IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Malware Detection Using Machine Learning

Yogita Madhukarrao Bhagwat
Computer Engineering
Jai Hind College of Engineering Kuran, Pune
Ganeshshirke100@gmail.com

Mr. Kapil D. Dere
Asst. Prof. Computer Engineering
Jai Hind College of Engineering
Kuran, Pune

Dr. A. A. Khatri
Head of The Department
Computer Engineering
Jai Hind College of Engineering
Kuran, Pune

ABSTRACT: -

Malware detection involves identifying and preventing dangerous software in computer security. It is not the only approach to safeguard a firm against a cyber-attack. Effective risk assessment is essential for both companies and administrations. This study explores various methods for identifying computer malware, harmful software, websites, and future instructions. It also discusses the rise of computer viruses and worms and how to counteract them. Traditional detection approaches are being replaced by innovative procedures and strategies including behavioural-based and signature-based models. Future instructions will focus on developing security solutions to address cyber fraud, a growing issue in the Asia-Pacific area. Traditional computer security measures are insufficient to protect against cyber security fraud and other threats due to their limitations. Researchers have developed new techniques, including heuristic and static analysis, to detect over 90% of malware samples without false positives or negatives.

Keyword: - Malware, Ransomware, Behavior-Based Methodology, Heuristics, Vulnerabilities, Signature-Based Models

INTRODUCTION

Permissions are the basis of the Android security paradigm. Permission is a security feature that restricts access to a certain piece of the device's code or data. The limitation is in place to prevent sensitive data and code from being manipulated to distort or impair the user experience. Permissions are used to allow or deny access to restricted APIs and resources. The Android INTERNET permission, for example, is required for applications to perform network communications; thus, the INTERNET permission limits the establishment of a network connection. In order to read items from a user's phonebook, an application must also have READ CONTACTS permission. It is the developer's responsibility to establish what permissions a program requires. Many users are unaware of what each permission means and approve them without thinking, allowing the application to access sensitive user data. Another flaw is that the user is unable to select which permissions to grant and which to refuse. numerous customers will still agree to install an app even if it demands a questionable permission among numerous seemingly genuine ones. Android has eclipsed iOS

as the most popular operating system for smartphones and tablets, with an estimated market share of 70-80%.

When an unusual app is released, the system recommends checking the Play Store Marketplace to see if it is hazardous. For each new application or updated version of an existing app, compute a risk score during runtime and categorize it based on a preset threshold. Implementing a malware detection system in a real-time Android application environment was a success.

LITERATURE SURVEY

Authorizations fundamental and mentioned are combined with six extra qualities from the manifest and the dismantled code in DREBIN [1]. AI calculations are utilized to consequently figure out the contrast among hazardous and harmless projects. Once prepared disconnected on a devoted machine, the Help Vector Machine is simply communicated the learnt model to the cell phone for recognizing perilous apps.

Huang et al. [2] research the possibility of distinguishing false Android applications utilizing consents and 20 elements from application packs. As per their discoveries, a solitary classifier can recognize around 81% of fake projects. It could be a quick channel to identify more suspect applications, as per them, by incorporating discoveries from a few classifiers.

Liu and Liu [3] utilize mentioned and vital consents by an application likewise. AI calculations and consents are utilized in this framework to classify an application as harmless or harmful.

Sanz et al. [4] propose an original methodology for identifying deceitful Android applications utilizing AI strategies and inspecting the application's separated consents. The presence of labels utilizes authorization and utilizations highlight in the manifest, as well as how much consents allowed to every application, are used to arrange them.

As indicated by [5] is a way for building AI classifiers and distinguishing malware by removing various properties from the Android manifest. These elements are the particular consents looked for and the purposes feature>> tag.

As per [6] present a structure for fostering an AI based malware identification framework for Android to recognize malware applications and further develop Cell phone clients' security and protection. This framework gathers various consent based qualities and occasions from Android applications and examinations them utilizing AI classifiers to decide whether the application is harmless or malicious.

Shabtai et al. [7] partition Android applications into two classifications: utilities and games. Effective partition among games and devices, as they would like to think, ought to offer a decent sign of such frameworks' ability to learn and demonstrate Android harmless projects and conceivably distinguish malware documents utilizing AI (ML) procedures on static credits gathered from Android program files.

The journalists of these books limit their exploration to the most frequently mentioned consents (or a specific determination of authorizations) [8]. Notwithstanding, contingent upon the attack, consents like READ LOGS may be similarly pretty much as hazardous as others (like Web). Each grant ought to be painstakingly evaluated as having the capacity to be hazardous when matched with another. This technique for choice, as per [9], creates impressively slanted results. AI based discovery methods are perceived to have two downsides: they have a high pace of phony problems, and picking which qualities ought to be gotten the hang of during the preparation stage is a troublesome errand. The technique of picking datasets for preparing is thusly a pivotal stage in these frameworks. The presentation of the classifier improves with time: for a specific month Mi, whose applications were utilized for the preparation datasets, the resultant classifier turns out to be less and less equipped for distinguishing all malware in ensuing months.

To address the applications, most of these endeavors separate a list of capabilities. The data conveyed by such attributes fluctuates relying upon the gig. There is no proof to demonstrate which ascribes give the best discovery results, but each exploration considers required consents. Moonsamy et al. [10] are keen on involving authorizations as the sole element to portray programs and recognizing specific consent examples to recognize spotless and malevolent applications.

OBJECTIVES

- To design developed a system for detect the malicious contents from third party API's which is generally used for android application development.
- To implement a machine learning or deep learning algorithm to mine the API codes.
- To validate the entire API's using background Knowledge which works like supervised learning approach.

PROPOSED METHODOLOGY

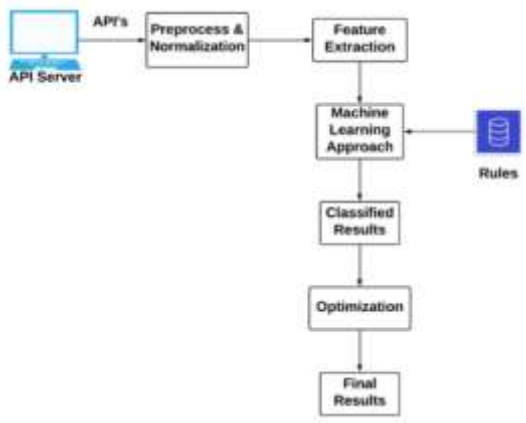

Fig: - System Architecture

The system suggests methods for detecting malware. We employed the following methods: I modified random forest classifier parameters; (ii) similarity-based permission feature selection; (iii) association rule mining; (iv) permission ranking-based feature selection; and (v) modified random forest classifier parameters. Both the permission feature selection technique based on similarity and the permission ranking-based feature selection strategy assign a frequency rating to the attributes. The association rule mining technique, which is widely used in both benign and malicious programs, is used to remove the rights. Additionally, we improve the detection accuracy of permission-induced malware using the random forest technique. The improved random forest constantly removes unnecessary features. By contrasting essential and non-essential features, we create an improved random forest method with fewer but more crucial features.

CONCLUSION: -

This suggested solution uses real-world Android apps, both malicious and benign, to identify hidden malware patterns. Few attempts have been made to solve the malware detection issue from an API perspective; most previous research has been on permissions, sensitive resources, intents, etc. In order to close this gap, we examine how dangerous apps use APIs in comparison to benign apps. By training the random forests classifier with fine-grained features, we can mine malware patterns and extract extremely sensitive APIs. We suggest an automated malware detection solution using a machine learning algorithm to help Android app markets.

REFERENCES:

[1] Burdisso, Sergio G., Marcelo Errecalde, and Manuel Montes-y-G˜A lmez. "A text classification framework for simple and effective early depression detection over social media streams." Expert Systems with Applications 133 (2019): 182-197

[2] Stankevich, Maxim, et al. "Depression detection from social media profiles." International Conference on Data Analytics and Management in Data Intensive Domains. Springer, Cham, 2019.

[3] Al Asad, Nafiz, et al. "Depression detection by analyzing social media posts of user." 2019 IEEE International Conference on Signal Processing, Information, Communication Systems (SPICSCON). IEEE, 2019.

[4] William, David, and Derwin Suhartono. "Text-based depression detection on social media posts: A systematic literature review." Procedia Computer Science 179 (2021): 582-589.

[5] Tadesse, Michael M., et al. "Detection of depression-related posts in reddit social media forum." IEEE Access 7 (2019): 44883-44893.

[6] Shah, Faisal Muhammad, et al. "Early depression detection from social network using deep learning techniques." 2020 IEEE Region 10 Symposium (TENSYMP). IEEE, 2020.

[7] Chiong, Raymond, Gregorious Satia Budhi, and Sandeep Dhakal. "Combining sentiment lexicons and content-based features for depression detection." IEEE Intelligent Systems 36.6 (2021): 99-105.

[8] Narayanrao, Purude Vaishali, and P. Lalitha Surya Kumari. "Analysis of machine learning algorithms for predicting depression." 2020 international conference on computer science, engineering and applications (iccsea). IEEE, 2020.

[9] Laijawala, Vidit, et al. ”Classification algorithms based mental health prediction using data mining.” 2020 5th International Conference on Communication and Electronics Systems (ICCES). IEEE, 2020.

[10] AlSagri, Hatoon S., and Mourad Ykhlef. ”Machine learning-based approach for depression detection in twitter using content and activity features.” IEICE Transactions on Information and Systems 103.8 (2020): 1825-1832