International Journal of
AI ELECTRICAL CIVIL AND MECHANICAL ENGINEERING
Peer Reviewed, Referred & Indexed Journal
: 3070-0434 www.zesterapublications.com Original Research Paper

DETECTION OF RANSOMWARE ATTACKS USING PROCESSOR AND DISK USAGE DATA

Mr.T.ROHINI KUMAR[1], KARI MARY STELLA[2], ARNEPALLI RISHI VARDHAN[3], DASARI PRAVEENA[4], GANDIKOTA NAGA VENKATESWARA RAO[5]

[1]Assistant Professor of CSE, V.K.R, V.N.B & A.G.K. College of Engineering, Gudivada

[2345]UG Students, Department of CSE, V.K.R, V.N.B & A.G.K. College of Engineering, Gudivada

ABSTRACT

Ransomware attacks have emerged as a significant cybersecurity threat, causing severe data loss and financial damage to individuals and organizations. Traditional signature-based detection methods often fail to identify new or evolving ransomware variants, making behavior-based monitoring an effective alternative. This study proposes a ransomware detection approach based on analyzing processor usage and disk activity patterns. By continuously monitoring system-level metrics such as CPU utilization, read/write operations, disk I/O rates, and process behavior, the system identifies abnormal resource consumption that typically occurs during encryption processes. Machine learning algorithms are applied to classify normal and malicious activities using extracted performance features. The proposed model aims to provide early-stage detection with minimal system overhead, enabling faster response and mitigation. Experimental results demonstrate that processor and disk usage data can serve as reliable indicators for identifying ransomware behavior, improving detection accuracy while reducing false positives. This approach contributes to proactive cybersecurity defense by enabling real-time monitoring and intelligent threat analysis.

I INTRODUCTION

Ransomware has become one of the most dangerous forms of cyberattacks in recent years, targeting individuals, businesses, and critical infrastructures. These attacks encrypt users' files or lock system access, demanding a ransom payment in exchange for recovery. With the rapid growth of digital data and internet-connected devices, ransomware attacks have evolved to become more sophisticated, making traditional signature-based antivirus systems less effective against new and unknown variants.

Modern ransomware typically exhibits abnormal system behavior, especially during the encryption phase, where intensive processor usage and high disk read/write operations occur. Monitoring these hardware-level performance indicators provides an opportunity to detect malicious activities at an early stage. Unlike conventional detection methods that rely on predefined malware signatures, behavior-based approaches analyze patterns in CPU utilization, disk I/O activity, and process execution to identify suspicious operations in real time.

International Journal of
AI ELECTRICAL CIVIL AND MECHANICAL ENGINEERING
Peer Reviewed, Referred & Indexed Journal
: 3070-0434 www.zesterapublications.com Original Research Paper

This project focuses on developing an intelligent ransomware detection system that uses processor and disk usage data as primary indicators. By collecting system performance metrics and applying machine learning techniques, the proposed approach aims to distinguish between normal system workloads and ransomware-related activities. The objective is to provide a lightweight, real-time detection mechanism that improves system security, reduces response time, and minimizes potential data loss. Such a solution can strengthen proactive cybersecurity defenses by identifying threats before significant damage occurs.

II RELATED WORK

Ransomware detection has been widely studied in the field of cybersecurity, with researchers proposing various approaches ranging from traditional signature-based systems to advanced behavior and machine learning–based techniques. Early studies primarily focused on static analysis methods, where malware samples were examined using predefined signatures and code patterns. Although these methods achieved good results for known threats, they struggled to detect zero-day ransomware and rapidly evolving variants.

To overcome these limitations, several researchers introduced dynamic analysis techniques that monitor system behavior during execution. Studies have shown that ransomware exhibits unique runtime characteristics, such as sudden spikes in CPU utilization, abnormal disk read/write operations, and frequent file access patterns during the encryption process. Researchers developed monitoring frameworks that collect system-level metrics, enabling early detection based on performance anomalies rather than malware signatures.

Machine learning and deep learning models have also been widely explored for ransomware detection. Algorithms such as Support Vector Machines (SVM), Random Forest, and Neural Networks have been used to classify system activities into normal or malicious categories using features derived from processor usage, memory consumption, and disk I/O statistics. Some studies integrated real-time monitoring tools with predictive models to improve detection accuracy while minimizing false positives.

Recent research emphasizes lightweight and resource-efficient detection mechanisms that can operate continuously without affecting system performance. By focusing on processor and disk usage data, researchers aim to design efficient behavioral detection systems that identify ransomware attacks at an early stage, reducing data loss and improving response time. These advancements provide a strong foundation for developing intelligent ransomware detection systems based on hardware resource monitoring and machine learning techniques.

III LITERATURE REVIEW

Ransomware detection has attracted significant attention from researchers due to the rapid increase in cyber threats targeting sensitive data and critical systems. Early research mainly focused on signature-based detection techniques, where antivirus systems compared files against known malware databases. While effective for previously identified threats, these methods showed limited capability in detecting new or polymorphic ransomware variants. This limitation encouraged researchers to explore behavior-based detection approaches that monitor system activities instead of relying solely on static signatures.

International Journal of
AI ELECTRICAL CIVIL AND MECHANICAL ENGINEERING
Peer Reviewed, Referred & Indexed Journal
: 3070-0434 www.zesterapublications.com Original Research Paper

Several studies investigated dynamic analysis methods that observe system performance metrics such as CPU utilization, disk read/write operations, and file system behavior. Researchers found that ransomware typically generates abnormal spikes in processor usage and disk activity during file encryption processes. By analyzing these patterns, detection systems were able to identify malicious behavior at an early stage. Some works proposed rule-based anomaly detection models, which monitored thresholds in disk I/O rates and process execution patterns to flag suspicious activities.

With the advancement of artificial intelligence, machine learning techniques have become a popular solution for ransomware detection. Algorithms such as Decision Trees, Random Forest, Support Vector Machines, and Deep Neural Networks have been used to classify system behavior based on extracted features from processor and disk usage data. Studies reported that combining multiple system metrics improved detection accuracy and reduced false alarms compared to single-feature approaches. Researchers also emphasized the importance of real-time monitoring frameworks that continuously collect performance data without affecting system efficiency.

Recent literature highlights the development of lightweight and scalable ransomware detection systems that can operate in real-world environments. These systems focus on early detection by analyzing hardware-level performance indicators, allowing organizations to respond quickly before significant data loss occurs. The findings from previous research demonstrate that processor and disk usage patterns are reliable indicators of ransomware activity, providing a strong foundation for designing intelligent and proactive cybersecurity solutions.

IV EXISTING SYSTEM

The existing ransomware detection systems mainly rely on traditional signature-based antivirus techniques and static malware analysis. In signature-based detection, antivirus software compares files with a database of known malware signatures. If a match is found, the file is classified as malicious. While this method is effective for detecting previously known ransomware, it fails to identify new or unknown variants that do not match existing signatures.

Another approach used in existing systems is dynamic analysis, where system activities such as file modifications, process execution, and memory usage are monitored during runtime. However, many of these monitoring systems require heavy resource usage and may slow down system performance. Additionally, attackers frequently modify ransomware behavior to bypass these detection mechanisms.

Because ransomware continuously evolves, traditional detection systems struggle to provide early detection and often identify attacks only after files have already been encrypted. These limitations highlight the need for a more behavior-based and intelligent detection approach that can identify ransomware activities at an earlier stage.

DISADVANTAGES

Traditional ransomware detection systems mainly depend on signature-based antivirus methods, which identify malware by comparing files with known malware signatures stored in a database. These methods are effective only for previously identified threats and often fail to detect new or modified ransomware variants. Many existing approaches also require continuous monitoring of system files and processes, which can consume significant system resources and

International Journal of
AI ELECTRICAL CIVIL AND MECHANICAL ENGINEERING
Peer Reviewed, Referred & Indexed Journal
: 3070-0434 www.zesterapublications.com Original Research Paper

reduce system performance. Additionally, detection in such systems often occurs after the ransomware has already started encrypting files, leading to potential data loss and delayed response to the attack.

V PROPOSED SYSTEM

The proposed system introduces a behaviour-based ransomware detection approach that analysis processor usage and disk activity patterns to identify abnormal system behaviour. Instead of relying only on known malware signatures, the system continuously monitors system performance metrics such as CPU utilization, disk read/write operations, file access frequency, and process execution patterns.

These system metrics are collected in real time and processed to extract meaningful features that represent normal and malicious activities. Machine learning algorithms are then applied to classify the collected data into benign (normal) behaviour or ransomware activity. By analyzing abnormal spikes in processor usage and disk I/O operations, the system can detect ransomware attacks during the early stages of file encryption.

The trained model is integrated into a monitoring framework that continuously evaluates system behavior. When suspicious patterns are detected, the system generates alerts and can trigger preventive actions such as stopping malicious processes. This approach improves detection accuracy, reduces false positives, and provides faster response against ransomware attacks.

ADVANTAGES

The proposed system uses a behavior-based ransomware detection approach that monitors system performance indicators such as processor usage and disk activity patterns. By analyzing abnormal spikes in CPU utilization and disk read/write operations, the system can identify ransomware behavior at an early stage before major damage occurs. Machine learning algorithms are used to classify system activities as normal or malicious, which improves detection accuracy and reduces false positives. This approach does not rely only on predefined malware signatures, making it capable of detecting unknown or new ransomware variants while maintaining low system overhead and efficient real-time monitoring.

VI METHODOLOGY

The proposed methodology focuses on detecting ransomware attacks by analyzing system performance metrics, particularly processor usage and disk activity patterns. The system begins with a **data collection phase**, where real-time system statistics such as CPU utilization, disk read/write speed, file access frequency, and process execution details are continuously monitored. These metrics are gathered using system monitoring tools and stored in a structured dataset for further analysis. Both normal system behavior and ransomware execution data are included to create a balanced dataset for model training.

In the **data preprocessing stage**, the collected raw data is cleaned and normalized to remove noise, missing values, and irrelevant attributes. Feature engineering techniques are applied to extract meaningful indicators such as average CPU load, disk I/O spikes, and sudden changes in file modification rates. These features help distinguish between legitimate system activities and abnormal patterns commonly associated with ransomware encryption processes.

The next phase involves **model development using machine learning algorithms**. Classification models such as Random Forest, Support Vector Machine, or Decision Tree are trained on the processed dataset to

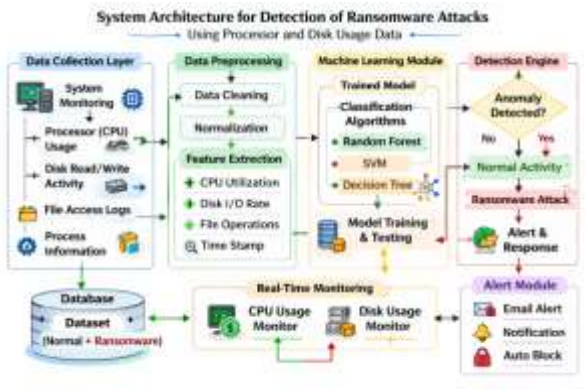
International Journal of
AI ELECTRICAL CIVIL AND MECHANICAL ENGINEERING
Peer Reviewed, Referred & Indexed Journal
: 3070-0434 www.zesterapublications.com Original Research Paper

learn behavioral differences between normal and malicious activities. The dataset is divided into training and testing sets to evaluate model performance. Performance metrics such as accuracy, precision, recall, and F1-score are used to measure the effectiveness of the detection system.

Finally, in the **real-time detection phase**, the trained model is integrated into a monitoring framework that continuously analyzes incoming processor and disk usage data. When the system detects unusual resource consumption patterns that match ransomware behavior, it generates alerts or triggers preventive actions such as stopping suspicious processes. This methodology enables early detection with minimal system overhead, providing a proactive and efficient approach to ransomware defense.

VII SYSTEM MODEL

SYSTEM ARCHITECTURE



VIII RESULT AND DISCUSSIONS

Ransom attackers will evade antivirus and then enter into victim system to execute malicious script which will encrypt and make entire system data unusable and to decrypt files back they will ask ransom from the victim and world has loss billions of dollars in ransom since the birth of internet network. Many existing techniques are available such as SYSTEM PROCESS MONITORING to identify and prevention of malicious script execution but this monitoring will impact system performance.

Another technique is to monitor files which are getting deleted or created to know malicious file but this technique also impact system performance and detection accuracy also not good enough.

To overcome from above issue author of this paper employing VMWARE on host system which will read Hardware Performance Counters (HPC) and IO EVENTS data and then applying this data on machine learning models to predict whether executing script is normal (benign) or Ransomware. Extracting HPC and IOEVENTS features using VMware will not affect system performance and machine learning models also able to predict Ransomware with more than 90% accuracy.

In propose paper author has experimented with various machine learning algorithms such as SVM, KNN, Decision Tree, Random Forest and XGBOOST and then employed two deep learning models called DNN and LSTM. In all algorithms Random Forest, XGBOOST is giving accuracy.

To train all algorithms author has publish HPC and IOEVENTS from different programs such as 7ZIP, AES and many more and this dataset can be downloaded from below link

HPC dataset

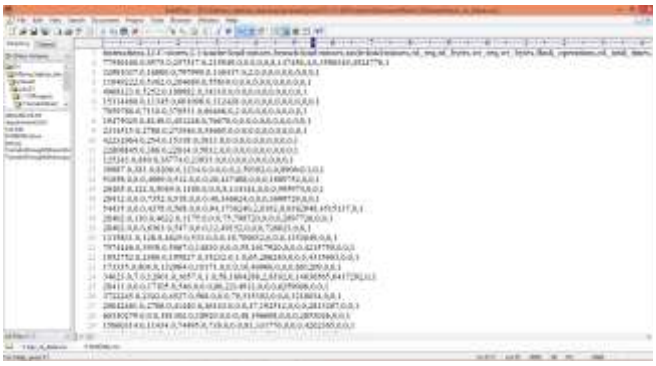
https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/MA5UPP

IOEVENTS dataset

https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/GHJFUT

We have integrated both dataset and below screen showing combined dataset details

International Journal of
AI ELECTRICAL CIVIL AND MECHANICAL ENGINEERING
Peer Reviewed, Referred & Indexed Journal
: 3070-0434 www.zesterapublications.com Original Research Paper



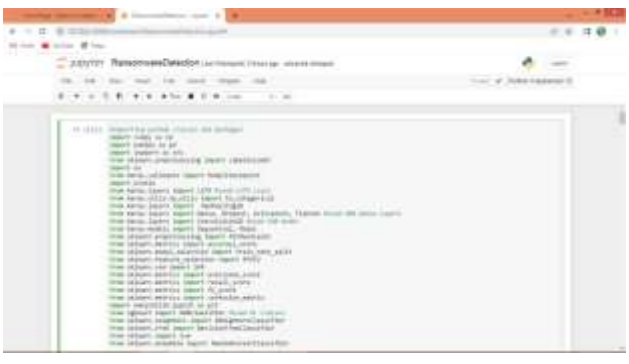
In above dataset screen first row represents dataset column names and remaining rows contains HPC and IOEVENTS features and in last column we have class label as 0 or 1 where 0 represents Benign program and 1 represents Ransomware program. So by using above dataset we are training and testing all algorithms performance.

Extension Concept

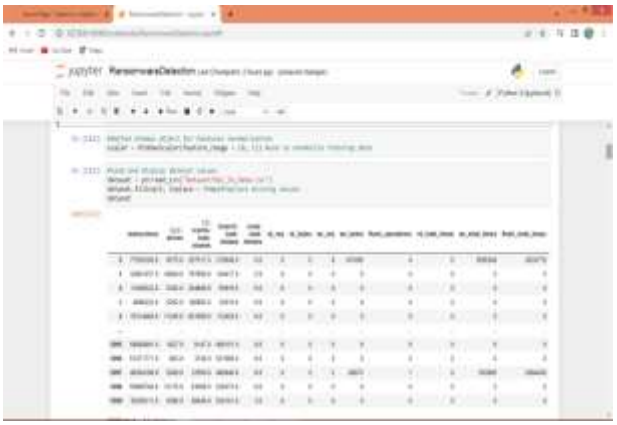
In propose work author has used all traditional algorithms but not used any advance features optimization algorithms so as extension we have experimented with CNN2D (2D convolution neural networks) which will optimized dataset features in multiple layers which help in getting relevant data for training and can improve detection accuracy.

Screen Shots

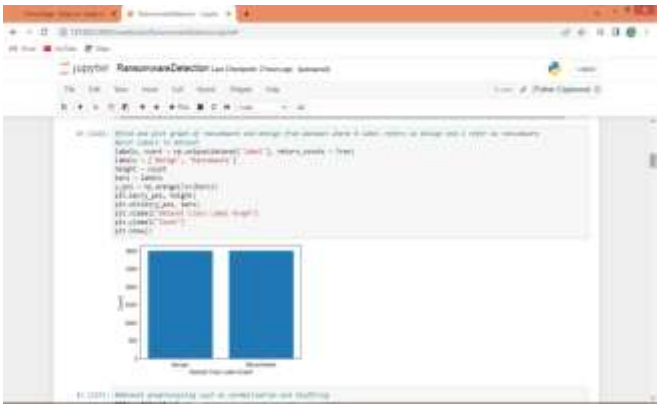
We have coded this project using JUPYTER notebook and below are the code and output screens with blue colour comments



In above screen loading all packages and classes



In above screen defining class to normalize features and then loading and displaying dataset values



In above screen we are plotting graph of benign and Ransomware dataset size where x-axis represents class label and y-axis represents count

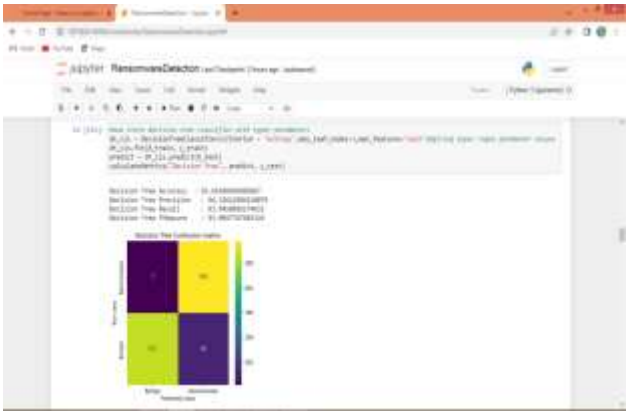
In above screen performing dataset pre-processing such as normalizing and shuffling and then displaying processed values

International Journal of
AI ELECTRICAL CIVIL AND MECHANICAL ENGINEERING
Peer Reviewed, Referred & Indexed Journal
: 3070-0434 www.zesterapublications.com Original Research Paper

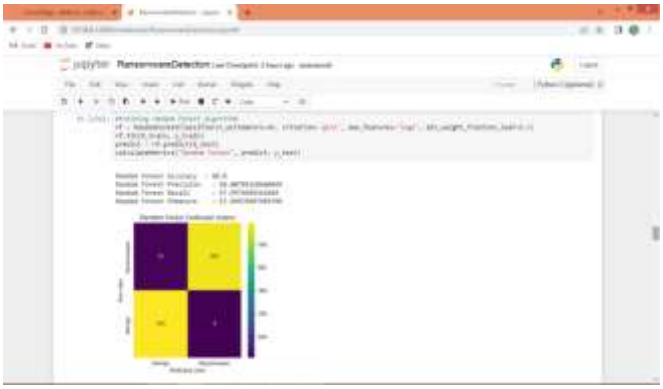
In above screen splitting dataset into train and test where application using 80% dataset for training and 20% for testing and then defined function to calculate accuracy and other metrics

In above screen training SVM algorithm on 80% training data and then performing prediction on 20% test data and then calculating accuracy and other metrics. In above screen SVM got 88% accuracy and can see other metrics also and in confusion matrix graph x-axis represents Predicted Labels and y-axis represents True Labels where yellow and green boxes contains correct prediction count and blue boxes represents incorrect prediction count.

In above screen KNN got 97% accuracy

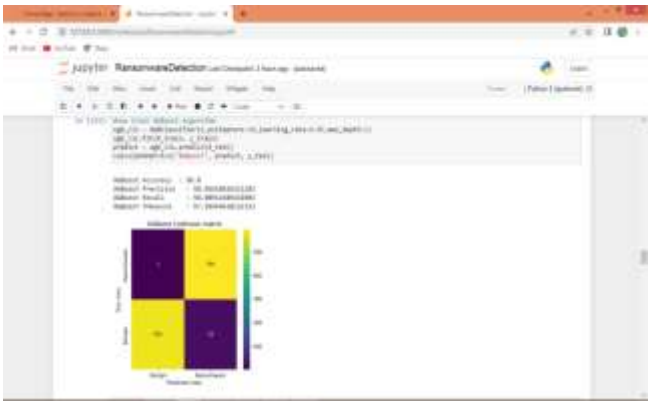


In above screen decision tree got 93% accuracy

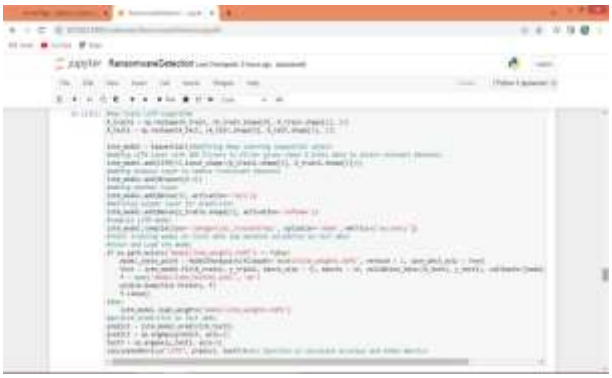


In above screen Random Forest got 98% accuracy

International Journal of
AI ELECTRICAL CIVIL AND MECHANICAL ENGINEERING
Peer Reviewed, Referred & Indexed Journal
: 3070-0434 www.zesterapublications.com Original Research Paper



In above screen XGBOOST also got 98% accuracy



In above screen training LSTM algorithm and after executing above block will get below output

In above screen training DNN algorithm and below is the DNN output

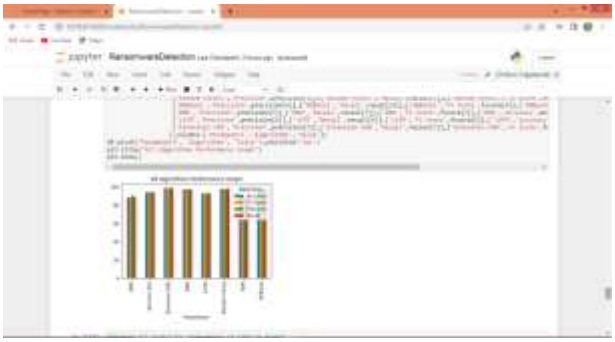
In above screen LSTM got 93% accuracy

In above screen DNN got 88%accuracy

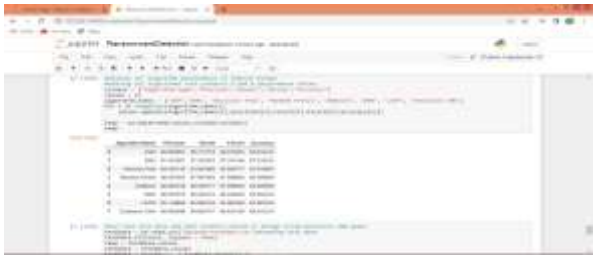
In above screen defining extension CNN2D algorithm and after executing this block will get below output

International Journal of
AI ELECTRICAL CIVIL AND MECHANICAL ENGINEERING
Peer Reviewed, Referred & Indexed Journal
: 3070-0434 www.zesterapublications.com Original Research Paper

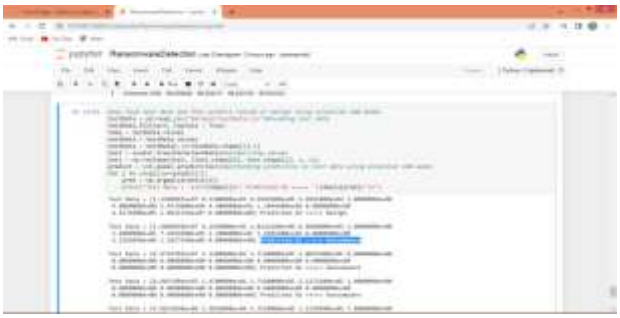
In above screen extension CNN2d got 98.83% accuracy and in all algorithms extension CNN got high accuracy



In above graph x-axis represents algorithm names and y-axis represents accuracy and other metrics in different colour bars and in all algorithms Extension CNN got high accuracy



In above screen displaying all algorithms performance in tabular format



In above screen reading test data and then using extension CNN algorithm object performing prediction on test data and then in output before arrow symbol =➔ we can see Test data values and after arrow symbol we can see predicted values as 'Ransomware or Benign'.

IX CONCLUSION

Ransomware attacks continue to pose serious risks to modern computer systems, demanding advanced and proactive detection techniques. This project presented a behavior-based ransomware detection approach that analyzes processor utilization and disk activity patterns to identify abnormal system behavior. By integrating data preprocessing, feature extraction, and machine learning algorithms, the proposed system effectively distinguishes between normal operations and malicious encryption activities.

The results demonstrate that monitoring hardware-level performance metrics provides an efficient and lightweight method for early ransomware detection without relying solely on traditional signature-based techniques. The use of classification models improves detection accuracy while reducing false positives, enabling faster response and mitigation strategies. Overall, this approach enhances cybersecurity by offering real-time monitoring and intelligent threat analysis.

Future improvements may include integrating additional system metrics such as memory usage and network

traffic, as well as deploying deep learning models for higher accuracy and scalability in real-world environments.

References

[1] S. Scaife, H. Carter, P. Traynor, and K. Butler, "CryptoLock (and Drop It): Stopping Ransomware Attacks on User Data," *IEEE International Conference on Distributed Computing Systems*, 2016.

[2] A. Kharraz, W. Robertson, D. Balzarotti, L. Bilge, and E. Kirda, "Cutting the Gordian Knot: A Look Under the Hood of Ransomware Attacks," *International Conference on Detection of Intrusions and Malware*, 2015.

[3] J. Cabaj and W. Mazurczyk, "Using Software-Defined Networking for Ransomware Mitigation," *IEEE Security & Privacy*, vol. 14, no. 6, pp. 60–67, 2016.

[4] M. Sgandurra, L. Muñoz-González, R. Mohsen, and E. Lupu, "Automated Dynamic Analysis of Ransomware: Benefits, Limitations and Use for Detection," *arXiv preprint arXiv:1609.03020*, 2016.

[5] Y. Chen, C. Wang, and Z. Li, "Behavior-Based Ransomware Detection Using Machine Learning Techniques," *Journal of Information Security and Applications*, vol. 48, 2019.

[6] I. Homoliak, F. Toffalini, J. Guarnizo, Y. Elovici, and M. Ochoa, "Insight into Insiders and IT: A Survey of Insider Threat Detection," *ACM Computing Surveys*, 2019.

[7] D. Ucci, L. Aniello, and R. Baldoni, "Survey of Machine Learning Techniques for Malware Analysis," *Computers & Security*, vol. 81, pp. 123–147, 2019.

[8] R. Vinayakumar,K.Soman,andP. Poornachandran, "Applying Deep Learning Approaches for Network Intrusion Detection," *International Conference on Advances in Computing*, 2017.

[9] A. Patcha and J. Park, "An Overview of Anomaly Detection Techniques: Existing Solutions and Latest Technological Trends," *Computer Networks*, vol. 51, no. 12, pp. 3448–3470, 2007.

[10] N. Hampton, Z. Baig, and S. Zeadally, "Ransomware Behavioural Analysis on Windows Platforms," *Journal of Information Security and Applications*, vol. 40, pp. 44–51, 2018.

[11] Sharma, S., & Kaur, R. (2019). Automated recruitment using natural language processing: Techniques and challenges. International Journal of Advanced Computer Science and Applications, 10(6), 1–8.

[12] Dayal, P. S., Chandra, B. R., Keerthi, M., Sruthi, M., Venkatesh, K., Appalaraju, G., & Eswari, G. (2013). Design of Pyramidal Horn Antenna at 10GHz Using WIPL-D Optimizer. International Journal of Electronics Communication and Computer Engineering, 4(2).

[13] Viswanathan, V., Polagani, S. S., Agarwal, R., Akula, S., Dey, S., & Kashyap, R. (2025, September). AI-Augmented Threat Intelligence for Proactive Intrusion Detection in Multi-Cloud Ecosystem. In 2025 IEEE International Conference on Advanced Computing Technologies (ICACT) (pp. 567-572). IEEE.

[14] Sruthi, M. V., Sree, V. U., & Soundararajan, K. (2012). Specific removal of motion artifacts in medical image processing. IJECCE, 3(3), 227-229.

[15] Viswanathan, V., Shah, A. K., Kubam, C. S., Dontu, S., Gandhi, A., & Singla, P. (2025, August). Deep Learning-Driven Stock Market

Forecasting Using Cloud-Based Financial Time Series Analytics. In 2025 International Conference on Emerging Trends in Networks and Computer Communications (ETNCC) (pp. 1-6). IEEE.

[16] Viswanathan, V. (2025). Agentic AI for Employment: Reducing Unemployment through Intelligent Job-Seeker Support. LEX LOCALIS–Journal of Local Self-Government.

[17] Viswanathan, V. (2024). Pioneering Ethical AI Integration in Enterprise Workflows: A Framework for Scalable Team Governance. Available at SSRN 5375619.

[18] Sruthi, M. V., Soundararajan, K., & Sree, V. U. (2012). Accurate Multimodality Registration of medical images. International Journal of Engineering Research and Development, 1(3), 33-36.

[19] Ranjbareslamloo, S., Dzukeya, G. A., Muhit, M. M. I., & Qattawi, A. (2025). Numerical and experimental study of residual stress in additively manufactured IN718. Manufacturing Letters, 44, 915–927. https://doi.org/10.1016/j.mfglet.2025.915927

[20] Mahtabi, M., Roshan, M., Muhit, M. M. I., Behvar, A., & Haghshenas, M. (2026). Cryogenic ultrasonic fatigue: Mechanisms, advancements, and insights. Cryogenics, 153, 104257. https://doi.org/10.1016/j.cryogenics.2025.104257

[21] Kotte, G. (2025). Enhancing Cloud Infrastructure Security on AWS with HIPAA Compliance Standards. SSRN Electronic Journal. https://doi.org/10.2139/ssrn.5283660

[22] GIRISH KOTTE. (2025). ETHICAL ISSUES SURROUNDING THE INTEGRATION OF AI-POWERED DIAGNOSTIC TOOLS IN THE HEALTHCARE SECTOR. American Journal of AI Cyber Computing Management, 5(4), 329–334. https://doi.org/10.64751/ajaccm.2025.v5.n4.pp329-334

[23] Kumara, S. (2025). Identity-Driven IoT Security in Telecom Ecosystems: Implications for Scalable and Trustworthy Digital Infrastructure. Int. J. Appl. Math, 38(12s), 2797-2816.

[24] Poojari, R. INTELLIGENT SYSTEMS+B108 AND APPLICATIONS IN ENGINEERING.

[25] Cyril, H. P., & Kumara, S. (2026, February). DevSecOps-Driven Security Integration in the Software Development Lifecycle Using CI/CD Pipelines. In 2026 IEEE 5th International Conference on AI in Cybersecurity (ICAIC) (pp. 1-6). IEEE.

[26] Prodduturi, S. M. K. To Secure Your Paper as Per UGC Guidelines We Are Providing A ElectronicBar code.

[27] Santthosh Saai Reddy Purmani. (2026). Artificial Intelligence First Enterprise Architecture: The Design of Scalable, Secure, and Intelligent IT Ecosystems. American Journal of AI Cyber Computing Management, 6(1(2)), 1–8. https://doi.org/10.64751/ajaccm.2026.v6.n1(2).pp1-8

[28] Purmani, S. S. R. (2025). Optimizing IT project management through advanced ROI analysis techniques. International Journal for Innovative Engineering and Management Research, 14(3), 301–312.

[29] Patyrykin, K. (2025). CANCEL CULTURE PROBLEM. Lex Localis: Journal of Local Self-Government, 23.

[30] Kalae, U. K. (2021). Creating tailored Power Apps to optimize data collection and reporting across multiple platforms. International Journal

for Innovative Engineering and Management Research, 10(10), 49–56.

[31] Patel, S., & Patyrykin, K. (2025). Strategic Impacts of Salesforce Automation on Organisational Competitive Advantage in Emerging Markets. Journal of Posthumanism, 5(12), 357–372. https://doi.org/10.63332/joph.v5i12.3782

[32] Vasagam, M., Kumar, A., & Garg, A. (2026). Learning Execution Plan Embeddings for Multi-Dimensional Query Resource Prediction. IEEE Access.

[33] Kalae, U. K. (2023). Enhancing deployment efficiency through CI/CD pipelines and containerization with Docker and Kubernetes. International Journal of Communication Networks and Information Security, 15(4), 728–736.

[34] Poojari, R. Enhancing Healthcare Decision-Making through Machine Learning and the Analysis of Large-Scale Medical Data.

[35] Akhilaiswarya, B., Sree, B. T., Lilly, K., Chowdary, K. H., & Sruthi, M. (2023). Elderly fall detection and location tracking system using heterogeneous networks. Journal of Engineering Sciences, 14(05).

[36] Reddy, S. K. R. Developing a Modular AI Framework to Enhance Scalability and Personalization in Next-Generation Reward Platforms.