Michael Harkess
9/10/2022
Topic: Static Android Application Analyzer

Problem Statement

Millions of people have Android devices and therefore are utilizing apps from the Google Play store and other third-party stores. As such, it is important to prevent malicious applications from being accessible in the app store. Having a program that can detect malicious code in apps would help mitigate the risk of a mobile app user being subjected to a potential risk in the security of their device.

This program would analyze code in a finished mobile application (after the mobile app compiles) and check if there are potential security vulnerabilities within the code itself. These vulnerabilities can take on a multitude of forms, so a machine learning model would be an ideal solution to auto-identify potential vulnerabilities by providing some example applications for the ML model to train on.

Usefulness

This topic is useful because it can help make the process of vetting mobile applications easier and keep the users of those apps safer. This form of mobile application vetting lets companies who run application stores automatically check if a mobile app is malicious, saving them time (from checking manually) and cost (hiring a third party to vet applications). If the application is developed to encompass common threats in compiled mobile apps, more time can be used to develop techniques to prevent other forms of malicious exploits in apps on the app store.

The main application for this project is malware detection on android apps. This detection would ideally be done by the Google Play Store, to vet apps being accessed through their storefront (similar to how Play Protect works in the Play Store). Consumers can keep the same

level (or greater level) of trust towards the app store as apps are being vetted for maliciousness. Along with App stores using the code analyzer to distinguish between malicious and benign apps being uploaded to their platform, companies that have many teams of developers to produce apps can also use the analyzer to make sure that their final product does not have a static security issue (or can be patched if a potential vulnerability is found).

There have been some studies that have shown that this code analyzer concept is possible (those studies are linked below) and there are some real-life applications of programs checking for vulnerabilities automatically (ex. Play Protect) from downloaded apps. So, while the concept does exist to some degree, I hope to further improve vulnerability detection in this area.

Papers to Look Further into:

- Wang, Wei, et al. "Detecting Android Malicious Apps and Categorizing Benign Apps with Ensemble of Classifiers." *Future Generation Computer Systems* 78 (2018): 987-94. Web.
    - https://www.sciencedirect.com/science/article/abs/pii/S0167739X17300742
- Singh, Pooja, Pankaj Tiwari, and Santosh Singh. "Analysis of Malicious Behavior of Android Apps." *Procedia Computer Science* 79 (2016): 215-20. Web
    - https://www.sciencedirect.com/science/article/pii/S1877050916001599
- P., Vinod, Akka Zemmari, and Mauro Conti. "A Machine Learning Based Approach to Detect Malicious Android Apps using Discriminant System Calls." *Future Generation Computer Systems* 94 (2019): 333-50. Web.
    - https://www.sciencedirect.com/science/article/abs/pii/S0167739X18306216
- W. Yang, X. Xiao, B. Andow, S. Li, T. Xie and W. Enck, "AppContext: Differentiating Malicious and Benign Mobile App Behaviors Using Context," 2015 IEEE/ACM 37th IEEE International Conference on Software Engineering, 2015, pp. 303-313, doi: 10.1109/ICSE.2015.50.
    - https://ieeexplore.ieee.org/abstract/document/7194583
- K. Shibija and R. V. Joseph, "A Machine Learning Approach to the Detection and Analysis of Android Malicious Apps," 2018 International Conference on Computer Communication and Informatics (ICCCI), 2018, pp. 1-4, doi: 10.1109/ICCCI.2018.8441472.
    - https://ieeexplore.ieee.org/abstract/document/8441472
- W. Wang, Z. Gao, M. Zhao, Y. Li, J. Liu and X. Zhang, "DroidEnsemble: Detecting Android Malicious Applications With Ensemble of String and Structural Static Features," in IEEE Access, vol. 6, pp. 31798-31807, 2018, doi: 10.1109/ACCESS.2018.2835654.
    - https://ieeexplore.ieee.org/abstract/document/8357771