Assignment 1

Eynav Ben Shlomo 209328970, Elon Ezra 313534133

November 20, 2022

Abstract

As part of the course "Attacks on the Android operating system" We will investigate the above system

1 Introduction

As part of the course "Attacks on the Android operating system"

2 Yes, Machine Learning Can Be More Secure! A Case Study on Android Malware Detection

To cope with the increasing variability and sophistication of modern attacks, machine learning has been widely adopted as a statistically-sound tool for malware detection. but machine learning itself can be the weakest link in a security system. This work is the proposal of a simple and scalable secure-learning paradigm that mitigates the impact of evasion attacks, while only slightly worsening the detection rate in the absence of attack by improve the security of Drebin against stealthier attacks. website article

2.1 Improving security

Machine learning can be used to improve system security, if one follows an adversary-aware approach that proactively anticipates the attacker. To this end, first exploited a general framework for assessing the security of learningbased malware detectors, by modeling attackers with different goals, knowledge of the system, and capabilities of manipulating the data. then considered a specific case study involving Drebin, an Android malware detection tool, and shown that the performance of Drebin can be significantly downgraded in the presence of skilled attackers that can carefully manipulate malware samples to evade classifier detection.

2.2 ANDROID MALWARE DETECTION

Drebin conducts multiple steps and can be executed directly on the mobile device, as it performs a lightweight static analysis of Android applications. Although Drebin has shown to be capable of detecting malware with high accuracy, it exhibits intrinsic vulnerabilities that might be exploited by an attacker to evade detection. Drebin only analyzes the Android manifest and classes.

2.2.1 Android Manifest

The manifest file holds information about the application structure. Such structure is organized in application components, The actions of each component are further specified through filtered intents; The manifest also contains the list of hardware components and permissions requested by the application to work.

2.2.2 Dalvik Bytecode (dexcode)

The classes.dex file contains the compiled source code of an application. It contains all the user-implemented methods and classes. Classes.dex might contain specific API calls that can access sensitive resources such as personal contacts . Moreover, it contains all system-related, restricted API calls whose functionality require permissions. this file can contain references to network addresses that might be contacted by the application.

2.3 The main contribution

This work is proposal of an adversary-aware machine-learning detector against evasion attacks , inspired from the proactive design approach advocated in the area of adversarial machine learning . Empirically evaluate our method on real-world data, including an adversarial security evaluation based on the simulation of the proposed evasion attacks. Show that method outperforms state-of-the-art classification algorithms, including secure ones, without losing significant accuracy in the absence of well-crafted attacks, and can even guarantee some degree of robustness against DexGuard-based obfuscations. define a novel, theoretically-sound learning algorithm to train linear classifiers with more evenly-distributed feature weights. This approach allows one to improve system security, without significantly affecting computational efficiency.

2.4 type of classifier

The type of classifier that analyze is static because Drebin performs a static analysis and this work on improving Drebin's security using a machine learning algorithm.