ABSTRACT:
Static analysis of Android applications can be hindered by the presence of the popular dynamic code update techniques: dynamic class loading and reflection. Recent Android malware samples do actually use these mechanisms to conceal their malicious behavior from static analyzers. These techniques defuse even the most recent static analyzers that usually operate under the "closed world" assumption (the targets of reflective calls can be resolved at analysis time; only classes reachable from the class path at analysis time are used at runtime). In this work we proposed the solution that allows existing static analyzers to remove this assumption. This is achieved by combining static and dynamic analysis of applications in order to reveal the hidden/updated behavior and extend static analysis results with this information. In this presentation we will describe design, implementation and preliminary evaluation results of our solution called StaDynA.

BIO:
http://www.zhauniarovich.com/bio.html