School of
Engineering and Computing Sciences

Permission-based Malware Detection in Android Devices

REU fellow: Nadeen Saleh[1], Faculty mentor: Dr. Wenjia Li[2]

Affiliation: 1. Florida Atlantic University, 2. School of Engineering and Computing Sciences, NYIT

NYIT Research Experience for Undergraduates (REU)

May 26 – July 30, 2015

Abstract

Android, the most popular mobile operating system of our day, endures the largest deal of exploitation efforts by malware authors. While models with large overhead have been proposed to secure oneself from the harms of malware, a simple and fast do-it-yourself method has yet to be widely distributed. This project proposes a method of malware detection that will add a lightweight layer of protection to any Android device.

Methodology

- We conducted a manual detection for outliers in our training set samples - 50 benign and 50 malicious applications.
- We weight three features from the data;

  (1) total permissions per application,

  (2) the occurrence of duplicate permissions in a single manifest,

  (3) permissions that are highly prevalent in our malware training set (e.g. SEND SMS malware training set 46% presence, benign training sample 6% presence).
- To obtain the best results, our classifier must be fed meaningful information that is justified by their distinguishable nature.
- In choosing, distributing and weighting our features it was important to note permissions with a significant or sole presence in our malware training set, concluding to consider those over a 100% percent-increase in Figure B.
- In optimizing the point system, the threshold value of 5 resulted in the highest accuracy.

References

[1] G. Kelly, “Report: 97% of mobile malware is on android. this is the easy way you stay safe,” 2014.
[2] O. Hou, “A look at google bouncer,” 2013.

Acknowledgement

The project is funded by National Science Foundation Grant number CNS-1263283 and New York Institute of Technology.

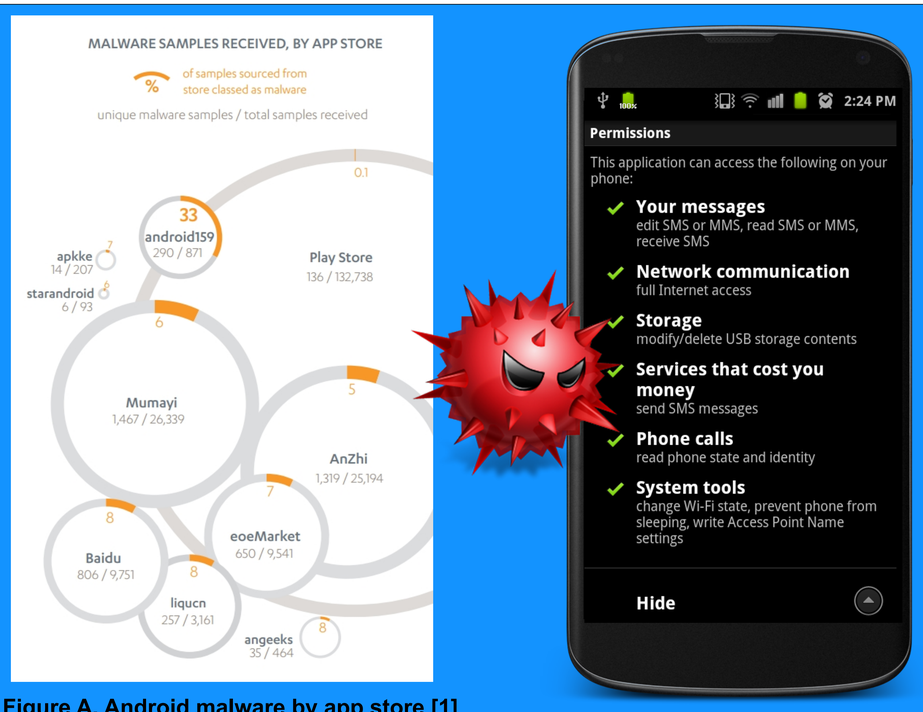

Figure A. Android malware by app store [1]

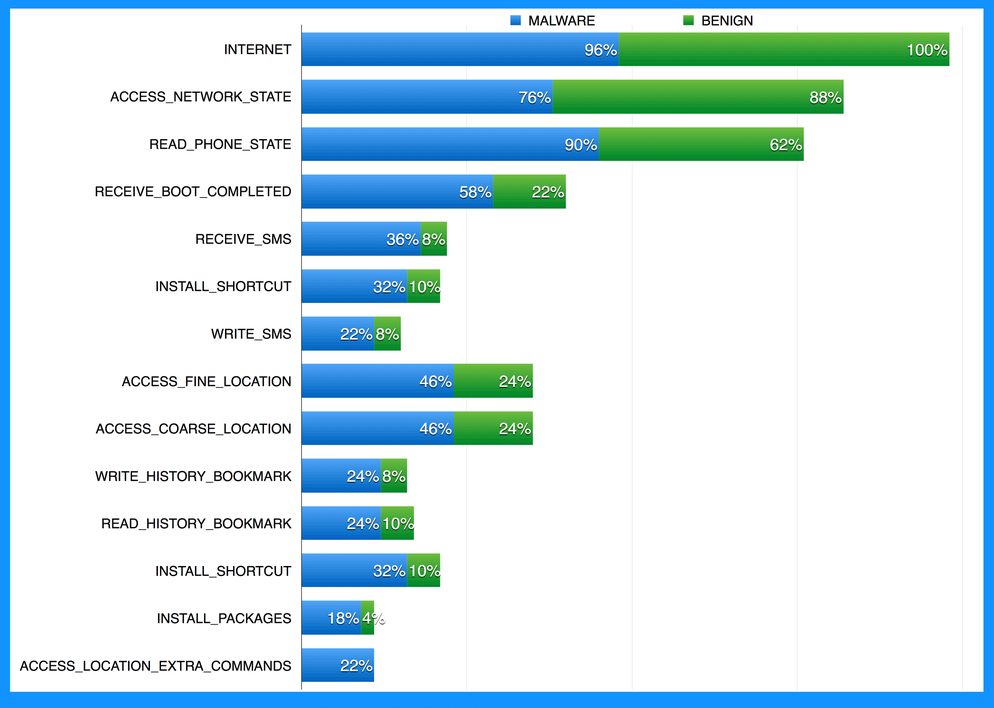

Figure B. Training set comparison

Experimental results

Weight	Feature
1	Total permissions per application (>= 11)
1	Occurrence of duplicate permissions
3	Permissions: Category 1
3	Permissions: Category 2
2	Permissions: Category 3
1	Permissions: Category 4
5 (static)	Permissions: Category 5

Table 1: Weighted point system

	Accuracy	False Neg	False Pos
SVM	79%	24.56%	16.28%
Point-system	78.9%	17.39%	25%

Table 2: Results

Discussion

- To estimate the performance of a predictive model (SVM) and validate our feature selection, we collected preliminary results from the whole-number point system. Our algorithm was run with different weights and category distributions, with a final and finest accuracy of 78.9%.
- To cross-validate our method, we use the Support Vector Machine machine learning algorithm on our training set. This supervised learning method accurately identified 79% of instances.
- The advantage of our model is its convenience. Our method makes use of 3 features and 13 permissions to amount to a conclusion. Any Android user with access to an apktool and basic knowledge of the command line can decompile their apk file (distribution is very accessible, in contrast to iOS applications), locate the manifest file, and determine for themselves if their application has malicious intent.

Future Work

In regards to feature selection, identifying more distinguishable features in a small training set (such as this one) can be challenging. Future works have the ability to test the performance of different parameters, more importantly, a wider and relevant feature selection. For comparison purposes, we tested the performance of fewer parameters to establish a correlation with accuracy, and as anticipated, saw a qualified decrease in accuracy.