Mini Review over “MalScan: Fast market-wide mobile malware scanning by social-network centrality analysis”

Author: Zhijie Liu, Student ID: 2022233365

1. What is the research problem, and what is the significance of the research?

Android devices and Android application grow explosively in several Android markets, so as to Android malware. Millions of apps have been installed by end users all around the world from various app markets (Google Play, AppChina and so on). Due to the limitation of current scanning process, more and more malware has appeared in these markets. Thus, stopping the spread of malware mainly depends on new automatically scanning process (new lightweight Android Malware Detection) for these app markets. Therefore, the new scanning method must be able to adapt the explosive growth of Android application and provide high enough detection performance for avoiding or alleviating the first escaping of Android malware.

2. What is state-of-the-art research status of the research problem?

DL-based detection has gradually become the most effective detection methods due to that there is a large amount of data available, but its high time consumption causes it difficult to be deployed on for Android markets scanning. Although ML-based detection methods have less detection performance than DL-based methods, effective behavior modeling is able to alleviate the performance gap between ML-based methods and DL-based methods. Existing app malware detection methods extract static program features for behavior modeling, including android components, stings, permissions, APIs, graphs and so on. Where graph-based methods are considered as the most effective since graphs contain program semantics. Moreover, further extracting the attributes in graphs for modeling app behaviors is efficient and effective for next step of malware detection.

3. Describe the methodology of the paper, and describe the advantage of the proposed method over state-of-the-art.

To address the mentioned above problem, this paper proposes a new method, based on call graph, called MalSan. MalScan leverages app call graph. Each node in call graph is a method and each directed edge represents call relationship. MalScan utilizes five kinds of centralities of sensitive API methods for constructing app feature vector respectively, representing app behaviors. Then use machine learning model techniques for training detection or classification model, for example, 1-Nearest Neighbor, 3-Nearest Neighbor, Random Forest, and so on. The basic idea of MalScan is impressive, since the process of extracting features captures both local and global semantic information.

The result shows that MalScan outperforms than other graph-based methods. Facing apps in different time periods (training and testing use the same period apps), MalScan achieves every accuracy higher than 95% with all centralities. In the scenario of Android app evolution and adversarial attack, MalScan performs good robustness against these two cases. Moreover, MalScan is about 200 times faster than the SOTA method, which makes it able to complete the task of fast scanning app markets.

4. What is the conclusion? On what way can one can possibly improve the performance of the method.

MalScan is a lightweight Android malware detection method based on call graph, extracting centralities of sensitive API methods. It achieves high detection performance and robustness. Moreover, its runtime overhead is quite low than other SOTA methods. Due to extracting information from call graph, MalScan is liable to suffer from structural attack, which modifying the call relationship between methods. One way to alleviate this kind of attack is to append some adversarial samples into training set for retraining detection model. Another method is enhancing API representation, combining function related API

methods together to generalize semantic information. These two methods are able to alleviate the escaping of adversarial samples generated by structural attack, but still cannot counter the attack.

5. What is the inspiration of the paper to your own research, like on writing, on theory development, on experimental design, or on research idea etc.?

MalScan is an impressive method, simple but effective. Comparing it with other graph-based methods, one finding is that all methods do not consider the level of graph abstracting. These methods often lose local or global information when extracting features from call graph, which makes adversarial attack able to conquer them. Thus, to fight against adversarial attack, especially structural attack, we must consider more on features extracted from call graph. Moreover, the experiments in this paper are sufficient. Especially, they consider time dimension for Android evolution, an important view point. Eight scenarios are presented in this paper, which gives a quite clear result of MalScan facing Android evolution.

For each question, no less than 100 words is preferred.

Words Count: 748