1. Boosting Static Analysis of Android Apps through Code Instrumentation

Li Li (1)

Source: *2016 IEEE/ACM 38th International Conference on Software Engineering Companion (ICSE-C)*, p 819-22, 2016; **ISBN-13:** 978-1-4503-4205-6; **DOI:** 10.1145/2889160.2889258; **Conference:** 2016 IEEE/ACM 38th International Conference on Software Engineering Companion (ICSE-C), 14-22 May 2016, Austin, TX, USA; **Publisher:** IEEE, Piscataway, NJ, USA

Author affiliation: (1) University of Luxembourg, Interdisciplinary Centre for Security, Reliability and Trust, Luxembourg

Abstract: Static analysis has been applied to dissect Android apps for many years. The main advantage of using static analysis is its efficiency and entire code coverage characteristics. However, the community has not yet produced complete tools to perform in-depth static analysis, putting users at risk to malicious apps. Because of the diverse challenges caused by Android apps, it is hard for a single tool to efficiently address all of them. Thus, in this work, we propose to boost static analysis of Android apps through code instrumentation, in which the knotty code can be reduced or simplified into an equivalent but analyzable code. Consequently, existing static analyzers, without any modification, can be leveraged to perform extensive analysis, although originally they cannot. Previously, we have successfully applied instrumentation for two challenges of static analysis of Android apps: Inter-Component Communication (ICC) and Reflection. However, these two case studies are implemented separately and the implementation is not reusable, letting some functionality, that could be reused from one to another, be reinvented and thus lots of resources are wasted. To this end, in this work, we aim at providing a generic and non-invasive approach for existing static analyzers, enabling them to perform more broad analysis. (0 refs)

Inspec controlled terms: program diagnostics - smart phones
Uncontrolled terms: static analysis boosting - Android apps - code instrumentation - code coverage characteristics - malicious apps - intercomponent communication apps - reflection apps
Classification Code: C6150G Diagnostic, testing, debugging and evaluating systems - C6190V Mobile, ubiquitous and pervasive computing
IPC Code: G06F9/44 - G06F11/36 - H04M1/725
Treatment: Practical (PRA)
Database: Inspec
Data Provider: Engineering Village