JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

MALWARE DETECTION USING MACHINE LEARNING

[1]Mrs. Navya VK, Assistant Professor, Department of Computer Science and Engineering, MVJ College of Engineering, India

[2]Jai Krishna Pandey, [3]Priyanshu Anjney, [4]Puttam Venkata Sesha Reddy, [5]Ompuri O

Department of Computer Science and Engineering,
MVJ College of Engineering, Bangalore, India

Abstract: With the growing popularity of Android devices, the threat of malware poses a significant risk to users' system integrity and privacy. To address this issue, a web-based framework has been developed to detect malware from Android devices. The proposed framework uses feature selection approaches to detect malware from real-world apps to select the most relevant features. These selected features and using some machine learning algorithms are made use of to build a model, including deep learning, farthest first clustering, Y-MLP, and nonlinear ensemble decision tree forest. Additionally, rough set analysis is used as a feature subset selection algorithm. The empirical data indicates that the model achieved a detection rate of 86% for identifying malware in real-world apps when all four machine-learning algorithms were used in parallel. This approach is highly effective and efficient in detecting malware belonging to unknown families. Overall, the web-based framework is an essential step in the fight against malware and provides an effective and efficient method for detecting malware from Android devices.

Keywords: Web Based Detection, API, Feature Selection Methods, Machine Learning, Y-MLP

I. INTRODUCTION

Smartphones and their apps have become an essential part of modern human life. However, this dependence on smartphone apps also presents a significant risk as cybercriminals continue to develop malware-infected apps on a daily basis. To protect Android devices from malware, researchers and academics have made it a primary task to develop effective solutions. This project aims to detect malware through a web-based framework that employs feature selection approaches and distinct machine-learning algorithms. To select the best features for training, the framework considers two approaches - feature ranking and feature subset selection. The framework is trained with different machine learning algorithms that work on the principle of supervised, semi-supervised, unsupervised, and hybrid approaches. The framework has achieved high accuracy in detecting malware under numerous data sets, including the Drebin data set. The algorithms used in the framework include Naive Bayes and Random Forest. The Naive Bayes algorithm is known for its accuracy and efficiency in training and generalizing well from small amounts of data. It increases accuracy and efficiency in detecting malware belonging to unknown families and real-world apps. Overall, this framework is an effective solution in the fight against malware and provides a reliable method for detecting malware from Android devices.

II. MOTIVATION

- Today we download a lot of many apps and files on our smartphones and our pc, and we do not check whether the files consist of any malware or not.
- This project will help millions of people as they can check any files or apps they download. This will separate malware files and clean files.
- This project will be free so that everyone in the world can use it because everyone cannot afford to buy or pay for software.
- This is to aim so that people do not fall into the trap of malware or any fraud. Many people are victims of this fraud and this project will help them.

III. PROBLEM STATEMENT

Given a file or an application, the goal is to develop a system that can automatically detect whether it is malware or not. This requires the development of an algorithm or a model that can analyze the characteristics of the file and differentiate between the ones that contain malicious code or behavior from those that are clean.

IV. OBJECTIVES

- Develop an early warning system for computers to detect potential threats.
- Prevent unauthorized access to the computer and protect sensitive information from being compromised.
- Implement machine learning techniques to improve threat detection accuracy.
- Utilize various techniques to enhance the accuracy of threat detection.

V. LITERATURE REVIEW

[1] Dragos Gavrilut, Mihai Cimpoes, Dan Anton, and Liviu Ciortuz utilized machine learning techniques to reduce the number of false positives in their model's results.

[2] Arvind Mahindru and Paramvir Singh discussed how Android permissions can be used by malware to infect devices, emphasizing that users often do not read the permissions before installing apps.

[3] Sanjay Sharma, C. Rama Krishna, and Sanjay K. Sahay highlighted the ineffectiveness of signature-based tools in detecting malware and instead used the frequency of opcode occurrence for malware detection.

[4] Umm-e-Hani Tayyab, Faiza Babar Khan, Muhammad Hanif Durad, Asifullah Khan, and Yeon Soo Lee researched various machine learning methods for detecting different types of malware, identifying the most effective techniques.

[5] Prof. Pritam Ahire, Mohanki Shreya, Shreya Shinde, Preeti Pisal, and Manasi Manikumar examined the efficiency of different machine learning techniques in detecting malware, providing insights for selecting the best approach for a project.

[6] S. Soja Rani and S. R. Reeja demonstrated how malware can penetrate anti-intrusion detection or prevention systems, and suggested various analysis methods to address the issue.

[7] Arvind Mahindru and A. L. Sangal developed a web framework to detect malware, using feature selection methods and comparing their efficiency to identify the most effective approach.

VI. PROPOSED SYSTEM

The process of identifying malware in files and applications is multi-faceted and includes collecting and organizing datasets for machine learning algorithms used in training and testing. The datasets utilized include a training set, a test set, and a "scale-up" set to ensure accuracy. To refine the system's effectiveness, datasets also include clean files that share similar characteristics to malware files. The Virus Heaven collection is the source for malware files used in the training dataset, while the WildList collection and files from different operating systems are included in the test dataset. These datasets encompass a range of malware types, including trojans, backdoors, hack tools, rootkits, worms, and others. The "scale-up" dataset will be split into ten parts, S10, S20, ..., and S100, to test the scalability of the learning algorithms. This division permits the assessment of training speed and the malware detection rate for increasingly more extensive datasets. The project will also gather Android application packages (.apk) and organize them into separate categories. These applications will run on an emulator called Bluestack software, allowing Android applications to function on a PC. The .apk files' permissions will be extracted, alongside other ordinary files from different collections. Five machine learning classifiers - Naive Bayes (NB), Decision Tree (J48), Random Forest (RF), Simple Logistic, and k-star - will evaluate the datasets based on TPR, FPR, Prec., Recall, and F-measure.

To achieve the utmost accuracy, the datasets will undergo three different tests utilizing WEKA software. The first option is to supply a training set and evaluate the test set. The second option is cross-validation, dividing the dataset into subsets, training the algorithm on k-1 of those subsets, and evaluating it on the remaining subset. The third option is to split the dataset on a percentage basis, using a portion for training and the remainder for testing. The collection and preparation of datasets are essential in detecting malware from files and applications using machine learning algorithms. Using different datasets, including clean files that share similarities to malware files, enhances the accuracy of training and testing, leading to a more effective early warning system for computers, preventing hacker attacks, and safeguarding information. Our System also describes various frameworks and approaches for detecting and analyzing malware. One such framework proposed is a behavioral-based framework that uses system calls to analyze and classify suspicious programs. The framework aims to reduce

the false identification of malware and provide a secure environment for end users. Another approach we have tried is a scalable clustering approach that will identify and clusters malware using the ANUBIS system with taint tracking for analyzing behavior. However, the approach has limitations such as trace dependence and dynamic data tainting, which can result in injected malicious binaries.

It also proposes a novel hybrid framework that combines deep transfer learning and machine learning for malware detection. The framework involves generating an image-based PE dataset, normalizing and augmenting it, and evaluating the performance of deep transfer learning models. The second step involves combining deep learning and machine learning models for malware detection. Furthermore, the system will discusses the pros and cons of static and dynamic malware analysis approaches. While the static method provides more specific details about malicious programs, the dynamic method detects malware by running the system. The integration of both approaches can result in better malware detection. It also briefly describes the Random Forest machine learning algorithm, which involves evolving multiple decision trees based on independent subsets of the data set. The algorithm randomly selects parameters and variables for node selection and partitioning and determines the frequency of misclassification to arrive at the total error rate. The final classification result is based on the class with the highest number of votes from the qualified trees.

VII. ADVANTAGES OF THE PROPOSED SYSTEM

- It can identify both known and unknown instances of malware, and the level of false positives is low. Additionally, it can effectively analyse multipath malware.
- The dataset used for this technique can also be applied to cloud-based or clustering malware detection systems. Moreover, it requires fewer resources compared to other techniques.
- This technique is relatively fast and provides accurate results. It is also effective against malware from the same family. It can provide protection from various malware.
- The detection of malware in this technique is not affected by obfuscation or encryption techniques.

VIII. LIMITATIONS AND CHALLENGES

We encountered following challenges and limitations while in the creation of our system:

- Constantly evolving malware: Malware authors regularly modify their tactics to evade detection, which poses a significant challenge for detection systems to keep up with.
- Polymorphic malware: Malware can change its code to avoid detection, making it difficult for traditional signature-based detection systems to identify.
- Zero-day attacks: Zero-day attacks exploit unknown vulnerabilities, which makes it hard for detection systems to detect and prevent them.
- False positives: Malware detection systems may generate false positives, resulting in unnecessary alerts and an increased workload for security teams.
- Resource-intensive: Malware detection systems can be resource-intensive, consuming significant amounts of CPU and memory, which can negatively impact system performance.
- Detection latency: Malware detection systems may not detect malware in real time, leaving systems vulnerable to attack.
- Advanced evasion techniques: Malware uses advanced techniques, such as encryption, to evade detection and infiltrate systems.
- Fileless malware: Fileless malware operates solely in memory and leaves no trace on disk, making it difficult for detection systems to identify.
- End-user behavior: Users can inadvertently download or install malware, bypassing detection systems and increasing the risk of infection.

IX. CONCLUSION

The project titled "Malware Detection Using Machine Learning" is a model for detecting malware from files and apps. This model will differentiate malware from clean files. This model will help millions of people in the world as we will keep this model as free to use for anyone, as many people cannot afford to buy software. This will also minimize the number of people falling into the trap of ransomware and malware. Additionally, our model will use fewer resources compared to other techniques, making it suitable for cloud-based or clustering malware detection systems.

Future work for this project could involve incorporating more advanced techniques such as deep learning or natural language processing for even more accurate results. Nonetheless, our project shows great promise in the field of malware detection and can be a valuable tool for ensuring the security of computer systems.

X. ACKNOWLEDGEMENT

The authors express their heartfelt gratitude to the Head of the Department of Computer Science and Engineering at MVJ College of Engineering for their guidance and support throughout the project. They would also like to thank their project guide for their constant assistance and encouragement. The authors are grateful to their families, friends, and all those who supported them throughout the project. Without the support of our guide and teachers, we would not have been able to do this project.

REFERENCES

[1] E.S. results andQ. Heal, " Q. Heal Daily trouble Report| Q1 2017, " 2017 url http//www.quickheal.co.in/resources/threat-reports.

[2] A. Govindaraju, "total Statistical Analysis for Discovery of Metamorphic Malware," Master's design report, Department of Computer Science, San Jose State University, 2010.

[3] "Discovery of Advanced Malware by Machine Learning ways, " Sanjay Sharma, C. Rama Krishna and SanjayK. Sahay.

[4] Umm-e-Hani Tayyab, Faiza Babar Khan, Muhammad Hanif Durad, Asifullah Khan and Yeon Soo Lee, " A Survey of the Recent Trends in Deep Learning Based Malware Detection, ".

[5] Manavi,F.; Hamzeh, A. A approach for ransomware discovery rested on PE title usinggraphembedding.J. Comput. Virol. Hacking Tech.

[6] Discovery of Advanced Malware by Machine Learning practices, " Sanjay Sharma,C. Rama Krishna and SanjayK. Sahay.

[7] Soja Rani(&)andS.R. Reeja, " A Survey on distinctive patterns for Malware Detection employing Machine Learning patterns"

[8] P.K. Chan andR. Lippmann, " Machine knowledge for computer defense, " Journal of Machine Learning Research,vol. 6,pp. 2669 – 2672, 2006.

[9] G. Dini,F. Martinelli,A. Saracino, andD. Sgandurra. MADAM Amulti- position anomaly sensor for android malware. In Computer Network Security, ser. Lecture Notes in Computer Science,I. Kotenko andV. Skormin, Eds. Springer Berlin Heidelberg, 7531240 – 253, 2012.

[10] A.P. Fuchs,A. Chaudhuri,andJ.S. Foster. Scandroid Automated security instrument of android operations.