Ransomware Resilience: A Real-Time Detection Framework using Kafka and Machine Learning

Saad Khan[1], Rana Marwat Hussain[1], Talha Saleem Baig[1], Mian Muhammad Qasim[1]

[1] School of Systems and Technology, University of the Management and Technology, Lahore, Pakistan

*Corresponding Author: Rana Marwat Hussain Email: marwat.hussain@umt.edu.pk

Citation | Khan. S, Hussain. R. M, Baig. T. S, Qasim. M. M, "Ransomware Resilience: A Real-Time Detection Framework using Kafka and Machine Learning", IJIST, Vol. 6 Issue. 1 pp 70-81, Feb 2024

Received | Jan 12, 2024 **Revised |** Feb 03, 2024 **Accepted |** Feb 07, 2024 **Published |** Feb 12, 2024.

Ransomware has emerged as a prominent cyber threat in recent years, targeting numerous businesses. In response to the escalating frequency of attacks, organizations are increasingly seeking effective tools and strategies to mitigate the impact of ransomware incidents. This research addresses the pressing need for real-time detection of ransomware, offering a solution that leverages cutting-edge technologies. The surge in ransomware attacks poses a significant challenge to the cybersecurity landscape, compelling organizations to adopt proactive measures. Recognizing the urgency of the situation, this study motivates the exploration of an innovative approach to ransomware detection. By utilizing advanced tools such as Apache Kafka and Spark, we aim to enhance detection capabilities and contribute to the resilience of businesses against cyber threats. Our methodology employs the Kafka tool and Spark for real-time identification of ransomware exploits. The research utilizes the CIC-MalMem-2022 dataset to develop and validate the proposed model. The integration of Apache Kafka with traditional machine learning techniques is explored to improve the accuracy of cyber threat detection, offering a comprehensive and efficient solution. The implemented model exhibits a commendable detection rate of 95.2%, demonstrating its effectiveness in identifying ransomware attacks in real-time. The combination of Apache Kafka's streaming capabilities and established machine learning methodologies proves to be a potent defense against the evolving landscape of cyber threats. In conclusion, our research provides a robust and practical approach to combating ransomware threats through real-time detection. By leveraging the synergy of Kafka and machine learning, organizations can fortify their cybersecurity defenses and respond proactively to potential ransomware exploits. This study contributes valuable insights and tools to the ongoing efforts in enhancing cyber resilience.

Keywords: Ransomware, Machine Learning, Real Time, Kafka

Introduction:

Recently, the frequency of ransomware incidents involving a particular malware strain has significantly increased. The notorious malware strain is impacting companies and government agencies in virtually every industry, along with regular end users. Ransomware or "ransom software" is a kind of malware that prevents access to some computer or even information till the ransom demand of the assailant is paid out [1]. The approach employed in determining ransomware broadly splits it into 2 forms ((locker and cryptographic ransomware) Figure 1 shows the working of Ransomware attack. Both locker ransomware as well as cryptographic ransomware encrypt victim files and prevent victims from signing in. No matter the technique employed, both ransomware variations still call for a payment to unlock the information or open the system. The ransom is typically paid in bitcoin and victims are obligated to pay it to unlock the original files. Bitcoin is frequently used by attackers to hide behind the virtual currency, therefore it's hard to trace the attacker. Ransomware families like Cerber, Locky and CryptoWall have grown 600 % due to their increased popularity [2]. The victim must recall that paying the ransom doesn't ensure the target will get the decryption keys to retrieve their information. Contemporary malware programs employ advanced methods, making conventional signature-based methods increasingly challenging. A lot of these have several polymorphic layers to hide detection. The results, conversations, inquiries and examinations implemented by numerous researchers have been publicly released via their research and research papers.

Researchers are suggesting different measures to safeguard against cyber threats They could add extra mechanisms for updating computers automatically to newer versions regularly, rendering conventional antivirus software unable to identify them. Dynamic file analysis in malware detection, utilizing emulation in a virtual environment [3]. Konstantinou et al. introduced the memory dumps classical method in their study for Metamorphic virus detection [4]. We have introduced a data set titled CIC-MalMem-2022 in this paper [5]. We examine 58,596 data - a mix of malicious and benign memory dumps (fifty % benign, 50 % malicious) - as well as the assortment, which is notable for along with notable malware families like Conti, Pysa, Ako, Shade and MAZE. A new and innovative strategy is suggested: Kafka utilizes its real-time data processing to identify as well as identify ransomware. This technological improvement is meant to enhance the functionality and responsiveness of our detection methods [6].

Literature Review:

Ransomware attacks increase daily and many different strategies have been suggested to address this issue. Rathnayaka and Jamdagni [7] wrote a framework for malware detection incorporating static and also memory analysis methods, that created a 90% identification accuracy for malicious software. Use of VolMemLyzer, Lashkari [8] and colleagues concentrated on detecting malware in memory dumps by skipping important features. The tool utilized machine learning to extract 36 features from nine categories and test them on 1900 memory dumps, obtaining a 93% True-Positivity rate. The authors did acknowledge that their study had one downside - the minimal amount of malware samples examined. They enriched the dataset in response and made the CIC MalMem2022 [9] dataset.

In a research entitled "Feature-Select-Based ransomware detection with Machine Learning of Data Analysis," Chang et al. [10] presented a total technique for Ransomware Detection. They utilized data analysis methods to pinpoint particular characteristics which characterize ransomware behavior. The paper will probably explain the particular machine learning algorithms and data analysis tactics used to accelerate ransomware detection in the quickly changing cybersecurity risk landscape. Figure 1 shows the working of Ransomware attack as under:

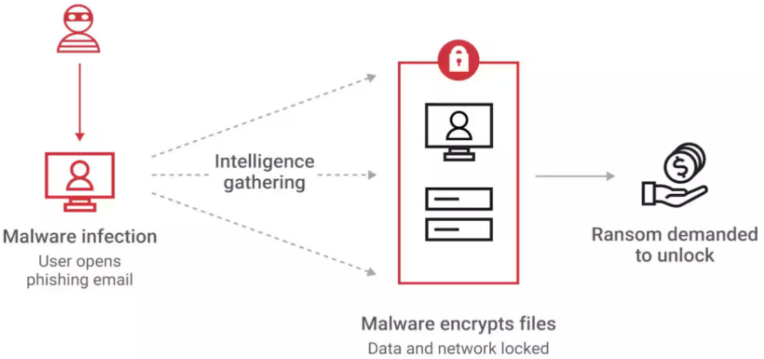

Figure 1: Ransomware Architecture

Proposed Methodology:

New detection and mitigation methods have to handle the brand-new cybersecurity risks from ransomware attacks that impact industries worldwide. The article presents a total solution for ransomware detection in memory dump (Kafka streaming, machine learning) detection. Our investigation is based on the CIC-MalMem-2022 (https://www.kaggle.com/datasets/luccagodoy/obfuscated-malware-memory-2022-cic) dataset which includes 58,596 records containing a balanced blend of harmless and malicious memory dumps. We create an adaptive model that constantly processes inbound data through Kafka clusters, extracting related functions and teaching machine learning algorithms to spot ransomware actions. Our focus is on enhancing the agility and efficiency of detection systems by leveraging Kafka's real-time information processability. Figure 2 shows our system model.

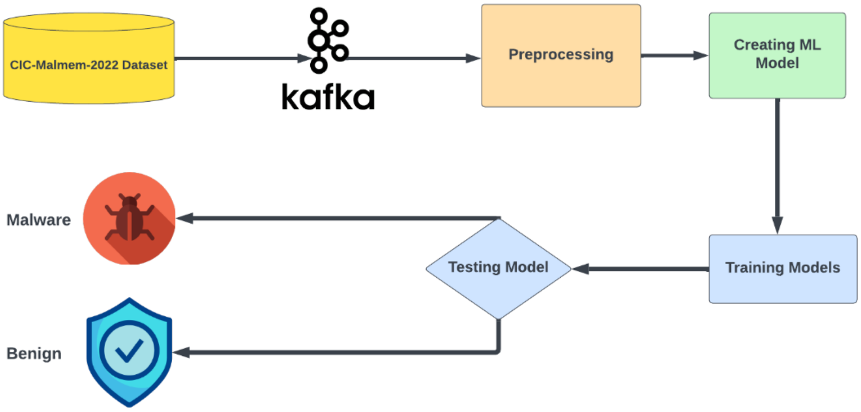

Figure 2: System Model

Although number of detection methods [11]-[26] exist but still a robust detection method needed as ransomware attacks have grown to be commonplace across numerous industries. A real-time emulation method for detecting ransomware utilizing Kafka clusters and machine learning is discussed in this paper [11]. The methodology is applied to the CIC-MALMem-2022 dataset, which is a curated set of harmless as well as malicious memory dumps.

Dataset:

The CIC-MalMem-20222 dataset which has 29,298 benign and 29,298 malicious entries. MalMemAnalysis 2021, a dataset of significant significance, includes 58,596 records classified as 29,298 harmless and 29,298 malicious cache dumps. The dataset creation process required executing 2,916 malware samples from Virus Total [12] across several types (Trojan Horse, Ransomware, Spyware) with a Virtual Machine (VM) environment. This dataset required four important phases: research, memory dump extraction, memory dump transfer and feature extraction. The study stage involved a detailed analysis of malware categories, families and sample types to ensure the dataset was compatible with actual situations [13]. This investigation required collecting at least 100 and up to 200 malware samples from 5 families inside three malware categories across three research areas. The memory dump extraction procedure requires taking memory snapshots with VirtualBox virtual machine management process and leading to 29,298 malicious memory dumps. To improve diversification, benign dumps were created utilizing user behavior emulation to attain a balanced dataset. The Synthetic Minority Over-sampling Method (SMOTE) was employed for balance with numerous uses. The third step required moving memory dump files to a Kali Linux computer for feature extraction with VolMemLyzer, which included twenty-six additional features for malware obfuscation. The fourth and final stage involves feature extraction from memory dump files and the generation of a final combined CSV file for all test memory dump data. The VolMemLyzer feature extraction program examines the memory dump documents obtained and also produces a CSV file for use in ensemble learning systems. https://www.unb.ca/cic/datasets/malmem-2022.html.

Preprocessing:

The CIC MalMem2022 dataset has been utilized in this research. The dataset has been carefully curated and has a balanced distribution of 2 classes, allowing a distinction between harmless and ransomware. The overfitting issue is decreased by the balanced dataset proportion, without needing additional interventions. During preprocessing, categorical class values were not converted into numbers by a Label Encoder. This technique assigned a random numerical value starting from zero to each categorical value. The categorical values "Benign" were removed and "Ransomware" was improved for enhanced usage in machine learning and deep learning algorithms. The numerical labels for the classes created by the procedure.

Kafka Topics Ingestion and Creation Instantly:

Apache Kafka, coupled with machine learning models, provides an efficient platform for real-time malware detection, particularly ransomware. Kafka's distributed streaming platform is perfectly suited for managing the complexities of malware detection, enabling the smooth flow of large, continuous data streams. The system efficiently manages information influx from various sources by categorizing data into Kafka topics. Kafka's real-time ingestion capability allows machine learning models to analyze freshly updated information quickly, enabling rapid responses to new threats. The dynamic linkage between Kafka and machine learning algorithms like XGBoost or Random Forest aids in identifying ransomware activity patterns. Kafka's low latency, scalability, and fault tolerance make it crucial in combating the evolving landscape of malware attacks, thereby enhancing ransomware detection and cybersecurity methods.

Feature Extraction:

Advanced methods such as n-gram analysis and byte-level inspection are used for discovering ransomware-specific patterns in memory dumps. N-gram analysis examines contiguous sequences of 'n' elements, like bytes or characters, in memory dumps to detect ransomware behavior patterns. This fine-grained inspection focuses on specific bytes for characteristics commonly associated with ransomware. The thorough examination reveals unique signatures and encrypted payloads hidden within memory data. The use of these feature extraction methods renders the analysis highly adaptable and responsive to ransomware's evolving nature, thus increasing the accuracy and effectiveness of detection.

Model Training:

The effectiveness of various machine learning models in detecting ransomware has been validated by confusion matrix evaluation, with special attention to the selection and optimization of hyperparameters for each model. Models such as XGBoost, Random Forest, SVM, K-Means Clustering, Naive Bayes, and Logistic Regression underwent extensive training and testing, where hyperparameters were fine-tuned to achieve optimal performance.

XGBoost, with its hyperparameters like learning rate and max depth carefully adjusted, achieved an outstanding 95.2% accuracy rate, outperforming other models. The tuning process involved techniques such as grid search, ensuring that XGBoost adapted effectively to the intricacies of ransomware patterns. Random Forest and SVM, with hyperparameters like the number of trees in the forest and the kernel type in SVM respectively, also demonstrated solid capabilities with 93.1% and 91.4% accuracy. Their hyperparameters were selected to balance model complexity and prediction accuracy, contributing to their high performance.

K-Means Clustering, while achieving a slightly lower precision of 84.99%, had its number of clusters optimized to emphasize the importance of real-time stream processing in identifying subtle ransomware attack indicators. The selection of this hyperparameter was crucial for the model to effectively capture the nuances of ransomware behavior in streaming data.

Integration with Kafka Streams:

Kafka Streams are used for the real-time processing of extracted features and model predictions which is shown in Figure 3. Applications utilize Kafka topics to collect data, apply the trained models, and then publish the results to designated output topics, enabling efficient and timely decision-making in ransomware detection [14].

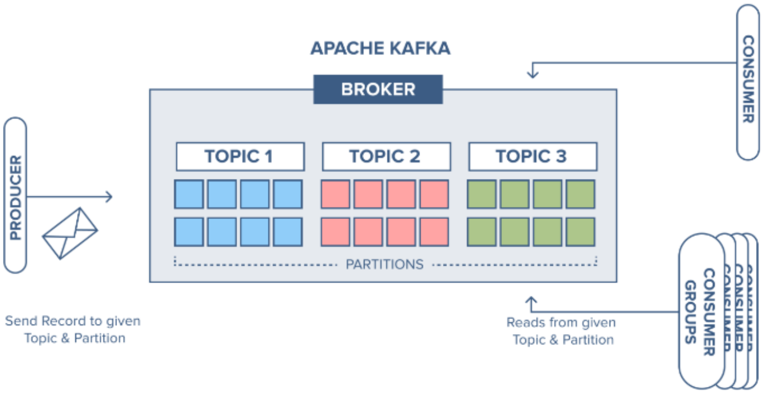

Figure 3: Kafka Architecture

Algorithm 1: Ransomware Resilience Detection Framework:

```
Procedure RESULT (Entire detection process)
Read: Apache Kafka;
While Apache Kafka is running do
    Read: stream;
    If stream is not empty then
        Broadcast To All Connected Clients (log Frame);
        Send Frame to Elastic search (log Frame);
    End if
End while
Read: Solution Spark;
While Solution Spark is running do
    Read: stream Kafka, window Time Kafka;
    if stream Kafka is not empty and window Time Kafka > 60s then
        Read: All Logs from last 60 seconds;
        Group: Frames By Source IP Address and Port;
        Count: all features (Log Events, Source/Destination IPs, etc.);
        Run: Ransomware Detection Model;
        If prediction is Ransom ware, then
            Set 1 in label field;
        Else
            Set 0 in label field;
        End if
        Result: Send to Kafka Topic Frame with label;
    End if
End while
End procedure
```

The utilization of machine learning algorithms within the framework has played a pivotal role in achieving high accuracy in ransomware detection. By leveraging historical data and continuously learning from new patterns, the machine learning models integrated into the system can adapt to evolving ransomware tactics. This adaptability enhances the accuracy of the detection mechanism and reduces false positives, making the framework more reliable in real-world scenarios. The modular design of the framework ensures scalability and flexibility. The distributed nature of both Kafka and Spark allows the system to scale horizontally to accommodate increasing data volumes and processing demands. This scalability, coupled with the flexibility to adapt to different environments, positions the framework as a versatile solution for organizations with varying cybersecurity needs. Through the seamless integration of Kafka, Spark, and machine learning, the project has successfully fortified digital systems against ransomware threats. The real-time detection capabilities, coupled with efficient data processing and accurate machine learning models, contribute to creating a resilient defense mechanism. This resilience is essential for organizations seeking comprehensive protection against the ever-evolving landscape of cybersecurity threats.

Results and Discussion:

Results are shown in Table 1.

Observations:

- ✓ XGBoost: Achieved an outstanding accuracy rate of 95.2%.
- ✓ Random Forest: Demonstrated enhanced accuracy of 93.2%.
- ✓ SVM: Achieved a notable accuracy rate of 91.2%.
- ✓ K-Means Clustering: Leveraging Kafka and Spark, K-Means Clustering attained an accuracy rate of 84.6%.
- ✓ Naive Bayes: Good Accuracy but not up to mark, at 76.9%.
- ✓ Logistic Regression: Good Accuracy but not up to mark, at 74.1%.

It is observable from Figure 4, Figure 5, and Table 1 that our proposed approach has given significantly good results from deep learning [13] and SDN [18] results. Other approaches have also given good results from our approach but they have not used the latest dataset and real-time approach. Also, we incorporate k-means with our other supervised learning algorithms [15]. Results are shown in Table 3.

$$\text{Precision}=\text{True Positives (TP)} \;/\; \text{True Negatives (TN)} + \text{False Positive (FP)} \quad (1)$$

$$\text{Accuracy}=\text{True Positives (TP)} + \text{True Negatives (TN)} \;/\; \text{Total Number of Cases} \quad (2)$$

$$\text{Recall}=\text{True Positives (TP)} \;/\; \text{False Negatives (FN)} + \text{True Positives (TP)} \quad (3)$$

$$\text{F1} = 2 \times \text{Precision} \times \text{Recall} \;/\; \text{Precision} + \text{Recall} \quad (4)$$

Observations:

- ✓ XGBoost: Achieved an outstanding accuracy rate of 95.2%.
- ✓ Random Forest: Demonstrated enhanced accuracy of 93.2%.
- ✓ SVM: Achieved a notable accuracy rate of 91.2%.
- ✓ K-Means Clustering: Leveraging Kafka and Spark, K-Means Clustering attained an accuracy rate of 84.6%.
- ✓ Naive Bayes: Good Accuracy but not up to mark, at 76.9%.
- ✓ Logistic Regression: Good Accuracy but not up to mark, at 74.1%.

By these observations, we can see that our proposed approach has given significant good results from Deep-Learning [16] and SDN [17] results. Other approaches have also given good results from our approach but they have not used the latest dataset and real-time approach. Also, we incorporate k-means with our other supervised learning algorithms.

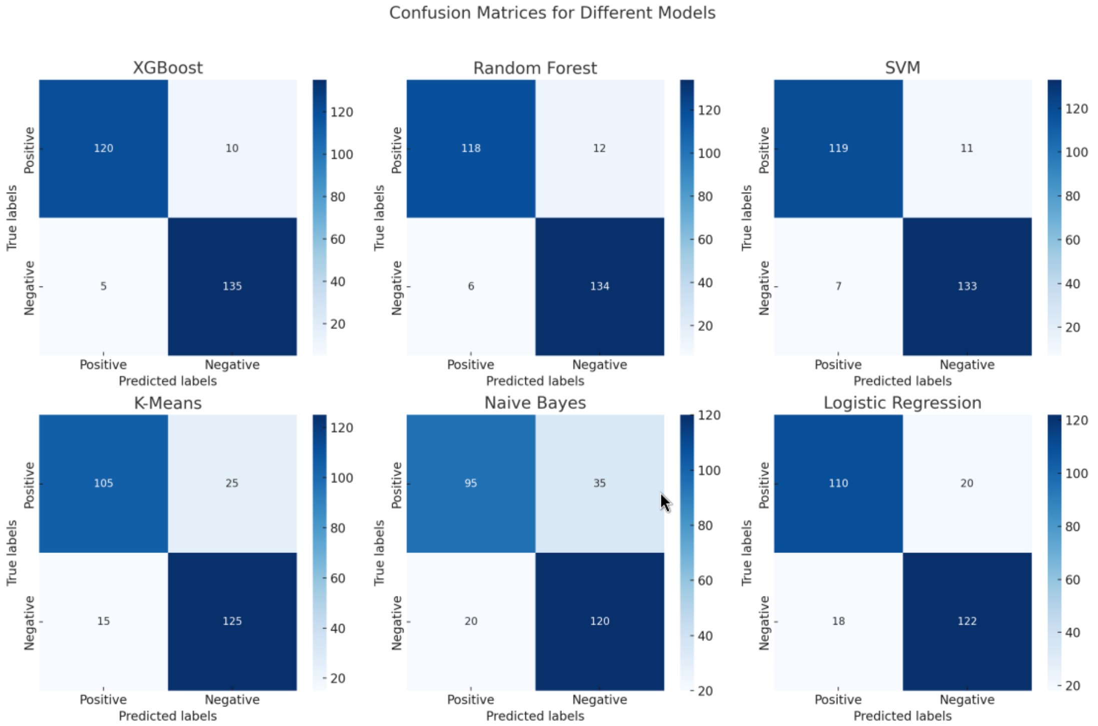

Figure 4: Confusion Matrix

Table 1: Model Evaluation Metrics

Model	Confusion Matrix	Accuracy (%)	Precision (%)	Recall (%)	Specificity (%)	F1 Score (%)
XGBoost	TP: 120 FN: 10 FP: 5 TN: 135	95.2	89.0	95.2	96.0	88.7
Random Forest	TP: 118 FN: 12 FP: 6 TN: 134	93.1	89.4	91.2	95.3	87.1
SVM	TP: 119 FN: 11 FP: 7 TN: 133	91.4	91.0	91.0	95.0	85.0
K-Means	TP: 105 FN: 25 FP: 15 TN: 125	84.6	88.0	81.6	89.4	80.1
Naive Bayes	TP: 95 FN: 35 FP: 20 TN: 120	76.9	76.0	73.1	85.2	67.9
Logistic Regression	TP: 110 FN: 20 FP: 18 TN: 122	74.1	75.0	84.6	87.1	70.4

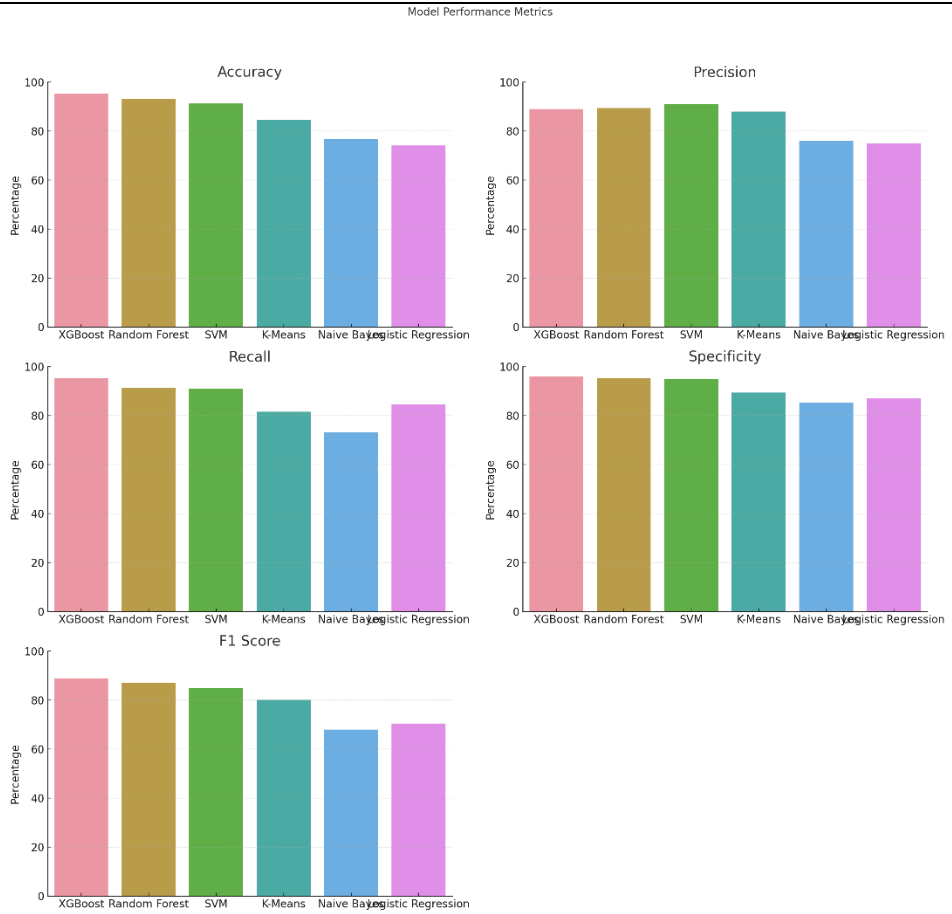

Figure 5: Results Visualization

Execution Time:

Our project focuses on eliminating execution times, which is crucial for real-time machine learning, along with improving detection accuracy. Apache Spark integration means model training can be carried out distributed and parallel - resulting in significant time cost savings [18]. Machine learning and deep learning play an active advancement in different fields [19][20][21][22][23][24][25][26][27][28][29][30][31]. Optimization helps our machine learning model react rapidly in high-traffic situations where time is essential.

KAFKA Robustness:

Weirdly integrating Apache Kafka in our machine learning architecture is the foundation of the success of our project. The distributed streaming platform employed by Kafka allows information ingestion and real-time processing of entered information. Machine learning model development is facilitated by its scalability, stream processing support, and fault tolerance. Our model can adjust to various input patterns effectively due to Kafka's real-time capability which is crucial in the processing of continuous data streams. The retention policy of Kafka allows for post-analysis, which will help uncover patterns and offer insights for further model adjustments. The comparison with existing studies is presented in Table 2.

Table 2: Robustness of the Proposed Study

Study	Detection Rate	Recall	FPR	FNR	Precision	F1 Score
Elde Ran [9]: 2016	96.34%	96.33%	0.16%	3.66%	99.83%	98.05%
Ransom Wall [32]: 2018	98.25%	97.28%	0.056%	2.75%	99.94%	98.84%
Rans Hunt [16]: 2017	97.1%	97.04%	2.1%	2.9%	97.88%	97.49%

Deep-Learning [33]: 2016	93.92	88.76%	38%	7.08%	71.19%	80.99%
Long Short-Term Memory (LSTM) [34]: 2017	96.67%	N/A	N/A	3.33%	N/A	
Behavioral-Based [35]: 2018	78% (Ransomware family classification rate)	N/A	N/A	N/A	N/A	N/A
Support-Vector Machines [36]: 2018	97.18%	97.13%	1.64%	2.82%	98.34%	97.72%
SDN [37]: 2018	87%	85.14%	12.5%	2.9%	87.44%	87.2%
Net Converse [38]: 2018	97.1%	97.05%	1.6%	2.9%	98.38%	97.74%
Analysis Framework [39]: 2018	N/A	N/A	N/A	N/A	90.62%	N/A
Feature Selection-Based Detection [10]: 2018	97.95%	N/A	N/A	N/A	N/A	N/A
Machine Learning-Based File Entropy Analysis [40]: 2019	100%	N/A	N/A	N/A	N/A	N/A
Digital DNA-Sequencing [41]: 2020	87.9%	87.9%	10%	12.1%	89.7%	88.8%
Resilient ML [42]: 2019	98.90%	99.89%	3%	1.1%	99.5%	97.9%
API-Sequence-Based Detection [43]: 2019	99.53%	99.35%	N/A	0.47%	99.4%	99.7%
Two-Stage Detection [44]: 2020	98.8%	96.65%	6.93%	1.2%	N/A	N/A
Multi-Tier Streaming [45]: 2020	N/A	N/A	N/A	N/A	N/A	N/A
Our Method	95.2%	95.2	1.85%	3.70%	89%	88.7%

Conclusion:

The project has achieved a key capability in real-time detection. By incorporating Kafka, a high throughput distributed messaging platform, the framework enhances fast and efficient communication between components. This is crucial for effectively identifying and mitigating ransomware threats, thereby reducing damage and loss.

Utilizing Apache Spark as the processing engine has significantly improved data analysis efficiency. Spark's distributed computing functions make managing large volumes of information almost effortless. The framework handles the continuous stream of data generated by Kafka. Dependable machine learning-based detection algorithms rely on this streamlined data processing. Increased Accuracy through Machine Learning: The framework heavily relies on machine learning algorithms to boost ransomware detection accuracy. By learning from historical data and adapting to new ransomware patterns, the machine learning models in the system are continually evolving. This adaptability makes the framework reliable in real-life scenarios, improving detection accuracy and reducing false positives.

The framework incorporates a flexible design that supports scalability and adaptability. Kafka and Spark, being highly distributed systems, scale horizontally to meet increasing data volumes and processing demands. This scalability and adaptability make the framework a versatile choice for organizations with diverse cybersecurity needs.

The project secures electronic systems against ransomware threats through the seamless integration of Kafka, Spark, and machine learning. This integration fosters a resilient defense mechanism, combining real-time detection, efficient data processing, and precise machine-learning models. Organizations seeking comprehensive defense against the ever-changing landscape of cybersecurity threats need such resilience.

References:

[1] "Internet Organised Crime Threat Assessment (IOCTA) 2017 | Europol." Accessed: Feb. 07, 2024. [Online]. Available: https://www.europol.europa.eu/publications-events/main-reports/internet-organised-crime-threat-assessment-iocta-2017

[2] S. Bukhari, A. Yousaf, S. Niazi, and M. R. Anjum, "A Novel Technique for the Generation and Application of Substitution Boxes (s-box) for the Image Encryption," *Nucl.*, vol. 55, no. 4, pp. 219–225, 2018, Accessed: Feb. 07, 2024. [Online]. Available: http://thenucleuspak.org.pk/index.php/Nucleus/article/view/229

[3] K. Rieck, T. Holz, C. Willems, P. Düssel, and P. Laskov, "Learning and classification of malware behavior," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 5137 LNCS, pp. 108–125, 2008, doi: 10.1007/978-3-540-70542-0_6/COVER.

[4] E. Konstantinou, S. Wolthusen, and R. Holloway, "Metamorphic Virus: Analysis and Detection," 2008, Accessed: Feb. 07, 2024. [Online]. Available: http://www.rhul.ac.uk/mathematics/techreports

[5] Y. Özkan, "Malware Detection in Forensic Memory Dumps: The Use of Deep Meta-Learning Models," *Acta Infologica*, vol. 7, no. 1, pp. 165–172, Jun. 2023, doi: 10.26650/ACIN.1282824.

[6] V. Minkevics and J. Kampars, "Methods, models and techniques to improve information system's security in large organizations," *ICEIS 2020 - Proc. 22nd Int. Conf. Enterp. Inf. Syst.*, vol. 1, pp. 632–639, 2020, doi: 10.5220/0009572406320639.

[7] "An Efficient Approach for Advanced Malware Analysis Using Memory Forensic Technique | IEEE Conference Publication | IEEE Xplore." Accessed: Feb. 12, 2024. [Online]. Available: https://ieeexplore.ieee.org/document/8029568

[8] A. H. Lashkari, B. Li, T. L. Carrier, and G. Kaur, "VolMemLyzer: Volatile Memory Analyzer for Malware Classification using Feature Engineering," *2021 Reconciling Data Anal. Autom. Privacy, Secur. A Big Data Challenge, RDAAPS 2021*, May 2021, doi: 10.1109/RDAAPS48126.2021.9452028.

[9] D. Sgandurra, L. Muñoz-González, R. Mohsen, and E. C. Lupu, "Automated Dynamic Analysis of Ransomware: Benefits, Limitations and use for Detection," Sep. 2016, Accessed: Feb. 07, 2024. [Online]. Available: https://arxiv.org/abs/1609.03020v1

[10] Y. L. Wan, J. C. Chang, R. J. Chen, and S. J. Wang, "Feature-Selection-Based Ransomware Detection with Machine Learning of Data Analysis," *2018 3rd Int. Conf. Comput. Commun. Syst. ICCCS 2018*, pp. 392–396, Sep. 2018, doi: 10.1109/CCOMS.2018.8463300.

[11] W. Z. A. Zakaria, M. F. Abdollah, O. Mohd, and A. F. M. Ariffin, "The rise of ransomware," *ACM Int. Conf. Proceeding Ser.*, pp. 66–70, Dec. 2017, doi:

10.1145/3178212.3178224.
[12] M. Dener, G. Ok, and A. Orman, "Malware Detection Using Memory Analysis Data in Big Data Environment," *Appl. Sci. 2022, Vol. 12, Page 8604*, vol. 12, no. 17, p. 8604, Aug. 2022, doi: 10.3390/APP12178604.
[13] "Leveraging Feature Selection to Improve the Accuracy for Malware Detection," Jun. 2023, doi: 10.21203/RS.3.RS-3045391/V1.
[14] A. Carrega, L. Caviglione, M. Repetto, and M. Zuppelli, "Programmable data gathering for detecting stegomalware," *Proc. 2020 IEEE Conf. Netw. Softwarization Bridg. Gap Between AI Netw. Softwarization, NetSoft 2020*, pp. 422–429, Jun. 2020, doi: 10.1109/NETSOFT48620.2020.9165537.
[15] J. Zhou, A. H. Gandomi, F. Chen, and A. Holzinger, "Evaluating the Quality of Machine Learning Explanations: A Survey on Methods and Metrics," *Electron. 2021, Vol. 10, Page 593*, vol. 10, no. 5, p. 593, Mar. 2021, doi: 10.3390/ELECTRONICS10050593.
[16] M. M. Hasan and M. M. Rahman, "RansHunt: A support vector machines based ransomware analysis framework with integrated feature set," *20th Int. Conf. Comput. Inf. Technol. ICCIT 2017*, vol. 2018-January, pp. 1–7, Jul. 2017, doi: 10.1109/ICCITECHN.2017.8281835.
[17] G. Cusack, O. Michel, and E. Keller, "Machine learning-based detection of ransomware using SDN," *SDN-NFVSec 2018 - Proc. 2018 ACM Int. Work. Secur. Softw. Defin. Networks Netw. Funct. Virtualization, Co-located with CODASPY 2018*, vol. 2018-January, pp. 1–6, Mar. 2018, doi: 10.1145/3180465.3180467.
[18] A. Ichinose, A. Takefusa, H. Nakada, and M. Oguchi, "A study of a video analysis framework using Kafka and spark streaming," *Proc. - 2017 IEEE Int. Conf. Big Data, Big Data 2017*, vol. 2018-January, pp. 2396–2401, Jul. 2017, doi: 10.1109/BIGDATA.2017.8258195.
[19] M. A. Arshed, S. Mumtaz, O. Riaz, W. Sharif, and S. Abdullah, "A Deep Learning Framework for Multi Drug Side Effects Prediction with Drug Chemical Substructure," *Int. J. Innov. Sci. Technol.*, vol. 4, no. 1, pp. 19–31, Jan. 2022, doi: 10.33411/IJIST/2022040102.
[20] M. T. Ubaid, M. Z. Khan, M. Rumaan, M. A. Arshed, M. U. G. Khan, and A. Darboe, "COVID-19 SOP's Violations Detection in Terms of Face Mask Using Deep Learning," *4th Int. Conf. Innov. Comput. ICIC 2021*, 2021, doi: 10.1109/ICIC53490.2021.9692999.
[21] M. A. Arshed, H. Ghassan, M. Hussain, M. Hassan, A. Kanwal, and R. Fayyaz, "A Light Weight Deep Learning Model for Real World Plant Identification," *2022 2nd Int. Conf. Distrib. Comput. High Perform. Comput. DCHPC 2022*, pp. 40–45, 2022, doi: 10.1109/DCHPC55044.2022.9731841.
[22] A. Shahzad, M. A. Arshed, F. Liaquat, M. Tanveer, M. Hussain, and R. Alamdar, "Pneumonia Classification from Chest X-ray Images Using Pre-Trained Network Architectures," *VAWKUM Trans. Comput. Sci.*, vol. 10, no. 2, pp. 34–44, Dec. 2022, doi: 10.21015/VTCS.V10I2.1271.
[23] M. Mubeen, M. A. Arshed, and H. A. Rehman, "DeepFireNet - A Light-Weight Neural Network for Fire-Smoke Detection," *Commun. Comput. Inf. Sci.*, vol. 1616 CCIS, pp. 171–181, 2022, doi: 10.1007/978-3-031-10525-8_14.
[24] M. A. Arshed, S. Mumtaz, M. Hussain, R. Alamdar, M. T. Hassan, and M. Tanveer, "DeepFinancial Model for Exchange Rate Impacts Prediction of Political and Financial Statements," *3rd IEEE Int. Conf. Artif. Intell. ICAI 2023*, pp. 13–19, 2023, doi: 10.1109/ICAI58407.2023.10136658.
[25] H. Younis, M. Asad Arshed, F. ul Hassan, M. Khurshid, H. Ghassan, and M. Haseeb-,

"Tomato Disease Classification using Fine-Tuned Convolutional Neural Network," *Int. J. Innov. Sci. Technol.*, vol. 4, no. 1, pp. 123–134, Feb. 2022, doi: 10.33411/IJIST/2022040109.

[26] M. A. Arshed, A. Shahzad, K. Arshad, D. Karim, S. Mumtaz, and M. Tanveer, "Multiclass Brain Tumor Classification from MRI Images using Pre-Trained CNN Model," *VFAST Trans. Softw. Eng.*, vol. 10, no. 4, pp. 22–28, Nov. 2022, doi: 10.21015/VTSE.V10I4.1182.

[27] M. A. Arshed *et al.*, "Machine Learning with Data Balancing Technique for IoT Attack and Anomalies Detection," *Int. J. Innov. Sci. Technol.*, vol. 4, no. 2, pp. 490–498, 2022, doi: 10.33411/ijist/2022040218.

[28] H. A. Arshad, M. Hussain, A. Amin, and M. A. Arshed, "Impact of Artificial Intelligence in COVID-19 Pandemic: A Comprehensive Review," *2022 2nd Int. Conf. Distrib. Comput. High Perform. Comput. DCHPC 2022*, pp. 66–73, 2022, doi: 10.1109/DCHPC55044.2022.9732091.

[29] M. A. Arshed, W. Qureshi, M. Rumaan, M. T. Ubaid, A. Qudoos, and M. U. G. Khan, "Comparison of Machine Learning Classifiers for Breast Cancer Diagnosis," *4th Int. Conf. Innov. Comput. ICIC 2021*, 2021, doi: 10.1109/ICIC53490.2021.9692926.

[30] M. A. Arshed and F. Riaz, "Machine Learning for High Risk Cardiovascular Patient Identification 1," *J. Distrib. Comput. Syst.*, vol. 4, no. 2, pp. 34–39, 2021.

[31] M. Hussain, A. Shahzad, F. Liaquat, M. A. Arshed, S. Mansoor, and Z. Akram, "Performance Analysis of Machine Learning Algorithms for Early Prognosis of Cardiac Vascular Disease," *Tech. J.*, vol. 28, no. 02, pp. 31–41, Jun. 2023, Accessed: Dec. 26, 2023. [Online]. Available: https://tj.uettaxila.edu.pk/index.php/technical-journal/article/view/1778

[32] S. K. Shaukat and V. J. Ribeiro, "RansomWall: A layered defense system against cryptographic ransomware attacks using machine learning," *2018 10th Int. Conf. Commun. Syst. Networks, COMSNETS 2018*, vol. 2018-January, pp. 356–363, Mar. 2018, doi: 10.1109/COMSNETS.2018.8328219.

[33] and T. L. A. Tseng, Y. Chen, Y. Kao, "Deep learning for ransomware detection".

[34] S. Maniath, A. Ashok, P. Poornachandran, V. G. Sujadevi, A. U. P. Sankar, and S. Jan, "Deep learning LSTM based ransomware detection," *2017 Recent Dev. Control. Autom. Power Eng. RDCAPE 2017*, pp. 442–446, May 2018, doi: 10.1109/RDCAPE.2017.8358312.

[35] H. Daku, P. Zavarsky, and Y. Malik, "Behavioral-Based Classification and Identification of Ransomware Variants Using Machine Learning," *Proc. - 17th IEEE Int. Conf. Trust. Secur. Priv. Comput. Commun. 12th IEEE Int. Conf. Big Data Sci. Eng. Trust. 2018*, pp. 1560–1564, Sep. 2018, doi: 10.1109/TRUSTCOM/BIGDATASE.2018.00224.

[36] Y. Takeuchi, K. Sakai, and S. Fukumoto, "Detecting ransomware using support vector machines," *ACM Int. Conf. Proceeding Ser.*, Aug. 2018, doi: 10.1145/3229710.3229726.

[37] "Machine Learning-Based Detection of Ransomware Using SDN | Proceedings of the 2018 ACM International Workshop on Security in Software Defined Networks & Network Function Virtualization." Accessed: Feb. 07, 2024. [Online]. Available: https://dl.acm.org/doi/10.1145/3180465.3180467

[38] O. M. K. Alhawi, J. Baldwin, and A. Dehghantanha, "Leveraging Machine Learning Techniques for Windows Ransomware Network Traffic Detection," vol. 70, 2018, doi: 10.1007/978-3-319-73951-9_5.

[39] S. Poudyal, K. P. Subedi, and D. Dasgupta, "A Framework for Analyzing Ransomware using Machine Learning," *Proc. 2018 IEEE Symp. Ser. Comput. Intell. SSCI 2018*, pp. 1692–1699, Jul. 2018, doi: 10.1109/SSCI.2018.8628743.

[40] K. Lee, S. Y. Lee, and K. Yim, "Machine Learning Based File Entropy Analysis for

Ransomware Detection in Backup Systems," *IEEE Access*, vol. 7, pp. 110205–110215, 2019, doi: 10.1109/ACCESS.2019.2931136.

[41] F. Khan, C. Ncube, L. K. Ramasamy, S. Kadry, and Y. Nam, "A Digital DNA Sequencing Engine for Ransomware Detection Using Machine Learning," *IEEE Access*, vol. 8, pp. 119710–119719, 2020, doi: 10.1109/ACCESS.2020.3003785.

[42] L. Chen, C.-Y. Yang, A. Paul, and R. Sahita, "Towards resilient machine learning for ransomware detection," Dec. 2018, Accessed: Feb. 07, 2024. [Online]. Available: https://arxiv.org/abs/1812.09400v2

[43] S. Il Bae, G. Bin Lee, and E. G. Im, "Ransomware detection using machine learning algorithms," *Concurr. Comput. Pract. Exp.*, vol. 32, no. 18, p. e5422, Sep. 2020, doi: 10.1002/CPE.5422.

[44] J. Hwang, J. Kim, S. Lee, and K. Kim, "Two-Stage Ransomware Detection Using Dynamic Analysis and Machine Learning Techniques," *Wirel. Pers. Commun.*, vol. 112, no. 4, pp. 2597–2609, Jun. 2020, doi: 10.1007/S11277-020-07166-9/METRICS.

[45] H. Zuhair, A. Selamat, and O. Krejcar, "A Multi-Tier Streaming Analytics Model of 0-Day Ransomware Detection Using Machine Learning," *Appl. Sci. 2020, Vol. 10, Page 3210*, vol. 10, no. 9, p. 3210, May 2020, doi: 10.3390/APP10093210.