Advanced Research on Information Systems Security

ISSN: 2795-4609 | ISSN: 2795-4560
Print & Online

Can machine learning be used to detect malware? Android OS Case Study

André Lima*

ISLA – Polytechnic Institute of Management and Technology, Vila Nova de Gaia, Portugal
Email: andrefmlima@gmail.com

Abstract

Nowadays everyone has one or even more than one smartphone or tablet. The existing applications with the most diverse purposes allow us to perform a series of tasks such as using home banking or checking the email, using only our smartphone/tablet. Android OS being one of the most used in this type of equipment becomes an appealing target for viruses, malware, and others. At a time when technology is evolving faster and faster, both in terms of hardware and software, Artificial Intelligence has more and more weight in technological evolution, being used in the most diverse purposes. This review aims to demonstrate how Machine Learning can assist in identifying vulnerabilities in Android OS.

Keywords: Android; Malware Detection; Vulnerabilities; Machine Learning.

* Corresponding author. Email address: andrefmlima@gmail.com

1. Introduction

The use of smartphones and mobile applications are increasing rapidly [1] due to the convenience and efficiency in various applications and the increasing improvement of hardware and software in smart devices. It is predicted that there will be 6,8 billion smartphone users by 2023 [1]. Android has a market share of 71,96% [2]. The second largest market share of 27,48 % is owned by Apple iOS [2].

The popularity of Android and the high number of devices/users of this operating system "puts" a very large target on it.

There are more and more threats to Android, both at the level of vulnerabilities in the operating system itself, as well as in the different "app stores", since besides the official "Play Store", there are several alternatives where we can download the APKS of the applications. These applications, coming from unknown sources, may contain malicious code that compromises the security and privacy of the data on the various devices.

While it is true that equipment manufacturers are increasingly concerned with security, releasing security updates more and more frequently, on the other hand, also increases the frequency with which "attackers" develop, for example malware, that exploit the vulnerabilities in these same updates. In this sense, there are several case studies, and that Machine Learning can contribute to the identification of malware on Android, through source code analysis.

Artificial intelligence has an increasing presence in our daily lives, be it through data collection/analysis, support for existing tools, applications in the automotive sector such as "autonomous driving", among others. This review aims to talk a bit about some of the Machine Learning methods that can contribute to the identification of malware on Android OS [3].

2. Materials and methods

This review was conducted according to the Preferred Reporting Items for Systematic reviews and Meta-Analysis (PRISMA) model [4].

First, define the problematic that the research would focus on, and then what the research criteria would be. Afterwards, a filtering of the articles found was carried out to draw the respective conclusions.

2.1 Research Question

This systematic review aims to answer the following research question: Can machine learning be used to detect malware?

2.2 Search Strategy

To conduct the research in a cabal methodology for the systematic literature review, PRISMA methodology was applied. The following references and research content were obtained from Institute from its IEEE Xplore Digital Library, Web of Science and Springer Link, Google Scholar, and Research Gate, and include the following information:

Keyword	Operator
Android	AND
Malware detection	OR
Vulnerabilities	AND
Machine Learning	OR

Table 1 – Exclusion of Duplicate articles.

2.3 Study Selection Criteria

By searching the indicated platforms, we obtained more than 1000 records, however, we only considered a few articles in English, since 2019 (since the technical evolution is quite fast, it would not make sense to consider articles on this subject), with relevant information to the question presented.

In order to conduct a systematic review, the collected studies were processed recurring to PRISMA approach. Figure 1 shows the revision method that was used.

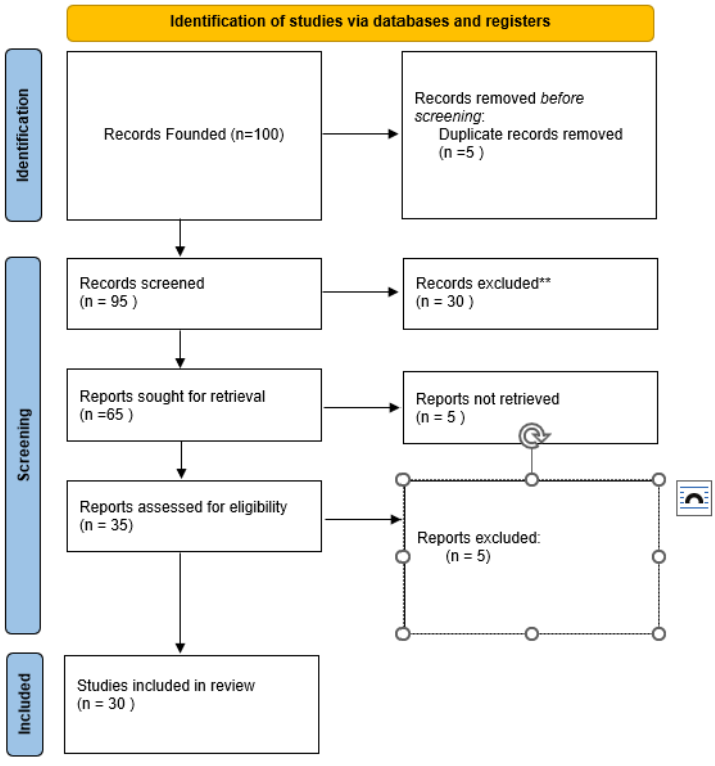

Figure 1 - Prisma review flowchart

3. Results

Thirty articles were selected within the previously defined criteria, whose information could answer the question initially raised.

Following are the most relevant topics and machine learning methods applied in Android OS.

3.1 Android Architecture

Android is a open source OS, that was developed in the form of a "software stack". This stack is based on the Linux kernel and is followed by a library layer, an android runtime layer, an application framework layer and finally an application layer [16].

In the second layer we have the libraries of the operating system itself, developed in java, containing libraries for the interface, navigation functionality, among others.

The Android Runtime is a kind of virtual machine, in this case the Dalvik Virtual Machine. This allows the Linux core, for example, to be used in cases like memory handoff and multi-task execution management. The application framework provides java classes that can be used for example to share data between applications.

Finally in the application bed, we have the applications themselves, like texting, contacts and others[5].

3.2 Android Security Features

Android has a number of features that are intended to make the system as secure as possible, namely, App sandbox, App signing, Authentication, Biometrics, Encryption, Keystore, Security-Enhanced Linux and the Trusty Trusted Execution Environment (TEE). These features not only increase the security of the devices and their data, but also allow applications developed by third parties to use these same features from the ground up [6].

Malware in Android, as in other operating systems (mobile or not), can have the following consequences: data loss/stolen, identity theft, remote access to the equipment (backdoors), access to the user's "activities" following the equipment's use, and Ransomware (which in these cases can lead to an eventual ransom demand for the "data"). Google describes malware as potentially harmful applications [7], [15].

3.3 Machine Learning used for Malware Detection

Within the field of Artificial Intelligence, Machine Learning can be defined as a set of techniques that can predict scenarios or classify data based on previous experiences, i.e., it uses previously analyzed and classified data to classify "new" data.

We have cases in which Machine Learning is useful for example in problem solving, evaluation of the efficiency of certain models. In this case Machine Learning is used to identify malware on Android, and there are a number of methods for that purpose.

The efficiency or degree of reliability of the models used is dependent on a number of factors. The data to be "tested" must be of "quality", to reduce the likelihood of wrong conclusions or false positives. Over time several datastores have been created, with data about certain applications and their respective malware classifications [8].

There are also some platforms, where it is possible to extract a list of applications, as well as their metadata, hash values.

This type of datasets allows to improve the "training" and "classification" of Machine Learning methods on the analyzed applications in the future [8]. In this specific case, although there are several studies on this subject, the analysis of the methods and results obtained has as its source the information collected in [3].

We can state that the process works as follows: it acts on a set of data existing in the DataSet, processing this data, for later action by the artificial intelligence. At this point both Deep and Machine Learning methods can be applied. Based on the assumptions defined when creating the methods, the system autonomously assigns the classification, i.e., we are "dealing" with malware or not [9].

As Machine Learning techniques we can consider supervised, unsupervised, deep and online learning [10]. The supervised approach is based on previously classified data and draws conclusions about the new data it will analyze. Usually this approach uses classification and regression models, specifically Regressions, Decision Trees, Support Vector Machines, the Naive Bays model and K-Nearest Neighbor [10].

In the unsupervised model, and since it works without having to have data "marked", it may be faster to implement, however it may not be as accurate as the previous one [10].

3.4 Machine Learning Techniques

There are three types of malware identification techniques using Machine Learning, they are Static, Dynamic and Hybrid.

Static analysis can be performed by analyzing the application's source code, or through the process of reverse reengineering, so it can be used without the need to use a mobile device.

Dynamic analysis can analyze the application while it is running on the system, however this carries security risks, since if the application contains malicious code, it is running on the device. Hybrid analysis combines the two approaches mentioned above.

About Static Analysis, we can consider Manifest File Analysis and Manifest and Dex Code Analysis.

One of the points to look at when we are trying to identify a certain application as containing malware or not, is through the system access permissions that it "requests". These same permissions are stored in the "Manifest File" of the Android [10]. In principle, any kind of permission that could endanger the system, or attempts to have Root permissions, should be considered suspicious. To help filter these same permissions, Google has documented/identified some of these permissions [11], [14].

In addition to Manifest file analysis, static analysis can also be performed on Dex files, i.e., bytecode files can be analyzed, for example to identify the API's that a particular application uses [10].

3.5 Malware Detection Tools

Following the research, and through some of the studies analyzed we can highlight some malware detection tools such as CuckooDroid, FlowDroid or DroidBox [12].

Cuckoo

It is a python-based tool that has the ability to analyze malware through the use of virtual machines. It can identify the files that the malware creates/accesses, as well as for example API calls or memory dumps. This software makes a dynamic analysis of the system [12].

FlowDroid

This java-based tool allows you to identify possible vulnerabilities associated with a particular application that may jeopardize the user's privacy [12].

DroidBox

This dynamic analysis tool, of adroid applications, allows, through the use of Android ADB, to send activity reports on the use of features such as communications, messages sent and others [12], [14].

4. Discussion

Although this article takes a rather superficial approach to the application of machine learning to detect malware on Android, we can conclude that the application of the different existing methods can contribute to a faster identification of malware on Android devices.

The evolution of technology in recent years has played a crucial role in the development of society and organizations worldwide, the growing number of active devices and the features associated with them can greatly facilitate the life and tasks of each one. However, and as they are increasingly used, they become a tempting target for organizations / people not so "well intentioned", whose goal / challenge is to put applications / malicious software on devices.

As is obvious, much of the "prevention" or filtering of "trusted" sources, is very much up to the user of the equipment, however this type of malware detection methods/tools, can have a crucial role in preventing/detecting undesirable software on our equipment and thus save for example the data that is stored on it.

The various methods identified throughout the articles consulted in this research, as well as others that, although they were analyzed, were not described in the article, demonstrate that machine learning can indeed be a very valid tool for identifying malware on the Android OS [12], [13].

5. References

[1] "Number of smartphone subscriptions worldwide from 2016 to 2021, with forecasts from 2022 to 2027." [Online]. Available: https://www.statista.com/statistics/330695/number-of-smartphone-users-worldwide/.

[2] "Mobile Operating System Market Share Worldwide." [Online]. Available: https://gs.statcounter.com/os-market-share/mobile/worldwide.

[3] J. Senanayake, H. Kalutarage, and M. O. Al-Kadri, "Android Mobile Malware Detection Using Machine Learning: A Systematic Review," *Electronics*, vol. 10, no. 13, p. 1606, 2021. [Online]. Available: https://www.mdpi.com/2079-9292/10/13/1606.

[4] M. J. Page *et al.*, "The PRISMA 2020 statement: an updated guideline for reporting systematic reviews," *BMJ*, vol. 372, p. n71, 2021, doi: 10.1136/bmj.n71.

[5] Google. "Android Platform Architecture" https://developer.android.com/guide/platform (accessed 20/12/2022.

[6] Google. "Android Security Features." https://source.android.com/docs/security/features (accessed.

[7] S. Hamdi *et al.*, "A Comprehensive Study of Malware Detection in Android Operating Systems," *Asian Journal of Research in Computer Science*, vol. 10, pp. 30-46, 07/20 2021, doi: 10.9734/AJRCOS/2021/v10i430248.

[8] K. Liu, S. Xu, G. Xu, M. Zhang, D. Sun, and H. Liu, "A Review of Android Malware Detection Approaches Based on Machine Learning," *IEEE Access*, vol. PP, pp. 1-1, 07/01 2020, doi: 10.1109/ACCESS.2020.3006143.

[9] H. Alkahtani and T. H. H. Aldhyani, "Artificial Intelligence Algorithms for Malware Detection in Android-Operated Mobile Devices," (in eng), Sensors (Basel), vol. 22, no. 6, Mar 15 2022, doi: 10.3390/s22062268.

[10] A. Muzaffar, H. Ragab Hassen, M. A. Lones, and H. Zantout, "An in-depth review of machine learning based Android malware detection," Computers & Security, vol. 121, p. 102833, 2022/10/01/ 2022, doi: https://doi.org/10.1016/j.cose.2022.102833.

[11] Google. "Permissions and APIs that Access Sensitive Information." https://support.google.com/googleplay/android-developer/answer/9888170?hl=en (accessed 20/12/2022, 2022).

[12] H. Shahriar, M. A. I. Talukder, and M. S. Islam, "An Exploratory Analysis of Mobile Security Tools," 10/12 2019.

[13] Duarte, N., Coelho, N., Guarda, T. (2021). Social Engineering: The Art of Attacks. In: Guarda, T., Portela, F., Santos, M.F. (eds) Advanced Research in Technologies, Information, Innovation and Sustainability. ARTIIS 2021. Communications in Computer and Information Science, vol 1485. Springer, Cham. https://doi.org/10.1007/978-3-030-90241-4_36

[14] F. Alves, N. Mateus-Coelho and M. Cruz-Cunha, "ChevroCrypto – Security & Cryptography Broker," 2022 10th International Symposium on Digital Forensics and Security (ISDFS), Istanbul, Turkey, 2022, pp. 1-5, doi: 10.1109/ISDFS55398.2022.9800797.

[15] N. Mateus-Coelho and M. Cruz-Cunha, "Serverless Service Architectures and Security Minimals," 2022 10th International Symposium on Digital Forensics and Security (ISDFS), Istanbul, Turkey, 2022, pp. 1-6, doi: 10.1109/ISDFS55398.2022.9800779.

[16] Nuno Mateus-Coelho, A New Methodology for the Development of Secure and Paranoid Operating Systems, Procedia Computer Science, Volume 181, 2021, Pages 1207-1215, ISSN 1877-0509, https://doi.org/10.1016/j.procs.2021.01.318.