

# HW3 Combined

## Analyzing HW-A-1.ex

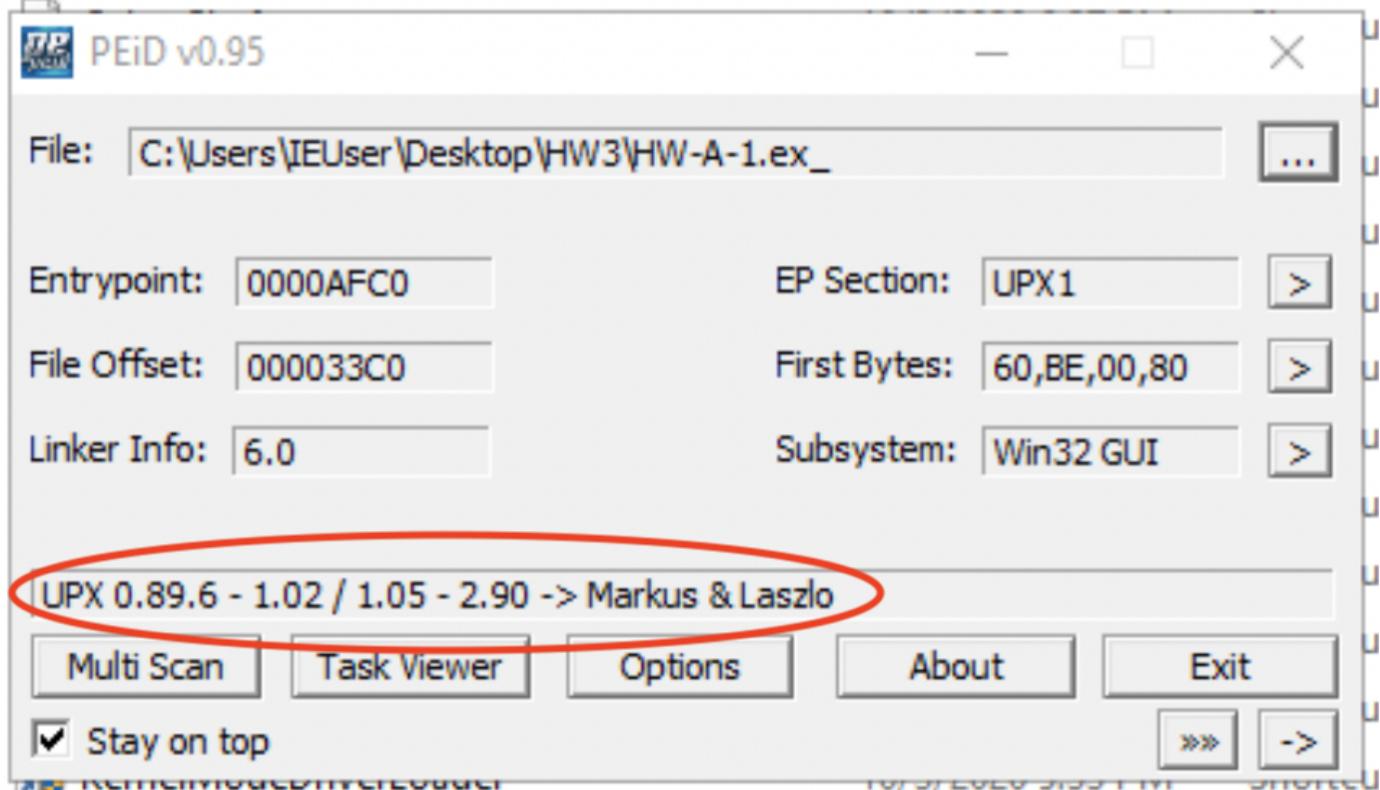
### HW-A-1.ex

1. Upload the file to <https://www.virustotal.com>. Does it match any existing antivirus definitions?

Yes, it matches 58/68 existing antivirus definitions.

DETECTION	DETAILS	RELATIONS	BEHAVIOR	COMMUNITY
Acronis (Static ML)	⚠ Suspicious			Ad-Aware ⚡ Trojan.Generic.15020151
AhnLab-V3	⚠ Trojan/Win32.Gen			Alibaba ⚡ Trojan:Win32/DocThief.d0928ec7
ALYac	⚠ Trojan.Generic.15020151			Antiy-AVL ⚡ Trojan/Generic.ASMalwS.14B3D7E
SecureAge APEX	⚠ Malicious			Arcabit ⚡ Trojan.Generic.DE53077
Avast	⚠ FileRepMalware			AVG ⚡ FileRepMalware
Avira (no cloud)	⚠ TR/Downloader.Gen			Baidu ⚡ Win32.Trojan.Agent.aqu
BitDefender	⚠ Trojan.Generic.15020151			BitDefenderTheta ⚡ Gen>NN.ZexxF.34218.amGfaulDCQk
Bkav Pro	⚠ W32.AIDetect.malware2			CAT-QuickHeal ⚡ Trojan.Docthief
Comodo	⚠ NetWorm.Win32.Koobface.FY@3sikb7			CrowdStrike Falcon ⚡ Win/malicious_confidence_100% (W)
Cylance	⚠ Unsafe			Cynet ⚡ Malicious (score: 100)
eGambit	⚠ Unsafe.AI_Score_99%			Elastic ⚡ Malicious (high Confidence)
Emsisoft	⚠ Trojan.Generic.15020151(B)			eScan ⚡ Trojan.Generic.15020151
ESET-NOD32	⚠ Win32/Agent.WON			FireEye ⚡ Generic.mg.7722227cae5b9192
Fortinet	⚠ W32/Agent.WON!tr			GData ⚡ Trojan.Generic.15020151
Ikarus	⚠ Trojan.Win32.Agent			Jiangmin ⚡ Trojan/DocThief.a
K7AntiVirus	⚠ Trojan ( 0055e3dd1 )			K7GW ⚡ Trojan ( 0055e3dd1 )
Kaspersky	⚠ Trojan.Win32.DocThief.a			Kingsoft ⚡ Win32.Troj.DocThief.a.(kcloud)
Lionic	⚠ Trojan.Win32.DocThief.4lc			MAX ⚡ Malware (ai Score=100)
MaxSecure	⚠ Trojan.Malware.300983.susgen			McAfee ⚡ RDN/Generic.grp
McAfee-GW-Edition	⚠ BehavesLike.Win32.Generic.lc			Microsoft ⚡ Trojan:Win32/Occamy.C57
NANO-Antivirus	⚠ Trojan.Win32.ThreatHLLSbased.dewzgf			Palo Alto Networks ⚡ Generic.ml
Panda	⚠ Trj/Genetic.gen			Sangfor Engine Zero ⚡ Suspicious.Win32.Save.a
SentinelOne (Static ML)	⚠ Static AI - Malicious PE			Sophos ⚡ Mal/Generic-S
Symantec	⚠ Trojan.Gen			TACHYON ⚡ Trojan/W32.DocThief.32768
Tencent	⚠ Win32.Trojan.Docthief.Agba			TrendMicro ⚡ Trojan:Win32.DOCTHIEF.A
TrendMicro-HouseCall	⚠ Trojan.Win32.DOCTHIEF.A			VBA32 ⚡ Trojan.DocThief
VIPRE	⚠ Trojan.Win32.Generic!BT			Yandex ⚡ Trojan.GenAsa!Y8Q6MbdNgc
Zillya	⚠ Trojan.DocThief.Win32.2			ZoneAlarm by Check Point ⚡ Trojan.Win32.DocThief.a

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



The PEiD has identified the program to be packed with UPX version 0.89 - 1.02 / 1.05-2.90.

To unpack, we use the command: "upx -d HW-A-1.ex\_" downloaded from [UPX]

```
C:\Users\IEUser\Desktop\HW3>upx -d HW-A-1.ex_
    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.ex_

Unpacked 1 file.
```

3. When was this program compiled?

It was compiled on 2011/11/17 Thu 17:58:55 UTC

PEview - C:\Users\IEUser\Desktop\HW3\HW-A-1.ex\_

File View Go Help

HW-A-1.ex\_

	pFile	Data	Description	Value
IMAGE_DOS_HEADER	000000E4	014C	Machine	IMAGE_FILE_MACHINE_I386
MS-DOS Stub Program	000000E6	0003	Number of Sections	
IMAGE_NT_HEADERS	000000E8	4EC54B5F	Time Date Stamp	2011/11/17 Thu 17:58:55 UTC
Signature	000000EC	00000000	Pointer to Symbol Table	
IMAGE_FILE_HEADER	000000F0	00000000	Number of Symbols	
IMAGE_OPTIONAL_HEADER	000000F4	00E0	Size of Optional Header	
IMAGE_SECTION_HEADER	000000F6	010F	Characteristics	
			0001	IMAGE_FILE_RELOCS_STRIPPED
			0002	IMAGE_FILE_EXECUTABLE_IMAGE
			0004	IMAGE_FILE_LINE_NUMS_STRIPPED
			0008	IMAGE_FILE_LOCAL_SYMS_STRIPPED
			0100	IMAGE_FILE_32BIT_MACHINE

Viewing IMAGE\_FILE\_HEADER

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

The DLL imported and their corresponding function:

KERNEL32.DLL: manipulates memory, files, and hardware

WININET.dll: Has networking function (ex: FTP, HTTP, NTP)

WS2\_32.dll: Networking DLL --> the malware has network related task

Putting it together, it seems that the malware manipulates memory, file, or hardware then sends and receive data from the attacker.

CFF Explorer VIII - [HW-A-1.ex\_]

The screenshot shows the CFF Explorer interface with the file HW-A-1.ex\_ open. The left sidebar shows the file structure. The main window displays the imports from the depends.dll library. The table has columns: Module Name, Imports, OFTs, TimeDateStamp, ForwarderChain, Name RVA, and FTs (IAT). The imports listed are:

Module Name	Imports	OFTs	TimeDateStamp	ForwarderChain	Name RVA	FTs (IAT)
000065EC	N/A	000064C4	000064C8	000064CC	000064D0	000064D4
szAnsi	(nFunctions)	Dword	Dword	Dword	Dword	
KERNEL32.DLL	44	00000000	00000000	00000000	000065EC	00006000
WININET.dll	5	00000000	00000000	00000000	000065F9	000060B4
WS2_32.dll	2	00000000	00000000	00000000	00006605	000060CC

Below the table is a detailed view of the szAnsi imports from WININET.dll:

OFTs	FTs (IAT)	Hint	Name
Dword	Dword	Word	szAnsi
N/A	00006610	0000	FindNextFileA
N/A	00006620	0000	FindClose
N/A	0000662C	0000	FindFirstFileA
N/A	0000663C	0000	FlushFileBuffers
N/A	0000664E	0000	GetStringTypeW
N/A	0000665E	0000	GetStringTypeA
N/A	0000666E	0000	LCMMapStringW
N/A	0000667C	0000	LCMMapStringA
N/A	0000668A	0000	MultiByteToWideChar
N/A	000066A0	0000	SetStdHandle
N/A	000066AE	0000	LoadLibraryA
N/A	000066BC	0000	GetProcAddress

There are 5 functions imported from WININET.dll.

CFF Explorer VIII - [HW-A-1.ex\_]

The screenshot shows the CFF Explorer interface with the file HW-A-1.ex\_ open. The left sidebar shows the file structure. The main window displays the imports from the depends.dll library. The table has columns: Module Name, Imports, OFTs, TimeDateStamp, ForwarderChain, Name RVA, and FTs (IAT). The imports listed are:

Module Name	Imports	OFTs	TimeDateStamp	ForwarderChain	Name RVA	FTs (IAT)
000065F9	N/A	000064D8	000064DC	000064E0	000064E4	000064E8
szAnsi	(nFunctions)	Dword	Dword	Dword	Dword	
KERNEL32.DLL	44	00000000	00000000	00000000	000065EC	00006000
WININET.dll	5	00000000	00000000	00000000	000065F9	000060B4
WS2_32.dll	2	00000000	00000000	00000000	00006605	000060CC

Below the table is a detailed view of the szAnsi imports from WININET.dll:

OFTs	FTs (IAT)	Hint	Name
Dword	Dword	Word	szAnsi
N/A	000068D2	0000	InternetCloseHandle
N/A	000068E8	0000	FtpPutFileA
N/A	000068F6	0000	InternetOpenA
N/A	00006906	0000	InternetConnectA
N/A	00006918	0000	FtpSetCurrentDirectoryA

From the functions, it seems that the malware opens a FTP connection, put a malicious file, set the current directory into the malicious file.

There are 2 functions imported from WS2\_32.dll

This DLL helps assist with the creating and maintaining the FTP connection.

There are 44 functions imported from KERNEL32.DLL

The screenshot shows the CFF Explorer VIII interface with the file 'HW-A-1.exe' open. The 'depends.dll' tab is selected, showing the imports of the executable. The table lists the module name, imports, OFTs, TimeStamp, ForwarderChain, Name RVA, and FTs (IAT). The 'szAnsi' section highlights 44 imports from KERNEL32.DLL. Below this, a detailed list of these imports is provided:

OFTs	FTs (IAT)	Hint	Name
Dword	Dword	Word	szAnsi
N/A	00006610	0000	FindNextFileA
N/A	00006620	0000	FindClose
N/A	0000662C	0000	FindFirstFileA
N/A	0000663C	0000	FlushFileBuffers
N/A	0000664E	0000	GetStringTypeW
N/A	0000665E	0000	GetStringTypeA
N/A	0000666E	0000	LCMapStringW
N/A	0000667C	0000	LCMapStringA
N/A	0000668A	0000	MultiByteToWideChar
N/A	000066A0	0000	SetStdHandle
N/A	000066AE	0000	LoadLibraryA
N/A	000066BC	0000	GetProcAddress

MSEdge - Win10 1 [Running]

CFF Explorer VIII - [HW-A-1.ex\_]

File Settings ?

depends.dll HW-A-1.ex\_

**File: HW-A-1.ex\_**

- Dos Header
- Nt Headers
  - File Header
  - Optional Header
    - Data Directories [x]
- Section Headers [x]
- Import Directory

**Address Converter**

**Dependency Walker**

**Hex Editor**

**Identifier**

**Import Adder**

**Quick Disassembler**

**Rebuilder**

**Resource Editor**

**UPX Utility**

Module Name	Imports	OFTs	TimeDateStamp	ForwarderChain	Name RVA	FTs (IAT)
000065EC	N/A	000064C4	000064C8	000064CC	000064D0	000064D4
szAnsi	(nFunctions)	Dword	Dword	Dword	Dword	Dword
KERNEL32.DLL	44	00000000	00000000	00000000	000065EC	00006000
WININET.dll	5	00000000	00000000	00000000	000065F9	000060B4
WS2_32.dll	2	00000000	00000000	00000000	00006605	000060CC

OFTs	FTs (IAT)	Hint	Name
Dword	Dword	Word	szAnsi
N/A	000066CC	0000	HeapAlloc
N/A	000066D8	0000	GetModuleHandleA
N/A	000066EA	0000	GetStartupInfoA
N/A	000066FC	0000	GetCommandLineA
N/A	0000670E	0000	GetVersion
N/A	0000671A	0000	ExitProcess
N/A	00006728	0000	HeapDestroy
N/A	00006736	0000	HeapCreate
N/A	00006742	0000	VirtualFree
N/A	00006750	0000	HeapFree
N/A	0000675A	0000	VirtualAlloc
N/A	00006768	0000	HeapReAlloc

CFF Explorer VIII - [HW-A-1.ex\_]

File Settings ?

depends.dll HW-A-1.ex\_

**File: HW-A-1.ex\_**

- Dos Header
- Nt Headers
  - File Header
  - Optional Header
    - Data Directories [x]
- Section Headers [x]
- Import Directory

**Address Converter**

**Dependency Walker**

**Hex Editor**

**Identifier**

**Import Adder**

**Quick Disassembler**

**Rebuilder**

**Resource Editor**

**UPX Utility**

Module Name	Imports	OFTs	TimeDateStamp	ForwarderChain	Name RVA	FTs (IAT)
000065EC	N/A	000064C4	000064C8	000064CC	000064D0	000064D4
szAnsi	(nFunctions)	Dword	Dword	Dword	Dword	Dword
KERNEL32.DLL	44	00000000	00000000	00000000	000065EC	00006000
WININET.dll	5	00000000	00000000	00000000	000065F9	000060B4
WS2_32.dll	2	00000000	00000000	00000000	00006605	000060CC

OFTs	FTs (IAT)	Hint	Name
Dword	Dword	Word	szAnsi
N/A	00006776	0000	TerminateProcess
N/A	00006788	0000	GetCurrentProcess
N/A	0000679C	0000	UnhandledExceptionFilter
N/A	00006786	0000	GetModuleFileNameA
N/A	000067CA	0000	FreeEnvironmentStringsA
N/A	000067E4	0000	FreeEnvironmentStringsW
N/A	000067FE	0000	WideCharToMultiByte
N/A	00006814	0000	GetEnvironmentStrings
N/A	0000682C	0000	GetEnvironmentStringsW
N/A	00006844	0000	SetHandleCount
N/A	00006854	0000	GetStdHandle
N/A	00006862	0000	GetFileType

CFF Explorer VIII - [HW-A-1.ex\_]

The screenshot shows the CFF Explorer interface with the file HW-A-1.ex\_ open. The left sidebar contains navigation links like File, Settings, and various tools such as Address Converter, Dependency Walker, Hex Editor, Identifier, Import Adder, Quick Disassembler, Rebuilder, Resource Editor, and UPX Utility. The main window has tabs for 'depends.dll' and 'HW-A-1.ex\_'. The 'HW-A-1.ex\_' tab is active, displaying a table of imports. The table has columns for Module Name, Imports, OFTs, TimeDateStamp, ForwarderChain, Name RVA, and FTs (IAT). The imports listed are:

Module Name	Imports	OFTs	TimeDateStamp	ForwarderChain	Name RVA	FTs (IAT)
000065EC	N/A	000064C4	000064C8	000064CC	000064D0	000064D4
szAnsi	(nFunctions)	Dword	Dword	Dword	Dword	Dword
KERNEL32.DLL	44	00000000	00000000	00000000	000065EC	00006000
WININET.dll	5	00000000	00000000	00000000	000065F9	000060B4
WS2_32.dll	2	00000000	00000000	00000000	00006605	000060CC

Below this table is another table showing the details of the szAnsi import:

OFTs	FTs (IAT)	Hint	Name
Dword	Dword	Word	szAnsi
N/A	0000682C	0000	GetEnvironmentStringsW
N/A	00006844	0000	SetHandleCount
N/A	00006854	0000	GetStdHandle
N/A	00006862	0000	GetFileType
N/A	00006870	0000	RtlUnwind
N/A	0000687C	0000	WriteFile
N/A	00006888	0000	GetLastError
N/A	00006896	0000	SetFilePointer
N/A	000068A6	0000	GetCPIInfo
N/A	000068B2	0000	GetACP
N/A	000068BA	0000	GetOEMCP
N/A	000068C4	0000	CloseHandle

In the KERNEL32.DLL, we see the functions used to open and manipulate files including the FindFirstFile and FindNextFile. It indicates that the malware searches through the filesystem, and it can open and modify files.

## 5. What host or network-based indicators could be used to identify this malware on infected machines?

It has WINNET.dll therefore it has networking functions.

We can check if there is an FTP connection on the infected machine.

The screenshot shows the IDA Pro interface with the title "IDA - HW-A-1.ex\_ C:\Users\IEUser\Desktop\HW3\HW-A-1.ex\_". The main window displays a table of functions:

Function name	Address	Ordinal	Name	Library
sub_404720	0000000000406078		WideCharToMultiByte	KERNEL32
sub_40481E	000000000040607C		GetEnvironmentStrings	KERNEL32
sub_404885	0000000000406080		GetEnvironmentStringsW	KERNEL32
sub_4048FF	0000000000406084		SetHandleCount	KERNEL32
sub_40493C	0000000000406088		GetStdHandle	KERNEL32
sub_404A11	000000000040608C		GetFileType	KERNEL32
sub_404A4C	0000000000406090		RtlUnwind	KERNEL32
sub_404AA8	0000000000406094		WriteFile	KERNEL32
sub_404AB1	0000000000406098		GetLastError	KERNEL32
sub_404B1E	000000000040609C		SetFilePointer	KERNEL32
sub_404B27	00000000004060A0		GetCPInfo	KERNEL32
sub_404D4B	00000000004060A4		GetACP	KERNEL32
sub_404D76	00000000004060A8		GetOEMCP	KERNEL32
sub_404EC0	00000000004060AC		CloseHandle	KERNEL32
sub_404F18	00000000004060B4		InternetCloseHandle	WININET
sub_404F6E	00000000004060B8		FtpPutFileA	WININET
_alloca_probe	00000000004060BC		InternetOpenA	WININET
sub_404FFF	00000000004060C0		InternetConnectA	WININET
sub_4050B2	00000000004060C4		FtpSetCurrentDirectoryA	WININET
RtlUnwind	00000000004060CC	115	_imp_WSASStartup	WS2_32
	00000000004060D0	57	_imp_gethostname	WS2_32

From the command "strings HW-A-1.ex\_" we can see that it calls for the string "ftp.practicalmalwareanalysis.net"

The screenshot shows the Windows PowerShell ISE interface with the title "Administrator: Windows PowerShell ISE". The PowerShell window displays the output of the "strings" command:

```
C:\> strings HW-A-1.ex_
CloseHandle
InternetCloseHandle
FtpPutFileA
InternetOpenA
InternetConnectA
FtpSetCurrentDirectoryA
.pdf
.doc
%<%d.pdf
pdfs
ftp.practicalmalwareanalysis.net
Home ftp client
%<%d.doc
docs
C:\*
&@_
.a@
Pa@
tc@
Pc@
$c@
`l@_
Db@
|a@
xa@
ha@
~@
~~~~~ H
PS C:\Users\IEUser\Desktop\HW3>
```

The output shows various file extensions and the string "ftp.practicalmalwareanalysis.net" which is highlighted in yellow.

## Analyzing HW-A-2.ex

Task 2: Answer the following questions after analyzing HW-A-2.exe

**1. Upload the file to <https://www.virustotal.com>. Does it match any existing antivirus definitions?**

Yes. It matches 46/68 existing antivirus definition.

The screenshot shows the VirusTotal analysis interface. At the top, a circular progress bar indicates a 'Community Score' of 46 / 68. Below this, the file hash is listed as 34d22db9c85069a1ff40917c319ba7a65097cd1d77ea250e2a6fe5a5b53bdfb9, and the file name is HW-A-2.exe\_. The file size is 172.00 KB, and it was analyzed on 2021-10-25 03:55:19 UTC, 23 hours ago. The file type is EXE. The main table lists 46 vendor detections, each with a vendor name, detection name, and a brief description. The table has columns for DETECTION, DETAILS, RELATIONS, BEHAVIOR, and COMMUNITY. The COMMUNITY column shows the number of detections (1). The table includes entries from Ad-Aware, Alibaba, Antiy-AVL, Arcabit, AVG, BitDefender, Comodo, Cynet, Elastic, eScan, FireEye, GData, K7AntiVirus, Kingsoft, MAX, McAfee, Palo Alto Networks, Sangfor Engine Zero, Sophos, Tencent, TrendMicro-HouseCall, and VIPRE.

DETECTION	DETAILS	RELATIONS	BEHAVIOR	COMMUNITY
Ad-Aware	① Trojan.GenericKD.2728766		AhnLab-V3	① Trojan/Win32.Inject.C1318505
Alibaba	① Trojan:Win32/Invader.39233a27		ALYac	① Trojan.GenericKD.2728766
Antiy-AVL	① Trojan/Generic.ASMalwS.B7022F		SecureAge APEX	① Malicious
Arcabit	① Trojan.Generic.D29A33E		Avast	① Win32:Malware-gen
AVG	① Win32:Malware-gen		Avira (no cloud)	① TR/Dropper.Gen
BitDefender	① Trojan.GenericKD.2728766		BitDefenderTheta	① AI:Packer.DE75EAB320
Comodo	① Malware@#ckh8u07jk971		CrowdStrike Falcon	① Win/malicious_confidence_100% (W)
Cynet	① Malicious (score: 100)		eGambit	① Unsafe.AI_Score_97%
Elastic	① Malicious (high Confidence)		Emsisoft	① Trojan.GenericKD.2728766 (B)
eScan	① Trojan.GenericKD.2728766		ESET-NOD32	① A Variant Of Win32/Agent.WOK
FireEye	① Generic.mg.d93adc942c53bdb9		Fortinet	① PossibleThreat.SBltr
GData	① Trojan.GenericKD.2728766		Ikarus	① Win32.SuspectCrc
K7AntiVirus	① Trojan ( 0000000c1 )		K7GW	① Trojan ( 0000000c1 )
Kingsoft	① Win32.Heur.KVM003.a.(kcloud)		Lionic	① Trojan.Win32.Invader.4lc
MAX	① Malware (ai Score=100)		MaxSecure	① Trojan.Malware.300983.susgen
McAfee	① GenericR-EMOID93ADC942C53		NANO-Antivirus	① Trojan.Win32.Invader.dllbjyj
Palo Alto Networks	① Generic.ml		Panda	① TrjCl.A
Sangfor Engine Zero	① Trojan.Win32.Save.a		SentinelOne (Static ML)	① Static AI - Malicious PE
Sophos	① Generic ML PUA (PUA)		Symantec	① ML.Attribute.HighConfidence
Tencent	① Malware.Win32.Gencirc.114bec55		TrendMicro	① TROJ_GEN.R002COPJI21
TrendMicro-HouseCall	① TROJ_GEN.R002COPJI21		VBA32	① Trojan.Invader
VIPRE	① Trojan.Win32.Generic!BT		Yandex	① Trojan.GenAsalBd6JlhQ2Vvo

**2. When was this program compiled?**

It was compiled on 2011/11/22 Tue 03:35:10 UTC

PEview - C:\Users\IEUser\Desktop\HW3\HW-A-2.ex\_

	pFile	Data	Description	Value
HW-A-2.ex_	000000F4	014C	Machine	IMAGE_FILE_MACHINE_I386
-- IMAGE_DOS_HEADER	000000F6	0004	Number of Sections	
-- MS-DOS Stub Program	000000F8	4ECB186E	Time Date Stamp	2011/11/22 Tue 03:35:10 UTC
-- IMAGE_NT_HEADERS	000000FC	00000000	Pointer to Symbol Table	
-- Signature	00000100	00000000	Number of Symbols	
-- IMAGE_FILE_HEADER	00000104	00E0	Size of Optional Header	
-- IMAGE_OPTIONAL_HEADER	00000106	0103	Characteristics	
-- IMAGE_SECTION_HEADER		0001		IMAGE_FILE_RELOCS_STRIPPED
-- IMAGE_SECTION_HEADER		0002		IMAGE_FILE_EXECUTABLE_IMAGE
-- IMAGE_SECTION_HEADER		0100		IMAGE_FILE_32BIT_MACHINE
-- SECTION .text				
-- SECTION .rdata				
-- SECTION .data				
-- SECTION .rsrc				

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

HW-A-2.ex\_

Module Name	Imports	OFTs	TimeStamp	ForwarderChain	Name RVA	FTs (IAT)	
szAnsi	(nFunctions)	Dword	Dword	Dword	Dword	Dword	
KERNEL32.dll	72	00007C3C	00000000	00000000	00007EB6	00006010	
ADVAPI32.dll	3	00007C2C	00000000	00000000	00007F08	00006000	
SHELL32.dll	1	00007D60	00000000	00000000	00007F26	00006134	

The KERNEL32.dll import tells me that the malware manipulates memory, files, and hardware.

Dependency Walker - [HW-A-2.ex\_]

File Edit View Options Profile Window Help

HW-A-2.EX\_

PI	Ordinal ^	Hint	Function	Entry Point
	N/A	82 (0x0052)	CloseHandle	Not Bound
	N/A	136 (0x0088)	CreateFileA	Not Bound
	N/A	169 (0x00A9)	CreateRemoteThread	Not Bound
	N/A	202 (0x00CA)	DecodePointer	Not Bound
	N/A	209 (0x00D1)	DeleteCriticalSection	Not Bound
	N/A	234 (0x00EA)	EncodePointer	Not Bound
	N/A	238 (0x00EE)	EnterCriticalSection	Not Bound
	N/A	281 (0x0119)	ExitProcess	Not Bound

E	Ordinal ^	Hint	Function	Entry Point
	1 (0x0001)	0 (0x0000)	AcquireSRWLockExclusive	NTDLL.RtlAcquireSRWLockExclusive
	2 (0x0002)	1 (0x0001)	AcquireSRWLockShared	NTDLL.RtlAcquireSRWLockShared
	3 (0x0003)	2 (0x0002)	ActivateActCtx	0x0001E860
	4 (0x0004)	3 (0x0003)	ActivateActCtxWorker	0x0001A570
	5 (0x0005)	4 (0x0004)	AddAtomA	0x0021A70
	6 (0x0006)	5 (0x0005)	AddAtomW	0x0000FE30
	7 (0x0007)	6 (0x0006)	AddConsoleAliasA	0x00022DE0

Module	File Time Stamp	Link Time Stamp	File Size	Attr.	Link Checksum	Real Checksum
API-MS-WIN-CORE-APIQUERY-L1-1-0.DLL						
API-MS-WIN-CORE-APIQUERY-L1-1-1.DLL						
API-MS-WIN-CORE-APPCOMPAT-L1-1-0.DLL						
API-MS-WIN-CORE-APPCOMPAT-L1-1-1.DLL						
API-MS-WIN-CORE-APPINIT-L1-1-0.DLL						
API-MS-WIN-CORE-ATOMS-L1-1-0.DLL						
API-MS-WIN-CORE-COMM-L1-1-0.DLL						
API-MS-WIN-CORE-CONSOLE-L1-1-0.DLL						

Error: At least one required implicit or forwarded dependency was not found.  
Error: At least one module has an unresolved import due to a missing export function in an implicitly dependent module.  
Error: Modules with different CPU types were found.  
Error: A circular dependency was detected.  
Warning: At least one delay-load dependency module was not found.  
Warning: At least one module has an unresolved import due to a missing export function in a delay-load dependent module.

For Help, press F1

Windows Taskbar: File Explorer, Mail, Photos, OneDrive, Start, Search, Task View, System, Task Manager, Show hidden icons, 10:59 PM, 5/2021

The import functions tells us that the malware loads data from resource section (LoadResoruce, FindResource) and writes a file to disk (CreateFile, WriteFile)

The ADVAPI32.dll import tells me that the malware is trying to access advanced core Windows components such as the Service Manager and Registry. We can assumes that it tries to access protected files by adjusting to admin special permissions.

Dependency Walker - [HW-A-2.ex\_]

File Edit View Options Profile Window Help

HW-A-2.EX\_

PI	Ordinal ^	Hint	Function	Entry Point
	N/A	31 (0x001F)	AdjustTokenPrivileges	Not Bound
	N/A	406 (0x0196)	LookupPrivilegeValueA	Not Bound
	N/A	503 (0x01F7)	OpenProcessToken	Not Bound

E	Ordinal ^	Hint	Function	Entry Point
	1000 (0x3E8)	N/A	N/A	0x00033EE0
	1001 (0x3E9)	382 (0x017E)	LScGetCurrentGroupStateW	0x0002AD70
	1002 (0x3EA)	0 (0x0000)	A_SHAFinal	NTDLL.A_SHAFinal
	1003 (0x3EB)	1 (0x0001)	A_SHInit	NTDLL.A_SHInit
	1004 (0x3EC)	2 (0x0002)	A_SHAUpdate	NTDLL.A_SHAUpdate
	1005 (0x3ED)	3 (0x0003)	AbortSystemShutdownA	0x0003DA00
	1006 (0x3EE)	4 (0x0004)	AbortSystemShutdownW	0x0003DA80

Module	File Time Stamp	Link Time Stamp	File Size	Attr.	Link Checksum	Real Checksum
API-MS-WIN-CORE-APIQUERY-L1-1-0.DLL						
API-MS-WIN-CORE-APIQUERY-L1-1-1.DLL						
API-MS-WIN-CORE-APPCOMPAT-L1-1-0.DLL						
API-MS-WIN-CORE-APPCOMPAT-L1-1-1.DLL						
API-MS-WIN-CORE-APPINIT-L1-1-0.DLL						
API-MS-WIN-CORE-ATOMS-L1-1-0.DLL						
API-MS-WIN-CORE-COMM-L1-1-0.DLL						
API-MS-WIN-CORE-CONSOLE-L1-1-0.DLL						

The SHELL32.dll import tells me that the malware is trying to open a powershell which can be used to open webpages and files. It executes teh program using the function ShellExecuteA.

Dependency Walker - [HW-A-2.exe]

File Edit View Options Profile Window Help

HW-A-2.EX\_

Kernel32.DLL

ADVAPI32.DLL

SHELL32.DLL

PI	Ordinal ^	Hint	Function	Entry Point
	N/A	286 (0x011E)	ShellExecuteA	Not Bound

E	Ordinal ^	Hint	Function	Entry Point
	2 (0x0 002)	141 (0x0 08D)	SHChangeNotifyRegister	0x0060D30
	3 (0x0 003)	175 (0x0 AF)	SHDefExtractIconA	0x023AA30
	4 (0x0 004)	140 (0x0 08C)	SHChangeNotifyDeregister	0x00712E0
	5 (0x0 005)	N/A	N/A	SHUNIMPL.#35
	6 (0x0 006)	176 (0x0 B0)	SHDefExtractIconW	0x003C540
	7 (0x0 007)	N/A	N/A	SHUNIMPL.#192
	8 (0x0 008)	N/A	N/A	SHUNIMPL.#193

#### 4. What host- or network-based indicators could be used to identify this malware on infected machines?

The malware attempts to reboot the computer with the attacker having admin privilege.

Administrator: Windows PowerShell ISE

File Edit View Tools Debug Add-ons Help

PowerShell 1 PowerShell 2 PowerShell 3

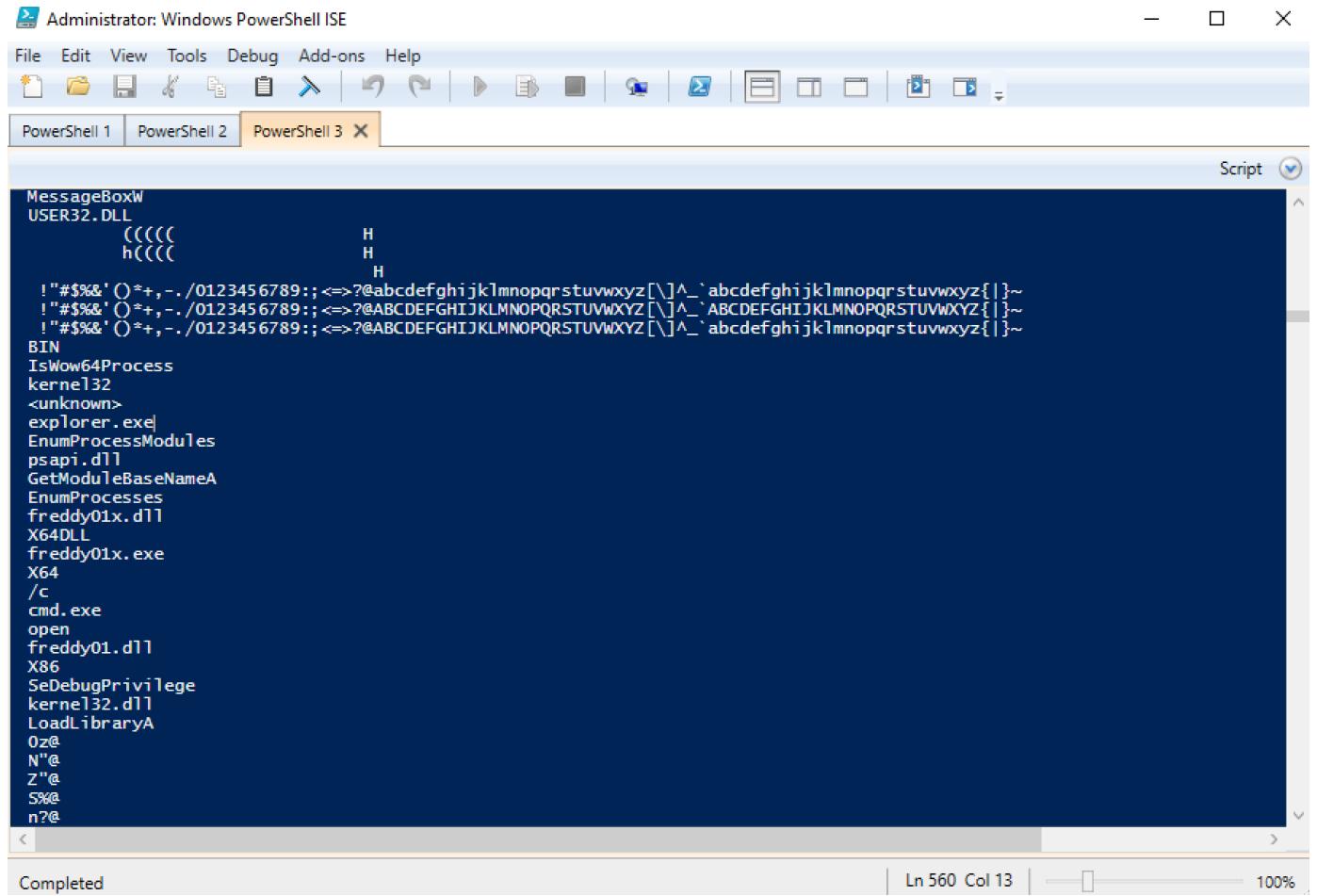
Script

```
InitializeCriticalSection
GetCPInfo
GetACP
GetOEMCP
VirtualAlloc
HeapReAlloc
SetStdHandle
RtlUnwind
MultiByteToWideChar
LCMapStringA
LCMapStringW
GetStringTypeA
GetStringTypeW
FlushFileBuffers
Press OK to reboot
Practical Malware Analysis %d
((((( H
PST
PDT
OK0d0n0y0
On1u1
2M2_2
3Z3b3
3z4
8-8>8
=2=-:=?~K=P=m=s=
>.>4><Z>`>Q>
%?+?U?{?}?
0<0Q0u0
171F1U1
2$2*2
4J4R414x4
5*5h5
556\6
```

Completed | Ln 2751 Col 15 | 100%

In order to do this the malware runs dll modules that is personally writen including freedy01x.dll and

attempts to execute it as seen with freedy01x.exe.



The screenshot shows a Windows PowerShell ISE window with three tabs: 'PowerShell 1', 'PowerShell 2', and 'PowerShell 3'. The 'PowerShell 3' tab is active and contains the following assembly code:

```
MessageBoxW
USER32.DLL
    (((((H
        h((((H
    !#$%&() *+, -./0123456789;; <=>?@abcdefghijklmnopqrstuvwxyz[\]^_`abcdefghijklmnopqrstuvwxyz{|}|~
    !#$%&() *+, -./0123456789;; <=>?@ABCDEFGHIJKLMNPQRSTUVWXYZ[\]^_`ABCDEFGHIJKLMNPQRSTUVWXYZ{|}|~
    !#$%&() *+, -./0123456789;; <=>?@ABCDEFGHIJKLMNPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz{|}|~
BIN
IsWow64Process
kernel32
<unknown>
explorer.exe
EnumProcessModules
psapi.dll
GetModuleBaseNameA
EnumProcesses
freedy01x.dll
X64DLL
freedy01x.exe
X64
/c
cmd.exe
open
freedy01.dll
X86
SeDebugPrivilege
kernel32.dll
LoadLibraryA
0z@
N"@
Z"@
S%@
n?@
```

We can check if freedy01.dll is running on the infected machine.

**5. This file has multiple resources in its resource section. What are their respective MD5 or SHA hashes? (Resources should be in BIN format)**



HashCalc



Data Format:

File

Data:

C:\Users\IEUser\Desktop\HW3\HW-A-2.ex\_

 HMAC

Key Format:

Text string

Key:

 MD5

d93adc942c53bdb9f4b4447e12e2784a

 MD4 SHA1

66b0045eccce09532454433b9cde4288877199252

 SHA256 SHA384 SHA512 RIPEMD160

3890f62d4491a1197d71477588be0f611aa18c42

 PANAMA TIGER MD2 ADLER32 CRC32

a2422731

 eDonkey/  
eMuleSlavaSoft

Calculate

Close

Help

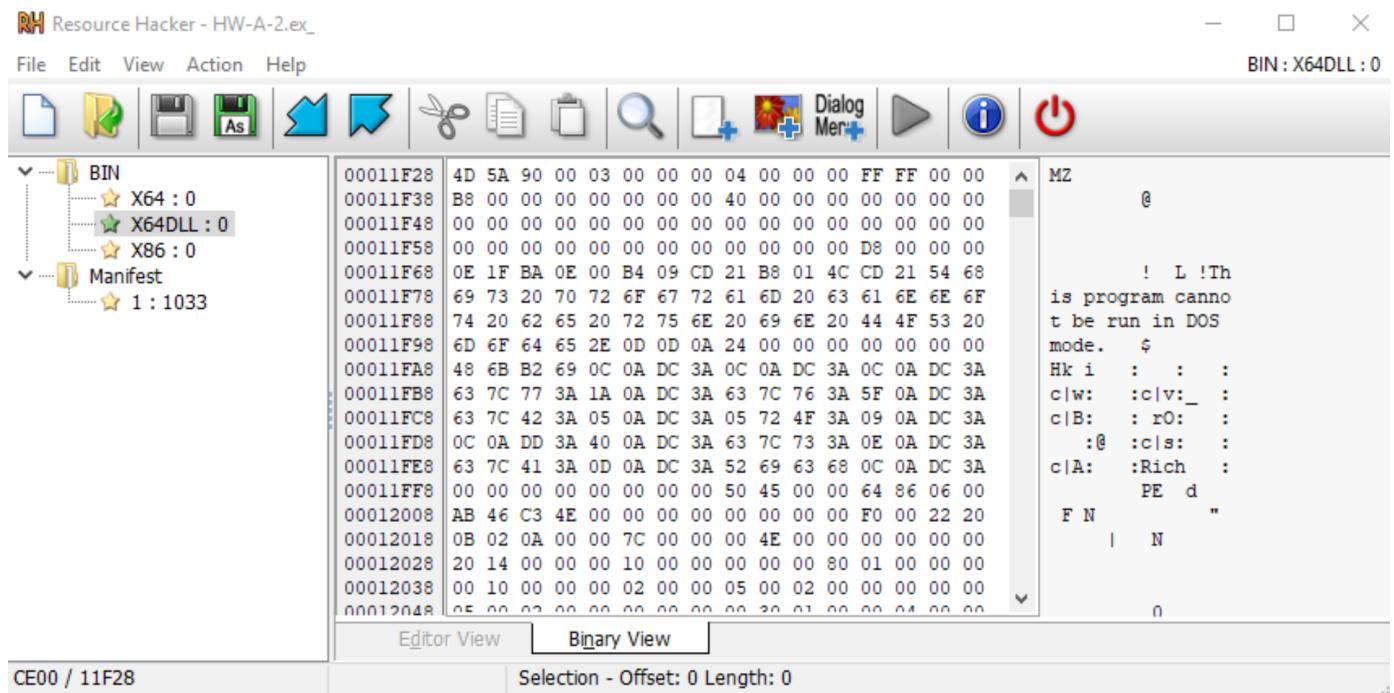
```
PS C:\Users\IEUser\Desktop\HW3> Get-FileHash -Algorithm MD5 .\HW-A-2.exe
```

Algorithm	Hash
MD5	D93ADC942C53BDB9F4B4447E12E2784A

```
PS C:\Users\IEUser\Desktop\HW3> Get-FileHash -Algorithm SHA1 .\HW-A-2.exe
```

Algorithm	Hash
SHA1	66B0045ECCE09532454433B9CDE4288877199252

## 6. What are the differences between the resources?



BIN has 3 resources sectuib x64, x64DLL, and x86.

x64 imports KERNEL32.DLL and ADVAPI32.DLL.

Dependency Walker - [BINX64.bin]

File Edit View Options Profile Window Help

BINX64.BIN

PI	Ordinal ^	Hint	Function
C	N/A	82 (0x0052)	CloseHandle
C	N/A	169 (0x00A9)	CreateRemoteThread
C	N/A	203 (0x00CB)	DecodePointer
C	N/A	210 (0x00D2)	DeleteCriticalSection
C	N/A	238 (0x00EE)	EncodePointer
C	N/A	242 (0x00F2)	EnterCriticalSection
C	N/A	287 (0x011F)	ExitProcess

E	Ordinal ^	Hint	Function
C	1 (0x0001)	0 (0x0000)	AcquireSRWLockExclusive
C	2 (0x0002)	1 (0x0001)	AcquireSRWLockShared
C	3 (0x0003)	2 (0x0002)	ActivateActCtx
C	4 (0x0004)	3 (0x0003)	ActivateActCtxWorker
C	5 (0x0005)	4 (0x0004)	AddAtomA
C	6 (0x0006)	5 (0x0005)	AddAtomW
C	7 (0x0007)	6 (0x0006)	AddConsoleAliasA

Module	File Time Stamp	Link Time Stamp	File Size
API-MS-WIN-CORE-APIQUERY-L1-1-0.DLL	Error opening file. The system cannot find the file specified.		
API-MS-WIN-CORE-APIQUERY-L1-1-1.DLL	Error opening file. The system cannot find the file specified.		
API-MS-WIN-CORE-APPCOMPAT-L1-1-0.DLL	Error opening file. The system cannot find the file specified.		
API-MS-WIN-CORE-APPCOMPAT-L1-1-1.DLL	Error opening file. The system cannot find the file specified.		

For Help, press F1

Dependency Walker - [BINX64.bin]

File Edit View Options Profile Window Help

BINX64.BIN

PI	Ordinal ^	Hint	Function
C	N/A	31 (0x001F)	AdjustTokenPrivileges
C	N/A	406 (0x0196)	LookupPrivilegeValueA
C	N/A	503 (0x01F7)	OpenProcessToken

E	Ordinal ^	Hint	Function
O#	1000 (0x03E8)	N/A	N/A
C	1001 (0x03E9)	382 (0x017E)	I_ScGetCurrentGroupStateW
C	1002 (0x03EA)	0 (0x0000)	A_SHAFinal
C	1003 (0x03EB)	1 (0x0001)	A_SHAInit
C	1004 (0x03EC)	2 (0x0002)	A_SHAUpdate
C	1005 (0x03ED)	3 (0x0003)	AbortSystemShutdownA
C	1006 (0x03FF)	4 (0x0004)	AbortSystemShutdownW

Module	File Time Stamp	Link Time Stamp	File Size
API-MS-WIN-CORE-APIQUERY-L1-1-0.DLL	Error opening file. The system cannot find the file specified.		
API-MS-WIN-CORE-APIQUERY-L1-1-1.DLL	Error opening file. The system cannot find the file specified.		
API-MS-WIN-CORE-APPCOMPAT-L1-1-0.DLL	Error opening file. The system cannot find the file specified.		
API-MS-WIN-CORE-APPCOMPAT-L1-1-1.DLL	Error opening file. The system cannot find the file specified.		

For Help, press F1

The most important things x64 does is that it adjusts the privilege of the malware to have admin access.

x64DLL imports two DLLs, KERNEL32.DLL and USER32.DLL. The USER32.DLL has functions such as GetKeyboardLayout, GetMouseMovePointsEx, indicating that it is probably a passive keylogger. Additionally, the MessageBoxA Displays a modal dialog box that contains a system icon, a set of buttons, and a brief application-specific message, such as status or error information. It stores the data from keylogging in a file indicated by the functions in KERNEL32.DLL containing functions such as CreateFileW, WriteFile.

The screenshot shows the Dependency Walker interface. The left pane displays the dependency tree for 'X64DLL.BIN', which includes 'X64DLL.BIN' itself and imports from 'KERNEL32.DLL' and 'USER32.DLL'. The right pane contains two tables: one for imports from 'USER32.DLL' and another for imports from 'KERNEL32.DLL'. Below these tables is a table for loaded modules, showing 'API-MS-WIN-CORE-APIQUERY-L1-1-0.DLL' with an error message: 'Error opening file. The system cannot find the file specified'. At the bottom, a status bar says 'For Help, press F1'.

PI	Ordinal	Hint	Function	Entry P
C	N/A	530 (0x0212)	MessageBoxA	Not Bo

E	Ordinal ^	Hint	Function	Entry ^
0#	1502 (0x05DE)	N/A	N/A	0x00
C	1503 (0x05DF)	0 (0x0000)	ActivateKeyboardLayout	0x00
C	1504 (0x05E0)	1 (0x0001)	AddClipboardFormatListener	0x00
C	1505 (0x05E1)	2 (0x0002)	AdjustWindowRect	0x00
C	1506 (0x05E2)	3 (0x0003)	AdjustWindowRectEx	0x00
C	1507 (0x05E3)	4 (0x0004)	AdjustWindowRectExForDpi	0x00
C	1508 (0x05E4)	5 (0x0005)	AlignRects	0x00
C	1509 (0x05E5)	6 (0x0006)	AllowForegroundActivation	0x00
C	1510 (0x05E6)	7 (0x0007)	AllowSetForegroundWindow	0x00
C	1511 (0x05E7)	8 (0x0008)	AnimateWindow	0x00
C	1512 (0x05E8)	9 (0x0009)	AnyPopup	0x00
C	1513 (0x05E9)	10 (0x000A)	AppendMenuA	0x00

Module	File Time Stamp	Link Time Stamp	File Size
API-MS-WIN-CORE-APIQUERY-L1-1-0.DLL ADLMS-WINCORE-APIQUERY-L1-1-0.DLL	Error opening file. The system cannot find the file specified	Error opening file. The system cannot find the file specified	

x86 imports two DLLs, KERNEL32.DLL and USER32.DLL. However, unlike x64DLL, x86 injects malicious information into the system through functions such as InjectSyntheticPointInput, InjectTouchInput.

Dependency Walker - [BINX86]

File Edit View Options Profile Window Help

BINX86  
+ KERNEL32.DLL  
+ USER32.DLL

PI	Ordinal	Hint	Function ^	Entry Po
	N/A	446 (0x01BE)	MessageBoxA	Not Bou

E	Ordinal ^	Hint	Function	Entry ^
C	2038 (0x07F6)	530 (0x0212)	InjectSyntheticPointerInput	0x00
C	2039 (0x07F7)	531 (0x0213)	InjectTouchInput	0x00
C	2040 (0x07F8)	532 (0x0214)	InsertMenuItemA	0x00
C	2041 (0x07F9)	533 (0x0215)	InsertMenuItemW	0x00
C	2042 (0x07FA)	534 (0x0216)	InsertMenuW	0x00
C	2043 (0x07FB)	535 (0x0217)	IntersectRect	0x00
C	2044 (0x07FC)	536 (0x0218)	InternalGetWindowIcon	0x00
C	2045 (0x07FD)	537 (0x0219)	InternalGetWindowText	0x00
C	2046 (0x07FE)	538 (0x021A)	InvalidateRect	0x00
C	2047 (0x07FF)	539 (0x021B)	InvalidateRgn	0x00
C	2048 (0x0800)	540 (0x021C)	InvertRect	0x00
C	2049 (0x0801)	541 (0x021D)	IsCharAlphaA	0x00
C	2050 (0x0802)	542 (0x021E)	IsCharAlphaNumerica	0x00
C	2051 (0x0803)	543 (0x021F)	IsCharAlphaNumericaA	0x00

Module	File Time Stamp	Link Time Stamp	File Size
API-MS-WIN-CORE-APIQUERY-L1-1-0.DLL	Error opening file. The system cannot find the file specified.		
API-MS-WIN-CORE-APIQUERY-L1-1-1.DLL	Error opening file. The system cannot find the file specified.		
API-MS-WIN-CORE-APPCOMPAT-L1-1-0.DLL	Error opening file. The system cannot find the file specified.		

For Help, press F1

Manifest has 1 resource. It requests privilege "asInvoker" without UI access.

Resource Hacker - HW-A-2.exe

File Edit View Action Help

BIN  
+ X64 : 0  
+ X64DLL : 0  
+ X86 : 0

Manifest  
+ 1 : 1033

```

1 <assembly xmlns="urn:schemas-microsoft-com:asm.v1" manifestVersion="1.0">
2 <trustInfo xmlns="urn:schemas-microsoft-com:asm.v3">
3   <security>
4     <requestedPrivileges>
5       <requestedExecutionLevel level="asInvoker" uiAccess="false"></requestedExecutionLevel>
6     </requestedPrivileges>
7   </security>
8 </trustInfo>
9 </assembly>

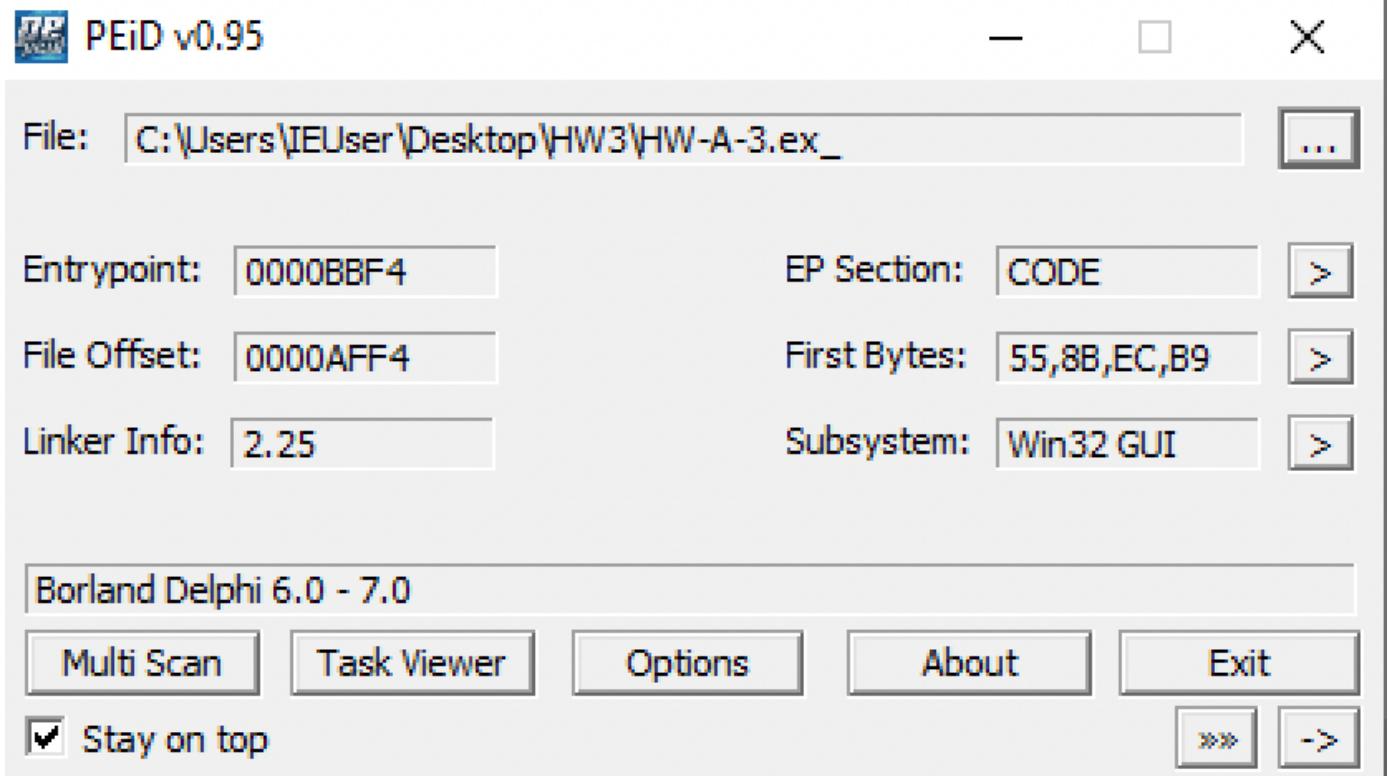
```

## Analyzing HW3-A-3.exe

Task 3: Answer the following questions after analyzing HW-A-3.exe

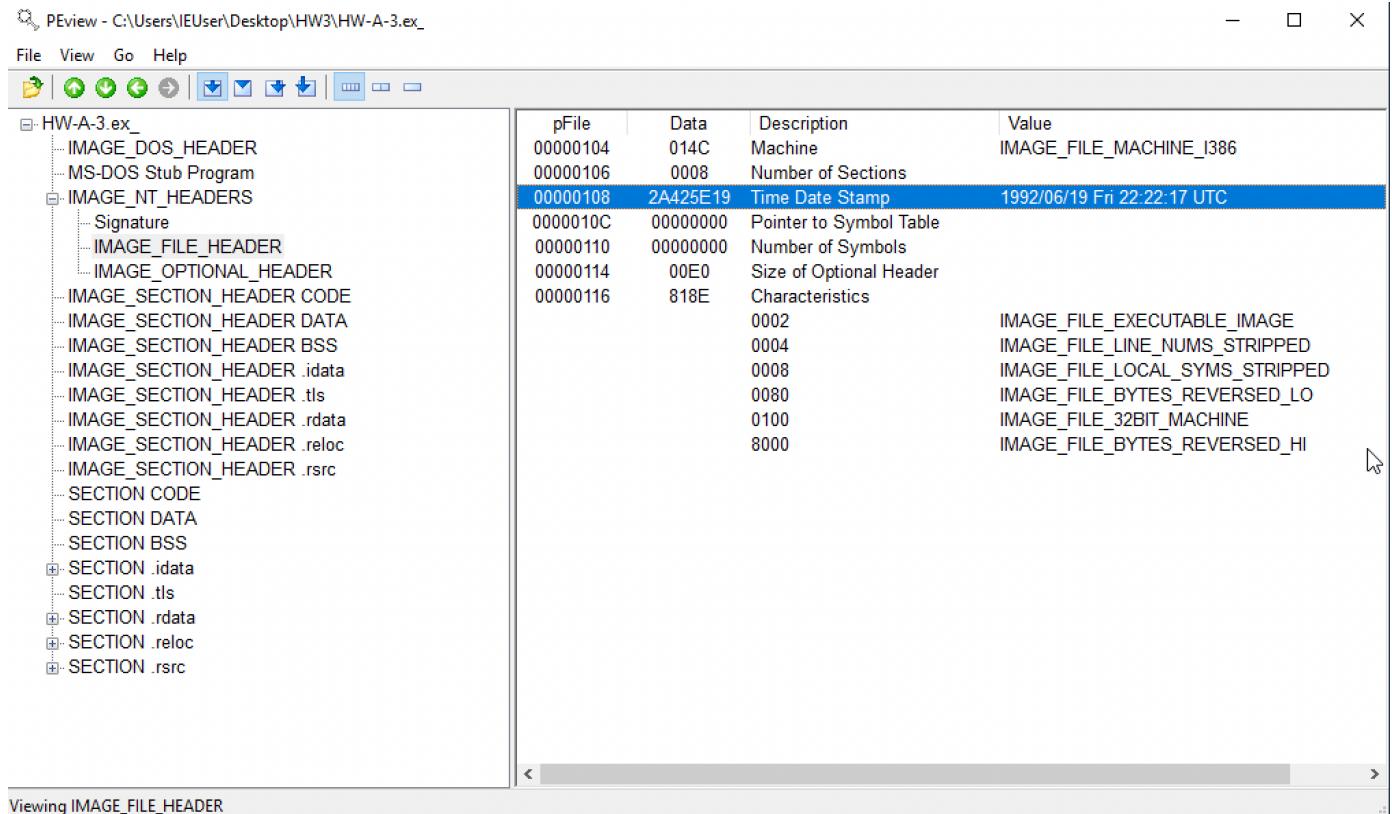
- Are there any indications that this file is packed or obfuscated?

No, it is not packed. The program is written in Borland Delphi version 6.0-7.0.



## 2. When was this program compiled?

It was compiled on 1992/06/19 Fri 22:22:17 UTC



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

Program's overall functionality: steal user's credentials and password and send it back to the attack on remote device.

## Detailed Analysis:

KERNEL32.DLL is the most basic import to manipulate files.

The screenshot shows the Dependency Walker interface for the file 'HW-A-3.ex\_'. The left pane displays a tree view of imports from 'HW-A-3.EX\_'. Under 'HW-A-3.EX\_>', there are imports from 'KERNEL32.DLL', 'USER32.DLL', 'OLEAUT32.DLL', 'ADVAPI32.DLL', and 'KERNEL32.DLL' again. The right pane is a detailed table of imports from 'KERNEL32.DLL'.

PI	Ordinal	Hint	Function ^	Entry Point
C	N/A	0 (0x0000)	ExitProcess	Not Bound
C	N/A	0 (0x0000)	FreeLibrary	Not Bound
C	N/A	0 (0x0000)	GetCommandLineA	Not Bound
C	N/A	0 (0x0000)	GetCurrentThreadId	Not Bound
C	N/A	0 (0x0000)	GetModuleFileNameA	Not Bound
C	N/A	0 (0x0000)	GetModuleHandleA	Not Bound
C	N/A	0 (0x0000)	GetProcessHeap	Not Bound
C	N/A	0 (0x0000)	HeapAlloc	Not Bound
C	N/A	0 (0x0000)	HeapFree	Not Bound
C	N/A	0 (0x0000)	HeapReAlloc	Not Bound
C	N/A	0 (0x0000)	LocalAlloc	Not Bound
C	N/A	0 (0x0000)	MultiByteToWideChar	Not Bound
C	N/A	0 (0x0000)	RaiseException	Not Bound
C	N/A	0 (0x0000)	RtlUnwind	Not Bound
C	N/A	0 (0x0000)	TlsGetValue	Not Bound
C	N/A	0 (0x0000)	TlsSetValue	Not Bound
C	N/A	0 (0x0000)	UnhandledExceptionFilter	Not Bound
C	N/A	0 (0x0000)	WideCharToMultiByte	Not Bound

E	Ordinal ^	Hint	Function	Entry Point
C	1 (0x0001)	0 (0x0000)	AcquireSRWLockExclusive	NTDLL.RtlAcquireSR

CharNextA from USER32.DLL gets the next character input from USER

The screenshot shows the Dependency Walker interface for the file 'HW-A-3.ex\_'. The left pane displays a tree view of imports from 'HW-A-3.EX\_'. Under 'HW-A-3.EX\_>', there are imports from 'KERNEL32.DLL', 'USER32.DLL', and 'OLEAUT32.DLL'. The right pane is a detailed table of imports from 'USER32.DLL'.

PI	Ordinal	Hint	Function ^	Entry Point
C	N/A	0 (0x0000)	CharNextA	Not Bound

OLEAUT32.DLL allows applications to handle files and information created by other applications. It indicates that the malware is talking to some software.

The screenshot shows the Dependency Walker interface for the file 'HW-A-3.ex\_'. The left pane displays a tree view of imports from 'HW-A-3.EX\_'. Under 'HW-A-3.EX\_>', there are imports from 'KERNEL32.DLL', 'USER32.DLL', and 'OLEAUT32.DLL'. The right pane is a detailed table of imports from 'OLEAUT32.DLL'.

PI	Ordinal	Hint	Function ^	Entry Point
C	N/A	0 (0x0000)	SysAllocStringLen	Not Bound
C	N/A	0 (0x0000)	SysFreeString	Not Bound
C	N/A	0 (0x0000)	SysReAllocStringLen	Not Bound

ADVAPI32.DLL: We can assume that it tries to access protected files by adjusting to admin special permissions.

Dependency Walker - [HW-A-3.ex\_]

File Edit View Options Profile Window Help

PI	Ordinal	Hint	Function ^	Entry Point
		0 (0x0000)	GetUserNameA	Not Bound
		0 (0x0000)	IsValidSid	Not Bound
		0 (0x0000)	LookupAccountNameA	Not Bound
		0 (0x0000)	OpenProcessToken	Not Bound
		0 (0x0000)	RegCloseKey	Not Bound
		0 (0x0000)	RegCreateKeyA	Not Bound
		0 (0x0000)	RegCreateKeyExA	Not Bound
		0 (0x0000)	RegDeleteKeyA	Not Bound
		0 (0x0000)	RegEnumValueA	Not Bound
		0 (0x0000)	RegOpenKeyExA	Not Bound
		0 (0x0000)	RegQueryValueExA	Not Bound
		0 (0x0000)	RegSetValueExA	Not Bound

KERNEL32.DLL Creates a directory(CreateDirectoryA) and is reading, writing, executing(CreateProcessA, FindFirstFileA, FindResourceA) resources

Dependency Walker - [HW-A-3.ex\_]

File Edit View Options Profile Window Help

PI	Ordinal	Hint	Function ^	Entry Point
		0 (0x0000)	CloseHandle	Not Bound
		0 (0x0000)	CopyFileA	Not Bound
		0 (0x0000)	CreateDirectoryA	Not Bound
		0 (0x0000)	CreateFileA	Not Bound
		0 (0x0000)	CreateMutexA	Not Bound
		0 (0x0000)	CreateProcessA	Not Bound
		0 (0x0000)	CreateRemoteThread	Not Bound
		0 (0x0000)	DeleteFileA	Not Bound
		0 (0x0000)	ExitProcess	Not Bound
		0 (0x0000)	FindClose	Not Bound
		0 (0x0000)	FindFirstFileA	Not Bound
		0 (0x0000)	FindResourceA	Not Bound
		0 (0x0000)	FreeLibrary	Not Bound
		0 (0x0000)	FreeResource	Not Bound
		0 (0x0000)	GetCurrentProcess	Not Bound
		0 (0x0000)	GetExitCodeThread	Not Bound
		0 (0x0000)	GetFileAttributesA	Not Bound
		0 (0x0000)	GetFileSize	Not Bound
		0 (0x0000)	GetLastError	Not Bound
		0 (0x0000)	GetModuleHandleA	Not Bound
		0 (0x0000)	GetPrivateProfileIntA	Not Bound
		0 (0x0000)	GetPrivateProfileStringA	Not Bound

Most likely evesdropping(PeekMessageA) on the keyboard (GetKeyboardStateA) input from user.

Dependency Walker - [HW-A-3.ex\_]

File Edit View Options Profile Window Help

PI	Ordinal	Hint	Function ^	Entry Point
		0 (0x0000)	CharLowerA	Not Bound
		0 (0x0000)	CharUpperA	Not Bound
		0 (0x0000)	DispatchMessageA	Not Bound
		0 (0x0000)	FindWindowA	Not Bound
		0 (0x0000)	GetKeyboardState	Not Bound
		0 (0x0000)	GetWindowThreadProcessId	Not Bound
		0 (0x0000)	PeekMessageA	Not Bound
		0 (0x0000)	SetWindowsHookExA	Not Bound
		0 (0x0000)	ToAscii	Not Bound
		0 (0x0000)	TranslateMessage	Not Bound
		0 (0x0000)	wvsprintfA	Not Bound

CoCreateInstance is used to access to COM functionality. Possibly to launch internet explorer and access an address

HW-A-3.EX_
+ KERNEL32.DLL
+ USER32.DLL
+ OLEAUT32.DLL
+ ADVAPI32.DLL
+ KERNEL32.DLL
+ USER32.DLL
+ OLE32.DLL
+ OLE32.DLL

PI	Ordinal	Hint	Function ^	Entry Point
[C]	N/A	0 (0x0000)	CoCreateInstance	Not Bound
[C]	N/A	0 (0x0000)	OleInitialize	Not Bound

Frees up a memory block. Malware most likely going to execute some program at this place

PI	Ordinal	Hint	Function ^	Entry Point
[C]	N/A	0 (0x0000)	CoTaskMemFree	Not Bound

Access the legacy Windows data store, Pstore. If successful a call to the PStoreCreateInstance function to get the pointer to protected storage will be made to allow calls to functions that enumerate and retrieve any stored credentials.

PI	Ordinal	Hint	Function ^	Entry Point
[C]	N/A	0 (0x0000)	PStoreCreateInstance	Not Bound

Converts CLSID into a string of printable characters. Allows the malware to run an application on Windows.

PI	Ordinal	Hint	Function ^	Entry Point
[C]	N/A	0 (0x0000)	StringFromCLSID	Not Bound

RAS is a remote access service. RasEnumEntriesA indicates a that the malware is using communication module to connect to the attacker's machine

	PI	Ordinal	Hint	Function ^	Entry Point
HW-A-3.EX_					
+ KERNEL32.DLL	[C]	N/A	0 (0x0000)	RasEnumEntriesA	Not Bound
+ USER32.DLL	[C]	N/A	0 (0x0000)	RasGetEntryDialParamsA	Not Bound
+ OLEAUT32.DLL					
+ ADVAPI32.DLL					
+ KERNEL32.DLL					
+ USER32.DLL					
+ OLE32.DLL					
+ PSTOREC.DLL					
+ OLE32.DLL					
+ RASAPI32.DLL					
+ SHELL32.DLL					
+ ADVAPI32.DLL					

CSIDLs is passed the following functions as arguments: SHGetSpecialFolderPathA. The purpose is for malware to access important folders.

	PI	Ordinal	Hint	Function ^
HW-A-3.EX_				
+ KERNEL32.DLL	[C]	N/A	0 (0x0000)	SHGetSpecialFolderPathA
+ USER32.DLL				
+ OLEAUT32.DLL				
+ ADVAPI32.DLL				
+ KERNEL32.DLL				
+ USER32.DLL				
+ OLE32.DLL				
+ OLE32.DLL				
+ PSTOREC.DLL				
+ OLE32.DLL				
+ RASAPI32.DLL				
+ SHELL32.DLL				
+ ADVAPI32.DLL				
+ CRYPT32.DLL				

The malware retrieves private data(LSARetrievePrivateData) which could be things like default password.

	PI	Ordinal	Hint	Function ^
HW-A-3.EX_				
+ KERNEL32.DLL	[C]	N/A	0 (0x0000)	ConvertSidToStringSidA
+ USER32.DLL	[C]	N/A	0 (0x0000)	LsaClose
+ OLEAUT32.DLL	[C]	N/A	0 (0x0000)	LsaFreeMemory
+ ADVAPI32.DLL	[C]	N/A	0 (0x0000)	LsaOpenPolicy
+ KERNEL32.DLL	[C]	N/A	0 (0x0000)	LsaRetrievePrivateData
+ USER32.DLL				
+ OLE32.DLL				
+ OLE32.DLL				
+ PSTOREC.DLL				
+ OLE32.DLL				
+ RASAPI32.DLL				
+ SHELL32.DLL				
+ ADVAPI32.DLL				
+ CRYPT32.DLL				
+ ADVAPI32.DLL				

## Malware steals credentials and cookies

	PI	Ordinal	Hint	Function ^	Entry Point
HW-A-3.EX_				CryptUnprotectData	Not Bound

CredEnumerateA API to identify and steal saved credentials.

	HW-A-3.EX_	PI	Ordinal	Hint	Function ^	Entry Point
+	KERNEL32.DLL					
+	USER32.DLL					
+	OLEAUT32.DLL					
+	ADVAPI32.DLL					
...	KERNEL32.DLL					
...	USER32.DLL					
+	OLE32.DLL					
+	OLE32.DLL					
+	PSTOREC.DLL					
+	OLE32.DLL					
+	RASAPI32.DLL					
+	SHELL32.DLL					
+	ADVAPI32.DLL					
+	CRYPT32.DLL					
+	ADVAPI32.DLL					
+	ADVAPI32.DLL					

Clean up afterwards.

PI	Ordinal	Hint	Function ^	Entry Point
[C]	N/A	0 (0x0000)	CryptAcquireContextA	Not Bound
[C]	N/A	0 (0x0000)	CryptCreateHash	Not Bound
[C]	N/A	0 (0x0000)	CryptDestroyHash	Not Bound
[C]	N/A	0 (0x0000)	CryptGetHashParam	Not Bound
[C]	N/A	0 (0x0000)	CryptHashData	Not Bound
[C]	N/A	0 (0x0000)	CryptReleaseContext	Not Bound

#### 4. What can you observe using Basic Static Analysis techniques?

## 1. Matches 66/69 existing antivirus definition

The screenshot shows a VirusShare analysis page for a file named 'server32.exe'. A red circle in the top left corner displays the number '66 / 69' in white, indicating the number of vendors that flagged the file as malicious. Below this, a message states '66 security vendors and 1 sandbox flagged this file as malicious'. The file's MD5 hash is listed as '9edc610285e3e2ca964d359d01b2151df3f73333c69d27651630bd6a5c89dafc'. To the right, the file size is '290.50 KB', the date is '2021-10-23 10:13:50 UTC', and it was uploaded '3 days ago'. A small icon of a person with a briefcase is labeled 'EXE'.

## 2. Hashing, MD5 Executeable

**f2b7e9d4d1daaa3c17dc84c8f6bceb7b**

## 3. A few important information including date created, number of functions, symbols, and language used to write the program:

Import Results Summary	
i	Project File Name: HW-A-3.ex_
	Last Modified: Wed Oct 27 01:29:23 PDT 2021
	Readonly: false
	Program Name: HW-A-3.ex_
	Language ID: x86:LE:32:default (2.9)
	Compiler ID: borlanddelphi
	Processor: x86
	Endian: Little
	Address Size: 32
	Minimum Address: 00400000
	Maximum Address: 0044f5ff
	# of Bytes: 302073
	# of Memory Blocks: 9
	# of Instructions: 0
	# of Defined Data: 1798
	# of Functions: 35
	# of Symbols: 131
	# of Data Types: 40
	# of Data Type Categories: 3
	Compiler: borland:pascal
	Created With Ghidra Version: 9.1.2
	Date Created: Wed Oct 27 01:28:41 PDT 2021
	Executable Format: Portable Executable (PE)
	Executable Location: /C:/Users/IEUser/Desktop/HW3/HW-A-3.ex_
	Executable MD5: f2b7e9d4d1daaa3c17dc84c8f6bceb7b
	Executable SHA256: 9edc610285e3e2ca964d359d01b2151df3f73333c69d27651630bd6a5c89dafc
	FSRL: file:///C:/Users/IEUser/Desktop/HW3/HW-A-3.ex_?MD5=f2b7e9d4d1daaa3c17dc84c8f6bceb7b
	Relocatable: true
	SectionAlignment: 4096

## 4. Its not packed.

5. PS C:\Users\IEUser\Desktop\HW3> strings HW-A-3.ex\_

1. Interesting String

1. Calls for internet explorer: software\microsoft\internet explorer

```
Address:  
User:  
Password:  
Uh<  
ZYYd  
[Y]  
Software\Microsoft\Internet Explorer\IntelliForms\Storage2  
ZYYd  
HtSU  
=t*f  
SVW  
YZ_^[  
Uhm  
ZYYd  
jjj  
ZYYd  
Version  
Software\Microsoft\Internet Explorer  
jjj  
Uhx  
ZYYd  
UserName  
SOFTWARE\Vitalwerks\DUC  
Password  
No-ip DUC|  
PCREDENTIAL  
UnitPasswords  
S112
```

2. Injects a library : OUnitInjectLibrary

```
Base64
KWindows
System
SysInit
UTypes
,uIE7_decode
UnitDiversos
TlHelp32
gUnitServerUtils
wcrypt2
KuURLHistory
sActiveX
3Messages
CryptApi
uRASReader
IEpasswords
Pstoreclib
SPSTORECLib_TLB
'unitStartup
UnitComandos
deleteUnit
UnitPasswords
EditSvr
UnitInstalacao
UUnitSettings
RUnitVariaveis
6UnitSandBox
OUnitInjectLibrary
```

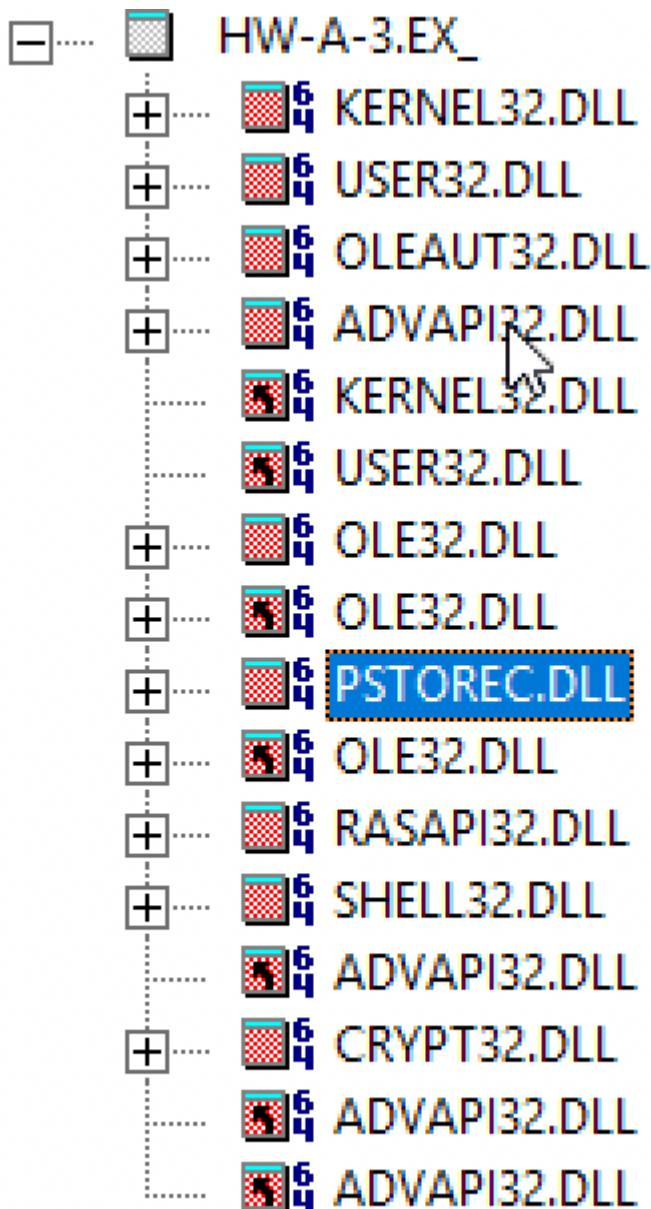
3. Program asks for User: Password: and Address:

```
Address:  
User:  
Password:  
Uh<  
ZYYd  
[Y]  
Software\Microsoft\Internet Explorer\IntelliForms\Storage2  
ZYYd  
HtSU  
=t*f  
SVW  
YZ_^[  
Uhm  
ZYYd  
jjj  
ZYYd  
Version  
Software\Microsoft\Internet Explorer  
jjj  
Uhx  
ZYYd  
UserName  
SOFTWARE\Vitalwerks\DUC  
Password  
No-ip DUC|  
PCREDENTIAL  
UnitPasswords  
G14
```

4. Looks for shell:

```
_`L  
ShellExecuteA  
shell32.dll  
ZYYd  
^[[YY]  
Software\Microsoft\Windows\CurrentVersion\Explorer\Shell Folders  
Uh/`@  
DVA
```

6. Import Functions were analysed in detail in question 4.



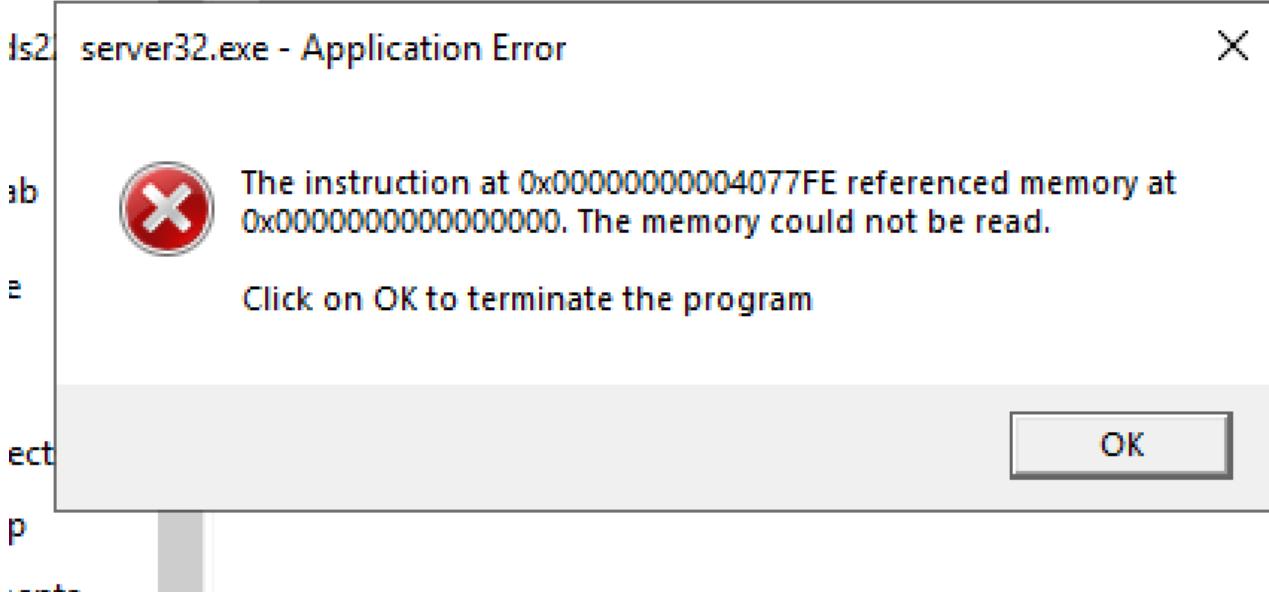
1.

summary: steal user's

credentials and password and send it back to the attack on remote device connected via internet explorer.

##### 5. What do you observe through dynamic analysis?

Running the program shows "Application Error"



1. Examine ApateDNS to see if DNS requests were performed.

Using ApateDNS, we see that DNS requests are preformed to "www.isitreal.edu" and "www.atwushere.net" every few seconds.

The screenshot shows the ApateDNS application interface. At the top, there are two tabs: "Capture Window" and "DNS Hex View", with "Capture Window" being the active tab. Below the tabs is a table with four columns: "Time", "Domain Requested", "DNS Returned", and an empty column. The table contains four rows of data corresponding to the captured DNS requests. At the bottom of the application window, there is a log of server operations with "[+]" prefixing each line. At the very bottom, there are several configuration options and control buttons.

Time	Domain Requested	DNS Returned	
07:52:28	www.itisreal.edu	FOUND	
07:52:36	www.atwushere.net	FOUND	
07:52:42	www.itisreal.edu	FOUND	
07:52:49	www.atwushere.net	FOUND	

```
[+] Attempting to find DNS by DHCP or Static DNS.  
[+] Using IP address 10.0.0.2 for DNS Reply.  
[+] DNS set to 127.0.0.1 on Intel(R) PRO/1000 MT Desktop Adapter.  
[+] Sending valid DNS response of first request.  
[+] Server started at 07:52:28 successfully.
```

DNS Reply IP (Default: Current Gateway/DNS):

# of NXDOMAIN's:

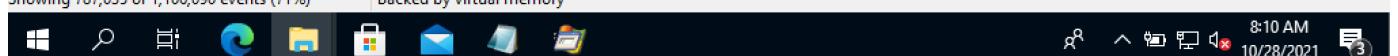
Selected Interface:

2. Review the procmon results for file system modifications.

Analysing with procmon, we see that the malware creates a lot of files in C:\Users\IEUser\AppData\Local\Temp. It attempts to read the file XxX.xXx specifically which is used to

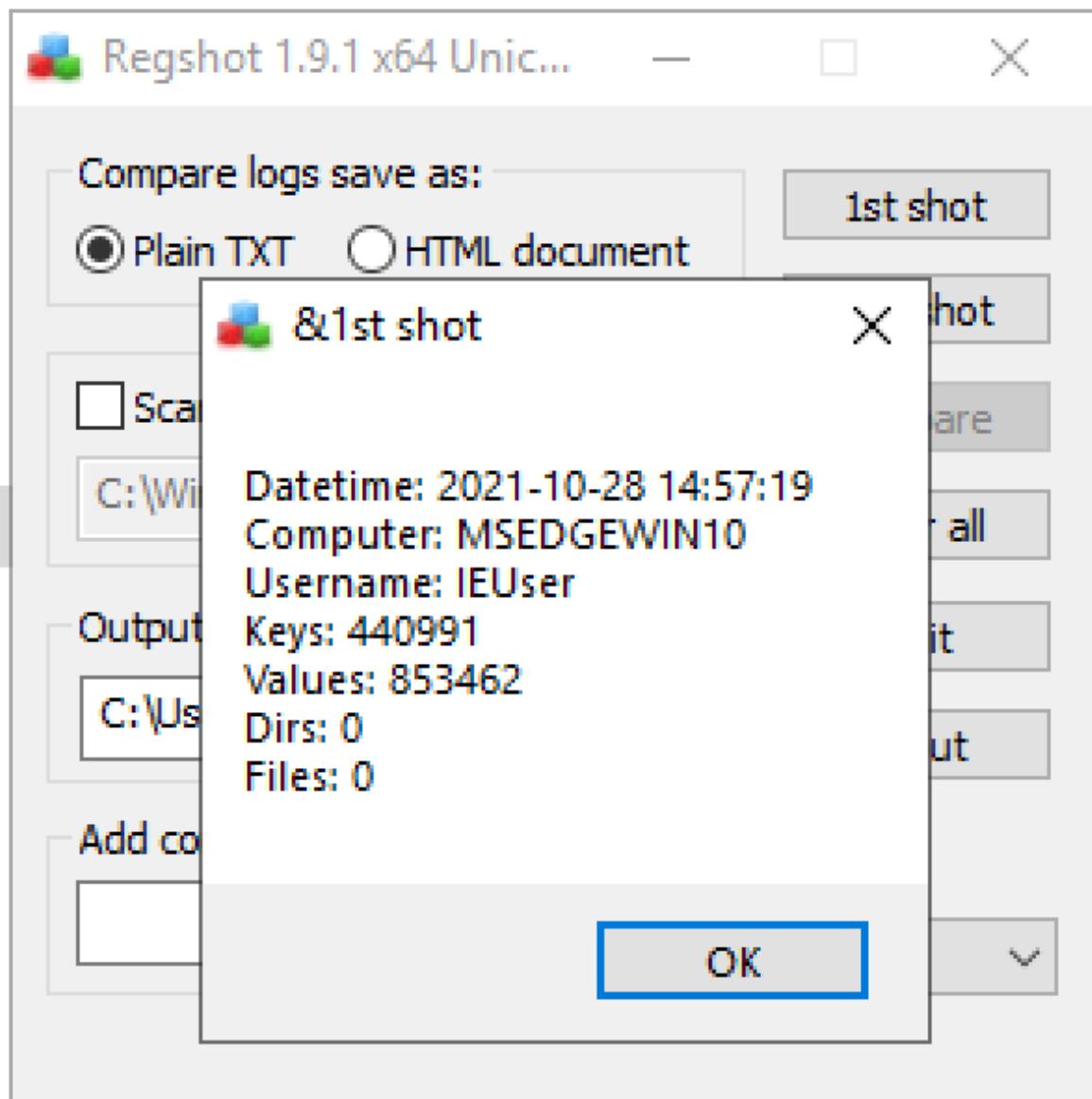
block the mouse.

 Process Monitor - Sysinternals: [www.sysinternals.com](http://www.sysinternals.com)



3. Compare the two snapshots taken with Regshot to identify changes.

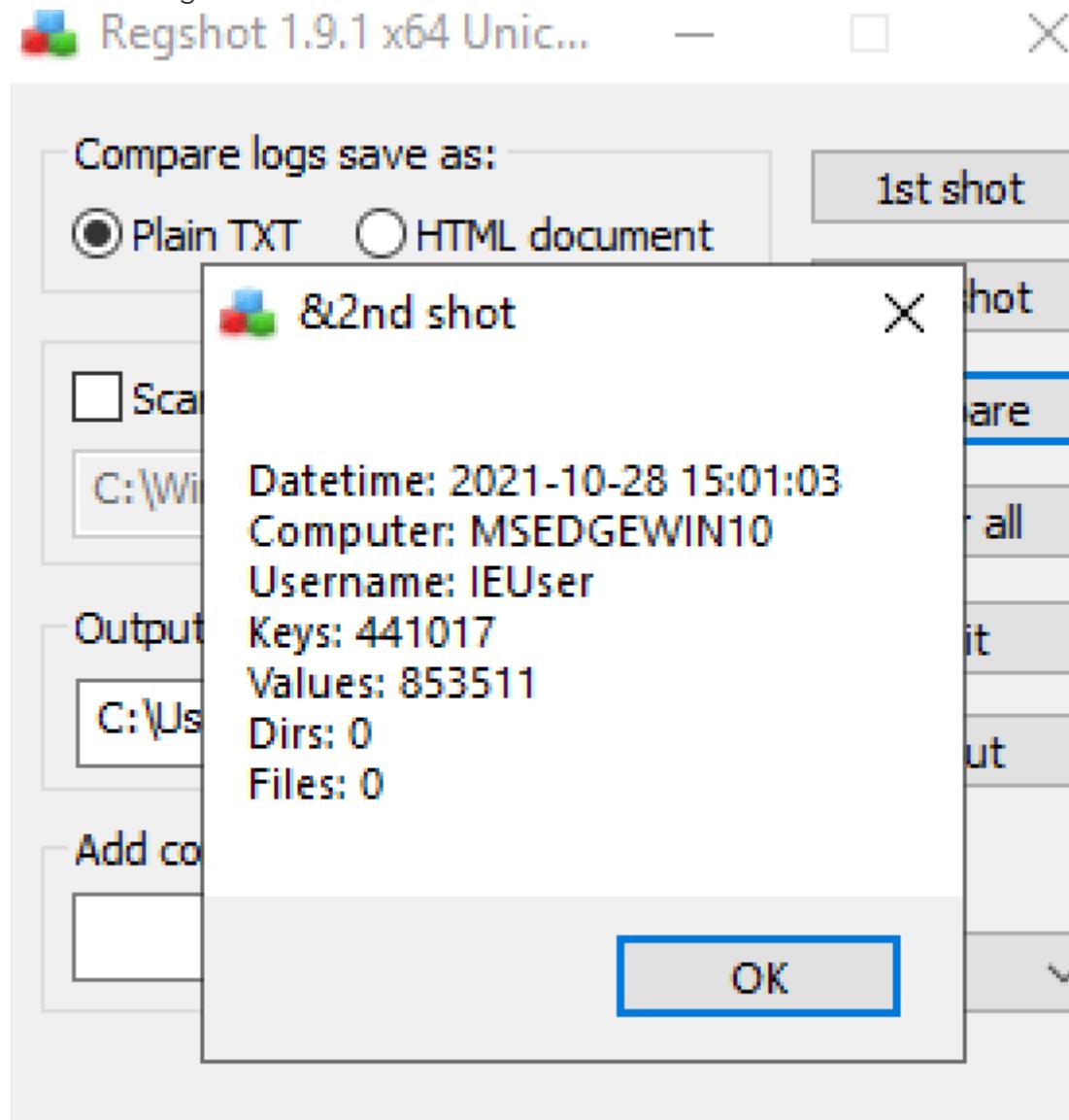
10/3/2020 6:30 PM      Shortcut



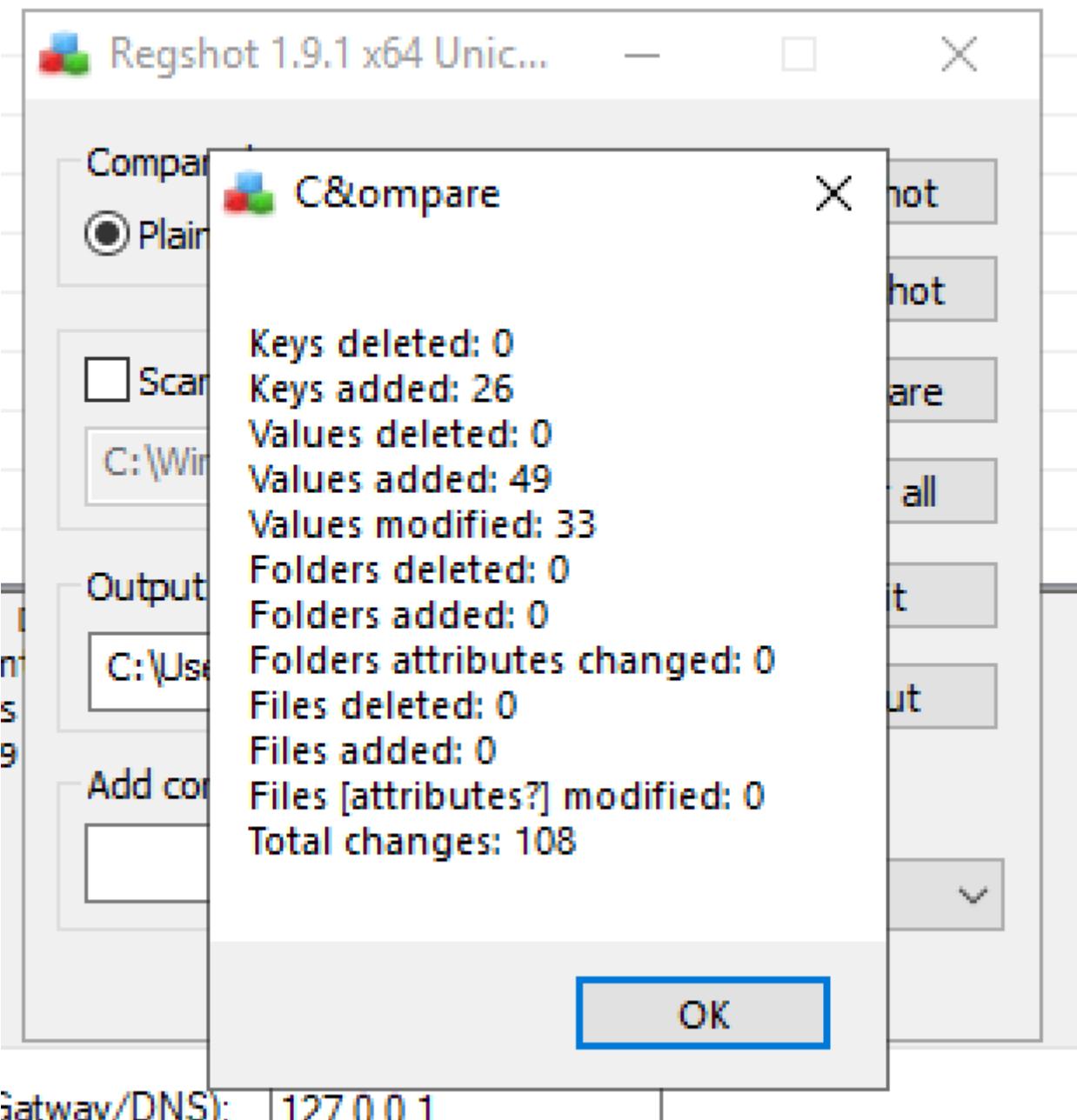
10/3/2020 6:28 PM      Shortcut

10/3/2020 5:32 PM      Shortcut

After installing FakeNet



Comparing the two there were several keys added, values deleted, and values added.



Gateway/DNS): 127.0.0.1

4. Use Process Explorer to examine the process to determine whether it creates mutexes or listens for incoming connections

Process	CPU	Private Bytes	Working Set	PID	Description	Company Name
RuntimeBroker.exe		2,244 K	12,580 K	2412	Runtime Broker	Microsoft Corporation
SearchIndexer.exe	< 0.01	18,144 K	22,848 K	908	Microsoft Windows Search I...	Microsoft Corporation
smartscreen.exe		8,020 K	22,880 K	3272	Windows Defender SmartScr...	Microsoft Corporation
SecurityHealthService.exe		3,996 K	15,500 K	3376	Windows Security Health Se...	Microsoft Corporation
svchost.exe		3,312 K	7,476 K	5860	Host Process for Windows S...	Microsoft Corporation
WinStore.App.exe	Susp...	16,100 K	432 K	6128	Store	Microsoft Corporation
ApplicationFrameHost.exe		10,400 K	28,012 K	6136	Application Frame Host	Microsoft Corporation
RuntimeBroker.exe		1,596 K	7,472 K	4412	Runtime Broker	Microsoft Corporation
SystemSettings.exe	Susp...	15,404 K	492 K	5220	Settings	Microsoft Corporation
SgmBroker.exe		2,728 K	5,052 K	1948	System Guard Runtime Monit...	Microsoft Corporation
svchost.exe		14,868 K	22,736 K	5980	Host Process for Windows S...	Microsoft Corporation
svchost.exe		10,612 K	18,196 K	5956		
svchost.exe		2,768 K	10,932 K	2364	Host Process for Windows S...	Microsoft Corporation
svchost.exe		2,264 K	9,240 K	5916	Host Process for Windows S...	Microsoft Corporation
svchost.exe		2,748 K	11,908 K	172	Host Process for Windows S...	Microsoft Corporation
WmiPrvSE.exe		1,972 K	8,344 K	2968		
explorer.exe		7,728 K	13,648 K	5964	Windows Explorer	Microsoft Corporation
svchost.exe		1,364 K	5,768 K	2760	Host Process for Windows S...	Microsoft Corporation
HW-A-3.exe	< 0.01	5,956 K	16,116 K	1372		
svchost.exe		2,080 K	8,884 K	6052	Host Process for Windows S...	Microsoft Corporation
svchost.exe		3,604 K	7,812 K	1484	Host Process for Windows S...	Microsoft Corporation
svchost.exe		2,348 K	7,028 K	5188	Host Process for Windows S...	Microsoft Corporation
svchost.exe		1,528 K	6,720 K	3188	Host Process for Windows S...	Microsoft Corporation
svchost.exe		1,716 K	6,724 K	3052	Host Process for Windows S...	Microsoft Corporation
svchost.exe		1,424 K	6,684 K	5920	Host Process for Windows S...	Microsoft Corporation
Registry		976 K	58,088 K	68		
System Idle Process	98.43	56 K	8 K	0		
System	< 0.01	192 K	152 K	4		
Interrupts	< 0.01	0 K	0 K	n/a	Hardware Interrupts and DPCs	

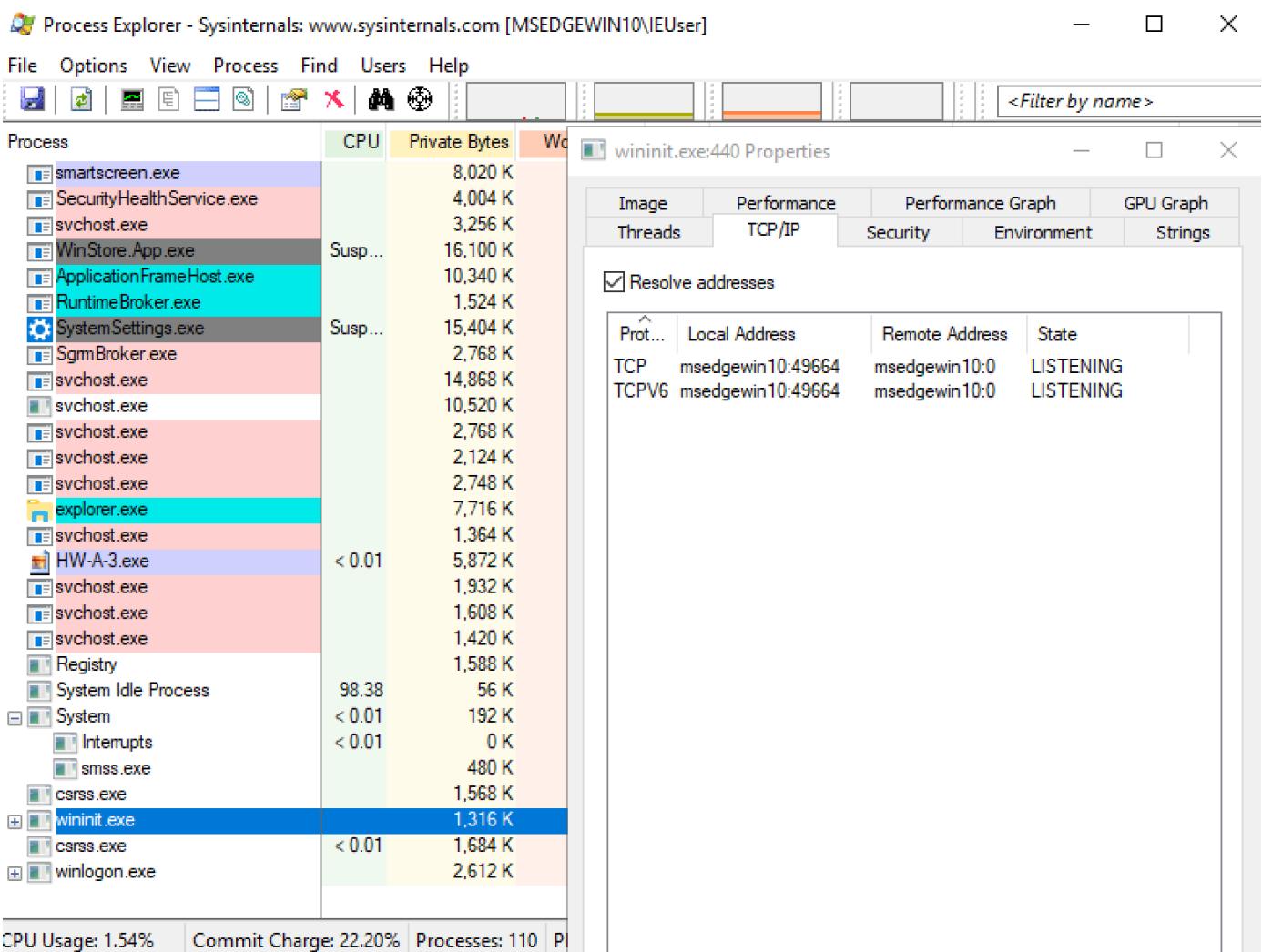
CPU Usage: 1.54% Commit Charge: 22.09% Processes: 114 Physical Usage: 28.93%

There are several instances of orphaned svchost.exe which are suspicious and unusual. They start new executable files such as TrustedInstaller.exe, SGrmBroker.exe, smartscreen.exe. Which disappears after some time.

The image shows three overlapping windows from the Process Explorer interface, each displaying the properties of a specific process. The windows are:

- TrustedInstaller.exe:5408 Properties**: Shows the 'Image' tab selected. Under 'Printable strings found in the scan:', it lists the string '!This program cannot be run in DOS mode.' and other memory dump details.
- smartscreen.exe:3272 Properties**: Shows the 'Image' tab selected. Under 'Printable strings found in the scan:', it lists the string '!This program cannot be run in DOS mode.' and other memory dump details.
- SGrmBroker.exe:1948 Properties**: Shows the 'Image' tab selected. Under 'Printable strings found in the scan:', it lists the string '!This program cannot be run in DOS mode.' and other memory dump details.

In the bottom right corner of the central window, there are 'Save', 'Find', 'OK', and 'Cancel' buttons.



It uses wininit.exe to snoop on the user. As evident by the TCP/IP connection

Using FAKENET we can see that the malware attempts to establish TCP connection to 192.0.2.123

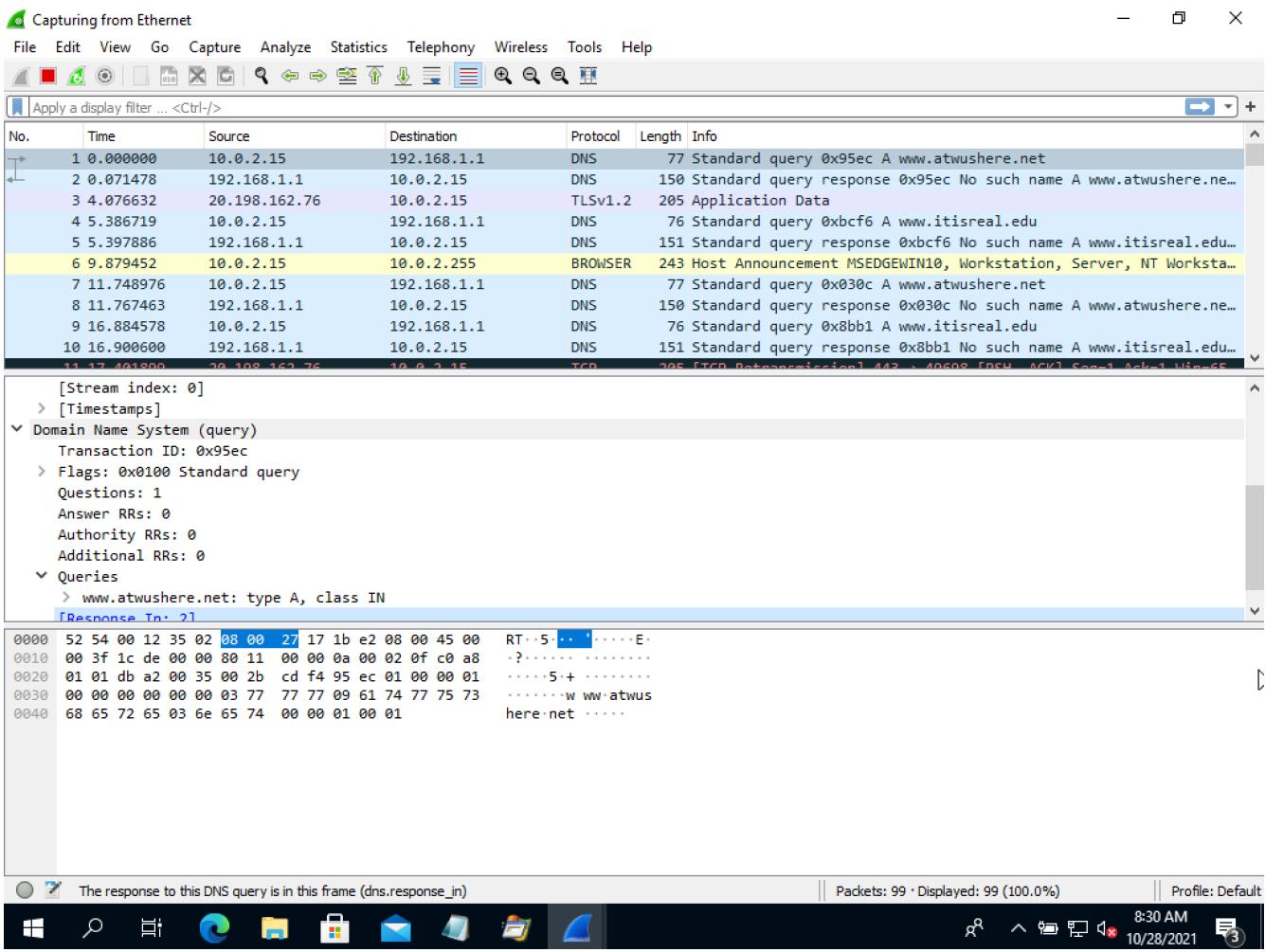
```

10/28/21 08:15:04 AM [           ] FakeNet] Loaded configuration file: configs\default.ini
10/28/21 08:15:04 AM [           ] Diverter] Capturing traffic to packets_20211028_081504.pcap
10/28/21 08:15:29 AM [           ]          FTP] >>> starting FTP server on 0.0.0.0:21, pid=4600 <<<
10/28/21 08:15:29 AM [           ]          FTP] concurrency model: multi-thread
10/28/21 08:15:29 AM [           ] Diverter] Failed getting registry value NameServer.
10/28/21 08:15:29 AM [           ]          FTP] masquerade (NAT) address: None
10/28/21 08:15:29 AM [           ]          FTP] passive ports: 60000->60010
10/28/21 08:15:29 AM [           ] Diverter] Failed to notify adapter change on {4AA86136-917B-45D2-BE98-087B589B8CA0}
10/28/21 08:15:29 AM [           ] Diverter] Failed getting registry value NameServer.
10/28/21 08:15:29 AM [           ] Diverter] Failed to notify adapter change on {DBC4E5E0-FFF0-43C9-A050-B1902B25E2A2}
10/28/21 08:15:29 AM [           ] Diverter] Failed to call OpenService
10/28/21 08:15:34 AM [           ] Diverter] svchost.exe (1256) requested UDP 192.168.1.1:53
10/28/21 08:15:34 AM [           ] DNS Server] Received A request for domain 'www.atwushere.net'.
10/28/21 08:15:34 AM [           ] Diverter] HW-A-3.exe (1372) requested TCP 192.0.2.123:80
10/28/21 08:15:35 AM [           ] Diverter] svchost.exe (1256) requested UDP 192.168.1.1:53
10/28/21 08:15:35 AM [           ] DNS Server] Received PTR request for domain '123.2.0.192.in-addr.arpa'.
10/28/21 08:15:39 AM [           ] DNS Server] Received A request for domain 'www.itisreal.edu'.
10/28/21 08:15:39 AM [           ] Diverter] HW-A-3.exe (1372) requested TCP 192.0.2.123:443
10/28/21 08:15:41 AM [           ] Diverter] svchost.exe (1256) requested UDP 192.168.1.1:53
10/28/21 08:15:41 AM [           ] DNS Server] Received A request for domain 'dns.msftncsi.com'.
10/28/21 08:15:46 AM [           ] DNS Server] Received A request for domain 'www.atwushere.net'.
10/28/21 08:15:46 AM [           ] Diverter] HW-A-3.exe (1372) requested TCP 192.0.2.123:80
10/28/21 08:15:51 AM [           ] Diverter] svchost.exe (1256) requested UDP 192.168.1.1:53
10/28/21 08:15:51 AM [           ] DNS Server] Received A request for domain 'www.itisreal.edu'.
10/28/21 08:15:51 AM [           ] Diverter] HW-A-3.exe (1372) requested TCP 192.0.2.123:443
10/28/21 08:15:57 AM [           ] Diverter] svchost.exe (1256) requested UDP 192.168.1.1:53
10/28/21 08:15:57 AM [           ] DNS Server] Received A request for domain 'www.atwushere.net'.
10/28/21 08:15:57 AM [           ] Diverter] HW-A-3.exe (1372) requested TCP 192.0.2.123:80

```

6. Review the Wireshark capture for network traffic generated by the malware.

The malware opens as evident in "Application Data"



1. It sends a standard DNS request to www.itisreal.edu destination IP 192.168.1.1
2. Then it attempts to open a browser, in this case is "MSEDGEWIN10"
3. Then it sends another DNS request www.atwushere.net.

**Summary:** The malware attempts to communicate whether the infected computer is real and if it is real establishes a connection to www.itisreal.edu and sends the location information via www.atwushere.net. Once confirmed it's real, it download files and attempts to decrypt sensitive user data.

154	201.238953	10.0.2.15	20.189.173.20	TCP	54 49952 → 443 [ACK] Seq=214 Ack=2921 Win=64240 Len=0
155	201.239093	20.189.173.20	10.0.2.15	TCP	1514 443 → 49952 [ACK] Seq=2921 Ack=214 Win=65535 Len=1460 [TCP seg...]
156	201.239094	20.189.173.20	10.0.2.15	TLSv1.2	76 Server Hello, Certificate, Server Key Exchange, Server Hello D...
157	201.239124	10.0.2.15	20.189.173.20	TCP	54 49952 → 443 [ACK] Seq=214 Ack=4403 Win=64240 Len=0
158	201.242274	10.0.2.15	20.189.173.20	TLSv1.2	212 Client Key Exchange, Change Cipher Spec, Encrypted Handshake M...
159	201.242499	20.189.173.20	10.0.2.15	TCP	60 443 → 49952 [ACK] Seq=4403 Ack=372 Win=65535 Len=0
160	201.487429	20.189.173.20	10.0.2.15	TLSv1.2	105 Change Cipher Spec, Encrypted Handshake Message
161	201.523967	10.0.2.15	20.189.173.20	TLSv1.2	1154 Application Data
162	201.524210	20.189.173.20	10.0.2.15	TCP	60 443 → 49952 [ACK] Seq=4454 Ack=1472 Win=65535 Len=0
163	201.524501	10.0.2.15	20.189.173.20	TCP	2974 49952 → 443 [ACK] Seq=1472 Ack=4454 Win=64189 Len=2920 [TCP se...
164	201.524505	20.189.173.20	10.0.2.15	TCP	60 443 → 49952 [ACK] Seq=1472 Ack=4454 Win=64189 Len=2920 [TCP se...

## 6. List the potential host-based indicators of this malware.

Can check if the file XxX.xXx is found in Users>IEUser>AppData>Local>Temp

The screenshot shows a Windows File Explorer window with the following details:

- Path:** <> Users > IEUser > AppData > Local > Temp >
- File Type:** All Files
- Search:** Search Temp
- Columns:** Name, Date modified, Type, Size
- Items:** 64 items
- Content:** A list of files in the Temp folder, including many TMP files and one XXX File named "XxX.xXx".

Name	Date modified	Type	Size
wct2B0A.tmp	10/18/2021 9:32 AM	TMP File	1 KB
wct3E0D.tmp	10/27/2021 10:05 ...	TMP File	41 KB
wct7D67.tmp	10/18/2021 9:32 AM	TMP File	1 KB
wct9EC3.tmp	10/21/2021 3:30 PM	TMP File	41 KB
wct71AE.tmp	10/26/2021 3:15 PM	TMP File	41 KB
wct415A.tmp	10/18/2021 9:32 AM	TMP File	1 KB
wct478.tmp	10/21/2021 3:30 PM	TMP File	41 KB
wct2197.tmp	10/27/2021 7:05 PM	TMP File	41 KB
wct2222.tmp	10/20/2021 4:00 PM	TMP File	41 KB
wct4963.tmp	10/27/2021 7:05 PM	TMP File	41 KB
wctA062.tmp	10/25/2021 5:40 PM	TMP File	41 KB
wctA116.tmp	10/18/2021 9:32 AM	TMP File	1 KB
wctAC77.tmp	10/25/2021 5:40 PM	TMP File	41 KB
wctB0DD.tmp	10/18/2021 9:32 AM	TMP File	1 KB
wctB1AA.tmp	10/27/2021 9:27 PM	TMP File	0 KB
wctBFF3.tmp	10/27/2021 8:22 AM	TMP File	46,899 KB
wctC2F2.tmp	10/21/2021 3:30 PM	TMP File	41 KB
wctC8B0.tmp	10/18/2021 9:32 AM	TMP File	1 KB
wctD7D5.tmp	10/18/2021 9:32 AM	TMP File	1 KB
wctD573.tmp	10/27/2021 7:05 PM	TMP File	41 KB
wctFBA4.tmp	10/26/2021 3:15 PM	TMP File	41 KB
XxX.xXx	10/28/2021 8:25 AM	XXX File	1 KB

We can check if the infected computer reboots and if the mouse becomes blocked.

explorer.exe

x\_X\_BLOCKMOUSE\_X\_x

The screenshot shows a command-line interface with the following output:

```
explorer.exe
explorer.exe
shell_traywnd
open
SV3
jJj
ZYYd
__PERSIST
Shell_TrayWnd
explorer.exe
QQQQQSV
UhJ
ZYYd
#####@#####
@jj
_x_X_BLOCKMOUSE_X_x_
Zyyd
```

\Software\Microsoft\Internet Explorer\IntelliForms\Storage2

```
[Y]
Software\Microsoft\Internet Explorer\IntelliForms\Storage2
ZYYd
HtSU
=t*f
SVW
YZ_^[
Uhm
]
ZYYd
jjj
ZYYd
Version
Software\Microsoft\Internet Explorer
jjj
```

IELOGIN.abc

IEPASS.abc

IEAUTO.abc

```
MSN.abc
FIREFOX.abc
IELOGIN.abc
IEPASS.abc
IEAUTO.abc
IEWEB.abc
```

We can also check if the following software is running on the infected computer.

SOFTWARE\Vitalwerks\DUC

```
ZYYd
UserName
SOFTWARE\Vitalwerks\DUC
Password
No-ip DUC|
PCREDENTIAL
```

## 7. List the potential network-based indicators of this malware.

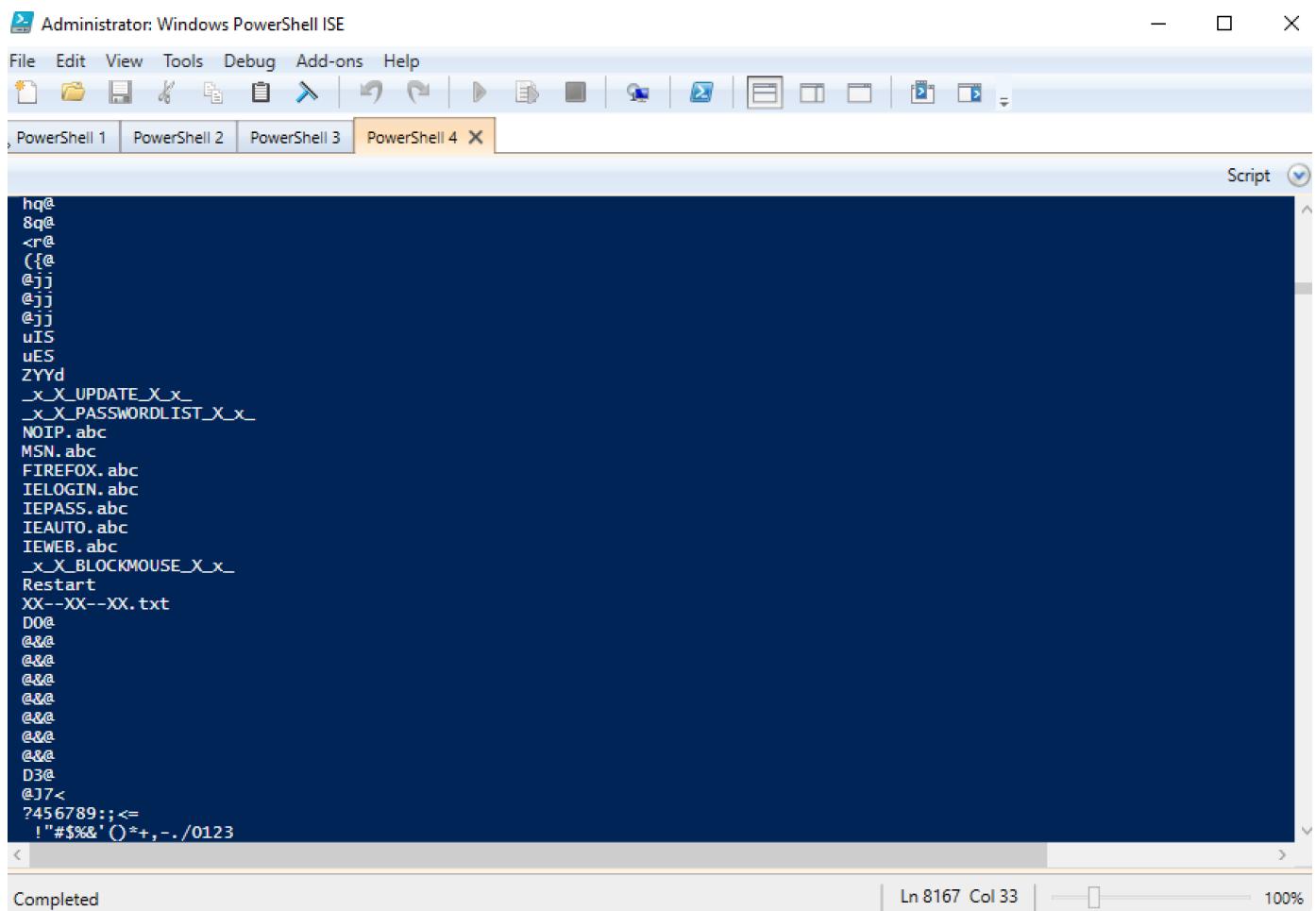
Can check if the infected computer has a TCP connection to 192.0.2.123 at port 443

Can check if the infected computer has TCP connection to 192.0.2.123 at port 80

Can check if the infected computer has a UDP connection to 127.0.0.1 at port 53

```
svchost.exe (1256) requested UDP 127.0.0.1:53
Received A request for domain 'www.itisreal.edu'.
HW-A-3.exe (1372) requested TCP 192.0.2.123:443
nc.exe (1848) requested TCP 10.0.2.15:49934
svchost.exe (1256) requested UDP 127.0.0.1:53
Received A request for domain 'www.atwushere.net'.
HW-A-3.exe (1372) requested TCP 192.0.2.123:80
```

FIREFOX.abc



Administrator: Windows PowerShell ISE

File Edit View Tools Debug Add-ons Help

PowerShell 1 PowerShell 2 PowerShell 3 PowerShell 4

Script

```
hq@  
8q@  
<r@  
({@  
@jj  
@jj  
@jj  
uIS  
uES  
ZYYd  
_x_X_UPDATE_X_X_  
_x_X_PASSWORDLIST_X_X_  
NOIP.abc  
MSN.abc  
FIREFOX.abc  
IELOGIN.abc  
IEPASS.abc  
IEAUTO.abc  
IEWEB.abc  
_x_X_BLOCKMOUSE_X_X_  
Restart  
XX--XX--XX.txt  
D0@  
@&@  
@&@  
@&@  
@&@  
@&@  
@&@  
D3@  
@J7<  
?456789:;<=  
!"#$%&'()^+, -./0123
```

Completed | Ln 8167 Col 33 | 100%

\Mozilla\Firefox

```
mozcrt19.dll
sqlite3.dll
nspr4.dll
plc4.dll
plds4.dll
nssutil3.dll
softokn3.dll
nss3.dll
NSS_Init
NSSBase64_DecodeBuffer
PK11_GetInternalKeySlot
PK11_Authenticate
PK11SDR_Decrypt
NSS_Shutdown
PK11_FreeSlot
userenv.dll
GetUserProfileDirectoryA
\Mozilla\Firefox\
profiles.ini
Path
Profile0
\signons3.txt
\signons2.txt
\signons1.txt
\signons.txt
##$$
(unnamed value)
(unnamed password)
ZYYd
```

\Software\Microsoft\Internet Explorer

```
=t*t
SVW
YZ_A[
Uhm
ZYYd
jjj
ZYYd
Version
Software\Microsoft\Internet Explorer
jjj
Uhx
ZYYd
```

Microsoft\Network\Connections\pbk\rasphone.pbk

```
RasDialIPParams!
L$_RasDefaultCredentials#0
Microsoft\Network\Connections\pbk\rasphone.pbk
WVS
[^_
ZYYd
^YY]
rasapi32.dll
```

\Internet Explorer\iexplore.exe

http\shell\open\command

```
ZTnra
http\shell\open\command
.exe
EXE
\Internet Explorer\iexplore.exe
QSVW
```

8. How many different domains do you think the malware can connect to? What are those domains?

2

```
svchost.exe (1256) requested UDP 127.0.0.1:53
Received A request for domain 'www.itisreal.edu'.
HW-A-3.exe (1372) requested TCP 192.0.2.123:443
nc.exe (1848) requested TCP 10.0.2.15:49934
svchost.exe (1256) requested UDP 127.0.0.1:53
Received A request for domain 'www.atwushere.net'.
HW-A-3.exe (1372) requested TCP 192.0.2.123:80
```

www.itisreal.edu

www.atwushere.net

## PE File Format

---

### 1. Output the following to standard output:

- Identify the file type as DLL or EXE or SYS regardless of the file's extension.

```
def get_file_type(pe):
    if pe.is_exe():
        return "EXE"
    if pe.is_dll():
        return "DLL"
    if pe.is_driver():
        return "SYS"
    return "UNKNOWN"
```

- Total number of imported DLLs.

```
def count_dlls(pe):
    i = 0
    for entry in pe.DIRECTORY_ENTRY_IMPORT:
        i = i+1
    return i
```

- Total number of imported functions.

```

def count_fns(pe):
    i = 0
    for entry in pe.DIRECTORY_ENTRY_IMPORT:
        for func in entry.imports:
            i = i+1

    return i

```

d. Output the compile time.

```

def get_compile_time(pe):
    return pe.FILE_HEADER.dump_dict()['TimeDateStamp']['Value'].split('[')[1][:-1]

```

2. Alert the user if the entry point of the code is not in a section with the name ".text", ".code", "CODE", or "INIT".

```

def get_valid_section_entry(pe):
    valid_sections = [".text", ".code", "CODE", "INIT"]
    entry = ""
    valid = False
    for section in pe.sections:
        if section.Name.decode().rstrip('\x00') == valid_sections[0]:
            entry = entry + valid_sections[0] + " "
            valid = True
        if section.Name.decode().rstrip('\x00') == valid_sections[1]:
            entry = entry + valid_sections[1] + " "
            valid = True
        if section.Name.decode().rstrip('\x00') == valid_sections[2]:
            entry = entry + valid_sections[2] + " "
            valid = True
        if section.Name.decode().rstrip('\x00') == valid_sections[3]:
            entry = entry + valid_sections[3] + " "
            valid = True

    if valid == False:
        return "Entry point NOT VALID."
    return entry

```

3. Automatically use the PEiD database that comes with pefile to identify packers. Confirm that this works with UPX. Output the detection to standard output.

```

def signature_match(pe, PEiD_database):
    signatures = peutils.SignatureDatabase(data=PEiD_database)
    matches = signatures.match_all(pe, ep_only=True)
    result = ''

    if matches != None:
        for match in matches:
            m = ','.join(match)
            result = result+m
    return result

else:
    return None

```

4. Calculate and output the entropy for each section. Alert the user when you suspect that a section maybe packed or compressed.

```

packers_sections = {
    #The packer/protector/tools section names/keywords
    '.aspack': 'Aspack packer',
    '.adata': 'Aspack packer/Armadillo packer',
    'ASPack': 'Aspack packer',
    '.ASPack': 'ASPAck Protector',
    '.boom': 'The Boomerang List Builder (config+exe xored with a single byte key 0x77)',
    '.ccg': 'CCG Packer (Chinese Packer)',
    '.charmve': 'Added by the PIN tool',
    'BitArts': 'Crunch 2.0 Packer',
    'DAStub': 'DAStub Dragon Armor protector',
    '!EPack': 'Epack packer',
    'FSG!': 'FSG packer (not a section name, but a good identifier)',
    '.gentee': 'Gentee installer',
    'kkrunchy': 'kkrunchy Packer',
    '.mackt': 'ImpRec-created section',
    '.MaskPE': 'MaskPE Packer',
    'MEW': 'MEW packer',
    '.MPRESS1': 'Mpress Packer',
    '.MPRESS2': 'Mpress Packer',
    '.neolite': 'Neolite Packer',
    '.neolit': 'Neolite Packer',
    '.nsp1': 'NsPack packer',
    '.nsp0': 'NsPack packer',
    '.nsp2': 'NsPack packer',
    'nsp1': 'NsPack packer',
    'nsp0': 'NsPack packer',
    'nsp2': 'NsPack packer',
    '.packed': 'RLPack Packer (first section)',
    'pebundle': 'PEBundle Packer',
    'PEBundle': 'PEBundle Packer',
    'PEC2TO': 'PECompact packer',
    'PECompact2': 'PECompact packer (not a section name, but a good identifier)',
    'PEC2': 'PECompact packer',
    'pec1': 'PECompact packer',
    'pec2': 'PECompact packer',
    'PEC2MO': 'PECompact packer',

```

```
'PELOCKnt': 'PELock Protector',
'.perplex': 'Perplex PE-Protector',
'PESHIELD': 'PEShield Packer',
'.petite': 'Petite Packer',
'petite': 'Petite Packer',
'.pinclie': 'Added by the PIN tool',
'ProCrypt': 'ProCrypt Packer',
'.RLPack': 'RLPack Packer (second section)',
'.rmnet': 'Ramnit virus marker',
'RCryptor': 'RPCrypt Packer',
'.RPCrypt': 'RPCrypt Packer',
'.seau': 'SeauSFX Packer',
'.sforce3': 'StarForce Protection',
'.spack': 'Simple Pack (by bagie)',
'.svkp': 'SVKP packer',
'Themida': 'Themida Packer',
'.Themida': 'Themida Packer',
'Themida ': 'Themida Packer',
'.taz': 'Some version os PESpin',
'.tsuarch': 'TSULoader',
'.tsustub': 'TSULoader',
'.packed': 'Unknown Packer',
'PEPACK!!!': 'Pepack',
'.Upack': 'Upack packer',
'.ByDwing': 'Upack Packer',
'UPX0': 'UPX packer',
'UPX1': 'UPX packer',
'UPX2': 'UPX packer',
'UPX!': 'UPX packer',
'.UPX0': 'UPX Packer',
'.UPX1': 'UPX Packer',
'.UPX2': 'UPX Packer',
'.vmp0': 'VMProtect packer',
'.vmp1': 'VMProtect packer',
'.vmp2': 'VMProtect packer',
'VProtect': 'Vprotect Packer',
'.winapi': 'Added by API Override tool',
'WinLicens': 'WinLicense (Themida) Protector',
'_winzip_': 'WinZip Self-Extractor',
'.WWPACK': 'WWPACK Packer',
'.yP': 'Y0da Protector',
'.y0da': 'Y0da Protector',
}
```

```
packers_sections_lower = {x.lower(): x for x in packers_sections.keys()}
```

```
def get_entropy(section):
    section_name = section.Name.decode().rstrip('\x00')

    # Your code here
    entropy = section.get_entropy()

    suspect_packed = False
    print("\t ", section_name, ":", entropy, end=" ")

    if section_name.lower() in packers_sections_lower.keys():
        suspect_packed == True
        print("Suspect packed")
    else:
        print()
    return
```

7. Alert the user when there is a zero sized section.

```
def is_zero_sized(section):
    data = section.get_data()
    try:
        # newbytes' count() takes a str in Python 2
        count = data.count("\0")
    except TypeError:
        # bytes' count() takes an int in Python 3
        count = data.count(0)

    if count == len(data):
        return True

    return False
```

8. Compare the PE Optional Header checksum with the actual checksum. Alert the user when

they don't match up.

```
def verify_checksum(pe):
    return pe.verify_checksum()
```

## 9. If there is a resource section, dump the first resource (of any type) to a file on disk.

```
def dump_rsrc(pe, dump_path):
    # If no resource section, return false;
    ret = {}
    if not hasattr(pe, 'DIRECTORY_ENTRY_RESOURCE'):
        return False
    # Otherwise, dump the first one to dump_path and return True.
    else:
        #For all possible resources
        for resource_type in pe.DIRECTORY_ENTRY_RESOURCE.entries:
            if resource_type.name is not None:
                name = "%s" % resource_type.name
            else:
                name = "%s" % pefile.RESOURCE_TYPE.get(resource_type.struct.Id)
            if name == None:
                name = "%d" % resource_type.struct.Id

            if hasattr(resource_type, 'directory'):

                #Try
                # resource_entry = resource_type.directory.entries[0]
                # resource_entry_directory = pe.parse_resource_entry(resource_entry.struct.OffsetToDirectory)
                # resource_entry_data = pe.parse_resource_data_entry(resource_entry.struct.OffsetToData)
                # print(resource_entry_data)
                # print(resource_entry_directory)

                ##Manual Loop through first resource
                resource_id = resource_type.directory.entries[0]
                if hasattr(resource_id, 'directory'):
                    for resource_lang in resource_id.directory.entries:
                        data = pe.get_data(resource_lang.data.struct.OffsetToData, resource_lang.data.struct.Size)
                        lang = pefile.LANG.get(resource_lang.data.lang, '*unknown*')
                        sublang = pefile.get_sublang_name_for_lang(resource_lang.data.lang, resource_lang.data.sublang )
                        ret[0] = (name, resource_lang.data.struct.OffsetToData, resource_lang.data.struct.Size, lang, sublang)

                #print(ret)
                f = open(dump_path, "a")
                f.write(str(ret))
                f.close()
    return True
```

## Results:

---

```
(base) yinyin@MacBook-Pro a3 % python3 hw3.py HW-A-1.ex_
Assignment 3: A0236589L Nandar Soe
[1]     Analysing file HW-A-1.ex_
[2.a]   File type: EXE
[2.b]   #DLLs: 3
[2.b]   #FNs: 8
[3]     Compile Time: Thu Nov 17 17:58:55 2011 UTC
[4]     Entry point found in Entry point NOT VALID.
[5]     Packer detected: UPX 2.90 [LZMA] -> Markus Oberhumer, Laszlo Molnar & John Reiser
[6]     Entropy for sections...
        UPX0 : 0.0 Suspect packed
        UPX1 : 7.8104290904126 Suspect packed
        UPX2 : 2.296234245718198 Suspect packed
[7]     Zero sized sections...
        UPX0
[8]     Checksum matched: False
[9]     No resource section has been found
```

```
(base) yinyin@MacBook-Pro a3 % python3 hw3.py HW-A-2.ex_
Assignment 3: A0236589L Nandar Soe
[1]     Analysing file HW-A-2.ex_
[2.a]   File type: EXE
[2.b]   #DLLs: 3
[2.b]   #FNs: 76
[3]     Compile Time: Tue Nov 22 03:35:10 2011 UTC
[4]     Entry point found in .text
[5]     Packer detected: None
[6]     Entropy for sections...
        .text : 6.443155266626159
        .rdata : 4.792233566888121
        .data : 2.4906229575121626
        .rsrc : 5.306569289651679
[7]     Zero sized sections...
        No zero sized section found.
[8]     Checksum matched: False
[9]     Dumped the first resource section to resource.txt
```

The screenshot shows a terminal window with the following content:

```
● ● ●
```

resource.txt

```
{0: ('RT_MANIFEST', 186664, 346, 'LANG_ENGLISH', 'SUBLANG_ENGLISH_US')}
```

```
(base) yinyin@MacBook-Pro a3 % python3 hw3.py HW-A-3.ex_
Assignment 3: A0236589L Nandar Soe
[1]     Analysing file HW-A-3.ex_
[2.a]   File type: EXE
[2.b]   #DLLs: 16
[2.b]   #FNs: 113
[3]     Compile Time: Fri Jun 19 22:22:17 1992 UTC
[4]     Entry point found in CODE
[5]     Packer detected: None
[6]     Entropy for sections...
        CODE : 6.41418124966198
        DATA : 2.7642577357404954
        BSS : 0.0
        .idata : 4.770955685378527
        .tls : 0.0
        .rdata : 0.20544562813451883
        .reloc : 6.245900512684813
        .rsrc : 7.957532959816957
[7]     Zero sized sections...
        BSS
        .tls
[8]     Checksum matched: False
[9]     Dumped the first resource section to resource.txt
```

[gd](#)