

Task 1 (5 marks, 1 mark for each question): Answer the following questions by analyzing HW-A-1.exe using basic static analysis techniques only.

1. Upload the programs to <https://www.virustotal.com> and check if they match any existing antivirus definition?

VirusTotal says that 58 vendors flag this file as malicious.

58/71 security vendors and no sandboxes flagged this file as malicious

5745055aa8ac073d428c581d5f8828d88aa2fba34c9b7f511e...

HW-A-1.exe

Size: 14.00 KB

Last Modification Date: 22 minutes ago

Community Score

Popular threat label: trojan.dochief/fragtor

Threat categories: trojan, downloader

Family labels: dochief, fragtor

2. Are there any indications that these files are packed or obfuscated? If so, what are these indicators? If the file is packed, unpack it.

The file is already packed. This can be checked by using upx on the file.

```
C:\Users\IEUser\Desktop\A3>upx HW-A-1.exe
Ultimate Packer for eXecutables
Copyright (C) 1996 - 2020
UPX 3.96w Markus Oberhumer, Laszlo Molnar & John Reiser Jan 23rd 2020

File size      Ratio      Format      Name
-----
upx: HW-A-1.exe: AlreadyPackedException: already packed by UPX
Packed 1 file: 0 ok, 1 error.
```

Unpacking the file.

```
C:\Users\IEUser\Desktop\A3>upx -d HW-A-1.exe
Ultimate Packer for eXecutables
Copyright (C) 1996 - 2020
UPX 3.96w Markus Oberhumer, Laszlo Molnar & John Reiser Jan 23rd 2020

File size      Ratio      Format      Name
-----
32768 <-      14336      43.75%      win32/pe      HW-A-1.exe
Unpacked 1 file.
```

3. When was the program compiled?

Using PEview: The program was compiled on 17th November 2011 17:58:55 UTC.

```
000000E8 4EC54B5F Time Date Stamp 2011/11/17 Thu 17:58:55 UTC
000000EC 00000000 Pointer to Symbol Table
```

4. Do any of the imports hint at the program's functionality? If so, which imports are they, and what do they tell you?

000060B4	000068D2	Hint/Name RVA	0000	InternetCloseHandle
000060B8	000068E8	Hint/Name RVA	0000	FtpPutFileA
000060BC	000068F6	Hint/Name RVA	0000	InternetOpenA
000060C0	00006906	Hint/Name RVA	0000	InternetConnectA
000060C4	00006918	Hint/Name RVA	0000	FtpSetCurrentDirectoryA
000060C8	00000000	End of Imports		WININET.dll
00006000	00006610	Hint/Name RVA	0000	FindNextFileA
00006004	00006620	Hint/Name RVA	0000	FindClose
00006008	0000662C	Hint/Name RVA	0000	FindFirstFileA
0000600C	0000663C	Hint/Name RVA	0000	FlushFileBuffers
00006010	0000664E	Hint/Name RVA	0000	GetStringTypeW
00006014	0000665E	Hint/Name RVA	0000	GetStringTypeA
00006018	0000666E	Hint/Name RVA	0000	LCMapStringW
0000601C	0000667C	Hint/Name RVA	0000	LCMapStringA
00006020	0000668A	Hint/Name RVA	0000	MultiByteToWideChar
00006024	000066A0	Hint/Name RVA	0000	SetStdHandle
00006028	000066AE	Hint/Name RVA	0000	LoadLibraryA
0000602C	000066BC	Hint/Name RVA	0000	GetProcAddress
00006030	000066CC	Hint/Name RVA	0000	HeapAlloc
00006034	000066D8	Hint/Name RVA	0000	GetModuleHandleA
00006038	000066EA	Hint/Name RVA	0000	GetStartupInfoA
0000603C	000066FC	Hint/Name RVA	0000	GetCommandLineA
00006040	0000670E	Hint/Name RVA	0000	GetVersion
00006044	0000671A	Hint/Name RVA	0000	ExitProcess
00006048	00006728	Hint/Name RVA	0000	HeapDestroy
0000604C	00006736	Hint/Name RVA	0000	HeapCreate
00006050	00006742	Hint/Name RVA	0000	VirtualFree
00006054	00006750	Hint/Name RVA	0000	HeapFree
00006058	0000675A	Hint/Name RVA	0000	VirtualAlloc
0000605C	00006768	Hint/Name RVA	0000	HeapReAlloc
00006060	00006776	Hint/Name RVA	0000	TerminateProcess
00006064	00006788	Hint/Name RVA	0000	GetCurrentProcess
00006068	0000679C	Hint/Name RVA	0000	UnhandledExceptionFilter
0000606C	000067B6	Hint/Name RVA	0000	GetModuleFileNameA
00006070	000067CA	Hint/Name RVA	0000	FreeEnvironmentStringsA
00006074	000067E4	Hint/Name RVA	0000	FreeEnvironmentStringsW
00006078	000067FE	Hint/Name RVA	0000	WideCharToMultiByte
0000607C	00006814	Hint/Name RVA	0000	GetEnvironmentStrings
00006080	0000682C	Hint/Name RVA	0000	GetEnvironmentStringsW
00006084	00006844	Hint/Name RVA	0000	SetHandleCount
00006088	00006854	Hint/Name RVA	0000	SetStdHandle
0000608C	00006862	Hint/Name RVA	0000	GetFileType
00006090	00006870	Hint/Name RVA	0000	RtlUnwind
00006094	0000687C	Hint/Name RVA	0000	WriteFile
00006098	00006888	Hint/Name RVA	0000	GetLastError
0000609C	00006896	Hint/Name RVA	0000	SetFilePointer
000060A0	000068A6	Hint/Name RVA	0000	GetCPInfo
000060A4	000068B2	Hint/Name RVA	0000	GetACP
000060A8	000068BA	Hint/Name RVA	0000	GetOEMCP
000060AC	000068C4	Hint/Name RVA	0000	CloseHandle

WININET.dll import Windows Internet (WinINet) application programming interface (API) enables your application to interact with FTP and HTTP protocols to access Internet resources. These imports show that the program will try to connect to a FTP server, perhaps a C2 belonging to the attacker, to upload files there.

Based on the imports such as `FindNextFileA` and `WriteFile`, we can also guess that the program is trying to find files and maybe upload them to the FTP server.

Running `strings` on the program we can find these which could show the files being accessed.

```
.pdf
.doc
%s-%d.pdf
pdfs
%s-%d.doc
docs
C:\*
```

Thus the malware could be uploading pdf and doc files from the C drive to the FTP server.

5. What host- or network-based indicators can you use to identify these malwares on infected machines?

Running `strings` on the program we can also find this string,

`ftp.practicalmalwareanalysis.net`. This could be a network-based indicator if the victim machine is connected to it.

```
ftp.practicalmalwareanalysis.net
Home ftp client
```

Host based indicators could be if pdf and doc files were accessed.

Task 2 (5 marks, 1 mark for each question): Answer the following questions by analyzing HW-A-2.exe using basic static analysis techniques only.

1. Upload the programs to <https://www.virustotal.com> and check if they match any existing antivirus definition?

53/71 security vendors and no sandboxes flagged this file as malicious

34d2db9c85069a1ff40917c319ba7a65097cd1d77ea250e2a6fe5a5b53bdfb9

HW-A-2.exe

Size: 172.00 KB | Last Modification Date: 1 hour ago

peexe persistence spreader detect debug environment runtime modules checks user input direct cpu clock access

DETECTION DETAILS RELATIONS BEHAVIOR TELEMTRY COMMUNITY

Join the VT Community and enjoy additional community insights and crowdsourced detections, plus an API key to automate checks.

Popular threat label: trojan.doina/invasion | Threat categories: trojan, dropper | Family labels: doina, invader, generic

2. When was the program compiled?

000000F8 4ECB186E Time Date Stamp 2011/11/22 Tue 03:35:10 UTC

3. Do any of the imports hint at the program's functionality? If so, which imports are they, and what do they tell you?

0000702C	00007ED8	Hint/Name RVA	0196	LookupPrivilegeValueA			
00007030	00007EC4	Hint/Name RVA	01F7	OpenProcessToken			
00007034	00007EF0	Hint/Name RVA	001F	AdjustTokenPrivileges	00007160	00007F16	Hint/Name RVA
00007038	00000000	End of Imports	ADVAPI32.dll		00007164	00000000	End of Imports
							011E ShellExecuteA
							SHELL32.dll

0000703C	00007046	Hint/Name RVA	01C0	GetCurrentProcess	000070DC	000080C0	Hint/Name RVA	02EF	InterlockedIncrement
00007040	0000705A	Hint/Name RVA	00A9	CreateRemoteThread	000070E0	000080D8	Hint/Name RVA	0473	SetLastError
00007044	000070D0	Hint/Name RVA	0025	WriteFile	000070E4	000080E8	Hint/Name RVA	01C5	GetCurrentThreadid
00007048	000070DC	Hint/Name RVA	0380	OpenProcess	000070E8	000080FE	Hint/Name RVA	02EB	InterlockedDecrement
0000704C	000070DE	Hint/Name RVA	04B1	SizeofResource	000070EC	00008116	Hint/Name RVA	0264	GetStdHandle
00007050	000070FC	Hint/Name RVA	026F	GetSystemDirectoryA	000070F0	00008126	Hint/Name RVA	0214	GetModuleFileNameW
00007054	00007E12	Hint/Name RVA	053E	IsLocalA	000070F4	0000813C	Hint/Name RVA	0213	GetModuleFileNameA
00007058	00007E1E	Hint/Name RVA	0245	GetProcAddress	000070F8	00008152	Hint/Name RVA	0161	FreeEnvironmentStringsW
0000705C	00007D96	Hint/Name RVA	0341	LoadResource	000070FC	0000816C	Hint/Name RVA	0511	WideCharToMultiByte
00007060	00007E40	Hint/Name RVA	04EA	VirtualAllocEx	00007100	00008182	Hint/Name RVA	01DA	GetEnvironmentStringsW
00007064	00007E52	Hint/Name RVA	033C	LoadLibraryA	00007104	0000819C	Hint/Name RVA	046F	SetHandleCount
00007068	00007E62	Hint/Name RVA	0354	LockResource	00007108	000081AE	Hint/Name RVA	01F3	GetFileType
0000706C	00007E72	Hint/Name RVA	0215	GetModuleHandleA	0000710C	000081BC	Hint/Name RVA	0263	GetStartupInfoW
00007070	00007E86	Hint/Name RVA	0052	CloseHandle	00007110	000081CE	Hint/Name RVA	02CD	HeapCreate
00007074	00007E94	Hint/Name RVA	052E	WriteProcessMemory	00007114	000081DC	Hint/Name RVA	03A7	QueryPerformanceCounter
00007078	00007EAA	Hint/Name RVA	0547	IsProcessA	00007118	000081F6	Hint/Name RVA	0293	GetTickCount
0000707C	00007D96	Hint/Name RVA	0165	FreeResource	0000711C	00008206	Hint/Name RVA	01C1	GetCurrentProcessid
00007080	00007D76	Hint/Name RVA	014B	FindResourceA	00007120	0000821C	Hint/Name RVA	0279	GetSystemTimeAsFileTime
00007084	00007E30	Hint/Name RVA	04E9	VirtualAlloc	00007124	00008236	Hint/Name RVA	02CF	HeapFree
00007088	00007D68	Hint/Name RVA	0088	CreateFileA	00007128	00008242	Hint/Name RVA	04B2	Sleep
0000708C	00007F32	Hint/Name RVA	0218	GetModuleHandleW	0000712C	0000824A	Hint/Name RVA	0172	GetCPInfo
00007090	00007F46	Hint/Name RVA	0119	ExitProcess	00007130	00008256	Hint/Name RVA	0168	GetACP
00007094	00007F54	Hint/Name RVA	00CA	DecodePointer	00007134	00008260	Hint/Name RVA	0237	GetOEMCP
00007098	00007F64	Hint/Name RVA	0186	GetCommandLineA	00007138	0000826C	Hint/Name RVA	030A	IsValidCodePage
0000709C	00007F76	Hint/Name RVA	0033	HeapSetInformation	0000713C	0000827E	Hint/Name RVA	02D4	HeapSize
000070A0	00007F8C	Hint/Name RVA	02E3	InitializeCriticalSectionAndSpinCount	00007140	0000828A	Hint/Name RVA	0418	RtlUnwind
000070A4	00007F94	Hint/Name RVA	00D1	DeleteCriticalSection	00007144	00008296	Hint/Name RVA	02CB	HeapAlloc
000070A8	00007FCC	Hint/Name RVA	00EE	EnterCriticalSection	00007148	000082A2	Hint/Name RVA	02D2	HeapReAlloc
000070AC	00007FE4	Hint/Name RVA	00EA	EncodePointer	0000714C	000082B0	Hint/Name RVA	0304	IsProcessorFeaturePresent
000070B0	00007FFC	Hint/Name RVA	0202	GetLastError	00007150	000082CC	Hint/Name RVA	032D	LCMapStringW
000070B4	0000801C	Hint/Name RVA	033F	LoadLibraryW	00007154	000082DC	Hint/Name RVA	0367	MultiByteToWideChar
000070B8	0000802C	Hint/Name RVA	04D3	UnhandledExceptionFilter	00007158	000082F2	Hint/Name RVA	0269	GetStringTypeW
000070BC	00008048	Hint/Name RVA	04A5	SetUnhandledExceptionFilter	0000715C	00000000	End of Imports		KERNEL32.dll
000070C0	00008066	Hint/Name RVA	0300	IsDebuggerPresent					
000070C4	0000807A	Hint/Name RVA	04C0	TerminateProcess					
000070C8	0000808E	Hint/Name RVA	04C5	TlsAlloc					
000070CC	0000809A	Hint/Name RVA	04C7	TlsGetValue					
000070D0	000080A8	Hint/Name RVA	04C8	TlsSetValue					
000070D4	000080B6	Hint/Name RVA	04C8	TlsFree					

ShellExecuteA indicates that the malware will attempt to execute terminal commands.

AdjustTokenPrivileges indicates that the malware will attempt to change privileges of resources.

CreateFileA and WriteFile also indicate that the malware will try to create a new file.

4. What host- or network-based indicators can you use to identify the malware on infected Machines?

```
explorer.exe
EnumProcessModules
psapi.dll
GetModuleBaseNameA
EnumProcesses
freddy01x.dll
X64DLL
freddy01x.exe
X64
/c
cmd.exe
open
```

Running `strings` show us these file names. `freddy01.dll`

Some host indicators could be the existence of `freddy01x.dll`, `freddy01.dll` and `freddy01x.exe`. Moreover, `explorer.exe` is a typical process for malware to inject themselves into thus if `explorer.exe` is taking too much resources when running. Maybe any unexpected shell commands in the terminal history as well.

5. The file has multiple resources in its resource section. What are their respective MD5 or SHA hashes? What are the differences between the resources? [Hint: Resources are usually in BIN format]

BIN X64:	<input checked="" type="checkbox"/> MD5	6b392635239b0d73a83d4ae2c2c4e379
BIN X86:	<input checked="" type="checkbox"/> MD5	a6fb0d8fdea1c15afb7a5ddb3d2867b
X64DLL:	<input checked="" type="checkbox"/> MD5	c2e078a662c8ae7bb98bc2d66b0d89d7

The differences could be resources for the different types of architecture, X86 or X64.

Task 3 (5 marks, 1 mark for each question): Answer the following questions by analyzing HW-A-3.exe using basic static and dynamic analysis techniques only.

1. What is this program's functionality and explain the basis for this guess?

This could be a trojan file. This is because the exe file has the icon for an image file, to disguise its true file type.

2. What are your observations about the program using Basic Static Analysis techniques?

68 / 72

68/72 security vendors and 1 sandbox flagged this file as malicious

Reanalyze Similar More

9edc610285e3e2ca964d359d01b2151df3f73333c69d2765...

Size: 290.50 KB

Last Modification Date: 28 minutes ago

EXE

peexe long sleeps detect-debug-environment persistence checks-user-input checks-usb-bus

DETECTION DETAILS RELATIONS BEHAVIOR TELEMETRY COMMUNITY 11+

Join the VT Community and enjoy additional community insights and crowdsourced detections, plus an API key to automate checks.

Popular threat label: trojan.llac/barys

Threat categories: trojan worm

Family labels: llac barys rebhip

```
Software\Microsoft\Windows\CurrentVersion
ProductId
55274-640-2673064-23950
hTT@
```

```
SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\Shell Folders
```

This could show that the malware will try to view and change registry keys. Especially with imports such as `RegSetValueExA` or `RegCreateKeyExA` and `RegCreateKeyA`.

Moreover, `XX--XX--XX.txt` or similar files might be created. Especially with imports such as `CreateFileA`, `WriteFile` and `CreateDirectoryA`.

3. What are your observations about the program through dynamic analysis?

HW-A-3 - Copy.exe	7804	1.25	7.98 kB/s	6.36 MB	MSEDGEWIN10\IEUser
server32.exe	3924			1.57 MB	MSEDGEWIN10\IEUser
server32.exe	3164			224 kB	MSEDGEWIN10\IEUser
4:27:3... HW-A-3 - Copy....	7968	Load Image	C:\Windows\System32\user32.dll	SUCCESS	Image base: 0x000...
4:27:3... HW-A-3 - Copy....	7968	Process Create	C:\Users\IEUser\AppData\Roaming\ins...	SUCCESS	PID: 4364, Comma...
4:27:3... server32.exe	4364	Process Start		SUCCESS	Parent PID: 7968, ...
4:27:3... server32.exe	4364	Thread Create		SUCCESS	Thread ID: 1960

There is a creation of a process called `server32.exe`

Moreover, there is a creation of the file XxX.xXx

Using `apatedns`, we can also see that the malware is trying to connect to the above domains.

Using regshot to compare, there are many changes and the picture below focuses only on the keys added.

```
HKLM\SOFTWARE\Microsoft\Windows\Windows Error Reporting\TermReason\7684
HKLM\SOFTWARE\Wow6432Node\Microsoft\Tracing\server32_RASAPI32
HKU\S-1-5-21-3461203602-4096304019-2269080069-1000\Software\Microsoft\Speech_OneCore\Isolated\N8QQ08uXrWkuVsSWU0SWhuEuISxhkQU1'
HKU\S-1-5-21-3461203602-4096304019-2269080069-1000\Software\Microsoft\Speech_OneCore\Isolated\N8QQ08uXrWkuVsSWU0SWhuEuISxhkQU1'
HKU\S-1-5-21-3461203602-4096304019-2269080069-1000\Software\Microsoft\Speech_OneCore\Isolated\N8QQ08uXrWkuVsSWU0SWhuEuISxhkQU1'
HKU\S-1-5-21-3461203602-4096304019-2269080069-1000\Software\Microsoft\Speech_OneCore\Isolated\N8QQ08uXrWkuVsSWU0SWhuEuISxhkQU1'
HKU\S-1-5-21-3461203602-4096304019-2269080069-1000\Software\Microsoft\Speech_OneCore\Isolated\N8QQ08uXrWkuVsSWU0SWhuEuISxhkQU1'
HKU\S-1-5-21-3461203602-4096304019-2269080069-1000\Software\Microsoft\Speech_OneCore\Isolated\N8QQ08uXrWkuVsSWU0SWhuEuISxhkQU1'
HKU\S-1-5-21-3461203602-4096304019-2269080069-1000\Software\Microsoft\Speech_OneCore\Isolated\N8QQ08uXrWkuVsSWU0SWhuEuISxhkQU1'
HKU\S-1-5-21-3461203602-4096304019-2269080069-1000\Software\Microsoft\Windows\CurrentVersion\Explorer\SessionInfo\1\Applicat:
HKU\S-1-5-21-3461203602-4096304019-2269080069-1000\Software\Microsoft\Windows\CurrentVersion\Search\JumplistData
HKU\S-1-5-21-3461203602-4096304019-2269080069-1000\Software\Classes\Local Settings\Software\Microsoft\Windows\CurrentVersion'
HKU\S-1-5-21-3461203602-4096304019-2269080069-1000\Software\Classes\Local Settings\Software\Microsoft\Windows\CurrentVersion'
HKU\S-1-5-21-3461203602-4096304019-2269080069-1000\Software\Classes\Local Settings\Software\Microsoft\Windows\CurrentVersion'
HKU\S-1-5-21-3461203602-4096304019-2269080069-1000\Software\Classes\Local Settings\Software\Microsoft\Windows\CurrentVersion'
HKU\S-1-5-21-3461203602-4096304019-2269080069-1000\Software\Classes\Local Settings\Software\Microsoft\Windows\CurrentVersion'
HKU\S-1-5-21-3461203602-4096304019-2269080069-1000\Software\Classes\Local Settings\Software\Microsoft\Windows\CurrentVersion'
HKU\S-1-5-21-3461203602-4096304019-2269080069-1000\Software\abc
HKU\S-1-5-21-3461203602-4096304019-2269080069-1000_Classes\Local Settings\Software\Microsoft\Windows\CurrentVersion\AppDataCont:
HKU\S-1-5-21-3461203602-4096304019-2269080069-1000_Classes\Local Settings\Software\Microsoft\Windows\CurrentVersion\AppDataCont:
HKU\S-1-5-21-3461203602-4096304019-2269080069-1000_Classes\Local Settings\Software\Microsoft\Windows\CurrentVersion\AppDataCont:
HKU\S-1-5-21-3461203602-4096304019-2269080069-1000_Classes\Local Settings\Software\Microsoft\Windows\CurrentVersion\AppDataCont:
HKU\S-1-5-21-3461203602-4096304019-2269080069-1000_Classes\Local Settings\Software\Microsoft\Windows\CurrentVersion\AppDataCont:
```

4. List the potential host-based of this malware.

The existence of the file in the `User\AppData\Temp\XxX.xXx.`

The malware can also be running as a process, called `server32.exe`.

We can also find if registry keys stated above are created, or the other registry keys changed.

5. List the potential network-based indicators of this malware? To which domains does the malware possibly connect?

02:20:46	www.atwushere.net
02:20:52	www.itisreal.edu
02:20:59	www.atwushere.net
02:21:05	www.itisreal.edu
02:21:13	www.atwushere.net
02:21:19	www.itisreal.edu
02:21:26	www.atwushere.net
02:21:33	www.itisreal.edu

Connection history to these sites can be used as network indicators.

PE File Format (5 marks)

PEfile Usage Examples: <https://github.com/erocarrera/pefile/blob/wiki/UsageExamples.md>

Task 4 (5 marks, 1 mark for each question): Write a Python program that uses the pefile API (<https://code.google.com/p/pefile/>). The program takes a PE file as input from the command line, and should perform the operations below. We have provided a template Python program (A3.py) that you can use to get started. Note that the template is provided as a reference and you should not rely on its implementation correctness, although we have ensured this to some extent. You may modify or completely rewrite the template if you wish. (Note: In to answering the questions below, please attach screenshots of the results in your report. Additionally, include your code in a single Python file inside your submitted zip file.)

1. Write a program to output the following to standard output:
 - a. Identify the file type as DLL, EXE, or SYS regardless of the file's extension. [Answer to this is provided in A3.py for you to get started]
 - b. The total number of imported DLLs.
 - c. The total number of imported functions.
 - d. The compile time.
2. Alert the user if the code's entry point is not in a section with the name .text, .code, CODE, or INIT. (Hint: Aforementioned usage examples may have some relevant code snippets that you can use. You can consider using `pe.OPTIONAL_HEADER.AddressOfEntryPoint` to get the address of the entry point. You can use `section.contains_rva()` for your checking.)
3. Use the PEiD database that comes with pefile to identify packers. Confirm that this works with UPX. Output the detection to standard output.
4. Calculate and output the entropy for each section. Alert the user if there is a suspicion that a section may be packed or compressed (if the section's entropy ≥ 6).
5. Compare the PE Optional Header checksum with the actual checksum. Alert the user when they do not match up.

```
C:\Users\IEUser\Desktop>python A3.py HW-A-3.ex_
Assignment 3: A0216695U Edward Ng
[1]   Analysing file HW-A-3.ex_
[2.a]   File type: PE
[2.b]   #DLLs: 16
[2.b]   #FNs: 113
[3]   Compile Time: 0x2A425E19 [Fri Jun 19 22:22:17 1992 UTC]
[4]   Entry point found in CODE
[5]   Packer detected: None
[6]   Entropy for sections...
      CODE : 6.41 (Packed?)
      DATA : 2.76
      BSS : 0.00
      .idata : 4.77
      .tls : 0.00
      .rdata : 0.21
      .reloc : 6.25 (Packed?)
      .rsrc : 7.96 (Packed?)
[8]   Checksum matched: False

C:\Users\IEUser\Desktop>
```