Forschungsinstitut Cyber Defence
Universität der Bundeswehr München

PATCH
Program Analysis, Testing, Comprehension, and Hardening

Prof. Dr. Johannes Kinder
FI CODE – PATCH
Universität der Bundeswehr München
Carl-Wery-Str. 22, 81739 München

+49 89 6004 7335
johannes.kinder@unibw.de
https://www.unibw.de/patch
December 22, 2020

Automatic Time Limits for Testing Android Apps

Type: Master's thesis
Starting date: immediately

Context

Android apps can contain hidden malicious behavior that executes in addition to the advertised behavior. A comprehensive forensic investigation to uncover this behavior must include both static and dynamic analysis. In dynamic analysis, the Android application is executed and monitored in a specially prepared environment. Our existing analysis platform, A³L (Automated Android Analysis Lab [1, 2]), focuses on observing events related to processes, strategies, network connections and the file system (see Figure 1).

Dynamic analysis necessarily has to set a time limit for the observation of an interactive application. If the time is too short, not all relevant behavior may be captured. If the time is too long, the analysis requires more resources and may become impractical to use. Currently, the analysis time is configured manually by the user of our platform.

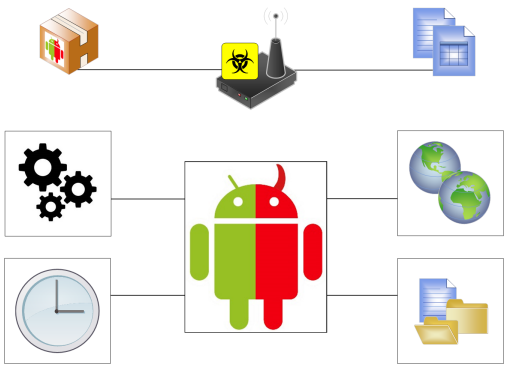

Figure 1: Principle of the A³L app analysis

Goal

The aim of this project is to develop a method and a system that automates the definition of an effective duration of the analysis of Android applications. One possible approach would be to automatically end the analysis at a suitable point if, for example, no new knowledge is being gained about the program and patterns start to repeat themselves. An alternative approach would be to use static features (such as code complexity metrics) to determine the required analysis time, possibly using machine learning. The A³L platform would thus be able to carry out analyses with appropriate resources, making the entire process more efficient while ensuring the results are meaningful in terms of observed functions and behavior.

Working Plan

1. Familiarize yourself with the principles of Android application analysis and the A³L platform.
2. Study state of the art in test automation for Android applications, focusing on analysis duration.
3. Develop a system for the automatic determination of suitable analysis duration.
4. Evaluate the developed method against static approaches on a significant number of applications.

References

[1] H. Winkler and C. Hummert. Android app dissection on the wandboard – automatic analysis of malicious mobile code. *Proceedings of the International Conference on Security and Management (SAM)*, pages 204–210, 2017.

[2] H. Winkler, C. Lenk, P. Engler, D. Pawlaszczyk, and C. Hummert. Android app dissection on the wandboard – evolving the platform into one-button solution. *Proceedings of the International Conference on Security and Management (SAM)*, pages 228–235, 2019.