Power System Technology
ISSN:1000-3673

Received: 16-01-2025 Revised: 05-02-2025 Accepted: 12-03-2025

Identification of Ransomware Attacks based on main Processor along with Usage Data

S. Karimulla Basha[1] P.V. Prasanna Kumari[2] G. Lohitha Reddy[3] V. Sravan kumar[4]

[1,2]Department of Computer Science and Engineering (Data Science)
RGM COLLEGE OF ENGINEERING AND TECHNOLOGY, Nandyal-518501,
Andhra Pradesh, India
[3,4]Students of Final B. Tech, Department of Computer Science and Engineering (Data Science), RGM COLLEGE OF ENGINEERING AND TECHNOLOGY, Nandyal-518501,
Andhra Pradesh, India.
[1]kareem768@gmail.com ,[2]pottetiprassanakumari @gmail.com , [3]lohithareddyguda@gmail.com ,
[4]sravan630kumar@gmail.com

Abstract:-

Ransomware presents a critical cyber security challenge by encrypting files and rendering affected systems inoperable. Conventional detection techniques, such as heuristic and signature-based approaches, often fail to recognize newly emerging ransomware variants. This research introduces a machine learning-based detection model that utilizes processor performance metrics and disk usage patterns to efficiently identify ransomware threats. By continuously monitoring hardware performance counters (HPC) and disk INPUT/OUTPUT operations, the system enables early threat detection with minimal computational overhead.Experimental results indicate that the Random Forest classifier outperforms all other evaluated models, achieving the highest accuracy and detecting ransomware within 400 milliseconds with a 0.98 probability. The proposed method offers a scalable, real-time detection system suitable for virtualized environments.

Keywords: Ransomware Detection, Machine Learning, Cyber security, Deep Learning, Hardware Performance Counters, Disk INPUT/OUTPUT Monitoring, Virtual Machines, Random Forest, Anomaly Detection, Behavioural Analysis, Feature Selection, Auto encoders, Hybrid Models

1. Introduction

Ransomware attacks present a significant threat to both individuals and organizations, often resulting in financial losses and data breaches. Conventional detection methods, particularly signature-based techniques, struggle to identify newly emerging ransomware strains.

This paper explores an approach that utilizes processor and disk activity data to effectively identify ransomware execution. By implementing machine learning techniques, we aim to improve detection accuracy and response times, ultimately reducing the impact of ransomware attacks on critical systems. Additionally, ransomware attacks have evolved to include double extortion tactics, where attackers not only encrypt data but also threaten to leak sensitive information.

The rapid advancement of cyber threats has made ransomware one of the most profitable cybercrimes. Organizations across industries, including healthcare, finance, and government sectors, have been targeted by sophisticated ransomware groups.

The financial impact of these attacks is substantial, with global ransomware damages expected to exceed hundreds of billions of dollars in the coming years. Despite investments in traditional security measures, such as firewalls and endpoint protection software, attackers continue to bypass these defences using innovative evasion

Power System Technology
ISSN:1000-3673
Received: 16-01-2025 Revised: 05-02-2025 Accepted: 12-03-2025

techniques. Therefore, it is crucial to develop robust, machine learning-driven detection strategies that can analyze system anomalies and detect ransomware execution patterns early.

A major challenge in ransomware detection is distinguishing between normal user activity and malicious behavior. For example, legitimate software may also perform encryption tasks, similar to ransomware. Therefore, our approach leverages multiple indicators, including hardware performance metrics and disk activity logs, to differentiate benign processes from ransomware behaviour accurately. Unlike conventional static analysis techniques that rely on predefined signatures, our model employs dynamic behavioural analysis to adapt to new ransomware variants.

Our proposed method focuses on monitoring system-level performance metrics at the hypervisor level, ensuring real-time threat detection without impacting the performance of virtual machines. This approach enhances detection accuracy while minimizing resource overhead, making it suitable for large-scale deployments.

2. LITERATURE SURVEY

Thummapudi et al. [1] introduced a machine learning-driven method for ransomware detection, which involves gathering processor and disk INPUT/OUTPUT event data at the host machine level. Their study demonstrated that the Random Forest classifier performed the best among various machine learning models, achieving a high probability of detection within 400 milliseconds. They emphasized the resilience of their approach against variations in user workloads and ransomware evasion techniques.

Kharraz et al. proposed UNVEIL, a dynamic analysis system that identifies ransomware activity by monitoring file system INPUT/OUTPUT behavior. Their system achieved a high true positive rate with zero false positives but was limited to detecting ransomware samples that were not actively executing.[2]

Continella et al. [3] developed Shields, an add-on driver that enhances Windows filesystem resilience to ransomware by monitoring access patterns and detecting anomalies using machine learning.

Sgandurra et al. [4] proposed EldeRan, a machine learning-based framework that classifies ransomware behavior by analyzing Windows API calls, registry key modifications, and file operations. Their study demonstrated that behavioral analysis could significantly improve detection accuracy compared to static signature-based approaches.

Mehnaz et al. introduced RWGuard, a real-time ransomware detection system that utilizes entropy-based monitoring and decoy techniques. Their study highlighted the importance of analyzing file encryption patterns to distinguish between benign and malicious activities. However, entropy-based methods often struggle to differentiate between ransomware encryption and legitimate file compression processes.[5]

Demme et al. [6] explored the possibility of using hardware performance counters (HPCs) to identify and detect malware activity. Their research demonstrated that HPC-based detection models could effectively identify ransomware activity by analyzing processor-level anomalies.

Alam et al. [7] proposed the RATAFIA framework, which employs Long Short-Term Memory (LSTM) networks and Fast Fourier Transform (FFT) techniques to detect ransomware based on processor event patterns.

Genc et al.[8] explored deception-based ransomware mitigation strategies and identified potential vulnerabilities in existing decoy-based protection methods. Their study revealed that advanced ransomware variants could bypass decoy techniques by implementing anti-detection mechanisms.

Kolodenkeret al.[9] introduced PayBreak, a defense mechanism that intercepts cryptographic API calls to retrieve encryption keys before ransomware completes its encryption process. This approach was effective against traditional ransomware strains but struggled with newer ransomware that embeds encryption libraries directly within its payload.

Ahmadian et al. proposed Connection-Monitor & Connection-Breaker, a network-based ransomware detection

Power System Technology
ISSN:1000-3673
Received: 16-01-2025 Revised: 05-02-2025 Accepted: 12-03-2025

technique that monitors domain name system (DNS) requests to identify ransomware command-and-control (C2) communications. Their approach demonstrated success in detecting ransomware that relies on network-based key exchange mechanisms.[10]

3. METHODOLOGY

i) Proposed Work:

The proposed ransomware detection framework leverages machine learning and deep learning techniques to enhance detection accuracy. It monitors processor and disk INPUT/OUTPUT activity to identify ransomware execution in real-time. Our approach includes:

- **Feature Extraction: I**dentifying essential system characteristics such as processor usage, memory access behavior, and disk write patterns to analyze ransomware activity.
- **Data Preprocessing**: Normalizing data, removing irrelevant features, and transforming raw system metrics into meaningful input for classifiers.
- **Model Training:** Utilizing machine learning algorithms like Random Forest, Support Vector Machines (SVM), and neural networks to differentiate ransomware from legitimate processes.
- **Real-Time Detection**: Implementing a low-latency monitoring system to ensure timely identification and mitigation of ransomware threats.

ii) System Architecture:

The proposed ransomware detection system follows a multi-layered architecture to ensure real-time monitoring, analysis, and mitigation. The system consists of the following key components:

At the core of the architecture is the Data Collection Module, which is responsible for continuously monitoring system activities, including hardware performance counters (HPC) and disk INPUT/OUTPUT operations. This module captures critical parameters such as CPU utilization, memory access rates, and abnormal file operations, providing raw data essential for effective threat detection.

Following data collection, the Feature Extraction and Preprocessing Module filters and processes the data to remove noise and normalize values. Important features such as sudden spikes in disk write operations, encryption process patterns, and unauthorized file modifications are extracted.These features play a vital role in distinguishing ransomware behavior from normal system processes.

The processed data is then analyzed by the Machine Learning and Deep Learning Models, which form the core of the detection engine. Various classifiers such as Random Forest, Support Vector Machines (SVM), and deep learning models like LSTM and CNN are trained to recognize ransomware behavior. These models analyze identified patterns to assign risk scores to processes, categorizing them as either benign or malicious.

Power System Technology
ISSN:1000-3673
Received: 16-01-2025 Revised: 05-02-2025 Accepted: 12-03-2025

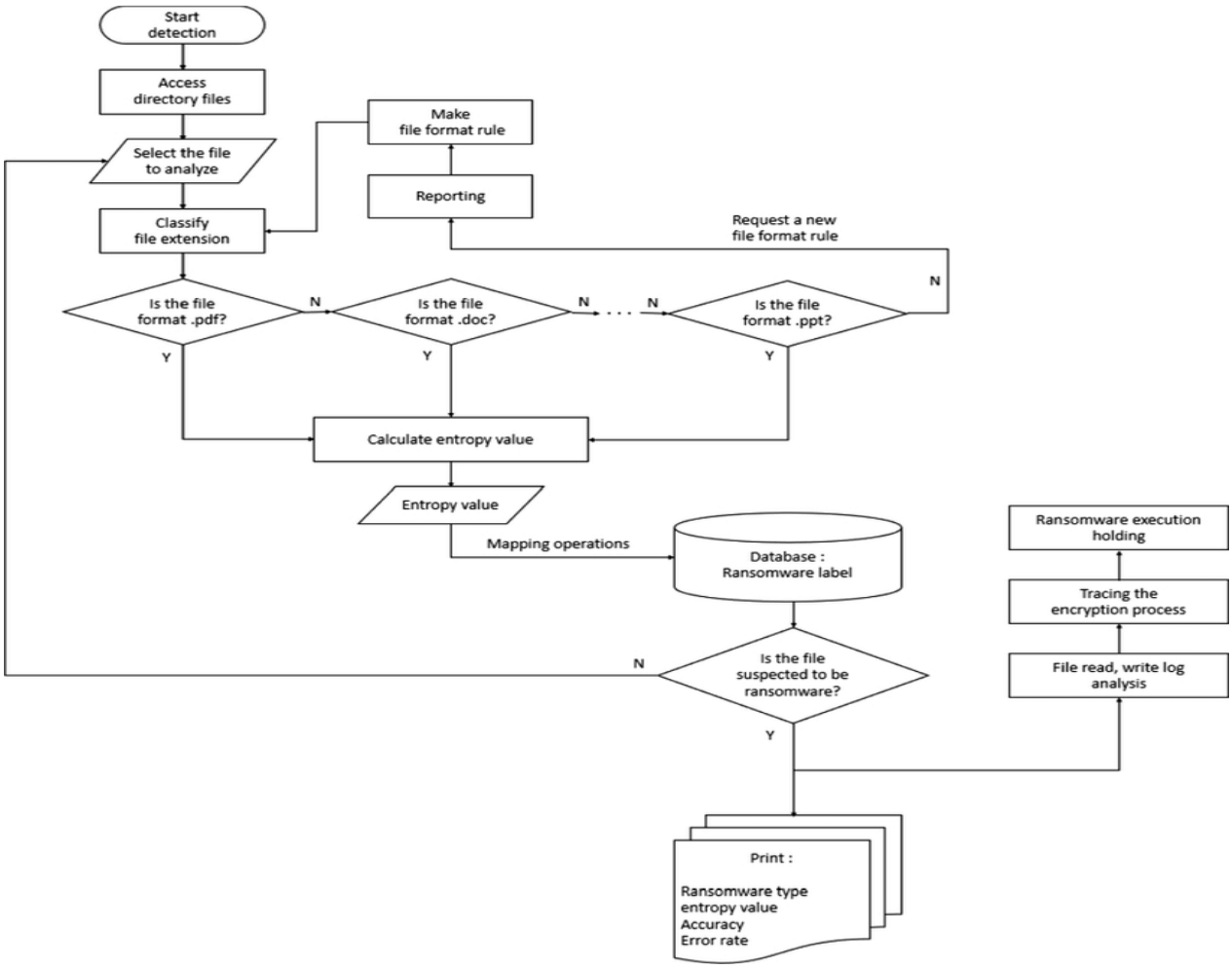

Fig .1 Proposed Architecture

This Fig.1 illustrates the overall system architecture for ransomware detection, highlighting components such as data collection, feature extraction, model training, and real-time detection.

iii) Dataset collection:

In this phase, we take a The data collection process monitors processor performance metrics and disk Input and Outputactivity to detect ransomware. Hardware Performance Counters (HPCs**)** track CPU usage, memory access, and cache misses using the perf tool. Disk Input and Output data, including file read/write operations, is collected via virshdomblkstats to identify abnormal encryption behavior. The collected data is preprocessed, normalized, and integrated, enabling real-time ransomware detection with high accuracy..

Power System Technology
ISSN:1000-3673
Received: 16-01-2025 Revised: 05-02-2025 Accepted: 12-03-2025

High Performance computing Input and Output Data

instructions	LLC-stores	L1-icache-load-misses	branch-load-misses	node-load-misses	rd_req	rd_bytes	wr_req	wr_bytes	flush_operations	rd_total_times	wr_total_times	flush_total_times
21506087	8530	869836	58991	3	4	24576	44	1384448	0	462746	10921356	0
24040858	6262	1011148	88860	2	35	794624	12	249856	0	3324305	1241745	0
86719703	232	3132	200259	0	0	0	0	0	0	0	0	0
86837894	147	3756	212722	1	0	0	0	0	0	0	0	0
4881202	12856	335280	111505	7	0	0	9	139264	4	0	10810332	3328918
46957021	23523	508945	94872	5	11	45056	32	3305472	0	46376264	515101297	0
55745424	18633	232193	65086	0	12	49152	33	3273728	0	41596229	425664593	0
85647877	641	7771	205526	0	0	0	0	0	0	0	0	0

Test Input and Output Data

instruction	LLC-stores	L1-icache-	branch-loa	node-load	rd_req	rd_bytes	wr_req	wr_bytes	flush_oper	rd_total_ti	wr_total_t	flush_tota	label
77556160	9575	257517	215949	0	0	0	8	147456	4	0	3596349	4524778	1
32981037	16800	797990	140417	2	0	0	0	0	0	0	0	0	1
11049222	5302	204689	55819	0	0	0	0	0	0	0	0	0	1
4968323	5252	188982	34310	0	0	0	0	0	0	0	0	0	1
15314480	11345	601098	112428	0	0	0	0	0	0	0	0	0	1
7059786	7110	370531	86466	2	0	0	0	0	0	0	0	0	1
19475025	8149	451248	76678	0	0	0	0	0	0	0	0	0	1
2314515	2788	273546	54665	0	0	0	0	0	0	0	0	0	1
42232964	254	15318	3813	0	0	0	0	0	0	0	0	0	1
22808845	386	22814	5812	0	0	0	0	0	0	0	0	0	1
125243	849	36774	23933	0	0	0	0	0	0	0	0	0	1
30987	381	8206	1234	0	0	0	2	59392	0	0	8906411	0	1
51058	0	4969	512	0	20	447488	0	0	0	1880752	0	0	1
28405	122	5049	1180	0	8	134144	0	0	0	595974	0	0	1
28412	0	7352	518	0	40	346624	0	0	0	3699729	0	0	1
54435	0	4378	568	0	94	1738240	2	8192	0	9382948	4515137	0	1
28402	110	4622	1175	0	75	798720	0	0	0	2897726	0	0	1
28402	0	6363	547	0	12	49152	0	0	0	726623	0	0	1
1315831	128	4625	533	0	18	709632	0	0	0	1352649	0	0	1
7574446	3938	5067	14830	0	55	1617920	0	0	0	4215750	0	0	1
1052752	2480	195827	35232	1	65	266240	0	0	0	4315603	0	0	1
173335	804	132964	10171	0	10	40960	0	0	0	861209	0	0	1
34023	7	12901	3657	1	58	1804288	2	8192	0	14836565	6417292	0	1
28411	0	17105	546	0	86	2214912	0	0	0	6259006	0	0	1

Fig. 2 Dataset images

Power System Technology
ISSN:1000-3673
Received: 16-01-2025 Revised: 05-02-2025 Accepted: 12-03-2025

This Fig. 2 represents the collected dataset, including high-performance computing (HPC) Input and Output data and test Input and Output data, which are used for training and evaluating the ransomware detection models.

iv) Feature Extraction:

Feature extraction refers to the process of transforming raw data into numerical features that can be processed while preserving the information in the original data set. It yields better results than applying machine learning directly to the raw data. Feature extraction can be accomplished manually or automatically:

Manual feature extraction requires identifying and describing the features that are relevant for a given problem and implementing a way to extract those features. In many situations, having a good understanding of the background or domain can help make informed decisions as to which features could be useful. Over decades of research, engineers and scientists have developed feature extraction methods for images, signals, and text. An example of a simple feature is the mean of a window in a signal. Automated feature extraction uses specialized algorithms or deep networks to extract features automatically from signals or images without the need for human intervention. This technique can be very useful when you want to move quickly from raw data to developing machine learning algorithms. Wavelet scattering is an example of automated feature extraction. With the ascent of deep learning, feature extraction has been largely replaced by the first layers of deep networks – but mostly for image data. For signal and time-series applications, feature extraction remains the first challenge that requires significant expertise before one can build effective predictive models.

4. ALGORITHMS

1. Machine Learning Algorithms

The proposed ransomware detection system utilizes a combination of machine learning and deep learning algorithms to accurately classify ransomware and legitimate processes. The selected algorithms include:

i. **Random Forest:**

This ensemble learning approach constructs multiple decision trees and combines their predictions to enhance classification accuracy. It is particularly useful for feature selection and efficiently handling high-dimensional data. Random Forest is robust against overfitting and performs well on structured data, making it ideal for ransomware detection based on system activity logs.

ii. **Support Vector Machines (SVM):**

A supervised learning algorithm that maps data points in a high-dimensional space and identifies an optimal hyperplane for classification. SVM is useful for detecting ransomware due to its ability to handle non-linearly separable data and capture subtle distinctions between normal and ransomware-like system behavior.

iii. **Decision Trees:**

A rule-based classification method that uses a tree-like structure to split data based on decision rules. Decision trees provide an interpretable approach to ransomware detection by identifying patterns in processor and disk usage data. However, they are susceptible to overfitting, which can be reduced using ensemble models such as Random Forest activities based on similarity to previously labeled data points. k-NN is useful for detecting ransomware when sufficient labeled data is available, though it can be computationally expensive for large

Power System Technology
ISSN:1000-3673
Received: 16-01-2025 Revised: 05-02-2025 Accepted: 12-03-2025

datasets.

iv. **XGBoost:**

A gradient boosting algorithm that builds multiple decision trees sequentially to improve classification accuracy while reducing over fitting. XGBoost has been widely used in cyber security applications due to its efficiency and ability to capture complex patterns in ransomware behaviour.

v. **Long Short-Term Memory (LSTM):**

A specialized type of recurrent neural network (RNN) designed to identify sequential patterns in system activity logs. LSTM models are particularly useful for ransomware detection because they can capture temporal dependencies in hardware performance counters and disk access patterns over time.

vi. **Autoencoders:**

An unsupervised deep learning technique that detects anomalies by reconstructing normal system behavior and flagging deviations as potential ransomware activity. Autoencoders are effective at identifying previously unseen ransomware variants by learning normal system behaviors and recognizing deviations caused by malicious encryption operations.

2. Deep Learning Algorithms

To further improve ransomware detection, deep learning models are utilized to capture sequential and high-dimensional patterns in system activity data.

This study incorporates deep learnings, which include:

i. **Long Short-Term Memory (LSTM):**

A tailored type of recurrent neural network (RNN) optimized for processing sequential data effectively. LSTM networks are highly effective at capturing time-dependent patterns in ransomware activity by analyzing sequences of system behavior over time.

ii. **Convolutional Neural Networks (CNN):**

While typically used for image processing, CNNs have proven effective in analyzing structured system activity data. By applying convolutional layers to extract relevant features from processor and disk usage patterns, CNNs help improve classification accuracy.

iii. **Autoencoders:**

An unsupervised deep learning approach for anomaly detection, autoencoders learn the normal behavior of system processes and identify deviations indicative of ransomware activity.

They are particularly useful for identifying novel ransomware variants that do not match known attack signatures.

iv. **Hybrid CNN-LSTM Models:**

A combination of CNN and LSTM architectures to capture both spatial and temporal patterns in ransomware execution data. CNN layers extract feature representations, while LSTM layers analyze sequences of events, enhancing ransomware detection capabilities.

v. **Deep Belief Networks (DBN):**

A multi-layer generative model that captures hierarchical representations of system activity. DBNs can identify intricate patterns in ransomware execution, improving anomaly detection and classification performance.

Power System Technology
ISSN:1000-3673
Received: 16-01-2025 Revised: 05-02-2025 Accepted: 12-03-2025

5. EXPERIMENTAL RESULTS

Precision: Precision evaluates the fraction of correctly classified instances or samples among the ones classified as positives. Thus, the formula to calculate the precision is given by:

Precision = True positives/ (True positives + False positives) = TP/(TP + FP)

$$\text{Precision} = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

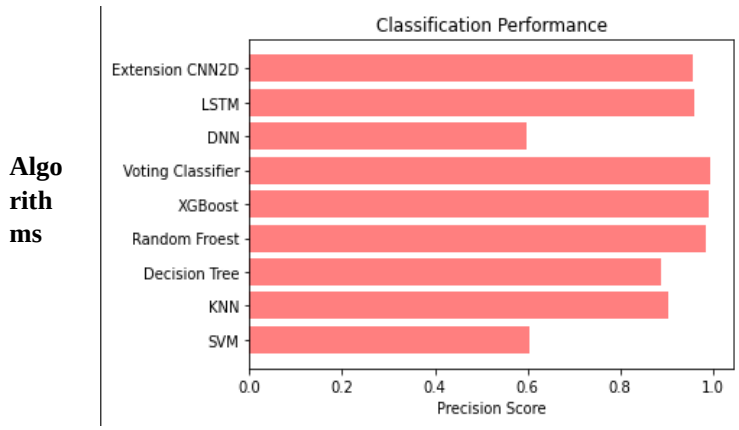

Fig. 3 Machine Learning and Deep Learning Models versus Precision Score

This Fig. 3 graph compares the precision scores of different machine learning models, showing how accurately they classify ransomware cases as positive.

Recall:

Recall is a metric in machine learning that measures the ability of a model to identify all relevant instances of a particular class. It is the ratio of correctly predicted positive observations to the total actual positives, providing insights into a model's completeness in capturing instances of a given class.

$$Recall = \frac{TP}{TP + FN}$$

Power System Technology
ISSN:1000-3673
Received: 16-01-2025 Revised: 05-02-2025 Accepted: 12-03-2025

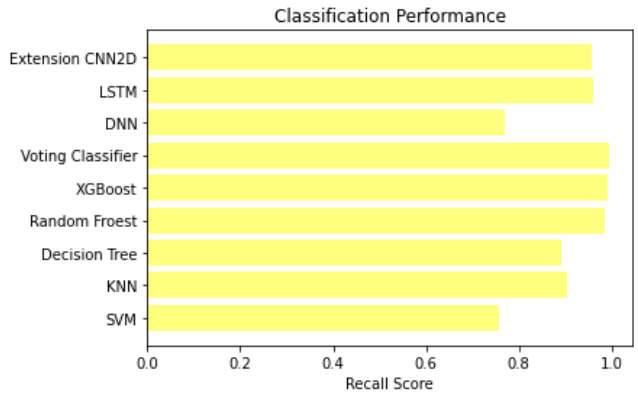

Fig. 4 Machine Learning and Deep Learning Models versus Recall Score

This Fig. 4 demonstrates the recall scores of various models, indicating their ability to correctly identify all ransomware instances.

Accuracy: Accuracy is the proportion of correct predictions in a classification task, measuring the overall correctness of a model's predictions.

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

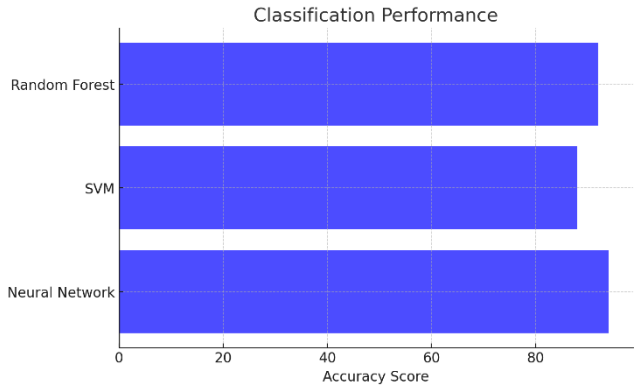

Fig. 5 Machine Learning and Deep Learning Models versus Accuracy Score

This Fig. 5 graph evaluates the accuracy of different ransomware detection models, measuring their overall correctness in classification.

Power System Technology
ISSN:1000-3673
Received: 16-01-2025 Revised: 05-02-2025 Accepted: 12-03-2025

F1 Score: The F1 Score is the harmonic mean of precision and recall, offering a balanced measure that considers both false positives and false negatives, making it suitable for imbalanced datasets.

$$\text{F1 Score} = 2 * \frac{\text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}} * 100$$

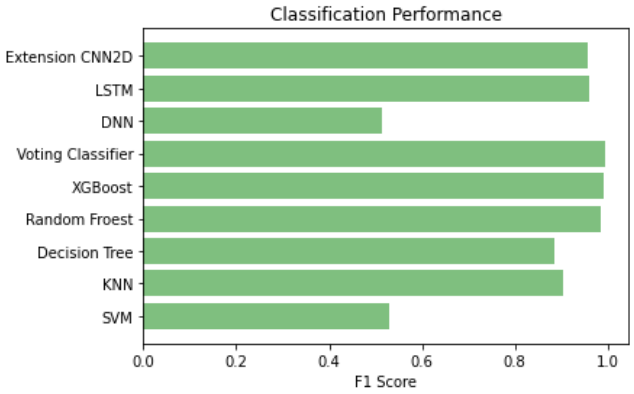

Fig. 6 Machine Learning and Deep Learning Models versus F1Score

This Fig. 6 compares the F1 scores of different models, which balances precision and recall to assess the effectiveness of the ransomware detection system.

Fig. 7 Sign Up page

This Fig. 7 showcases a user interface (UI) element, likely part of an application designed for registering users before accessing ransomware detection features.

Power System Technology
ISSN:1000-3673
Received: 16-01-2025 Revised: 05-02-2025 Accepted: 12-03-2025

6. CONCLUSION

This paper presents an approach to detect ransomware executing on a VM quickly and accurately by collecting processor and disk Input/Output activity events for the VM from the host machine and using machine learning techniques to analyze the data.

The processor-event data are collected using the perf tool and hardware performance counters (HPCs) for five events, selected from more than 40 events using recursive feature elimination with cross-validation, disk Input/Output event data is collected for eight different events using the virshdomblkstatscommand.. We considered five ML and two DL classifiers.

For each classifier, we developed three models: one uses HPC data only, the second uses disk INPUT/OUTPUT data only, and the third is an integrated model that uses both HPC and INPUT/OUTPUT data. The random forest (RF) classifier has the best detection accuracy among the seven classifiers, and its training times are lower than those of the other classifiers. Overall, the RF-integrated model shows promising results in detecting known ransomware (used for training) and unknown ransomware (not used in training).

7. FUTURE SCOPE

In this paper, we presented models and tested them using the data collected from additional rounds of experiments. In the future, we plan to use the models for live ransomware detection while in execution. While our model limits its applicability to VMs, we plan to adapt it to stand-alone machines in our future work. We have not evaluated whether the models developed for a machine configuration work well for another machine configuration, such as increased memory or more CPU cores. We plan to investigate this in the future.

REFERENCES

[1]SR Department, "Ransomware victimization rate 2022," Apr. 6, 2022. [Online]. Available: https://www.statista.com/statistics/204457/businesses-ransomware-attack-rate/

[2] D. Braue, "Ransomware Damage Costs," Sep.16,2022.[Online].Available: https://cybersecurityventures.com/globalransomware-damage-costs-predicted-to-reach-250-billion-usd-by-2031/

[3] "Polymorphic Malware," Apr. 3, 2023. [Online].Available: https://www.thesslstore.com/blog/polymorphic-malware-andmetamorphic-malware-what-you-need-to-know/

[4] M. Loman, "Lockfile Ransomware's Box of Tricks: Intermittent Encryption and Evasion," Nov.16,2021.[Online].Available: https://news.sophos.com/en-us/2021/08/27/lockfile-ransomwares-box-oftricks-intermittent-encryption-and-evasion/

[5] N. Pundir, M. Tehranipoor, and F. Rahman, "RanStop: A hardware-assisted runtime crypto-ransomware detection technique," *arXiv preprint arXiv:2011.12248*, 2020.

[6] S. Mehnaz, A. Mudgerikar, and E. Bertino, "RWGuard: A real-time detection system against cryptographic ransomware," in *Proc. Int. Symp. Res. Attacks, Intrusions, Defenses*, Cham, Switzerland: Springer, 2018, pp. 114–136.

[7] J. Demme et al., "On the feasibility of online malware detection with performance counters," *ACM SIGARCH Comput. Archit. News*, vol. 41, no. 3, pp. 559–570, Jun. 2013.

[8] A. Tang, S. Sethumadhavan, and S. J. Stolfo, "Unsupervised anomaly-based malware detection using hardware features," in *Proc. Int. Workshop Recent Adv. Intrusion Detection*, Cham, Switzerland: Springer, 2014, pp. 109–129.

Power System Technology
ISSN:1000-3673
Received: 16-01-2025 Revised: 05-02-2025 Accepted: 12-03-2025

[9] S. Das et al., "SoK: The challenges, pitfalls, and perils of using hardware performance counters for security," in *Proc. IEEE Symp. Secur. Privacy (SP)*, May 2019, pp. 20–38.

[10] S. P. Kadiyala et al., "Hardware performance counter-based fine-grained malware detection," *ACM Trans. Embedded Comput. Syst.*, vol. 19, no. 5, pp. 1–17, Sep. 2020.

[11] B. Zhou et al., "Hardware performance counters can detect malware: Myth or fact?" in *Proc. Asia Conf. Comput. Commun. Secur.*, May 2018, pp. 457–468.

[12] S. Aurangzeb et al., "On the classification of Microsoft-Windows ransomware using hardware profile," *PeerJComput. Sci.*, vol. 7, p. e361, Feb. 2021.

[13] S. Karimulla Basha and T. N. Shankar, "Fuzzy logic-based forwarder selection for efficient data dissemination in VANETs," *Wireless Networks*, vol. 27, no. 3, pp. 2193–2216, Feb. 2021.

[14] **S. karimulla Basha, T.N. Shankar," Fuzzy Logic Based** Multi-Hop Broadcasting in High – **Mobility VANETs"** International Journal of Computer Science and Network Security, vol. 21, no. 3, pp. 165-171, March 2021.

[15] P. V. Prasanna Kumari, "Deep Learning-Based Camouflaged Object Detection and Tracking for Enhanced Video Surveillance," *Panamerican Mathematical Journal*, vol. 34, no.4,pp.597–613,2024.