

# Black Energy Rootkit - Analysis

John Tolomay

<b>Static Analysis.....</b>	<b>2</b>
PeStudio.....	2
Capa.....	3
Cutter.....	4
<b>Dynamic Analysis.....</b>	<b>5</b>
Process Monitor.....	5
<b>Conclusion.....</b>	<b>6</b>

Filename: rootkit.ex1

Rootkits are designed to gain access to a computer or device without being detected. Their goal is to give cybercriminals full access to infected machines remotely while trying to blend in with the machine's system software.

## Static Analysis

### PeStudio

names	
file	c:\users\administrator\desktop\rootkit.ex1
debug	n/a
export	n/a
<u>version &gt; original-file-name</u>	notepad.exe
<u>manifest</u>	(C) Microsoft FullCopy
.NET > module	n/a
certificate > program-name	n/a

The malware uses notepad.exe for some task that is unknown right now.

flag (8)	label (37)	group (8)	value (2321)
-	import	windowing	GetMessagePos
-	-	windowing	FindWindow
-	-	windowing	FindWindowEx
-	-	resource	LoadCursor
x	-	reconnaissance	SearchPath
-	import	reconnaissance	GetTickCount
-	import	reconnaissance	IsDebuggerPresent
-	-	reconnaissance	GetDiskFreeSpace
x	import	memory	VirtualAlloc
x	import	memory	VirtualProtect
-	import	memory	GlobalFree
-	import	memory	VirtualFree
x	import	file	UnmapViewOfFile
-	import	file	CompareFileTime
-	import	file	GetFileSize
x	import	execution	GetCurrentThread
x	import	execution	SetProcessAffinityMask
x	-	execution	ZwSetInformationThread
x	-	execution	SleepEx
-	import	execution	PostQuitMessage
-	import	execution	ExitProcess
-	import	execution	GetExitCodeProcess
-	-	execution	Sleep
-	import	dynamic-library	GetProcAddress
-	-	dynamic-library	GetModuleHandle
-	-	dynamic-library	LoadLibrary
-	import	diagnostic	GetLastError

List of strings used by the malware, sorted by group so we can see what functions are used for what tasks. Something that stands out is the SleepEx function. This function temporarily stops a

thread for an amount of time, or until a certain condition is met. This could be used by the malware to delay its own execution. By delaying execution, the malware is more likely to go undetected and without triggering any security mechanisms.

<a href="#">section:.dKVU</a>	-	-	-	OLLYDBG
<a href="#">section:.dKVU</a>	-	-	-	Debugger status:
<a href="#">section:.dKVU</a>	-	-	-	Debugger status:
<a href="#">section:.dKVU</a>	-	-	-	Debugger not found!
<a href="#">section:.dKVU</a>	-	-	-	Debugger found!

These strings found in the binary are interesting to note. OLLYDBG is a debugger commonly used for malware analysis, specifically binary code. This malware is searching to see if OLLYDBG is being used on the machine to detect if it is being analyzed. If it is found, the malware will likely change its behavior or terminate itself.

characteristics	0xC0000020	0xE0000020	0xE0000020
write	x	x	x
execute	-	x	x
share	-	-	-
self-modifying	-	x	x
virtual	-	-	-

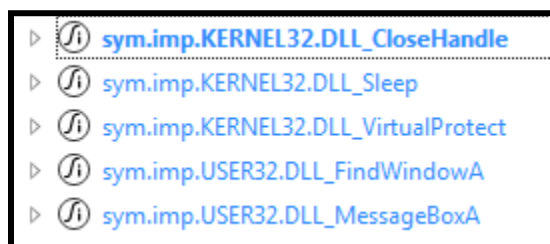
These are the characteristics of the malicious file. This data shows that the malware writes to files, executes programs, and has a self-modifying capability. This self-modification is because when it looks for analysis tools like OLLYDBG that signal it is being analyzed, it can change its behavior to not detonate the payload or even terminate itself.

## Capa

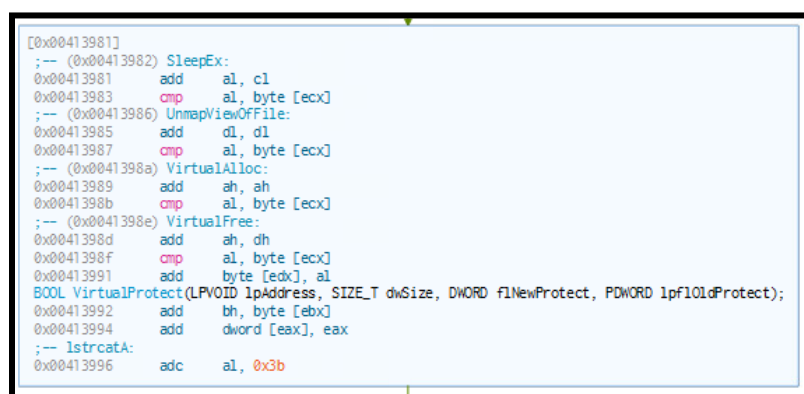
md5	9219e2cfcc64ccde2d8de507538b9991
sha1	181e59600d057dc6b31a3b19d7f4f75301a3425e
sha256	5af3fd53aea5e008d8725c720ea0290e2e0cd485d8a953053ccf02e5e81a94a0
os	windows
format	pe
arch	i386
path	rootkit.exe
ATT&CK Tactic	ATT&CK Technique
EXECUTION	Shared Modules T1129
MBC Objective	MBC Behavior
DISCOVERY	Analysis Tool Discovery::Process detection [B0013.001]
CAPABILITY	NAMESPACE
reference analysis tools strings	anti-analysis
link function at runtime on Windows	linking/runtime-linking

Based on this result, it seems like execution is the malware's main capability. It achieves its goals via shared modules such as libraries and DLLs, which has been common among the malware previously analyzed. The capability "reference analysis tools strings" comes from the OLLYDBG string that was found in PeStudio, meaning that this malware is trying to avoid being analyzed.

## Cutter



List of functions used by the imported DLLs. VirtualProtect is a function that changes the access permissions in a region of address space (read, write, execute). The malware is likely using this function to make its own code executable or giving write permissions on a file.




This sequence of function calls is interesting to note. It starts off by delaying its execution using SleepEx, then uses UnmapViewOfFile to free up the space used to store the view of a file. It allocates virtual memory with VirtualAlloc and uses VirtualFree to likely free up the space that it didn't use in the previous function. Finally, it calls VirtualProtect, which changes the access permissions of a region in memory, as explained earlier. It is probable that the malware is using these functions to store malicious payload efficiently in memory and get it ready for execution.













## Dynamic Analysis

When the malware was detonated, an error message popped up saying the application had crashed. From the static analysis, we know that this malware is self-modifying, so it's possible that the malware recognized it was being analyzed in a virtual environment, so it terminated itself. Part of the malware did execute however, which can be seen in Process Monitor.






### Process Monitor

Process	Description	Image Path
 rootkit.exe (2544)	notepad	C:\Users\Administ.

This malicious file's description is "notepad", most likely to seem legitimate to the system. This answers the question of why notepad.exe was listed in PeStudio.

3:31:0...	 rootkit.exe	2544		RegOpenKey	HKLM\System\CurrentControlSet\Control\Terminal Server	REPARSE
3:31:0...	 rootkit.exe	2544		RegOpenKey	HKLM\System\CurrentControlSet\Control\Terminal Server	SUCCESS
3:31:0...	 rootkit.exe	2544		RegSetInfoKey	HKLM\System\CurrentControlSet\Control\Terminal Server	SUCCESS
3:31:0...	 rootkit.exe	2544		RegQueryValue	HKLM\System\CurrentControlSet\Control\Terminal Server\TSAppCompat	NAME NOT FOUND
3:31:0...	 rootkit.exe	2544		RegQueryValue	HKLM\System\CurrentControlSet\Control\Terminal Server\TSUserEnabled	SUCCESS
3:31:0...	 rootkit.exe	2544		RegCloseKey	HKLM\System\CurrentControlSet\Control\Terminal Server	SUCCESS

After executing the malware, it modifies keys in the "Terminal Server" directory in the registry. Terminal Server is a hosting feature that allows users to remotely access a Windows desktop session from another device via Terminal Services or Remote Desktop Services. The TSAppCompat key contains settings that affect how applications behave when in a remotely accessed environment. Some applications may not run correctly when in this type of environment, so changing the settings in the TSAppCompat key can make it so the applications work as intended when accessed remotely. TSUserEnabled is a key that determines whether users can access the computer remotely or not. Based on this data, it looks like the malware is getting everything set up to start a connection from an outside device to control the desktop remotely.

2756		RegOpenKey	HKLM\Software\Microsoft\Windows NT\CurrentVersion\AppCompatFlags\Layers	NAME NOT FOUND
2756		RegOpenKey	HKCU\Software\Microsoft\Windows NT\CurrentVersion\AppCompatFlags\Layers	SUCCESS
2756		RegSetInfoKey	HKCU\Software\Microsoft\Windows NT\CurrentVersion\AppCompatFlags\Layers	SUCCESS
2756		RegQueryValue	HKCU\Software\Microsoft\Windows NT\CurrentVersion\AppCompatFlags\Layers\VC:\Users\Administrator\Desktop\rootkit.exe	SUCCESS
2756		RegCloseKey	HKCU\Software\Microsoft\Windows NT\CurrentVersion\AppCompatFlags\Layers	SUCCESS

The AppCompatFlags\Layers key stores compatibility settings for applications. This is similar to the previous operations as the malware could be modifying compatibility settings to ensure applications run smoothly in different environments.

The malware also changes settings in the “Session Manager” key

1372	RegOpenKey	HKLM\System\CurrentControlSet\Control\Session Manager
1372	RegOpenKey	HKLM\System\CurrentControlSet\Control\Session Manager
1372	RegSetInfoKey	HKLM\System\CurrentControlSet\Control\SESSION MANAGER
1372	RegQueryValue	HKLM\System\CurrentControlSet\Control\SESSION MANAGER\CWDIllegalInDLLSearch
1372	RegCloseKey	HKLM\System\CurrentControlSet\Control\SESSION MANAGER

in the registry. This is an important key containing configuration settings that initializes the new session at startup. It's possible that the malware is changing values in this key to establish persistence in the target machine.

RegOpenKey	HKLM\Software\Wow6432Node\Microsoft\Windows\CurrentVersion\Internet Settings	SUCCESS
RegSetInfoKey	HKLM\SOFTWARE\Wow6432Node\Microsoft\Windows\CurrentVersion\Internet Settings	SUCCESS
RegQueryValue	HKLM\SOFTWARE\Wow6432Node\Microsoft\Windows\CurrentVersion\Internet Settings\DisableImprovedZoneCheck	NAME NOT FOUND
RegCloseKey	HKLM\SOFTWARE\Wow6432Node\Microsoft\Windows\CurrentVersion\Internet Settings	SUCCESS
RegQueryKey	HKLM	SUCCESS
RegOpenKey	HKLM\Software\Wow6432Node\Policies\Microsoft\Windows\CurrentVersion\Internet Settings	REPARSE
RegOpenKey	HKLM\SOFTWARE\Policies\Microsoft\Windows\CurrentVersion\Internet Settings	SUCCESS
RegSetInfoKey	HKLM\SOFTWARE\Policies\Microsoft\Windows\CurrentVersion\Internet Settings	SUCCESS
RegQueryValue	HKLM\SOFTWARE\Policies\Microsoft\Windows\CurrentVersion\Internet Settings\Security_HKLM_only	NAME NOT FOUND
RegCloseKey	HKLM\SOFTWARE\Policies\Microsoft\Windows\CurrentVersion\Internet Settings	SUCCESS

The Internet Settings key in the registry is also modified by the malware. The function, `DisableImprovedZoneCheck`, makes it so downloaded files from the internet bypass Windows security zones. It's possible that part of the malware's objective is to download more malicious files from the internet and wants to avoid triggering security mechanisms. This will allow more attacks to take place on the target machine.

## Conclusion

After doing static analysis on the malicious file, it was found that the malware imports various libraries and uses their functions to perform various tasks such as evading detection, manipulating system resources, and executing malicious code. It also searches the system to see if it is being analyzed by searching for tools like OLLYDBG. The malware aims to give itself permissions to execute malicious code in memory, as it uses the function `VirtualProtect` to do so. In the dynamic analysis, the malicious file was detonated to see more of what the malicious process looks like. As reported above, the malicious file crashed everytime it was run, meaning that either the file was simply corrupted or the malware saw that it was being analyzed, so it terminated itself. Because the malware was detected as being self-modifying, the latter is the more likely option. Using Process Monitor, it was found that the malware modifies values in the registry to allow users to remotely access the desktop and ensures the applications are compatible and will work correctly when under such an environment. The malware changes values in the Session Manager key that configures new sessions at startup, which the malware could be using to establish persistence. Finally, it was found that the malware modifies the Internet Settings key so that it can download more malicious files from the internet without being detected, likely to perform more attacks. Based on the findings, this is standard for rootkits as they aim to access devices remotely to infiltrate data. They try to stay undetected and give themselves elevated privileges to execute malicious code. This all checks out as everything found in this analysis backs it up.