



Malware Reverse Engineering Handbook by

- Justin Valent
 - Sendy Sanjaya
 - John Lee Sheppard
-

A detailed guide to start reverse engineering malware and understand how malware works.

Table of contents

Abstract	3
Introduction	4
Static Analysis	20
Dynamic Analysis	22
Debugging / Patching	23

Abstract

The background of this problem is that the target of malware attacks is all people who have devices connected to banks, companies, and data that are like being stolen. Therefore, the purpose of this research is to gain knowledge in the form of detailed documentation about the malicious functions that exist in terms of reverse engineering and about how malware works that have been circulating in cyberspace and perform reverse engineering on malware so that everyone can understand how malware works and what it does to computer systems.

The research method is to use Static Analysis by testing and evaluating software without running the software by comparing various signatures using string and hashing mechanisms and Dynamic Analysis is testing and evaluating software when running. The results achieved are malware that has been weakened will be able to help explain in more detail, directly, and more easily explained so that the workings are easier to understand and can be shared with everyone who needs a practical/real-time explanation of how malware works.

Keywords: Reverse Engineering, Malware, Static Analysis, Dynamic Analysis

Introduction

1. Why to perform malware analysis and compile them in one database?

With increasingly advanced technological developments, this can be a benefit for humans who use it, but this can also be exploited by malicious actors to exploit weaknesses in the system by inserting malware which can result in leaks, damage or loss of data. Based on 2022 Global Attack Trends statistics (Conner, Sonicwall Cyber Threat Report 2022), malware attacks rose 2% with a total of 5.5 billion compared to the previous year. Of course, with the existence of various types of malware, antiviruses are also increasingly monitoring programs that are downloaded or installed on personal computers.

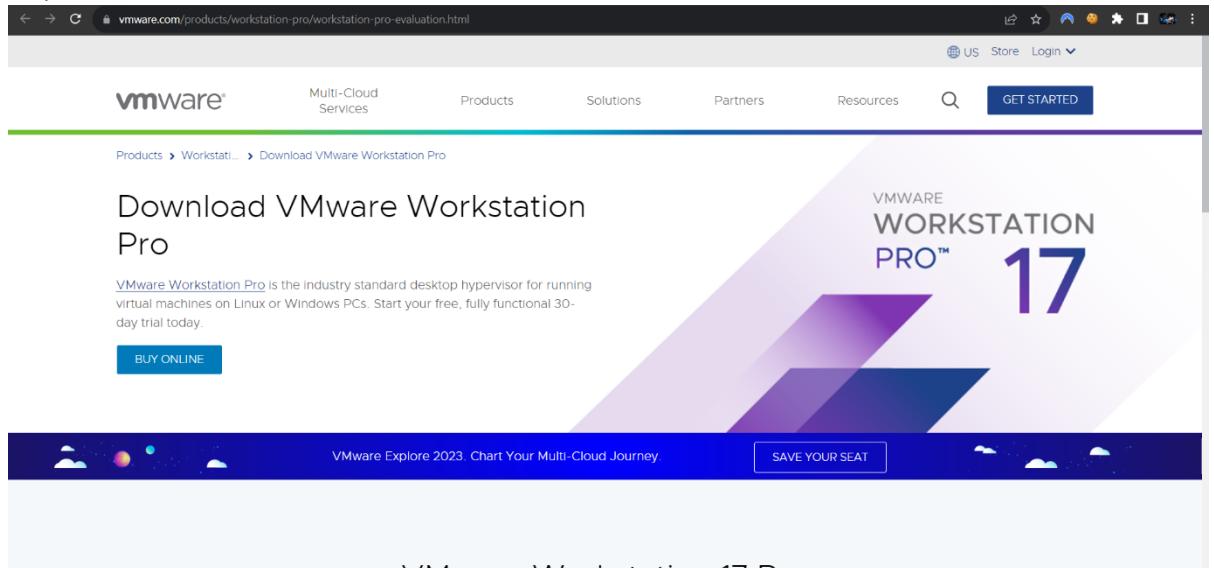
Antivirus performs techniques such as Signature-based analysis, Heuristic analysis, Behavior analysis, Cloud analysis and Sandbox analysis to identify malware (Zeltser, How antivirus software works: 4 detection techniques 2016 and Chu et al., How Anti-Virus Software Works 2000). However, sometimes there is malware that can be more difficult or even undetectable due to the use of obfuscation or encryption on the malicious function to be executed, disguising the malware's function as a genuine program and there is malware that can bypass sandbox detection by operating according to the conditions in which the malware is executed (Alzarooni, Core – aggregating the world's Open Access Research Papers 2012).

These things make it more difficult to identify new variants of malware if you only rely on antivirus software. Apart from that, quite a few false-positive notifications were found where the program was indeed useful and no malicious behavior was found but it was deleted and blocked by the antivirus, for example Windows update files where the code used was similar to malware even though the file was a genuine and important file. for windows updates. (Mishra, Finding and solving contradictions of false positives in virus scanning 2013). Based on the above, analyzing and disabling malware through reverse engineering can be a solution for false-positives and malware that bypasses antivirus detection because reverse engineering can help clarify and validate whether the function being carried out is part of existing malware (new variants). from existing malware), finding weak points/exploitation points of malware and providing information about how to handle it, informing about methods used to avoid malware detection by antivirus and clarifying the structure of how the malware itself works (Baker, Malware analysis: Steps & examples - crowdstrike 2023).

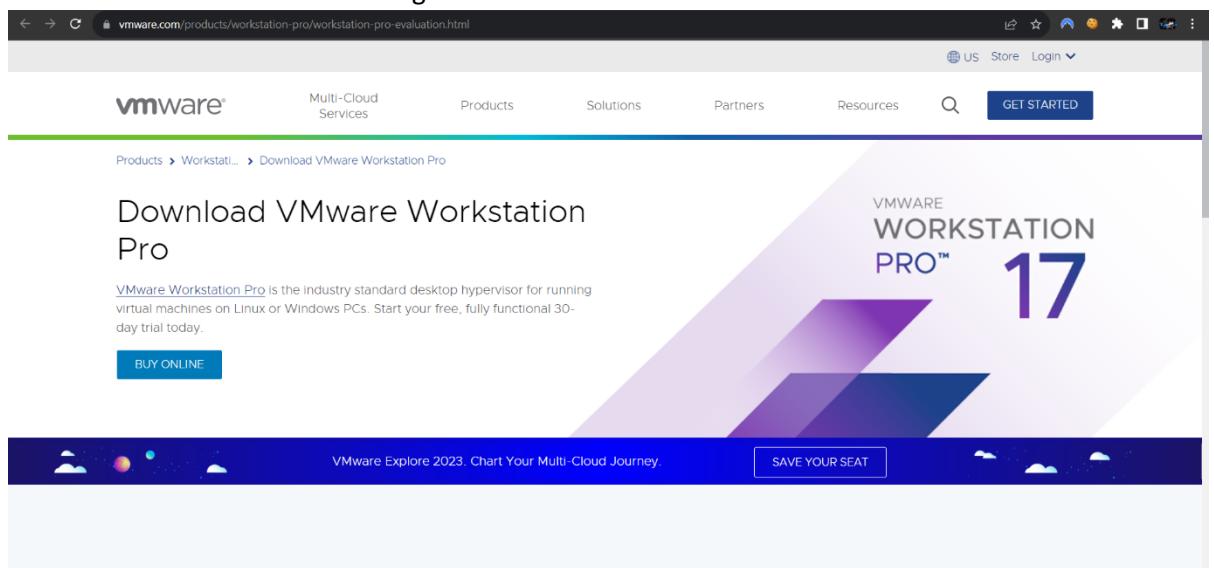
Compiled or distributed repositories/databases are available on Github.com/johnax0/OCMar as one of many ways to gain more knowledge if the time comes when you / enterprises need to use one of many software available but there are no reviews or has no reputation yet. The database can be used to give companies or individuals insight into the application which is treated as false-positive and if researchers give more malware example then it will be useful for people to find which malware is infecting their system and how to immobilize them.

2. How to set up a VM Ware to perform malware analysis?

- To install VMWare, you can go here <https://www.vmware.com/products/workstation-pro/workstation-pro-evaluation.html> and download the file needed. Afterwards, follow the steps needed.



VMware Workstation 17 Pro
Figure 1.1 VM Ware Website



VMware Workstation 17 Pro
Figure 1.2 VM Ware Download

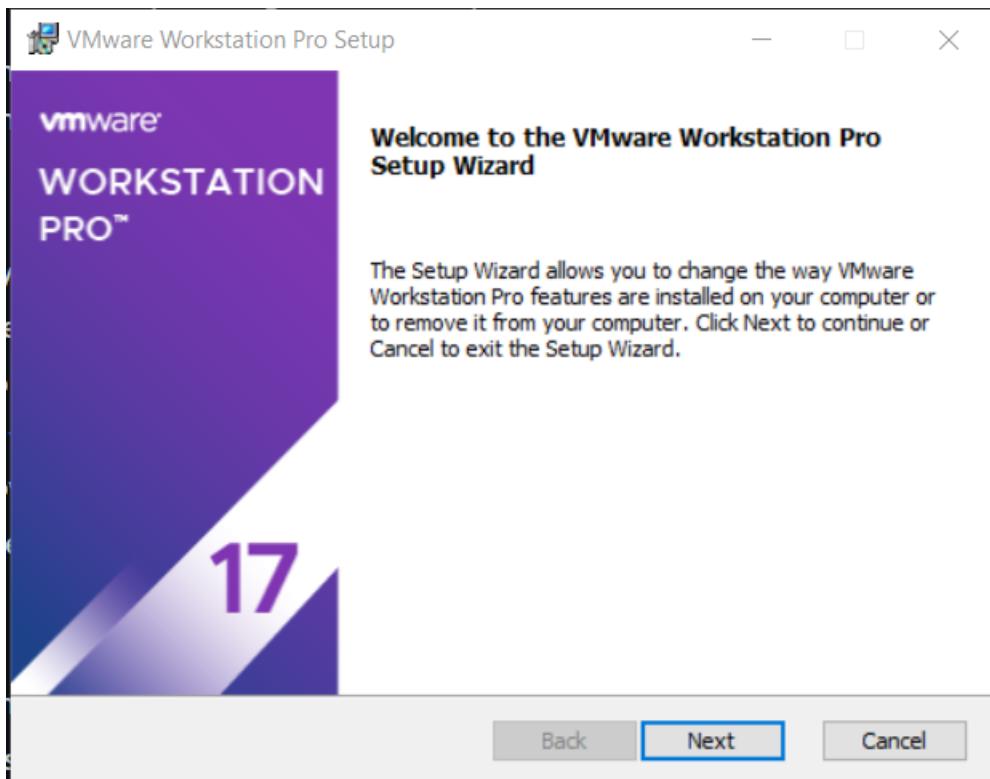


Figure 1.3 VM Ware Installation

- Prepare to install windows ISO File from here <https://support.microsoft.com/id-id/windows/create-an-iso-file-for-windows-10-38547366-1dcb-7af7-1726-9eb222d72705>.

Create an ISO file for Windows 10

Windows 10

Unlike in previous versions of Windows, you'll need to use the media creation tool to create an ISO file to install Windows 10. Make sure you have a license to install Windows 10, and then follow these steps:

1. On the [Windows 10 download page](#), download the media creation tool by selecting **Download tool now**, then run the tool.
2. In the tool, select **Create installation media (USB flash drive, DVD, or ISO) for another PC > Next**.
3. Select the language, architecture, and edition of Windows, you need and select **Next**.
4. Select **ISO file > Next**, and the tool will create your ISO file for you.
5. To use the ISO file as a backup for your PC or on another PC, burn it onto a DVD.

[Download the tool and learn more](#)

Figure 1.4 ISO Creation Media Installation Guide

- The windows ISO File can be downloaded by downloading a Media Creation Tool and Follow the steps to install it on USB/as a ISO File.

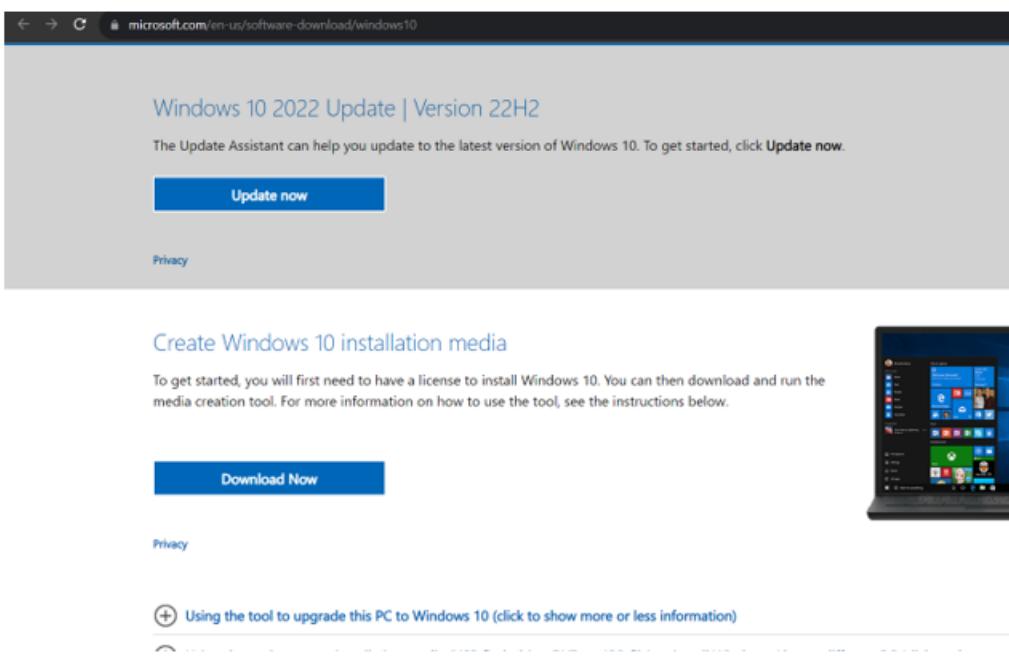


Figure 1.5 ISO Creation Media Download

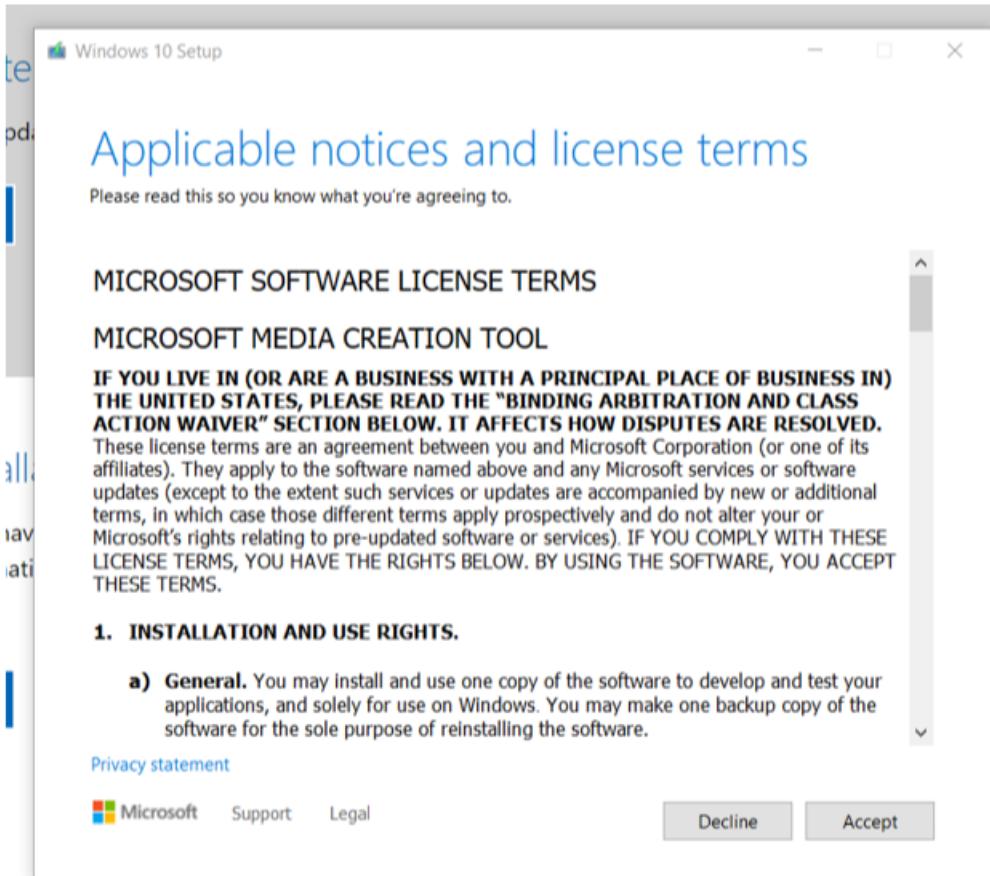


Figure 1.6 ISO Creation Media Installation

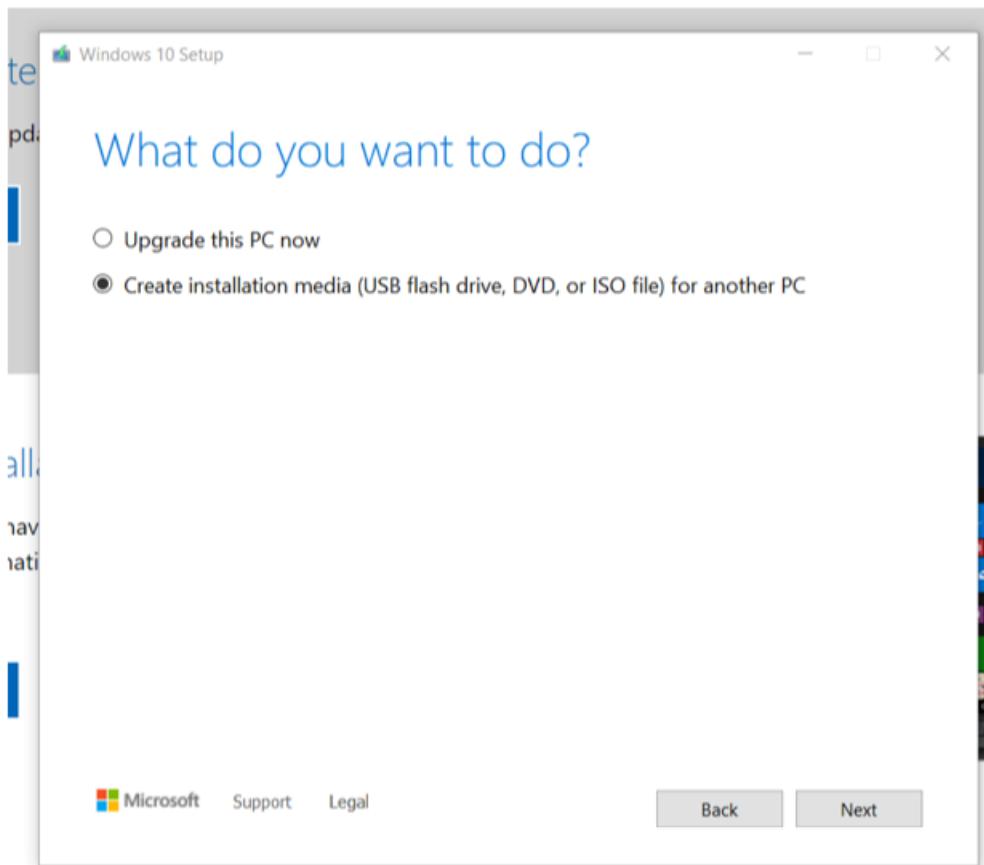


Figure 1.7 ISO Creation Media Installation Steps



Choose which media to use

If you want to install Windows 10 on another partition, you need to create and then run the media to install it.

USB flash drive

It needs to be at least 8 GB.

ISO file

You'll need to burn the ISO file to a DVD later.



Figure 1.8 ISO Creation Media Installation Steps – Choose ISO file

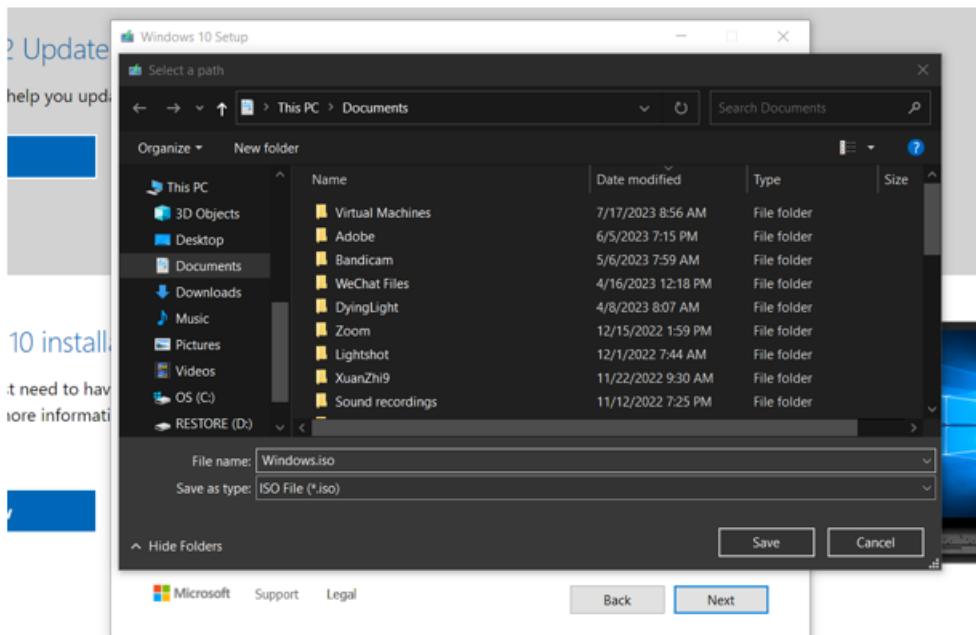


Figure 1.9 ISO Creation Media Installation Steps – Saves ISO File

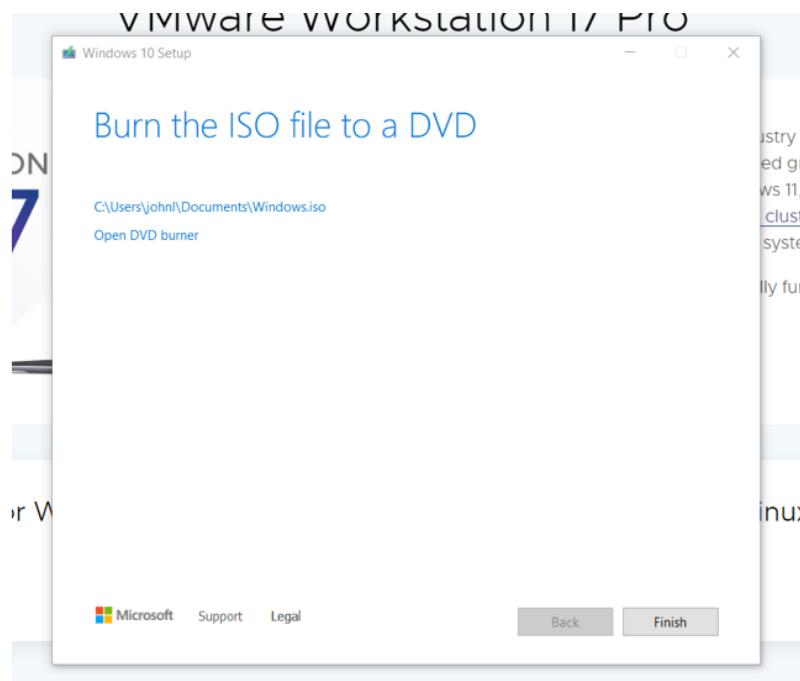


Figure 1.9 ISO Creation Completed

- Create a new device in VMWare that you just downloaded. Select the ISO File and configure the system, then complete the windows setup.

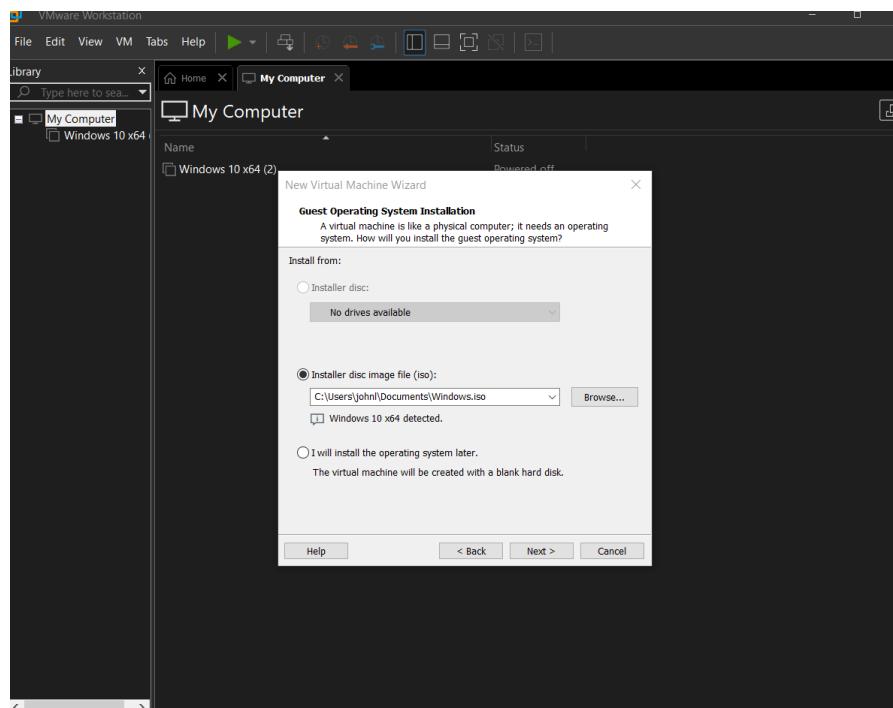


Figure 1.10 VM Ware choose ISO file

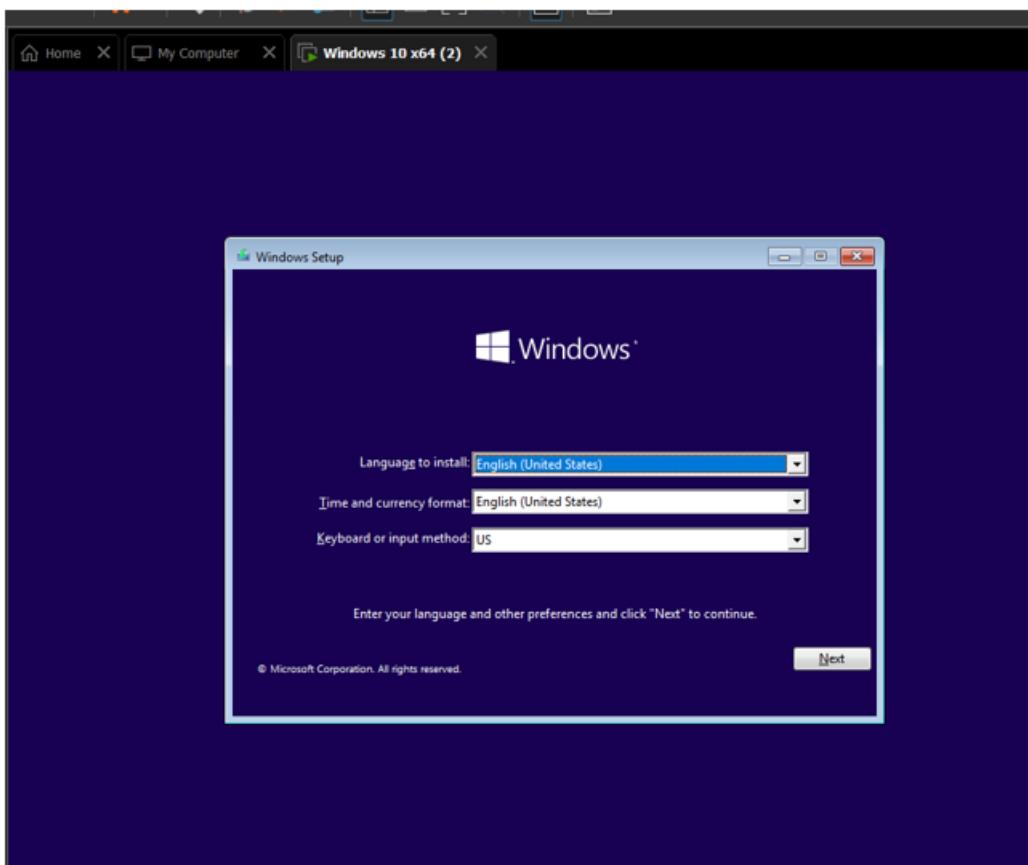


Figure 1.11 VM Ware Windows Installation

- Download Necessary Tools to analyze the malware such as: For Static Analysis: ExeinfoPE, PEStudio, APK Editor / apktool

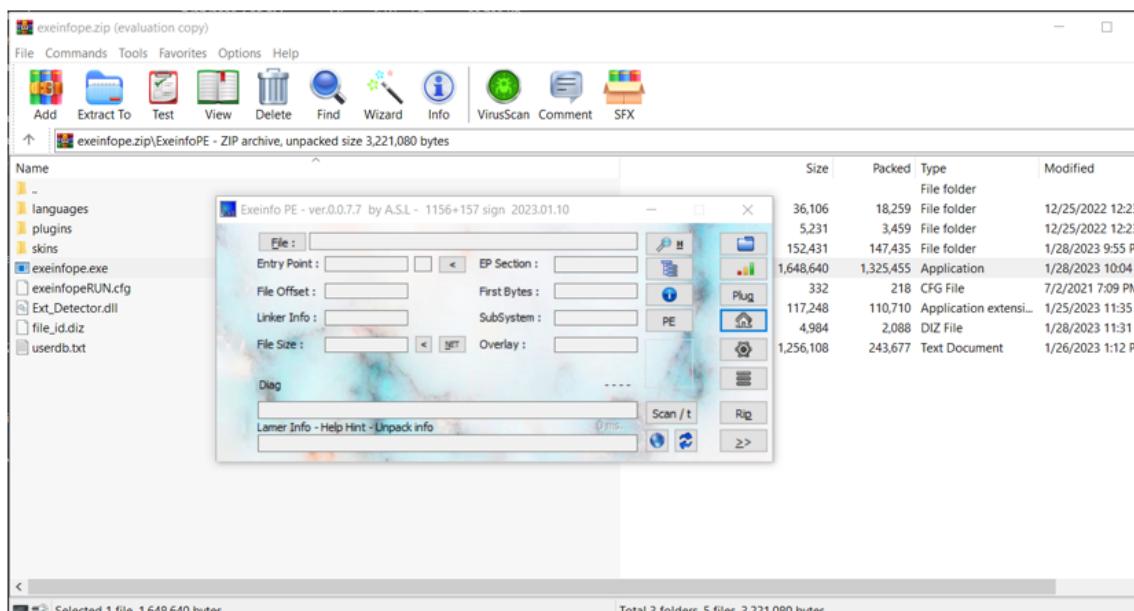


Figure 1.12 Static Analysis Tools Installation

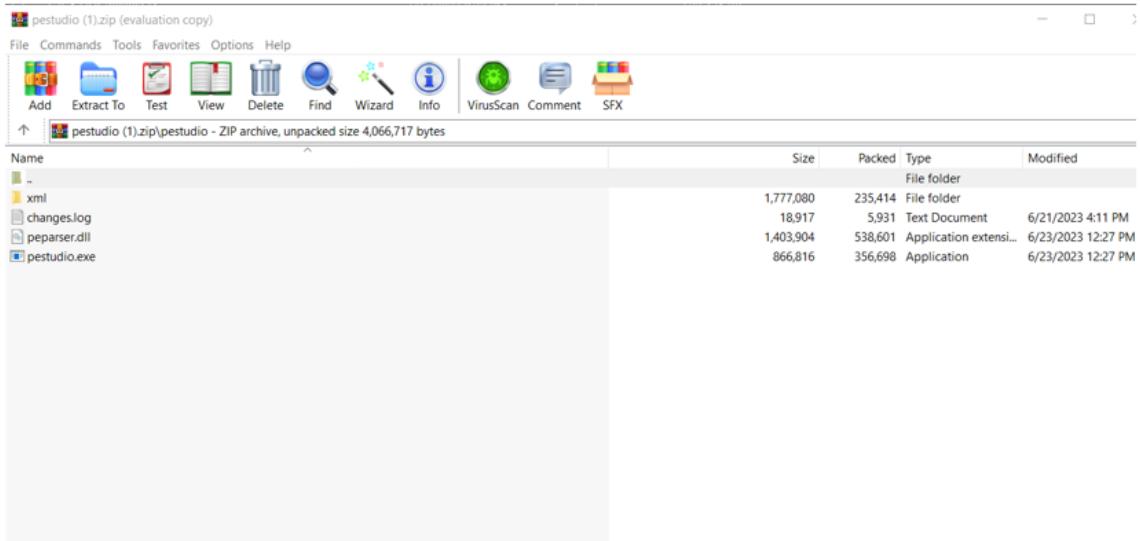


Figure 1.13 Static Analysis Tools Installation

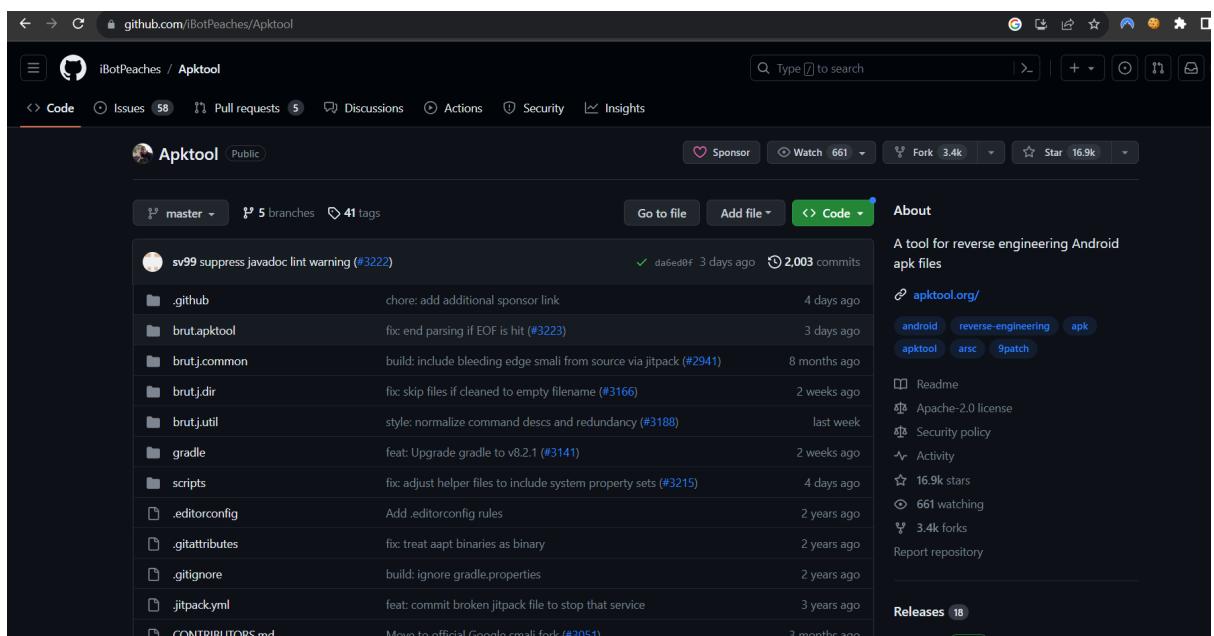


Figure 1.14 Static Analysis Tools Installation

- For Dynamic Analysis: Procmon, Procdot, Fakenet-ng

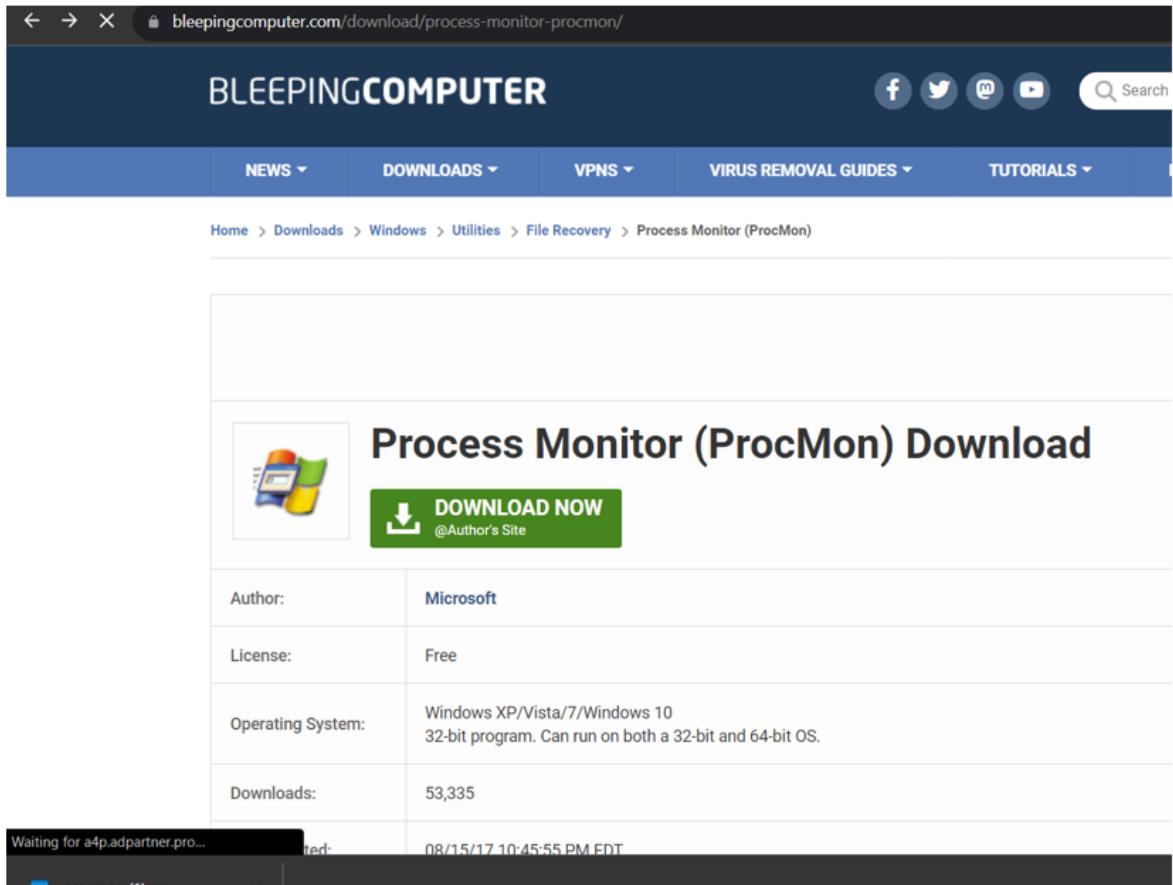


Figure 1.15 Dynamic Analysis Tools Installation

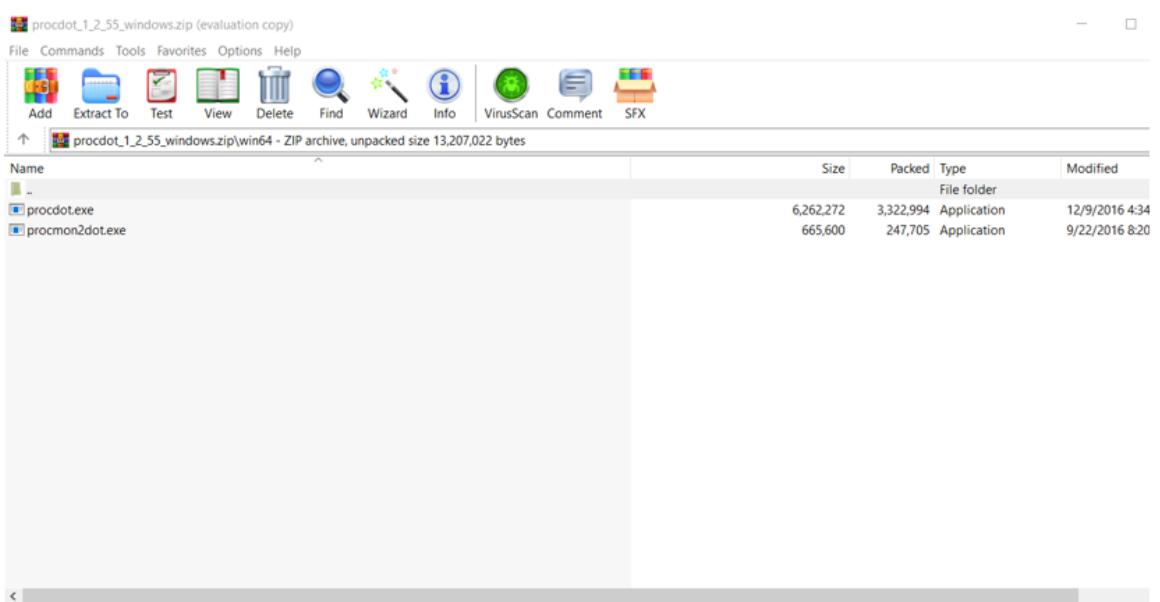


Figure 1.16 Dynamic Analysis Tools Installation

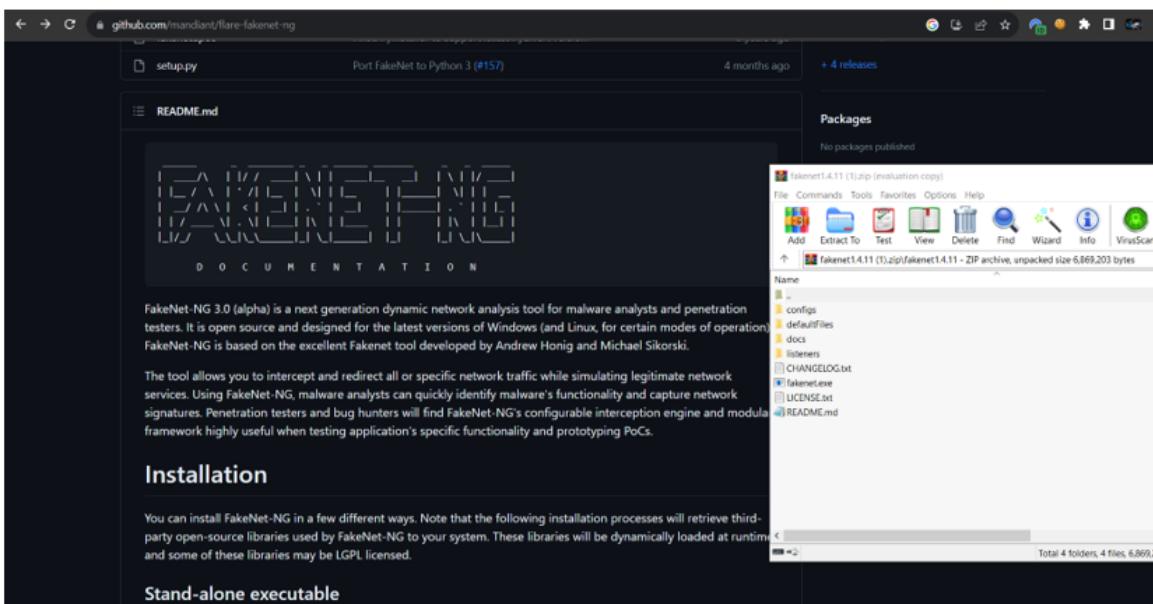


Figure 1.16 Dynamic Analysis Tools Installation

- For Reverse and Patching: Ghidra Disassembler, X64dbg / X32dbg

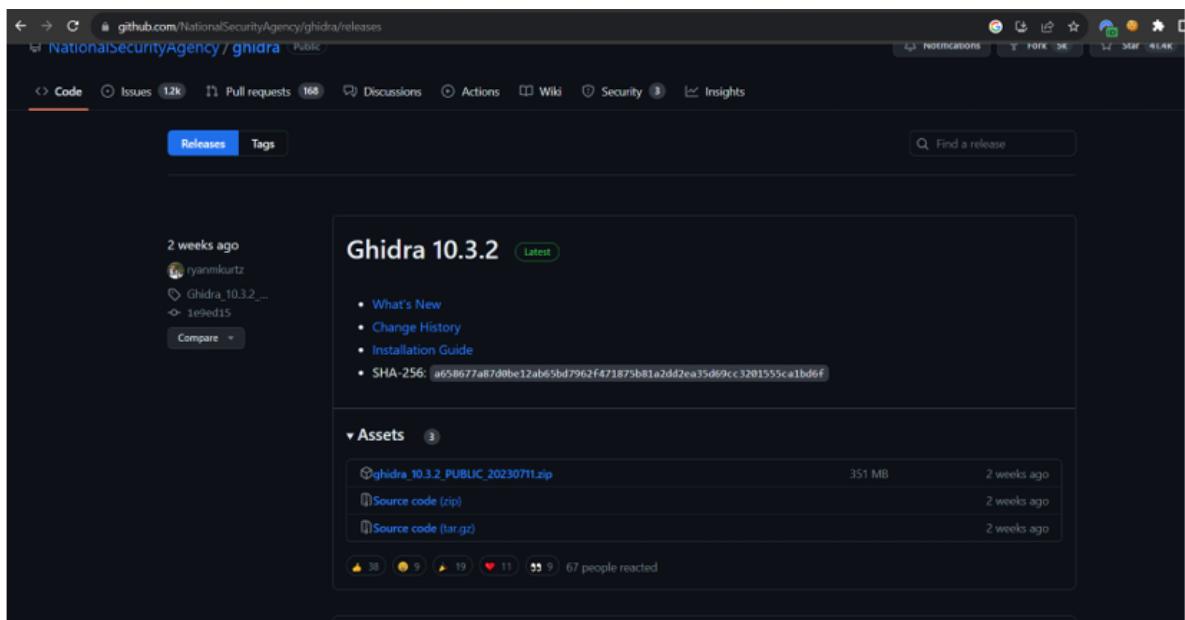


Figure 1.17 Reverse/Disassembly Analysis Tools Installation

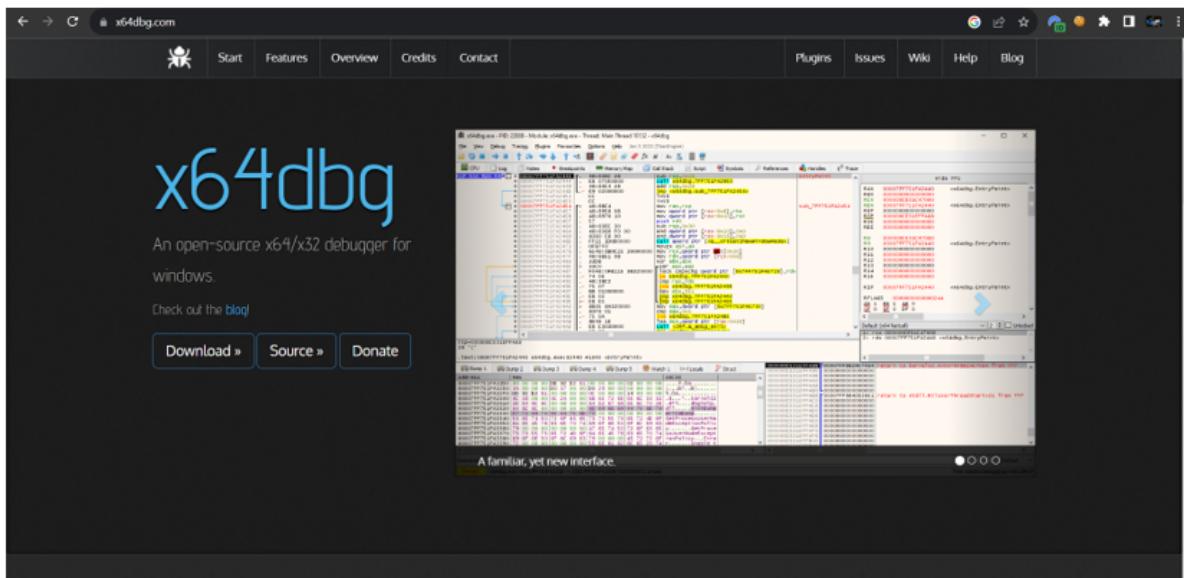


Figure 1.18 Reverse/Disassembly Analysis Tools Installation

- Ghidra Disassembler requires you to have Java 17 64-bit Runtime and Development Kit (JDK), there are few sources to install from:
 - o Free LTS by Adoptium Temurin (<https://adoptium.net/temurin/releases>)
 - o Free LTS by Amazon Coretto (<https://docs.aws.amazon.com/corretto/latest/corretto-17-ug/downloads-list.html>)

https://github.com/NationalSecurityAgency/ghidra/blob/10.3.2_build/GhidraDocs/InstallationGuide.html

NOTE: All 32-bit OS installations are now deprecated. Please contact the Ghidra team if you have a specific need.

Minimum Requirements

Hardware

- 4 GB RAM
- 1 GB storage (for installed Ghidra binaries)
- Dual monitors strongly suggested

Software

- Java 17 64-bit Runtime and Development Kit (JDK) (see [Java Notes](#))
 - Free long term support (LTS) versions of JDK 17 are provided by:
 - [Adoptium Temurin](#)
 - [Amazon Corretto](#)

[Back to Top](#)

Installing Ghidra

To install Ghidra, simply extract the Ghidra distribution file to the desired filesystem destination using any unzip program (built-in OS utilities, 7-Zip, WinZip, WinR/

Installation Notes

Figure 1.19 Reverse/Disassembly Analysis Tools Installation Guide

- Procdot, you will need WinDump/TCPDump and Graphviz to be able to create a visualization of the processes. You can download from here:

- WinDump : <http://www.winpcap.org/windump/install/default.htm>
- Graphviz : http://www.graphviz.org/download_windows.php On the first run, Procdot will ask you to locate the WinDump and Graphviz executables files.

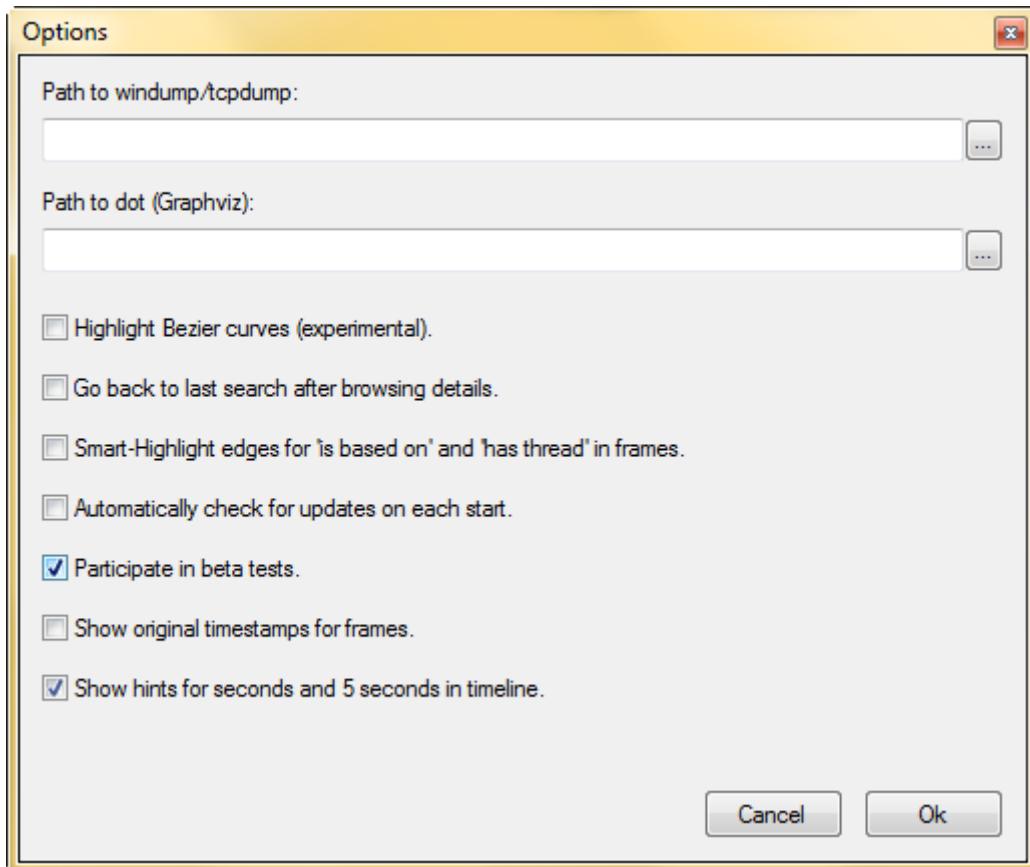


Figure 1.20 Reverse/Disassembly Analysis Tools Installation Guide

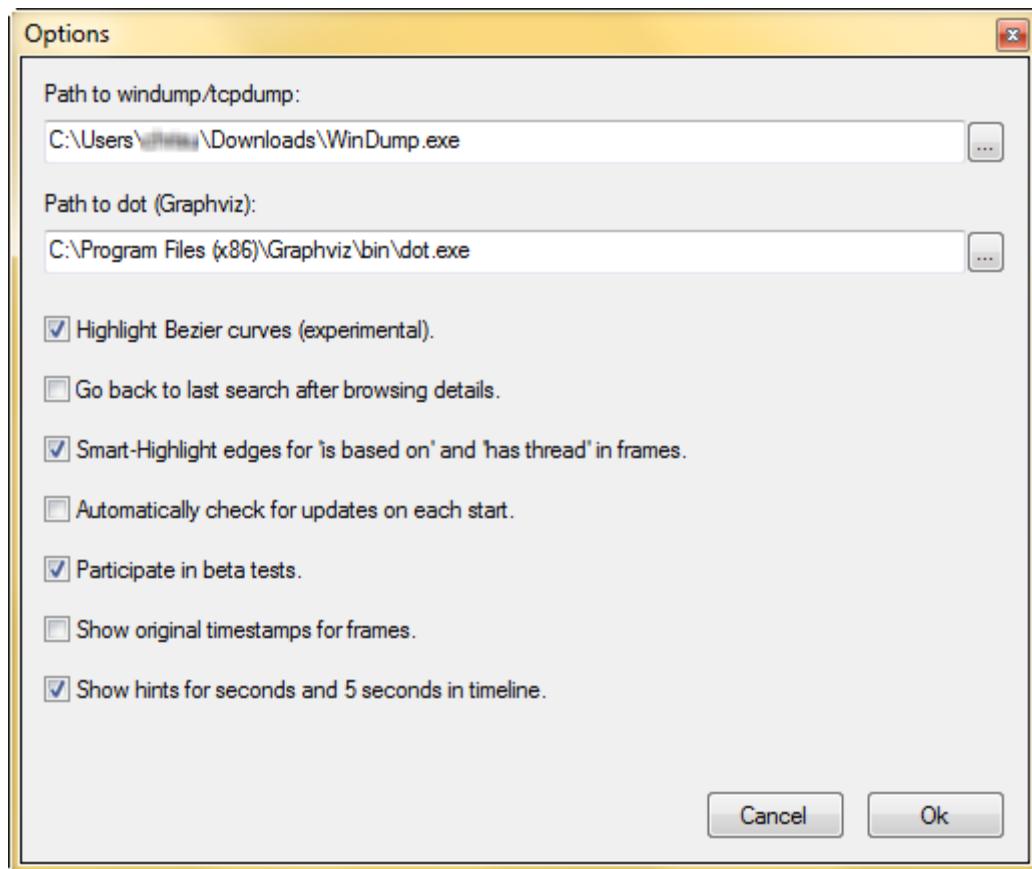


Figure 1.21 Reverse/Disassembly Analysis Tools Installation Guide

3. Step by step Instructions on How to Analyze (Windows Malware: financials.xls.exe)

- Open up VMWare, and disconnect/the internet just to be cautious of the malware. While you do so, the windows defender should be turned off to prevent any interference from it.

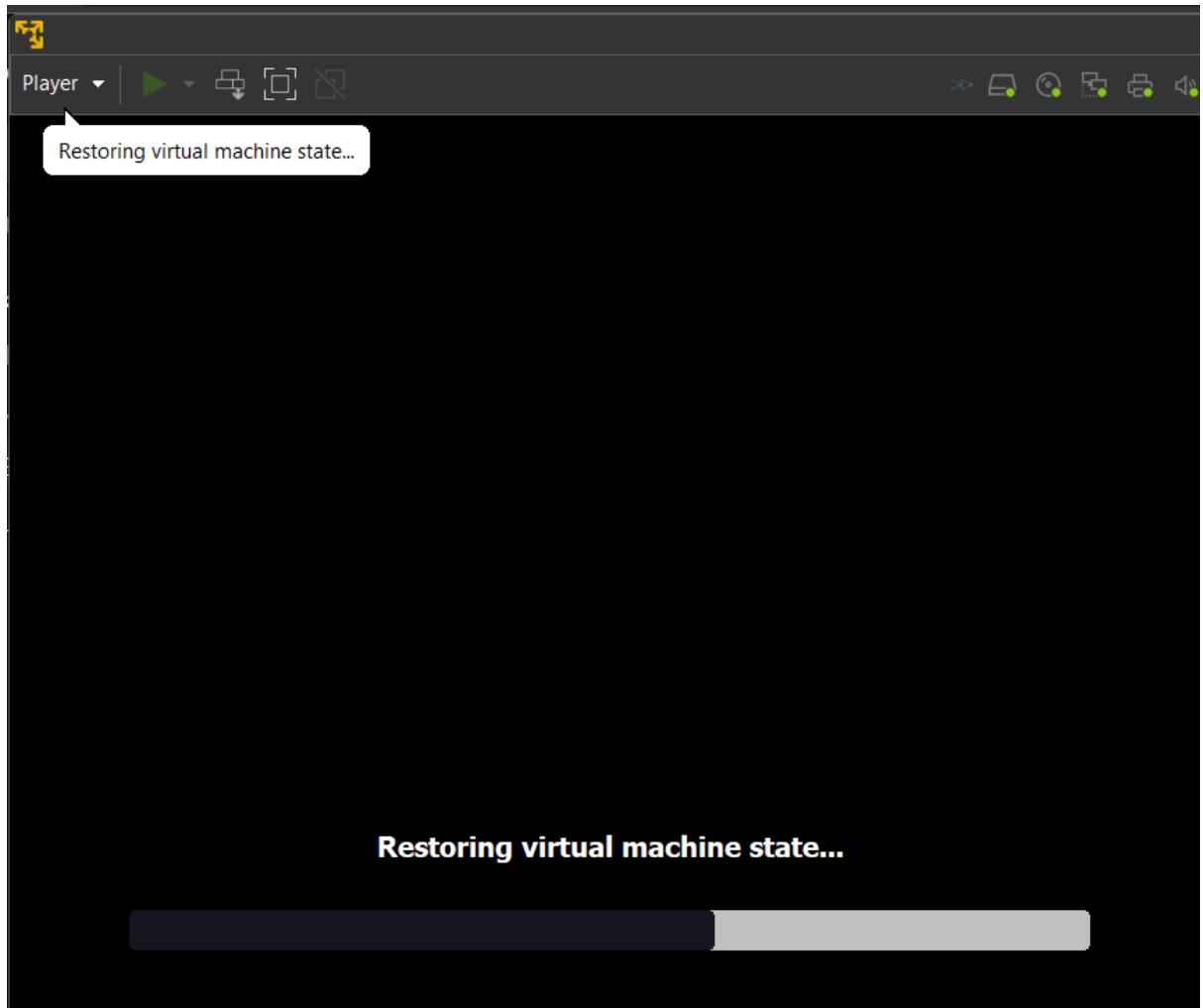


Figure 2.1 Opening VM Ware Windows

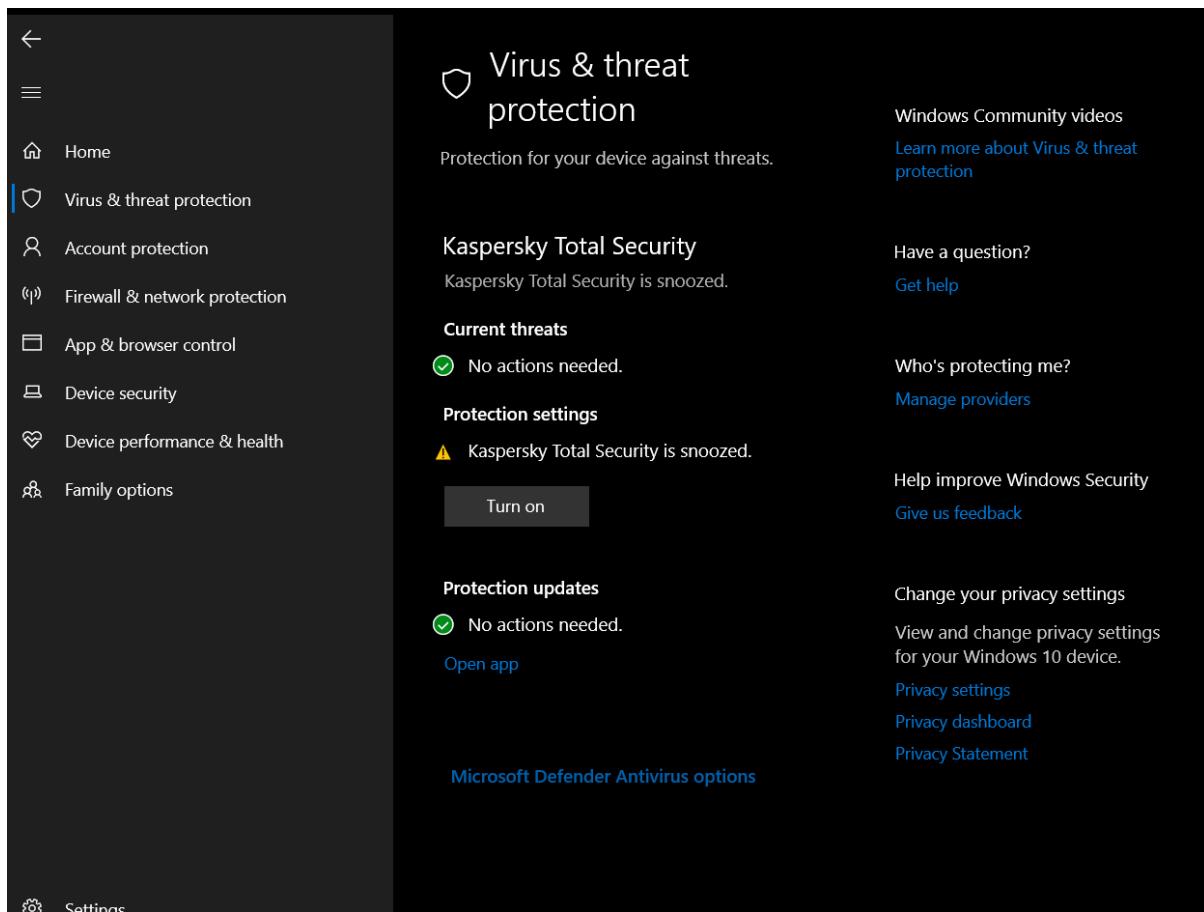


Figure 2.2 Turning Windows Defender Off

3.1 Static Analysis

Drag and drop the executables that you want to analyze inside of Exeinfope and PEStudio to analyze it further.

3.1.1 Exeinfope will give you an information if the malware is packed and the good part is, it also tells you how to unpack.

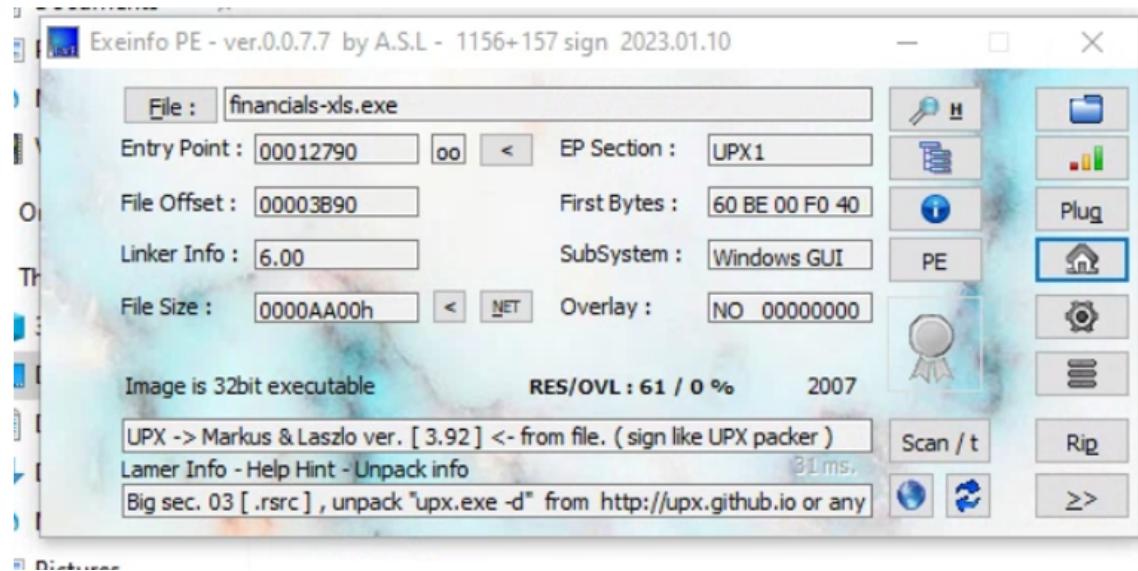


Figure 2.2 Use Exeinfo PE

```
C:\Users\malware_research\Downloads\upx-4.0.2-win64>upx -d -o upxed-malware.exe "C:\Users\malware_research\Desktop\Malware\finale\financials-xls.exe"
Ultimate Packer for eXecutables
Copyright (C) 1996 - 2023
UPX 4.0.2      Markus Oberhumer, Laszlo Molnar & John Reiser   Jan 30th 2023
File size      Ratio      Format      Name
-----      -----
57344 <-    43520    75.89%    win32/pe    upxed-malware.exe

Unpacked 1 file.

C:\Users\malware_research\Downloads\upx-4.0.2-win64>
C:\Users\malware_research\Downloads\upx-4.0.2-win64>
```

Figure 2.3 Use Exeinfo PE to unpack with recommended tools

3.1.2 PEStudio will tell you indications on what functions/api that will be used alongside the windows libraries file.

pestudio 9.48 - Malware Initial Assessment - www.wimitor.com - [c:\users\malware research\Desktop\malware\upx-ed-malware.exe]

file	settings	about						
imports (85)	flag (18)	first-thunk-original (INT)	first-thunk (IAT)	hint	group (11)	technique (10)	type (1)	
GetDesktopWindow	x	n/a	0x00006736	0 (0x0000)	windowing	-	implicit	
RegDeleteKeyA	x	n/a	0x00006650	0 (0x0000)	registry	Data Destruction	implicit	
RegSetValueExA	x	n/a	0x00006660	0 (0x0000)	registry	Modify Registry	implicit	
RegCreateKeyExA	x	n/a	0x00006682	0 (0x0000)	registry	Data Destruction	implicit	
17 (rcvfrom)	x	n/a	0x00000011	0 (0x0000)	network	-	implicit	
4 (connect)	x	n/a	0x00000014	0 (0x0000)	network	-	implicit	
23 (socket)	x	n/a	0x00000017	0 (0x0000)	network	-	implicit	
115 (WSASStartup)	x	n/a	0x00000073	0 (0x0000)	network	-	implicit	
10 (inet_addr)	x	n/a	0x00000004	0 (0x0000)	network	-	implicit	
9 (htons)	x	n/a	0x00000009	0 (0x0000)	network	-	implicit	
20 (sendto)	x	n/a	0x00000014	0 (0x0000)	network	-	implicit	
WriteFile	x	n/a	0x0000633A	0 (0x0000)	file	-	implicit	
DeleteFileA	x	n/a	0x00006350	0 (0x0000)	file	Data Destruction	implicit	
GetEnvironmentStringsW	x	n/a	0x00006434	0 (0x0000)	execution	-	implicit	
GetEnvironmentStrings	x	n/a	0x0000644C	0 (0x0000)	execution	-	implicit	
TerminateProcess	x	n/a	0x000064C6	0 (0x0000)	execution	-	implicit	
WinExec	x	n/a	0x0000654A	0 (0x0000)	execution	Execution through A...	implicit	
ShowWindow	-	n/a	0x0000670A	0 (0x0000)	windowing	-	implicit	
IsWindowVisible	-	n/a	0x00006716	0 (0x0000)	windowing	-	implicit	
UpdateWindow	-	n/a	0x00006728	0 (0x0000)	windowing	-	implicit	
ShowMessageA	-	n/a	0x0000673C	0 (0x0000)	windowing	Process Injection	implicit	
FindWindowExA	-	n/a	0x0000676A	0 (0x0000)	windowing	Process Injection	implicit	
FindWindowA	-	n/a	0x0000677A	0 (0x0000)	windowing	Process Injection	implicit	
GetMessageA	-	n/a	0x000067A8	0 (0x0000)	windowing	-	implicit	
TranslateMessage	-	n/a	0x000067B6	0 (0x0000)	windowing	-	implicit	
DispatchMessageA	-	n/a	0x000067C8	0 (0x0000)	windowing	-	implicit	
GetFocus	-	n/a	0x000067D4	0 (0x0000)	windowing	-	implicit	
CreateEventA	-	n/a	0x000068A2	0 (0x0000)	synchronization	-	implicit	
LoadIcon	-	n/a	0x0000679C	0 (0x0000)	resource	-	implicit	
RegDeleteA	x	n/a	0x00006694	0 (0x0000)	registry	Query Registry	implicit	
RegQueryValueExA	x	n/a	0x0000669E	0 (0x0000)	registry	Query Registry	implicit	

sha256: 726A072434E751B2781D49F4F85EC213B60D0F0EF6AA6377D5D55FAD0171E7DE9

cpu: 32-bit file-type: executable subsystem: GUI entry-point: 0x00003510

Figure 2.4 Use PEStudio to get insight of malware and suspicious functions

pestudio 9.48 - Malware Initial Assessment - www.wimitor.com - [c:\users\malware research\Desktop\malware\upx-ed-malware.exe]

file	settings	about						
imports (11)	label (106)	group (11)	technique (11)	value (2857)				
-	import	windowing	-	ShowWindow				
-	import	windowing	-	IsWindowVisible				
-	import	windowing	-	UpdateWindow				
x	import	windowing	-	GetDesktopWindow				
-	import	windowing	Process Injection	Process Injection				
-	import	windowing	Process Injection	ShowMessage				
-	import	windowing	Process Injection	FindWindowEx				
-	import	windowing	-	FindWindow				
-	import	windowing	-	GetMessage				
-	import	windowing	-	TranslateMessage				
-	import	windowing	-	DispatchMessage				
-	import	windowing	-	GetFocus				
-	import	windowing	-	GetLastActivePopUp				
-	import	windowing	-	GetActiveWindow				
-	import	synchronization	-	CreateEvent				
-	import	resource	-	LoadIcon				
x	import	registry	Data Destruction	RegDeleteKey				
x	import	registry	Modify Registry	RegSetValueExA				
x	import	registry	Data Destruction	RegDeleteValue				
x	import	registry	Modify Registry	RegCreateKeyEx				
-	import	registry	Query Registry	RegQueryValueExA				
-	import	registry	-	RegCloseKey				
-	import	reconnaissance	File and Directory Disc...	GetWindowsDirectory				
-	import	reconnaissance	-	GetStartupsInfo				
-	utility	network	-	GET /download.php?&advuid=00000717&us...%0d%0a HTTP/1.0\r\nHost: download.braev...				
-	utility	network	-	GET http://download.braeventry.com/download.php?&advuid=00000717&us...%0d%0a...				
-	file	network	-	WSOCK32.dll				
-	-	network	-	ProxyServer				
-	-	network	-	ProxyEnable				
-	-	memory	-	GetStringType				
-	-	memory	-	GetStringType				
-	-	memory	Process Injection	VirtualAlloc				

sha256: 726A072434E751B2781D49F4F85EC213B60D0F0EF6AA6377D5D55FAD0171E7DE9

cpu: 32-bit file-type: executable subsystem: GUI entry-point: 0x00003510

Figure 2.5 Use PEStudio and suspicious functions are found

From the static analysis, it has been found out that the malware uses double extension to perhaps mislead the victim into thinking that it was an XLS or Excel file instead of an EXE file. While running, the malware will be downloading another malware on malicious IP and URL. These types of malware usually works when the victim is not paying attention to application type or when they are distracted. By downloading another program which is also another malware, the computer will later become more infected.

3.2 Dynamic Analysis

3.2.1 Fakenet-ng

Before trying to run the malware it is recommended to start fakenet-ng incase the malware is trying to connect to an external network/IP.

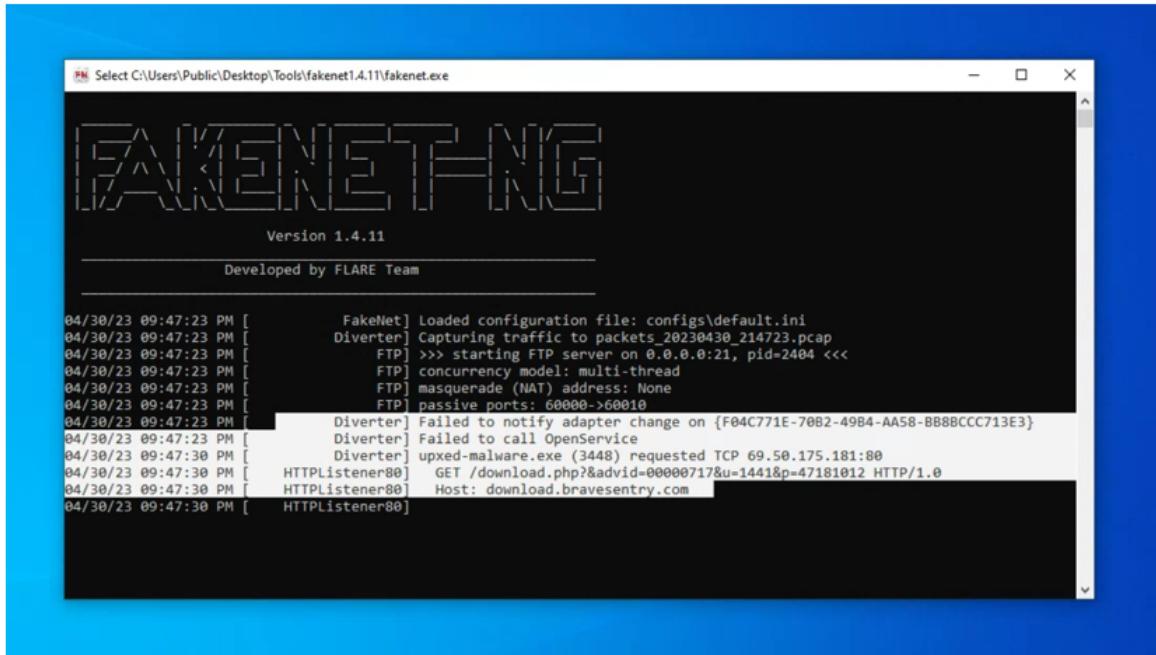


Figure 2.6 Use fakenet to intercept suspicious request

From fakenet, it sent a request to a website called bravesentry and it is another malware which will be used to act as a regular program but inside it will execute malicious payload.

3.2.2 Procmon can be used to capture and log the activity of the malware.

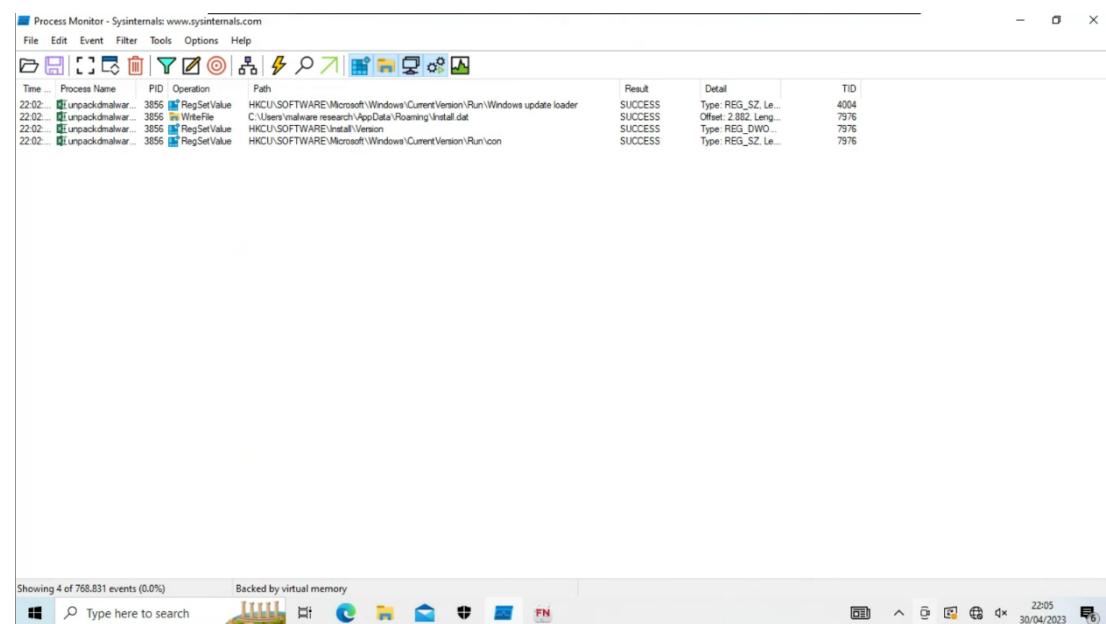


Figure 2.7 Use procmon to analyze processes

Filter can be applied also into the captured logs, those filter are ProcessCreate, WriteFile, SetDispositionInformationFile, TCP and UDP, & RegSetValue

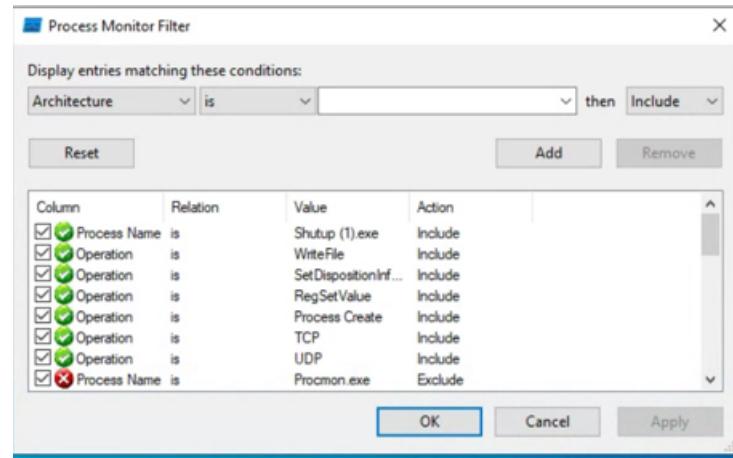


Figure 2.8 Use procmon to filter suspicious processes

ProcessCreate can show the malware is creating a new process when it is executed, while WriteFile and SetDispositionInformationFile is tracking what file or temporary files are created. RegSetValue also keep tracks of the value of registry that are being set. TCP and UDP are to logs if the malware is requesting a request to external IP/Websites. Afterwards, Procmon can create/export the logs using CSV so that Procdot can read them.

3.2.3 Procdot

While using Procdot, you can use the CSV file that you got from Procmon and insert them. Pick the process of the malware that was running on the device and hit refresh to visualize the process.

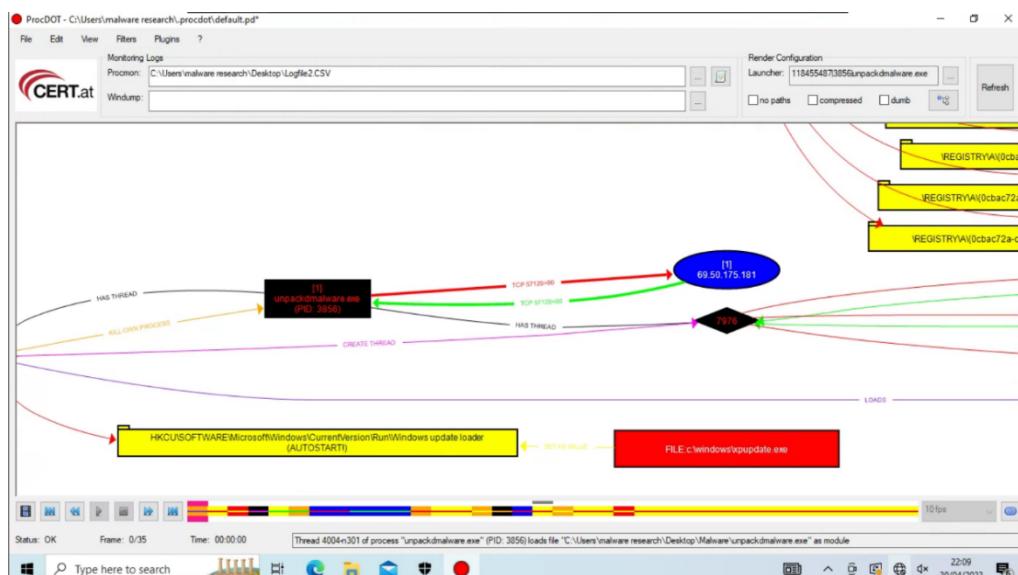


Figure 2.9 Use procdot to visualize suspicious processes

3.3 Disassembly and Patching

3.3.1 Ghidra, you can use it to analyze the executables. It will analyze the imports such as functions and necessary DLL file that are being used also being able to decompile some part of the malware is a nice thing to have.

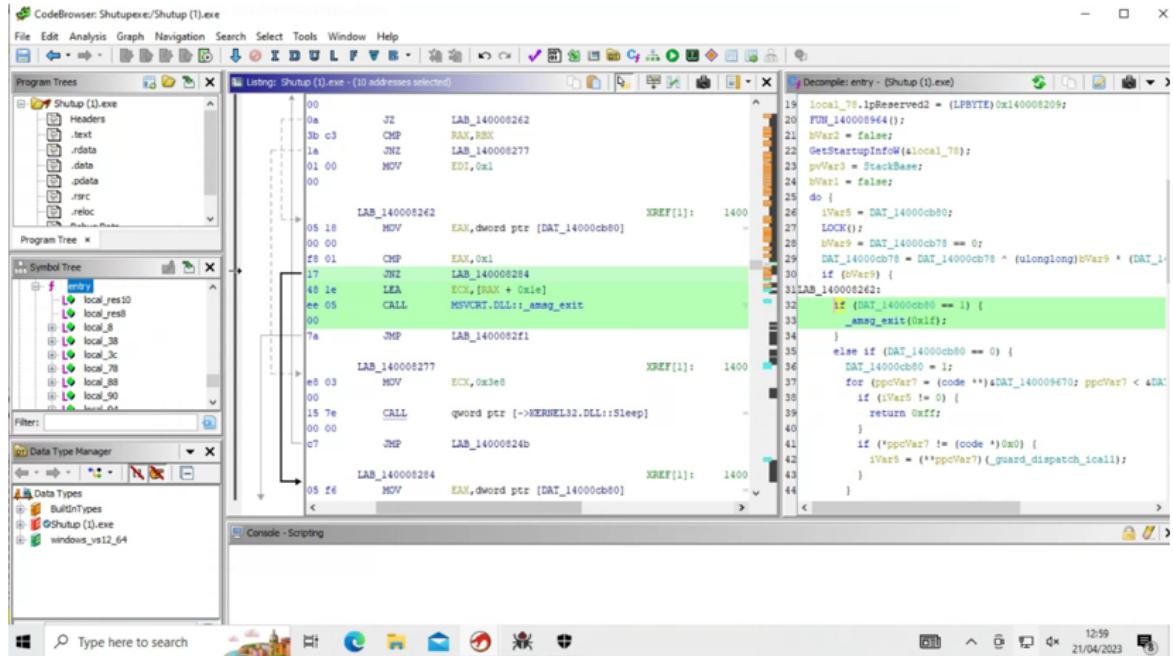


Figure 2.10 Use ghidra to find killswitch or keyword

By doing decompile on the code, you can analyze what conditions does it have to run certain functions and/or finds out what is the killswitch.

3.3.2 X64dbg or X32dbg

Finally, after you have got the killswitch and analyze the conditions you can import/open the executables in X64dbg or X32dbg to patch that specific functions. You can either change the value like changing from 1 to a 0 or going with the NOP on functions that are marked as suspicious/dangerous.

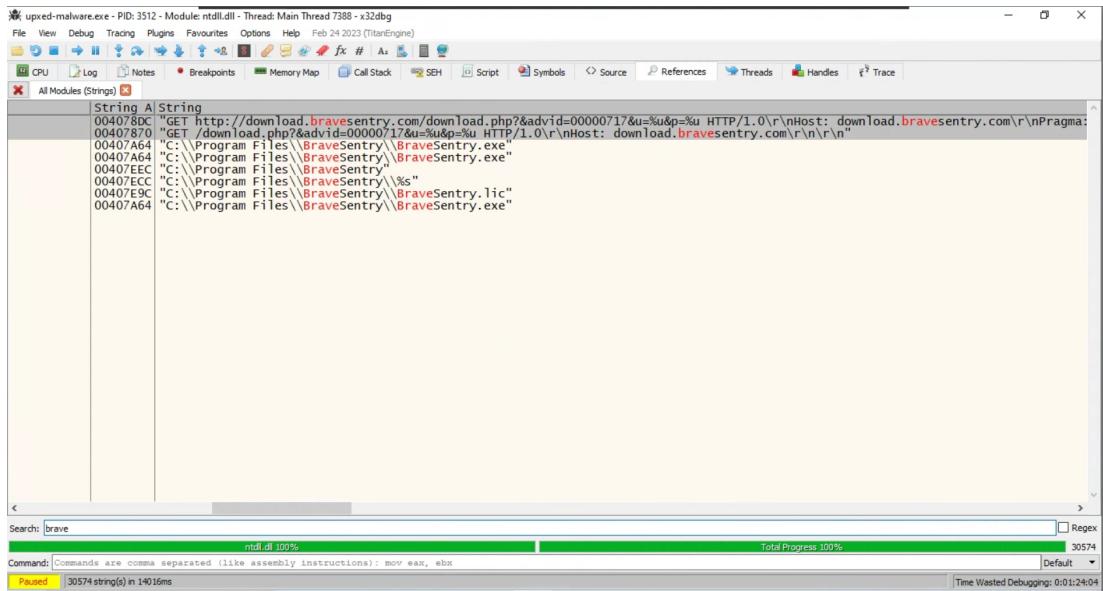


Figure 2.11 Use X64dbg or X32dbg to patch

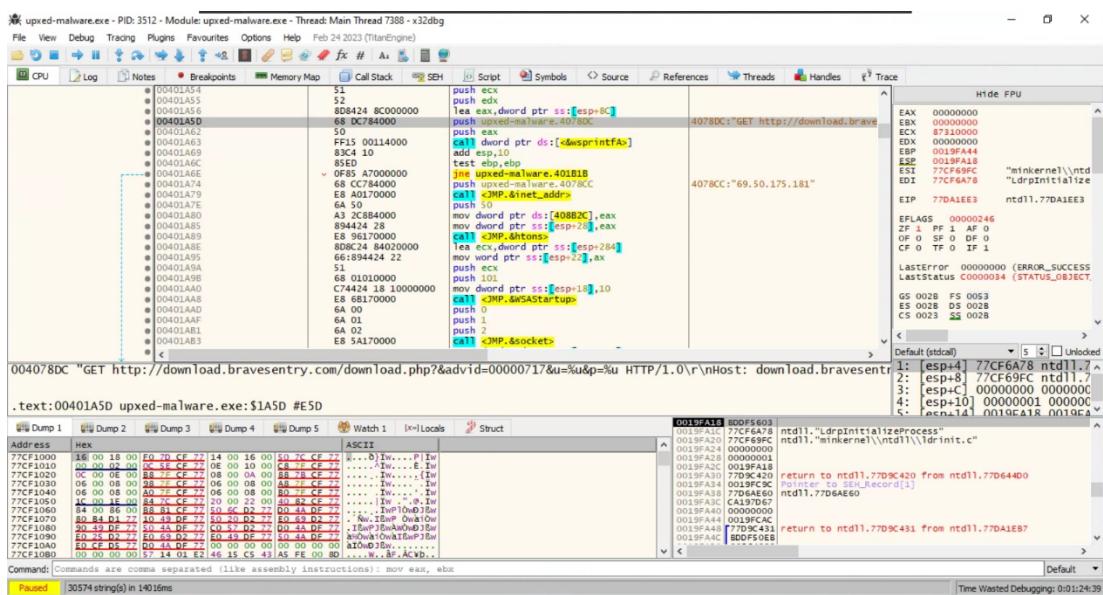


Figure 2.12 Use X64dbg or X32dbg to patch functions

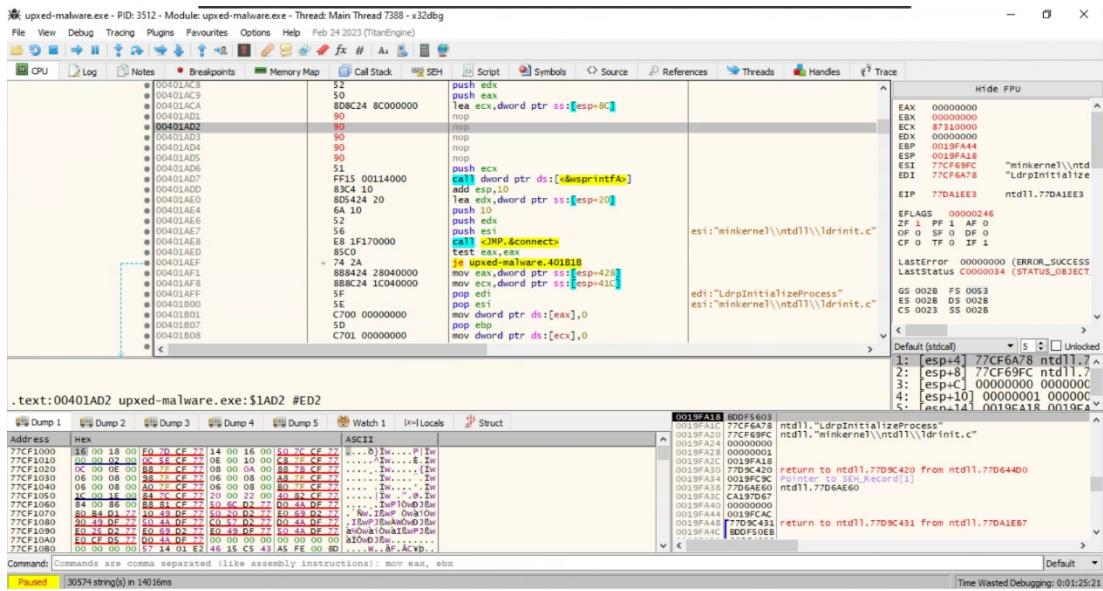


Figure 2.11 Malicious functions got patched

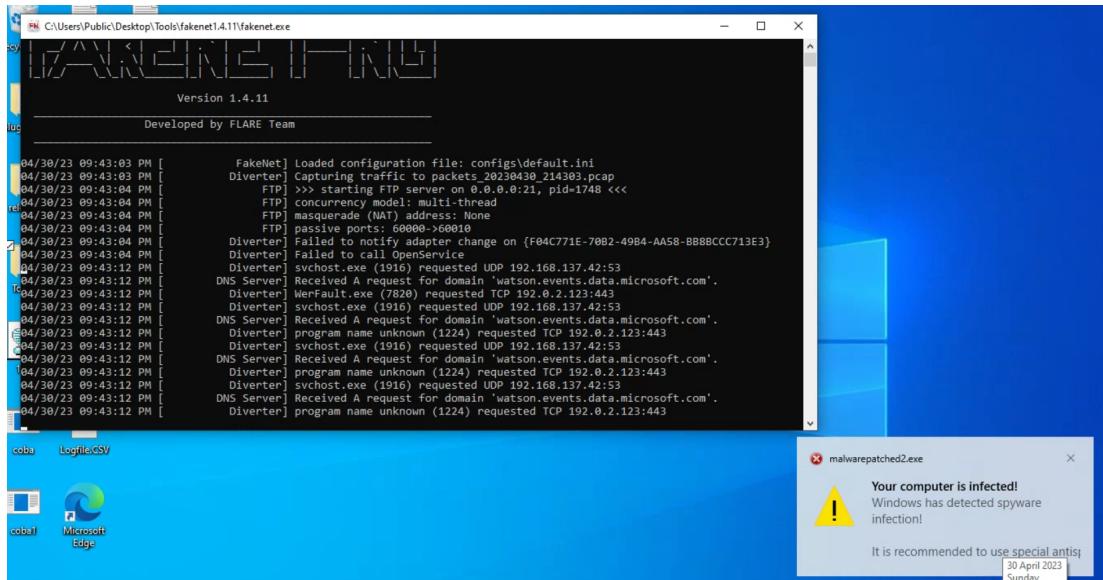


Figure 2.12 Malware no longer send request to download another malware

After doing a patching, the malware has no ability to request or download its other payload to infect the computer any further therefore, the malware itself is rendered useless. This can also happen by changing the value a function to NOP to prevent the whole malware being executed.