Android Malware Detection and Malware Behavior Analysis Based on Machine Learning

Jingya Liu

School of Software, Zhengzhou University, 450002

Keywords: Android; Malicious software; Testing; Malicious act

Abstract: With the rapid development of mobile Internet, Android system occupies most of the market share of mobile platforms, while the number of Android applications released is also on an explosive growth. Accurate and efficient Android malware detection technology is not only the urgent need of users for their own security, but also the premise of Android market development. The openness of Android platform and the lack of relevant regulatory standards make it a hotbed for malware. However, the research on detection and defense of Android malware is still in the initial stage. Thanks to the enhancement of data processing ability in the cloud, it has become a trend to apply data mining, machine learning and other classic theories and tools to the field of mobile terminal malware detection. The purpose of this paper is to study the Android malware detection based on machine learning classification algorithm, to realize the security model of static detection and dynamic detection, terminal and cloud, detection and control.

1. Introduction

Android system with its open source, free features, makes it in today's smartphone market share has reached 75% [1]. Android application development threshold is low, the number of people using Android mobile phones is large, and the application software developed for it is various and numerous. With the rapid growth and worsening of Android malware, there is an urgent need for effective methods to detect Android malware [2]. More and more people use Android phones to store and process personal data, including personal privacy data related to information security, such as contact information, geographic location, web browsing records, etc. [3]. Malicious software will not only collect the user's geographic location, address book, SMS and other private information, but also maliciously deduct fees and consume system resources, bringing harm to users and mobile phones. Although Android itself has provided corresponding security measures, there are still many defects and problems, increasing the security risks of the platform [4]. Attackers exploit a variety of malware steals, listen for sensitive information, and send malicious toll links and text messages based on existing security vulnerabilities. Some existing detection methods are relatively simple to design, some only match the characteristic behavior, and some only count the frequency of the kernel call. Moreover, these methods do not fully consider the need to overcome malware upgrades and variants in practical applications.

Android security has become a hotspot of security research. It is an important Android security research direction to strengthen system security through permission mechanism extension [5]. Due to the increase in the number and variety of malicious code and the continuous updating of technology, typical analysis methods based on signature and behavior analysis have lost some timeliness [6]. The key difference between smart devices and traditional non-smart devices is that ordinary users can easily download and install third-party applications from online stores, resulting in widespread malware distribution, resulting in more security and privacy than traditional devices. Question [7]. Among the many detection methods, behavior-based detection methods are the hotspots of research [8]. In testing, usually only a small number of samples are applied to verify whether the detection method used can identify malware, and then the relevant conclusions are reached [9]. Thanks to the enhancement of the ability to process data in the cloud, it has become a trend to apply the classic

theories and tools of data mining and machine learning to the field of malware detection in mobile terminals [10]. This article analyzes an automated Android malware analysis tool. It can not only predict whether the software is malware, but also analyze its malicious behavior. It uses machine learning to learn the behavior of malware from annotated samples and builds models to predict software categories for unlabeled test sets.

2. Detection Scheme Based on Machine Learning

Malware is a huge threat to Android system. Due to the continuous use of new technologies and methods by malicious software, people are constantly applying various algorithms and technologies to Android malicious code detection. During the installation phase of Android system, it is necessary to comprehensively analyze the security performance of the system. At this time, it is also necessary to check the specific structure of Android system structure to check whether it can prevent intrusion and ensure that the network system meets the current development requirements. The sharing of information resources has effectively improved social productivity. However, at the same time, it not only creates a wider space for the spread of the virus, but also spreads the spread and harm of the virus. Digital signature technology is actually to sign a private key number to an application program, and the same private key can sign multiple programs. Android system provides developers with application framework. Developers can make better use of framework API through open source platform to achieve network access, location information, multimedia, background services and other functions. Security requirements are users' security requirements in system information integrity, availability, confidentiality and so on, especially it adopts a series of security policies. Security requirements are the most important issues involved in current technological development and economic exchanges in different fields. Without effective guarantee of safety, economic transactions and business development cannot be reasonably carried out.

Security policy is used to determine whether the master has access to the object, and the actions of system users or processes should meet the requirements of security behavior. There are many loopholes in the Android system structure, and attackers repeatedly attack weak links according to certain attack patterns. Theoretically, only the attacker will try enough time to complete the attack on the vulnerability. For the management of system service calls, Android has a special service management process serviceManager. Each service must be registered with serviceManager when it is created, and the application process can only call the corresponding service through serviceManager [11]. In addition to the disclosure of private information, the security of Android's structural system will be greatly reduced if the content of stored information is arbitrarily changed without the permission of the parties concerned, or if users are illegally prevented from obtaining information and providing false information to interfere with the judgment of others.

After data preprocessing, the data has been formally expressed in the form of a two-dimensional table that can be analyzed. When the newly developed Xinanjou system structure system is put into the market, professionals should carry out system testing to prevent system installation defects. The research on the reliability of network topology needs a comprehensive, complete and reasonable analysis from the two aspects of network devices and service load. The algorithm flow is shown in Fig. 1.

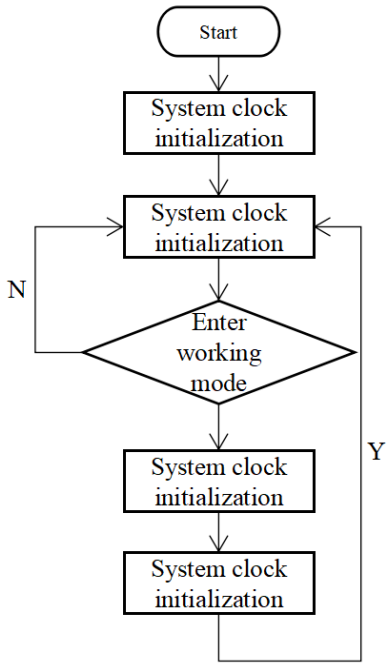

Fig. 1 Algorithm flow

3. Design and Implementation of Detection Methods

From a regulatory point of view, the mainstream third-party application market is seeing more and more applications coming online every day, and the quality is uneven, requiring automated testing for security review. The topology of the model is based on the structure and function of Android structural devices. It needs to locate its main working mode and connect multiple devices to ensure the security of information data. People in different fields have different requirements and different priorities for the security requirements of Android structural systems. In the network system, the storage and transmission of information data need a certain quality medium. System calls in the kernel layer can reflect the interaction characteristics between the application layer and the underlying system, and can also collect program execution information from the underlying, which is accurate and not easy to be interfered and bypassed. Malware has some common behaviors, which can be considered as potentially malicious behaviors [12]. In machine learning problems, for a complex task, a single algorithm performs better on a specific data set and worse in other cases. The application program interface layer provides packaged services for the software, and the monitor is arranged on the application program interface layer and can intuitively monitor the behavior of the software. Attackers usually inject malicious code into popular regular software, repackage it, adopt the same name or use confusing means such as more attractive update functions to form new similar software. If criminals want to invade the system through technical means, they need to find the system entrance and break through the protection barrier. And increase the difficulty of attack and ensure network security.

In order to solve the problems in the field of Android malware, it is necessary to learn based on the existing deep learning methods of machine learning and redesign the processing methods of discrete data. Determine and calculate inspection statistics. In the hypothesis test of two independent sample ratios, the statistical data used are:

$$e_j = -k\sum_{i=1}^{n} f_{ij} \ln f_{ij} \tag{1}$$

Can get:

$$W_j = 1 + k\sum_{i=1}^{n} f_{ij} \ln f_{ij} \Big/ \sum_{j=1}^{m} (1 + k\sum_{i=1}^{n} f_{ij} \ln f_{ij}) \tag{2}$$

Replace data with calculations:

$$W_j = d_j \Big/ \sum_{j=1}^{m} d_j \tag{3}$$

After the configuration is completed, run four data analyses. The number of information nodes used for each analysis is different, and the corresponding processing time is also different. For example, Table 1 shows the number of nodes and processing time for each analysis. The relationship between the number of nodes and processing time is shown in Fig. 2.

Table 1 Number of nodes and processing time

Serial number	Number of nodes	Processing time
1	2000	120000
2	3500	290000
3	5000	710000

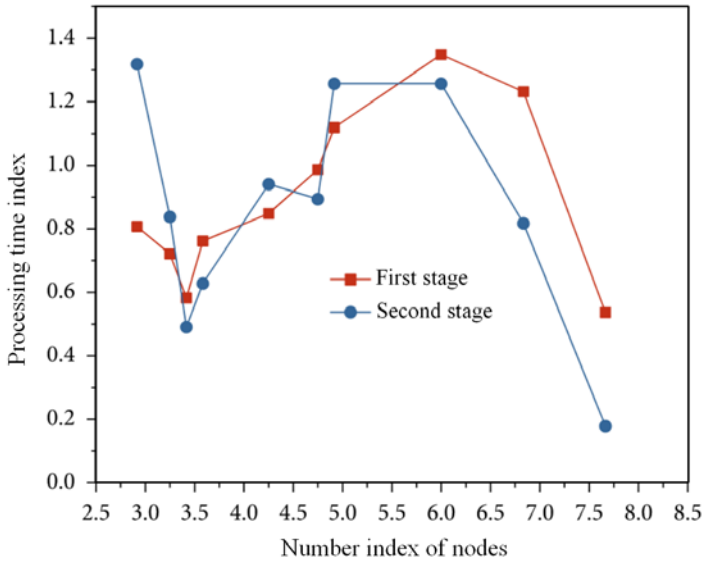

Fig. 2 Number of nodes and processing time

Although in theory, information gain can be used for feature selection in machine learning classification problems, it can bring noise to classification features in practical applications. The weighted average of the prediction results of multiple models is based on the fact that initially, each training sample is given the same weight, and through multiple rounds of training, the training failed samples are given greater weight. When the application is installed, the user can only accept all permission requests or reject all permissions, which makes it impossible to install the application. This coarse-grained permission grant mode increases the possibility of malware stealing data. Each privilege escalation is likely to prepare for the next attack. When the system is attacked, the user cannot use the information normally and cannot operate the operating system. The virus may not only cause personal information to be leaked, but also may cause repeated invasion in future use, which will affect users or economic and social public opinion. The person in charge of security can analyze the attack path to explore the attacker's attack target, and then combine with other models to find the vulnerability efficiently and accurately. Android permissions are mainly used to restrict the use of certain restrictive features within applications and component access between applications. In order to monitor the terminal's security status, the system uses Android monitoring mechanism to monitor the communication information, unauthorized installation and uninstallation of programs, and sensitive behavior status of the network status system. It also analyzes malicious behavior based on specific behavior patterns and provides defense log function for users to view.

4. Summary

Accurate and efficient Android malware detection technology has important practical value, how to better detect malware is still the focus of research. With the further development of the mobile Internet and the maturity of the business model of the mobile application market, people's demand for malicious application monitoring technology on the mobile terminal will be more urgent. The research of Android security is more and more urgent. It is an important research direction to strengthen the security of Android system and prevent privacy disclosure through the extension of permission mechanism. Although Android itself has provided the corresponding security mechanism, there are many defects and loopholes, which lead to greatly reduced platform security. Real user data may not be comparable due to different user usage habits and preferences, and it is also very dangerous for users to install malicious software. Since the logic of the program code will not change during operation, the characteristics of the program during operation can be effectively characterized by counting the access of various interfaces. In real life, each feature selection method can be appropriately improved to improve the processing effect. The implementation of the Android platform malware dynamic detection system should fully consider its particularity, not only to ensure that the detection can be performed locally, but also to occupy as few hardware resources as possible.

References

[1] Singh A K, Jaidhar C D, Kumara M A A. Experimental analysis of Android malware detection based on combinations of permissions and API-calls[J]. Journal of Computer Virology and Hacking Techniques, 2019, 15(4):1-10.

[2] Andoor J T. A Filtering Based Android Malware Detection System for Google PlayStore[J]. Advances in Intelligent Systems and Computing, 2014, 327:559-566.

[3] Lee H T, Kim D, Park M, et al. Protecting data on android platform against privilege escalation attack[J]. International Journal of Computer Mathematics, 2016, 93(2):401-414.

[4] Bastani O, Anand S, Aiken A. Interactively Verifying Absence of Explicit Information Flows in Android Apps[J]. ACM SIGPLAN Notices, 2015, 50(10):299-315.

[5] Bielik P, Raychev V, Vechev M. Scalable Race Detection for Android Applications[J]. ACM SIGPLAN Notices, 2015, 50(10):332-348.

[6] Singh R, Kumar H, Singla R K, et al. Internet attacks and intrusion detection system: A review of the literature[J]. Online Information Review, 2017, 41(2):171-184.

[7] Laurent Delosières, David García. Infrastructure for Detecting Android Malware[J]. Lecture Notes in Electrical Engineering, 2014, 264:389-398.

[8] Nauman M, Tanveer T A, Khan S, et al. Deep neural architectures for large scale android malware analysis[J]. Cluster Computing, 2018, 21(3):1-20.

[9] Wuchner T, Cislak A, Ochoa M, et al. Leveraging Compression-Based Graph Mining for Behavior-Based Malware Detection[J]. IEEE Transactions on Dependable and Secure Computing, 2019, 16(1):99-112.

[10] Bat-Erdene M, Park H, Li H, et al. Entropy analysis to classify unknown packing algorithms for malware detection[J]. International Journal of Information Security, 2016, 16(3):1-22.

[11] Aljohani M, Alam T. Real Time Face Detection in Ad Hoc Network of Android Smart Devices[J]. Advances in Intelligent Systems and Computing, 2016, 509:245-255.

[12] Kaur P, Sharma S. Spyware Detection in Android Using Hybridization of Description Analysis, Permission Mapping and Interface Analysis[J]. Procedia Computer Science, 2015, 46:794-803.