Capita Selecta: Mobile Security Module 2015/2016

Project Assignment

- To register to the assignment send a mail by the 15th of April, 23.59 CET with Subject: "MOBILE SECURITY" to bruno.crispo@cs.kuleuen.be confirming your registration and specifying the app and the malware sample you have selected for the assignment (test the malware with Flowdroid before!).
- Oral discussions about the work you delivered, will be scheduled from 10th to 13th of May .

Description

Step 1: APPLICATION ANALYSIS

- Choose an app available on any Android market and one sample from the malware dataset you find at this link. https://www.dropbox.com/s/zkwkjgjp5da2dta/MALWARE_DATASET.rar?dl=0
- List all permissions they use
- List which app's components use these permissions
- Are the application and the sample overprivileged? List which permissions are declared and not used.
- Do not use any existing tool to perform Step 1.

Step 2: REVERSE ENGINEERING AND INSTRUMENTATION

- Choose an app from any Android market that uses an advertisement library (i.e. AdMod, a non exhaustive list of applications that use AdMod can be found at https://www.google.com/admob/success.html) and one app that sends information over internet (it can be the same app)
- Reverse engineering the application(s)
- For the app that used the advertisement library, remove the calls to it and repackage the app
- For the app sending information over internet, log in a file all its network accesses (saving invoked methods and arguments). Repackage the app
- Deliver the modified apk(s) that should work run normally on a phone.

Step3: STATIC ANALYSIS

- Choose a malware sample (can be the same of Step 1. The samples should be no bigger than 1-2 MB). The selected sample must have potentially dangerous sources/sinks paths
- Using Flowdroid identify potentially dangerous sinks and sources in the selected sample
- Report the output of Flowdroid and the list of sinks and sources found

Step 4: INSTRUMENTATION FOR DYNAMIC ANALYSIS

- Instrument the locations of the sources and sinks identified by Flowdroid in Step 3
- Instrumentation should register in a file the sinks and all the actual arguments that reach the sinks at runtime

Step 5: TESTING AND TRIGGERING

- To stimulate sinks and sources of the instrumented app of Step 4 use an automatic testing tool (i.e. DroidBot)
- This should be done using the emulator (running the malware sample on the phone is risky)
- Describe in the final report how you set up the testing tool and report which events have been generated

Submission

- Write a final report (in pdf) where you explain in detail what you have done for all the 5 steps and include the outputs produced by all the tools you used.
- Include all the apks used for the assignment, the initial ones and the ones modified. They all need to be fully functional and run on an emulator or a real phone. Include the also the program developed to solve Step 1
- Pack all the above (report + apks) in a zip file and send it by email to bruno.crispo@cs.kuleuen.be by 9th of May, 23.59 CET