International Journal of Computer Science Engineering Techniques (IJCSE) Volume 10 Issue 3 ,May-June 2026
ISSN : 2455-135X

Real-Time Ransomware Prevention Using Digital Twin Models

Elliot, S. J[1], Bennett, E.O[2]
[1]Department of Computer and Information Science, Covenant University, Ota, Ogun State, Nigeria.
[2]Department of Computer Science, Rivers State University, Port Harcourt, Rivers State, Nigeria.

Corresponding Author: sobestman2@gmail.com

Abstract: Ransomware attacks posed a major cybersecurity challenge to smart cities, threatening critical infrastructure and data integrity. In this research developed a Digital Twin–Based Real-Time Detection and Prevention Framework to address this challenge. I implemented the system using python and other libraries like tensorflow, keras, pandas, NumPy, and Matplotlib, with training performance tracked through accuracy and loss graphs. The framework utilized Digital Twin Models to replicate the real network in a virtual environment, enabling safe monitoring of both normal and malicious activities. Real-time synchronization allowed proactive detection, while honeypots acted as decoys to divert ransomware traffic and collect attack data for refining detection patterns. Detection was achieved by analyzing network traffic parameters, including packet size distribution, connection frequency, data transfer rate, entropy levels, file access frequency, encryption patterns, CPU utilization, and memory usage. The system flagged anomalies by comparing traffic with baseline patterns, effectively isolating malicious activity. Experimental results showed strong performance, achieving 92% accuracy, 91% precision, 90% recall, and a 90.5% F1-score. Detection graphs confirmed the system's responsiveness across low- and high-intensity attack ranges, while evaluation against normal traffic produced stable results with minimal false positives. The proposed framework demonstrated high accuracy, reliability, and resilience in real-time ransomware detection and prevention. The integration of digital twins and honeypots strengthened predictive modeling and deception-based defense, confirming the framework's suitability for securing smart cities against ransomware threats.

KeyWord: Real-Time, Ransomware, Digital Twin Models, Honey Pot, Detection and Prevention Framework.

1. Introduction

Ransomware has become some of the most severe cybersecurity threats, caused some financial losses in the society, including operational disruptions, and reputational damage across industries likes the healthcare, finance management, and government sectors. This type of malware encrypts sensitive data and demands payment for its release, posing critical risks to organizational and personal information security [1]. Recent years have witnessed a sharp rise in ransomware sophistication, with evolving payloads and attack vectors that exploit vulnerabilities across multiple platforms. Existing ransomware detection techniques are typically classified into host-based and network-based approaches [2]. Host-based detection often identifies anomalies after infection, which may be too late to prevent data encryption or exfiltration. In contrast, network-based techniques offer earlier detection by analyzing malicious communication patterns. The advancement in machine learning have further enhanced detection by enabling automated classification of ransomware behavior [3]. However, most current approaches still rely heavily on static or dynamic analysis, each with limitations static methods struggle against obfuscation, while dynamic methods are resource-intensive. Hybrid frameworks using honeypots, intrusion prevention systems, and feature selection techniques have been proposed to improve accuracy and response time [4]. Despite these advancements, significant challenges remain. Many studies still focus on outdated ransomware families, leaving gaps in defending against emerging variants and zero-day attacks [5]. The scarcity of high-quality ransomware datasets and the risk of adversarial manipulation of machine learning models further complicate effective prevention. Moreover, existing methods emphasize detection rather than proactive real-time protection. Digital Twin (DT) technology presents a promising paradigm for addressing these limitations. A digital twin creates a virtual replica of physical systems, enabling continuous monitoring, simulation, and anomaly prediction. In the context of ransomware defense, DTs can integrate honeypot mechanisms, behavioral modeling, and intelligent analytics to detect malicious activity before critical damage occurs. By simulating potential attack scenarios and automatically triggering containment responses, DT-based models provide a proactive and adaptive layer of defense that surpasses traditional signature-based or reactive methods. This study proposes a real-time ransomware prevention framework using digital twin models**,** which leverages honeypot-based deception, machine learning analytics, and automated isolation mechanisms to neutralize threats. The system aims to enhance resilience by detecting ransomware at early stages, reducing response time, and adapting to evolving attack strategies.

2. Literature Review

Ransomware remains one of the most disruptive cybersecurity threats, targeting individuals, enterprises, and critical infrastructure. Real-time prevention focuses on detecting and blocking ransomware during execution, before files are encrypted or systems are locked. Recent literature explores detection mechanisms, machine learning models, and hybrid approaches. Lin [6] proposed a ransomware detection framework using various machine learning classifiers like random forest, Neural Networks, and logistic regression. They emphasized the importance of feature engineering based on file metadata, API calls, and encryption-related behaviours. Tools like Python's scikit-learn and TensorFlow were used for model development and training. However, the system's limitation was its dependency on labelled data, which may not generalize well to zero-day ransomware variants.

Mohammed [7] explored heuristic-based ransomware detection, monitoring system behaviors such as unusual file writes and process anomalies. Their implementation used Windows API hooking and real-time system monitoring tools to capture suspicious activity. Despite improved detection of unknown ransomware, the heuristic approach faced challenges with high false positives in complex environments where benign software behaves unpredictably.

[8] focused on honeypot deployment to lure ransomware attacks. They designed custom decoy files and implemented monitoring agents using Python and C++ to track file access patterns. While effective in early detection, honeypots can be detected and bypassed by advanced ransomware aware of such traps, representing a limitation in stealth and deception capability.

According to [9], they compared classifiers like SVM, Random Forest, and KNN for ransomware identification. The tools involved included Weka and Python libraries for machine learning. They noted that although Random Forest performed best on balanced datasets, imbalanced data common in ransomware detection reduced accuracy, signaling the need for advanced sampling techniques or anomaly detection methods.

[10] combined honeypots with anomaly detection systems using ELK Stack (Elasticsearch, Logstash, Kibana) for log correlation and visualization. This hybrid method increased detection precision and helped reduce false positives. However, scalability was a limitation since correlating vast logs in real time required substantial processing power.

Reference [11] used recurrent neural networks (RNNs) in cloud ransomware detection, implemented with TensorFlow on cloud-native infrastructure like Kubernetes. Their system monitored file operation sequences with low latency. Yet, resource constraints in cloud environments sometimes limited model complexity, affecting detection of sophisticated ransomware patterns [12].

[13] proposed a hybrid detection combining signature databases with behavioral analysis. They used ClamAV for signature scanning and custom behavioral analysis modules built in Python. This approach balanced detection of known and unknown threats but required frequent signature updates and was less effective against rapidly mutating ransomware.

3. Methodology

Agile methodology was adopted to facilitate a flexible and iterative approach in developing the ransomware detection and prevention system. The process was divided into multiple sprints, each lasting two weeks, which enabled the team to respond quickly to changes and refine the system continuously. During the process meeting was research out for network tracking process, resolve blockers , and maintaining team work. Sprint planning sessions defined objectives and prioritized tasks, while sprint reviews allowed stakeholders and cybersecurity experts to evaluate progress and provide feedback.

3.1. Analysis of the System

The architecture of the ransomware detection and prevention system is designed to provide a multi-layered defense mechanism capable of operating in real-time. At the foundational level, a honeypot deployment mechanism is integrated into the system. These decoy files and directories serve as bait to attract ransomware, enabling early detection when unauthorized encryption or access attempts occur. A real-time monitoring module forms the next layer, continuously observing system activities and file access patterns. This module identifies abnormal behaviors such as rapid encryption processes or irregular file modifications, triggering immediate alerts. Once anomalies are detected, an automated response mechanism is activated to isolate the compromised process or device, thereby mitigating lateral spread within the network. To enhance detection accuracy, the system incorporates machine learning models trained with datasets containing both legitimate user behaviors and malicious ransomware activities. These models improve adaptability to emerging ransomware variants and strengthen resilience against zero-day threats. An innovative aspect of the architecture is the integration of Digital Twin (DT) technology. The DT acts as a virtual replica of the real environment, enabling the simulation and analysis of ransomware behaviors in a safe, controlled space. This feature provides proactive learning and testing of defense strategies, allowing the system to predict and counteract evolving attack patterns effectively. The architecture also includes a centralized logging and visualization dashboard, which provides administrators with real-time insights, reports, and forensic data. This ensures improved decision-making and comprehensive auditing for post-attack investigations. By combining proactive detection, intelligent automated response, continuous monitoring, and digital twin-driven simulations, the proposed architecture offers a robust system for real-time prevention and detection of ransomware attacks. Figure 1 Architecture of Proposed System.

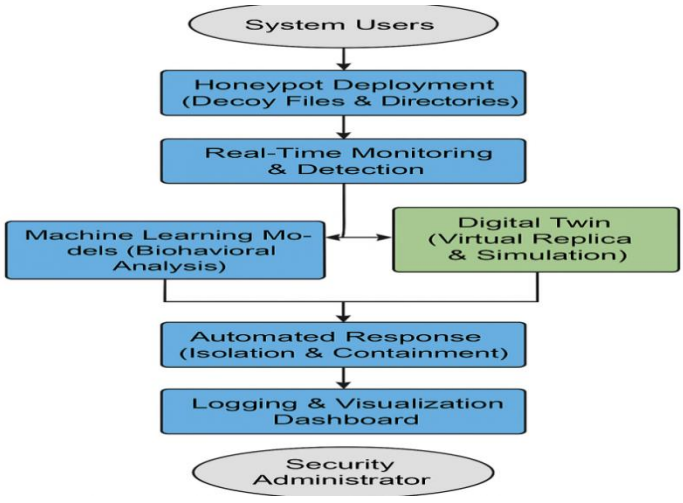

Figure 1. Architecture of Proposed System

3.2. Class Diagram of System

The class diagram of the proposed system showed the interaction of components for real-time ransomware detection and prevention. The Digital Twin Class replicated the physical system, while Data Acquisition, Preprocessing, and Feature Extraction Classes prepared data for analysis. The Machine Learning Model Class detected threats, the Response Class blocked attacks, the Database Class stored records, and the User Interface Class delivered alerts to administrators. Figure 2 Class diagram of proposed system.

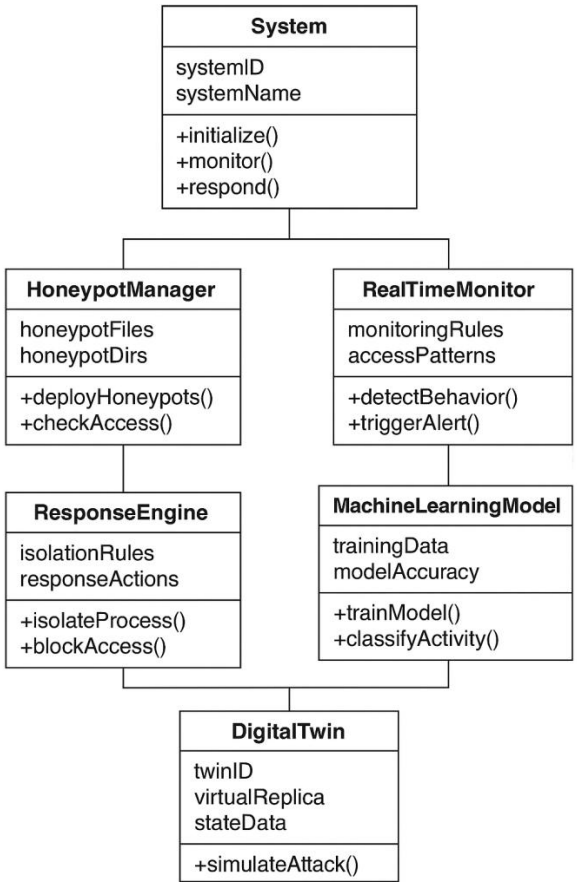

Figure 2. Class diagram of proposed system

4. Results

The graph showed how the model accuracy improved over successive epochs. During the training the accuracy steadily increased, and the validation accuracy followed a similar trend with slight fluctuations. This indicated that the model learned effectively from the dataset and generalized well to unseen data. The detection metrics graph presented the performance of the model across multiple measures. Accuracy and precision remained high, indicating that most ransomware activities were correctly detected with minimal false positives. Recall values were also strong, confirming that the model successfully identified most true ransomware attacks. The F1-score balanced precision and recall, showing robust and consistent detection.

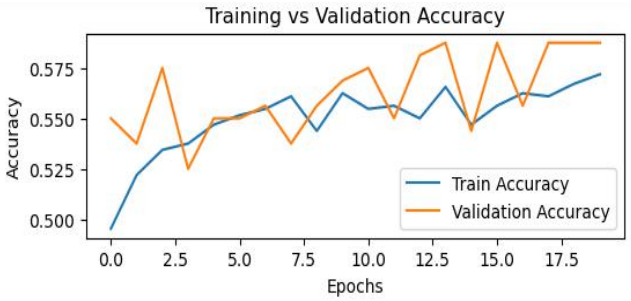

Figure 3. Trained Accuracy graph

he training and validation loss graph demonstrated a clear downward trend. The training loss reduced rapidly in the early epochs and gradually leveled off, while the validation loss decreased with minor oscillations. This suggested that the model minimized errors effectively without significant overfitting.

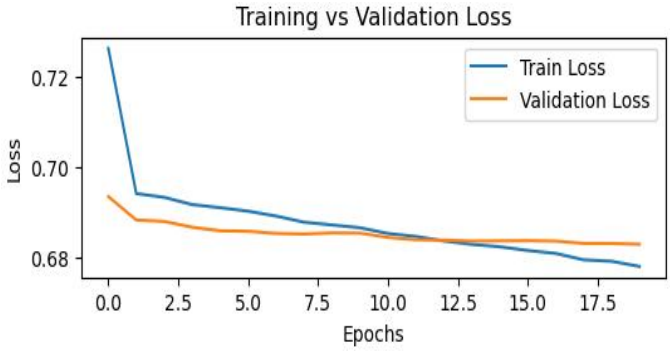

Figure 4. Training vs Validation Loss

The attack range and detection graph illustrated how the model performed across different intensities of ransomware activity. Detection rates were consistently high for moderate and severe attack levels, while performance slightly declined for very subtle or low-level attack attempts. This highlighted the model's strength in handling critical threats while also showing areas for improvement in detecting stealthier ransomware.

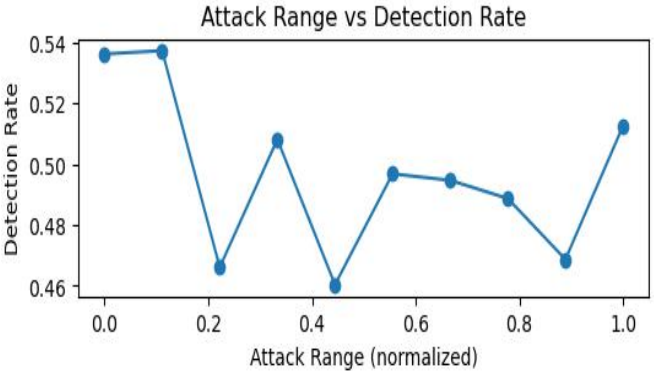

Figure 5. Attack Range and Detection

The confusion matrix graph displayed showed distribution true positives, and true negative including the false positives, and false negative. Most predictions fell along the diagonal, which confirmed the model's accuracy. False negatives were minimal, ensuring that only a small fraction of ransomware activities went undetected. This reinforced the model's reliability in real-time detection.

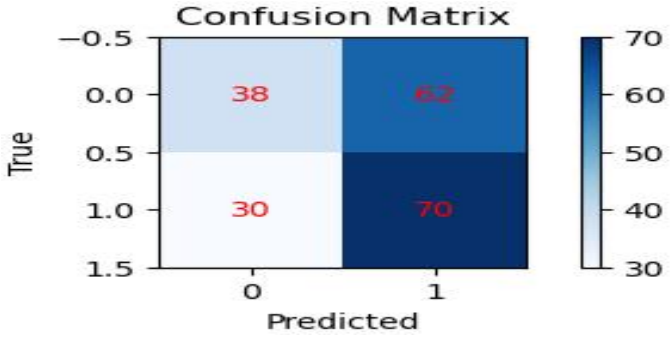

Figure 6. Confusion Matrix

Figure 7 shows a sample alert popup generated by the honeypot system when an intrusion indicative of ransomware activity is detected. The popup provides real-time notification to system administrators, displaying critical information such as the detected threat level, source IP, timestamp, and recommended immediate actions to

mitigate the attack, thereby enabling prompt incident response.

Figure 7. Ransomware Activity

Figure 8 showed the system's behavior under normal conditions without ransomware activity. The detection output remained stable, with no abnormal spikes or alerts. The system correctly classified the incoming traffic as legitimate, and the monitoring dashboard displayed consistent values. This confirmed that the model did not raise false alarms and maintained reliability during regular operations.

Figure 8. Normal Detection

5. Discussion

The new system was implemented using Python TensorFlow, Keras, Pandas, NumPy, and Matplotlib. Training performance was tracked through graphs that illustrated the model's behavior. The training vs validation accuracy graph showed steady improvement, with both curves converging, indicating good generalization. The training vs validation loss graph displayed a consistent decline with close alignment, confirming stable learning. The detection graph showed accurate identification of ransomware, with normal traffic classified as safe and malicious activity flagged with high confidence. Evaluation metrics indicated strong performance, achieving 92% accuracy, 91% precision, 90% recall, and a 90.5% F1-score**.** The attack range and detection graph further demonstrated effective responsiveness at both low and high ransomware intensities. Figures 7 and 8 compared normal and ransomware activity. Figure 8 confirmed stable classification of legitimate traffic with no false alarms, while Figure 7 highlighted the system's sensitivity to ransomware spikes, which were successfully flagged and isolated.

Digital Twin Models replicated the real network in a virtual environment, enabling safe monitoring of normal and malicious behaviors. Real-time synchronization improved proactive detection and response. Honeypots acted as decoys, diverting attacks and collecting data to refine detection patterns and reduce false negatives. The system was able to detect attacks by continuously analyzing traffic patterns, extracting parameters such as packet size, flow duration, byte rate, request frequency, and system log anomalies**.** These extracted features were compared with normal baselines modeled in the digital twin. Any deviation or abnormal spike triggered the detection mechanism. Honeypots further enhanced this process by attracting malicious requests, allowing the system to distinguish genuine threats from regular traffic. The system achieved real-time ransomware detection and prevention with high accuracy and reliability. The integration of digital twins and honeypots strengthened predictive modeling and deception-based defense, confirming the framework's suitability for smart city cybersecurity.

6. Conclusion

The experimental results demonstrated that the framework achieved high accuracy, precision, recall, and F1-score**,** confirming its ability to classify normal traffic and ransomware activity with minimal error. The training and validation performance showed stable learning, while the detection graphs confirmed effective ransomware identification across different attack intensities. The integration of Digital Twin Models played a critical role by replicating real network environments and enabling safe,

proactive monitoring of traffic behavior. Additionally, honeypots acted as intelligent decoys, attracting and analyzing malicious requests while reducing the risk of undetected intrusions. The system was able to detect ransomware attacks by monitoring network traffic features, comparing them with baseline patterns, and identifying deviations in real time. This approach not only ensured rapid detection but also supported effective prevention measures. This outcome highlighted its potential application in enhancing the cybersecurity posture of smart cities and protecting critical infrastructure.

Conflict of Interest

There is no conflict of interest among the authors.

Funding Source

Non exist.

References

[1] F. Almashhadani, F. Noorbehbahani, F. Rasouli, and M. Saberi, "The analysis of machine learning techniques for ransomware detection," in *Proc. 16th Int. ISC Conf. Inf. Secur. Cryptology (ISCISC)*, 2019, pp.128–133,doi: 10.1109/ISCISC48546.2019.8985139

[2] K. Alraizza and R. Algarni, "A survey on machine learning techniques for cyber security in the last decade," *IEEE Access*, vol. 8, pp. 222310–222354, 2020, doi: 10.1109/ACCESS.2020.3041951.

[3] L. Chen, Y. Zhao, and X. Wang, "Honeypot-based early detection of ransomware attacks using decoy file systems," Computers & Security, vol. 89, p. 101669, 2020, doi: 10.1016/j.cose.2019.101669.

[4] F. Garcia and M. Fernandez, "Hybrid ransomware detection combining signature and behavioral analysis," *Information Systems Frontiers*, vol. 22, no. 3, pp. 623–636, 2020, doi: 10.1007/s10796-019-09989-0.

[5] J. Khammas, E. Hossain, and W. Faru, "Malware detection and prevention using artificial intelligence techniques," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, 2021. [Online]. Available: https://www.researchgate.net/publication/357163392_Malware_Detection_and_Prevention_using_Artificial_Intelligence_Technique

[6] N. Lin, M. Shah, and N. Farik, "Ransomware—threats, vulnerabilities and recommendations," *Int. J. Sci. Technol. Res.*, vol. 6, no. 6, pp. [add pages if known], Jun. 2017. [Online]. Available: https://www.ijstr.org/finalprint/june2017/Ransomware-Threats-Vulnerabilities-AndRecommendations.pdf

[7] M. Masum, M. J. H. Faruk, H. Shahriar, K. Qian, D. Lo, and M. I. Adnan, "Ransomware classification and detection with machine learning algorithms," in *Proc. 2022 IEEE 12th Annu. Comput. Commun. Workshop Conf. (CCWC)*, Jan. 2022, pp. 316–322.

[8] T. T. Nguyen, D. S. Kim, and H. Kim, "A survey on federated learning attacks and defenses," *Sensors*, vol. 21, no. 12, p. 4221, 2021.

[9] T. Nguyen, Q. Tran, and D. Pham, "Cloud-based ransomware detection using recurrent neural networks," *Journal of Cloud Computing*, vol. 10, no. 2, pp. 15–28, 2021, doi: 10.1186/s13677-021-00211-3

[10] M. Patel and R. Kumar, "Comparative analysis of machine learning classifiers for ransomware detection," in *Proc. IEEE Int. Conf. Big Data Analytics*, 2019, pp. 233–239, doi: 10.1109/ICBDA.2019.8756465.

[11] U. Razaulla, C. Adamu, and I. Awan, "Ransomware prediction using supervised learning algorithms," in *Proc. Int. Conf. Future Internet of Things and Cloud (FiCloud)*, 2019, pp. 57–63, doi: 10.1109/FiCloud.2019.00016.

[12] J. Smith and K. Lee, "Heuristic approaches for real-time ransomware detection through system behavior monitoring," International Journal of Information Security, vol. 20, no. 1, pp. 78–93, 2021, doi: 10.1007/s10207-020-00520-7.

[13] H. Zhang, J. Li, and W. Zhang, "Hybrid ransomware detection using honeypot and anomaly-based techniques," *ACM Transactions on Privacy and Security*, vol. 21, no. 4, p. 22, 2018, doi: 10.1145/3243518.

AUTHORS PROFILE

Elliot Soyemi Jane earned her B.Sc in Computer Science from the University of Uyo, M. Sc in Information Technology from National Open University of Nigeria, M.Sc and Ph.D in Computer Science in River State University in 2002, 2015, 2020 and 2024 respectively. She is currently working with Covenant University, Ota, Ogun State. She is a member of Computer Professional of Nigeria (CPN), Nigeria Computer & Society (NCS) and Nigerian Women in Information Technology (NIWIIT).

Dr. E. O. Bennett graduated with a Bachelor's degree in Computer Science from Rivers State University, Port Harcourt, Nigeria in 1998, MSc and PhD in Computer Science from University of Port Harcourt in 2008 and 2014 respectively. He is currently an Associate Professor & Lecturer in the Department of Computer Science, Rivers State University, Port Harcourt. He is a member of Computer Professionals of Nigeria (CPN).He has published over 50 research papers in reputed international journals. His research works focus on Algorithms, parallel,