Android malware detection using Genetic Algorithm based Optimized Feature Selection and Machine Learning

Mr Srinivas
Department of Information Technology
Aditya Engineering College
Surampalem, India

Sappa Narendra
Department of Information Technology
Aditya Engineering college
Surampalem, India

Vusirika Sushma
Department of Information Technology
Aditya Engineering college
Surampalem, India

Manne Thanmai
Department of Information Technology
Aditya Engineering College
Surampal, India

Abstract-Due to its popularity and support, the Android os does have the biggest market share worldwide OS has attracted the attention of cybercriminals who operate mostly through the widespread distribution of malicious software. In order to effectively detect Android malware, this study suggests a machine-learning-based method that makes use of an evolving evolutionary algorithms for such collection appropriate discriminatory features. Machine learning classifiers are taught employing specific features using genetic algorithms, and their performance in identifying malware front and back feature selection is compared. The findings of the experiment confirm that the genetic algorithm gives the much more efficient feature data. For the machine learning-based classifiers, classification accuracy of more than 94% is maintained after feature selection while working on a considerably smaller feature dimension, positively affecting the computational complexity of learning classifiers.

Keywords-Android malware detection, machine learning, genetic algorithms, classifiers.

I. INTRODUCTION

Users can download free Android apps from Playstore, which is official Android app store, and also from the other apps like Chrome etc. Malware writers are increasingly concentrating on creating dangerous programmes for the Android operating system popularity. Despite Google Playstore numerous attempts to stop that kind of harmful apps, users by misusing personal information from them, by third parties or else remotely controlling the phones. Thus, such malicious applications that constitute a major threat to Android platforms must be subjected to malware analysis or reverse engineering. Analysis like Static and dynamic are the two main types of Android malware analysis. While dynamic analysis examines behaviour of apps when they are running in a limited context, static analysis simply involves studying the code structure without actually running it. Given the growing number of Android malware types that pose zero-day threats, an effective method of detecting Android malware is necessary. Unlike signature-based approaches, which necessitate updating the signature database on a regular basis.

II. LITERATURE REVIEW

Several studies introduce Android Malware Detection Using algorithm like genetic algorithm and some classifiers from machine learning in these papers. These papers present numerous methodologies and techniques for detecting Android malware. We shall now view those papers. The document is the first thing we see and is based on the book Drebin: Efficient and Explainable Detection of Android Malware in Your Pocket by D. Arp, M. Spreitzenbarth, M. Hubner, H. Gascon, and K. Rieck.16Android platform security is at risk from malicious applications. Because of the quantity and variety of these programmers, traditional defences are becoming inadequate, leaving Android smartphones frequently vulnerable to new viruses. In this study, a simple technique for detecting the Android malware. For many years, malware has been a tool employed in cyberattacks. widespread use of Smartphones are a popular target for malware developers because they contain a lot of sensitive information. A lot of Android malware species infect vulnerable users every day, making manual malware research unfeasible. Android, the most popular mobile operating system, has long been an interesting platform for malware.
developers. Cyber forensic investigators would benefit from using machine learning techniques for malware forensics to help them combat harmful software. In this research, we offer two methods for static analysis of mobile applications that use machine learning. J. Li, L. Sun, Q. Yan, Z. Li, W. Srisa-An, and H. Ye’s Important Permission Identification for Machine-Learning-Based Android Malware Detection is the following publication. In this research, the authors demonstrate how to examine less permissions for mobile malware detection while still keeping high efficacy and accuracy. SIGPID uses a methodical, three-level trimming methodology to extract only significant permissions, and current virus scanners, SIGPID is very successful. It can identify 93.62% of the data set’s malware and 91.4% of new or undiscovered malware. The following work, by A. Saracino, D. Sgandurra, G. Dini, and F. Martineelli, is titled MADAM: Effective and Efficient Behavior-Based Android Malware Detection and Prevention. We introduce a novel multi-level, behavior-based malware detector for Android in this study. MADAM-branded devices This monitoring is done in order to identify app misbehaviour. MADAM uses an anomaly-based approach for some feature groups while implementing an 18 signature-based method for other feature groups that takes into account behavioural patterns that we have deduced from well-known malware misbehaviour. Ultimately, the fact that MADAM is efficient and requires little to no human input makes it usable. It also has no negative effects on the user experience. Attackers expanded their focus on Android smartphones and tablets at the end of 2011, creating and disseminating apps. These apps are difficult to spot because they appear to behave as real programmers that do no harm while actually posing a threat to user data privacy, financial security, and device integrity. MADAM is able to detect misbehaviour from malware behavioural classes that evaluate 125 extant malware families, which include the majority of known malware, by assessing and correlating numerous features at four different Android layers. Ultimately, the fact that MADAM is efficient and requires little to no human input makes it usable. It also has no negative effects on the user experience. According to knowledge, MADAM is the only system that uses a multi-level strategy to attempt to detect and prevent at run module any type of malware. MADAM not only achieves very high runtime detection accuracy,. This study proposes SAMADroid, a revolutionary 3-level hybrid Android malware detection model. It is a mix of the following three stages for analysing and detecting malware: Static analysis and dynamic analysis, both local and remote hosting, and ML intelligence are the first three. In the static analysis phase, trials are carried out to pick the features that can give the most comprehensive and helpful information on the behaviour of the application. Although Drebin lacks dynamic analysis and is unable to detect malicious code that is encrypted or loaded dynamically, it nonetheless detects malware with a high degree of accuracy [3]. During runtime, system calls are tracked for 19 dynamic analysis. To find the most accurate machine learning method, various machine learning algorithms are used, and their performance is compared. The proposed technique aims to solve the present anti-malware systems’ resource efficiency issue. We developed the issue that current research lags in accurately detecting Android malware while assuring the low hardware resource consumption of Android devices based on the advantages and drawbacks of present antimalware techniques. Hence, for the Android operating system, which suggests a 3-level hybrid malware detection architecture. There is no three-level hybrid malware detection system that we are aware of. SAMADroid is an innovative malware detection model as a result because it combines the advantages of machine learning intelligence, dynamic analysis, and static analysis. Moreover, it uses both local and remote hosts to run in order to maximise resource usage. For Android-based devices, the SAMADroid client application has been created.

### III. METHODOLOGY

**Step 1:** Start the algorithm defining an initial set of population generated randomly.

**Step 2:** Assign a fitness score calculated by the defined fitness function for genetic algorithms.

**Step 3:** Selection of Parent Chromosomes with good fitness scores. Parent chromosomes undergo mutation operations to produce next generations.

**Step 4:** Perform crossover and mutation operations on the selected parents with the crossover probability of crossover and mutation probability of mutation.

**Step 5:** Perform the function evaluation on the individuals of the new generation.

**Step 6:** Perform the next generation until the convergence is reached, and the optimal feature subset is chosen.

IV. RESULTS

Firstly we have to upload the data.

After uploading the dataset we have to test the data which we have taken.
Now train the system according to the classifiers we are using.

We got 98.9% of accuracy in svm and for NN accuracy is 98.6%. now we have to check with genetic.

Here is the accuracy graph comparison.

This is the graph which tells the time period to run the software.

V. CONCLUSION

It is important to develop a software so it can detect such malware, most of which spread via malicious applications or malware. In order to obtain the most dataset that can be utilised by training algorithms in the most effective manner, the suggested methodology makes use of an evolutionary genetic algorithm. Using ML classifiers to operate with reduced dimension feature sets, experiments show that a
respectable classification accuracy of more than 94% is maintained, which lowers the training difficulty of classifiers. Genetic Algorithm, future work can be improved by using larger datasets for better outcomes and by examining the impact on other machine learning methods.

VI. ACKNOWLEDGEMENT

We would also like to acknowledge the contribution of our colleagues and fellow researchers who have provided us with valuable feedback and insights throughout the project.

VII. REFERENCES