EURASIAN JOURNAL OF ACADEMIC RESEARCH
Innovative Academy Research Support Center
UIF = 8.1 | SJIF = 5.685 www.in-academy.uz

A COMPARATIVE ANALYSIS OF VIRUSTOTAL AND DESKTOP ANTIVIRUS DETECTION CAPABILITIES

[1]Latipova Nodira Xalimovna
Teacher of the Department, "Systematic and Practical Programming", Tashkent University of Information Technologies named after Muhammad Al-Khwarizmi, UZBEKISTAN
[2]Ibragimov Jaloliddin Obidjon o'g'li
Teacher of the Department, "Systematic and Practical Programming", Tashkent University of Information Technologies named after Muhammad Al-Khwarizmi, UZBEKISTAN
https://doi.org/10.5281/zenodo.7478812

ARTICLE INFO

Received: 13th December 2022
Accepted: 21th December 2022
Online: 23th December 2022

KEY WORDS

VirusTotal, antivirus evasion, malware detection.

ABSTRACT

VirusTotal has been widely used and being adopted by researchers mainly for the classification of files as malicious or not. Unfortunately, it is not well understood how reliable the results from the antivirus engines on VirusTotal are, especially compared to their desktop counterparts. In this paper, we shed light on the blackbox testing functionality of VirusTotal by evaluating the detection results of VirusTotal antivirus engines and their equivalent desktop versions. Based on our results, we arrive to the conclusion that there are discrepancies between the engines on VirusTotal and the desktop engines. In general, the malware detection rate of the engines on VirusTotal is lower compared to desktop products. This is mainly attributed to the fact that VirusTotal engines do not take advantage of cloud-based detection deteriorating their performance.

I. Introduction

Malware, a portmanteau for malicious software, is designed to cause harmful consequences to computer systems varying from minor inconveniences such as slowing computers and popup ads to more severe repercussions such as unauthorized access and data theft and loss. Because of the alarming ramifications that malware can lead to, it is imperative to detect malware as fast as possible. Nowadays, several online scanning platforms exist that are able to scan files to detect malware or URLs for detecting phishing and malware hosts [1][2][3][4][5]. These platforms scan files and URLs using multiple vendors and present the combined output of different products. VirusTotal is perhaps the most popular one out of these scanning platforms. In VirusTotal, users can upload files to be checked by over 70 antivirus (AV) vendors, in order to determine if the files are malicious or not [6]. After scanning the file, the platform provides a total detection result from all the employed detection engines. As such, the platform does not label applications as malicious and benign; instead, it is up to the user to decide upon strategies to interpret the provided information.

A small number of previous works use VirusTotal's results to build their own system or as their comparison baselines. VirusTotal also includes an API for integration with external services [8]. Despite the heavy usage of VirusTotal by previous works, there is no clear understanding of how this platform delivers its results. Since the platform operates basically as a blackbox, the produced results may be questionable. Even some previous works acknowledge this fact such as, which mentions that "AV vendors only deployed light-weighted engines on VirusTotal, with most of them being signature- based in order to achieve instant detection". Moreover, as stated by "VirusTotal runs stripped-down engine versions that do not always reflect the best detection capability of an AV vendor". After all, VirusTotal itself reports, "AV solutions on VirusTotal may differ from their public commercial versions" [7]. That being said, it is crucial to determine how big this difference is, why it exists, and how reliable VirusTotal is in general, since it affects published papers and, more critically will affect future research.

In this paper, we evaluate the reliability of VirusTotal by examining discrepancies in the detection results between VirusTotal and desktop engines. We focus exclusively on malware analysis since URL scanning has been extensively examined in the past . More specifically, our main objectives are:

1. Pinpoint discrepancies between VirusTotal and desktop engines.
2. Identify how large those discrepancies are and if they could majorly affect the reliability of the VirusTotal platform.
3. Analyze the reasons behind these discrepancies.

In order to answer the research questions posed above, first we created our malware dataset using a set of 16 open source AV evasion tools that obfuscated 2 Metasploit payloads resulting in 50 malicious files in total. We also selected the 12 most popular AV products to measure their detection capabilities.

The rest of the paper unfolds as follows. Section II presents the related work and the contradictory results obtained. Section III describes the methodology adopted for the carried- out experiments. Section IV presents the results while Section V lists a set of observations. Finally, Section VI concludes the article.

II. Related work

As we mentioned previously, VirusTotal has been extensively used by researchers to annotate sample datasets as malicious (executable .exe or Android APK files) or for system evaluation. However, very few papers have examined the reliability of VirusTotal in terms of its malware detection results. For example, the work in [12] mentions but does not verify that AV vendors change the settings of their products specifically for VirusTotal compared to their commercial versions.

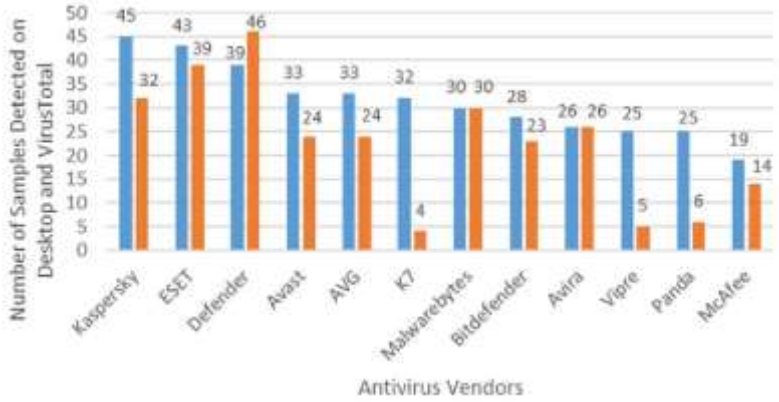
The closest works with ours are [10] and [11] that compare the detection capabilities of desktop and VirusTotal engines. More specifically, the authors in [11] set up malicious phishing websites in order to test the reliability of the engines on.

EURASIAN JOURNAL OF ACADEMIC RESEARCH
Innovative Academy Research Support Center
UIF = 8.1 | SJIF = 5.685 www.in-academy.uz

Desktop VirusTotal

Fig. 1: First Scenario - Overall Malware Detection on Desktop and VirusTotal

More specifically, from Fig. 1, we can observe that the AV engines on VirusTotal (the 12 selected AV engines), managed to detect in total 273 out of 600 compared to the 378 that the equivalent desktop engines detected. That being said, we have to take into account that the desktop products did not have their cloud capabilities enabled which would likely result in higher detection rates for the desktop products. This will be examined below in the second scenario of our experiments. Regarding the detection phase of the desktop AVs, out of the 378 detected samples, 222 were detected at download, 50 at scanning and 106 at execution (see Fig. 2). When malware is detected at download or scanning, we can assume that it was detected with signature based detection techniques. On the other hand, when malware is detected at runtime, it is probably due to behavioral or heuristic based detection techniques. All in all, roughly 1/5 of the samples were detected by behavioral or heuristic means. We cannot reproduce the same results for VirusTotal, as the latter does not provide any information regarding the detection method.



■ Desktop ■ VirusTotal

Fig. 3: First Scenario - Malware Detection for each Engine on Desktop and VirusTotal

Next, we elaborate on the detection rate of each AV individually both on desktop and VirusTotal. Fig. 3 demonstrates the malicious files each AV engine managed to detect both on desktop and on VirusTotal (the number of files detected out of the 50 files in total). It is important to mention here that the evaluation of the detection capabilities between the AV engines is out of scope of the paper. Instead, the goal here is to identify and demonstrate the differences between the engines on VirusTotal and desktop engines.

We continue by pinpointing the possible reasons of the presented irregularities

between online (VirusTotal) and desktop engines. A careful examination of the results reveals that K7 seems to primarily use behavioral or heuristic based detection, since it detected most samples at execution phase. In particular, out of the 50 malicious files, K7 detected 32, with 28 of those being detected at execution (through behavioral or heuristic based detection). The rest of the detected malicious files (i.e., four malicious files) were detected at download. These four malicious files were the only ones that were detected by K7 in VirusTotal. This behavior was exhibited by almost every desktop AV engine that was tested. hat is, if a malicious file was detected at execution, it was not detected by the corresponding VirusTotal AV engine, while every file detected at download or

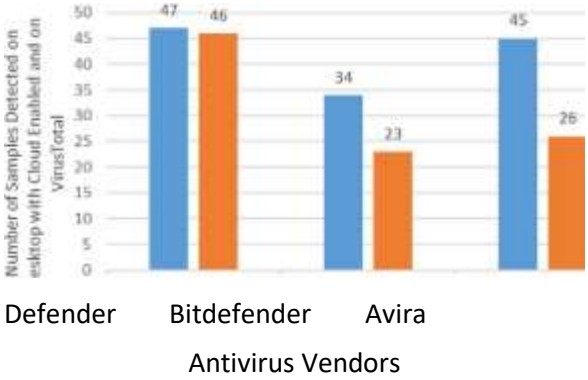


Fig. 4: Second Scenario - Malware Detection on Desktop and VirusTotal with Cloud Enabled

scanning by the desktop engines (through signatures), was detected by the corresponding VirusTotal engines. The above corroborates also the fact that there were no irregularities in Avira and Malwarebytes, simply because these two AVs do not employ behavioral or heuristic based detection (at least they do not without their cloud capabilities).

Another interesting outcome was that Windows Defender was the only AV to have increased detection on VirusTotal. We speculate that Windows Defender relies heavily on cloud detection nowadays and this is the reason why its VirusTotal version had higher detection rates. We validate this assumption in our second scenario of experiments in which we have three AVs connected to the internet, Windows Defender, Avira and Bitdefender. Keep in mind, that this set of experiments was performed one day after our 50 malware files were submitted to VirusTotal during our first scenario of experiments.

VI. Conclusions

Due to VirusTotal having seen such an extensive usage, it is imperative to find whether there are discrepancies in malware detection between the AV products on the VirusTotal platform and their corresponding desktop versions, and how large those discrepancies are. In this paper, we examined the reliability of the VirusTotal platform. First, we created our dataset which was comprised of 50 malicious files and used it to compare the detection rate of the AV products on VirusTotal with their desktop counterparts. Results showed that there is inconsistency between the engines on VirusTotal and their desktop versions. That is, VirusTotal exhibited lower malware detection rates for most AV engines compared to their desktop counterparts. While some AVs such as Avira demonstrate small inconsistencies, others such as K7 exhibit large differences between their VirusTotal and desktop engines. We concluded that this may be attributed to the fact that AV engines on VirusTotal do not include cloud-based

detection, or they are shipped with different settings compared to their desktop counterparts.

References:

1. VirusTotal. https://www.virustotal.com/. [Last accessed on May 2022].
2. Hybrid-Analysis. https://www.hybrid-analysis.com/. [Last accessed on May 2022].
3. AntiScan. https://antiscan.me/. [Last accessed on May 2022].
4. VirSCAN. https://antiscan.me/._[Last accessed on May 2022].
5. Jotti's malware scan. https://virusscan.jotti.org/. [Last accessed on May 2022].
6. VirusTotal - Contributors. https://support.virustotal.com/hc/en-us/articles/115002146809-Contributors. [Last accessed on May 2022].
7. VirusTotal - Frequently Asked Questions (FAQ). https://support.virustotal.com/hc/en-us/articles/115002122285-AV- product-on-VirusTotal-detects-a-file-and-its-equivalent-commercial- version-does-not. [Last accessed on May 2022].
8. VirusTotal API. https://developers.virustotal.com/reference/overview. [Last accessed on May 2022]. Abrams, R. VirusTotal Tips, Tricks and Myths.
9. https://www.virusbulletin.com/uploads/pdf/magazine/2017/VB2017- Abrams.pdf [Last accessed on May 2022].
10. VirusTotal Article. https://support.virustotal.com/hc/en-us/articles/11500 2122285-AV-product-on-VirusTotal-detects-a-file- and-its-equivalent-commercial-version-does-not. [Last accessed on May 2022].
11. Avet - AntiVirus Evasion Tool. https://github.com/govolution/avet. [Last accessed on May 2022].
12. Phantom-Evasion. Python antivirus evasion tool. https://github.com/oddcod3/Phantom-Evasion. [Last accessed on May 2022].