International Journal of Advanced Computer Research (ISSN (print): 2249-7277 ISSN (online): 2277-7970)
Volume-4 Number-1 Issue-14 March-2014

Operating System's Changes When Installing an Anti-virus Application

Mohammed A. Shehab[1], Wegdan A. Hussien[2]

Abstract

Operating systems are the interface between hardware and end user. This software are under attack from malicious programs, so we need anti-malware application to make those operating systems save. However, most people use many anti-malwares to be protected, but they did not know when they install this application what this application can do inside their devices. In this paper, we conduct a test on an anti-malwares and investigate its effects inside the operating system. We exploit virtual box to create virtual machine and use windows XP as a target operating system with three anti-viruses: Kaspersky, AVG and Norton. We examine differences that happen after installing these anti-viruses (we mainly focus on files kernel, processes, sockets and registry keys).

Keywords

windows XP operating system, anti-viruses, filter drivers, malwares.

1. Introduction

Computer is the main infrastructure of any companies on these days, because it saves theirs information and can finish works faster and more accurate than human. Any person on this era need PC, laptop or any smart device to help him/her to do their work or to save it [1]. However, all these devices are under attacks from malicious people, because these devices save a very sensitive information like bank accounts, company information, company customer's data and important personal data. In addition, this information is not only under attack from parasitic people, also this information or data can be lost, destroyed or modified be malicious programs. To be safe from all the previous problems, you must install anti-malware scanner such as Kaspersky, Node32, Norton, and so on.

After installing anti-malware application most of people do not know what this application does inside their devices. Besides, most people do not know what are the new files installed inside the operating system and what they actually do.Our work is to search for these new changes inside an operating system and discover what they do inside the device's files [2].

We use windows XP as operating system and three types of anti-viruses: Kaspersky, AVG and Symantec anti-virus. We design a tool in C# language to search for processes, kernel files, system registry and internet sockets or connection ports. Also, we use *Sysinternal* application from Microsoft Company to scan the same fields, then we compare the results.

In the next section, we give brief description on windows XP and its drivers. In addition, we talk about where anti-viruses must be installed and why must be installed there. In section three, we write about some relevant work in the literature. In section four, we explain our approach as well as describe the tool that we programed to retrieve specific information for our research and report it. In section five, we show the results for our tool and for the *Sysinternal* package. Finally, section six states the main conclusions and future work.

2. Background

The kernel of windows XP is saved in a folder named windows. This folder size is about 5 gigabytes. The main core of windows XP is stored on a folder named system32. This folder save the drivers and settings of system configuration.

File System Drivers (FSDs) are the drivers which mange all files format. These drivers run in kernel mode of windows XP and it is also inside system32 folder.

Most of Security System Components are saved inside system32 folder such as:

- Security reference monitor (SRM)
- Local security authority subsystem (Lsass)
- Security Accounts Manager (SAM), Active Directory
- Logon process (Winlogon), Graphical Identification and Authentication (GINA)

Manuscript received March 20, 2014.

Mohammed Ali Alawi Shehab Department of Computer Science Faculty of Computer and Information Technology.

Wegdan Abdulqader Hussien, Department of Computer Science Faculty of Computer and Information Technology.

- Network logon service (Netlogon) and Kernel Security Device Driver (KSecDD).

Kernel Security Device Driver (KSecDD) is a library on kernel mode that uses local procedure call (LPC) as interface with another kernel components. Windows XP uses Software Restriction Policies as a mechanism to access images and scripts execute on their systems to make it difficult to attack kernel components on it [3]. Malwares must attack at the first the kernel files to get all permissions to access all files. So, anti-viruses need a high permission to access all operating system files, but they still are the same as any application added to any version of windows. On the other hand, Microsoft make its operating system closed so that no one can know its components. To solve these problems, Microsoft gives programmers permission only by using authentication drivers. These drivers known as Filter Drivers.

Filter Driver: is an additional driver that adds or edits the behavior of a device. One filter device can servers one or more devices [4].

3. Literature Review

Most of researchers who work on antiviruses focus on the way the antivirus works to detect malicious programs weather it is signature-based or behavior-based. In spite of signature-based is considered as traditional approach used to detect known malware, but it is still an efficient approach because of its low rate of false positive [5] [6]. Unfortunately, the development of malware have increased and introduced a lot of techniques to bypass signature detection that depends on pattern-matching [7]. In order to keep up with the rapid growth of malware, behavioral based techniques have been introduced to discover new malicious software by monitoring their behaviors for any suspicious activities on the operating system (such registry editing, system files modifying or downloading from Internet). Recently, using machine learning techniques has a great interest among antivirus companies since it has a strong impact on detecting new malware based on behavior activities [8] [9] [10]. All commercial anti-viruses are not open source and some of these anti-virus check files signature with their database (this database sometime exists on server side like Symantec antivirus), because of that you cannot trust if the server of anti-virus still read your files even when finish scanning local system. Slowdown system performance and consuming time when operating system need to read or write files can proof that the anti-virus search engine needs this time to do this operation for each file under scanning [2] [11] .

In our paper we don't concern on the detection mechanism of antivirus but we focus on how the antivirus interact with the operating system by knowing the changes happen to operating system after antivirus being installed. Unfortunately we faced a lot of problems collecting information about commercial antivirus because the companies don't give full information about their products and consider it as their personal effort.

4. Approach

This section is partitioned into two subsections. First, we give brief description on the Windows XP and anti-viruses. Then second, we expatiate talking about the tool design, implementation, and functionality.

A) anti-viruses and windows XP

Operating system is the interface between users and machines of devices. In addition, it is the first program runs inside a computer by boot system. This program manage all applications that are installed in computer [1].

One of the most famous and popular operating systems is windows XP. It is produced by Microsoft for personal computers use.

Anti-viruses are applications that protect our operating systems from malicious programs. These malicious programs are used to harm operating systems and make it unstable. There are many types of these malicious applications which use many techniques to delete, steal or edit files in your device. For example: Viruses, worms, Trojans.

Detecting these malwares is so difficult and needs sophisticated and complex model as shown on figure1

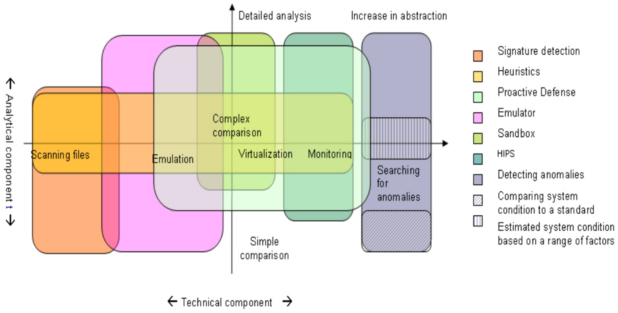

Figure 1: Describing model for Kaspersky anti-virus

This complex model makes an anti-virus accurate when it makes decision about any scanned file [6].
Anti-virus application scans files by reading the signature (hash function) and looks for this signature on its database for known malware, but this technique has limitations in protection. So, an anti-virus must apply addition techniques such as system monitoring. This technique monitors all registry keys, files, internet connections and event of processes inside operating system [12] [13].

B) Tool (design and implementation)

We wrote our tool in C# programming language. Also we designed this tool to read and obtain specific information from windows XP operating system. This tool focus in four main parts in windows XP: kernel files, processes and threads, Registry keys and internet connection ports "TCP protocol". The tool works in multi- threads programming and divided the tasks to four main threads. For example, one thread for files and another thread for getting processes' information. The threads increase the performance of the tool by doing multi-tasks at same time virtually. However, this tool needs high privilege to access this information, so we change the running mode of the tool to be in administrator mode because this is the highest privilege that we can receive from Microsoft operating systems.

- **Kernel files:**

This part divided into to two main parts: First, searching for new directories. This operation is done by using a recursion function with pre-order traversal. And second, filtering files by its extensions. This operation is done by saving the main extensions of files that are relevant to the kernel files like .EXE and .INI files [14]. This is shown in figure 2.

- **Processes and Threads:**

This part of code is done by two main threads. The first thread has the main processes that are running in the user mode. Microsoft blocks any access to the kernel space and makes it difficult due to this we work only on the user mode. After that, the second thread collects and saves all threads for each processes by its ID.

- **Registry:**

This subsection in our tool implements a recursion function similar to the function that reads the kernel files with pre-order traversal technique. In [9] they mention information about each registry key and what it saves. We focus in our tool in HKEY_LOCAL_MACHINE registry key, because it saves the all information about all software's that installed in host operating system. Moreover, we focus on another registery key called HKEY_CURRENT_CONFIG because it saves all configurations of the host operating system.

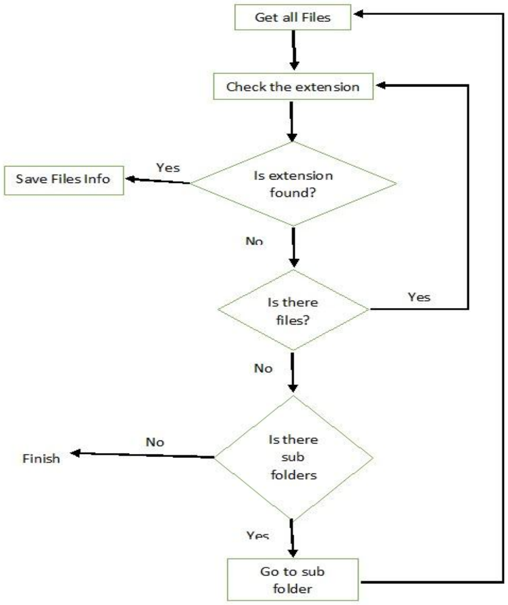

Figure 2: The tool's Flow chart

We faced some problems with some registry keys. Thus, we try to get privilege as administrator user and then access the registry key. We use the security and Permissions libraries in C# language to make this section of code accessible. We create two threads for the two main registry keys that we mentioned above.

- **Internet connection ports:**

This part restores information about all process and their threads at first, then moniters the connecton ports. The reason for that is our need to know which process or its threads are causing the connection event using TCP protocol. We rely on the *NetworkInformation* package from .NET library and also we use ***iphlpapi.dll*** file that gets the process ID, which makes the connection through the internet and its port number. Next, we save all data inside access database and make a copy of this machine to exploit it with another anti-virus appliction. We copy also the database file and reset it for the next step. Then, we download kaspersky as first anti-viruse to test it and install the trial version. Aftre running our tool again we discover new files like new drivers, new processes and new internet connection, which will be explained in details later on.

5. Evaluation and Results

a) Kaspersky anti-virus

1. Our tool

1.1 Processes and sockets

Our tool discovered four main new processes wmi32.exe, avp.exe, avpui.exe and WPFFontCache_v0400.exe. Also, we discover many packages installed by Kaspersky like .net framework version 4.0 where this version is designed by Microsoft Company to support its languages like Visual Basic.Net, F#, Visual C++ and C#. Kaspersky application is written in C# language and we discovered that when we were searching for new files installed in XP operating system. We found all libraries of C# in windows system files. Table 1 shows the details.

Table 1: show Kaspersky anti-virus processes

Process name	Company name	Size bytes	Description
wmi32.exe	Kaspersky Internet Security 2012	19,000	It is monitoring application for any data will be send from any ports on LAN network
avp.exe	Kaspersky Internet Security 2012	139,367	This process create 8 ports connection it use (1110, 1111, 1124, 1125, 1126, 1127, 1128, and 1129)
avpui.exe	Kaspersky Internet Security 2012		It one of Kaspersky classes that scheduled some jobs for this application
WPFFontCache_v0400.exe	Windows Presentation Foundation Font Cache	753,504	This process need because Kaspersky need .net framework version 4.0.0.0 to run.

1.2 Files and Registry

We found 9 new drivers in folder system32. The drivers provide Kaspersky application more powerful access to the kernel. There are 7 filter drivers that designed for windows operating system one of them is for networking. Additionally, we found Logon Visualizer Module in more than 40 locations in system 32 folder which is the core folder for windows XP operating system. Also, we found 370 new files written in assembly language with .dat extension. All these files locate in path location C:\WINDOWS\assembly\. The total new files after Kaspersky installation were 1044 new files. For Registry aspect, our tool did not find any new created Registry keys.

2. Sysinternal (Kaspersky)

2.1 Processes and sockets

The results are as the same as our tool's.

2.2 Files and Registry

For files Sysinternal does not discover any new files, however, it discovers new keys.

b) AVG anti-virus

1. Our tool

1.1 Processes and sockets

On AVG antivirus our tool discovers six main new processes avgui.exe, avgemcx.exe, avgidsagent.exe, avgrsx.exe, avgwdsvc.exe and avgcsrvx.exe. Also, we discover about 395 new files installed by AVG anti-virus. Most of the files installed in system32 folder and 18 new driver files also installed in the following paths: C:\WINDOWS\system32\drivers\disdn, C:\WINDOWS\system32\drivers\etc folders.

1.2 Files and Registry

We found 8 new drivers in system32 folder. These drivers give AVG application more powerful access to the kernel. There is one filter driver that designed for windows operating system and another one for networking. Besides, we found three new drivers installed for intrusion detection system (IDS) and two drivers for hard ware such as: printer, mouse, .etc.

All this drivers are located in two main paths:

1- C:\WINDOWS\system32\drivers\etc
2- C:\WINDOWS\system32\drivers\disdn

Table 2: registry keys of Kaspersky by sysinternal

Name	Location
klwtbbho.dll	- HKLM\Software\Microsoft\Windows\CurrentVersion\Explorer\Browser Helper Objects
avp.exe	- HKLM\System\CurrentControlSet\Services
Ie_virtual_keyboard_plugin.dll	- HKLM\Software\Microsoft\Internet Explorer\Extensions

Table 3: show AVG anti-virus processes

Process name	Company name	Size	Description
avgui.exe	*AVG anti-virus 2013*	3,147,384 bytes	Works as the interface of AVG anti-virus. Also it uses to manage the all modules of AVG application
avgemcx.exe	*AVG anti-virus 2013*	973,664 bytes	scanning user emails and detected any malicious files on it
avgidsagent.exe	*AVG anti-virus 2013*	5,832,712 bytes	Discover the intrusion detection system (IDS)
avgrsx.exe	*AVG anti-virus 2013*	287,000 bytes	Monitoring files operations
avgwdsvc.exe	*AVG anti-virus 2013*	297,752 bytes	Protect system from rootkit files
avgcsrvx.exe	*AVG anti-virus 2013*	693,016 bytes	main process that AVG anti-virus use it to scanning all system

The total new files after installing AVG were 395 new files. For Registry, our tool did not find any new Registry keys.

2. Sysinternal(AVG)

2.1 Processes and sockets

Same results as our tool has got before.

2.2 Files and Registry

For files, Sysinternal does not discover any new files, but for registry keys it discovered new keys.

Table 4 registry keys of AVG by sysinternal

Name	Location
AVG_UI	- HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run
AVG-Secure-Search Update_1213b	- HKLM\SOFTWARE\Microsoft\Active Setup\Installed Components
AVG Shell Extension	- HKLM\Software\Classes\*\ShellEx\ContextMenuHandlers - HKLM\Software\Classes\Folder\ShellEx\ContextMenuHandlers
AVGIDSAgent	- HKLM\System\CurrentControl Set\Services
Avgdiskx AVGIDSDriver AVGIDSHX AVGIDSShim Avgldx86 Avglogx Avgmfx86 Avgrkx86 Avgtdix	- HKLM\System\CurrentControl Set\Services
avgrsx.exe	- HKLM\System\CurrentControl Set\Control\Session Manager\BootExecute

c) Symantec (Norton antivirus)

1. Our tool

1.1 Processes and sockets

On Norton antivirus our tool discovers three main new processes wmiprvse.exe, NAV.exe and NST.exe. Also, we discover about 54 new files installed by Norton anti-virus. Most of the files installed in system32 folder and 8 new driver files also installed in C:\WINDOWS\system32\drivers\NST C:\WINDOWS\system32\drivers\etc folders.

1.2 Files and Registry

We found 8 new drivers in folder system32. These drivers grant Norton application more powerful access to the kernel. All these drivers named SYMEVENT.SYS. However, they are distributed in many directories with different sizes. These drivers are created as libraries for Norton anti-virus events.

The total new files after installation were 54 new files. Though, our tool did not discover any new Registry keys.

2. Sysinternal (Norton)

2.1 Processes and sockets

The results are exactly same compared to our tool ones.

2.2 Files and Registry

For files Sysinternal does not discover any new files, but for registry keys it discover new keys as shown in table 6.

Table 5: show Norton anti-virus processes

Process name	Company name	Size	Description
wmiprvse.exe	*Microsoft corporation*	218,112 bytes	Works for windows operating system to management the services for new program that was installed in windows XP
NAV.exe	*Symantec corporation*	50,795 bytes	Scanning the devices by Norton anti-virus also using port 1073 to connect with internet
NST.exe	*Symantec corporation*	129,424 bytes	Scanning for the network connection

Table 6 registry keys of AVG by sysinternal

Name	Location
NAV NCO	- HKLM\System\Current ControlSet\Services
IDSxpx86 Secdrv SRTSP SRTSPX SymDS SymEFA SymEvent	- HKLM\System\Current ControlSet\Services

6. Conclusion and Future Work

In this paper, we try to discover the changes occur in windows XP operating system when installing an anti-virus on it. Therefore, we choose three popular anti-viruses: Kaspersky, AVG and Norton to conduct experimentations. We implemented a tool with C# language to get more details deep inside operating system as possible as we can, then we run some tools from Sysinternal package to check our results and finally compare them. We observe more details by using our tool in kernel files and obtain same results with processes and sockets like Sysinternal tools; however, our tool failed to find any new registry keys as Sysinternal does. In future work we plan to use another tool like Event Tracing for Windows (ETW) in order to add more information about the differences inside windows operating system. Moreover, we could conduct the same tests for addition anti-viruses.

References

[1] D. A. S. Mark E. Russinovich, Microsoft Windows Internals, Microsoft Press, 2004.

[2] A. D. C. P. W. a. E. Z. Yevgeniy Miretskiy, “Avfs: An On-Access Anti-Virus File System,” USENIX Association, August 9–13, 2004.

[3] MSDN, 2013. [Online]. Available: http://msdn.microsoft.com/en-us/library/windows/hardware/ff545890(v=vs.85).aspx.

[4] L. M. a. A. C. L. Radvilavicius*, “Overview of Real-Time Antivirus Scanning Engines,” Journal of Engineering Science and Technology Review 5 (1) (2012) 63-71, 10 July 2012.

[5] K. Griffin, S. Schneider, X. Hu and T.-c. Chiueh, “Automatic Generation of String Signatures for Malware Detection,” Symantec Research Laboratories, p. 28, 2009.

[6] Y. Tang and S. Chen, “Defending against internet worms: A signature-based approach,” in INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE, 2005.

[7] S. A. L. C. G. P. and M. M. , “Evasion- Resistant Malware Signature Based on Profiling Kernel Data Structure Objects,” in Risk and Security of Internet and Systems (CRiSIS), 2012 7th International Conference on, Cork, 2012.

[8] F. I. L. C. E. A. and N. A. , “Analysis of machine learning techniques used in behaviorbased malware detection,” in Advances in Computing, Control and Telecommunication Technologies (ACT), 2010 Second International Conference on, Jakarta, 2010.

[9] P. T. Carsten Willems, Thorsten Holz and Konrad Rieck, “A Malware Instruction Set for Behavior-Based Analysis,” Journal of Computer Security, vol. 19, no. 1875-8924, 2011.

[10] Konrad Rieck, Philipp Trinius, Carsten Willems and Thorsten Holz, “Automatic Analysis of Malware Behavior using Machine Learning,” ACM, vol. 19, no. 4, 2011.

[11] M. Rouse, “searchcio-midmarket.techtarget.com,” 5 6 2007. [Online]. Available: http://searchcio-midmarket.techtarget.com/definition/operating-system.

[12] M. Landesman, “About.com,” 2013. [Online]. Available: http://antivirus.about.com/od/whatisavirus/tp/malwaredetect.htm.

[13] K. L. Alisa Shevchenko Virus Analyst, “Malicious Code Detection Technologies,” 2008.

[14] "Open Office," [Online]. Available: http://www.openoffice.org/dev_docs/source/file_extensions.html.

Mohammed Shehab is a master student at Computer Science Department at Jordan University of Science and Technology. His B.Sc. degree has been received from Muta'h University in Computer Science. His main research interests include Data Mining, Computer Security, Image processing and Machine learning.

Wegdan Abdulqader Hussien is a master student at Computer Science Department at Jordan University of Science and Technology. His B.Sc. degree has been received from Aden University in Computer Science and Engineering. His main research interests include Wireless Sensor Networks, Computer Security and Data Mining.