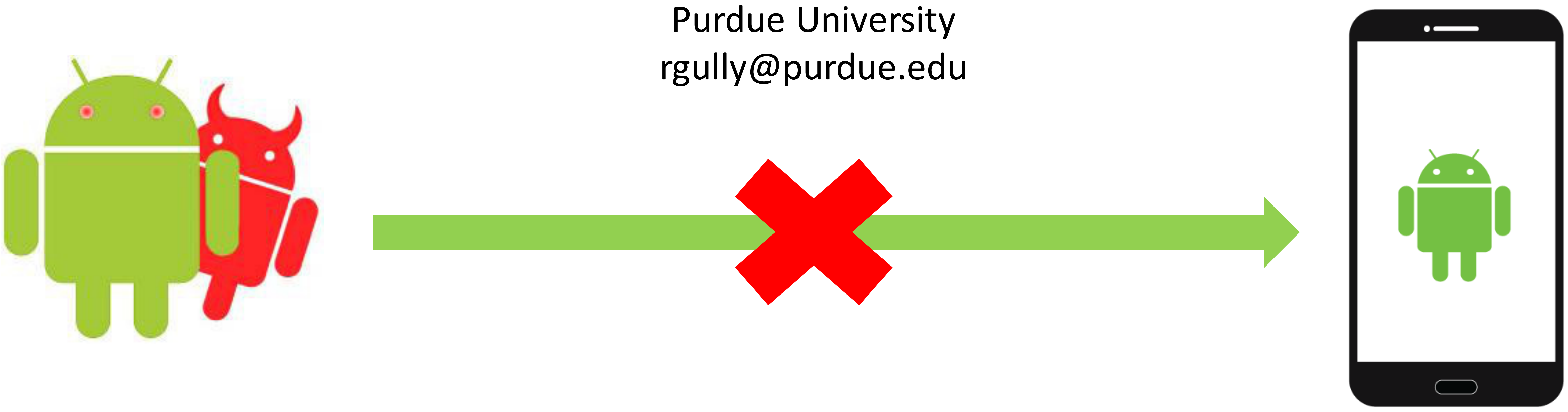
CERIAS

The Center for Education and Research in Information Assurance and Security

Detecting Android Malware Before Reaching the User (WIP)

Rachel B. Gully

Purdue University
rgully@purdue.edu



Abstract

The rise in Android’s popularity amongst users has seemingly made it a great target for malware. Currently, many analysis tools that are supposed to detect malware before reaching a user fail to do so (Kumar, 2014). This research will investigate the efficacy of different methods of dynamic and static analysis on Android malware to determine if malware can accurately be detected before reaching a user.

Significance

With the Android OS being so popular, it would make sense that roughly 1 out of 5 Android applications are malware (Whitwam 2015). Google’s attempt to combat malware entering their app store was called Bouncer, which was a dynamic analysis tool that vetted an application prior to being uploaded to the Google Play store. Unfortunately, Bouncer could easily be fooled by common evasion techniques (Kumar 2014). The problem is, current antimalware tools, dynamic and static analysis tools, are able to be fooled by malware, leaving Google’s large user base vulnerable to having their personal data (banking information, GPS locations, contacts, etc.) and privacy violated by the malware hiding in the wild.

References

Kumar, M. (2014). *Dynamic Analysis tools for Android Fail to Detect Malware with Heuristic Evasion Techniques.* Retrieved from http://thehackernews.com/2014/05/dynamicanalysistoolsforandroidfail.html

Whitwam, R. (2015). *Android antivirus apps are useless here’s what to do instead.*Retreived from http://www.extremetech.com/computing/104827-androidantivirusappsareuselesshere swhattodoinstead

Methodology

For this research, six automated Android analysis tools will be evaluated against four criteria:

- Analysis type
- Features Analyzed
- Supported Android Version
- Ability to detect a malware sample

The six Android malware family samples that will be used to evaluate the tools were obtained from the AndroMalShare database:

- DroidKungFu3
- Geinimi
- CounterClank
- Tapsnake
- HippoSMS
- Plankton

Analysis Tool	Analysis Method(s)
Droidbox	• Static pre-check • Taint Analysis • API Monitoring
Drozer	• IPC Analysis
CopperDroid	• System call tracking • Behavior reconstruction
CuckooDroid	• Dalvik API Hooking • Emulator detection prevention • Androgaurd integration for static analysis.
SandDroid	• Taint Analysis • Static Analysis
Tracedroid	• Behavior analysis • Static Analysis