IJIRSET
International Journal of Innovative Research in
SCIENCE | ENGINEERING | TECHNOLOGY

e-ISSN: 2319-8753 | p-ISSN: 2347-6710

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

Volume 13, Issue 4, April 2024

ISSN INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 8.423

9940 572 462 | 6381 907 438 | ijirset@gmail.com | www.ijirset.com

International Journal of Innovative Research in Science, Engineering and Technology (IJIRSET)

| e-ISSN: 2319-8753, p-ISSN: 2347-6710| www.ijirset.com | Impact Factor: 8.423| A Monthly Peer Reviewed & Referred Journal |

|| Volume 13, Issue 4, April 2024 ||

| DOI:10.15680/IJIRSET.2024.1304072 |

A Review Paper on Ransomware Detection and Classification using Machine Learning

Deepraj Borse[1], Sanjana Ahire[2], Rohit Chaudhari[3], Rutuja Sonawane[4], Sunil Kale[5]

Department of Computer Engineering, Sandip Institute of Technology and Research Centre, Nashik, Maharashtra, SPPU, India[1,2,3,4]

Assistant Professor, Sandip Institute of Technology and Research Centre, Nashik, Maharashtra, SPPU, India[5]

ABSTRACT: Ransomware attacks represent a significant and growing threat in today's digital world, causing substantial financial losses and disrupting essential services for individuals and organizations globally. Traditional methods of detection, reliant on signatures, are proving increasingly inadequate against the ever-changing tactics of cybercriminals. In response, this research proposes an innovative approach using machine learning techniques for the detection and classification of ransomware. By harnessing machine learning, this study offers a proactive defence against ransomware attacks. Through thorough analysis and experimentation, it contributes to ongoing cybersecurity efforts by providing a robust and adaptive solution. This method enables the timely identification and categorization of ransomware variants, strengthening the resilience of systems and networks against malicious intrusions. This research not only enhances cybersecurity defences but also safeguards sensitive data, preserves financial resources, and protects critical infrastructure from the devastating effects of ransomware attacks in the dynamic digital landscape. By embracing machine learning technology, organizations can bolster their security measures and mitigate the risks posed by ransomware threats, promoting a safer and more secure digital environment for all stakeholders.

KEYWORDS: Ransomware, Machine Learning, Cybersecurity, Threat Detection, Classification, Adaptive Defense, Cyber Threads, Digital Security, Data Protection

I. INTRODUCTION

In the contemporary digital landscape, ransomware attacks have emerged as a pervasive menace, posing substantial risks to data integrity and security. Existing detection methods are increasingly inadequate against evolving ransomware strains, underscoring the urgency for advanced machine learning-driven solutions. This research endeavours to address this challenge by developing a sophisticated model for ransomware detection and classification. By harnessing the power of machine learning, cybersecurity practitioners can better mitigate risks and safeguard critical digital assets and infrastructure.

II. PROBLEM DEFINATION

Addressing the rapid evolution of ransomware variants has resulted in a notable deficiency in cybersecurity defence. Consequently, the primary challenge revolves around crafting a resilient, real-time ransomware detection and classification system adept at precisely identifying novel and polymorphic strains. This endeavour aims to mitigate the considerable financial, operational, and reputational hazards entailed by ransomware attacks.

III. LITERATURE SURVEY

Malicious attacks, including ransomware, pose severe security threats across various industries. Conventional anti-ransomware systems struggle against sophisticated attacks, necessitating innovative solutions. Researchers have proposed a feature selection-based framework employing machine learning algorithms, including Decision Trees, Random Forests, Naive Bayes, Logistic Regression, and Neural Networks, to classify ransomware security levels. (1). Ransomware employs encryption to render data inaccessible, causing substantial harm across governments, corporations, and private users. In response to the proliferation of these threats, researchers have proposed diverse detection and classification schemes, predominantly utilizing advanced machine learning techniques (2). Ransomware attacks pose severe threats to data security, causing privacy breaches, financial losses, and reputational damage (3).

| e-ISSN: 2319-8753, p-ISSN: 2347-6710| www.ijirset.com | Impact Factor: 8.423| A Monthly Peer Reviewed & Referred Journal |

|| Volume 13, Issue 4, April 2024 ||

| DOI:10.15680/IJIRSET.2024.1304072 |

Ransomware, an enduring threat, prompts an ongoing battle between evolving techniques and detection methods. Existing detection systems, while widely used, often struggle with the reactive nature of ransomware, necessitating continuous evolution (4). Ransomware, a form of malware, encrypts user data, blocking access until a ransom is paid. The evolving behaviours of ransomware make traditional detection and classification methods less effective(5). As web applications proliferate, the security landscape becomes increasingly precarious. Traditional intrusion detection systems, focusing on individual requests, struggle to address evolving cyber threats and are limited to known vulnerabilities (6). The global impact of sophisticated targeted attacks emphasizes the challenges in detection. Leveraging Microsoft's Sysmon tool, researchers propose a real-time method to detect malicious tools by analysing DLL information. The approach focuses on creating "common DLL lists" for identifying malicious processes universally, implementing a practical detection system using Elastic Stack as a Security Information and Event Management (SIEM). Evaluation with four US-CERT-introduced tools demonstrates the method's effectiveness, successfully detecting China Chopper, Mimikatz, and PowerShell Empire with minimal false positives. The common DLL lists prove valuable for real-time detection using Elastic Stack (7). As the dominant mobile operating system, Android is widely utilized for everyday activities, making it a prime target for hackers seeking to compromise personal information. To counteract this threat, a malware detection technique named MaplDroid is introduced. MaplDroid statically analyses application files by extracting features from the manifest file, employing a Naive Bayes-based supervised learning model to classify applications as benign or malicious. The proposed technique demonstrated high efficacy with a Recall score of 99.12 (8). Mobile Ad Hoc Networks (MANETs) are infrastructure-less networks crucial for communication during network failures. This manuscript introduces a Time Interval Based Blockchain Model (TIBBM) for security in MANETs, identifying malicious nodes. TIBBM establishes a Blockchain information structure, enabling the identification of malicious nodes at specified intervals. Compared to traditional models, TIBBM demonstrates superior performance in detecting malicious nodes during data transmission in MANETs.(9)

IV. FEATURES

[1]User Class and Characteristic: Our system caters to individuals with varying levels of cybersecurity expertise, accommodating those with extensive knowledge, some with moderate familiarity, and others with limited understanding of the subject matter.[2]Assumptions and Dependencies: The system's prerequisites entail the installation of Python and Spyder on the user's computer, along with logging into the system. Additionally, it is assumed that the implementation of transfer learning models can enhance the system's accuracy.[3]Functional Requirements: Admin: Admin module will be on window module. Admin will verify user information and allow or reject to user. Load the Data set. User: User registers into system with personal information. Automatically user verification request send to admin. After verification user can login into system. System: Utilizing the Support Vector Machine (SVM) algorithm, the system enhances ransomware detection and classification through machine learning techniques.[4]External Interface Requirements: User Interface: Application for Ransomware detection and classification using machine learning. [5] Hardware Requirements: To support the demanding tasks of machine learning, our system necessitates specific hardware configurations, including a minimum of 8GB RAM, a 40GB hard disk, an Intel i5 processor, and the utilization of the Spyder IDE. These hardware specifications are essential for handling large datasets and complex computational processes inherent in machine learning algorithms. [6] Software Requirements: Critical software prerequisites for our system include the Windows 10 operating system, Spyder IDE, and the Python programming language. Spyder IDE enhances coding efficiency, while Python's extensive libraries are instrumental in executing various machine learning tasks effectively. [7] Non-Functional Requirements: In ensuring the optimal performance of our system, it must exhibit rapid data encryption and efficient provisioning of virtual environments. Furthermore, the system's safety is assured through a modular design, facilitating easy error detection and updates, while robust encryption and access controls maintain the security of sensitive user data.

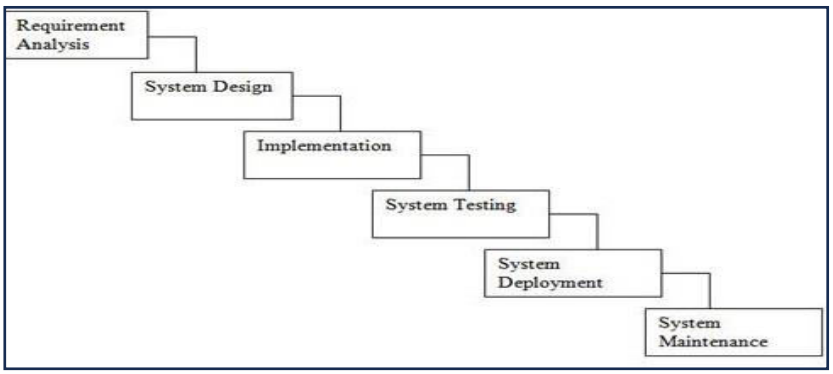
V. ANALYSIS MODEL

The software development cycle, employing the waterfall model, involves phases like requirements analysis, design, implementation, testing, deployment, and maintenance. It begins with documenting business requirements and use cases, followed by defining data models and conducting data analysis. Development using suitable algorithms occurs in the implementation phase, leading to testing for accuracy validation. Upon validation, deployment and maintenance are crucial for addressing discrepancies and adapting to changing business needs.

| e-ISSN: 2319-8753, p-ISSN: 2347-6710| www.ijirset.com | Impact Factor: 8.423| A Monthly Peer Reviewed & Referred Journal |

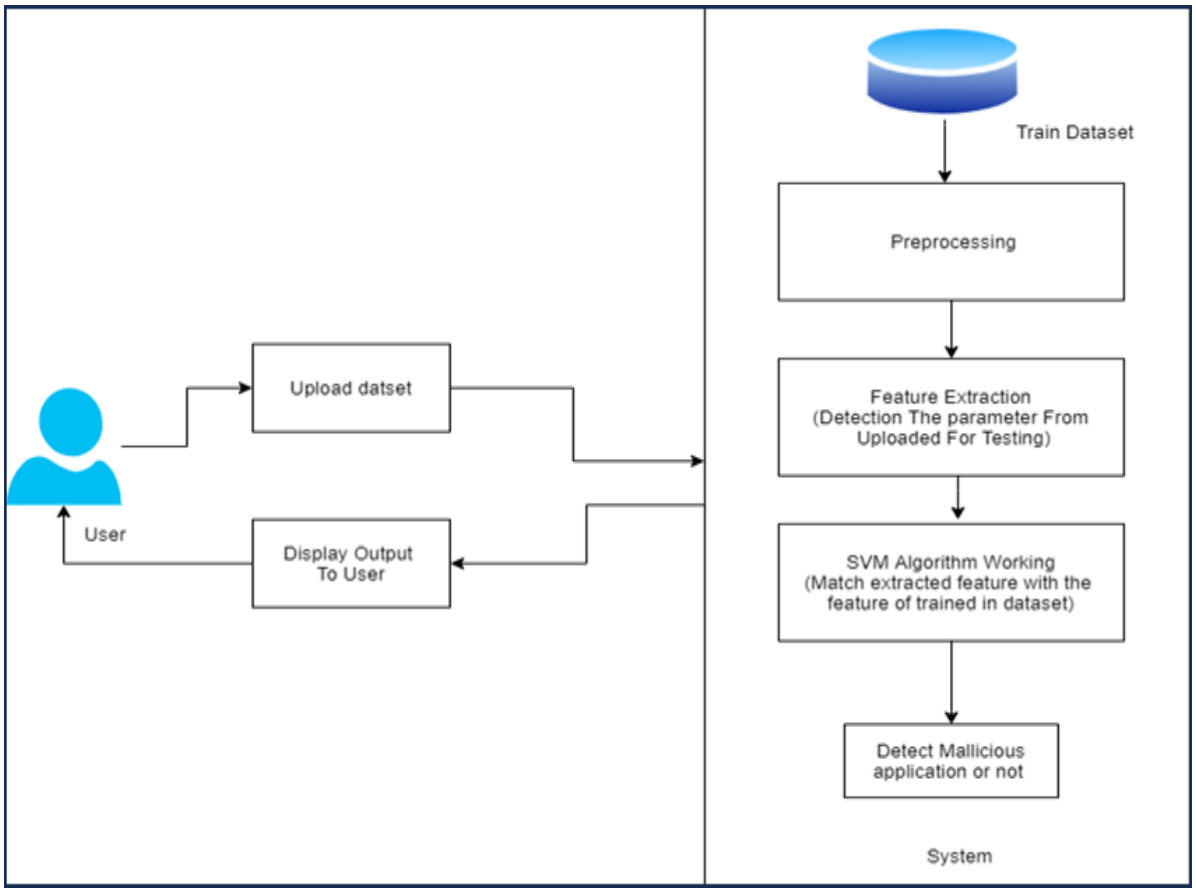
|| Volume 13, Issue 4, April 2024 ||

| DOI:10.15680/IJIRSET.2024.1304072 |



VI. SYSTEM ANALYSIS

The system initiates with input text, comprising a dataset containing both ransomware and benign software samples. Pre-processing, involving data cleaning, enhances dataset quality by removing missing or irrelevant data. Feature extraction isolates key characteristics relevant to ransomware identification. The Support Vector achine (SVM) algorithm is pivotal in classification, trained on 80% of the dataset and tested on the remaining 20% for generalization. The system determines whether the text contains ransomware characteristics or is benign. In case of malicious detection, an alert message is promptly sent to the user, enabling proactive threat mitigation.



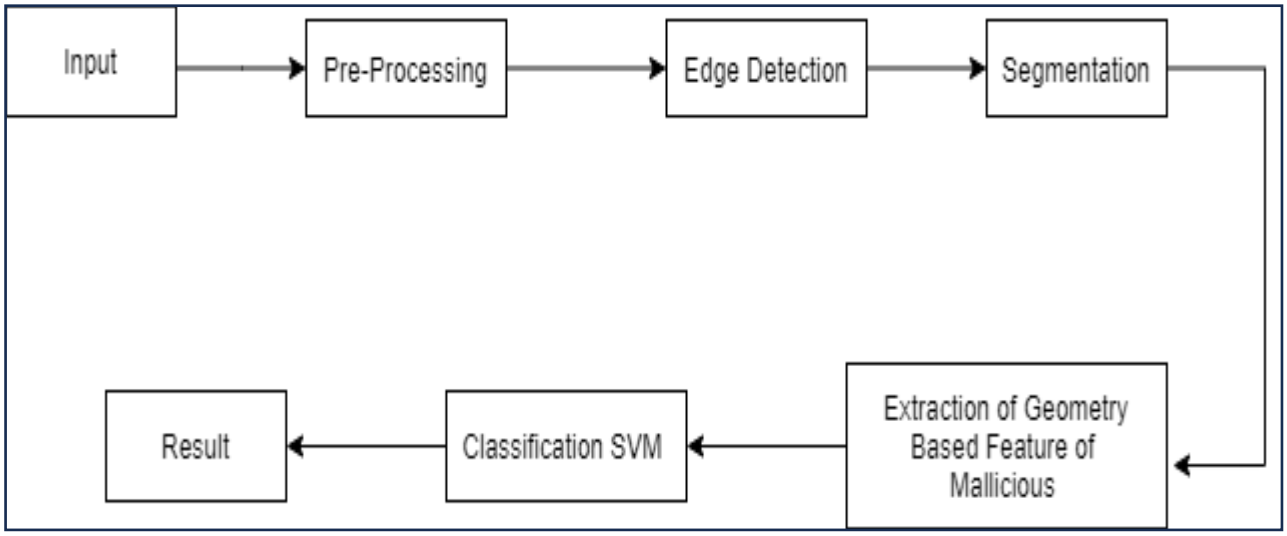
| e-ISSN: 2319-8753, p-ISSN: 2347-6710| www.ijirset.com | Impact Factor: 8.423| A Monthly Peer Reviewed & Referred Journal |

|| Volume 13, Issue 4, April 2024 ||

| DOI:10.15680/IJIRSET.2024.1304072 |

VII. DATA FLOW DIAGRAM

In the Data Flow Diagram (DFD), we illustrate the flow of data within our system. DFD0 represents the foundational DFD, where rectangles depict input and output, while circles represent our system. DFD1 elaborates on the actual input and output of our system, where the input comprises text or images, and the output indicates detected rumours. Similarly, DFD2 delineates the operations of both users and administrators within the system.



VIII. SOFTWARE INFORMATION

PYTHON: Python, the superhero of programming languages, emerged in the late 1980s as a remarkable upgrade to ABC. Since its debut in 1991, Python has soared in popularity, thanks in part to its unique readability feature known as significant whitespace. Over the years, Python has undergone significant improvements, notably with the releases of Python 2.0 in 2000 and Python 3.0 in 2008. With powerful additions like list comprehensions and enhanced garbage collection, Python 3 officially replaced Python 2 upon its retirement in 2020. Python's versatility shines through its support for structured, object-oriented, and functional programming paradigms, making it suitable for projects of any scale. Moreover, its rich ecosystem of built-in tools and modules simplifies development tasks, akin to having a comprehensive toolkit readily available. Beyond its technical prowess, Python owes much of its success to its vibrant developer community, which remains active and supportive, bolstered by the dedicated efforts of the Python Software Foundation. Despite Guido van Rossum's step back from leadership in 2018, Python continues to thrive and evolve, fueled by the passion and collaboration of its community members.

ANACONDA: Anaconda, founded by Peter Wang and Travis Oliphant in 2012, is a vital distribution of Python and R for scientific computing. Offering seamless package management and deployment, it supports data science, machine learning, and large-scale data processing. With an intuitive interface, Anaconda provides over 250 pre-installed packages and access to 7,500 open-source ones, catering to diverse needs. Its conda package management system simplifies handling package versions and dependencies, ensuring compatibility. Anaconda Navigator offers a graphical interface for easy package management. Unlike pip, conda prevents conflicts by thoroughly analyzing the environment. Users can effortlessly create and share custom packages through integration with Anaconda Cloud and PyPI. Anaconda's user-centric design and robust capabilities make it indispensable for developers and researchers, fostering innovation in scientific computing.

IX. ALGORITHM DETAILS

SVM: Support Vector Machine (SVM) is a supervised learning algorithm used for classification and regression tasks. It's particularly effective in cases where the data is not linearly separable. SVM works by finding the optimal hyperplane that best separates classes in a high-dimensional space.

SVM Algorithm for Ransomware Detection and Classification: In our endeavour to develop a robust Ransomware Detection and Classification system using machine learning techniques, Support Vector Machine (SVM) emerges as a pivotal algorithm. SVM, a supervised learning method, plays a crucial role in discerning between ransomware and

| e-ISSN: 2319-8753, p-ISSN: 2347-6710| www.ijirset.com | Impact Factor: 8.423| A Monthly Peer Reviewed & Referred Journal |

|| Volume 13, Issue 4, April 2024 ||

| DOI:10.15680/IJIRSET.2024.1304072 |

benign software by delineating optimal decision boundaries in complex feature spaces. Within the context of our project, SVM undergoes adaptation and customization to effectively address the unique challenges posed by ransomware detection.

X. ALGORITHMIC ADAPTATIONS

[1]Data Preprocessing: To ensure SVM's optimal performance, preprocessing steps involve normalization or standardization of features within ransomware datasets, alongside meticulous handling of categorical variables and missing values to maintain data integrity. [2]Kernel Function Selection: Tailoring kernel function choices based on the intricacies of ransomware data becomes paramount. Selections include linear kernels for linearly separable features, polynomial kernels for capturing non-linear decision boundaries, and Gaussian (RBF) kernels adept at handling complex, non-linear feature distributions.[3]Optimization Objective: SVM resolves a constrained optimization problem aimed at maximizing the margin between ransomware and benign software samples while minimizing misclassifications, thereby enhancing detection accuracy. [4]Decision Boundary Establishment: Upon identifying the optimal hyperplane, SVM delineates the decision boundary, facilitating the classification of new samples. Points situated on opposite sides of the hyperplane belong to distinct classes, enabling effective ransomware classification. [5]Handling Non-separable Data: In scenarios where ransomware data appears non-separable, SVM employs kernel tricks to transform it into higher-dimensional spaces, rendering it linearly separable if necessary. [6]Regularization Parameter Adjustment: Incorporating a regularization parameter (C) enables balancing between maximizing margin and minimizing misclassification errors, with fine-tuning to mitigate overfitting or underfitting tendencies specific to ransomware detection tasks.[7]Kernel Parameter Tuning: Kernel parameters undergo meticulous tuning to optimize SVM performance, utilizing cross-validation methodologies to identify the most suitable parameters for accurate ransomware classification. [8]Model Training, Prediction, Evaluation, and Fine-tuning: During the training phase, SVM learns discriminative patterns from ransomware and benign software datasets, adapting its hyperplane to delineate optimal decision boundaries.[9]Integration and Deployment: Integration of the trained SVM model into our Ransomware Detection and Classification system enables real-time detection and classification of ransomware threats, forming an integral component of our comprehensive approach to cybersecurity.

XI. SOFTWARE TESTING

In the domain of software development, testing plays a pivotal role, its execution intricately linked to the chosen methodology. While testing may occur at different stages, a significant portion typically follows requirement delineation and code finalization. Agile methodologies, such as test-driven development, emphasize early developer involvement in testing, whereas traditional models often prioritize testing after requirement specification and coding. Consequently, the testing approach varies depending on the development model adopted, with contemporary methodologies advocating for early and iterative testing processes.

TYPE OF TESTING USED:

[1]Unit Testing: Unit testing, vital in software development, occurs after unit completion but before integration. It verifies internal logic and output accuracy with crafted test cases, scrutinizing decision branches and code flow. [2] Integration Testing: Integration testing is pivotal for assessing the smooth integration and functionality of software components within the system. This phase emphasizes event-driven scenarios and core outcomes to ensure cohesive operation of individual components. By validating the interaction of previously tested components, integration testing effectively identifies any discrepancies resulting from their combination.[3]System Test: In our application evaluation, we employ a systematic testing approach, covering unit, integration, validation, GUI, low-level, and high-level test cases, as well as major scenarios. Starting with GUI and integration testing, we proceed to validate data transmission across various inputs and outputs, ensuring accurate and reliable results. [4]White-box Testing: This method is instrumental in identifying potential implementation flaws within individual components and algorithms.[5]Black-box Testing:. This method ensures that the software aligns with specified requirements and user expectations, regardless of its internal workings.

| e-ISSN: 2319-8753, p-ISSN: 2347-6710| www.ijirset.com | Impact Factor: 8.423| A Monthly Peer Reviewed & Referred Journal |

|| Volume 13, Issue 4, April 2024 ||

| DOI:10.15680/IJIRSET.2024.1304072 |

XII. TEST CASES AND TEST RESULTS

Login Test Case:

Test Case ID	Test Case	Test Case I/P	Actual Result	Expected Result	Test case criteria(P/F)
001	Enter The Wrong username or password click on submit button	Username or password	Error comes	Error Should come	P
002	Enter the correct username and password click on submit button	Username and password	Accept	Accept	P

Registration Test Case:

Test Case ID	Test Case	Test Case I/P	Actual Result	Expected Result	Test case criteria(P/F)
001	Enter the number in username, middle name, last name field	Number	Error Comes	Error Should Comes	P
001	Enter the character in username, middle name, last name field	Character	Accept	Accept	p
002	Enter the invalid email id format in email id field	Kkgmail.com	Error comes	Error Should Comes	P
002	Enter the valid email id format in email id field	kk@gmail.com	Accept	Accept	P
003	Enter the invalid digit no in phone no field	99999	Error comes	Error Should Comes	P
003	Enter the 10 digit no in phone no field	9999999999	Accept	Accept	P

| e-ISSN: 2319-8753, p-ISSN: 2347-6710| www.ijirset.com | Impact Factor: 8.423| A Monthly Peer Reviewed & Referred Journal |

|| Volume 13, Issue 4, April 2024 ||

| DOI:10.15680/IJIRSET.2024.1304072 |

GUI Testing:

Test Case ID	Test Case	Test Case I/P	Actual Result	Expected Result	Test case criteria(P/F)
001	Store Xml File	Xml file	Xml file store	Error Should come	P
002	Parse the xml file for conversion	parsing	File get parse	Accept	P
003	Attribute identification	Check individual Attribute	Identify Attributes	Accepted	P
004	Weight Analysis	Check Weight	Analyze Weight of individual Attribute	Accepted	P
005	Tree formation	Form them-Tree	Formation	Accepted	P
006	Cluster Evaluation	Check Evaluation	Should check Cluster	Accepted	P
007	Algorithm Performance	Check Evaluation	Should work Algorithm Properly	Accepted	P
008	Query Formation	Check Query Correction	Should check Query	Accepted	P

XIII. RESULTS

SCREENSHOTS:

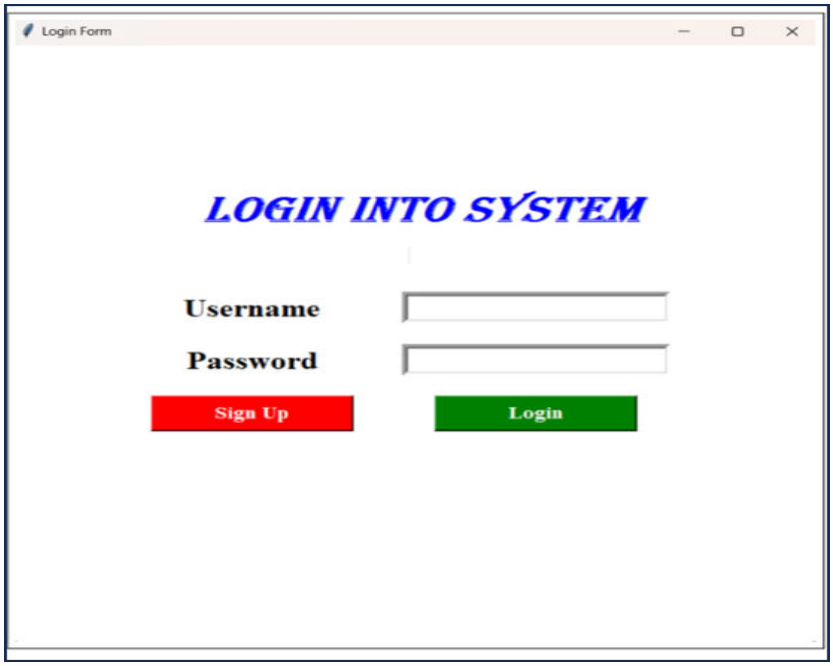
GUI Main:

| e-ISSN: 2319-8753, p-ISSN: 2347-6710| www.ijirset.com | Impact Factor: 8.423| A Monthly Peer Reviewed & Referred Journal |

|| Volume 13, Issue 4, April 2024 ||

| DOI:10.15680/IJIRSET.2024.1304072 |

Login Page:



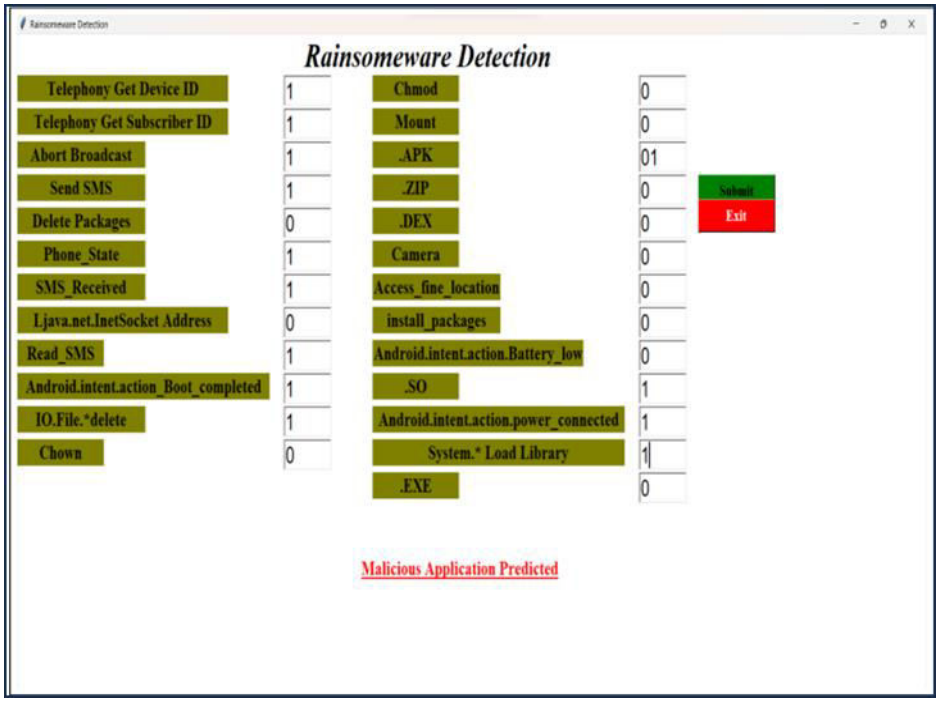
Registration Page:

| e-ISSN: 2319-8753, p-ISSN: 2347-6710| www.ijirset.com | Impact Factor: 8.423| A Monthly Peer Reviewed & Referred Journal |

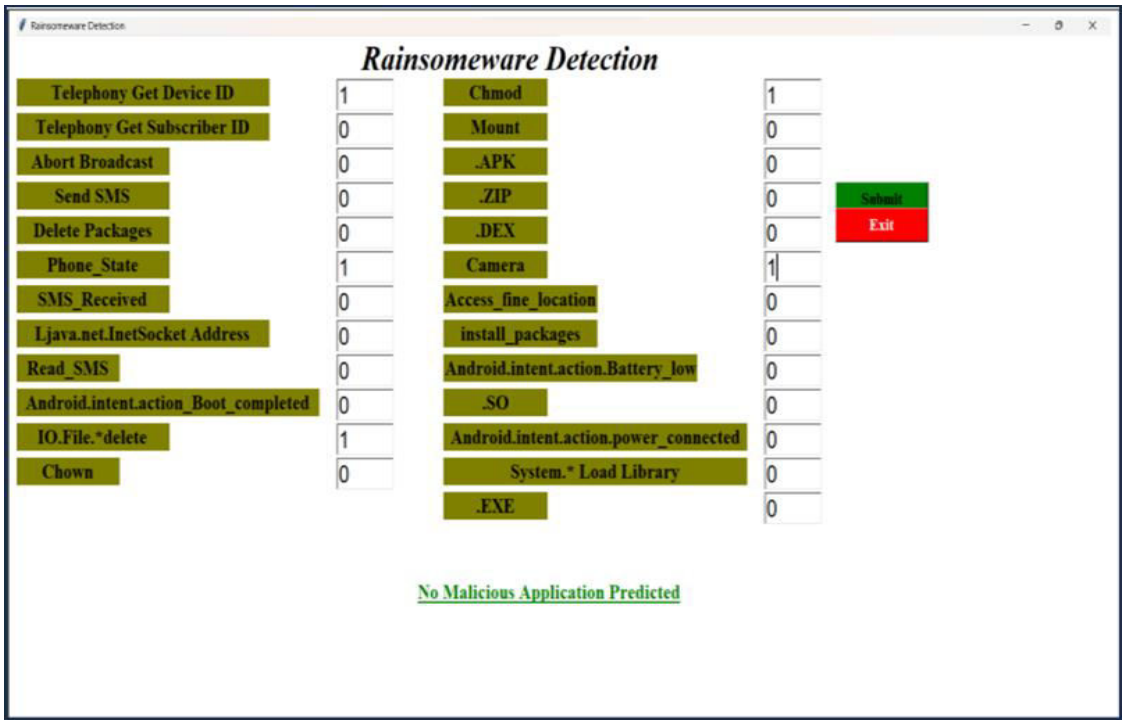
|| Volume 13, Issue 4, April 2024 ||

| DOI:10.15680/IJIRSET.2024.1304072 |

Output 1:



Output 2:



| e-ISSN: 2319-8753, p-ISSN: 2347-6710| www.ijirset.com | Impact Factor: 8.423| A Monthly Peer Reviewed & Referred Journal |

|| Volume 13, Issue 4, April 2024 ||

| DOI:10.15680/IJIRSET.2024.1304072 |

XIV. ADVANTAGES

- Real-time detection: Machine learning empowers swift response and mitigation measures against ransomware attacks before significant damage occurs.
- Automated analysis: ML algorithms automatically analyse large datasets, saving time and reducing the need for manual analysis.
- Adaptability to new threats: ML models adeptly identify patterns indicative of ransomware, even in previously unknown variants, enhancing threat detection capabilities.

XV. LIMITATIONS

- Machine learning models rely on high-quality, diverse datasets to achieve effective training.
- The risk of overfitting during training can compromise the model's ability to generalize to unseen data.
- While deep learning models boast high accuracy, their lack of interpretability presents challenges in comprehending decision-making processes, especially for non-experts.

XVI. APPLICATION

- Enterprise Security
- Critical Infrastructure
- Protection Healthcare Sector
- Financial Institutions

XVII. CONCLUSIONS

In summary, leveraging Support Vector Machines (SVM) for ransomware detection and classification presents a substantial advancement in enhancing cybersecurity measures. This study's outcomes not only enrich the existing knowledge base but also provide actionable guidance for deploying resilient ransomware detection systems in practical settings. With the dynamic nature of the cybersecurity landscape, ongoing research and innovation in SVM-based methodologies will remain integral in safeguarding digital environments against ransomware threats.

XVIII. FUTURE SCOPE

To enhance the transparency and robustness of machine learning models, integrating explainable AI techniques is paramount. These methods enable us to comprehend how the model makes decisions, which is crucial in sensitive domains like finance, healthcare, and criminal justice. Additionally, safeguarding against adversarial attacks through techniques like adversarial training strengthens the model's resilience[1]. Collaborative approaches to sharing threat intelligence among organizations are vital in cybersecurity. By exchanging information about potential threats and attack patterns, institutions can collectively bolster their defences and stay ahead of emerging risks, particularly in industries like finance where threats to one institution may signal danger for others[2]. Remaining abreast of advancements in deep learning architectures is essential. Continuous monitoring of the latest research ensures that machine learning models benefit from cutting-edge methodologies across domains such as image recognition, speech processing, and natural language understanding[3]. Human-in-the-loop systems, where machine learning models collaborate with human analysts, offer significant potential. By leveraging the computational power of machines and the nuanced insights of human experts, these systems can improve decision-making accuracy, particularly in fields like cybersecurity where human validation enhances the reliability of threat detection[4]. Practically, these strategies find application across diverse sectors. In healthcare, explainable AI fosters trust by making medical diagnosis models understandable to professionals. Collaborative threat intelligence sharing in finance aids in collective defence against cyber threats. Human-in-the-loop systems enhance threat detection accuracy in defence and security[5].These strategies are crucial in safeguarding critical data, preventing ransomware attacks, protecting sensitive information in healthcare and government institutions, securing intellectual property in the corporate sector, and ensuring the security of interconnected devices in smart cities and IoT ecosystems[6].The adaptability of SVM-based ransomware detection positions it as a valuable tool in diverse sectors, contributing to a more secure digital future[7].

| e-ISSN: 2319-8753, p-ISSN: 2347-6710| www.ijirset.com | Impact Factor: 8.423| A Monthly Peer Reviewed & Referred Journal |

|| Volume 13, Issue 4, April 2024 ||

| DOI:10.15680/IJIRSET.2024.1304072 |

REFRENCES

[1] M. J. H. F. H. S. K. Mohammad Masum, "Ransomware Classification and Detection With Machine Learning," in IEEE 12th Annual Computing and Communication Workshop and Conference (CCWC), mm, 2022.

[2] N. G. ,. E. B.-H. ,. J. C. ,. Aldin Vehabovic1, "Ransomware Detection and Classification Strategies," in 2022 IEEE International Black Sea Conference on Communications and Networking (BlackSeaCom). IEEE, 2022., 2022.

[3] *. a. A. A. 2. Amjad Alraizza 1, "Ransomware Detection Using Machine Learning: A Survey," in Big Data Cogn. Comput, 2023.

[4] S. P. a. A. C. Samuel Egunjobi1, ":Classifying Ransomware Using Machine Learning Algorithms," 2019. *

[5] R. B. A. Dr. Nirmala Hiremani1, 2020. * D. Narayana1, "A Time Interval based Blockchain Model for Detection of Malicious Nodes in MANGET Using Network Block Monitoring Node," in Proceedings of the Second International Conference on Inventive Research in Computing Applications (ICIRCA-2020), 2019.

[6] Gao, Yang & Ma, Yan & Li, Dandan. (2017). Anomaly detection of malicious users' behaviors for web applications based on web logs. 1352-1355. 10.1109/ICCT.2017.8359854..

[7] Matsuda, Wataru & Fujimoto, Mariko & Mitsunaga, Takuho. (2019). Real-Time Detection System Against Malicious Tools by Monitoring DLL on Client Computers. 36-41. 10.1109/AINS47559.2019.8968697.

[8] Bhat, Parnika & Dutta, Kamlesh & Singh, Sukhbir. (2019). MaplDroid: Malicious Android Application Detection based on Naive Bayes using Multiple. 49-54. 10.1109/ICCT46177.2019.8969041.

[9] D. Narayana1, "A Time Interval based Blockchain Model for," in Proceedings of the Second International Conference on Inventive Research in Computing Applications (ICIRCA-2020), Guntur. Andrapradesh, India, 2020.

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INNO SPACE SJIF Scientific Journal Impact Factor

doi cross ref

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

9940 572 462 6381 907 438 ijirset@gmail.com

www.ijirset.com

Scan to save the contact details