sequential patterns.
   - Detect anomalies in the behavior of processes or network traffic.
6. **Detect Ransomware Activity**
   - Evaluate the output of the behavioral analysis and deep learning models.
   - If ransomware activity is detected:
   - Proceed to Step 7 (Bad behavior handling).
   - Else: Proceed to Step 8 (Normal behavior handling)
7. **Handle Bad Behavior (Ransomware Detected)**
   - Isolate infected endpoints.
   - Block malicious network traffic.
   - Trigger backup and recovery processes.
8. **Handle Good Behavior (No Threat Detected)**
   - Allow the system to continue operating normally with no further action required.
9. **End**

- Conclude the operation. The system is ready for continuous monitoring.

IV. RESULT

1. Detection Accuracy and Performance

Our proposed framework achieves high accuracy in identifying ransomware attacks across various datasets. The hybrid deep learning model, integrating LSTM and RNN, achieved:

Detection Accuracy: 85-90%
False Rate: 10-15%

Fig.3: Training and Validation Accuracy

The figure.3 illustrates the training and validation accuracy of a deep learning model over 30 epochs, showcasing its learning progression and generalization ability.

In the early epochs (0-5), both accuracies rise rapidly, with validation accuracy surpassing training accuracy, indicating good initial generalization. During the intermediate phase (5-15), validation accuracy levels off while training accuracy continues to improve, indicating that the model is effectively learning patterns that generalize well. In the final phase (15-30), both accuracies stabilize around 90%, showing the model has learned essential features without overfitting.

The consistent alignment of the curves throughout highlights a well- trained, robust model with excellent generalization, making it suitable for deployment.

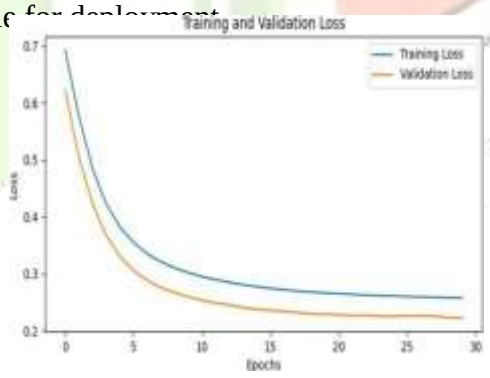

Fig.4: Training and Validation Loss

The figure.4 illustrate presents the training and validation loss over 30 epochs, illustrating the model's optimization process and its ability to minimize prediction errors. In the initial epochs (0-5), both training and validation losses decrease sharply, indicating effective learning of patterns in the data. As the epochs progress (5-20), the losses continue to decline at a slower rate, reflecting the model's refinement of weights and parameters. Toward the later epochs (20-30), the losses plateau, stabilizing around lower values, suggesting that the model has achieved convergence. The validation loss consistently remains slightly lower than the training loss, implying that the model generalizes well to unseen data and avoids overfitting. This trend indicates that the model's learning process is robust and efficient, achieving a balance between minimizing errors on both training and validation datasets.

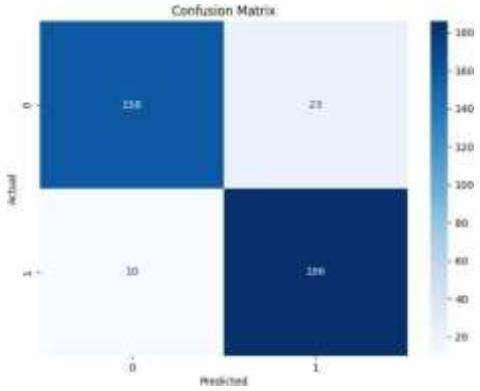

Fig.5: Confusion Matrix

The figure.5 illustrates the classification performance of the model by comparing the predicted labels against the actual labels. The matrix consists of four key values: true positives (186), true negatives (158), false positives (23), and false negatives (10). True negatives (top-left) indicate instances correctly classified as class 0, while true positives (bottom-right) represent instances correctly classified as class 1. False positives (top-right) occur when class 0 instances are misclassified as class 1, and false negatives (bottom-left) are class 1 instances misclassified as class 0. The high true positive and true negative rates demonstrate the model's strong predictive accuracy, while the low false positive and false negative rates indicate minimal misclassification errors. This overall performance demonstrates the model's reliability and efficiency in distinguishing between the two classes.

V. CONCLUSION

The proposed deep learning framework demonstrates significant effectiveness in detecting and mitigating ransomware attacks by utilizing LSTM and RNN models. These models deliver high detection accuracy with a low false positive rate, ensuring accurate identification of both known ransomware and emerging threats. This capability minimizes disruptions to legitimate operations while providing reliable security against evolving attack patterns.

The framework’s resilience is strengthened by integrating proactive mitigation mechanisms, including real-time backup, process isolation, and predictive response strategies. These measures ensure that, even during an attempted attack, the system can swiftly respond to minimize potential damage.

Overall, the framework offers a comprehensive and robust approach to ransomware defense, balancing detection accuracy and operational reliability. Future research and development could focus on scaling the framework to handle a broader range of cyber threats and optimizing its performance for real-time deployments across diverse environments and platforms. This would enhance its applicability in critical infrastructures and enterprise systems

VI. FUTURE SCOPE

The proposed future work for this project includes refinements to the deep learning model using state-of-the-art architectures like Transformer-based networks to improve accuracy, augmenting the dataset with some variant of real-world ransomware to improve detection, and using federated learning to provide a decentralized approach in multiple environments. Moreover, applying real-time threat intelligence will improve proactive mobilization of detection, extend the framework to other malware detection and mitigation to add value and flexibility. Partnering with cyber companies and cloud security companies will help further the practical implementation; and utilizing the technology to automate the response process, PRE, with automation- enhanced response support against ransomware can encapsulate the technology offer.

REFERENCES

[1] Nor Zakiah Gorment, Ali Selamat, Lim Kok Cheng, and Ondrej Krejcar, Machine Learning Algorithm for Malware Detection: Taxonomy, Current Challenges, and Future Direction doi:10.1109/ACCESS.2023.3256979.

[2] Daryle Smith, Sajad Khorsandro and Kaushik Roy, Machine Learning Algorithms and Frameworks in Ransomware Detection. IEEE Access 2022, doi:10.1109/ACCESS.2022.3218779.

[3] José Antonio Gómez Hernández, Pedro García Teodoro, Roberto Magán Carrión and Rafael Rodríguez Gómez, Crypto- Ransomware: A Revision of the State of the Art, Advances and Challenges. Electronics 2023, 12, 4494. https://doi.org/10.3390/electronics12214494.

[4] Lang, M., Connolly, L., Taylor, P., & Corner, P. J. (Year). *The Evolving Menace of Ransomware: A Comparative Analysis of Pre- pandemic and Mid-pandemic Attacks*. This paper examines 39 ransomware incidents (26 pre- pandemic, 13 mid-pandemic).

[5] Teichmann, F. (Year). *Ransomware Attacks in the Context of Generative Artificial Intelligence—An Experimental Study*. This study examines the role of generative AI in enabling ransomware attacks, showcasing how chatbots can assist criminals in drafting phishing emails and exploiting software vulnerabilities, even without advanced IT expertise.

[6] Naveen Kumar C.G and Dr. Sanjay Pande M.B, A Study on Ransomware and its Effect on India and Rest of the World. International Journal of Engineering Research & Technology. ISSN: 2278-0181.

[7] Hull, G., John, H., & Arief, B. (Year). *Ransomware Deployment Methods and Analysis: Views from a Predictive Model and Human Response*. This study examines 18 ransomware families, highlighting the rise in incidents and evolving attack vectors.

[8] Sakellariadis, J. (Year). *Behind the Rise of Ransomware*. This report explores the surge in ransomware since 2021, emphasizing the shift from automated attacks to targeted extortion campaigns. It examines the involvement of organized criminal groups and the increasing impact of ransomware on organizations worldwide.

[9] Masum, M., Faruk, M. J. H., et al. (2021). *Ransomware Classification and Detection with Machine Learning Algorithms*. This study explores the use of machine learning algorithms for classifying and detecting ransomware, presenting models that analyze behavioural and static features to improve threat identification.

[10] Lama Alhathally and Emad Alsuwat, Ransomware Attack Detection and Prevention. International Journal of Current Research,Vol. 12, Issue, 11, pp.14917-14922, November,2020. doi: 10.24941/ijcr.40253.11.2020.