Malware Analysis Challenge

by Benjamin Feldman

Date: 02/12/2024

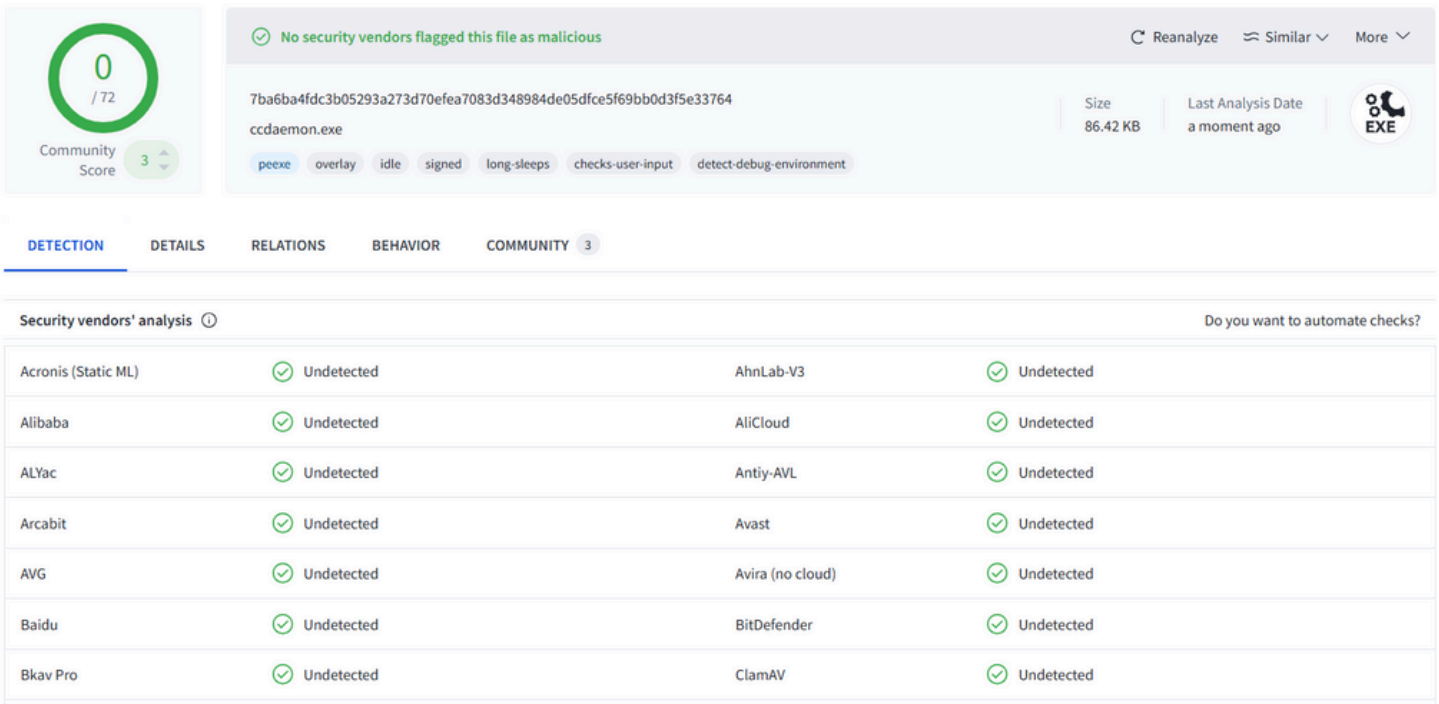
Phase 1: Hashes

I’m given 5 hashes to investigate and supply their threat level and any other necessary information.

Presenting in each header the file’s signature.

1st File - 7BA6BA4FDC3B05293A273D70EFEA7083D348984DE05DFCE5F69BB0D3F5E33764

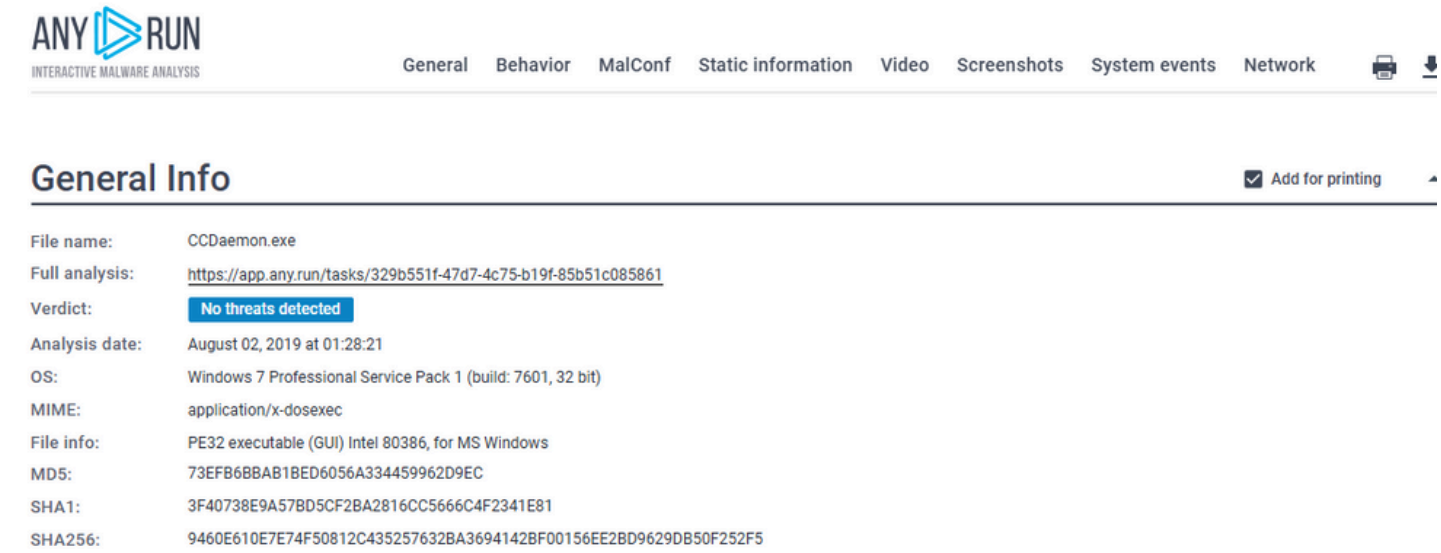
Always as a first step using Virus-Total web engine as a trusted source to analyze and examine the files I’m investigating, searching the hash in the databases of all these engines we’ve come to a result of the file being completely okay.



Continuing with a simple Google search for this executable file’s name and hash i have not found a case that this specific program was used maliciously.

Found a redirect for a site called “Any.Run” this site provides services as a Sandbox in the cybersecurity community to analyze any program given to it, in exchange for saving the file’s content and signatures to add it to a database incase it’s in-fact malicious.

Presented in the website is the verdict of the file not being a threat written in the blue box.

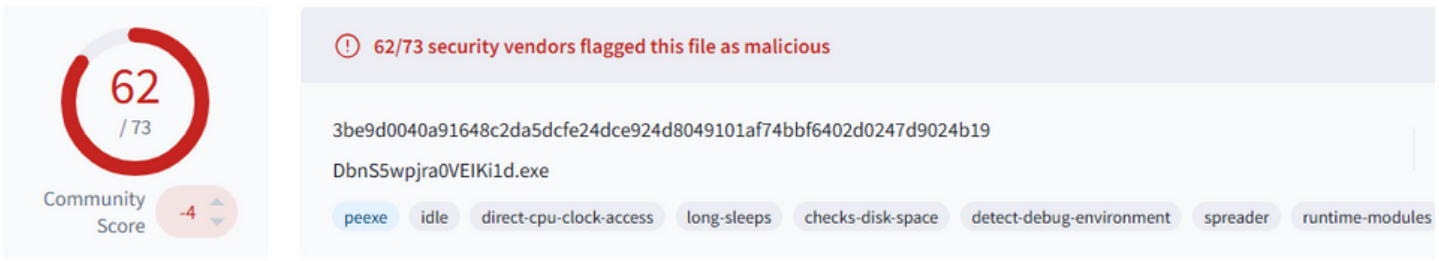


Based on these results above I concluded that this file is a non-threat.

2nd File - 3be9d0040a91648c2da5dcfe24dce924d8049101af74bbf6402d0247d9024b19

Again running a search for this SHA-256 signature in the Virus-Total engine, we're seeing a score of the file being 62/73 marked all red.

This means most engines found this file malicious and as a threat, labeled **trojan.emotet/dovs**



"Emotet" trojan file is a malicious malware that is being transmitted through mail attachments and links, from one victim to another by spamming mails from the infected system.

It acts as a kind of trojan delivered by mail, downloading other malware after being infected by itself, and also trying to get as many users infected and steal their banking information, passwords and emails.

the malware establishes a persistency by using Run registry keys, scheduled tasks, and services. It connects to the attacker's command-and-control (C2) network and exfiltrates the collected data.

Entering the details tab in Virus-Total which presenting the facts about this file, such as Properties , Creation time, File type, File Names ,Sections , imported dynamic link libraries.

The most important for us is the file type and the imported dlls.

The file type being is a Windows portable executable file, meaning it can only run on Windows Operating system, and the dynamic link libraries that are being imported by the program are - Secur32.dll, ADVAPI32.dll, WININET.dll , msi.dll, USER32.dll , KERNEL32.dll, VERSION.dll, GDI32.dll

These kind of libraries can have a really large number of functions that they can perform, meaning it can be of use to many things, just the “kernel32.dll” can have an impact on processes and files on the system which it does by using “GetProcessHeap” function.

Skipping to the third tab “relations” in Virus-Total, there we can view much information about this Portable executable file.

Contacted URLs, Contacted IP Addresses and dropped files shown in the pictures below.

Contacted URLs (1)

Scanned	Detections	Status	URL
2024-11-14	12 / 96	-	http://212.5.159.61:7080/

First lead is the contacted URL (http://212.5.159.61:7080/), a website that the APT (Advance Persistence Threat) most likely used to send the collected data to by the malware.

Contacted IP addresses (23)

IP	Detections	Autonomous System	Country
104.71.214.69	0 / 94	16625	US
178.79.208.1	1 / 94	22822	NL
184.25.191.235	0 / 94	16625	US
192.168.0.40	0 / 94	-	-
192.168.0.46	0 / 94	-	-
192.168.0.58	0 / 94	-	-
192.229.211.108	0 / 94	15133	US
20.62.24.77	0 / 94	8075	US
20.99.133.109	1 / 94	8075	US
20.99.184.37	2 / 94	8075	US
20.99.185.48	1 / 94	8075	US
20.99.186.246	0 / 94	8075	US
204.79.197.203	2 / 94	8068	US
212.5.159.61	13 / 94	8866	BG
23.216.147.64	2 / 94	20940	US
23.216.147.76	1 / 94	20940	US
23.216.81.152	0 / 94	16625	US
23.62.210.8	0 / 94	16625	US
52.154.209.174	0 / 94	8075	US
64.182.125.6	0 / 94	54489	US

Down below, we can see the dropped files by this specific portable executable, indicating a use of function “WriteFile” by one of the imported dlls.

The files dropped by the malware contain a score, confirming the behavioral of an “Emotet” that dropped a second Malware.

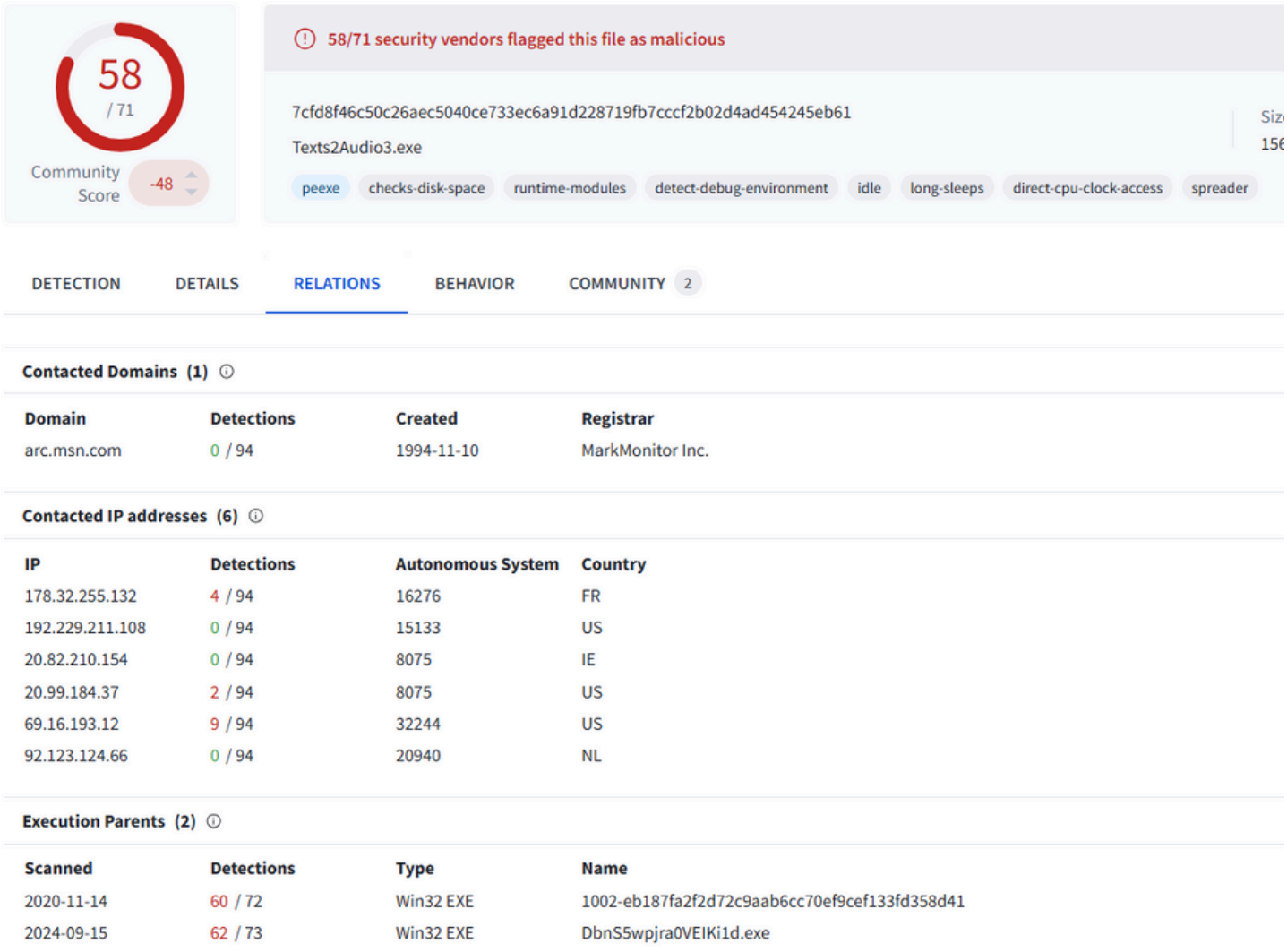
Dropped Files (2)

Scanned	Detections	File type	Name
2024-12-09	0 / 61	JSON	Download-1.tmp
2023-12-30	58 / 71	Win32 EXE	Texts2Audio3.exe
SHA-256	7cfd8f46c50c26aec5040ce733ec6a91d228719fb7cccf2b02d4ad454245eb61		
File Size	156.00 KB		

Name: “Text2Audio3.exe”

Hash: “7cfd8f46c50c26aec5040ce733ec6a91d228719fb7cccf2b02d4ad454245eb61"

Searching for the related file in the engine, we find another related file, that can be added to our investigations.



name : “1002-eb187fa2f2d72c9aab6cc70ef9cef133fd358d41"

hash : “3aa3467100088e053f0fad6c52a46bb3c12e392a5b37ff26597b781943c9b58e”

In conclusion I’ve decided to give this file a medium risk considering it’s a very easy and likely to encounter it in the mail, and by the risk of harm that could be done by itself if it can overcome the protection system installed on the system, depending on it’s software and version.

3rd File - bf0b450b758d57781b6197bec222198434f5b4fcf8da741de97c07c10765af85

From first sight after searching the file's hash in VT it's only flagged by one vendor as malicious, but after conducting a more thorough investigation we find out it's a false positive

looking up the md5 sum of the file, according to Source(1)

the file hasn't been reported as being used for malicious purposes, it only may be considered as infected if it consumes a lot of memory and CPU.

the second time it may be malicious is if it's started and dropped by malware, which we can check by looking up the related file hash signature and name.

Execution Parents (2) ⓘ

Scanned	Detections	Type	Name
2021-05-23	60 / 70	Win32 EXE	d97362005f4a3e82bb78148dcfa2f7ea.virus
2021-05-23	60 / 70	Win32 EXE	e1a4b701262bb4a6c1848473d3f3384e1e713e845345bc0268ca3181669367d0

Name: d97362005f4a3e82bb78148dcfa2f7ea.virus

Hash: 7438fa2cc311a2175a23d23f3ca7ae4e5bcbd2548047e0deb6ead82341c58050

or

Hash: e1a4b701262bb4a6c1848473d3f3384e1e713e845345bc0268ca3181669367d0

Concluding the results in this investigation, I categorize this file as being a low risk, due to it may be an attempt to exploit the system by some other malware that uses this tool.

4th File - 269253135ed7108a0981a821dcbd41b5f3037e2f55bba790dba5955287344efd

searching up the signature in VT - immediately seeing results of the file being a virus with a label with "**trojan.blocker/mint Ransomware**", this means the malware encrypts a bunch of files (making them unreadable or usable) and uses them as leverage to make a company pay a bunch money to get their files back to previous state.

In the "community" tab in VT, we can find a research someone has done on this specific file, in a sandbox environment, providing a full report about it's functionality.

The analysis is provided by "Recorded Future Triage" (Source - 2)

Looking at the report we can see exactly which programs and commands this Ransom uses,

a few commands and their explanations:

"C:\Windows\System32\cmd.exe" /C vssadmin.exe Delete Shadows /All /Quiet”

This process is launched by the parent process the malware itself, starting a command interpreter and using “vssadmin.exe” - a default Windows process that controls volume shadow duplicates of the documents on a given PC to Delete all the shadow copies made by system’s processes.

This tactic is used to make system administrators not have any way to recover the system after the ransom attacks it.

"C:\Windows\System32\cmd.exe" bcdedit /set {default} bootstatuspolicy ignoreallfailures “

This command also launched via command line interpreter CMD, uses bcdedit.exe a boot configuration data tool to modify the boot options, so that when the system would need to be repaired, the system administrator wouldn’t be able to launch windows’ recovery environment

I find this file being at threat level of High risk, even thought it’s not looking like it has intention of spreading itself in the system, only encrypting it’s host with the ransomware.

5th file - eec5d2f069cac9efb4d0f8b66fa778e405dd3fe61d0116036f1ad93457fbaa75

searching up the Hash of the file in VT -

The score is 57/72 meaning it’s flagged by most vendors in the system.

the threat label is **adware.elex/adwaresig**

Once installed on a system, the adware can serve advertisements to the user. This often involves generating pop-up windows containing advertising sites. Since the adware operator receives revenue from the advertiser for every view of the ads, the adware can provide revenue to the malware operator.

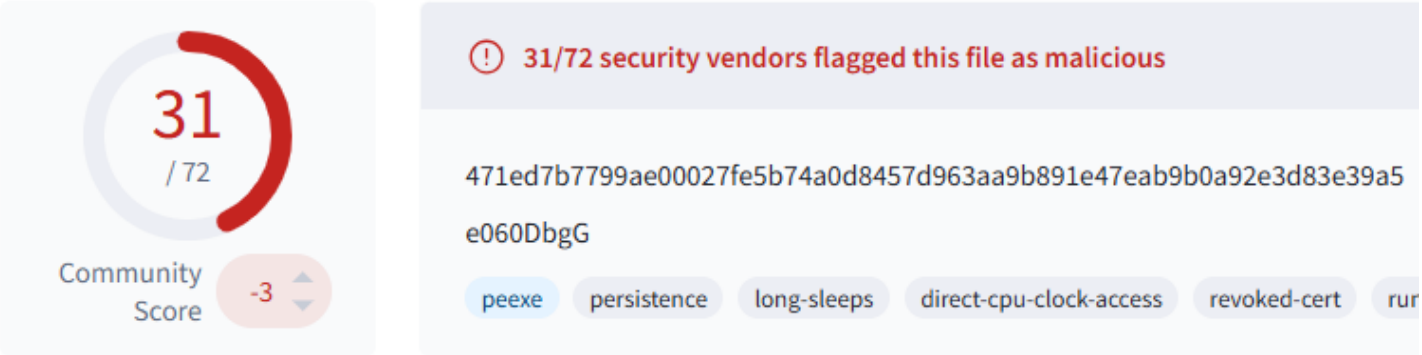
Adware itself isn’t always harmful as other viruses on the internet, but sometimes they do pose a security risk, such as; Data Theft , Malware Delivery , Vulnerabilities , Man-in-the-

Middle Attacks.

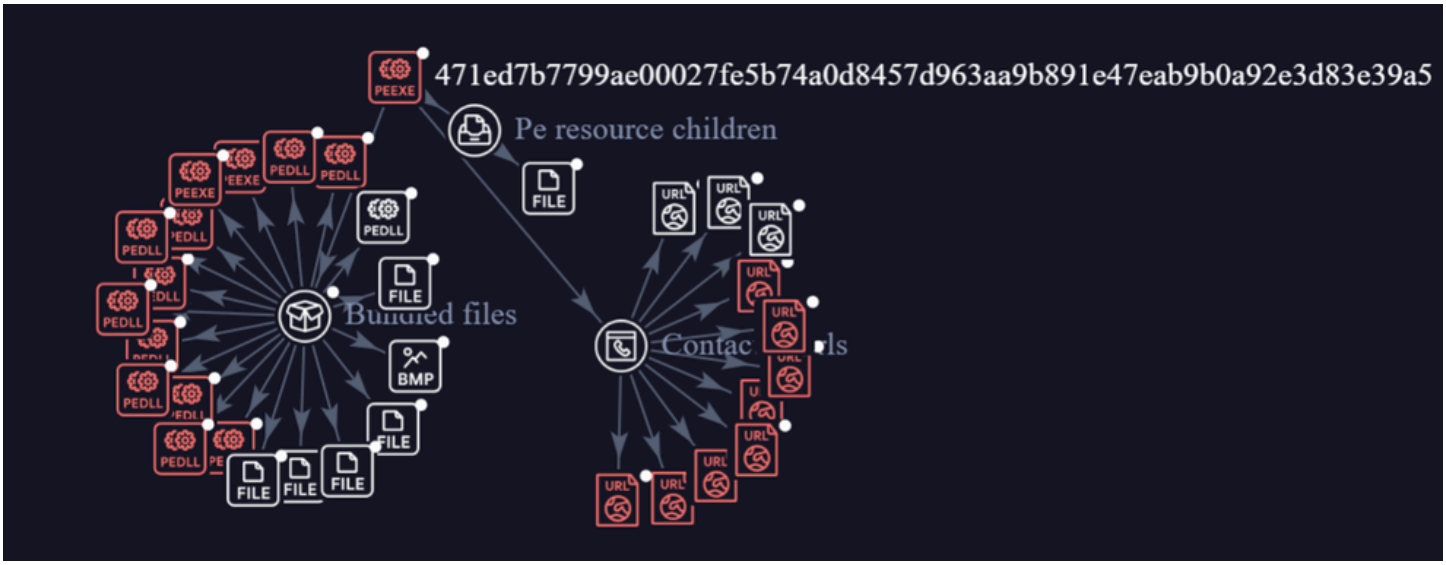
According to malwarebytes (source 3) this type of malware is of a large family Windows-oriented adware of Chinese origin. Adware.Elex arrives on a system as a file downloaded from the Internet. Sometimes it disguises itself as a tool that can detect and remove adware. At times, it hides under the guise of an Adobe Flash or Java update. Adware.Elex can also be dropped by “Trojan.Elex” which has been known to use rootkits.

Last file - 471ed7b7799ae00027fe5b74a0d8457d963aa9b891e47eab9b0a92e3d83e39a5

This Portable executable is also flagged by the engine as malicious, It is Adware.



The Graph summary below shows lots of related to the bin portable executable file , that all together construct a virus.



The threat level I would give this file is **low**.

Not a chance it would bypass up to date anti-virus software.

Sources -

1. https://processchecker.com/file/jre-8u65-windows-au.exe.html
2. https://tria.ge/230425-v1gr9sbh36/behavioral1
3. https://www.malwarebytes.com/blog/detections/adware-elex
4. https://www.virustotal.com/graph/471ed7b7799ae00027fe5b74a0d8457d963aa9b891e47eab9b0a92e3d83e39a5

Phase 2: Malware

Q.1. Please provide 3 different types of malwares -

A.1: Torjan -

The purpose of this malware is to enter while pretending to be an innocent program\tool,

sometimes being downloaded from a sketchy source or a program that was received by someone reaching out.

A.2: Worm -

A Virus that spreads itself and multiples as a strategy to remain persistence.

This type of virus may try to exploit many types of systems such as IoT to run bot net.

worm virus exploits vulnerabilities in your security software to steal sensitive information, install backdoors that can be used to access the system, corrupt files, and do other kinds of harm

A.3: Adware -

Adware is something you should never underestimate as it could be an incident of a system breach, but at times could just be a sign of weakness in the system.

It’s something that at times would be installed on systems by itself exploiting a vulnerability in the software along the way.

Adware’s persistence may not be high but it definitely gets more malware onto your way.

Phase 3: Malware Analysis

File under investigation:
e142a1e51ce0e8d28fd852683b65688dcc97a6b705e8adc799d5af0bdefefecf.bin

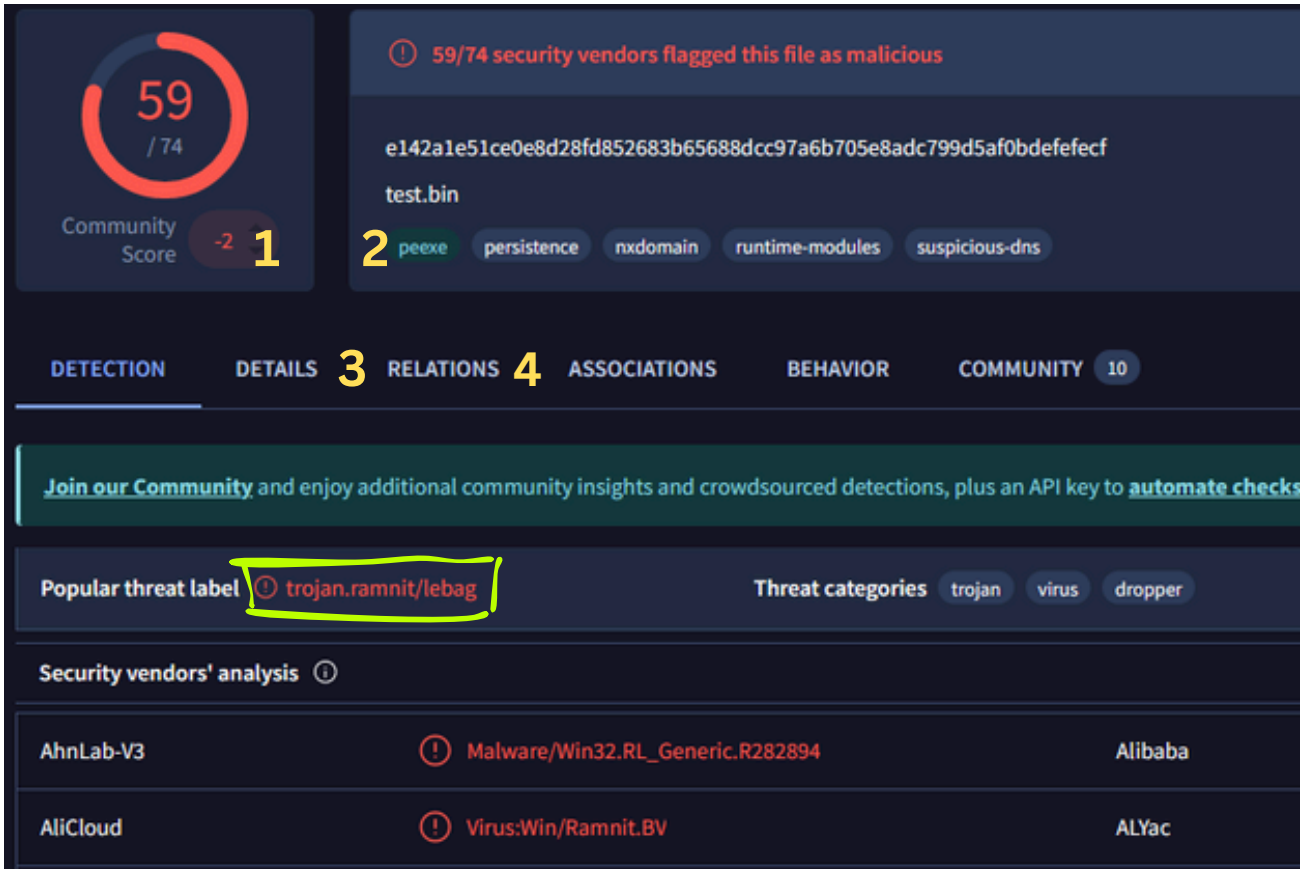
First, Right after receiving the sample I'm going to be investigating I immediately ran it throughout **VirusTotal web site.**

Indicators numbered by numbers 1 to 4 are marking the score (1), the file header(2) , details (3) and relations(4).

This specific file is signature signed by multiple Anti-virus engines being from "Ramnit malware " family as can be noticed. (Yellow brackets)

The bin file is indeed a Windows Operating system portable executable with malicious intent.

From a simple google search I found out that Ramnit is highly modular banking trojan that is intended to steal account credentials for online banking, but also collects all kinds of credentials for example social media,email and other accounts and may deploy other malware.



In the "Details" (3) section of the file's results in VirusTotal; checking the binary file's signature hashes.

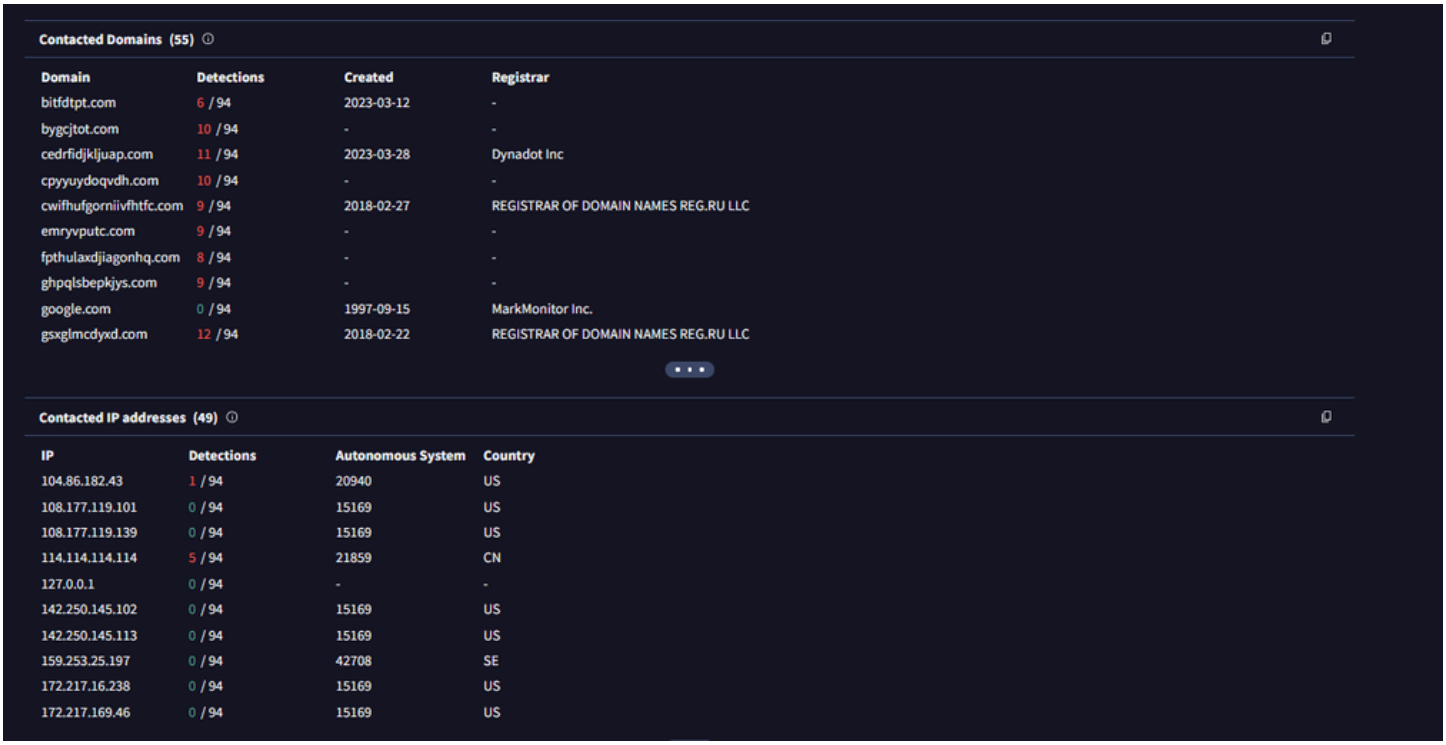
MD5 118962ea993c489f14c068235b1a8397

SHA-1 0f6e1c3388f65c6f483b15e6a35b8acdf0a93de6

SHA-256 e142a1e51ce0e8d28fd852683b65688dcc97a6b705e8adc799d5af0bdefefecf

Moving forward looking at the information related to this file - Relations tab(4)

In the relations sections of the file; can see all domains/IP addresses that might be contacted after executing the bin.



Contacted Domains (55)

Domain	Detections	Created	Registrar
bitfdtpt.com	6 / 94	2023-03-12	-
bygcjtot.com	10 / 94	-	-
cedrfidjkljuap.com	11 / 94	2023-03-28	Dynadot Inc
cpyyuydoqvdh.com	10 / 94	-	-
cwifhufgorniivfhtfc.com	9 / 94	2018-02-27	REGISTRAR OF DOMAIN NAMES REG.RU LLC
emryvputc.com	9 / 94	-	-
fpthulaxdjiagonhq.com	8 / 94	-	-
ghpqlsbepkjys.com	9 / 94	-	-
google.com	0 / 94	1997-09-15	MarkMonitor Inc.
gsxglmcdyxd.com	12 / 94	2018-02-22	REGISTRAR OF DOMAIN NAMES REG.RU LLC

Contacted IP addresses (49)

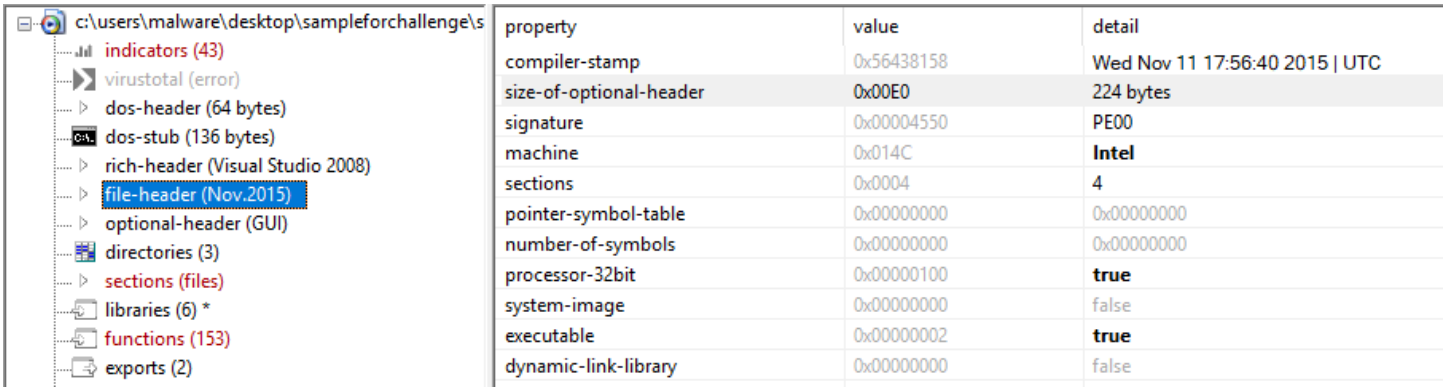
IP	Detections	Autonomous System	Country
104.86.182.43	1 / 94	20940	US
108.177.119.101	0 / 94	15169	US
108.177.119.139	0 / 94	15169	US
114.114.114.114	5 / 94	21859	CN
127.0.0.1	0 / 94	-	-
142.250.145.102	0 / 94	15169	US
142.250.145.113	0 / 94	15169	US
159.253.25.197	0 / 94	42708	SE
172.217.16.238	0 / 94	15169	US
172.217.169.46	0 / 94	15169	US

First step in my Static analysis was launching a Virtual Environment that could be vulnerable to this trojan attack - A Sand Box to conduct a static malware analysis.

Using a Windows 7 launched via VMware that is prepared exactly for this purpose with the right tools on the machine to perform all the research required.

Second step is using a very useful tool called PEstudio.

The file-header section contained a date and time the executable was compiled 11/11/2015 17:56:40 .

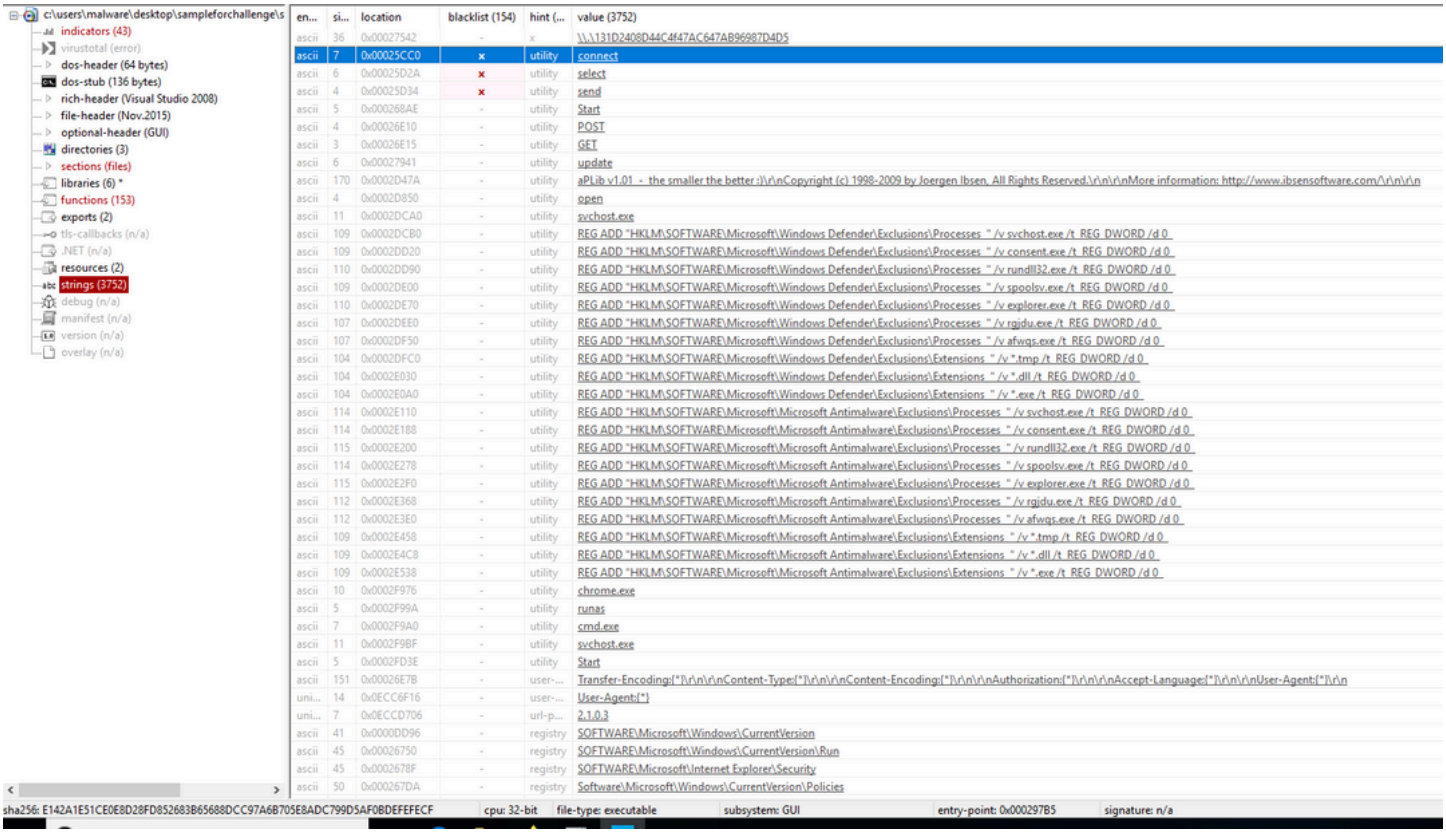


c:\users\malware\desktop\sampleforchallenge\s
- indicators (43)
- virustotal (error)
- dos-header (64 bytes)
- dos-stub (136 bytes)
- rich-header (Visual Studio 2008)
- file-header (Nov.2015)
- optional-header (GUI)
- directories (3)
- sections (files)
- libraries (6) *
- functions (153)
- exports (2)

property	value	detail
compiler-stamp	0x56438158	Wed Nov 11 17:56:40 2015 | UTC
size-of-optional-header	0x00E0	224 bytes
signature	0x00004550	PE00
machine	0x014C	**Intel**
sections	0x0004	4
pointer-symbol-table	0x00000000	0x00000000
number-of-symbols	0x00000000	0x00000000
processor-32bit	0x00000100	**true**
system-image	0x00000000	false
executable	0x00000002	**true**
dynamic-link-library	0x00000000	false

We’re looking for any signs or things that could assist us in the investigations.

I performed a “strings” search on the bin file, this kind of search would find leads on contained information inside, like shown in the picture below.



I did find many interesting strings but I’m going to be concentrating on these written below.

Suspected strings that indicate presence of other executables:

VWRQRh.exe

h.exe

SRQVWh.exe

tvh.exe

Registry keys that might be changed :

SOFTWARE\Microsoft\Windows\CurrentVersion\Run

SOFTWARE\Microsoft\Internet Explorer\Security DisableSecuritySettingsCheck

Other related informative strings that can be useful later on:

“**aPLib v1.01 - the smaller the better :)**”

- this specific string indicates that this file is packed.

“**%temp%\..\..\LocalLow\cmd.%username%.bat**”

- the second one is a command that is gonna be run and execute a batch file located in the so mentioned path in the string.

“**connect**”

- last indicator is signaling a connection attempt would be performed by the said file.

Entering the import tab of the portable executable we can see a few dynamic link libraries, and the function that they’re performing if launched.

These Dynamic link libraries(DLL) can have different functions aswell as calling another DLL.

library (6)	blacklist (0)	type (1)	functions (153)	description
kernel32.dll	-	implicit	87	Windows NT BASE API Client DLL
advapi32.dll	-	implicit	26	Advanced Windows 32 Base API
gdi32.dll	-	implicit	8	GDI Client DLL
shell32.dll	-	implicit	1	Windows Shell Common Dll
shlwapi.dll	-	implicit	3	Shell Light-weight Utility Library
user32.dll	-	implicit	28	Multi-User Windows USER API Client DLL

for instance KERNEL32.DLL is gonna be creating a process which I would need to capture it later and look what kind is it and what’s its name.

WriteFile	x	-	kernel32.dll
WriteProcessMemory	x	-	kernel32.dll
CreateProcessA	x	-	kernel32.dll

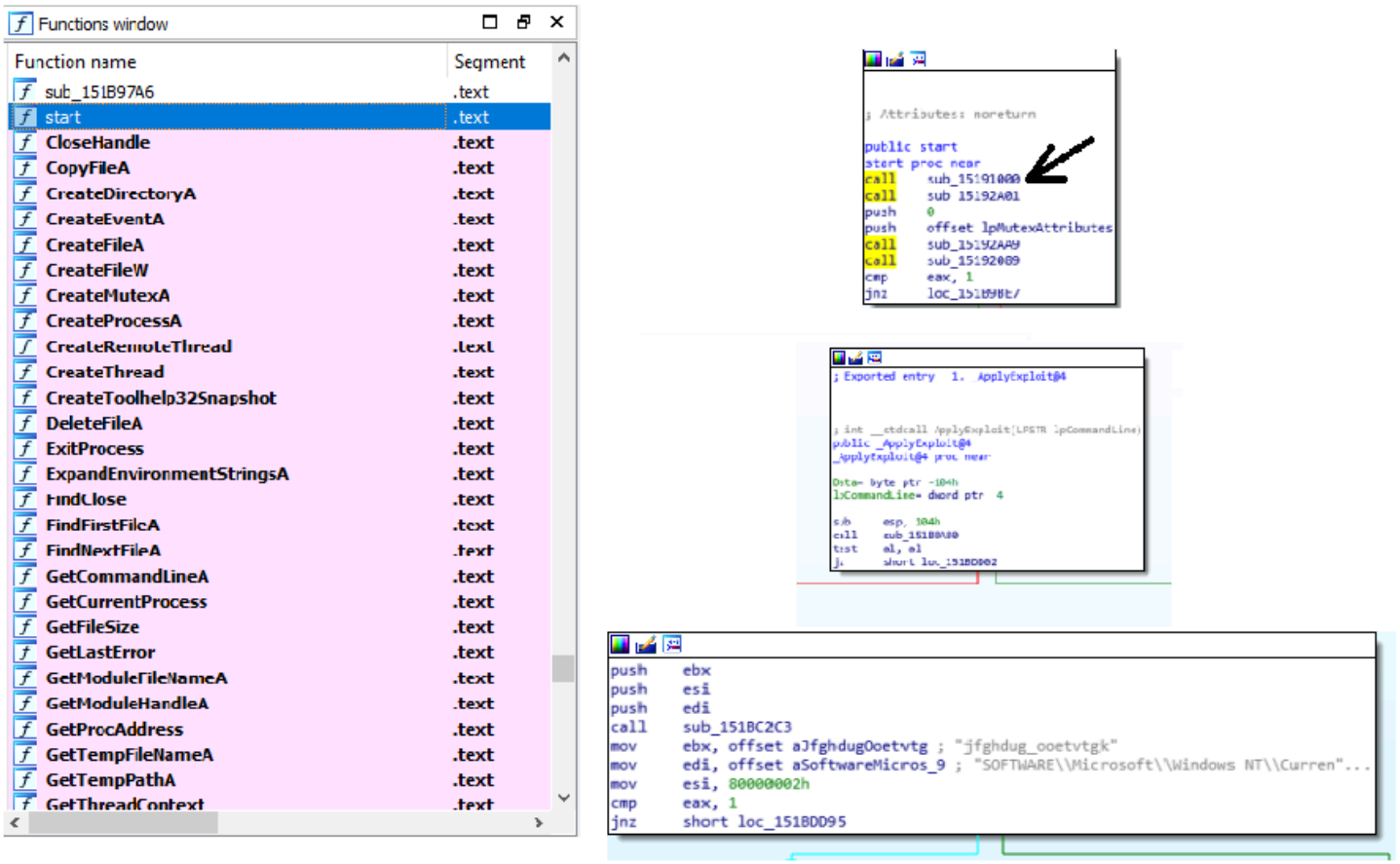
Export section showing two different modules with suspicious names as can be seen below.

indicating a presence of a malicious module that is going to be used to check variables while running the malware.

Ordinal	RVA	Name	
0001	0002dc9e	00030287	_ApplyExploit@4
0002	0002dbce	00030297	_CheckBypassed@0

Opening a disassembly program, used to reverse engineer many malware programs,

we getting started looking at the functions' names and their calls.



Starting at the top we get an the "CheckBypassed" entry.

By checking if the program has set the needs to exploit the system, like those below.

```
.rdata:151BEEA0 aRegAddHklmSoft db 'REG ADD "HKLM\SOFTWARE\Microsoft\Windows Defender\Exclusions\Exte'
.rdata:151BEEA0                                         ; DATA XREF: sub_151BD4FA↑o
.rdata:151BEEA0                 db 'nsions  " /v *.exe /t  REG_DWORD /d 0  ',0
.rdata:151BEF09                 align 10h
.rdata:151BEF10 aRegAddHklmSoft_18 db 'REG ADD "HKLM\SOFTWARE\Microsoft\Microsoft Antimalware\Exclusions'
.rdata:151BEF10                                         ; DATA XREF: .text:151BD5DC↑o
.rdata:151BEF10                 db '\Processes  " /v svchost.exe /t  REG_DWORD /d 0  ',0
.rdata:151BEF83                 align 8
.rdata:151BEF88 aRegAddHklmSoft_17 db 'REG ADD "HKLM\SOFTWARE\Microsoft\Microsoft Antimalware\Exclusions'
.rdata:151BEF88                                         ; DATA XREF: .text:151BD5D0↑o
.rdata:151BEF88                 db '\Processes  " /v consent.exe /t  REG_DWORD /d 0  ',0
.rdata:151BEFFB                 align 10h
.rdata:151BF000 aRegAddHklmSoft_16 db 'REG ADD "HKLM\SOFTWARE\Microsoft\Microsoft Antimalware\Exclusions'
.rdata:151BF000                                         ; DATA XREF: .text:151BD5C4↑o
.rdata:151BF000                 db '\Processes  " /v rundll32.exe /t  REG_DWORD /d 0  ',0
.rdata:151BF074                 align 8
.rdata:151BF078 aRegAddHklmSoft_15 db 'REG ADD "HKLM\SOFTWARE\Microsoft\Microsoft Antimalware\Exclusions'
.rdata:151BF078                                         ; DATA XREF: .text:151BD5B8↑o
.rdata:151BF078                 db '\Processes  " /v spoolsv.exe /t  REG_DWORD /d 0  ',0
.rdata:151BF0EB                 align 10h
.rdata:151BF0F0 aRegAddHklmSoft_14 db 'REG ADD "HKLM\SOFTWARE\Microsoft\Microsoft Antimalware\Exclusions'
.rdata:151BF0F0                                         ; DATA XREF: .text:151BD5AC↑o
.rdata:151BF0F0                 db '\Processes  " /v explorer.exe /t  REG_DWORD /d 0  ',0
.rdata:151BF164                 align 8
.rdata:151BF168 aRegAddHklmSoft_13 db 'REG ADD "HKLM\SOFTWARE\Microsoft\Microsoft Antimalware\Exclusions'
.rdata:151BF168                                         ; DATA XREF: .text:151BD5A0↑o
.rdata:151BF168                 db '\Processes  " /v rgjdu.exe /t  REG_DWORD /d 0  ',0
.rdata:151BF1D9                 align 10h
.rdata:151BF1E0 aRegAddHklmSoft_12 db 'REG ADD "HKLM\SOFTWARE\Microsoft\Microsoft Antimalware\Exclusions'
.rdata:151BF1E0                                         ; DATA XREF: .text:151BD594↑o
.rdata:151BF1E0                 db '\Processes  " /v afwqs.exe /t  REG_DWORD /d 0  ',0
.rdata:151BF251                 align 8
.rdata:151BF258 aRegAddHklmSoft_11 db 'REG ADD "HKLM\SOFTWARE\Microsoft\Microsoft Antimalware\Exclusions'
.rdata:151BF258                                         ; DATA XREF: .text:151BD588↑o
.rdata:151BF258                 db '\Extensions  " /v *.tmp /t  REG_DWORD /d 0  ',0
.rdata:151BF2C6                 align 4
.rdata:151BF2C8 aRegAddHklmSoft_10 db 'REG ADD "HKLM\SOFTWARE\Microsoft\Microsoft Antimalware\Exclusions'
.rdata:151BF2C8                                         ; DATA XREF: .text:151BD57C↑o
.rdata:151BF2C8                 db '\Extensions  " /v *.dll /t  REG_DWORD /d 0  ',0
.rdata:151BF336                 align 4
.rdata:151BF338 aRegAddHklmSoft_9 db 'REG ADD "HKLM\SOFTWARE\Microsoft\Microsoft Antimalware\Exclusions'
.rdata:151BF338                                         ; DATA XREF: .text:loc_151BD572↑o
.rdata:151BF338                 db '\Extensions  " /v *.exe /t  REG_DWORD /d 0  ',0
.rdata:151BF3A6                 align 4
```

Right after that if not exploited yet, it continues and develops the exploit and adds Exclusions to the Windows Anti Virus Software it creates a gap that the malware can use to run itself no questions asked.

The call “Apply Exploit” entry uses A technique - ntdll.dll and IsWow64Process to set persistence.

```
; CHAR aIswow64process[]
aIswow64process db 'IsWow64Process',0   ; DATA XREF: sub_151BBDC6+D↑o
                align 4
; CHAR aComSSdb[]
aComSSdb        db 'com.%s.sdb',0       ; DATA XREF: sub_151BBE87+3A↑o
                align 10h
; CHAR aSCmdSBat[]
aSCmdSBat       db '%s\cmd.%s.bat',0    ; DATA XREF: sub_151BBF21+70↑o
                align 10h
; CHAR aUsername[]
aUsername       db 'username',0         ; DATA XREF: sub_151BBF21+57↑o
                align 4
; CHAR aLocallow[]
aLocallow       db '\..\..\LocalLow\',0 ; DATA XREF: sub_151BBF21+39↑o
                                        ; sub_151BC087+1C↑o ...
                align 10h
; CHAR aStartS[]
aStartS         db 'start "" "%s"',0    ; DATA XREF: sub_151BBFA4+6D↑o
                align 10h
; CHAR aWindirSIscsicl[]
aWindirSIscsicl db '"%%windir%%\%s\iscsicli.exe"',0
                                        ; DATA XREF: sub_151BC087+DE↑o
                align 10h
aSystem32       db 'system32',0         ; DATA XREF: sub_151BC087+D8↑o
                align 4
aSyswow64       db 'syswow64',0         ; DATA XREF: sub_151BC087+D1↑o
                align 4
; CHAR aQS[]
aQS             db ' /q "%s"',0         ; DATA XREF: sub_151BC087+A1↑o
                align 4
; CHAR aSystem32Sdbins[]
aSystem32Sdbins db '\system32\sdbinst.exe"',0
                                        ; DATA XREF: sub_151BC087+93↑o
                                        ; sub_151BC199+68↑o
                align 4
; CHAR aWindir[]
aWindir         db '"%%windir%%',0      ; DATA XREF: sub_151BC087+81↑o
```

We get these processes executing files in order to exploit the system.

Right after these system calls we get the changes in

“\AppPatch\Custom\{f48a0c57-7c48-461c-9957-ab255ddc986e}.sdb”

used to bypass User Account Control (UAC) in malware so it get high authority privileges.

It uses a legitimate process as sdbinst.exe to execute an “.sdb” file when Boot occurs.

We can see proceeding with the functions, changes to the settings of Windows Defender and policies that are required to protect the system properly.

```
.data:151C160E aWindowsDefende db 'Windows Defender',0 ; DATA XREF: start+3FB↑o
.data:151C161F aSoftwareMicros_28 db 'SOFTWARE\Microsoft\Internet Explorer\Security',0
.data:151C164D aDisablesecurit_0 db 'DisableSecuritySettingsCheck',0
.data:151C166A aSoftwareMicros_29 db 'Software\Microsoft\Windows\CurrentVersion\Policies',0
.data:151C169D aSoftwareMicros_30 db 'Software\Microsoft\Windows\CurrentVersion\Policies\Associations',0
.data:151C16DD aLowriskfiletyp_0 db 'LowRiskFileTypes',0
.data:151C16EE aExe_1           db '.exe',0
```

Continuing forward, we get to a command “ ’htpp/shell/open/command’ , 0 ”

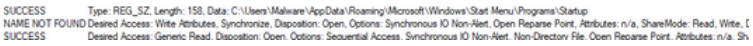
which checks for default web browser used by the user.

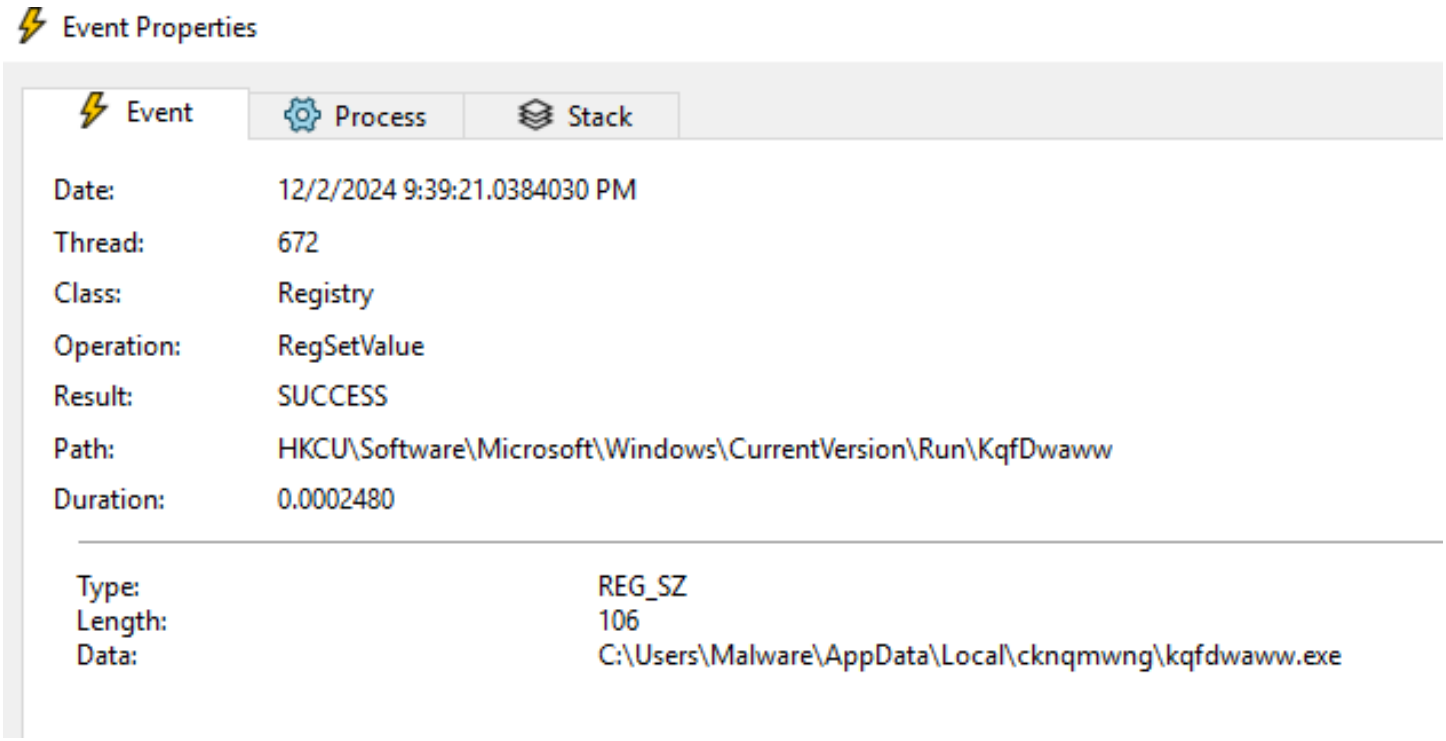
```
; CHAR aHttpShellOpenC[]
aHttpShellOpenC db 'http\shell\open\command',0
                                        ; DATA XREF: sub_151925C6+A2↑o
aChromeExe      db 'chrome.exe',0       ; DATA XREF: sub_151925C6+12B↑o
aOperaExe       db 'opera.exe',0
                align 4
aShellTraywnd   db 'Shell_TrayWnd',0    ; DATA XREF: .text:15192569↑o
aRunas          db 'runas',0            ; DATA XREF: sub_151927A3+137↑o
aCmdExe         db 'cmd.exe',0          ; DATA XREF: sub_151927A3+A5↑o
; CHAR aCSS[]
aCSS            db '/C ""%s"" %s',0     ; DATA XREF: sub_151927A3+7A↑o
; CHAR aCS[]
aCS             db '/C ""%s""',0        ; DATA XREF: sub_151927A3+5F↑o
; CHAR aSvchostExe[]
aSvchostExe     db 'svchost.exe',0      ; DATA XREF: start+22A↑o
; CHAR aUser32Dll_1[]
aUser32Dll_1    db 'user32.DLL',0       ; DATA XREF: EnumFunc+3D↑o
; CHAR aSwitchtothiswi[]
```

Last step in this malware analysis is the dynamic one where I'm going to execute the malware in my sandbox virtual machine.

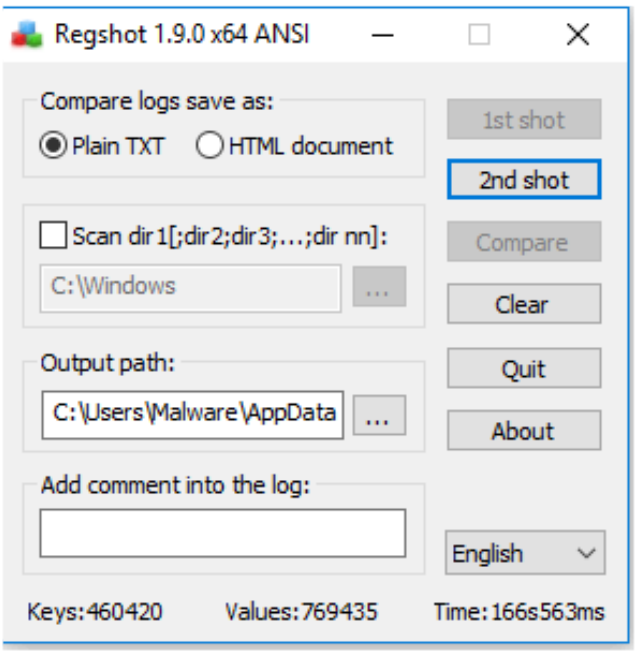
Opening Sysinternals' tool called Procmon - the tool that is used to Monitor Processes in with advanced capabilities.

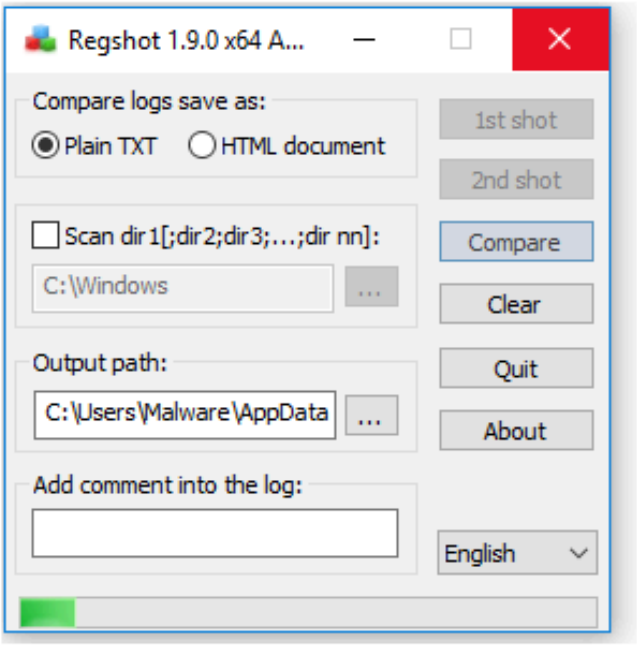
Filtering for Process name as "SampleForChallenge.exe" we can see a bunch of Write-File Functions, confirming all our investigation.





To capture the registry changes made by the malware we need to use a tool called RegShot, taking a shot before and after running the malware and then comparing between the two.





In the compare file that we have in results of Regshot scanning the changes in registries

I noticed these keys , which are being used to establish persistence in the current machine using the dropped files by the first executable.

Confirming all our findings about persistence from all previously found data.

HKU\S-1-5-21-1497642843-3941697449-2312607874-1002\Software\Microsoft\Windows\CurrentVersion\Run\KqfDwaww: "C:\Users\Malware\AppData\Local\cknqmwng\kqfdwaww.exe"

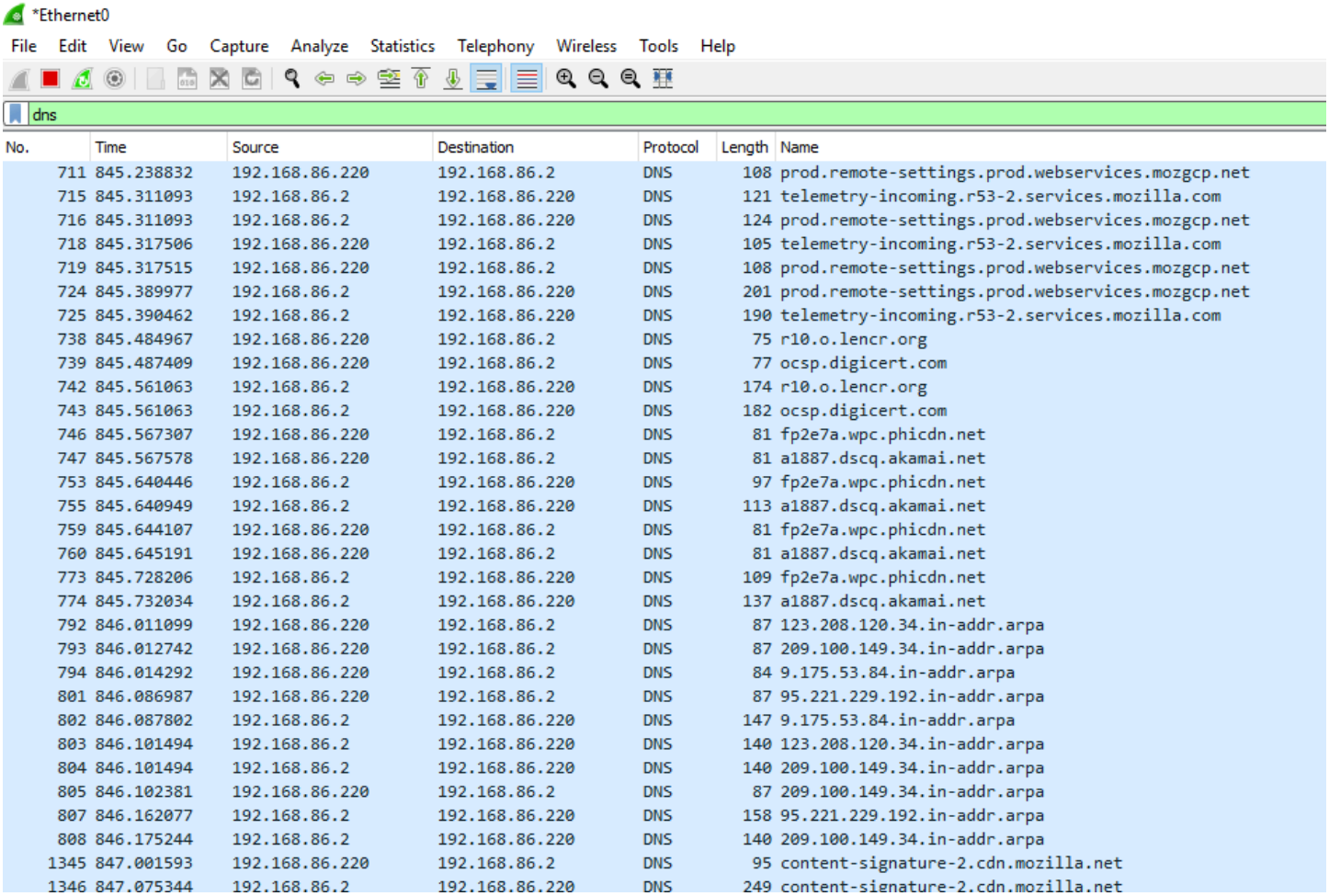
HKU\S-1-5-21-1497642843-3941697449-2312607874-1002\Software\Microsoft\Windows\CurrentVersion\Run\BseBceru: "C:\Users\Malware\AppData\Local\Temp\bsebceru.exe"

Wireshark - a network monitoring and capture tool

Filtering for “Domain Name Service” traffic that is transmitted;

We capture a nice amount of domains that are being contacted while the malware is running.

this kind of information could be helpful at understanding who the attacker is or the purpose of the attack itself.



No.	Time	Source	Destination	Protocol	Length	Name
711	845.238832	192.168.86.220	192.168.86.2	DNS	108	prod.remote-settings.prod.webservices.mozgcp.net
715	845.311093	192.168.86.2	192.168.86.220	DNS	121	telemetry-incoming.r53-2.services.mozilla.com
716	845.311093	192.168.86.2	192.168.86.220	DNS	124	prod.remote-settings.prod.webservices.mozgcp.net
718	845.317506	192.168.86.220	192.168.86.2	DNS	105	telemetry-incoming.r53-2.services.mozilla.com
719	845.317515	192.168.86.220	192.168.86.2	DNS	108	prod.remote-settings.prod.webservices.mozgcp.net
724	845.389977	192.168.86.2	192.168.86.220	DNS	201	prod.remote-settings.prod.webservices.mozgcp.net
725	845.390462	192.168.86.2	192.168.86.220	DNS	190	telemetry-incoming.r53-2.services.mozilla.com
738	845.484967	192.168.86.220	192.168.86.2	DNS	75	r10.o.lencr.org
739	845.487409	192.168.86.220	192.168.86.2	DNS	77	ocsp.digicert.com
742	845.561063	192.168.86.2	192.168.86.220	DNS	174	r10.o.lencr.org
743	845.561063	192.168.86.2	192.168.86.220	DNS	182	ocsp.digicert.com
746	845.567307	192.168.86.220	192.168.86.2	DNS	81	fp2e7a.wpc.phicdn.net
747	845.567578	192.168.86.220	192.168.86.2	DNS	81	a1887.dscq.akamai.net
753	845.640446	192.168.86.2	192.168.86.220	DNS	97	fp2e7a.wpc.phicdn.net
755	845.640949	192.168.86.2	192.168.86.220	DNS	113	a1887.dscq.akamai.net
759	845.644107	192.168.86.220	192.168.86.2	DNS	81	fp2e7a.wpc.phicdn.net
760	845.645191	192.168.86.220	192.168.86.2	DNS	81	a1887.dscq.akamai.net
773	845.728206	192.168.86.2	192.168.86.220	DNS	109	fp2e7a.wpc.phicdn.net
774	845.732034	192.168.86.2	192.168.86.220	DNS	137	a1887.dscq.akamai.net
792	846.011099	192.168.86.220	192.168.86.2	DNS	87	123.208.120.34.in-addr.arpa
793	846.012742	192.168.86.220	192.168.86.2	DNS	87	209.100.149.34.in-addr.arpa
794	846.014292	192.168.86.220	192.168.86.2	DNS	84	9.175.53.84.in-addr.arpa
801	846.086987	192.168.86.220	192.168.86.2	DNS	87	95.221.229.192.in-addr.arpa
802	846.087802	192.168.86.2	192.168.86.220	DNS	147	9.175.53.84.in-addr.arpa
803	846.101494	192.168.86.2	192.168.86.220	DNS	140	123.208.120.34.in-addr.arpa
804	846.101494	192.168.86.2	192.168.86.220	DNS	140	209.100.149.34.in-addr.arpa
805	846.102381	192.168.86.220	192.168.86.2	DNS	87	209.100.149.34.in-addr.arpa
807	846.162077	192.168.86.2	192.168.86.220	DNS	158	95.221.229.192.in-addr.arpa
808	846.175244	192.168.86.2	192.168.86.220	DNS	140	209.100.149.34.in-addr.arpa
1345	847.001593	192.168.86.220	192.168.86.2	DNS	95	content-signature-2.cdn.mozilla.net
1346	847.075344	192.168.86.2	192.168.86.220	DNS	249	content-signature-2.cdn.mozilla.net

The malware uses a svchost.exe process to communicate back to it’s C2

as a technique to evade being noticed, using a legit process.

It also changes and modifies setting in the protection resources such as Firewall and UAC and also updates won’t be done, making it harder for systems to repair themselves.

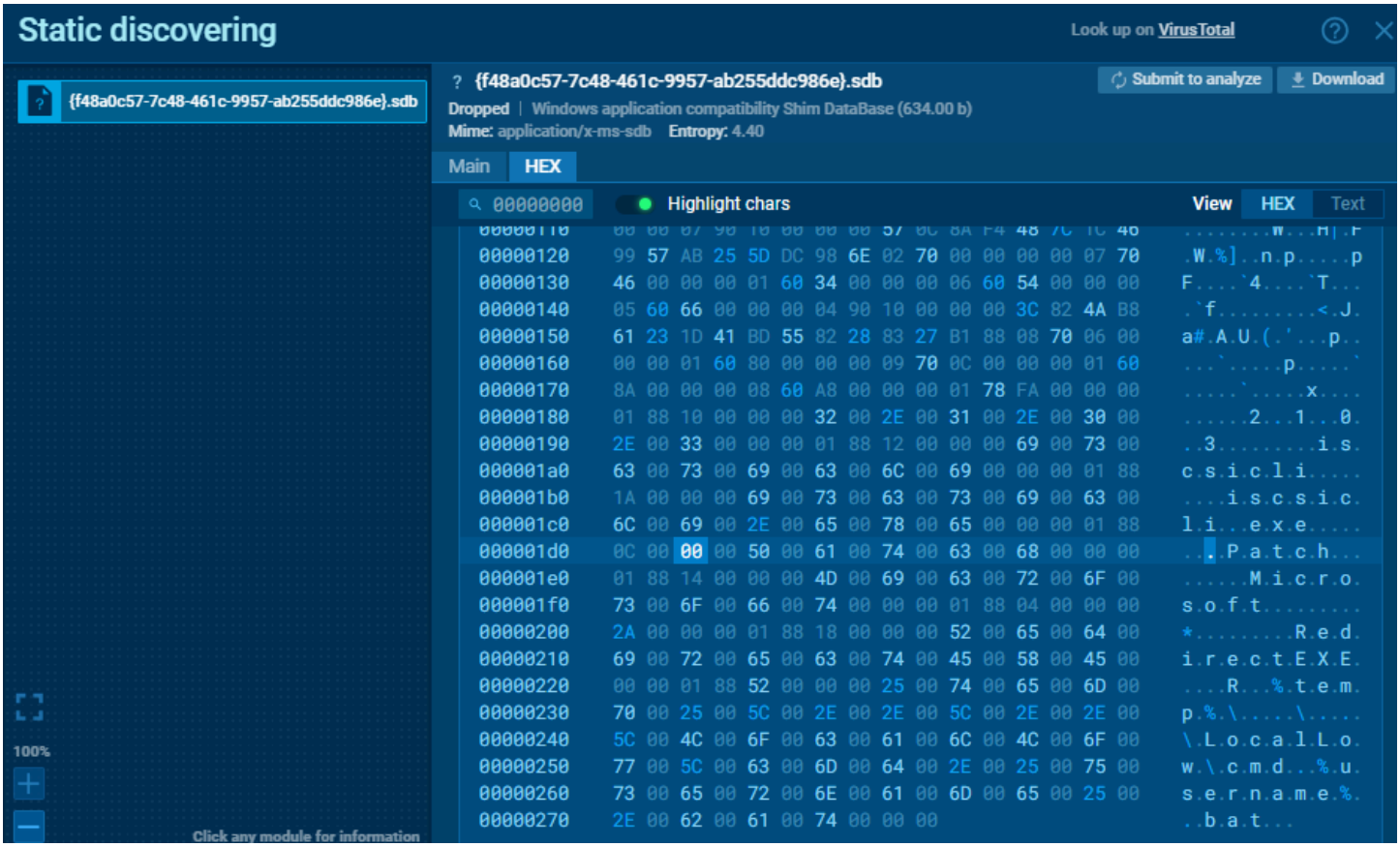
It creates itself in a startup directory and also a Run registry key that will autorun when PC will be started.

Last Step is examining the dropped files by the malware, and the changes made by it.

Examining the .sdb dropped file discussed beforehand, we can see it in the HEX view it launches a different file, which we also found that has been dropped.

it launches the malware using a legitimate process called “sdbinst.exe” which elevates the privileges and therefore bypasses UAC.

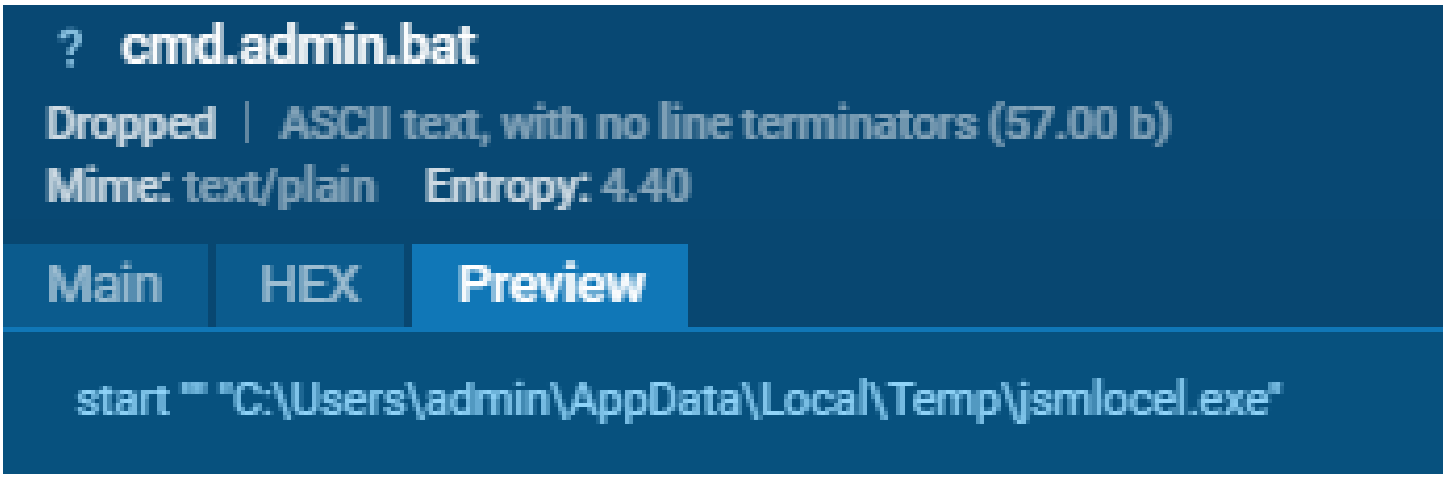
“C:\Users\admin\AppData\LocalLow\com.admin.sdb”




```
CommandLine: "C:\Windows\system32\sdbinst.exe" /q /u "C:\Users\Malware\AppData\Local\Temp\\..\..\LocalLow\com.Malware.sdb"
CurrentDirectory: C:\Windows\system32\
User: DESKTOP-OD61403\Malware
LogonGuid: {b7c2ec31-aab8-674f-cd12-040000000000}
LogonId: 0x412CD
TerminalSessionId: 1
IntegrityLevel: High
Hashes: MD5=21A1E1A8598CB71A907658D1C013B483,SHA256=8D900C613197795C9A24EDBCCB80EE184B9FE6E2A1CABA791ACA41F0DDF0C20F,IMPHA
ParentProcessGuid: {b7c2ec31-bee4-674f-3f05-000000002600}
ParentProcessId: 5492
ParentImage: C:\Users\Malware\AppData\Local\Temp\tmwiejda.exe
ParentCommandLine: C:\Users\Malware\AppData\Local\Temp\tmwiejda.exe
```

“com.admin.sdb” launches C:\Users\admin\AppData\LocalLow\cmd.admin.bat

following the lead - “cmd.admin.bat” we can see it also calls a launch of a third file called “jsmlocel.exe”



which is just the same Ramnit Virus that has been under the investigation from the beginning,

we can see it from the hashes recovered.