Analysis of Malware Detection Techniques

Abstract

Today’s digital world is pervaded with malware. In response to this reality, there are copious studies being conducted around the world on how best to improve the detection of malware, as malware becomes more sophisticated with every passing year. In the following report, we will discuss some current studies of interest on malware detection, and propose some of our own suppositions on how these suggested techniques can be improved upon.

Malware is evolving at a rapid pace. The techniques used to analyze and detect malware must evolve as well, as the evasion techniques in newer malware are making them increasingly difficult to detect. In the survey by Robert Grimes [5], the author discusses dynamic analysis techniques, both manual and automated, to detect malware and to study how each evasion operates. Grimes further classifies the known types of evasion techniques that malware vendors use and their efficacy against different types of analysis and detection approaches.

Static detection: A static detection metric is one which does not require the suspicious program to be executed. One of the most popular forms of static detection is signature matching. This method relies on a database of known malware fingerprints. When a file is downloaded, its fingerprint is checked against the database. Most commercial antivirus programs employ this technique due to its low false negative (falsey identified as illegitimate) rate [8]. However, as the method relies on a database of known malware, it is not useful in detecting novel malware.

Dynamic detection: The bulk of malware detection in this current environment relies on dynamic detection, which is analysis of the behavior of the program or process at runtime. There are two main categories of Dynamic detection: manual and automated.

In this paper, we will discuss some of the techniques currently being employed by malicious programs to evade detection.

Methods

The project addresses the efficacy of the two-stage system proposed as proposed in “Malware Intrusion Detection For System Security,” by Katkar, Shukla, Shaikh and Dange [1] from two directions.

The first is from a theoretical perspective. To this end, we are researching evasion tactics that are currently being employed by malicious programs to see if there are any well-known tactics that can evade this detection strategy.

The second is from an experimental approach in which we replicate the experiment performed by Katkar, Shukla, Shaikh and Dange in order to critically assess the performance of the classification algorithm. To this end, we will determine the accuracy (specifically with respect to data that the model has not yet seen) in addition to False Positive/Negative ratios.

Potential improvements to the baseline model include:
- Experimenting with alternative classification models.
- Outlier detection and deletion.
- Hyperparameter tuning.

Conclusions

We have replicated the experiment shown in Malware Intrusion Detection For System Security [1] on a larger dataset of 200k samples, achieving a similar score of 98.9% accuracy. FalsePositive: 1.4%, FalseNegative: 0.9%

MTA-KDD’19: A Dataset for Malware Traffic Detection suggests a Multi-Layer Perceptron classifier performs better than Random Forest after cleaning the dataset. We were not able to replicate this finding on our dataset.

To conclude, simple classification algorithms are effective for detecting malware. But we cannot ignore the invasive tactics employed by malicious programmers. One clear shortfall of the system proposed in [1] is the absence of continuous monitoring. This is necessary in order counter a hard-coded delayed activation of malicious behaviour.

If a malicious program can detect the presence of a resource monitor, then it may still evade detection by hiding malicious activity until the detection system finishes collecting the dynamic behavioural data required for classification. Unless, of course, the static attributes of the program are sufficient to label it as malicious.

References