Risk for Android Mobile Applications
Aashay A. Raje, Bhagyashri B. Bhole, Akshay Aachat, Revati Baviskar
Department of Computer Engineering
L.G.N.S.C.o.E, India

Abstract:
Before a user installs an app, warns the user about the permissions the app requires, trusting that the user will make the right decision. One of Android’s main defense mechanisms against malicious apps is a risk communication mechanism which, as it presents the risk information of each app in a “stand-alone” fashion and in a way that requires too much technical knowledge and time to distill useful information, this approach has been shown to be ineffective. In order to generate another metric that users can utilize when choosing apps, we discuss the desired properties of risk signals and relative risk scores for Android apps. We present a wide range of techniques to generate both risk signals and risk scores that are based on heuristics as well as principled machine learning techniques. Experimental results conducted using real world data sets show that these methods can effectively identify malware as very risky, are simple to understand, and easy to use.

Index Terms: Risk, mobile, malware, data mining

I. INTRODUCTION

MOBILE devices are becoming ubiquitous, and they provide access to personal and sensitive information such as phone numbers, contact lists, geolocation, and SMS messages, making their security an especially important challenge. Compared with desktop and laptop computers, mobile devices have a different paradigm for installing new applications. Most of which are from reputable vendors, with niche applications increasingly being replaced by web-based or cloud services, for traditional personal computers, a typical user installs relatively few applications. For mobile devices, one often downloads and uses many applications (or apps) with limited functionality from multiple unknown vendors. A large degree on decisions made by the users, therefore, the defense against malicious applications must depend. To enable the user to make informed decisions about whether to choose and install specific apps, an important part of malware defense on mobile devices is to communicate the risk of installing an app to users. With a focus on the Android platform, we study how to effectively evaluate the risk of mobile applications. The Android platform has emerged as one of the fastest growing operating systems. In May 2013 Google Inc. announced that 900 million Android devices have been activated. Additionally Google Play (formerly known as Android Market) crossed more than 48 billion downloads, and is now averaging about 2.5 billion downloads per month. makes Android an attractive target for malicious developers that seek personal gain while costing users’ money and invading users’ privacy, such as with user base, coupled with ease of developing and distributing applications. Trusting that the user will make the right decision, one of Android’s main defense mechanisms against malicious apps is a risk communication mechanism which warns the user about the permissions an app requires before the app is installed by the user. The specific approach used in Android has been shown to be ineffective at informing users about potential risks. The majority of Android apps request multiple permissions.

warnings quickly lose any effectiveness as the users are conditioned to ignore such warnings, when a user sees what appears to be the same warning message for almost every app, the risk information of each app in a “stand-alone” fashion and in a way that requires too much technical knowledge and time to distill useful information, we believe that the main reason for the failure of the current Android warning approach is that it presents, as a mechanism to improve the existing warning mechanism, recently, binary risk signals based on the set of permissions an app requests have been proposed. Requesting a certain permission or certain combinations of two or three permissions triggers warning that the app is risky. In this paper, we investigate permission-based risk signals that use the rarity of critical permissions and pairs of critical permissions. 26 permissions that have significant security or privacy impact are identified as critical, and if an app requests a critical permission (or a pair of critical permissions) that is rarely requested by apps in the same category as the app, the app is labeled as risky, in this approach, initially reported. labeling each app as either risky or not risky, however, has significant limitations, using a binary risk signal. many more apps are in a gray area where their behaviors are questionable, but not overly malicious, while some apps are clearly malicious. Furthermore, whether an app is malicious or not may depend on the user’s privacy preference. We thus propose the concept of risk scoring functions, a binary risk signal forces the designer to draw a line somewhere, when no line may be “correct.”. Which indicates how risky the app is, Such a function assigns to each app a numerical score. Each app’s risk is presented in a way so that it can be easily compared to other apps, this approach presents “comparative” risk information. A risk signal by choosing a threshold above which the signal is raised, given a risk scoring function, one can construct. For risk communication in the following way, however, we believe that it is better to use arise scoring function. Identifying the percentile of the app in terms of its risk score, given this function, one can compute a risk ranking for each app. This percentile number has a well defined and easy-to-

understand meaning. Users can appreciate the difference between an apps ranked in the top 1 percent group versus one in the bottom 50 percent. Translated into categorical values such as high risk, medium risk, low risk, and very low risk, this ranking can be presented in a more user-friendly fashion. When choosing a mobile app, an important feature of the mobile app ecosystem is that users often have choices and alternatives. Then that may cause the user to choose the less risky one. Such an approach complements well other approaches that try to identify malicious apps, if the user knows that one app is significantly more risky than another with similar functionality, the remaining ones can be ranked according to their risks, after malicious apps are removed. In this paper, we examine two approaches to develop risk scoring functions based on permissions requested by apps. Which have been used extensively in various applications in machine learning, computer vision, and computational biology, to model complex data, the first approach, initially reported, uses probabilistic generative models. One strength of this technique is that it can model features from a large amount of unlabeled data. Using these models, we assume that some parameterized random process generates the app data and learn the model parameters based on the data. Then we can compute the probability that each app was generated by the model, so that the lower probability translates into a higher risk score, the risk score can be any function that is inversely related to the probability. We consider four models: Basic Naive Bayes (BNB), Naive Bayes with informative Priors (PNB), Mixture of Naive Bayes (MNB), and Hierarchical Mixture of Naive Bayes (HMNB), and explore their pros and cons. Our results suggest that the PNB model offers the best tradeoff among the desirable properties. The second approach is motivated by the observation that the probabilistic modeling approach and the risk signal technique are closely connected, the PNB model can be naturally interpreted as a simple function capturing the rarity of permissions, by taking logarithm of the computed probabilities. We thus present a simple framework so that the PNB model and rarity-based signals are all instances in this framework. This also presents a simple interpretation of the PNB model that may be easier to understand. We then investigate two other instantiations of the framework and evaluate them. In summary, the contributions of this paper are as follows:

- We introduce the idea of risk score functions to improve risk communication for Android apps, and identify three desiderata for an effective risk scoring function.
- We examine several Bayesian approaches for risk scoring functions, and analyze whether they satisfy the desiderata.
- We introduce a framework that includes both the rarity-based risk signals and probabilistic models, and explore other ways to instantiate the framework.
- We conduct extensive evaluations using real-world data sets. Our experimental results demonstrate the effectiveness and relative strengths and weaknesses of different approaches.

II. CONCLUSIONS

We discuss the importance of effectively communicating the risk of an application to users, and propose several methods to rate this risk. We test these methods on large real-world data sets to understand each method’s ability to assign risk to applications. One effective method is the RSS method which has several advantages. It is monotonic, and can provide feedback as to why risk is high for a specific app and how a developer could reduce that risk. It performs well in identifying most current malware apps as high risk. This method allows for highly critical permissions and less-critical permissions to affect the overall score in an easy to understand way, making it more intuitive as well as difficult to evade when compared with other models.

III. REFERENCES


