MILCOM 2015
LEVERAGING TECHNOLOGY—THE JOINT IMPERATIVE
OCTOBER 26–28, 2015 · TAMPA, FL

Tutorial: Practical Program Analysis for Discovering Android Malware

Module 3: **Malware in the Wild**

Suresh Kothari – kothari@iastate.edu
Benjamin Holland – bholland@iastate.edu

Acknowledgment: co-workers and students
DARPA contracts FA8750- 12-2-0126 & FA8750-15-2-0080

ensoft
conquer complexity

IOWA STATE UNIVERSITY

Agenda

- Discuss techniques and tools for detecting malware in the wild
- Lab 3 (audit application binary)
- Lab 4 (scaling up operations)

Discussion of Technical Approaches

- Signature Based Detection
- Machine Learning
- Dynamic Testing and Sandboxing
- Changes to the Android Platform
- Two-Pass Analysis Systems

Agenda

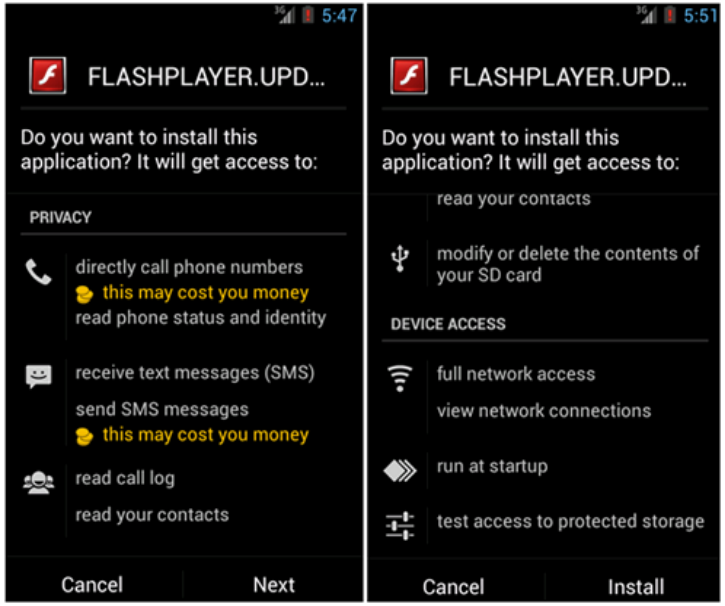
- Discuss techniques and tools for detecting malware in the wild
- Lab 3 (audit application binary)
- Lab 4 (scaling up operations)

Lab 3: FlashBang Malware Audit

- Representative of something in the wild…
  - Should be a “new” sample
  - Application disguised as a Flash player update
- What functionalities does this application have?
  - Focus on program comprehension
- What malware in the wild does this malware descend from?

Original Sample: Stels Malware

- App Description: Disguised as a Flash player update after clicking a link in a phishing email.
- Malware: Multi-purpose Android trojan horse that can harvest a victim's contact list, send and intercept SMS (text) messages, make phone calls (including calls to premium numbers), and install additional malware packages.
- Reference: http://www.secureworks.com/cyber-threat-intelligence/threats/stels-android-trojan-malware-analysis/
- FlashBang is a modified recompilation of Stels



Agenda

- Discuss techniques and tools for detecting malware in the wild
- Lab 3 (audit application binary)
- Lab 4 (scaling up operations)

Acquiring Samples to Audit

- APK Extraction
  - Most reliable, works for paid apps.
  - Download application to device like normal, extract APK from device through Android Debug Bridge or an app designed to extract installed applications
  - APK Extractor Application
- App Store Crawlers
  - May result in banned accounts, IPs, etc.
  - https://github.com/Akdeniz/google-play-crawler
    - (requires 1 disposable Google Account)
  - https://apps.evozi.com/apk-downloader/
    - (third party service, limited to 400 requests/day, AdBlocker recommended!)
  - https://github.com/nviennot/playdrone
    - (crawls and downloads Google Play Stores apps in mass, requires ~500 disposable Google Accounts)
    - https://github.com/questionablecode/account-harvester

Lab 4: Scaling up Operations

- Exercise: Use the Starter-Toolbox Headless plugin to run the analysis program developed earlier over a set of APKs.
  - What APKs are good candidates for a follow up analysis?