AIRMAIL: Scaling Mobile Vulnerabilities through the AI Supply Chain

Johns Hopkins
APPLIED PHYSICS LABORATORY

Bradley Potteiger and Rachel Cohen

Johns Hopkins Applied Physics Laboratory

Introduction

- Artificial Intelligence and Machine Learning libraries are being increasingly incorporated within mobile applications.
- Supply chain distribution channel and homogeneous software structure means that one software bug can lead to an exploit scaling to millions of devices around the world. **Think SolarWinds for Mobile.**
- AI dependencies for mobile applications provide a new attack vector beyond traditional adversary machine learning approaches to covertly obtain and maintain a foothold into adversary systems.
- Developing an autonomous reverse engineering and exploitation framework will allow designers to more rapidly identify vulnerabilities in critical applications.

Architecture

Package Extractor

- Extracts application metadata and source code
- Static analysis to identify symbols, functions, variables, and file names corresponding to ML libraries

Vulnerability Analysis

- Ghidra based reverse engineering framework for identifying bugs and vulnerabilities within code

Correlation Engine

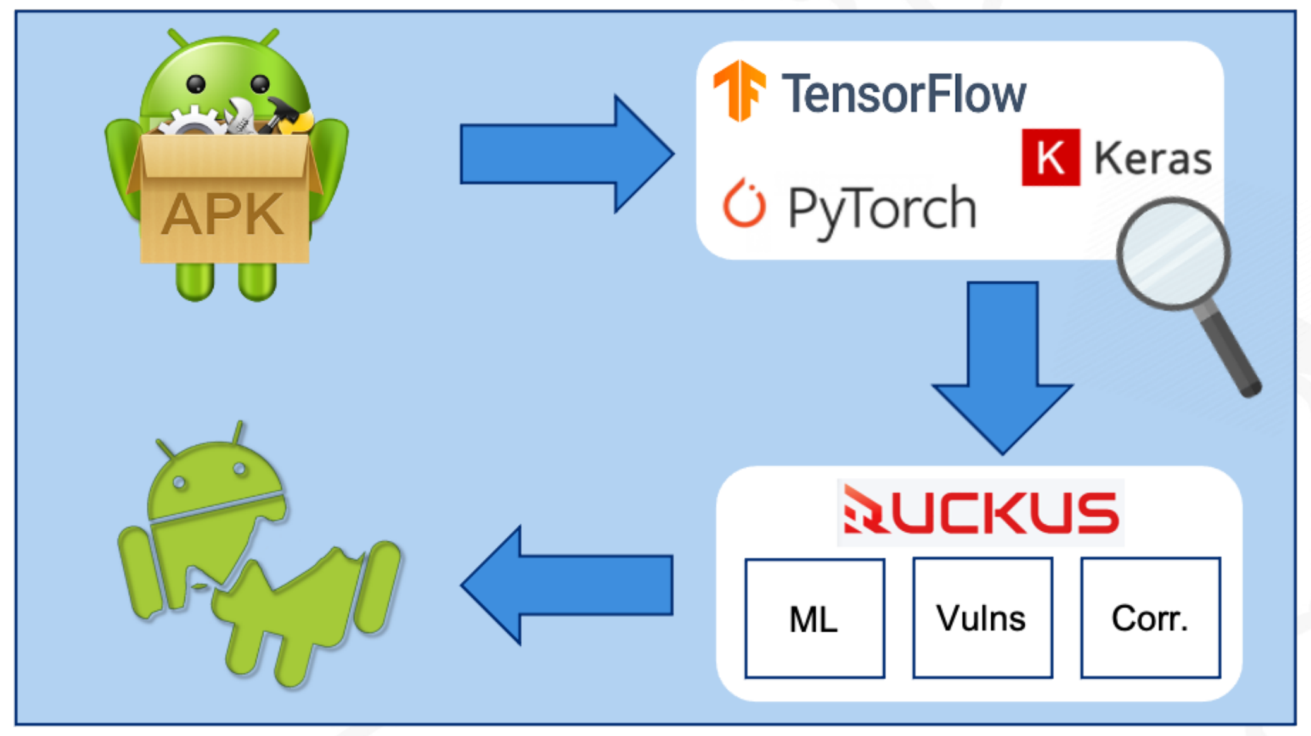
- Graph based correlation of similar dependencies, and vulnerabilities from applications

Backend Database

- Hybrid graph and relational database for storing high level and low level relationships

Exploitation Engine

- Operationalizes vulnerabilities into proof of concept exploits
- Provides scalability score to demonstrate potential impact and reach of attacks.



Case Study

- Analyzed 10 Android applications from app store
- Conducted quantitative analysis to identify most popular ML dependencies, shared libraries, and functions
- Executed vulnerability discovery and correlation analysis on the relationship and similarity between the ML based Android applications
- Extrapolated vulnerabilities to assess scale of several potential attacks.

Results

Top Library

- TensorFlow Lite

ML Utilization

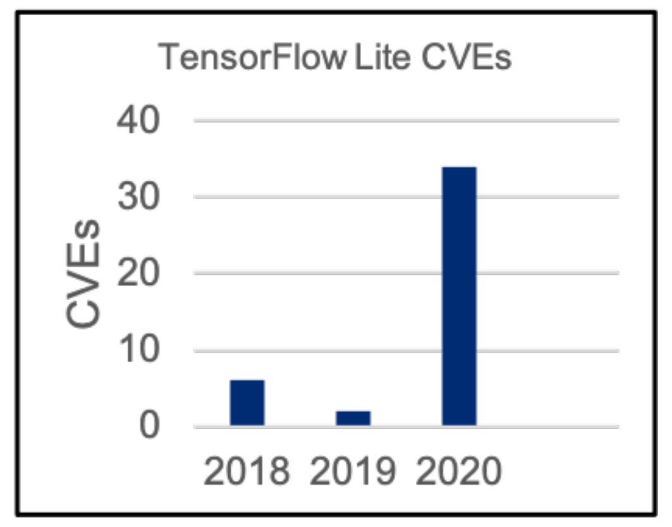
- 30% of apps
- Expected to grow to 65%

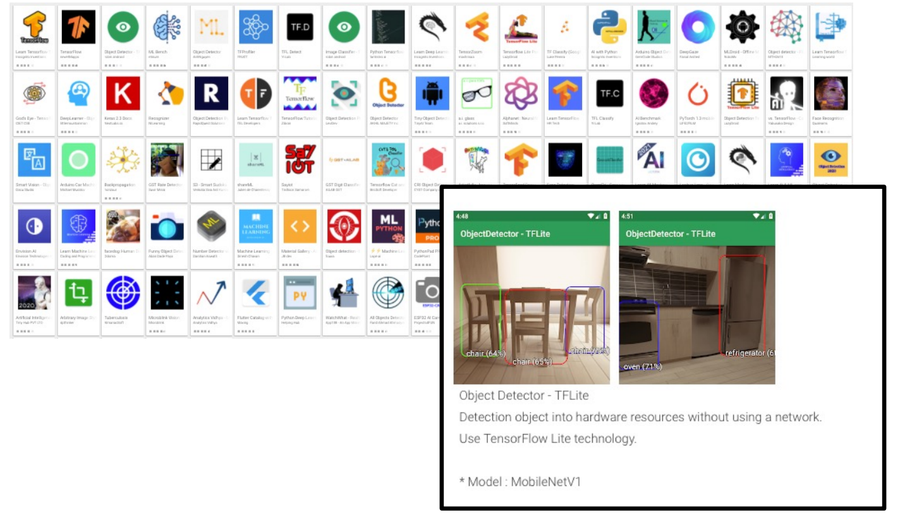
Vulnerability Identification

- 5 known and 3 unknown

Scaling Factor

- Apps correlated on average 62%
- Vulnerability in dependency applied to 7 out of 10 apps




ObjectDetector - TFLite

Object Detector - TFLite

Detection object into hardware resources without using a network.

Use TensorFlow Lite technology.

* Model : MobileNetV1

Conclusion & Future Work

- Demonstrated the reliance of modern mobile applications on ML libraries and dependencies
- Demonstrated impact of supply chain vulnerabilities on scaling exploits
- Developed a proof of concept autonomous reverse engineering and exploitation framework
- Future Work
  - Obtain application download metrics to assess real world scalability of exploits
  - Expand evaluation to larger subset of applications
  - Develop mitigations to counter approach

HOTSOS 2021
8TH ANNUAL
HOT TOPICS in the SCIENCE OF SECURITY
APRIL 13-15, 2021 | VIRTUAL