Machine learning methods for detection and classification of malware

iqrar khan Reg No: 17jzbcs0018

30th june 2021

1 Abstract

Malware detection is an important factor in the security of the computer systems. However, currently utilized signature-based methods cannot provide accurate detection of zero-day attacks and polymorphic viruses. That is why the need for machine learning-based detection arises.

The purpose of this work is to determine the best feature extraction, feature representation, and classification methods that result in the best accuracy when used on the top of Cuckoo Sandbox. Specifically, k-Nearest-Neighbors, Decision Trees, Support Vector Machines, Naive Bayes and Random Forest classifiers were evaluated.

This work presents recommended methods for machine learning based malware classification and detection, as well as the guidelines for its implementation. Moreover, the study performed can be useful as a base for further research in the field of malware analysis with machine learning methods.

2 Introduction

With the rapid development of the Internet, malware became one of the major cyber threats nowadays. Any software performing malicious actions, including information stealing, espionage, etc. can be referred to as malware. Kaspersky Labs define malware as "a type of computer program designed to infect a legitimate user's computer and inflict harm on it in multiple ways."

While the diversity of malware is increasing, anti-virus scanners cannot fulfill the needs of protection, resulting in millions of hosts being attacked.

Therefore, malware protection of computer systems is one of the most important cyber security tasks for single users and businesses, since even a single attack can result in compromised data and sufficient losses. Massive losses and frequent attacks dictate the need for accurate and timely detection methods. Current static and dynamic methods do not provide efficient detection, especially when dealing with zero-day attacks.

For this reason, machine learning-based techniques can be used. This project discuss the main points and concerns of machine learning-based malware detection, as well as looks for the best feature representation and classification methods.

The goal of this project is to develop the proof of concept for the machine learning based malware classification.

3 Motivation

The increasing number of personal computers [4] and malware [5] poses an imminent threat to the casual person behind his desktop. Modern antivirus software vendors struggle to convince people that they need their product. This is partially due to the software updates that most of them have frequently so that they keep up to date with the most advanced malware. Another reason is that people do not realize they need it until an unfortunate event happens.

In 2014, Edward Snowden, a former Central Intelligence Agency employee, leaked documents proving that National Security Agency intended to spy on millions of its citizens using malware [6]. Since then antivirus software companies have realized they need to protect their customers not only from private malware producers but from the governments as well.

All this proves the necessity for better malware detection software around the world. This project is about demonstrating the main techniques modern antivirus software companies use in the discovery of malware and exploring the possibility of creating a malware detection cycle that could improve existing techniques.

4 Methodology

Various Machine learning techniques are used for malware classification such as Support Vector Machine, Decision Tree, Naive Bayes, Random Forest, etc., and machine learning clustering techniques are used for clustering malware samples. For this project we will use machine learning approaches for malware analysis, detection and classification. Static Analysis: Static analysis can be viewed as "reading" the source code of the malware and trying to infer the behavioral properties of the file. Static analysis can include various techniques.

1. File Format Inspection: file metadata can provide useful information. For example, Windows PE (portable executable) files can provide much information on compile time, imported and exported functions, etc.

2. String Extraction: this refers to the examination of the software output (e.g. status or error messages) and inferring information about the malware operation.

3. Fingerprinting: this includes cryptographic hash computation, finding the environmental artifacts, such as hardcoded username, filename, registry strings.

4. AV scanning: if the inspected file is a well-known malware, most likely all anti-virus scanners will be able to detect it. Although it might seem irrelevant, this way of detection is often used by AV vendors or sandboxes to "confirm" their results.

5. Disassembly: this refers to reversing the machine code to assembly language and inferring the software logic and intentions. This is the most common and reliable method of static analysis.

Static analysis often relies on certain tools. Beyond the simple analysis, they can provide information on protection techniques used by malware. The main advantage of static analysis is the ability to discover all possible behavioral scenarios. Researching the code itself allows the researcher to see all ways of malware execution that are not limited to the current situation. Moreover, this kind of analysis is safer than dynamic, since the file is not executed and it cannot result in bad consequences for the system. On the other hand, static analysis is much more time-consuming. Because of these reasons it is not usually used in real-world dynamic environments, such as anti-virus systems, but is often used for research purposes, e.g. when developing signatures for zero-day malware. Dynamic Analysis: Unlike static analysis, here the behavior of the file is monitored while it is executing and the properties and intentions of the file are inferred from that information. Usually, the file is run in the virtual environment, for example in the sandbox. During this kind of analysis, it is possible to find all behavioral attributes, such as opened files, created murexes' etc. Moreover, it is much faster than static analysis. On the other hand, the static analysis only shows the behavioral scenario relevant to the current system properties. For example, if our virtual machine has Windows 7 installed, the results might be different from the malware running under Windows 8.1.

5 Conclusion

This paper presents the survey about malware detection and classification using different machine learning algorithms. The paper defines the different tools used in this work, what are the machine learning algorithms used in this work, from what sources dataset is collected, and what are the future works are proposed all are discussed in this paper. In the discussion, it clearly identifies that machine learning algorithms are very useful for the classification and clustering of malware samples for small datasets and for large volumes of data.