Malware Analysis & Tools

By ThreatBlogger

Contents

Malware Analysis

Malware Analysis is the study of programs behaviour; there are many ways and tools which allow you to gain as much information from a suspicious program.

There are two approaches to analysing malware: static and dynamic analysis. Static analysis is analysing the malware without executing it. This involves using various tools to glean as much information from the source code, about what is does. Dynamic Analysis is executing the program in a controlled environment and analysing its behaviour and what's it's doing.

Static Analysis

Static Analysis is a way to study the programs behaviour, without actually executing it, there are tools such as decompilers, disassemblers, and source code analysers all which help to determine what its intentions are. An advantage of static analysis is that it gives an indication of what to look out for in the dynamic analysis, an example would be if a IP address, was viewed in the code, during the dynamic analysis it might try to contact this address over the internet.

Dynamic Analysis

In dynamic analysis the executable is run in a controlled environment or a "Sandbox" as it's known. This controlled environment is usually set up in a virtual machine and is usually a basic OS with nothing of importance stored on it. Dynamic analysis uses tools such as debuggers and packet sniffers. Dynamic analysis is about monitoring the system as the program is running and studying what changes its making to the system, what files it accesses, what changes to files it makes like adding or deleting files, remote logging: what external URLs or IP addresses does it connect to.

Malware Analysis Tools

This section will take you through the different technology used for analysis. Books have be written about some of these applications, they have complex functionally, here will be a brief overview of each application and the functionally which was helpful.

Static Analysis Tools

IDA PRO

The Interactive Disassembler, or simply known as IDA is a disassembler and debugger. It's an application which has many features such as programmable, extendible and a multi-processor disassemble which can be run on the leading operating systems. IDA both supports the x86 and x64 architectures. IDA supports many file formats such as Portable Executable (PE), Executable and Linking Format (ELF), Common Object (COFF). IDA PRO can disassemble a program and let you perform tasks such as function discovery, local variable identification and stack analysis, plus much more. Using IDA Pro lets you dig deeper into a program, allowing you to dig deeper into what it's doing and its intentions.

Using IDA PRO

You load an executable into IDA Pro by either dragging it in or by opening up the application and selecting it. First IDA Pro will recognise the files format and its processor architecture.
IDA maps the PE file into memory as if loaded by the operation system loader. Different options are available to disassemble the file either by PE format, MS-DOS executable or binary format.

- PE Format:
- MS-DOS executable :
- Binary File: which disassembles the file as raw data, allowing extracting more information than the PE format.

- **Manual Mode**

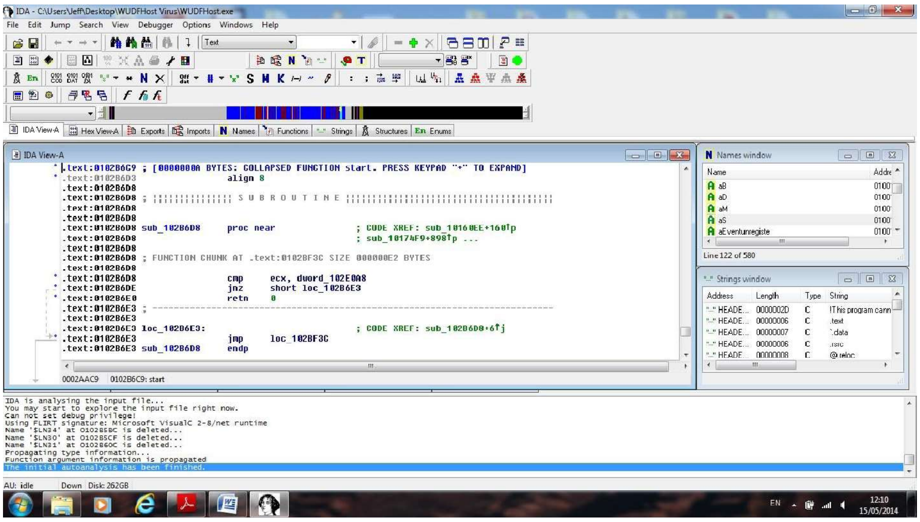
  IDA Pro does not load the PE header and the resources sections by default, by clicking Manual mode loads each section manually, so no section would escape analysis.

- **The Interface**

  After loading has complete, the disassembly window is opened. This is where the assembly code resides and where you can view the Functions, Imports and Exports and Strings windows, etc. IDA comes with different modes to view the binaries, Graph mode and Text mode, to simply change between modes press the space bar.

- **Text mode**

  The text mode displays a traditional view of the binary data regions, allowing you to view the memory addresses, the section name (.text) and opcodes (83EC18).



```
.text:0102B6C9 ; [0000000A BYTES: COLLAPSED FUNCTION start. PRESS KEYPAD "+" TO EXPAND]
.text:0102B6D3                 align 8
.text:0102B6D8
.text:0102B6D8 ; ||||||||||||||| S U B R O U T I N E |||||||||||||||||||||||||||||||||||||||
.text:0102B6D8
.text:0102B6D8
.text:0102B6D8 sub_102B6D8     proc near               ; CODE XREF: sub_10160EE+160↑p
.text:0102B6D8                                         ; sub_10174F9+898↑p ...
.text:0102B6D8
.text:0102B6D8 ; FUNCTION CHUNK AT .text:0102BF3C SIZE 000000E2 BYTES
.text:0102B6D8
.text:0102B6D8                 cmp     ecx, dword_102E0A8
.text:0102B6DE                 jnz     short loc_102B6E3
.text:0102B6E0                 retn    0
.text:0102B6E3 ; ---------------------------------------------------------------------------
.text:0102B6E3
.text:0102B6E3 loc_102B6E3:                            ; CODE XREF: sub_102B6D8+6↑j
.text:0102B6E3                 jmp     loc_102BF3C
.text:0102B6E3 sub_102B6D8     endp
```

0002AAC9 0102B6C9: start

Name	Addre
aB	0100
aD	0100
aM	0100
aS	0100
aEventunregiste	0100

Line 122 of 580

Address	Length	Type	String
HEADE...	0000002D	C	!This program cann
HEADE...	00000006	C	.text
HEADE...	00000007	C	`.data
HEADE...	00000006	C	.rsrc
HEADE...	00000008	C	@.reloc

```
IDA is analysing the input file...
You may start to explore the input file right now.
Can not set debug privilege!
Using FLIRT signature: Microsoft VisualC 2-8/net runtime
Name '$LN34' at 0102B5BC is deleted...
Name '$LN30' at 0102B5CF is deleted...
Name '$LN31' at 0102B60C is deleted...
Propagating type information...
Function argument information is propagated
The initial autoanalysis has been finished.
```

AU: idle Down Disk: 262GB

- **Graph mode**

  Graph mode displays operation codes and line numbers, plus allowing you to view the programs flow. Arrows indicate the paths the program flows in based on particular decisions having been made. The green arrow displays if a conditional jump is taken, red if it is not taken, and blue displays an unconditional jump. Upward arrows indicate loop situations.

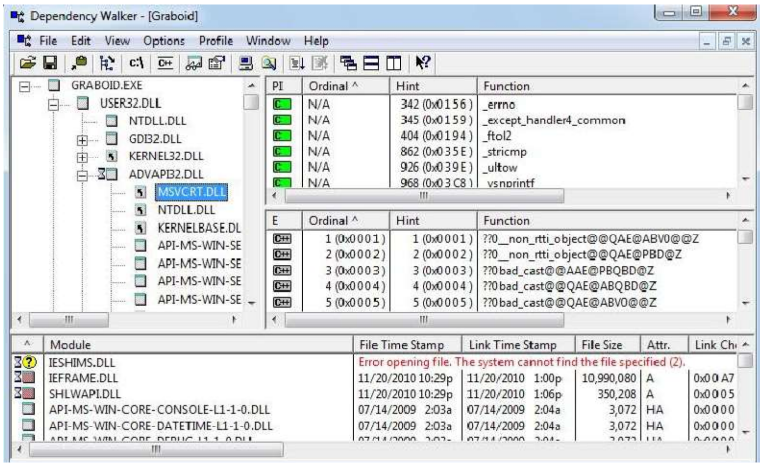
IDA windows

- **Functions window:** Displays all functions listed in the executable showing the length of each.
- **Names window:** displays address, functions, code, data and strings which has a name attached to it.
- **Imports window:** All imported DLL's the executable uses are listed.
- **Exports window:** All exported functions are listed for a file.
- **Structures window:** Lists all active data structures.
- **Strings window:** Shows all strings of text.

When analysing code these windows are particularly important, allowing you to cross-reference functions, by double clicking on an item of interest, it jumps directory to the main body of the code where that item is located.

Dependency Walker

The Dependency Walker tool can scan for all dependent DLL's used by a program, including missing DLLs and DLLs which are not valid. Dependency walker checks for import and export function match. Dependency Walker can help correct and prevent problems, with the use of DLLs.



PI	Ordinal ^	Hint	Function
C	N/A	342 (0x0156)	_errno
C	N/A	345 (0x0159)	_except_handler4_common
C	N/A	404 (0x0194)	_ftol2
C	N/A	862 (0x035E)	_stricmp
C	N/A	926 (0x039E)	_ultow
C	N/A	968 (0x03C8)	vsnprintf

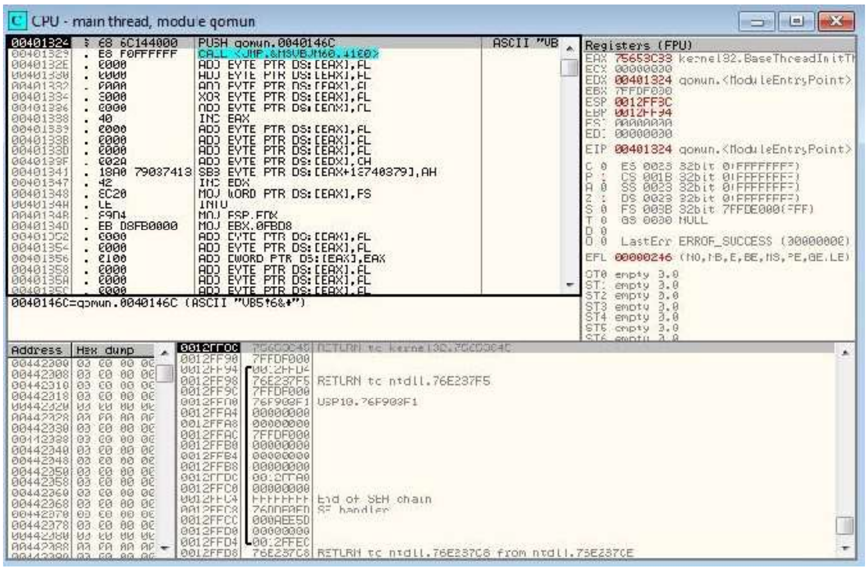
E	Ordinal ^	Hint	Function
C++	1 (0x0001)	1 (0x0001)	??0__non_rtti_object@@QAE@ABV0@@Z
C++	2 (0x0002)	2 (0x0002)	??0__non_rtti_object@@QAE@PBD@Z
C++	3 (0x0003)	3 (0x0003)	??0bad_cast@@AAE@PBQBD@Z
C++	4 (0x0004)	4 (0x0004)	??0bad_cast@@QAE@ABQBD@Z
C++	5 (0x0005)	5 (0x0005)	??0bad_cast@@QAE@ABV0@@Z

^	Module	File Time Stamp	Link Time Stamp	File Size	Attr.	Link Ch
?	IESHIMS.DLL	Error opening file. The system cannot find the file specified (2).				
	IEFRAME.DLL	11/20/2010 10:29p	11/20/2010 1:00p	10,990,080	A	0x00A7
	SHLWAPI.DLL	11/20/2010 10:29p	11/20/2010 1:06p	350,208	A	0x0005
	API-MS-WIN-CORE-CONSOLE-L1-1-0.DLL	07/14/2009 2:03a	07/14/2009 2:04a	3,072	HA	0x0000
	API-MS-WIN-CORE-DATETIME-L1-1-0.DLL	07/14/2009 2:03a	07/14/2009 2:04a	3,072	HA	0x0000

Dynamic Analysis Tools

OllyDbg

OllyDbg is an assembler level debugger for windows 32-bit executables. It analyses binary code, can be used to correct programs or go in dept in analysing assembly level code when no source code is available. You can view the assemble as its runs and see what changes its making to memory, what procedures and API calls its making, what DLLs are been used.



```
00401324  $ E8 6C144000    PUSH qomun.0040146C                    ASCII "VB
00401329  . E8 F0FFFFFF    CALL <JMP.&MSVBVM60.#100>
0040132E  . 0000           ADD BYTE PTR DS:[EAX],AL
00401330  . 0000           ADD BYTE PTR DS:[EAX],AL
00401332  . 0000           ADD BYTE PTR DS:[EAX],AL
00401334  . 3000           XOR BYTE PTR DS:[EAX],AL
00401336  . 0000           ADD BYTE PTR DS:[EAX],AL
00401338  . 40             INC EAX
00401339  . 0000           ADD BYTE PTR DS:[EAX],AL
0040133B  . 0000           ADD BYTE PTR DS:[EAX],AL
0040133D  . 0000           ADD BYTE PTR DS:[EAX],AL
0040133F  . 002A           ADD BYTE PTR DS:[EDX],CH
00401341  . 18A0 79037413  SBB BYTE PTR DS:[EAX+13740379],AH
00401347  . 42             INC EDX
00401348  . 8C20           MOV WORD PTR DS:[EAX],FS
0040134A  . CE             INTO
0040134B  . 89D4           MOV ESP,EDX
0040134D  . BB D8FB0000    MOV EBX,0FBD8
00401352  . 0000           ADD BYTE PTR DS:[EAX],AL
00401354  . 0000           ADD BYTE PTR DS:[EAX],AL
00401356  . 0100           ADD DWORD PTR DS:[EAX],EAX
00401358  . 0000           ADD BYTE PTR DS:[EAX],AL
0040135A  . 0000           ADD BYTE PTR DS:[EAX],AL
0040135C  . 0000           ADD BYTE PTR DS:[EAX],AL
0040146C=qomun.0040146C (ASCII "VB5!6&*")
```

```
Registers (FPU)
EAX 75653C33 kernel32.BaseThreadInitTh
ECX 00000000
EDX 00401324 qomun.<ModuleEntryPoint>
EBX 7FFDF000
ESP 0012FF8C
EBP 0012FF94
ESI 00000000
EDI 00000000
EIP 00401324 qomun.<ModuleEntryPoint>
C 0  ES 0023 32bit 0(FFFFFFFF)
P 1  CS 001B 32bit 0(FFFFFFFF)
A 0  SS 0023 32bit 0(FFFFFFFF)
Z 1  DS 0023 32bit 0(FFFFFFFF)
S 0  FS 003B 32bit 7FFDE000(FFF)
T 0  GS 0000 NULL
D 0
O 0  LastErr ERROR_SUCCESS (00000000)
EFL 00000246 (NO,NB,E,BE,NS,PE,GE,LE)
ST0 empty 0.0
ST1 empty 0.0
ST2 empty 0.0
ST3 empty 0.0
ST4 empty 0.0
ST5 empty 0.0
ST6 empty 0.0
```

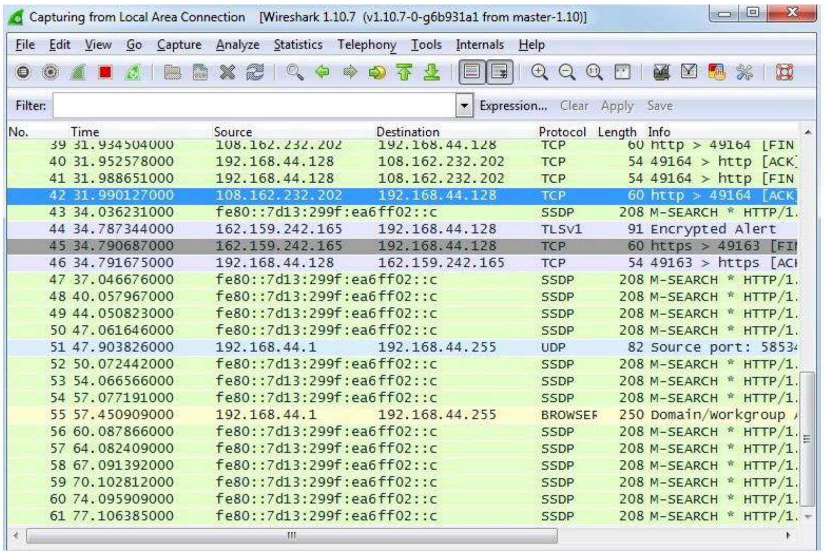
```
Address   Hex dump
00442300  03 C0 00 0C
00442308  03 C0 00 0C
00442310  03 C0 00 0C
00442318  03 C0 00 0C
00442320  03 C0 00 0C
00442328  03 C0 00 0C
00442330  03 C0 00 0C
00442338  03 C0 00 0C
00442340  03 C0 00 0C
00442348  03 C0 00 0C
00442350  03 C0 00 0C
00442358  03 C0 00 0C
00442360  03 C0 00 0C
00442368  03 C0 00 0C
00442370  03 C0 00 0C
00442378  03 C0 00 0C
00442380  03 C0 00 0C
00442388  03 C0 00 0C
00442390  03 C0 00 0C
```

```
0012FF8C   75653C45  RETURN to kernel32.75653C45
0012FF90   7FFDF000
0012FF94  ┌0012FFD4
0012FF98  │76E237F5  RETURN to ntdll.76E237F5
0012FF9C  │7FFDF000
0012FFA0  │76F903F1  USP10.76F903F1
0012FFA4  │00000000
0012FFA8  │00000000
0012FFAC  │7FFDF000
0012FFB0  │00000000
0012FFB4  │00000000
0012FFB8  │00000000
0012FFBC  │0012FFA0
0012FFC0  │00000000
0012FFC4  │FFFFFFFF  End of SEH chain
0012FFC8  │76E0FAED  SE handler
0012FFCC  │000AEE5D
0012FFD0  │00000000
0012FFD4  └0012FFEC
0012FFD8   76E237C8  RETURN to ntdll.76E237C8 from ntdll.75E237CE
```

- **CPU/Coding Panel:** This is the largest panel, and displays all assemble code produced. If stepping through it line-by-line, the code is highlighted showing the assemble code as its executing.
- **Registers Panel:** The registers panel is at the right hand side, and is used for holding information on the flags, standard and section registers plus extra information
- **Stack Panel :** The stack panel holds addresses which are used in the disassembles such as CPU section or memory dump section.
- **Memory Dump:** This is a dump for the memory addresses. Any raw data displayed alongside these addresses can be displayed in different formats such an ASCII, UNICODE, etc.

WireShark

Wireshark is a network protocol analyser and the standard application used by network analysist's. Wireshark is a network packet analyser which is used to capture network traffic and analyse the different protocols.



No.	Time	Source	Destination	Protocol	Length	Info
39	31.934504000	108.162.232.202	192.168.44.128	TCP	60	http > 49164 [FIN
40	31.952578000	192.168.44.128	108.162.232.202	TCP	54	49164 > http [ACK]
41	31.988651000	192.168.44.128	108.162.232.202	TCP	54	49164 > http [FIN
42	31.990127000	108.162.232.202	192.168.44.128	TCP	60	http > 49164 [ACK]
43	34.036231000	fe80::7d13:299f:ea6	ff02::c	SSDP	208	M-SEARCH * HTTP/1.
44	34.787344000	162.159.242.165	192.168.44.128	TLSv1	91	Encrypted Alert
45	34.790687000	162.159.242.165	192.168.44.128	TCP	60	https > 49163 [FI
46	34.791675000	192.168.44.128	162.159.242.165	TCP	54	49163 > https [AC
47	37.046676000	fe80::7d13:299f:ea6	ff02::c	SSDP	208	M-SEARCH * HTTP/1.
48	40.057967000	fe80::7d13:299f:ea6	ff02::c	SSDP	208	M-SEARCH * HTTP/1.
49	44.050823000	fe80::7d13:299f:ea6	ff02::c	SSDP	208	M-SEARCH * HTTP/1.
50	47.061646000	fe80::7d13:299f:ea6	ff02::c	SSDP	208	M-SEARCH * HTTP/1.
51	47.903826000	192.168.44.1	192.168.44.255	UDP	82	Source port: 5853
52	50.072442000	fe80::7d13:299f:ea6	ff02::c	SSDP	208	M-SEARCH * HTTP/1.
53	54.066566000	fe80::7d13:299f:ea6	ff02::c	SSDP	208	M-SEARCH * HTTP/1.
54	57.077191000	fe80::7d13:299f:ea6	ff02::c	SSDP	208	M-SEARCH * HTTP/1.
55	57.450909000	192.168.44.1	192.168.44.255	BROWSER	250	Domain/Workgroup
56	60.087866000	fe80::7d13:299f:ea6	ff02::c	SSDP	208	M-SEARCH * HTTP/1.
57	64.082409000	fe80::7d13:299f:ea6	ff02::c	SSDP	208	M-SEARCH * HTTP/1.
58	67.091392000	fe80::7d13:299f:ea6	ff02::c	SSDP	208	M-SEARCH * HTTP/1.
59	70.102812000	fe80::7d13:299f:ea6	ff02::c	SSDP	208	M-SEARCH * HTTP/1.
60	74.095909000	fe80::7d13:299f:ea6	ff02::c	SSDP	208	M-SEARCH * HTTP/1.
61	77.106385000	fe80::7d13:299f:ea6	ff02::c	SSDP	208	M-SEARCH * HTTP/1.

Some of Wiresharks feature set include:

- Deep inspection of all standard and new protocols
- Live packet capture plus offline analysis
- Network data can be viewed via a GUI or browsed via the TShare utility or TTY-mode.
- Can read and write different data capture formats from other sniffer and network monitor applications.
- Live data can be read from other channels such as 802.11, Ethernet, PPP/HDLC Bluetooth, etc.

Some of Wiresharks feature set include:

- Deep inspection of all standard and new protocols
- Live packet capture plus offline analysis
- Network data can be viewed via a GUI or browsed via the TShare utility or TTY-mode.
- Can read and write different data capture formats from other sniffer and network monitor applications.

Sandboxes

Virtualisation Software provides a save and time saving environment to test malware. Easy to set up and configure a lab environment can be set up in minutes. A virtual machine allows you to investigate the behavioural analysis of the malware sample, by how the specimen interacts with the file system, registry and the network.

- With a virtual machine, it’s possible to take a snapshot of the systems state before, during and after and infection takes place. This allows identification of what changes to the system were made.

- The host-only option allows you to interconnect virtual systems.
- Update the security patches regularly.
- Don’t enable networking on the virtual machine
- Do not connect the virtual machine to the host machine
- Monitor the host for any signs of infection from the virtual machine.

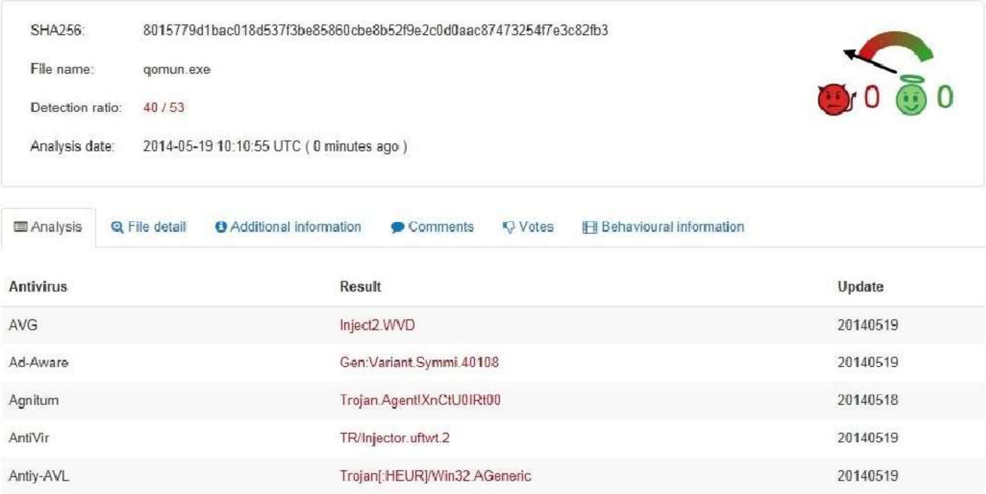
Malware Detection Tools

There are a number of tools used for spotting malware.

VirusTotal

VirusTotal is an online service which analyses suspicious executables files which are uploaded by the users. It quickly detects the different malware types such as Trojans, viruses and worms. It gives an indication if the file is harmful or not, with a 0/50 rating.

When a file is uploaded and the content of the file is scanned by 40 different Antivirus engines all at the same time. The information gathered by these antivirus engines is outputted and a rating is given as to how harmful the file is.



SHA256: 8015779d1bac018d537f3be85860cbe8b52f9e2c0d0aac87473254f7e3c82fb3

File name: qomun.exe

Detection ratio: 40 / 53

Analysis date: 2014-05-19 10:10:55 UTC (0 minutes ago)

Antivirus	Result	Update
AVG	Inject2.WVD	20140519
Ad-Aware	Gen:Variant.Symmi.40108	20140519
Agnitum	Trojan.Agent!XnCtU0IRt00	20140518
AntiVir	TR/Injector.uftwt.2	20140519
Antiy-AVL	Trojan[:HEUR]/Win32.AGeneric	20140519

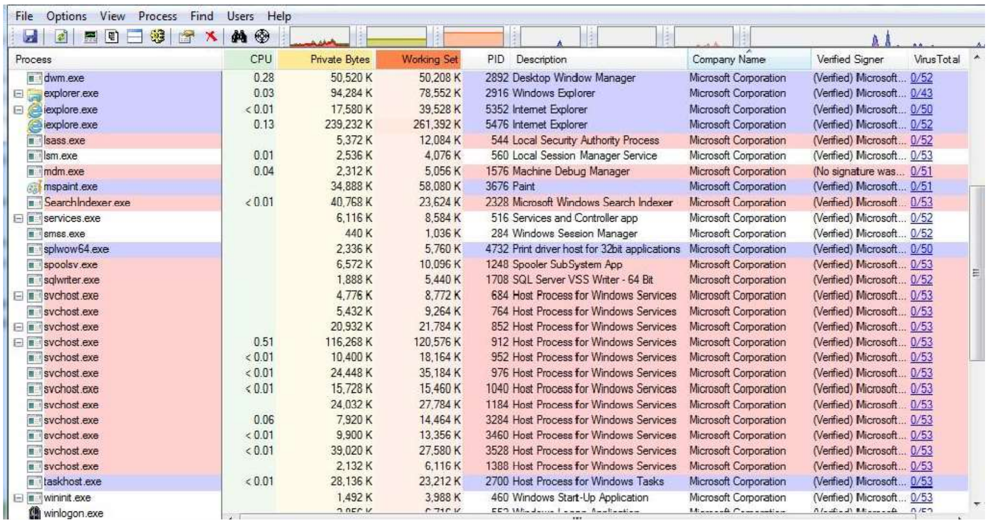
It acts an information aggregator, gathering data about the different types of malware, and storing this information, for security professionals. Total Virus gives you a date first seen (or date it first uploaded to its database), this indicates its new malware, and a detection ratio of 1/50 means an Antivirus has also detected it. By submitting a file to the website you’re allowing VirusTotal to copy the file to its own database, and at the same time it produces its own MD5 Signature.

Sysinternals Suite

Sysinternals (Suite) is a collection of Microsoft utilities which allow you to diagnose, monitor and troubleshoot computer problems or hunt down Malware which may be on your system. To find Malware on a System these tools are ideal, but a lot of experience is needed to find the abnormal form the normal

Process Explorer

Processor Explorer is a monitoring tool which allows you to view running processes or applications. It's a task manager but gives a lot more information about the state of your system, such as which exe and DLLs processing are currently running and what other resources are they using.



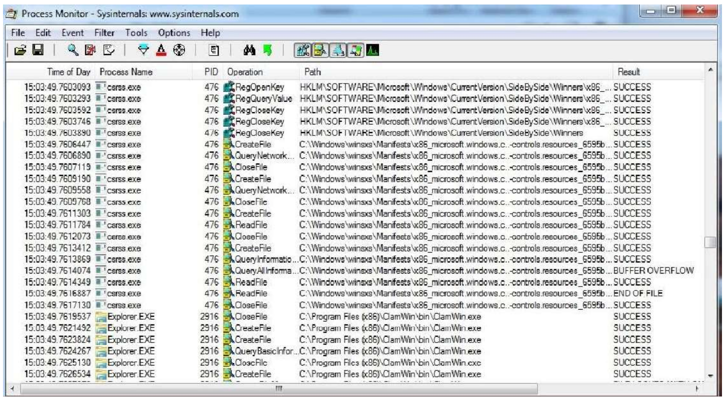
Process	CPU	Private Bytes	Working Set	PID	Description	Company Name	Verified Signer	VirusTotal
dwm.exe	0.28	50,520 K	50,208 K	2892	Desktop Window Manager	Microsoft Corporation	(Verified) Microsoft...	0/52
explorer.exe	0.03	94,284 K	78,552 K	2916	Windows Explorer	Microsoft Corporation	(Verified) Microsoft...	0/43
iexplore.exe	< 0.01	17,580 K	39,528 K	5352	Internet Explorer	Microsoft Corporation	(Verified) Microsoft...	0/50
iexplore.exe	0.13	239,232 K	261,392 K	5476	Internet Explorer	Microsoft Corporation	(Verified) Microsoft...	0/52
lsass.exe		5,372 K	12,084 K	544	Local Security Authority Process	Microsoft Corporation	(Verified) Microsoft...	0/52
lsm.exe	0.01	2,536 K	4,076 K	560	Local Session Manager Service	Microsoft Corporation	(Verified) Microsoft...	0/53
mdm.exe	0.04	2,312 K	5,056 K	1576	Machine Debug Manager	Microsoft Corporation	(No signature was...	0/51
mspaint.exe		34,888 K	58,080 K	3676	Paint	Microsoft Corporation	(Verified) Microsoft...	0/51
SearchIndexer.exe	< 0.01	40,768 K	23,624 K	2328	Microsoft Windows Search Indexer	Microsoft Corporation	(Verified) Microsoft...	0/53
services.exe		6,116 K	8,584 K	516	Services and Controller app	Microsoft Corporation	(Verified) Microsoft...	0/52
smss.exe		440 K	1,036 K	284	Windows Session Manager	Microsoft Corporation	(Verified) Microsoft...	0/52
splwow64.exe		2,336 K	5,760 K	4732	Print driver host for 32bit applications	Microsoft Corporation	(Verified) Microsoft...	0/50
spoolsv.exe		6,572 K	10,096 K	1248	Spooler SubSystem App	Microsoft Corporation	(Verified) Microsoft...	0/53
sqlwriter.exe		1,888 K	5,440 K	1708	SQL Server VSS Writer - 64 Bit	Microsoft Corporation	(Verified) Microsoft...	0/52
svchost.exe		4,776 K	8,772 K	684	Host Process for Windows Services	Microsoft Corporation	(Verified) Microsoft...	0/53
svchost.exe		5,432 K	9,264 K	764	Host Process for Windows Services	Microsoft Corporation	(Verified) Microsoft...	0/53
svchost.exe		20,932 K	21,784 K	852	Host Process for Windows Services	Microsoft Corporation	(Verified) Microsoft...	0/53
svchost.exe	0.51	116,268 K	120,576 K	912	Host Process for Windows Services	Microsoft Corporation	(Verified) Microsoft...	0/53
svchost.exe	< 0.01	10,400 K	18,164 K	952	Host Process for Windows Services	Microsoft Corporation	(Verified) Microsoft...	0/53
svchost.exe	< 0.01	24,448 K	35,184 K	976	Host Process for Windows Services	Microsoft Corporation	(Verified) Microsoft...	0/53
svchost.exe	< 0.01	15,728 K	15,460 K	1040	Host Process for Windows Services	Microsoft Corporation	(Verified) Microsoft...	0/53
svchost.exe		24,032 K	27,784 K	1184	Host Process for Windows Services	Microsoft Corporation	(Verified) Microsoft...	0/53
svchost.exe	0.06	7,920 K	14,464 K	3284	Host Process for Windows Services	Microsoft Corporation	(Verified) Microsoft...	0/53
svchost.exe	< 0.01	9,900 K	13,356 K	3460	Host Process for Windows Services	Microsoft Corporation	(Verified) Microsoft...	0/53
svchost.exe	< 0.01	39,020 K	27,580 K	3528	Host Process for Windows Services	Microsoft Corporation	(Verified) Microsoft...	0/53
svchost.exe		2,132 K	6,116 K	1388	Host Process for Windows Services	Microsoft Corporation	(Verified) Microsoft...	0/53
taskhost.exe	< 0.01	28,136 K	23,212 K	2700	Host Process for Windows Tasks	Microsoft Corporation	(Verified) Microsoft...	0/53
wininit.exe		1,492 K	3,988 K	460	Windows Start-Up Application	Microsoft Corporation	(Verified) Microsoft...	0/53
winlogon.exe								

The GUI when opened shows colour coded active processes, the pink processes are windows services, the blue Explorer services and white are start-up services.

Its' a suburb tool for malware hunting as it names the company which created the exe or DLL, provides a signature which can be verified and shows that it's a genuine product from that company, also allows you to run it against VirusTotal.com's database and 40 antivirus companies, to see if it is actually malware.

Process Monitor

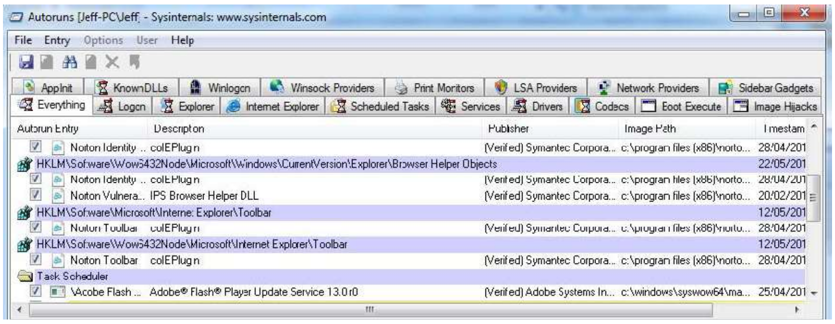
Processor Monitor displays different filters allowing you to display activity from the Registry, File System and the activity of the different process and threads. Only an advanced user could make sense of the extensive amount of information that is constantly streaming through its spreadsheet-like GUI. The user can choose the toolbar to filter information, from the Registry, File system, process and thread, the stack or networking activity.


Process Monitor - Sysinternals: www.sysinternals.com
File Edit Event Filter Tools Options Help

Time of Day	Process Name	PID	Operation	Path	Result
15:03:49.7603093	csrss.exe	476	RegOpenKey	HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\SideBySide\Winners\x86_...	SUCCESS
15:03:49.7603293	csrss.exe	476	RegQueryValue	HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\SideBySide\Winners\x86_...	SUCCESS
15:03:49.7603592	csrss.exe	476	RegCloseKey	HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\SideBySide\Winners\x86_...	SUCCESS
15:03:49.7603746	csrss.exe	476	RegCloseKey	HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\SideBySide\Winners\x86_...	SUCCESS
15:03:49.7603890	csrss.exe	476	RegCloseKey	HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\SideBySide\Winners	SUCCESS
15:03:49.7606447	csrss.exe	476	CreateFile	C:\Windows\winsxs\Manifests\x86_microsoft.windows.c..-controls.resources_6595b...	SUCCESS
15:03:49.7606890	csrss.exe	476	QueryNetwork...	C:\Windows\winsxs\Manifests\x86_microsoft.windows.c..-controls.resources_6595b...	SUCCESS
15:03:49.7607119	csrss.exe	476	CloseFile	C:\Windows\winsxs\Manifests\x86_microsoft.windows.c..-controls.resources_6595b...	SUCCESS
15:03:49.7609190	csrss.exe	476	CreateFile	C:\Windows\winsxs\Manifests\x86_microsoft.windows.c..-controls.resources_6595b...	SUCCESS
15:03:49.7609558	csrss.exe	476	QueryNetwork...	C:\Windows\winsxs\Manifests\x86_microsoft.windows.c..-controls.resources_6595b...	SUCCESS
15:03:49.7609768	csrss.exe	476	CloseFile	C:\Windows\winsxs\Manifests\x86_microsoft.windows.c..-controls.resources_6595b...	SUCCESS
15:03:49.7611303	csrss.exe	476	CreateFile	C:\Windows\winsxs\Manifests\x86_microsoft.windows.c..-controls.resources_6595b...	SUCCESS
15:03:49.7611784	csrss.exe	476	ReadFile	C:\Windows\winsxs\Manifests\x86_microsoft.windows.c..-controls.resources_6595b...	SUCCESS
15:03:49.7612073	csrss.exe	476	CloseFile	C:\Windows\winsxs\Manifests\x86_microsoft.windows.c..-controls.resources_6595b...	SUCCESS
15:03:49.7613412	csrss.exe	476	CreateFile	C:\Windows\winsxs\Manifests\x86_microsoft.windows.c..-controls.resources_6595b...	SUCCESS
15:03:49.7613869	csrss.exe	476	QueryInformatio...	C:\Windows\winsxs\Manifests\x86_microsoft.windows.c..-controls.resources_6595b...	SUCCESS
15:03:49.7614074	csrss.exe	476	QueryAllInforma...	C:\Windows\winsxs\Manifests\x86_microsoft.windows.c..-controls.resources_6595b...	BUFFER OVERFLOW
15:03:49.7614349	csrss.exe	476	ReadFile	C:\Windows\winsxs\Manifests\x86_microsoft.windows.c..-controls.resources_6595b...	SUCCESS
15:03:49.7616887	csrss.exe	476	ReadFile	C:\Windows\winsxs\Manifests\x86_microsoft.windows.c..-controls.resources_6595b...	END OF FILE
15:03:49.7617130	csrss.exe	476	CloseFile	C:\Windows\winsxs\Manifests\x86_microsoft.windows.c..-controls.resources_6595b...	SUCCESS
15:03:49.7619537	Explorer.EXE	2916	CloseFile	C:\Program Files (x86)\ClamWin\bin\ClamWin.exe	SUCCESS
15:03:49.7621492	Explorer.EXE	2916	CreateFile	C:\Program Files (x86)\ClamWin\bin\ClamWin.exe	SUCCESS
15:03:49.7623824	Explorer.EXE	2916	CreateFile	C:\Program Files (x86)\ClamWin\bin\ClamWin.exe	SUCCESS
15:03:49.7624267	Explorer.EXE	2916	QueryBasicInfor...	C:\Program Files (x86)\ClamWin\bin\ClamWin.exe	SUCCESS
15:03:49.7625130	Explorer.EXE	2916	CloseFile	C:\Program Files (x86)\ClamWin\bin\ClamWin.exe	SUCCESS
15:03:49.7626534	Explorer.EXE	2916	CreateFile	C:\Program Files (x86)\ClamWin\bin\ClamWin.exe	SUCCESS

Autoruns

Autoruns shows the programs which are running at start-up or login. You can review the different start-up folder locations (start-up, RunOnce, Services).Autoruns allows you to look for malware in these locations, it will be highlighted.



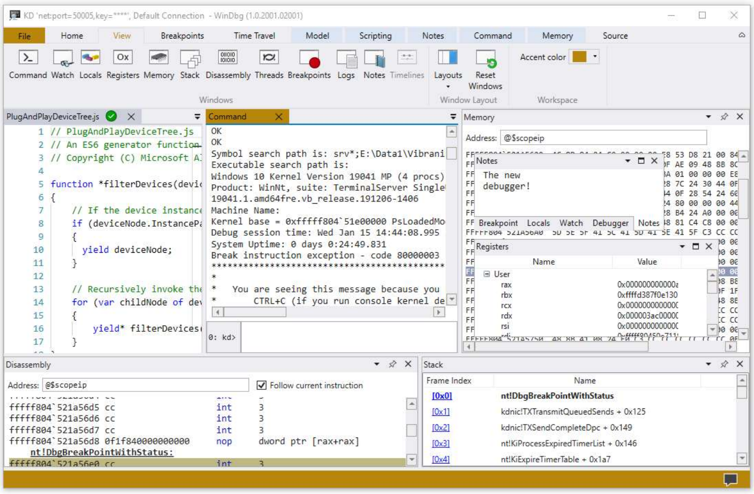
Autorun Entry	Description	Publisher	Image Path	Timestam
☑ Norton Identity ...	coIEPlugin	(Verified) Symantec Corpora...	c:\program files (x86)\norto...	28/04/201
HKLM\Software\Wow6432Node\Microsoft\Windows\CurrentVersion\Explorer\Browser Helper Objects				22/05/201
☑ Norton Identity ...	coIEPlugin	(Verified) Symantec Corpora...	c:\program files (x86)\norto...	28/04/201
☑ Norton Vulnera...	IPS Browser Helper DLL	(Verified) Symantec Corpora...	c:\program files (x86)\norto...	20/02/201
HKLM\Software\Microsoft\Internet Explorer\Toolbar				12/05/201
☑ Norton Toolbar	coIEPlugin	(Verified) Symantec Corpora...	c:\program files (x86)\norto...	28/04/201
HKLM\Software\Wow6432Node\Microsoft\Internet Explorer\Toolbar				12/05/201
☑ Norton Toolbar	coIEPlugin	(Verified) Symantec Corpora...	c:\program files (x86)\norto...	28/04/201
Task Scheduler				
☑ \Adobe Flash ...	Adobe® Flash® Player Update Service 13.0 r0	(Verified) Adobe Systems In...	c:\windows\syswow64\ma...	25/04/201

Hex Editor

A hex editor is an application which allow the viewing and editing of binary files. The binary executables instructions and strings can be viewed giving the analyst an idea of what the program does and its intention.

Windbg

Windbg is a free Window Debugger from Windows. Its original purpose was to help Window software developers debug code and fix software problems. It can debug user-mode and kernel-mode code, analyse crash dumps, and examine the CPU registers while the code executes.



```
// PlugAndPlayDeviceTree.js
// An ES6 generator function
// Copyright (C) Microsoft A

function *filterDevices(devi
{
    // If the device instance
    if (deviceNode.InstancePa
    {
      yield deviceNode;
    }

    // Recursively invoke the
    for (var childNode of dev
    {
        yield* filterDevices(
    }
```

```
OK
OK
Symbol search path is: srv*;E:\Data1\Vibrani
Executable search path is:
Windows 10 Kernel Version 19041 MP (4 procs)
Product: WinNt, suite: TerminalServer Single
19041.1.amd64fre.vb_release.191206-1406
Machine Name:
Kernel base = 0xfffff804`51e00000 PsLoadedMo
Debug session time: Wed Jan 15 14:44:08.995
System Uptime: 0 days 0:24:49.831
Break instruction exception - code 80000003
********************************************
*
*   You are seeing this message because you
*       CTRL+C (if you run console kernel de
0: kd>
```

Address: @$scopeip

The new
debugger!

Name	Value
⊟ User	
rax	0x000000000000a
rbx	0xffffd387f0e130
rcx	0x00000000000000
rdx	0x000003ac00000
rsi	0x00000000000000

Address: @$scopeip ☑ Follow current instruction

```
fffff804`521a56d5 cc                int     3
fffff804`521a56d6 cc                int     3
fffff804`521a56d7 cc                int     3
fffff804`521a56d8 0f1f840000000000  nop     dword ptr [rax+rax]
    nt!DbgBreakPointWithStatus:
fffff804`521a56e0 cc                int     3
```

Frame Index	Name
[0x0]	**nt!DbgBreakPointWithStatus**
[0x1]	kdnic!TXTransmitQueuedSends + 0x125
[0x2]	kdnic!TXSendCompleteDpc + 0x149
[0x3]	nt!KiProcessExpiredTimerList + 0x146
[0x4]	nt!KiExpireTimerTable + 0x1a7

To perform Windows kernel analysis, we need a specific setup of the Virtual Machine and Windbg in order to remotely debug the VM from the host. Rootkits leverage kernel level components to facilitate activities such as hiding processes, files, network connections and other common objects.