**CS4110 - Software Testing and Reverse Engineering**

**Assignment # 3**

**Analyzing android apps (90 points)**

Study the following paper:

De Ruiter, Joeri, and Erik Poll. "Protocol State Fuzzing of TLS Implementations." In *USENIX Security Symposium*, pp. 193-206. 2015.

Study the MSc. thesis of Wesley van der Lee on learning state machines from android apps [uuid:8699be26-b226-4c55-bf0a-fd290455cd57](http://resolver.tudelft.nl/uuid:8699be26-b226-4c55-bf0a-fd290455cd57)

Setup the state machine learner for apps from [here](https://github.com/wesleyvanderlee/AppSecurity/tree/master/code/bsc-modify) and learn a state machine for [insecure banking app](https://github.com/wesleyvanderlee/AppSecurity/blob/master/code/bsc-modify/apk/insecurebankv2.apk). When all works as it is supposed to, you will obtain a state machine like Figure 1.

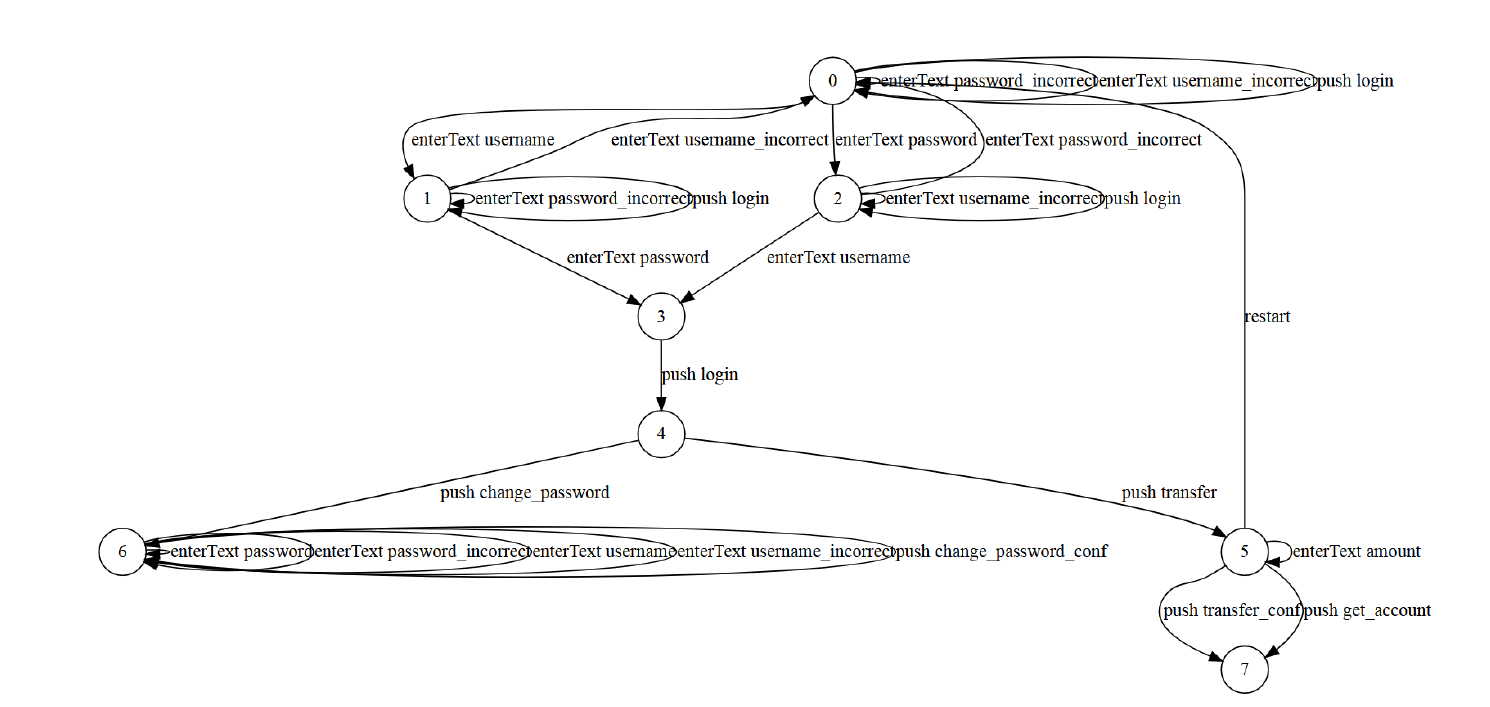


Figure : State machine of Insecure Banking App

States 0-2 deal with pre-login phase where the user can enter username and password. State 3 performs the login step. State 4 and 5 are post-login where state 5 allows for funds transfer. State 6 is the change password state.

Your task is to download the apk of an android app of your choice from any Appstore (for hints on where to find interesting apps, see below) and learn a state machine of that app. Following the guest lecture from Riscure, reverse engineer the app and introduce a bug in it by changing a function call, parameter value, or adding/deleting lines to/from the code. Additional instructions on reverse engineering Android apps are listed below. Build a state machine of the buggy app and analyze the differences in the two state machines. Are you able to discover where the bug is in the learnt state machine?

**Hint:** Following are some resources where you can find malicious apk files or get ideas on where to find them:

* <https://github.com/ashishb/android-malware>
* <https://www.extremetech.com/mobile/263488-new-android-malware-mines-cryptocurrency-phone>
* <https://0xc1r3ng.wordpress.com/2016/06/02/sample-source-andorid-malware-apk/>
* Or you can hunt for shady apps on Google Playstore.

**Deliverable:** Make a video of at most 10 minutes where you analyze the difference in behavior between the two learnt state machines for the app of your choice. Major part of the video should be spent analyzing the introduced bug in the app. Optionally, you can add additional installation and set up instructions in a readme file accompanied with the video. The steps you show in your video must be reproducible.

Aim to answer the following questions (but not limited to these):

* Which application did you select and why?
* Describe any issues you faced while using the tool.
* Which bug did you introduce and why? Describe the problems encountered when introducing it.
* Relate your analysis to the security and vulnerability analysis performed in the above paper and thesis.

You will be asked to indicate whether you allow us to share your video with next year’s students. Please do not include any privacy sensitive information in the video, such as student numbers.

**Grading criteria:**

* Correctness
* The depth of the analysis
* The meaning of the obtained results
* Reproducibility of the study
* Presentation style (clarity, speed, visualization)

**Additional resources:**

* (Revere engineering apks) <https://youtu.be/rgV98YMioCA>
* (Revere engineering apks) <https://youtu.be/7SRfk321I5o>