Corso di Laurea Magistrale in Ingegneria Informatica
Corso di Laurea Magistrale in Ingegneria Telematica

Android's applications danger level evaluator

Canditati: Giovanni Pessiva, Tao Su
Relatori: Prof. Antonio Lioy, Ing. Andrea Atzeni

Dicembre 2012

Introduction

Android has become one of the most popular mobile operating systems. Every day, there are thousands of different applications uploaded on Google Play and other third party Android repositories. In this large amount of applications, a significant share of malicious code is also present, trying to exploit the vulnerabilities from which the mobile devices are not immune. The malware statistics in the last two years also demonstrate that the open source Android platform has been the biggest target for malware. Therefore, this thesis addresses the problem of distinguishing Android malicious applications from their legitimate counterparts.

The work presented in this thesis is a part of a wider project by Telecom Italia, whose main goal is the design of a partially automated Android applications analyser. Right now, the two software modules that we developed are in use at the Telecom Italia laboratory; soon the static analysis module is going to be released as open source.

Study of Android security

The thesis begins with the presentation of the architectures of the main mobile operating systems, with a particular attention to the security; these operative systems are Android, Windows Phone, iOS, BlackBerry OS, Symbian, Firefox OS.

Afterwards the focus will move to Android security, with a presentation of its vulnerabilities and exploits. Then the malicious behaviours of several different Android malware families are shown, together with the existing analysis tools (both static and dynamic) and the malware detection systems which are already available.

Telecom Italia analysis system

The system developed by the Telecom Italia researchers is able, using both static and dynamic analysis, to evaluate the dangerousness of the target *apk* file (Android package), expressing it as a numerical value: the "risk score". After an overview of the whole analysis system developed by Telecom Italia researchers, fuzzy logic is introduced, since it is the method applied in this thesis in order to calculate the risk scores, using as inputs the results of application analyses.

Static analysis

Static analysis checks its target application without executing it; the thesis introduces a new software module called "FileScan", capable of performing static analysis by scanning the *apk* and every file inside it. This module has been designed to complete the analysis of the code

and of the manifest file performed by the module "Behaviour", developed by Telecom Italia researchers.

All the work steps are detailed in the thesis, from the initial project to the final tests, explaining which problems have arisen during the development and how they have been solved.

FileScan uses magic number analysis to identify the potentially interesting files; then, it performs an analysis of those files (textual files, compressed archives, native executables, Android *dex* code files) in order to produce a "risk score" for the application.

FileScan results can be used in order to provide general information about the content of the *apk*, to point out suspicious behaviours (e.g., hidden files, shell scripts with potentially dangerous commands), to find URL addresses and phone numbers, and to identify malwares (e.g., by detecting known malwares and calculating the risk score of the application). The final results are particularly interesting for the URL addresses; compared tests with other already available analysis systems show that the approach used by FileScan (looking into the files) is indeed unique: the other tools only check the code. Also, the tests on the datasets of goodware and malware applications (respectively 3558 and 1361 files) gave good results: FileScan had a detection rate of 57.1% for the malware *apk* files, and 0.9% of false positives with the goodware *apk* files.

Dynamic analysis

Dynamic analysis is the analysis of the applications by executing them on an emulator, therefore it is a run-time analysis. The thesis provides a new module that performs dynamic analysis, which was developed on top of Droidbox, a well-known dynamic analysis tool for Android applications. Droidbox was modified in several ways. In order to provide increased automation, given the name of a directory as an input argument, the module will analyse all the *apk* files found in this directory automatically. Moreover, a dynamic risk module was added, which computes a risk score, based on the critical activities that Droidbox detects while the target application is running. The output of this module is a text file which stores all the activity logs, an array with the values of the intermediate risks that contributed to the risk score, and a data structure, which contains phone numbers, URLs and file names presented during the target running. Danger evaluation metrics and the testing results are exhibited, showing the performance of these analysis modules.

The program has been tested more than 3000 times during the development, the results of 1292 malware samples and 659 market samples are stored as to show how the system works. For the malware samples, the average risk score is 57.86, with 284 samples (21.98%) that have a risk score equalling 85. For the market samples, the average risk score is 46.6, and 42 samples (6.37%) have a risk score equalling 85.

The detailed risks array shows that the file manipulating operations have been performed the most number of times. The second observation is that most of the malware applications (94.35%) use system's native libraries, mainly in the broadcast receiver activity. The third observation is that use of *dex* class loader and the SMS related operations are only performed by the malware samples; these two types of activities are critical: if they are performed, the target sample have a high possibility to be a malware.

During the analysis of all these 1951 applications, 294 different URL addresses have been detected, with 96 belonging to the malware sample, and 198 belonging to the market samples. The most part of these addresses are legitimate websites which provide the advertise-

ment service for the Android platform; the rest are for malicious operations, like downloading additional applications. A total number of 5 different phone numbers have been detected, which are only shown in the malware samples, and are premium rate numbers. If a phone number is detected in one application, we are absolutely sure it is a malware.

Integration and conclusions

Finally, the integration tests are presented, pointing out that it is possible to combine these two modules together and make the separation between malware and goodware applications more clear and accurate.

To conclude the thesis, a list of possible enhancements to our work is presented, in both the static and the dynamic parts.