KÖTÜ AMAÇLI YAZILIMLARIN TESPİTİNDE ANTİVİRÜS PROGRAMLARI VE STATİK ANALİZ ARAÇLARININ PERFORMANS KARŞILAŞTIRMASI

Performance Comparison of Static Malware Analysis Tools Versus Antivirus Scanners To Detect Malware

Ömer ASLAN
Department of Computer Engineering, University of Siirt
omer.aslan@siirt.edu.tr

Abstract

Any software which executes malicious payloads on victim machines is considered as a malware such as the following: Viruses, worms, Trojan horses, rootkits, backdoor and ransomware. In recent years, the number and the severity of these malicious software have been increasing rapidly. The harm that malware inflicts on the world economy and private companies' assets is increasing every day. Thus, there is an urgent need to detect and prevent malware before damaging to the important assets in world wide. There are lots of different methods and tools to combat against malware. In this paper, static malware analysis tools such as (Peid, PEview, Bintext, MD5deep, Dependency walker, and IDA Pro) and antivirus scanner tools such as (Norton, McAfee, Kaspersky, Avast, Avira, Bitdefender, and ClamAV) have been examined. In a test case, 200 malware and benign were collected from different sources and analyzed under different version of Window machines. Test results show that for existing malware, antivirus software detect malware fast and efficient when compared to static analysis tools. However, for unknown malware static analysis tools performed reasonably better than antivirus software.

Keywords: Malware Analysis, Static Malware Analysis Tools, Malware Detection, Performance Comparison of Tools to Detect Malware

I. INTRODUCTION

Malware stands for malicious software, which is installed on a computer system without the knowledge of the system owner [2], and it performs malicious actions such as stealing confidential information, allowing remote code execution, and it can cause denial of service such as the following: viruses, worms, Trojan horses, and ransomware. These days, the number and the severity of these malicious software have been increasing sharply. The harm that malware inflicts on the world economy and private companies' assets is increasing every day. For example, according to Cyber Security Business Report written by Steve Morgan in 2016, whole world losed approximately $3 trillion in 2015, and expected to cost the world more than $6 trillion by 2021 [11]. Most of the cyber related attacks coming from malware.

To understand what malware is doing, it needed to be analyzed. Malware analysis is the process of determining the functionality of given malware samples and providing countermeasures to it based on findings [2]. There are mainly two approaches to analyzing malware: Static and dynamic malware analysis. Static analysis examines the malware without actually running the malicious code; on the other hand, dynamic analysis examines the malware by running malicious code and observing its behavior. In this study, static malware analysis is examined.

The rest part of this paper is organized as follows: Section II, and III describes the malware analysis methods and tools. Case study is presented in section IV. Results and discussion are explained in section V. Finally the conclusions and future work are given in section VI.

II. MALWARE ANALYSIS METOD

In order to understand malware malicious intent, the malware sample must be analyzed. There two ways to analyze malware: Static and Dynamic analysis. In this study, static malware analysis method have been examined deeply.

1) Dynamic Malware Analysis

Dynamic malware analysis observes the behavior of the malware during its execution. In order to specify the system behaviors; function calls, function parameters, information flow and instructions are analyze [1]. For dynamic analysis, protected environment such as virtual machines and sandboxes are used. To detect unknown malware, the dynamic analysis works better than the static ones, but, static analysis work faster for known malware.

2) Static Malware Analysis

Static analysis shows the structure of the program without actually running it. Static analysis begins with

basic to advanced analysis. Basic static analysis provides a signature, string or hash which view complete structure of program. For already known malware, it is quite fast and precise [2]; however, it fails to detect unknown malware. On the other hand, advanced static analysis use debugger and disassembler to provide program instructions which describe program semantic [1, 2]. Since advanced static analysis catch some unknown malware, applying this technique requires advanced knowledge including operating system concept, knowledge of disassembly [2, 3]. In static malware analysis there are a lot of different ways to extract information from sample including [2]: Antivirus scanning, hashing, string analysis, and reverse compiling.

2.1) Antivirus Scanning: Antivirus scanner used as a one of the first step to analyze malware. It uses signature of the file which describe suspicious code [2, 3]. First, predefined malware signatures are stored on a database. Second, antivirus scanner find the signature of the file and it compares with signatures which has been stored in database. If signature is in database, sample marked as malware, otherwise marked as benign. Even though antivirus scanning works fast and accurate in order to detect known malware, it fails to detect unknown malware. Unknown malware is generally variant versions of known malware that change its signature to bypass antivirus scanner [2].

2.2) Hashing: It is a method that defines the malware. Firstly, sample is submitted to hashing program, and hash value extracted such as MD5, or SHA-1, SHA-2 [2]. Then, the extracted hash value searched online and seen whether the same hash value identified malware or not.

2.3) String Analysis: Looking for specific string may provide appropriate information about malware such as URL's associated with malicious code, email addresses belong to attacker, or information related to malicious password. These information can be extracted by using string extraction tools.

2.4) Reverse Compiling: It takes binary version of malware samples and generates assembly level instructions [4]. During the reverse compiling the structure of the program can be analyzed such as which functions have been used, registers status, status of stack .Analyzing malicious code reversely may give some clue about malware, however, it requires deep understanding of assembly level language [2, 4].

III. STATIC MALWARE ANALYSIS AND DETECTIONS TOOLS

There are two main categories to analyze malware statically: Using antivirus scanner and static tools. In antivirus software, program automatically identify given sample malware or benign, on the other hand, static tools use such as string analysis, reverse compiling and produce raw data which is needed to be interpreter by malware analyst.

A) Antivirus Software:

Antivirus is a software program which prevent the devices such as computers, mobiles and pen-drive [5, 6] from malicious software. It works without manual interaction on user devices. Antivirus software uses signature to detect malware. Signature is a unique feature of a malware which is similar to fingerprint. It is quite fast, but it fails to detect new types of malware. Antivirus software uses different types of scan such as full scan, quick scan and custom scan.

Full Scan: Full scan is performed on computer system to ensure that there is no malware on system [6]. It scans files, local drive, and folders. Full scan also can be performed on external devices such as digital camera, hard drives, usb drivers so on.

Quick Scan: It used to scan most infected folders and files on the system. Common scanned areas including: hard drive, temporary files, computer memory, and the operating system directory. It takes approximately 25 minutes to complete scanning.

Custom Scan: It allows user to customize which files and folders to scan. It is useful when the user wants to perform a scan for a particular folder.

Table I: Well-known antivirus software

Name of Tools Antivirus Tool	Description
Norton	It is one of the well-known antivirus software and it updates itself every 10 to 15 minutes
McAfee	It is a software packages protect computer system from viruses, and malware. It also has firewall to prevent suspicious traffic to come computer.
Kaspersky	It is an antivirus software that try to protect computer system from viruses, spyware, Trojans, and ransomware.
Avast	It is a software that protect computer system from malware. It includes antivirus with Avast Passwords, antispyware, streaming updates, Secure HTTPS scanning, and Home Network Security Scanner
Avira	It is a software packages that offers various antivirus programs for private users and companies. IT provides automatic real-time monitoring and integrated cloud-scanning technology
Bitdefender	It protect from spyware, viruses, root-kits, and offers a laptop and gamer model
ClamAV	It is a free, cross-platform, open source antivirus software that detect malware

The common steps to use antivirus:

Step 1: Install antivirus software to computer system.

Step 2: Right-click on the Antivirus System icon and navigate it.

Step 3: Click on scan tab.

Step 4: Select the type of scan available in provided option (full scan, quick scan, custom scan).
Step 5: Click the start scan.
Step 6: Scan result will be displayed on the screen.
Step 7: User select the remove button to remove malware from system.

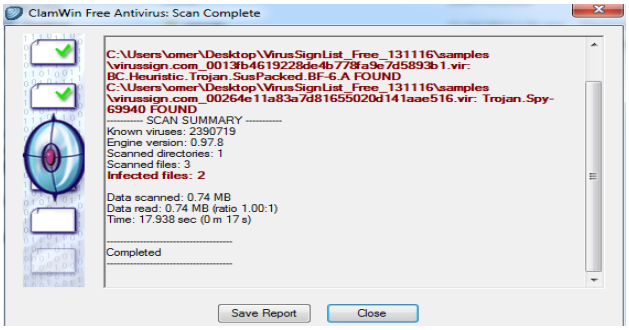

Fig. 1. Typical antivirus scanning result

B) Static Analysis Tools:

There are variety of static tools such as PEiD, BinText, UPX, and Dependency Walker to analyze and detect malware [2]. Many of these tools mainly use signature and pattern matching techniques. These tools require manual work to analyze and detect malware [2]. The detailed information about tools can be seen from table II.

Table II. Summary of Static Malware Analysis Tools.

Name of Tools	Description
PEiD	Tool for detecting the packed malware.
PEview	Tool to display the structure and content of the portable executable.
PEBrowser Professional	It is a disassembler for Win32 and Win64 executables.
BinText	Tool that capable of searching and displaying the character strings from in a binary file.
UPX	Tool for executable packer which is used to compress malware sample.
MD5deep	Tool to compute MD5, SHA-1, SHA-256
Dependency Walker	Tool to explore DLLs and functions which has been imported by malware.
Resource Hacker	Tool to view, modify, and extract resources from PE.
IDA Pro	It is a disassembler which is widely used by malware analysts, reverse engineers, and vulnerability analysts.

More detailed information about static malware analysis tools:

PEiD: It is a free tool which detect packers and compilers in PE format. It has approximately 600 [3] signatures to detect packers, cryptors, and compilers.

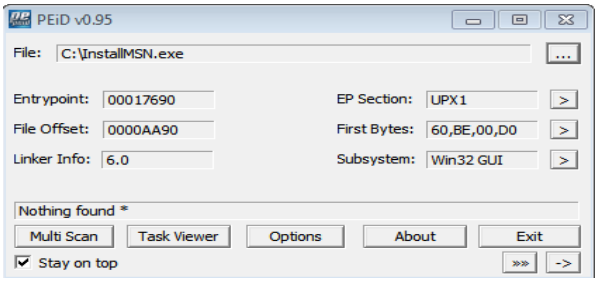

Fig. 2. Specify suspicious sample has been packed using PEiD.

PEview: It is a free tool that provides information about PE file header and its sections [7]. A number of useful information can be extracted by using this tool such as program complied time, import-export functions, and size of the program when it resides on the memory and disk [7].

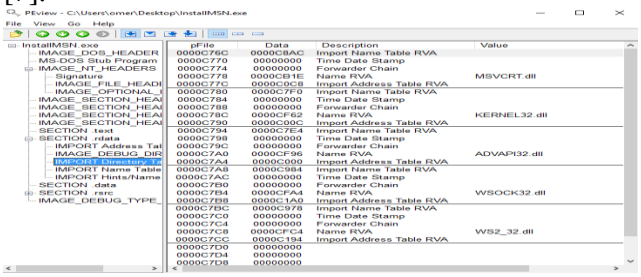

Fig. 3. Analyzing PE InstallMSN.exe malware with program PEview.

PE Explorer: It is a tool that provides information about PE header, and its sections [3]. PE Explorer also provides unpacker for packed files which has been packed by using common malware packer such as UPX, and NsPack [2].

BinText: It extracts relevant information used as a text in malware [3]. It can be used for different file extension formats and text representation such as plain text, ASCII text, and Unicode [8].

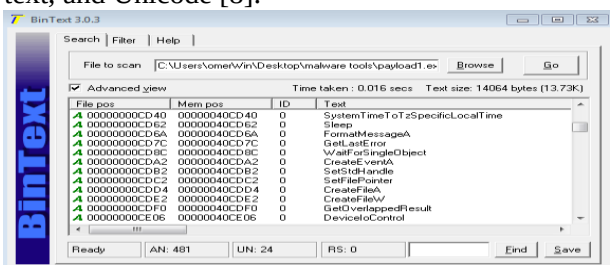

Fig. 4. Analyzing malware strings by using BinText (list of strings are shortened)

UPX: Ultimate Packer for Executables (UPX) is a free executable packer that used by malware author to pack malware [2, 7]. It is difficult to analyze and detect packed malware without unpack it.

Resource Hacker: Resource hacker known as ResHack which is a resource extraction and editing tool that has been developed for Windows Operating System [9]. A lot of useful tasks can be performed by using the tool include: Viewing, compiling, recompiling resources [9] for both 32 bit and 64 bit Windows executables.

IDA Pro: Interactive Disassembler (IDA) is a disassembler that generates assembly language from binary executables. It support various different operating systems (Microsoft Windows OS, Mac OS X, and Linux OS) and executable file formats such as PE, common object file format, and executable and linking format [7]. IDA Pro performs tasks including stack analysis, local variable identification, and function analysis [2]. By using IDA Pro, researchers can identify characteristics of malware with analyze the assembly language.

Dependency Walker*:* It is a free static analysis tool [2, 7] that implemented for Microsoft Windows which is used to view DLLs and functions [10] imported by malware for PE file. It also shows lists of dependencies in a tree view when malware is run.

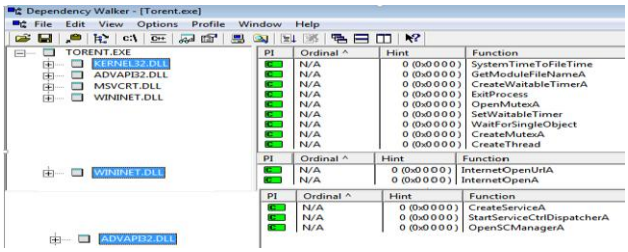

Fig. 5. Explore DLLs and functions imported by malware by using dependency walker (list of functions are shortened)

IV. CASE STUDY

This section expresses case study and their results. We performed test case on Windows virtual machines including: Windows 7, Windows 8.1, and Windows 10. For test case, 9 well-known static analysis tools and 7 well-known antivirus scanner software has been used. To show the effectiveness of the static tools and antivirus scanner, 200 malware and 200 benign samples have been tested. Malware samples gathered from variety of sources including: Online websites such as contagio malware dump,open malware, and kernelMode.info;backdoors generated by using Metasploit framework; viruses, and worms created by using Python scripting language.

To detect malware effectively and fast, combination of different static tools, and antivirus scanner have been tested on variety of samples. Since different tools represent different aspect of the given sample, using appropriate tools together provide more information when detecting malware. Detailed test results and discussion can be seen in section V.

V. RESULTS AND DISCUSSION

To compare the tools' efficiency, TP, TN, FP, FN, DR, and ACY metrics, which stand for true positive, true negative, false positive, false negative, detection rate, and accuracy, are measured respectively.

Based on the TP, TN, FP, and FN results, DR and ACY are calculated as:

$$DR = TP / (TP + FN) \quad (1)$$

$$ACY = (TP + TN) / (TP + TN + FP + FN) \quad (2)$$

The test results summary can be seen from Fig. 6, Fig. 7, Table. III, and Fig. 8. It can be clearly seen from test results that it is difficult to detect malware by only using one static tool or a few tools. Using only static analysis tools or antivirus software may not be enough as well. Test results also showed that for known malware antivirus software outperformed static analysis tools. However, for unknown malware static tools outperformed antivirus software. To correctly mark a suspicious program, it is recommended to use static tools with antivirus scanners.

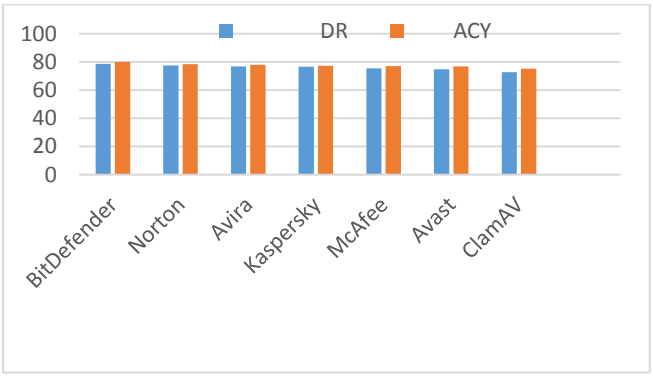

Fig. 6. Performance of antivirus scanner on known malware measuring by detection and accuracy rate

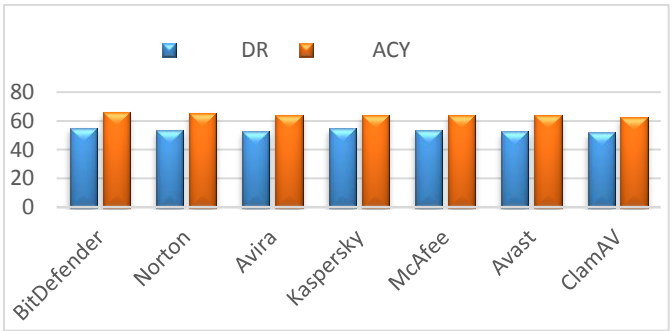

Fig. 7. Performance of antivirus scanner on unknown malware measuring by detection and accuracy rate

For known malware selected antivirus software detection and accuracy rate's results are more or less the same, it measured about 79% and 80%, respectively [Fig. 6]. Since antivirus software could not update its signature

every seconds, it could not catch all known malware. The performance of the selected antivirus software rated as follows: BitDefender, Norton, Avira, Kaspersky, McAfee, Avast, and ClamAV [Fig. 6]. For unknown malware, the performance of the antivirus software declined sharply (Fig. 7). The detection rate declined from 79% to 56% and accuracy declined from 80% to 65% [Fig. 7]. This results show that antivirus software cannot detect zero-day malware.

Table. III. Performance of antivirus scanner versus static analysis tools for known malware and unknown malware measuring by detection rate and accuracy

Malware Analysis	Name of Tools	For Known Malware		For Unknown Malware	
		DR	ACY	DR	ACY
Static Analysis Tools	1. BinText, Dependency Walker	66.2 %	71.2 %	54.2 %	56.4 %
	2. PEiD, Dependency Walker	66.4 %	72.4 %	55.4 %	56.2 %
	3. Dependency Walker, PeiD, PE Explorer	67.6 %	75.9 %	58.8 %	58.6 %
	4.PEiD, BinText, PEview, IDA Pro	74.2 %	78.2 %	60.7 %	61.9 %
	5. UPX, IDA Pro, BinText MD5deep, Resource Hacker	83.2 %	84.2 %	66.3 %	68.2 %
Antivirus Scanners	1. Norton	78.2 %	79.2 %	52.2 %	53.2 %
	2. Bitdefender	77.2 %	77.8 %	53.6 %	53.7 %
	3. Avira, Kaspersky	77.3.2 %	80.5%	54.2 %	55.2 %
	4. ClamAV, McAfee Avast	79.6 %	80.2 %	56.2 %	56.8 %
	5. Norton, ClamAV, Kaspersky, Bitdefender	84.2 %	85.7 %	59.2 %	58.9 %

Table. III shows that the performance of static analysis tools versus antivirus software for known and unknown malware. Test results show that for known malware antivirus scanners outperformed static tools by 84.2 %, 85.7 % versus 83.2 %, 84.2 % detection rate and accuracy. Since antivirus software works automatically, it is pretty fast, on the other hand, static tools requires hard man-power to interpret the raw tool results, it took a lot of times to marked sample as malware or benign. However, for zero-day malware, static tools outperformed antivirus scanners by 66.3 %, 68.2 % versus 59.2 %, 58.9 % detection rate and accuracy. This imply that for unknown malware neither static tools nor antivirus scanners are effective. But, using static tools may be better choice for unknown malware. Fig. 8 shows that the performance of static tools and antivirus software declined sharply for zero-day and complex malware.

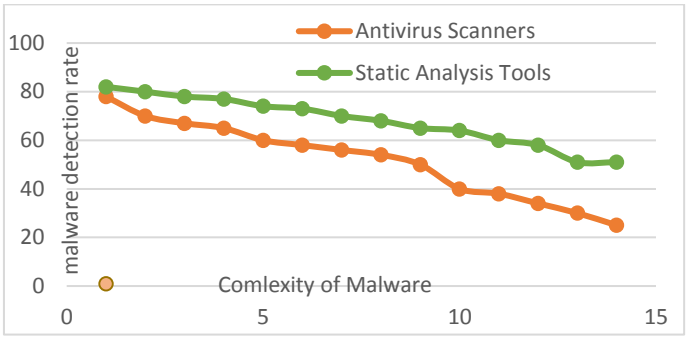

Fig. 8. Performance of antivirus scanners versus static analysis tools for known malware to unknown and complex malware

VI. CONCLUSIONS AND FUTURE WORK

This research focus on comparing the performance of static malware analysis tools versus antivirus software for existing and unknown malware. Test results have shown that it is almost impossible to detect malware by using only one static tool. However, by using a combination of static tools, the detection rate was improved immensely: 83.2% DR and 84.2% ACY. It was found that using UPX, BinText, IDA Pro, MD5deep, and Resource hacker together generated the best results for detecting malware samples. Test results also indicated that antivirus scanners outperformed static analysis tools by 83.2% DR, 84.2% ACY, 84.7% DR, and 85.7% ACY. Signature-based detection tools such as antivirus scanners are fast and effective when detecting existing malware, but it almost impossible to detect unknown malware. On the other hand, static detection tools are more accurate when detecting more complex and zero-day malware. However, static analysis tools cannot detect a lot of new unknown malware too. Most of the tools used for this research are completely free or have free versions that can be easily downloaded from websites. However, using these tools requires expertise and a lot of time. For example, analyzing a complex unknown malware may take an analyst several days by using static analysis tools. Hence, there is a huge demand to make analysis processes automated. Therefore, it can be concluded that static malware detection tools and antivirus scanners are fairly effective when detect existing malware, but unable to detect new malware. The researchers need to focus on detection mechanisms that can effectively detect more complex and unknown malware. In the future, we are going to develop static tools which will detect specifically zero-day malware.

REFERENCES

[1] Uppal, Dolly, Vishakha Mehra, and Vinod Verma. "Basic survey on malware analysis, tools and techniques." International Journal on Computational Science and Application (IJCSA) 4.1 (2014): 103-112.

[2] Ömer Aslan, Refik Samet "Investigation of Possibilities to Detect Malware Using Existing Tools",14th ACS/IEEE International Conference on Computer Systems and Applications AICCSA 2017 October 30th to November 3rd, 2017.

[3] Sikorski, Michael, and Andrew Honig. Practical malware analysis: the hands-on guide to dissecting malicious software. no starch press, 2012.

[4] Ligh, Michael, et al. Malware analyst's cookbook and DVD: tools and techniques for fighting malicious code. Wiley Publishing, 2010.

[5] https://www.1and1.com/digitalguide/server/security/comparison-of-the-best-antivirus-programs/

[6] http://www.computerscijournal.org/vol10no1/performance-investigation-of-antivirus-a-comparative-analysis/

[7] Eilam, Eldad. Reversing: secrets of reverse engineering. John Wiley & Sons, 2011.

[8] Prayudi, Yudi, and Imam Riadi. "Implementation of malware analysis using static and dynamic analysis method." International Journal of Computer Applications 117.6(2015).

[9] http://www.angusj.com/resourcehacker/

[10] Davis, Michael, Sean Bodmer, and Aaron LeMasters. Hacking Exposed Malware and Rootkits. McGraw-Hill, Inc., 2009.

[11] Steve Morgan, "CyberSecurity Business Report", Aug 22, 2016

Arş. Gör. Ömer ASLAN, Siirt Üniversitesi, email:omer.aslan@siirt.edu.tr