International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 05 | May - 2023 SJIF 2023: 8.176 ISSN: 2582-3930

ANALYSIS OF MALWARE DETECTION TECHNIQUES USING MACHINE LEARNING

Shivakumar Nethani[1], Lade Gunakar Rao[2], Kusumba Jyoshna Devi[3]

[1]*Department of Computer Science and Engineering & Hyderabad Institute of Technology and Management, Hyderabad, Telangana, India*
[2]*Department of Computer Science and Engineering & SR University, Warangal,* Telangana, India
[3]*Department of Computer Science and Applications & Govt Degree College, Warangal,* Telangana, India

---

Abstract - ***Malicious software that has the potential to infect a single computer or an entire company's network is referred to as malware. One of the most serious threats to online safety at the moment is malware and viruses. Given how quickly the amount of malware is growing, there is a severe threat to global security. All modern malware applications have a tendency to include multiple polymorphic layers or side mechanisms that automatically update themselves at regular intervals in order to avoid detection by any antivirus software and avoid detection for longer periods of time. We provide a modular framework that enables the use of several machine learning techniques, such as decision-tree [1], It has become the hardest job for security vendors to disguise a program as malware. Android malware has advanced to the point where it is increasingly resistant to common detection methods in terms of intelligence and cognition. Machine learning-based strategies have become a significantly more effective technique to deal with the complexity and uniqueness of emerging Android threats. They work by first identifying current malware activity patterns, then using this data to differentiate between known risks and unknown threats [2].***

Key Words: *Malware, Malware Detection, Machine Learning, Malware Analysis, Android applications, feature extraction, malware detection, Android, co-existence, Android*.

1. INTRODUCTION

One of the most prevalent possessions among humans is In the current world, data and information are precious assets. Because they are so important to people, It is imperative that their security and protection be guaranteed at all costs [1]. Furthermore, Android smartphones currently account for an average of 80% of the global market share over the previous years, making them the dominant operating system for the vast majority of mobile devices. One of the most popular operating systems for smartphones nowadays is Android. This is the key cause of why hackers and attackers have started to favour Android as a target. Detecting and identifying malicious software in Android applications is becoming increasingly difficult due to their advanced embedding techniques [2]. In more than 190 countries, Android is the most popular operating system with over 2.5 billion active users. Smartphones' extensive feature sets and the growing amount of activities—including social networking, online banking, and gaming—that their users engage in has given rise to very severe issues about device security and personal privacy. Because Android is an open-source operating system, malware developers can easily launch their attacks and create Android malware apps that pose a serious threat. Clearly, Android malware is having a greater impact on contemporary society [3].Through the Internet, malware is disseminated globally and grows more prevalent every day, posing a severe threat. However, researchers are working to provide a number of new strategies for identifying and eliminating malware. To achieve efficacy and efficiency in detecting malware, one suggested method (solution) combines automatic dynamic behaviour malware analysis with data mining tasks, such as machine learning (classification) algorithms [4].However, researchers are working to provide a number of new strategies for identifying and eliminating malware. To achieve efficacy and efficiency in detecting malware, one suggested method (solution) combines automatic dynamic behaviour malware analysis with data mining tasks, such as machine learning (classification) algorithms. It is impractical for real-world deployment due to duplicate information brought on by a large number of features that result in low accuracy and high False Positive Rate (FPR) [5], Different statistical features are being used by dataset repositories like OpenML and Kaggle to describe each dataset they hold. Dataset profiling is a useful technique for systematically bringing these elements to the fore. By combining other datasets or samples from those datasets according to these features, it is possible to preserve a qualitative dataset if a dataset is profiled from distinct attributes. The attributes, however, are numerical and challenging to comprehend and contrast [6].

2. LITERATURE REVIEW

Machine learning approaches for malware detection have been proposed by many researchers. To categorize malware, machine learning techniques including association classifiers, support vector machines, decision trees, random forests, and Naive Bayes are used. We provide a few instances of these methods in this section [1].

According to a Sonic Wall analysis, the number of malware attacks in India increased by 31% in 2022, which should motivate businesses to step up their efforts to protect themselves online.

The classification of data via machine learning uses a variety of approaches. The strong learner SVM represents each data point as a point in n-dimensional space (where n represents the number of features you have), with the value of each feature being the vector value. In order to improve the recognition properties of any two parameters, it then performs classification by identifying the hyper plane that best separates the two groups. Contrarily, boosting or ensemble

techniques, such as Ada boost, give higher weights to variables that are misclassified in order to improve the behavior of the variables when used in conjunction with other machine learning algorithms[2]. In order to avoid being discovered by anti-malware scanners, mobile malware is regularly updated with new capabilities. To gain access to the user's devices, Android malware programs often use three different cutting-edge methods.

1) Repackaging: Repackaging is one of the popular ways to install malware applications. With this technique, developers install popular software, disassemble them, add malicious code, and then reassemble them before posting the finished product for users to download through a third party.

2) Update: Although the developer may still be employing repackaging, they establish an update component that even downloads harmful malware while the programme is running, rather than blocking the malicious code.

3) Downloading: By luring consumers to download engaging and practical apps, developers most frequently persuade users to download their software. These apps, however, are dangerous and could damage users' devices [3].

3. EXISTING SYSTEM

Antivirus software is frequently used to identify malware since it examines each program on the system for known malware. But it's well recognized in the security community that the current signature-based approach to virus identification is no longer sufficient. Antivirus programs that rely on traditional signature matching miss polymorphic executables and newly identified hazardous executables.

Traditional anti-virus depends on signatures, but polymorphic malware can evade these methods, making this paradigm less trustworthy.

4. PROPOSED SYSTEM

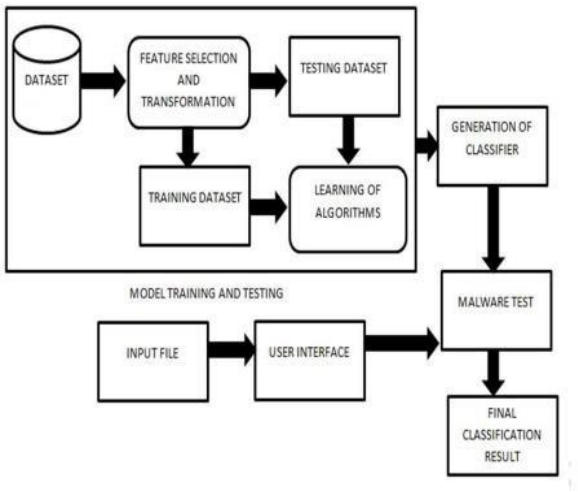
The user interface, the train module, and the malware test module are the three main parts that make up our product. The first element of our system is the user interface, and the second is the train module.

The system's front-end architecture is contained in the user interface module, which is also known as the front-end module. In essence, it provides the user with an interface for entering the file to be checked for potentially harmful content.

The following module is the train module. The selected models are trained and tested using this module. The model that will be used is selected depending on how accurate each candidate. The final categorization outcome is primarily determined by this module. The model's classifier is also generated in this module. It is primarily accountable for the file's data extraction, data determination, uploading, and partition into numerous sections or characteristics [1].

The architecture is primarily composed of three modules: 1. Feature Database. 2. Feature Selection and Transformation, and 3. Algorithm Learning

Fig -1: List of Modules.



The Kaggle Microsoft malware classification challenge dataset, which is a CSV file (comma separated file), was used for this project. Then, in the subsequent step, several feature selection techniques, including chi-square, information gain, fisher score, gain ratio, and symmetric uncertainty feature selection techniques, are used. The dataset will be split into two portions, Testing Dataset and Training Dataset, after feature selection and transformation.

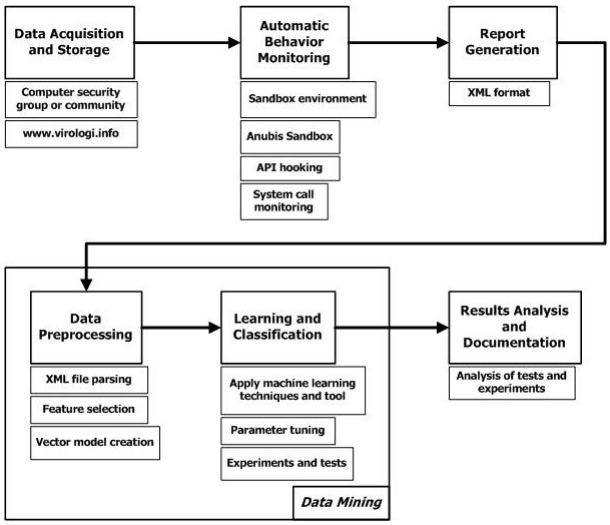
In this instance, we used a variety of techniques to find unidentified malware in the file. In the suggested technique, Random Forest, Decision Tree, Linear Regression, and Ada boost identify malware with high accuracy and improve efficiency. The architecture's final step is to classify the discoveries. The gathering of the data set is the initial phase. By browsing the web and using tools like Kaggle, this can be done. The features of the data site are chosen and altered once the data has been obtained. Following feature selection and transformation, an important process known as feature importance takes place.

Malware data set and benign instance data set are both included in the data set. Windows Portable Executable (PE) file binaries are the format used by both malicious and benign instance data sets. We acquired 220 distinct Malware (Indonesian malware) samples in total.

The system files from the "System32" directory of a freshly installed copy of Windows 7 Ultimate 64-bit with Service Pack 1 were used to create the benign instance data set samples. We obtained a total of 250 distinct benign software samples. The next step is to conduct learning and classification based on the ARFF files. Machine learning techniques were applied for the learning and classification of the ARFF file.

The tests and experiments were conducted using Weka 3.6.2 for Windows OS version. These tests and experiments were conducted based upon four data sets, Each data set was applied to 5 different classifier algorithms: J48 decision tree, Naive Bayes, Support Vector Machine (SVM), and k-Nearest Neighbour (k-NN) neural network [2].

Fig -2: General overview of the research methodology.



5. IMPLEMENTATION

The gathering of the data set is the initial phase. By browsing the web and using tools like Kaggle, this can be done. The features of the data site are chosen and altered once the data has been obtained.

Following feature selection and transformation, an important process known as feature importance takes place. The evaluation of features' importance is referred to as feature importance.

Identifying the most important characteristics is a technique known as feature importance. It speaks of the characteristics that affect the database or system the most.

The data set is then split into two sections:

80% of the dataset is used for training:

This dataset's component is mostly used for training. In essence, the model learns from this dataset.

20% of the dataset is used for testing:

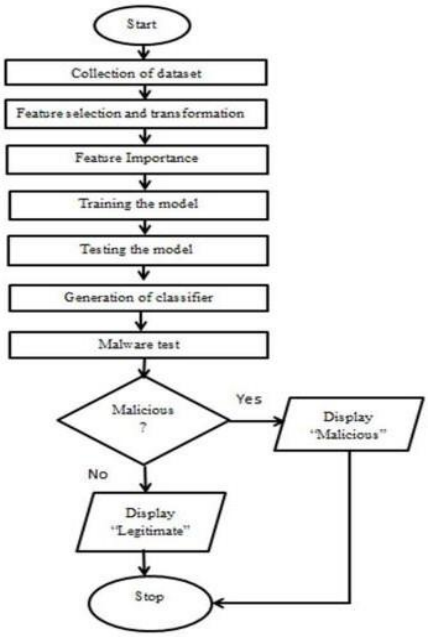
The majority of the dataset is utilized for testing. This dataset is used to test the model. The testing dataset is used to assess the model's correctness.

This section of the dataset is primarily used for testing. Themodel is tested using this dataset. The model's accuracy is thus determined using the testing dataset.

Our dataset indicates that Random Forest is the algorithm with the highest level of accuracy.

It was consequently selected for inclusion in the system. The model is then trained using the dataset after that.

Fig -3: Process of Detecting Malwares in Dataset.



6. CONCLUSIONS

In this research, we describe a brand-new malware analysis methodology that is capable of quickly identifying and categorizing malware. Our suggested method employs two distinct feature selection algorithms to extract the most pertinent features, which shortens training time and improves the precision of detection and classification.

In compared to other classifiers, experimental data demonstrate that the decision tree provides high accuracy for detection and classification. Additionally, we categorize malware according to their family and verify the accuracy of each malware family.

This shows that the accuracy of our technology, which is sufficient for identifying malware, is around 99 percent. As a result, it may be said that your system produces findings that are 99 percent accurate.

A classification approach that will accurately identify the type of malware that has attacked the file and can act as a foundation for various research to identify the most frequently attacking malwares can be added to the malware detection system that is currently being presented. As a result, this offers a feasible option for further project work.

ACKNOWLEDGEMENT

We suggest a flexible framework that enables the use of various machine learning techniques for successfully identifying malware files.

REFERENCES

1. Prati Jain, Ishita Rajvaidya, Keshav Kumar Sah, M.K Jayanthi Kannan, "Machine Learning Techniques for Malware Detection", 2022

2. Beenish Uroojmunam Ali Shah, Carsten Maple, Muhammad Kamran Abbasi, Sidra Riasat, "Malware Detection: A Framework for Reverse Engineered Android Applications Through Machine Learning Algorithms", 2022

3. Esraa Odat, Qussai M. Yaseen, "A Novel Machine Learning Approach for Android Malware Detection Based on the Co-Existence of Features", 2022

4. Ivan Firdausi, Charles Lim, Alva Erwin, Anto Satriyo Nugroho, "Analysis Of Machine Learning Techniques Used In Behavior-Based Malware Detection", 2010

5. Kamalakanta Sethi, Rahul Kumar, Lingaraj Sethi, Padmalochan Bera, and Prashanta Kumar Patra, "A Novel Machine Learning Based Malware Detection and Classification Framework", 2019

6. Gürol Canbek, Seref Sagiroglu, Seref Sagiroglu, "New Techniques in Profiling Big Datasets for Machine Learning with A Concise Review of Android Mobile Malware Datasets",2018

BIOGRAPHIES

NETHANI SHIVAKUMAR received the M.Tech in Computer Science and Engineering from the University of JNTU, Hyderabad, Telangana, India. He is currently working as an Assistant Professor with the Department of Computer Science and Engineering, Hyderabad Institute of Technology and Management. His research interests include Machine Learning and Data Mining.