

Assignment 1: Basic Static and Dynamic Analysis of david and bob Malware Samples

Thomas Green

School of Engineering and Computer Science, Victoria University of Wellington

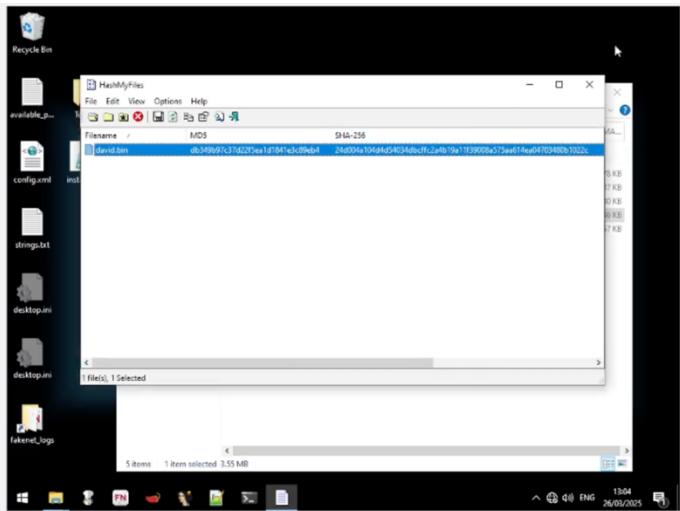
1. Analysis Setup

Analysis environment was set up using VirtualBox on an ECS lab machine with a FlareVM virtual machine for malware execution and a REMnux virtual machine configured with FakeDNS for DNS resolution. Network configuration consisted of NAT and host-only networking to isolate malware behaviour while enabling limited internet simulation.

2. David

2.1. Basic Static Analysis

VirusTotal



Began investigation acquiring MD5 hash of file to search within VirusTotal.

71 out of 73 vendors identified the malware as malicious. Identified by antivirus engines under names such as Trojan.Ransom.WannaCryptor.H and Ransom:Win32/WannaCry.398. I can determine that david.bin is related to WannaCry ransomware, which encrypts files, demands Bitcoin ransom, and has worm-spreading behaviour. Did not manually compare AV signatures with ClamAV, however, VirusTotal includes ClamAV engine, which confirms detection – Win.Ransomware.Wanna-9769986-0.

The screenshot shows the VirusTotal analysis interface. At the top, there's a search bar and various analysis filters like 'pefile', 'check-network-adapters', etc. Below the search bar, the file hash is listed as J4D9KA4L2U4D4D2-KU4D0C9C2B4D19A11F39002a575aaef1aead4703480b1022c (detected). The file size is 3.55 MB and it was last analyzed 4 hours ago. The file type is EXE. The 'Basic properties' section contains detailed technical information including MD5, SHA-1, SHA-256, VHash, AntivirusHash, ImpHash, Rich PE header hash, SSDeep, File type (PE32), and various compiler details (Microsoft Visual C++). The 'History' section shows the creation time as 2010-11-20 09:03:08 UTC.

The malware compiler is Microsoft Visual C++. Can identify the MD5 and SHA-256 hashes that match the .bin malware. We can observe that malware is PE32 architecture, meaning it's not immediately obfuscated. Cyren packer indicates a .rsrc packer, signifying that resource packing is used potentially to hide payloads or avoid detection.

History	
Creation Time	2010-11-20 09:03:08 UTC
First Seen In The Wild	2013-05-04 10:00:45 UTC
First Submission	2017-05-12 08:57:51 UTC
Last Submission	2025-03-26 17:54:07 UTC
Last Analysis	2025-03-26 20:22:00 UTC

The creation time was 2010-11-20 - when the binary was first compiled.

File Version Information	
Copyright	© Microsoft Corporation. All rights reserved.
Product	Microsoft® Windows® Operating System
Description	Microsoft® Disk Defragmenter
Original Name	lhdfrgui.exe
Internal Name	lhdfrgui.exe
File Version	6.1.7601.17514 (win7sp1_rtm.101119-1850)

Malware does some sort of file version spoofing to appear legitimate. The original/internal name was lhdfgui.exe. Malware is impersonating a legitimate Windows utility – Windows Disk Defragmenter. It uses this technique to avoid suspicion from casual observers.

Header						
Target Machine		Intel 386 or later processors and compatible processors				
Compilation Timestamp		2010-11-20 09:03:08 UTC				
Entry Point		39446				
Contained Sections		4				
Sections						
Name	Virtual Address	Virtual Size	Raw Size	Entropy	MD5	Chi2
.text	4096	35786	36864	6.13	c7613102e2ecec5dcef144f83189153	514235.06
.rdata	40960	2456	4096	3.5	d8037d744b539326c06e897625751cc9	359720.38
.data	45056	3164316	159744	6.1	22a0590dc29cad7078c291e94612ce26	2564680.25
.rsrc	3211264	3515476	3518464	8	12e1bd7375d82cca3a51ca48fe22dia9	32632.63

PE header indicates the malware targets Intel386 or later processes. Has four contained sections: .text, .rdata, .data and .rsrc. Indicates 6.13, 3.5, and 6.1 entropy for .text, .rdata and .data respectively. This is relatively low and doesn't indicate packing. I can see .rsrc has high entropy of 8, indicating a likely use of obfuscation or packing. The MD5 hashes of each section can be observed, which is helpful for threat hunting, signature matching, and detection.

Imports	
+	KERNEL32.dll
+	ADVAPI32.dll
+	WS2_32.dll
+	MSVCP60.dll
+	iphlpapi.dll
+	WININET.dll
+	MSVCRT.dll

Can see the DLLs that the malware uses. Malware to be aware of while analyzing are WS2_32.dll and WININET.dll library, which is the Windows socket API library which implies the use of internet functions such as HTTP requests. Also, iphlpapi.dll allows the program to access information about the system IP configuration and routing table.

The screenshot shows the VirusTotal interface with two tabs open: 'VirusTotal - File - a23ef053ccf6a35fda9ad5f1702ba99a7be695107d3ba5d1ea8c9c258299e4' and 'VirusTotal - File - 24d004a10'. The main content area displays the 'Zenbox' report. Key findings include:

- Detections:** 3 detections, including 1 Malware, 1 Trojan, and 1 Evader.
- Mitre Signatures:** 3 HIGH, 40 LOW, 62 INFO.
- IDS Rules:** NOT FOUND.
- Sigma Rules:** NOT FOUND.
- Dropped Files:** 1 CAB, 1 TEXT, 1 PE_EXE.
- Network comms:** 3 DNS, 125 IP.

Behavior Tags: `calls_api`, `check_disk`, `check_disk_space`, `checks_memory_address`, `checks_user_input`, `detect_debug_breakpoint`, `idle`, `long_sleeps`, `persistence`, `service_com`.

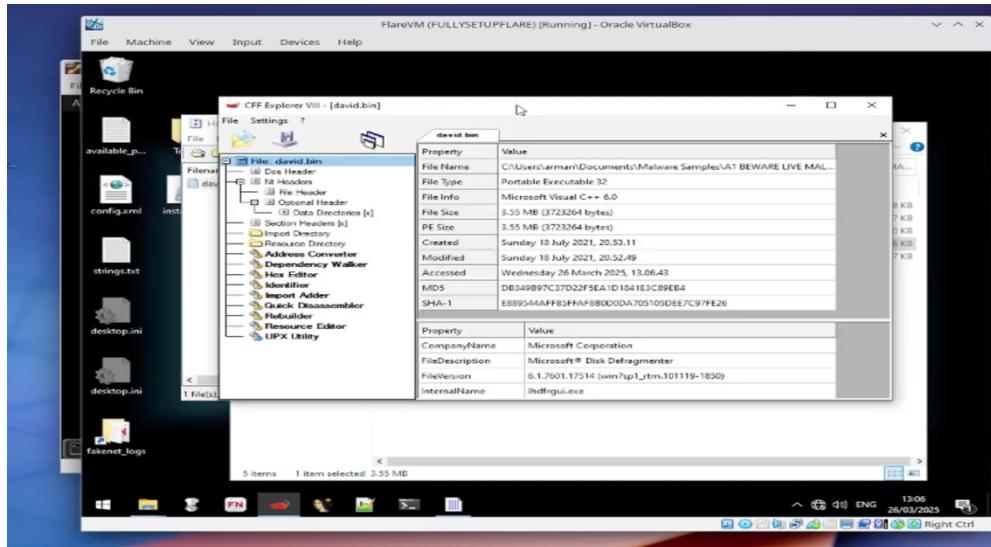
Dynamic Analysis Sandbox Detections: The sandbox Zenbox flagged this file as: MALWARE TROJAN EVADER.

MITRE ATT&CK Tactics and Techniques:

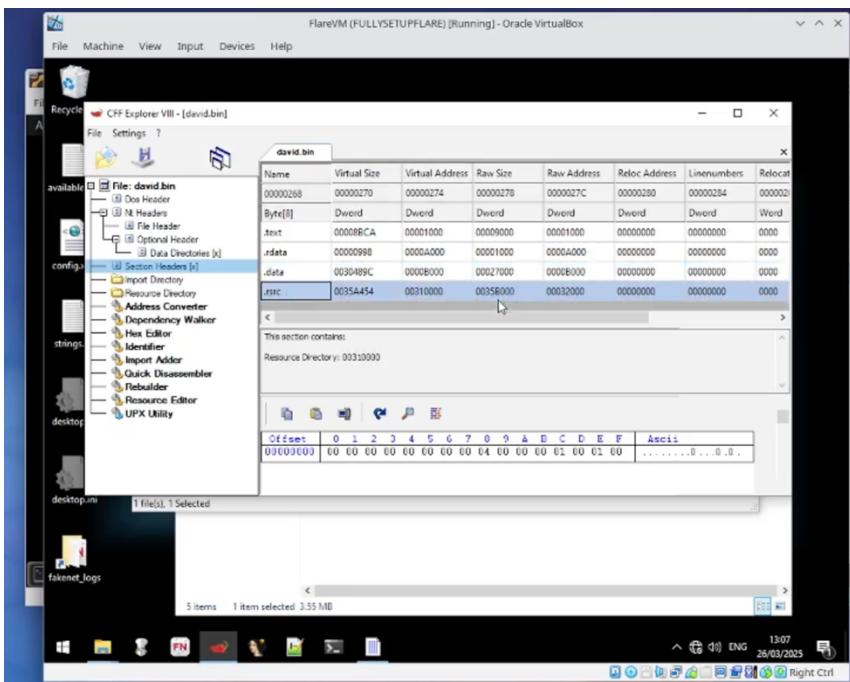
- Execution:**
 - Windows Management Instrumentation (T1047)
 - Creates processes via WMI
 - Creates BIOS Information (via WMI, Win32_Bios)
 - Queries sensitive disk information (via WMI, Win32_DiskDrive, often done to detect virtual machines)
 - Creates memory information (via WMI, Win32_Process)
 - Creates process information (via WMI, Win32_Process)
 - Creates sensitive Operating System Information (via WMI, Win32_ComputerSystem, often done to detect virtual machines)
 - Checks if Antivirus program is installed (via WMI)
 - Creates sensitive processor information (via WMI, Win32_Processor, often done to detect virtual machines)
- Scheduled Task/Job:**
 - Uses schtasks.exe or at.exe to add and modify task schedules
 - Creates COM task schedule object (often to register a task for autostart)
- Command and Scripting Interpreter:**
 - Obfuscated command line found
 - Uses cmd line tools excessively to alter registry or file data
 - Very long cmdline option found, this is very uncommon (may be encrypted or packed)
- Persistence:**
 - Scheduled Task/Job
 - Uses schtasks.exe or at.exe to add and modify task schedules
 - Creates COM task schedule object (often to register a task for autostart)

Analyzed the Sandbox report done by Zenbox on VirusTotal. Execution uses command tools extensively for registry/file manipulation and scheduled tasks. For persistence, it uses registry run keys and startup folder to auto-start the malware upon reboot.

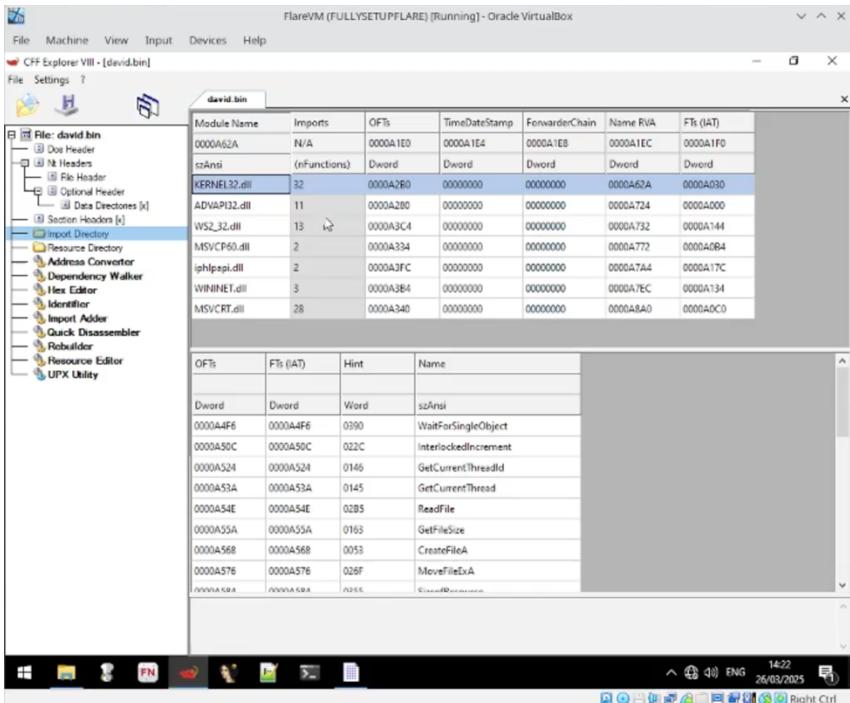
CFF Explorer



Confirms the malware is 32-bit PE and is compiled using Microsoft Visual C++. It also shows the internal name `lhdfgui.exe` and is trying to pass off as a Microsoft Disk Defragmenter. You can also see creation and modification date, which doesn't match the timestamps we acknowledged earlier in VirusTotal.

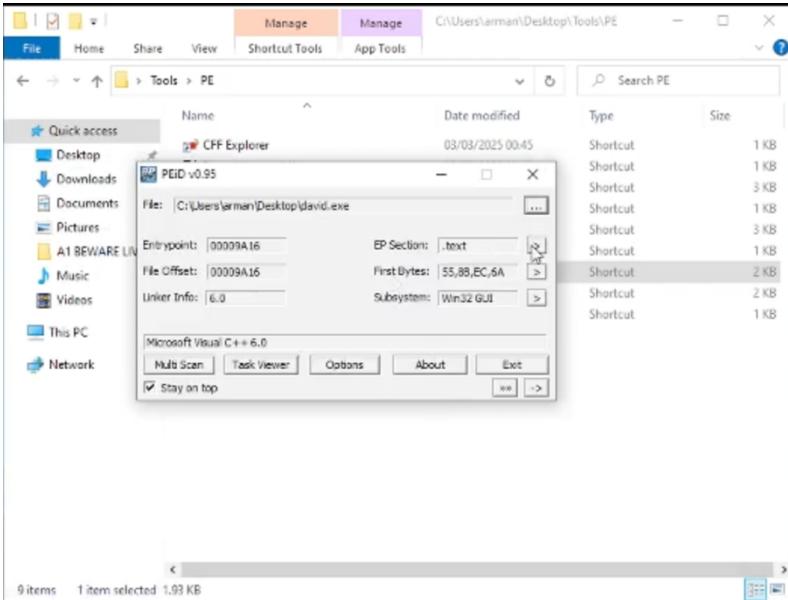


Analyzed PE Section headers. Clear that the malware is not packed with tools like UPX (e.g. UPX0, UPX1). .data section appears larger than the others, which may indicate embedded payloads or buffers.

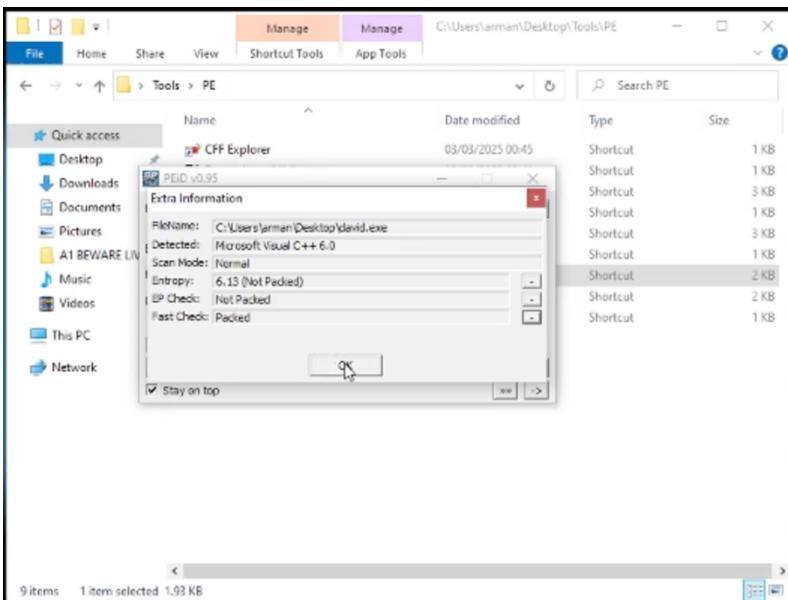


We can see DLLs identified earlier in VirusTotal. Shows that the malware is capable of performing system-level operations such as registry access, networking and file handling. The malware can use functions such as CreateFileA, ReadFile, and MoveFileExA from KERNEL32.dll.

PEiD

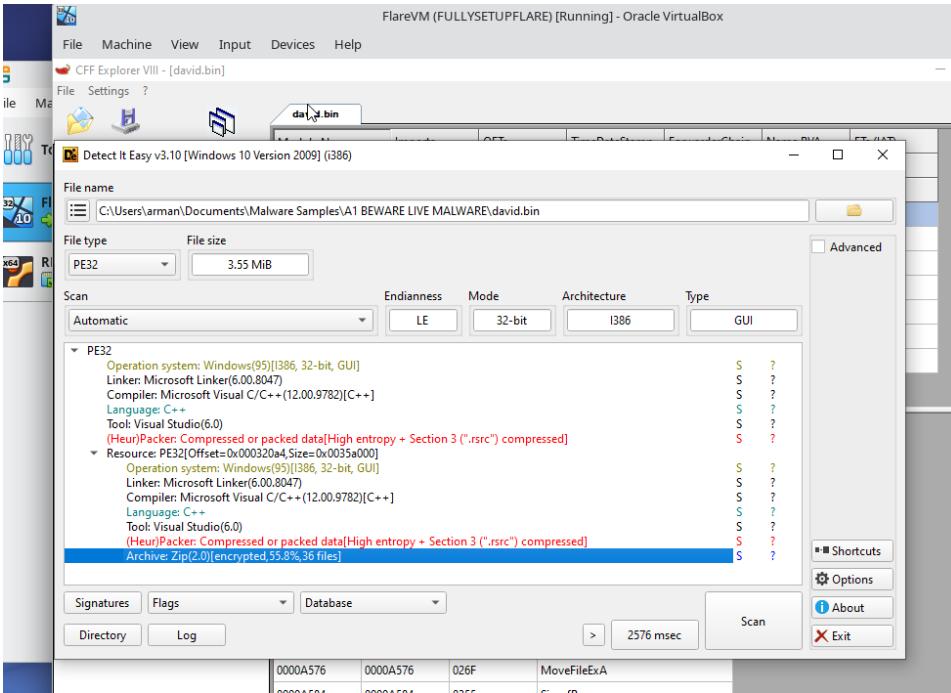


Identifies the entry point of file, which lies in the .text section. Identified malware was compiled using Microsoft Visual C++.



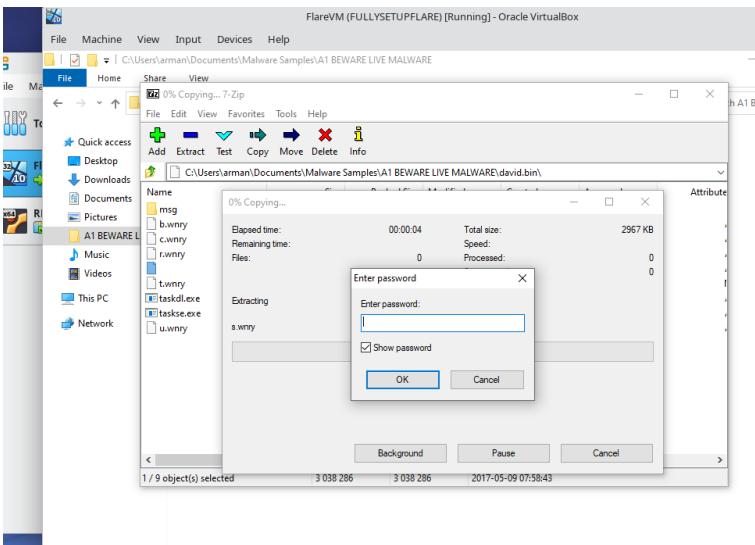
Malware has an entropy of 6.13, which generally falls within the range of unpacked/lightly obfuscated binaries. EP Check also reported not packed. Fast Check says packed, but this sometimes produces false positive results. Cyren Packer, an antivirus engine that detects characteristics of packed/obfuscated files, indicated that .rsrc was packed.

Detect it Easy



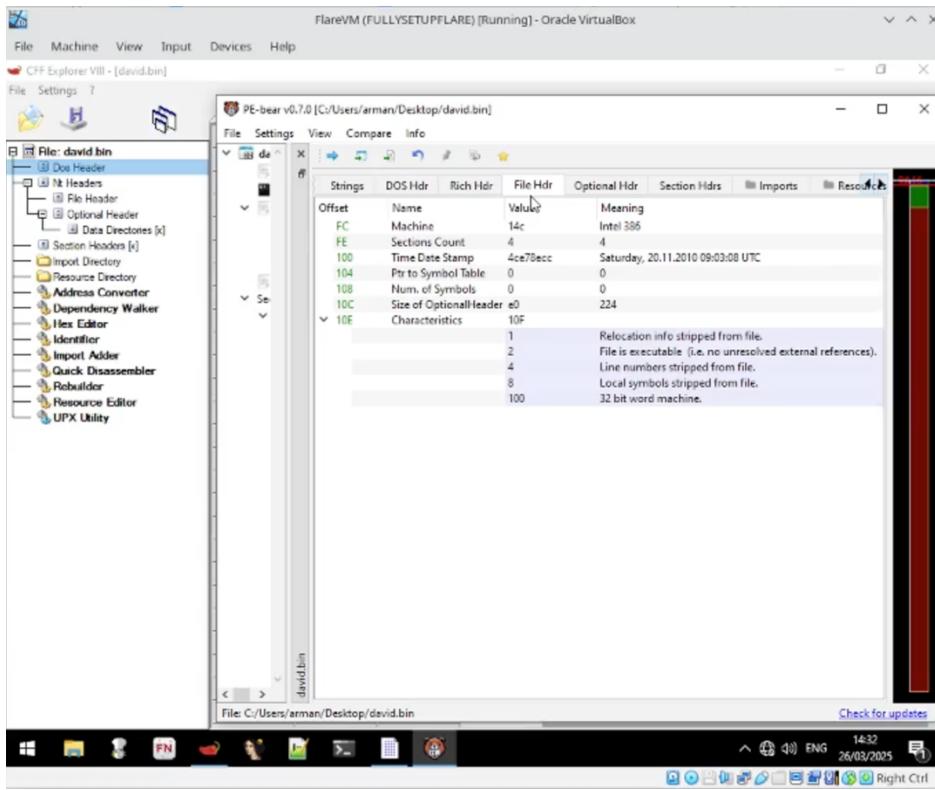
Section 3 (.rsrc) is both compressed and has high entropy, which suggests that it's compressed or packed data. Additionally identified, the .rsrc section contains a ZIP archive with 36 encrypted files. This may become unpacked/executed at runtime.

7-Zip

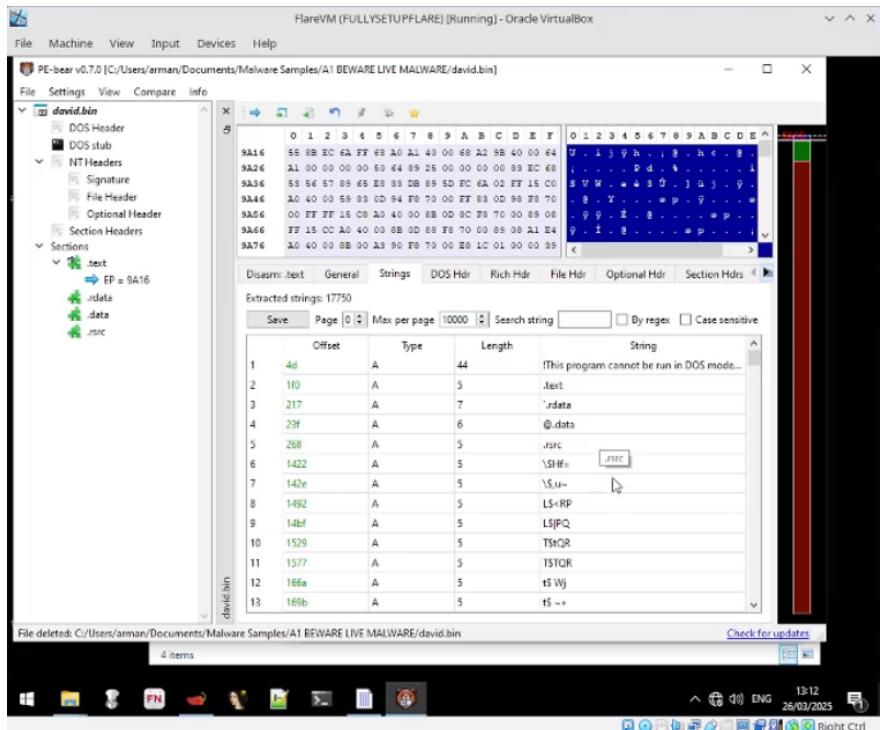


I tried to use 7-zip tool to extract the .rsrc zip contents; however, the archives were password-protected.

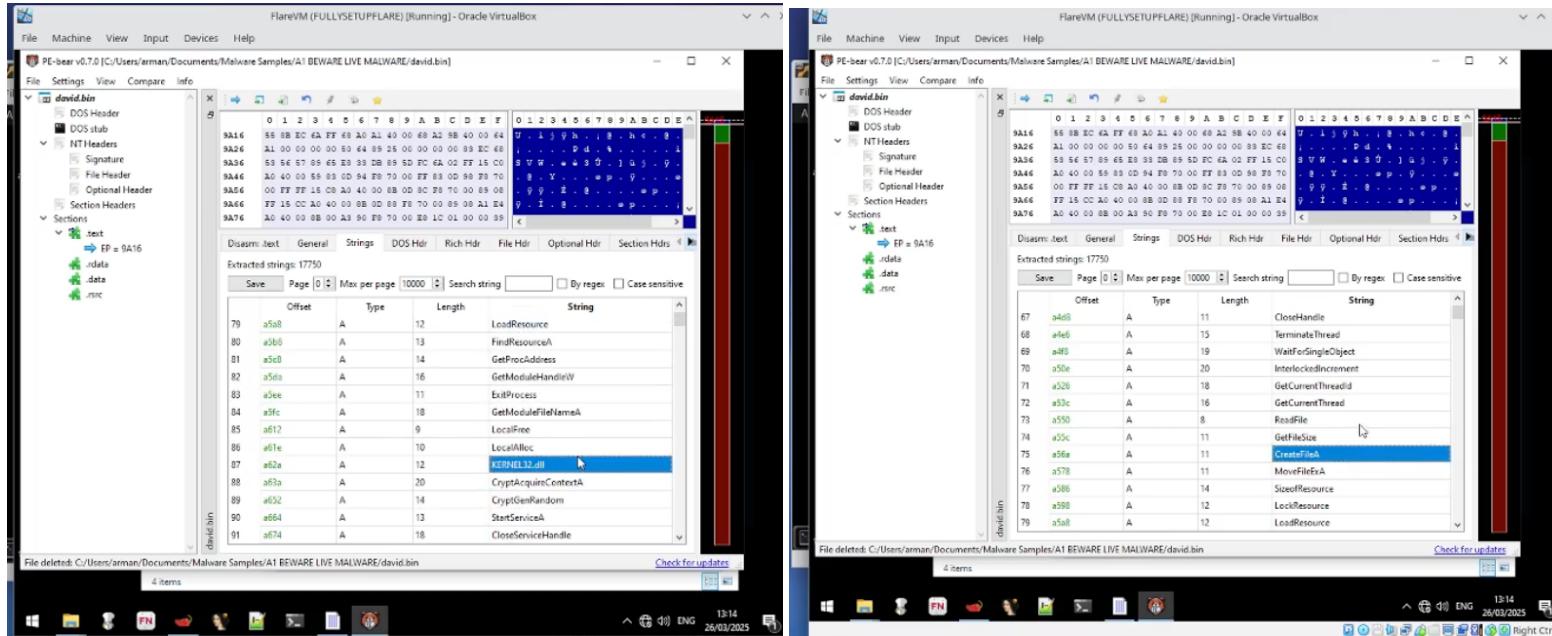
PE-Bear



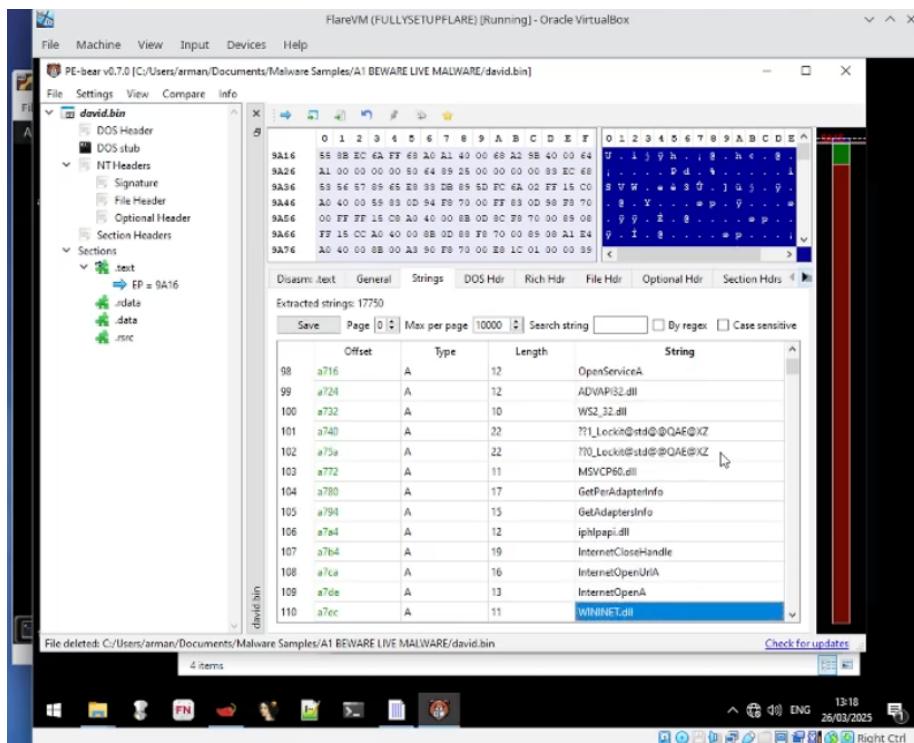
File header confirmed compilation date of david.bin. Characteristics revealed the file is executable, and several components have been stripped, such as relocation info, local symbols, and line numbers. Confirmed compilation date 20.11.2010, as seen earlier in VirusTotal.



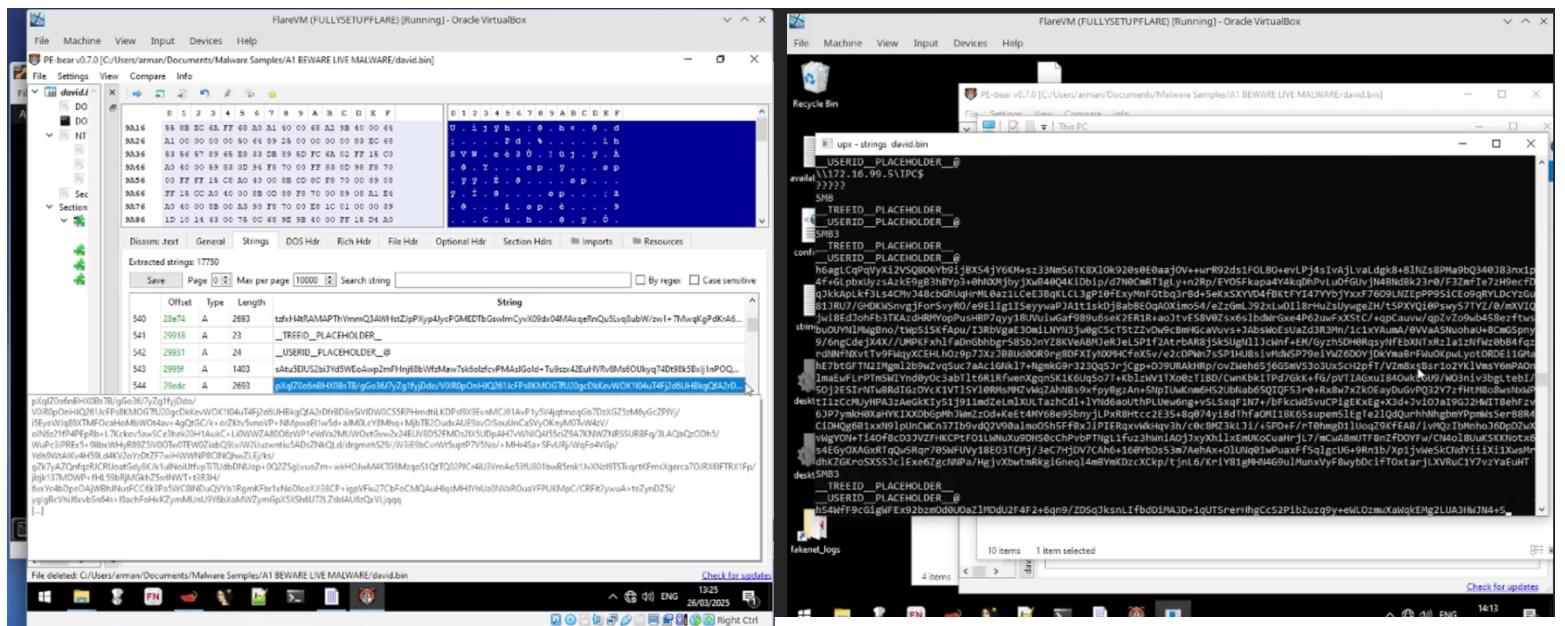
Identified 17750 extracted strings. See initial PE Section references for .text, .rdata, .data and .rsrc. You can also see a bunch of encoded or obfuscated strings at the start as well.



Can identify functions ReadFile, GetFileSize, CreateFileA, and MoveFileExA indicating interaction with files, most likely to read, modify or move data. Presence of KERNEL32.dll reinforces this, as it's commonly used for system operations like file and memory management. Additionally, CryptAcquireContextA and CryptGenRandom, which are part of advapi32.dll, shows the malware is performing encryption.



Can see network-based indicators; GetPerAdapterInfo, GetAdaptersInfo, InternetOpenA, InternetOpenUrlA, and InternetCloseHandle. Used to retrieve network configuration details and establish HTTP connections.



Here, we can see what appears to be payloads. Use of placeholders suggests they are used for injecting user or system values. Strings were too large for PE-bear to display so printed strings on UPX terminal. Can observe SMB3 at the end of the payload. SMB is used for sharing information over a network connection. Suggests the malware uses SMB-based exploits or features to move laterally within a network and deliver payloads to other vulnerable machines.

PE-bear v0.7.0 [C:/Users/arman/Documents/Malware Samples/A1 BEWARE LIVE MALWARE/david.bin]

File Settings View Compare Info

Hex Dump View

Offset	Type	Length	String
688	A	20	_TREEPATH_REPLACE_
689	A	9	\%s\PCS
690	A	42	Microsoft Base Cryptographic Provider v1.0
691	A	11	%d.%d.%d.%d
692	A	11	mssecsvc2.0
693	A	39	Microsoft Security Center (2.0) Service
694	A	14	%s -m security
695	A	16	C:\\$1periuwjhfr
696	A	8	C:\\$1%
697	A	7	WINDOWS
698	A	12	task sche.exe

Extracted strings: 17750

Save Page Max page 10000 Search string By regex Case sensitive

PE-bear v0.7.0 [C:/Users/arman/Documents/Malware Samples/A1 BEWARE LIVE MALWARE/david.bin]

File Settings View Compare Info

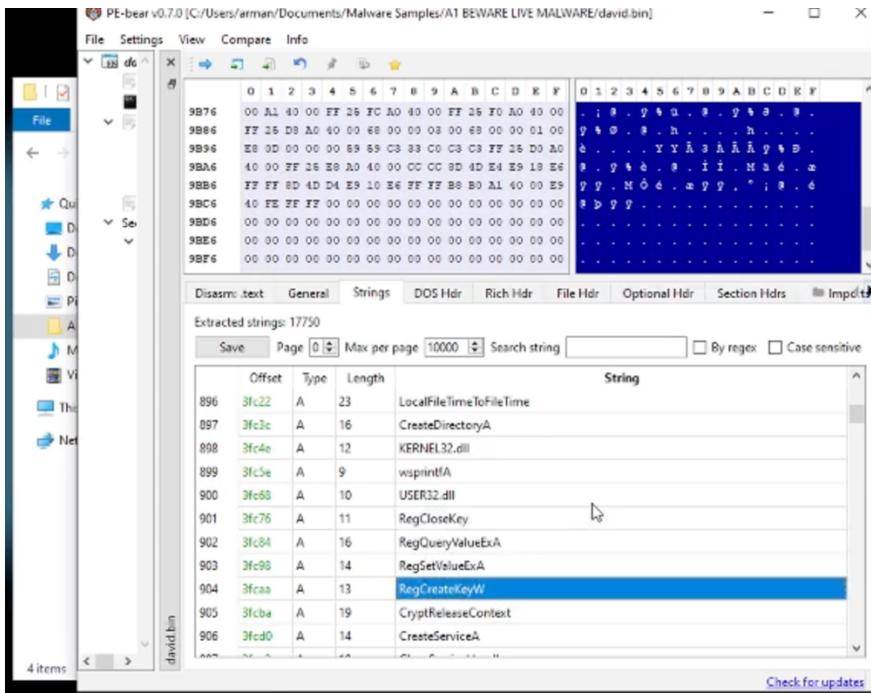
Hex Dump View

Offset	Type	Length	String
697	A	7	WINDOWS
698	A	12	task sche.exe
699	A	11	CloseHandle
700	A	9	WriteFile
701	A	11	CreateFileA
702	A	14	CreateProcessA
703	W	12	kernel32.dll
704	A	56	http://www.iuqerfsodp8ifjaposdfjhgosurijfaewrxergwea.com
705	A	44	(This program cannot be run in DOS mode. S
706	A	5	.text
707	A	7	ndata

Extracted strings: 17750

Save Page Max page 10000 Search string By regex Case sensitive

Can see a HTTP domain, which is a kill switch. When malware is executed, it attempts to request an HTTP to this URL. If connection is successful, it will drop/terminate execution, and if it fails, dropper will infect the system with ransomware. Malware is designed to run as a service with the parameters -m security.



Can see registry-related functions; RegCloseKey, RegQueryValueExA and RegCreateKeyW. Functions can enable malware to maintain persistency at startup through registry manipulation by creating/modifying keys so malware runs after system reboot.

Can observe hardcoded Bitcoin wallet addresses. Can use the addresses to look up the actual Bitcoin wallets on www.blockchain.com.

PE-bear v0.7.0 [C:/Users/arman/Documents/Malware Samples/A1 BEWARE LIVE MALWARE/david.bin]

File Settings View Compare Info

File

Extracted strings: 17750

	Offset	Type	Length	String
17704	38b852	W	8	040904B0
17705	38b06a	W	11	CompanyName
17706	38b884	W	21	Microsoft Corporation
17707	38b8b6	W	15	FileDescription
17708	38b8d8	W	8	DiskPart
17709	38b8f2	W	11	FileVersion
17710	38b90c	W	40	6.1.7601.17514 (win7sp1_itm.101119-1850)
17711	38b966	W	12	InternalName
17712	38b980	W	12	diskpart.exe
17713	38b9a2	W	14	LegalCopyright
17714	38b9c4	W	43	Microsoft Corporation. All rights reserved.

Shows strings DiskPart and diskpart.exe. Is used by the malware to disguise itself as a legitimate Windows System utility by adopting the internal name diskpart.exe. Allows it to evade detection and appear as a typical system process.

2.2. Basic Dynamic Analysis

Setup and Running Malware

FlareVM (FULLYSETUPFLARE) [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Process Monitor - Sysinternals: www.sysinternals.com

File Edit Event Filter Tools Options Help

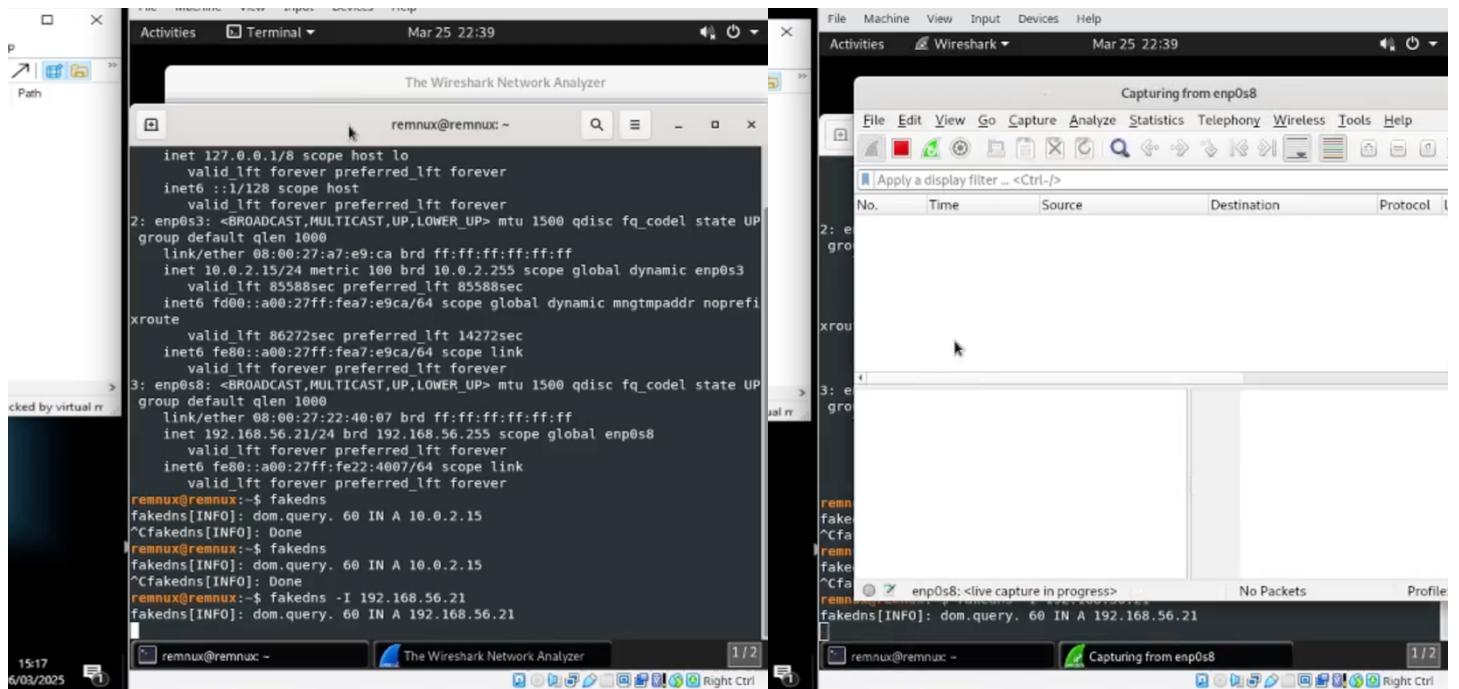
No events (capture disabled) Backed by virtual memory

Process Explorer - Sysinternals: www.sysinternals.com [CYBR473arman] (Administrator)

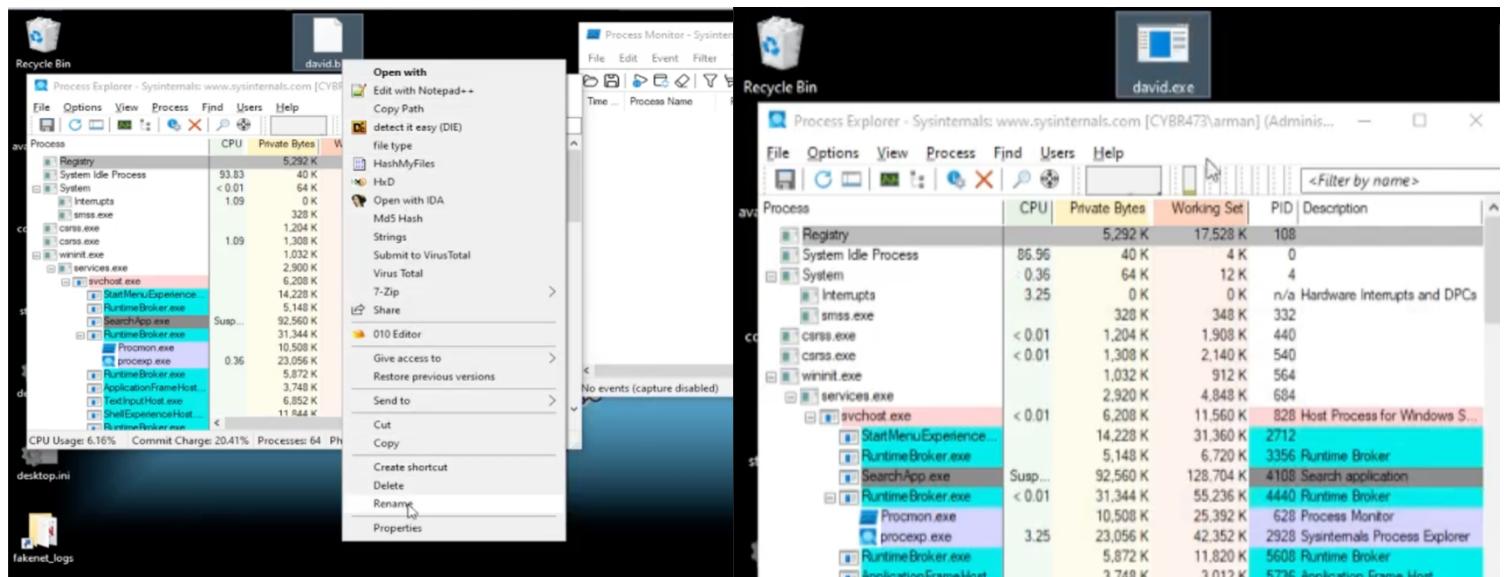
File Options View Process Find Users Help

Process	CPU	Private Bytes	Working Set	PID	Description	Company Name
Registry	4,036 K	16,536 K	108			
System	63.16	40 K	4 K	0		
System	0.49	64 K	12 K	4		
Interrupt	0.98	0 K	0 K	n/a	Hardware Interrupts and DPCs	
msasn1.exe		328 K	348 K	332		
coss.exe	< 0.01	1,236 K	1,336 K	440		
corse.exe	< 0.01	1,340 K	2,140 K	540		
wminit.exe		1,032 K	912 K	564		
services.exe	0.49	2,900 K	4,704 K	604		
evchost.exe	0.24	6,472 K	11,644 K	828	Host Process for Windows S... Microsoft Corporation	
evchost.exe		14,652 K	31,490 K	2712		
StartMenuExperience		5,244 K	7,049 K	3356	Runtime Broker Microsoft Corporation	
RuntimeBroker.exe		53,400 K	53,768 K	4440	Runtime Broker Microsoft Corporation	
SearchApp.exe	Susp...	91,264 K	126,524 K	4108	Search application Microsoft Corporation	
RuntimeBroker.exe	< 0.01	31,768 K	53,400 K	4440	Runtime Broker Microsoft Corporation	
Procmon.exe	< 0.01	10,880 K	25,384 K	628	Process Monitor Sysinternals - www.sysinter...	
RuntimeBroker.exe		16,164 K	34,740 K	3996	Sytematic Process Explorer Sysinternals - www.sysinter...	
RuntimeBroker.exe		6,532 K	13,148 K	5609	Runtime Broker Microsoft Corporation	
ApplicationFrameHost...	< 0.01	3,749 K	3,012 K	5736	Application Frame Host Microsoft Corporation	
TestInputHost.exe		11,844 K	30,952 K	1636	Windows Shell Experience H... Microsoft Corporation	
ShellExperienceHost...		5,736 K	14,336 K	1404	Runtime Broker Microsoft Corporation	
RuntimeBroker.exe		4,656 K	6,764 K	5856	COM Surrogate Microsoft Corporation	
dihost.exe		2,012 K	8,076 K	5940	WMI Provider Host Microsoft Corporation	
MoUsrComWorker.exe	< 0.01	3,584 K	15,076 K	2848	MoISO Core Worker Process Microsoft Corporation	
WmiFrvSE.exe		2,012 K	8,076 K	5940	WMI Provider Host Microsoft Corporation	
evchost.exe	0.24	5,972 K	7,840 K	952	Host Process for Windows S... Microsoft Corporation	
svchost.exe	8.81	28,100 K	41,116 K	1164	Host Process for Windows S... Microsoft Corporation	
shot.exe	< 0.01	4,344 K	14,264 K	3400	Shell Infrastructure Host Microsoft Corporation	
taskhostw.exe		5,388 K	10,312 K	3496	Host Process for Windows T... Microsoft Corporation	
MicrosoftEdgeUpdate...	< 0.01	1,736 K	3,540 K	3532	Microsoft Edge Update Microsoft Corporation	

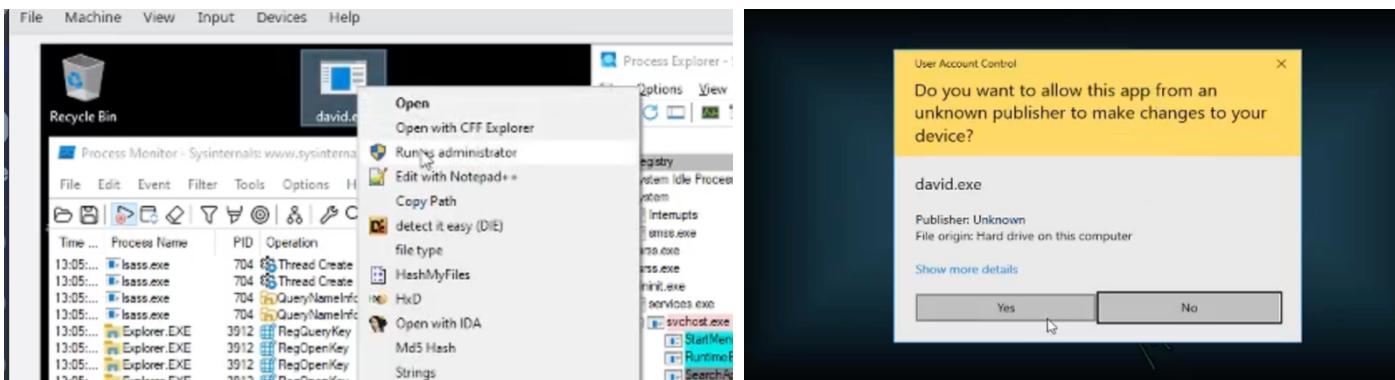
First launched Process Monitor and Process Explorer to monitor system processes in real-time.



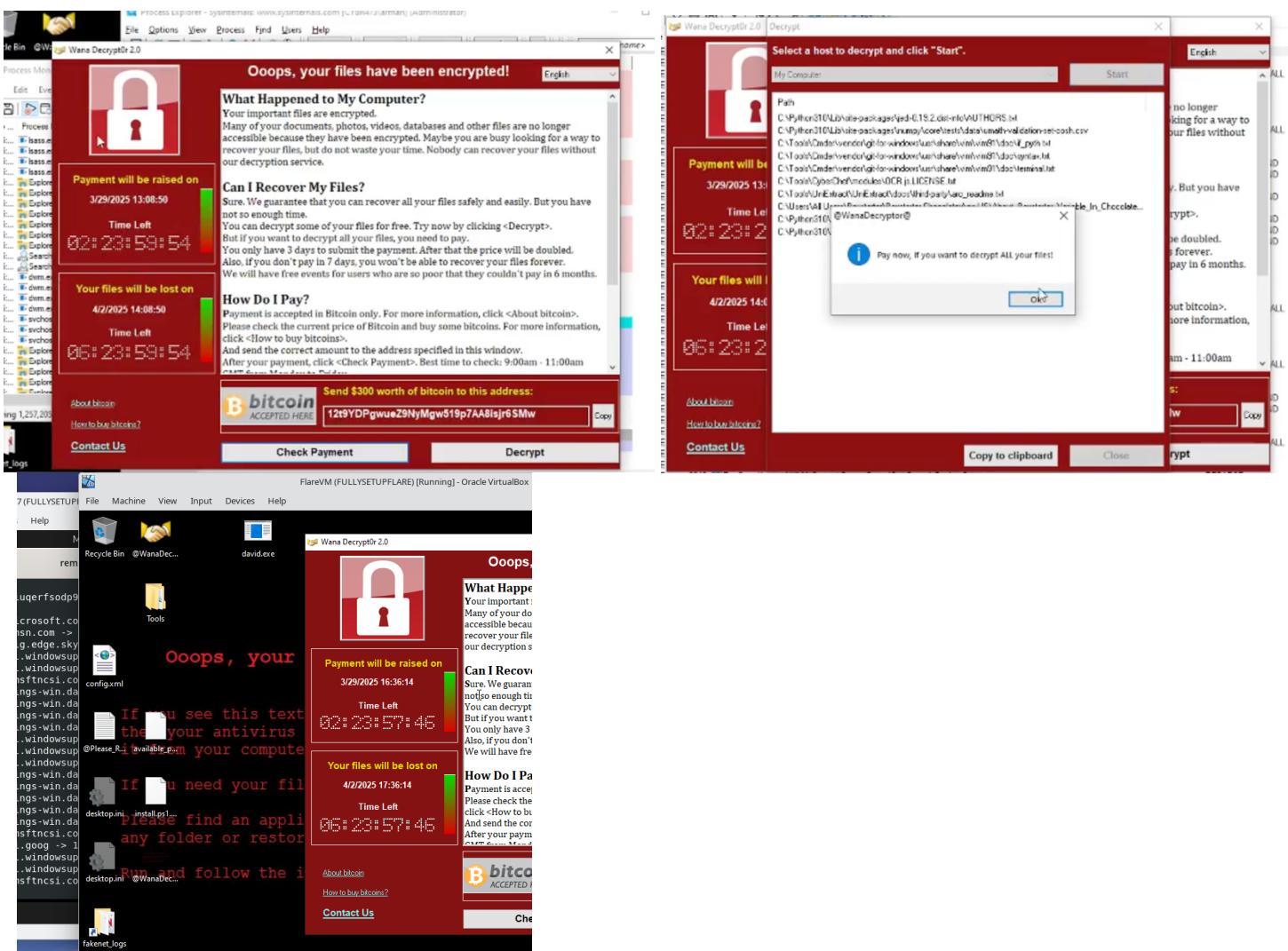
Setup a FakeDNS server to respond to DNS requests and Wireshark to listen to network traffic on enp0s8.



Changed david.bin to an executable.



Executed the malware with administrative privileges.



After running, we are greeted with a ransom note indicating files are encrypted and a Bitcoin wallet address to pay. Has a decrypt button allowing partial file decryption as proof, but full decryption requires payment. Ransom-related files are on the desktop, and the desktop background has changed. This is an indicator malware has finished running.

Process Explorer

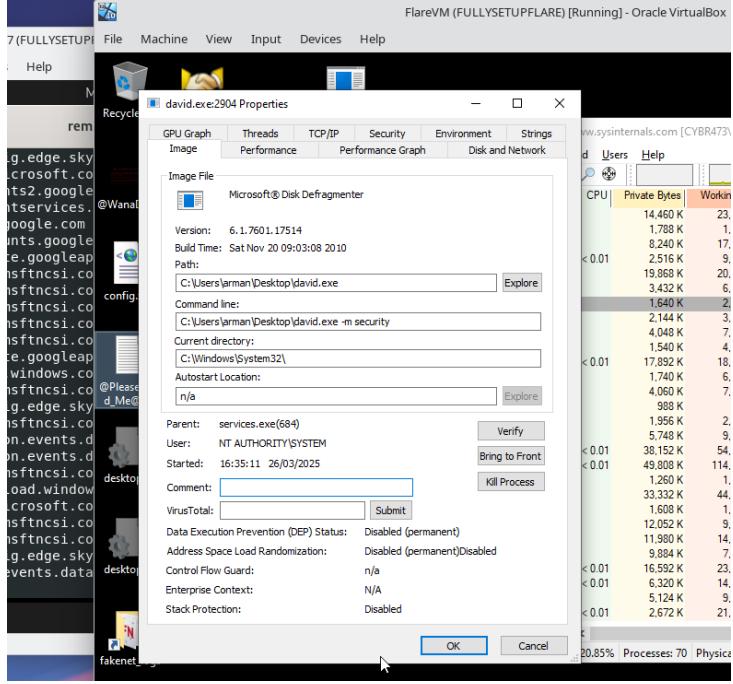
Screenshot 1: A screenshot of Process Explorer showing a complex process tree. Processes include svchost.exe, SearchIndexer.exe, SearchProtocolHost.exe, SecurityHealthService.exe, TrustedInstaller.exe, daniel.exe, tasksche.exe, lass.exe, ffontdrvhost.exe, winlogon.exe, dwm.exe, explorer.exe, SecurityHealthSysTray.exe, david.exe, msedge.exe, mmedge.exe, taskhsvc.exe, and conhost.exe. A red '1' is drawn over the window title.

Screenshot 2: A screenshot of Process Explorer showing a detailed view of the task switcher interface. It lists various Microsoft processes such as svchost.exe, SearchIndexer.exe, SecurityHealthService.exe, daniel.exe, tasksche.exe, lass.exe, ffontdrvhost.exe, winlogon.exe, dwm.exe, explorer.exe, SecurityHealthSysTray.exe, msedge.exe, mmedge.exe, taskhsvc.exe, and conhost.exe. A red '2' is drawn over the window title.

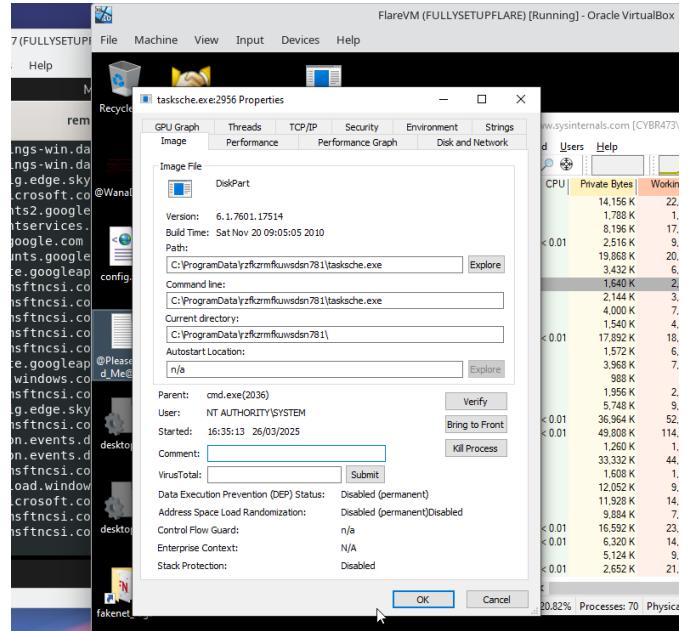
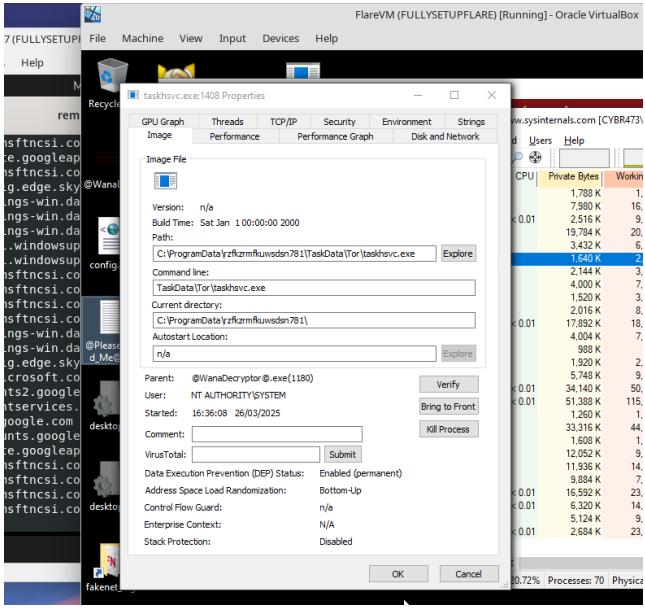
Screenshot 3: A screenshot of Process Explorer showing a detailed view of the task switcher interface. It lists various Microsoft processes such as svchost.exe, SearchIndexer.exe, SecurityHealthService.exe, daniel.exe, tasksche.exe, lass.exe, ffontdrvhost.exe, winlogon.exe, dwm.exe, explorer.exe, SecurityHealthSysTray.exe, msedge.exe, mmedge.exe, taskhsvc.exe, and conhost.exe. A red '3' is drawn over the window title.

Screenshot 4: A screenshot of Process Explorer showing a detailed view of the task switcher interface. It lists various Microsoft processes such as svchost.exe, SearchIndexer.exe, SecurityHealthService.exe, daniel.exe, tasksche.exe, lass.exe, ffontdrvhost.exe, winlogon.exe, dwm.exe, explorer.exe, SecurityHealthSysTray.exe, msedge.exe, mmedge.exe, taskhsvc.exe, and conhost.exe. A red '4' is drawn over the window title.

During malware run, tasksche.exe is spawned as a child process of david.exe. Shortly after, taskdl.exe appears as a child process of tasksche.exe but disappears within a few seconds. See the creation of @WanaDecryptor@.exe, which is the ransomware interface, along with taskhsvc.exe and child process conhost.exe.

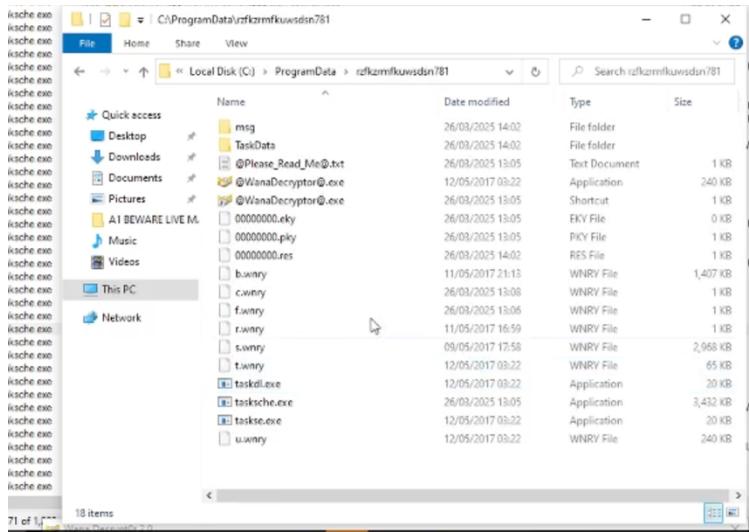


Working Set	PID	Description	Company Name
8,536 K	1884	Host Process for Windows S...	Microsoft Corporation
9,464 K	1896	Host Process for Windows S...	Microsoft Corporation
14,040 K	484	Host Process for Windows S...	Microsoft Corporation
13,248 K	488	Spooler SubSystem App	Microsoft Corporation
12,892 K	500		
12,556 K	1352	Host Process for Windows S...	Microsoft Corporation
6,300 K	2145	Host Process for Windows S...	Microsoft Corporation
29,932 K	2256	Host Process for Windows S...	Microsoft Corporation
7,452 K	3032	Host Process for Windows S...	Microsoft Corporation
33,048 K	3436	Host Process for Windows S...	Microsoft Corporation
14,608 K	1128	Host Process for Windows S...	Microsoft Corporation
30,988 K	812	Microsoft Windows Search ...	Microsoft Corporation
9,836 K	3232	Microsoft Windows Search P...	Microsoft Corporation
7,956 K	4736	Microsoft Windows Search P...	Microsoft Corporation
7,304 K	5040	Microsoft Windows Search P...	Microsoft Corporation
13,792 K	5484		
6,910 K	5348	Host Process for Windows S...	Microsoft Corporation
9,144 K	5115		
13,936 K	5180	Host Process for Windows S...	Microsoft Corporation
18,464 K	4056	Microsoft® Disk Defragmenter	Microsoft Corporation
10,948 K	320		
7,300 K	4552	Host Process for Windows S...	Microsoft Corporation
14,944 K	704	Local Security Authority Po...	Microsoft Corporation
3,120 K	864	UserMode Font Driver Host	Microsoft Corporation
10,624 K	612	Windows Logon Application	Microsoft Corporation
6,884 K	856	UserMode Font Driver Host	Microsoft Corporation
83,240 K	1032	Desktop Window Manager	Microsoft Corporation
132,220 K	3912	Windows Explorer	Microsoft Corporation
8,084 K	5452	Windows Security notificati...	Microsoft Corporation
35,968 K	5272	Task Manager	Microsoft Corporation
31,508 K	1936	Microsoft Management Cons...	Microsoft Corporation
29,152 K	552	Microsoft Management Cons...	Microsoft Corporation
91,566 K	4076	Microsoft Edge	Microsoft Corporation



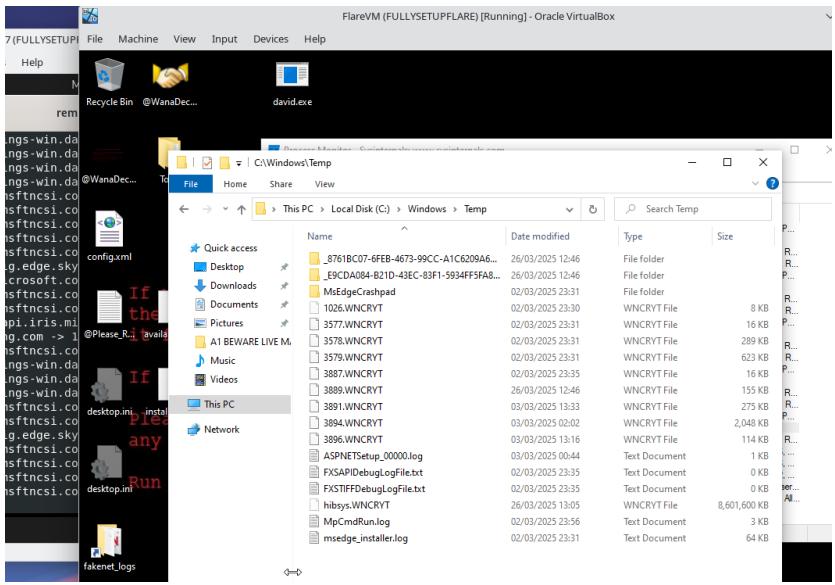
Have captured executables; tasksche.exe, taskhsvc.exe, taskdl.exe and David.exe. I can see the parameter -m security on david.exe confirming malware attempting to make an HTTP request to kill-switch domain. Observed tasksche.exe has the name DiskPart. Another interesting thing is the command line that the executables are run from, zfkzrmfkuwsdsn781, which appears not to represent an actual/typical ProgramData name.

I can see rzfkzrmfkuwsdsn781 as a registered service. It shows this is one of the persistent services the malware created. Although marked "Stopped" this still validates it was registered on the system and capable of being executed on startup.



Can identify rzfkzrmfkuwsdsn781 directory in File Explorer. Directory contains the .exe that was previously analyzed as part of the process (i.e. taskdl.exe, tasksche.exe). Can see PLEASE_READ_ME.txt file and the WannaDecryptor.exe, as well as .eky, .pky, and .res files, which are likely encryption/decryption-related files for ransomware. TaskData directory also contains a Tor folder, suggesting malware may have intended to communicate via Tor network for anonymity.

Applied PID filter of taskdl.exe to analyze and observe the process encrypting files and locating them in the Windows\Temp directory appending a .WNCRYT extension.



Can observe .WNCRYT files in identified folder using File Explorer.

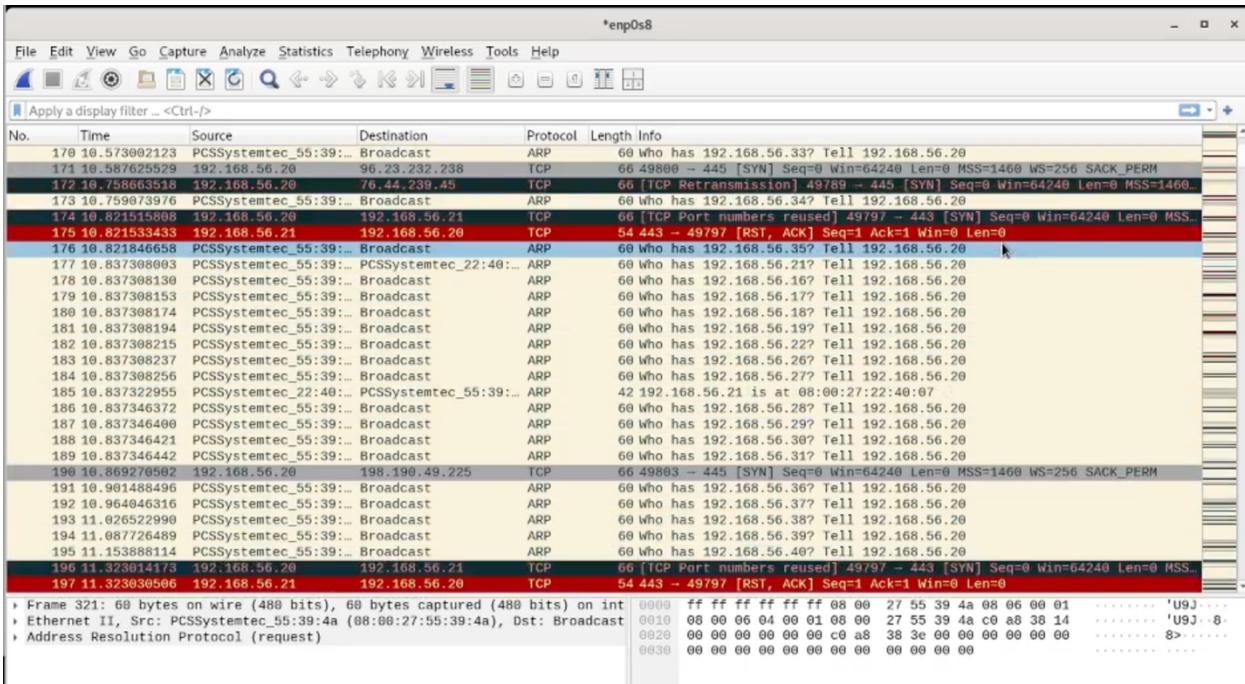
FakeDNS

```
remnux@remnux: ~
[1] 11189 228.414931000 192.168.56.20
fakedns[INFO]: dom.query: 60 IN A 192.168.56.21
fakedns[INFO]: Response: www.iuqerfsodp9ifjaposdfjhgosurijfaewrwegwea.com -> 192.168.56.21
fakedns[INFO]: Response: www.iuqerfsodp9ifjaposdfjhgosurijfaewrwegwea.com -> 192.168.56.21
fakedns[INFO]: Response: settings-win.data.microsoft.com -> 192.168.56.21
fakedns[INFO]: Response: settings-win.data.microsoft.com -> 192.168.56.21
fakedns[INFO]: Response: settings-win.data.microsoft.com -> 192.168.56.21
fakedns[INFO]: Response: config.edge.skyape.com -> 192.168.56.21
fakedns[INFO]: Response: fs.microsoft.com -> 192.168.56.21
fakedns[INFO]: Response: config.edge.skyape.com -> 192.168.56.21
fakedns[INFO]: Response: www.iuqerfsodp9ifjaposdfjhgosurijfaewrwegwea.com -> 192.168.56.21
fakedns[INFO]: Response: www.iuqerfsodp9ifjaposdfjhgosurijfaewrwegwea.com -> 192.168.56.21
fakedns[INFO]: Response: 1.56.168.192.in-addr.arpa -> 192.168.56.21

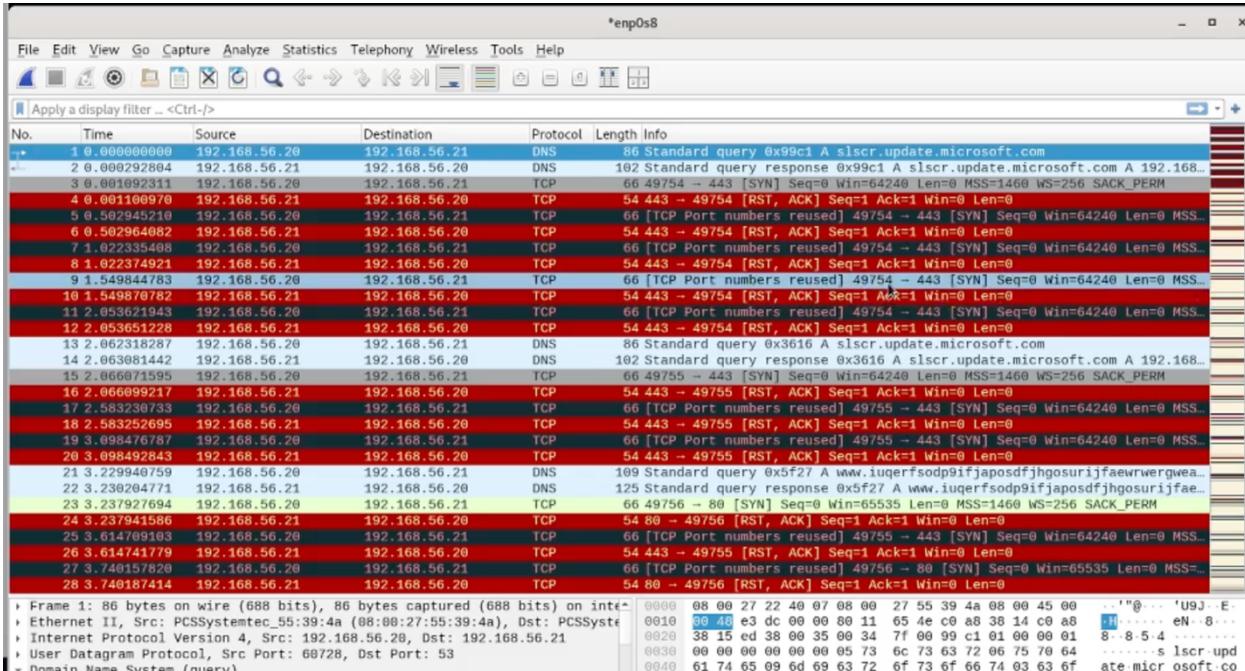
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
remnux@remnux: ~
[2] 11190 228.414931000 192.168.56.20
fakedns[INFO]: Response: config.edge.skyape.com -> 192.168.56.21
fakedns[INFO]: Response: dns.msftncsi.com -> 192.168.56.21
fakedns[INFO]: Response: dns.msftncsi.com -> 192.168.56.21
fakedns[INFO]: Response: settings-win.data.microsoft.com -> 192.168.56.21
fakedns[INFO]: Response: settings-win.data.microsoft.com -> 192.168.56.21
fakedns[INFO]: Response: dns.msftncsi.com -> 192.168.56.21
fakedns[INFO]: Response: dns.msftncsi.com -> 192.168.56.21
fakedns[INFO]: Response: config.edge.skyape.com -> 192.168.56.21
fakedns[INFO]: Response: fs.microsoft.com -> 192.168.56.21
fakedns[INFO]: Response: dns.msftncsi.com -> 192.168.56.21
fakedns[INFO]: Response: update.googleapis.com -> 192.168.56.21
fakedns[INFO]: Response: config.edge.skyape.com -> 192.168.56.21
fakedns[INFO]: Response: dns.msftncsi.com -> 192.168.56.21
fakedns[INFO]: Response: config.edge.skyape.com -> 192.168.56.21
fakedns[INFO]: Response: fs.microsoft.com -> 192.168.56.21
fakedns[INFO]: Response: dns.msftncsi.com -> 192.168.56.21
fakedns[INFO]: Response: dns.msftncsi.com -> 192.168.56.21
fakedns[INFO]: Response: download.windowsupdate.com -> 192.168.56.21
fakedns[INFO]: Response: dns.msftncsi.com -> 192.168.56.21
[3] 11191 228.414931000 192.168.56.20
fakedns[INFO]: Response: 66 [TCP Retransmission] 60873 -> 445
[4] 11192 228.414931000 192.168.56.20
fakedns[INFO]: Response: 67 -> 445
[5] 11193 228.414931000 192.168.56.20
fakedns[INFO]: Response: 59 -> 445
[6] 11194 228.414931000 192.168.56.20
fakedns[INFO]: Response: 72 -> 445
[7] 11195 228.414931000 192.168.56.20
fakedns[INFO]: Response: 52 -> 445
[8] 11196 228.414931000 192.168.56.20
fakedns[INFO]: Response: 57 -> 445
[9] 11197 228.414931000 192.168.56.20
fakedns[INFO]: Response: 60 -> 445
[10] 11198 228.414931000 192.168.56.20
fakedns[INFO]: Response: 69 -> 445
[11] 11199 228.414931000 192.168.56.20
fakedns[INFO]: Response: 63 -> 445
[12] 11200 228.414931000 192.168.56.20
fakedns[INFO]: Response: 67 -> 445
[13] 11201 228.414931000 192.168.56.20
fakedns[INFO]: Response: 59 -> 445
[14] 11202 228.414931000 192.168.56.20
fakedns[INFO]: Response: 72 -> 445
```

Can observe the first domain the malware attempts to connect with is the kill switch domain identified during string analysis. FakeDNS responded to this, simulating a successful connection, however, malware still executes, indicating connection may not have been treated as valid by the malware. Malware then continues querying other domains.

Wireshark



Large number of ARP packets being broadcasted, attempting to discover other devices on the subnet. This visual indicates the malware spreading mechanism, scanning 192.168.0-255 IP range to see other potential victims.

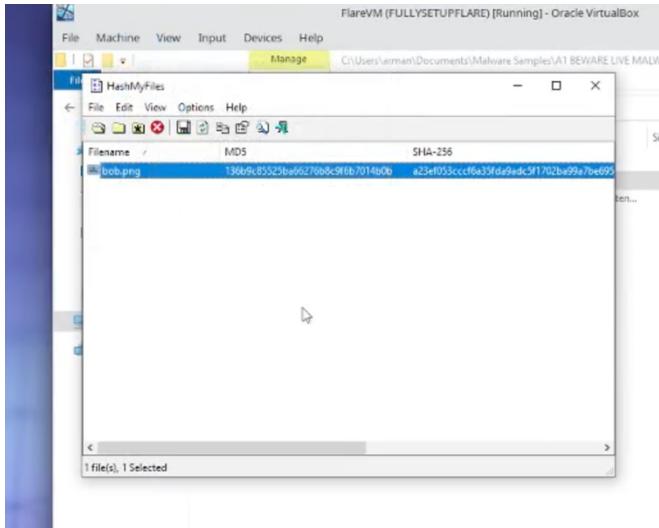


I can see DNS queries being made to the kill switch domain. Although it simulates a DNS response, the malware still does not receive a proper HTTP response from the server since we are not hosting the actual domain, resulting in it believing the kill switch is unreachable.

3. Bob

3.1. Basic Static Analysis

VirusTotal



I began investigating acquiring MD5 hash of files to search within VirusTotal. This would provide information on the malware if it had been previously analyzed and flagged by antivirus engines.

A screenshot of the VirusTotal website. At the top, it says "61 / 73 security vendors flagged this file as malicious". Below this is a table of vendor detections. The table includes columns for vendor name, threat type, and detection status (e.g., "detected", "malicious"). Key entries include "AlmLab-V3" (Trojan/Win32.RL_Generic.R33592Z), "AlCloud" (Trojan[stealer]!Win!Avaddon.Gen), "Anty-AVL" (Trojan(Banker)!Win32.Qbot), "Avast" (Win32.DangerousSig![?]), "Avira (no cloud)" (HEUR/HGEN.1369812), and "Bkav Pro" (W32!AI!Detect!Malware). The table also lists "Alibaba", "AVG", "BitDefender", and "ClamAV" as other vendors. A sidebar on the right shows a "Community Score" of 61/73 and a "Join our Community" button.

eScan	Trojan.Agent.ERUJ	ESET-NOD32	A Variant Of Win32/Krypt14-H.UH
Fortinet	W32/Kryptik.HG9ttr	GData	Trojan.Agent.ERUJ
Google	Detected	Gridinsoft (no cloud)	Trojan.Win32.Kryptik.ddm
Haorong	HVM_VerTool/OBFuscator.k	Ikarus	Trojan.Banker.QabBot
Jiangmin	Trojan.Banker.Qbot.qe	K7Antivirus	Trojan!Trojan.Banker!Win32.Qbot.vho
K7GW	Trojan (.009616c1)	Kaspersky	HEUR:Trojan-Banker:Win32.Qbot.vho
Lionic	Trojan.Win32.QabBot.7c	Malwarebytes	Backdoor.Qbot
MaxSecure	Banker.Banker.Win32.Qbot.vho_106495	McAfee Scanner	Trojan!Trojan!Qbot
Microsoft	Ransom!Win32!Avaddon	Palo Alto Networks	Generic.ml
Panda	Troj/GdSdA	QuickHeal	Trojan.Ghanarava!T7347913!0!4b0d
Rising	Trojan.Kryptik!C745 (CLASSIC)	Sangfor Engine Zero	Trojan.Win32.Save.a
SecureAge	Malicious	SentinelOne (Static ML)	Static AI - Malicious PE
Skyhigh (SWG)	W32/PwnBot-GUJ1H69C85525B	Sophos	Mal/EncPw-APV
Symantec	Trojan.Horse	Tencent	Malware.Win32.Gencart.10bc0d832
TreliX (ENS)	W32/PwnBot-GUJ1H69C85525B	TreliX (HK)	Generic.mg.1360c03525ba062
TrendMicro	Backdoor.Win32.QAKBOT.SME	TrendMicro-HouseCall	Backdoor.Win32.QAKBOT.SME
Varist	W32/Trojan.DZ!genEldorado	VBA32	BScope.TrojanDownloader.Dridex

61 out of 73 vendors identified the malware as malicious. Identified by antivirus engines under Backdoor.QBot.gen and Trojan.Agent.ERUJ. Can determine that bob.png is related to Qbot ransomware, which steals banking data (e.g. credentials, session information). Did not manually compare AV signatures with ClamAV, however, VirusTotal includes ClamAV engine, which confirms detection, Win.Ransomware.Locky-9779179-0.

VirusTotal analysis results for a file flagged as malicious by 61 security vendors. The file is an EXE named 8888888.png, size 1.16 MB, last analyzed 2 days ago. The analysis highlights several tags: Peer, checks-user-input, spreader, runtime-modules, idle, overlay, long-sleeps, direct-cpu-clock-access, signed, service-scan, checks-memory-available, checks-bcos, detect-debug-environment, invalid-signature, calls-wmi, persistence, and checks-disk-space.

Basic properties

MD5	13d9c85525ba66276b8f9fb7014b0b
SHA-1	0c5ba13d142b605867f4b497925/a4d172
SHA-256	a23ef053cccfa6a35fda9adc5f1702ba99a7be695107d3ba5d1ea8c9c258299e4
Vhash	8888888.png
Authentihash	a23ef053cccfa6a35fda9adc5f1702ba99a7be695107d3ba5d1ea8c9c258299e4
ImpHash	01605051dc5551100600060006007171501000bf
SSDeep	6144_JarboBox5TBRC6nVFW4LvsKgTSjEtIovH02gJGuhRTISUgsS108YbC3g/T4.JaKoRtBHWc4JNSp/EMGu/SHomal
TLSH	T1405C0407305CAF1F9EF4D8656A4409E4BA4D01B919068CFDF529295F87A3
File type	Win32 EXE [executable] [window] [win32] [pe] [pexe]
Magic	PE32 executable (GUI) Intel 80386, for MS Windows
TiID	Win32 Executable MS Visual C++ (generic) (45.8%) Win64 Executable (generic) (15.4%) Win32 Dynamic Link Library (generic) (9.6%) Win16...
DetectIEEasy	PE32 Compiler: Embarcadero Delphi (2009-2010) Linker: Polink (2.50') Sign tool: Windows Authenticode (2.0 [PKCS #7])
Maxika	PERIN
File size	1.16MB (1214992 bytes)
PEiD packer	Microsoft Visual C++ vx.x

Compiler of the malware is Embarcadero Delphi. However, was flagged being compiled with Microsoft Visual C++. Might suggest multiple components or packing. PEiD packer states using a Microsoft Visual C++ vx.x packer. Can observe that it's PE32 architecture. Can also see VirusTotal tags, persistence, checks-debug-environment and service scan, suggesting it employs anti-analysis, persistence and network scanning techniques.

History	
Creation Time	2020-06-04 13:33:54 UTC
First Seen In The Wild	2020-06-06 08:11:56 UTC
First Submission	2021-06-21 14:01:43 UTC
Last Submission	2025-03-27 21:54:44 UTC
Last Analysis	2025-03-25 06:06:57 UTC

Can see creation time was 2020-06-04, which is likely when the binary was compiled for the first time.

File Version Information	
Copyright	Copyright © 2008-2013 Lovelysoft. All rights reserved
Product	AdminToys Suite
Description	Remote Performance Expert Helper Object
File Version	1.0.0.1800

Malware does file spoofing to appear as legitimate software. Claims to be part of the AdminToys Suite by Lovelysoft, describing itself as a "Remote Performance Helper Object".

Portable Executable Info						
Header						
Target Machine		Intel 386 or later processors and compatible processors				
Compilation Timestamp		2020-06-04 13:33:54 UTC				
Entry Point		10576				
Contained Sections		5				
Sections						
Name	Virtual Address	Virtual Size	Raw Size	Entropy	MDS	Chi2
.text	4096	1077644	1077760	2.96	cfb90e8a6aa6adb0067eadcd6ec851015	142733296
.rdata	1085440	261	512	2.56	6394519a6326430bbdfcc4ed9a21b43	71031
.data	1089536	20868	20992	5.67	dd275be55c007ec0311de1360f7f1e56	311888.19
r2	1114112	43415	43520	5.16	f4e0a11b39757039da7b4a8cef59453f	2422658.25
.rsrc	1159168	69636	70144	5.77	aafaf61c81fed312ab375d66904ce42032	1854768.62

PE header indicates the malware targets Intel386 or later processes. Has five contained sections: .text, .rdata, .data, .r2, and .rsrc. Indicates entropy of 2.96 and 2.56 for .text and .rdata respectively. This is relatively low and doesn't indicate packing. The entropy of 5.67, 5.16 and 5.77 for .data, .r2 and .rsrc, respectively, indicates a possible use of obfuscation or packing. .r2 refers to the Rich Header, which contains information about the build environment used to compile/link the file.

Imports

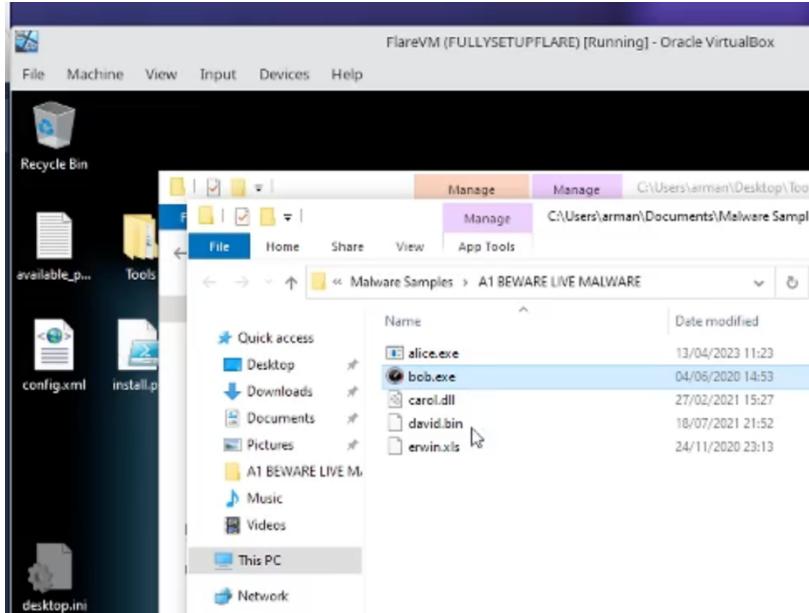
- + KERNEL32.dll
- + USER32.dll
- + GDI32.dll
- + COMDLG32.dll
- + ADVAPI32.dll
- + SHELL32.dll
- + ole32.dll
- + SHLWAPI.dll
- + COMCTL32.dll

Can see the DLLs that the malware uses. SHELL32.dll and SHLWAPI.dll imply malware interacting with Windows Shell to launch executables and manage files for executing payloads. ole32.dll is used for object linking and embedding, which is used in malware to interact with documents/applications.

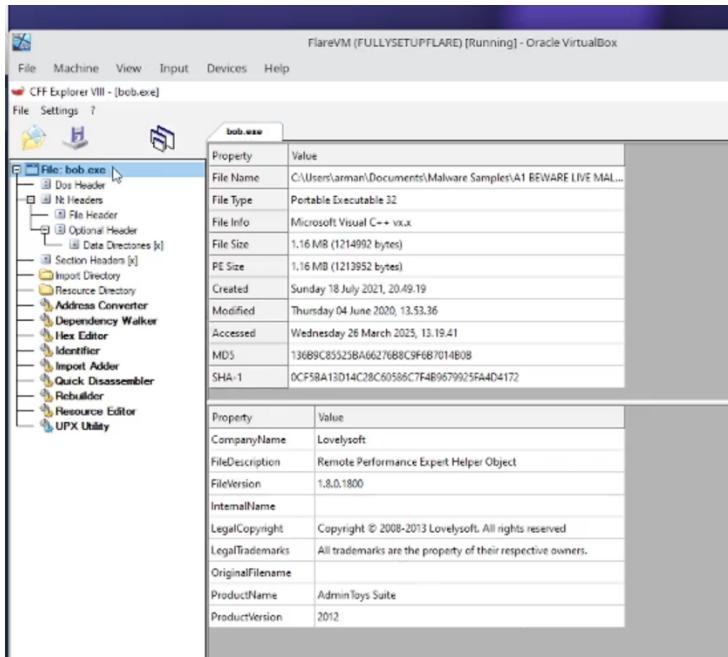
The screenshot shows the VirusTotal analysis interface. At the top, it lists several engines that detected the sample as malware, including Intel HASU, VirusTotal.JujuBox, VirusTotal.Cuckoo, VirusTotal.Observer, and Zenbox. Below this is an 'Activity Summary' section with various metrics: 2 detections (1 malware, 1 ransom), 10 Mitre Signatures (1 High, 10 Low, 44 Info), 2 IDS Rules (2 High, 1 Low), 0 Sigma Rules (NOT FOUND), 1 Dropped Files (1 HTML, 1 TEXT, 1 PE_32, 1 OTHER, 1 LINK, 1 SCRIPT, 1 BMP, 1 ZIP), and 1 Network comms (1 HTTP, 3 DNS, 300 IP). The 'Behavior Tags' section includes tags like @droppedfile, @detected-instrument, @log-drops, and @malware-detect. A 'Dynamic Analysis Sandbox Detections' section shows that the Zenbox sandbox flagged the file as MALWARE RANSOM. The 'MITRE ATT&CK Tactics and Techniques' section details various exploit methods, including Execution (e.g., Command shell drops VBS files, Executes powershell scripts, Executes batch files), Persistence (e.g., Boot or Login Autostart Execution, Registry Run Keys / Startup Folder, Stores files in the Windows startup directory), and Privilege Escalation (e.g., Process Injection, Creates a process in suspended mode (likely to inject code), Spawns processes).

Analyzed the Sandbox report done by Zenbox on VirusTotal. Stated that execution is performed by command shell drops and executes scripts and batch files. It has persistency from startup, boot, and logon execution and registry keys stored in the Windows startup directory. It has also several defence evasion techniques through file manipulation, indicator removal and hiding artifacts and processes.

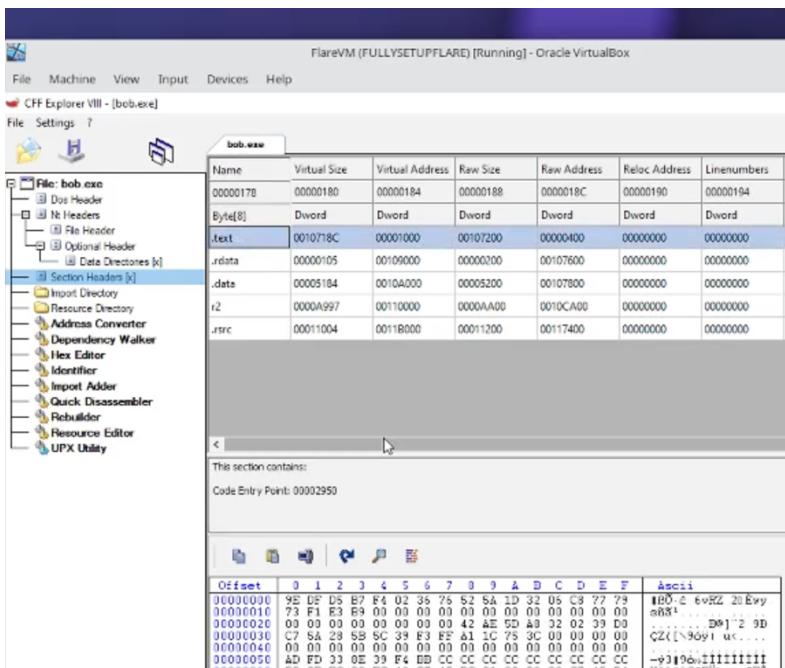
CFF Explorer



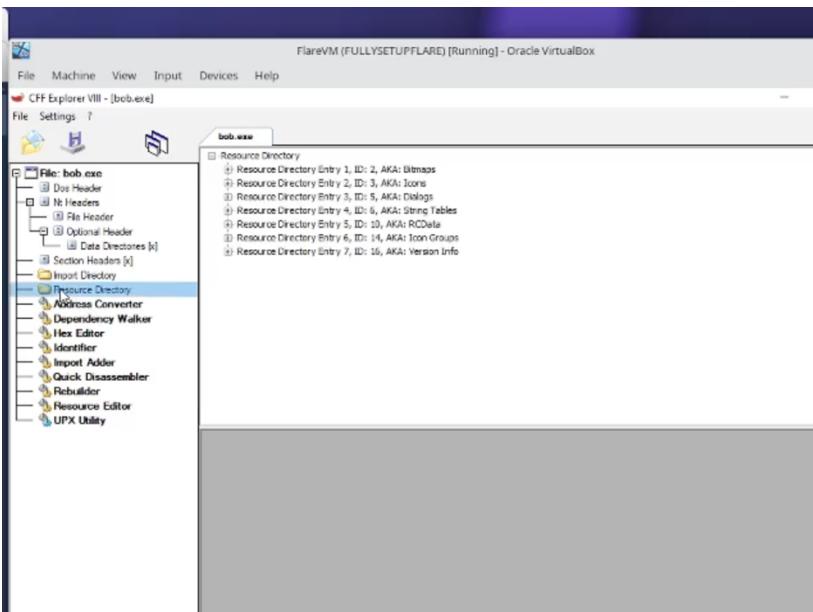
First, I changed the bob.png extension to bob.exe so we can analyze it using the tools we want to use to perform static analysis.



The malware is a 32-bit PE compiled using Microsoft Visual C++ vx.x. Can also see the creation date of July 18 2021, which does not match the creation time seen on VirusTotal.

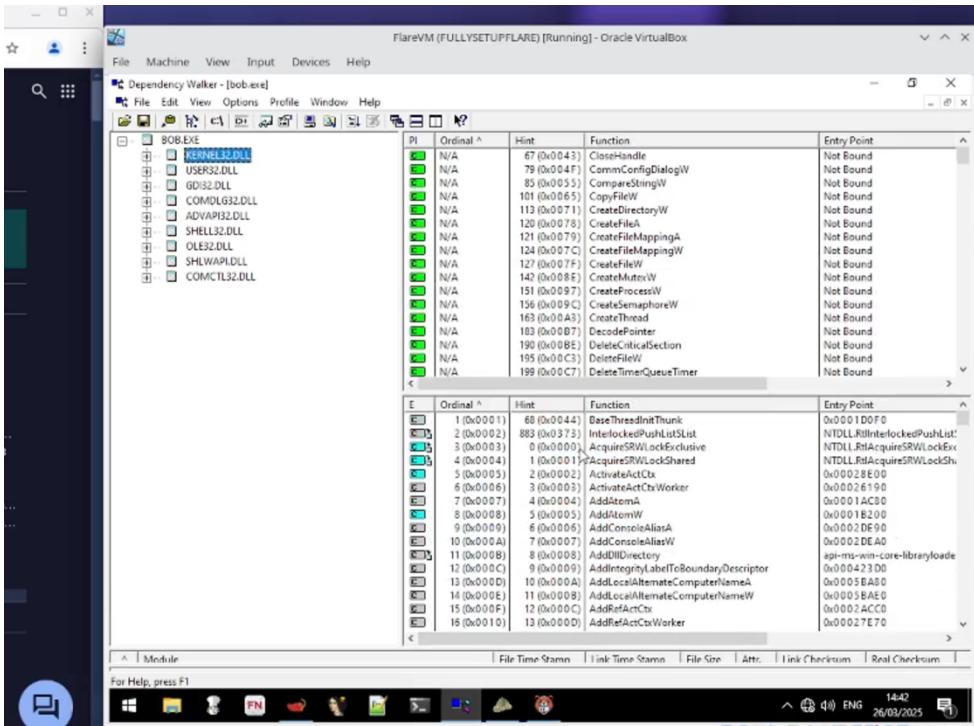


Analyzed PE Section headers. Clear that the malware is not packed with tools like UPX (e.g. UPX0, UPX1). Virtual and raw sizes are relatively similar, which would be unlikely if packed.



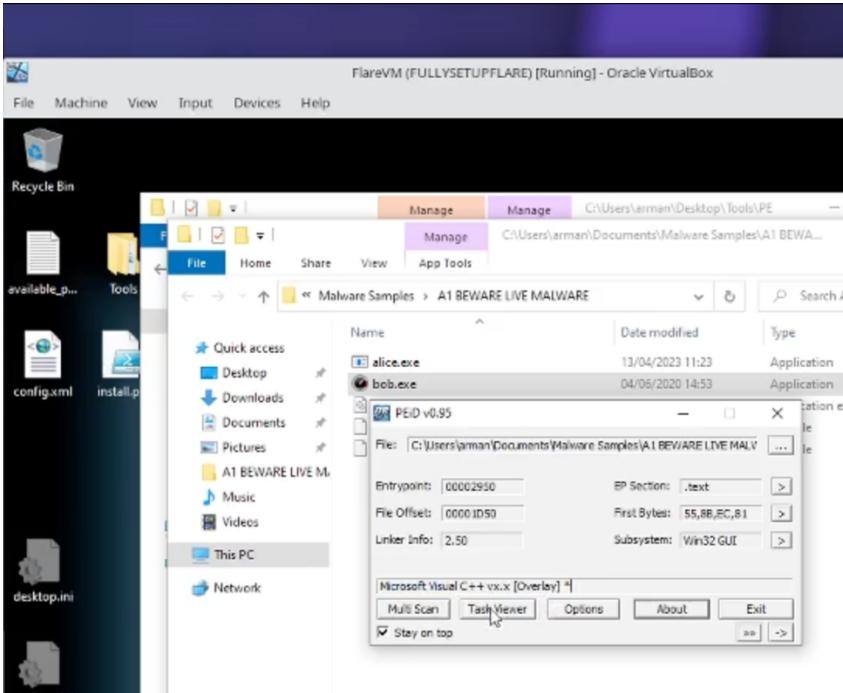
Next, I looked at the resource directory, which included typically GUI-related assets like Bitmaps, icons, dialogs, string tables, and version info.

Dependency Walker

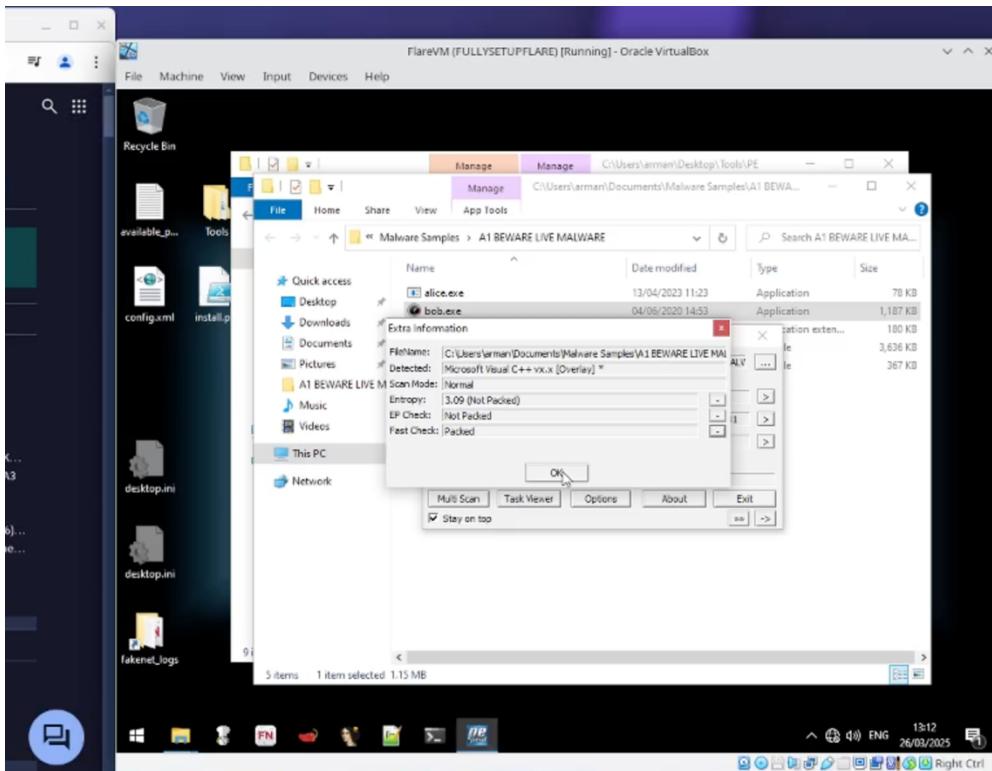


Upper pane displays functions from KERNEL32.dll where we can see dynamically loaded functions at runtime like CreateFileW, CreateThread, CreateProcessW, and DeleteFileW, suggesting it performs file manipulation, etc. From the extent of DLLs, I found it better to analyze them through string analysis.

PEiD

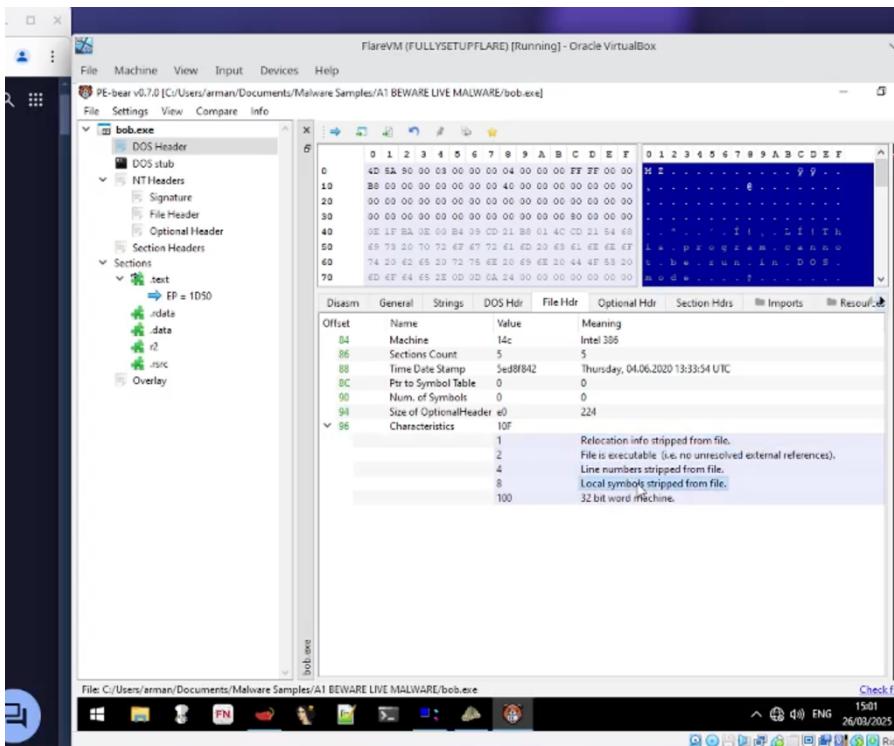


Identifies the entry point of file, which lies in the .text section. Identifies malware was compiled using Microsoft Visual C++ vx.x



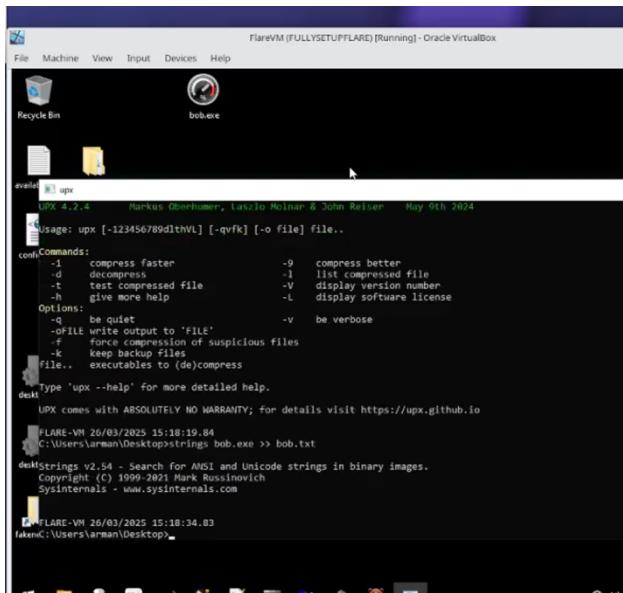
According to PEiD, the malware has an entropy of 3.09, which generally falls within the range of unpacked and non-obfuscated binaries. EP Check also reported not packed. However, the fast check says packed, sometimes producing false positive results.

PE-Bear



File header confirmed timestamp of compilation date of bob.exe. Characteristics revealed the file is an executable, and several components had been stripped, such as; relocation info, local symbols and line numbers. Confirms compilation date 04.06.2020 as seen before in VirusTotal.

String Analysis through txt file



```
FlareVM (FULLYSETUPFLARE) [Running] - Oracle VirtualBox
File Machine View Input Devices Help
Recycle Bin bob.exe
available_p... upx
UPX 4.2.4 Markus Oberhumer, Laszlo Molnar & John Reiser May 9th 2024
Usage: upx [-123456789dltvhVL] [-qvfk] [-o file] file...
  -1 compress faster           -9   compress better
  -d decompress                -l   list compressed file
  -t test compressed file      -V   display version number
  -h give more help            -L   display software license
Options:
  -q   be quiet                 -v   be verbose
  -FILE write output to 'FILE'
  -r   report list of suspicious files
  -k   keep backup files
File..  executables to (de)compress
Type 'upx --help' for more detailed help.

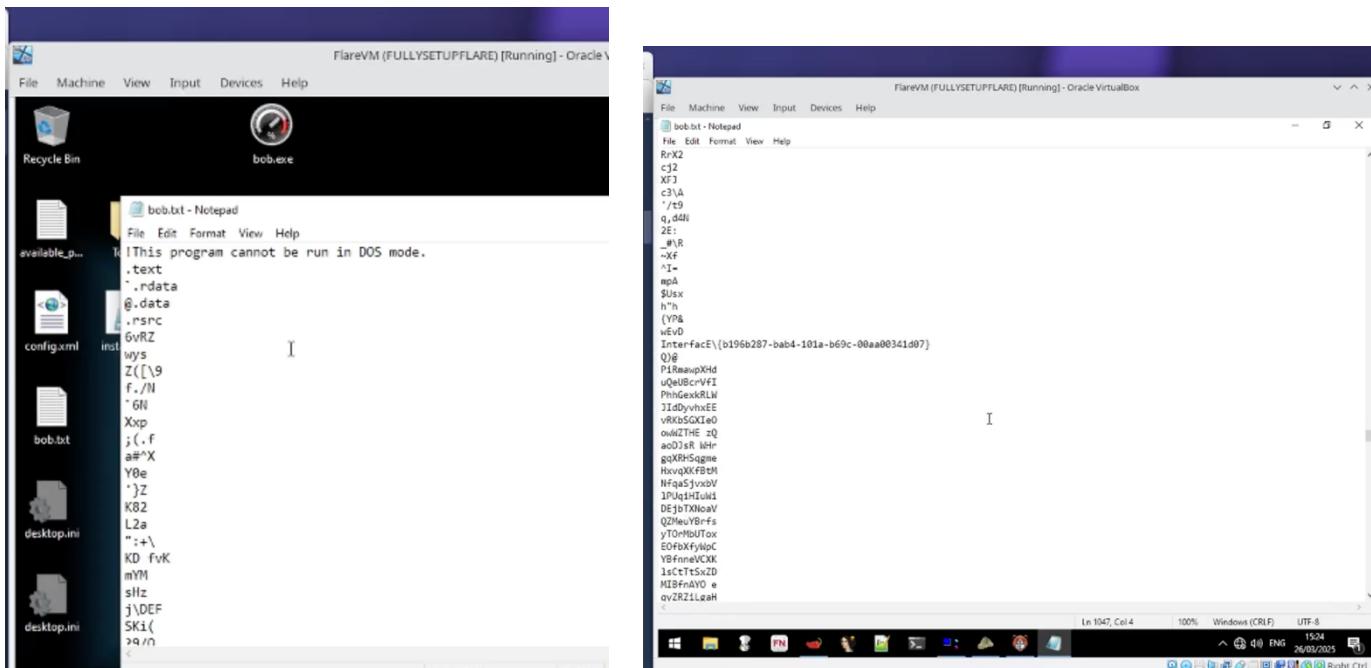
UPX comes with ABSOLUTELY NO WARRANTY; for details visit https://upx.github.io

FLARE-VM 26/03/2025 15:18:19.84
C:\Users\larman\Desktop>strings bob.exe >> bob.txt

Strings v2.54 - Search for ANSI and Unicode strings in binary images.
Copyright (C) 1999-2021 Mark Russinovich
Sysinternals - www.sysinternals.com

FLARE-VM 26/03/2025 15:18:34.03
larman:C:\Users\larman\Desktop>
```

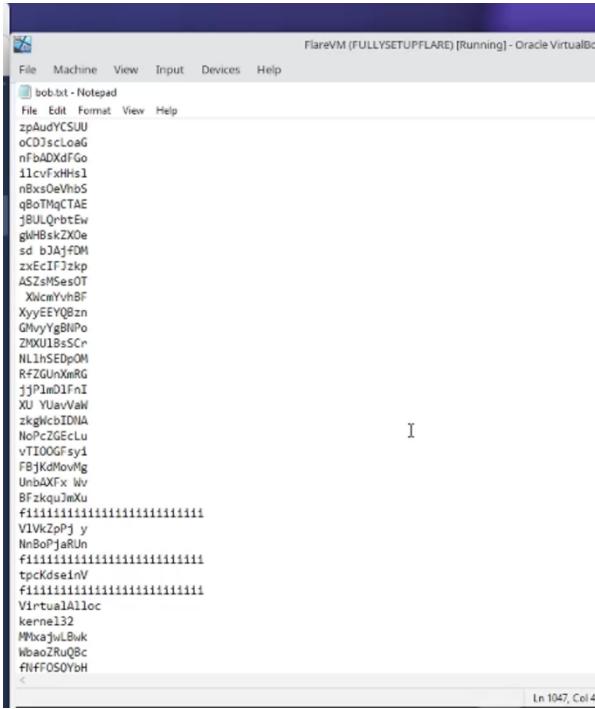
Analyzed strings by outputting the strings to a text file. This was achieved through UPX terminal using command line "strings bob.exe >> bob.txt". Then, I opened file in Notepad to analyze.



```
FlareVM (FULLYSETUPFLARE) [Running] - Oracle VirtualBox
File Machine View Input Devices Help
Recycle Bin bob.exe
available_p... bob.txt - Notepad
File Edit Format View Help
This program cannot be run in DOS mode.
.rtext
-.rdata
@.rsrc
6vRZ
wys
Z((\9
f./\1
`GN
Xxp
j(. f
#^X
Y0e
)`Z
K82
L2a
"+\
KD fvK
mYM
sflz
j\DEF
SK1(
>9/\n

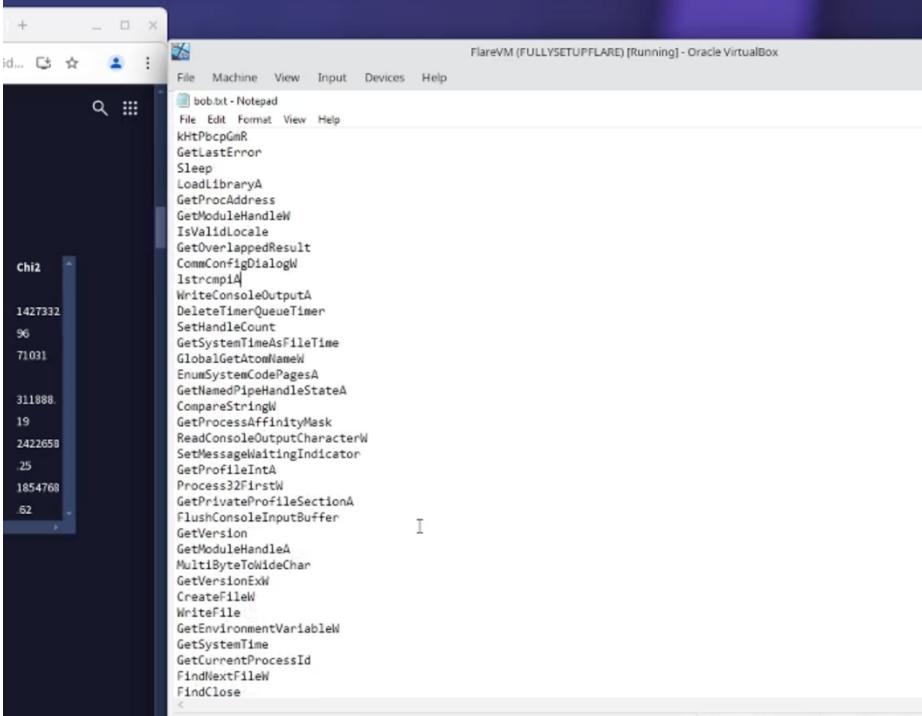
bob.txt - Notepad
File Edit Format View Help
RrX2
c3J2
X\7
c3\A
`/t9
q,\$dN
2E;
_"\R
-xF
^I-
npA
\$usx
`Vn
(Y08
wEvD
InterfacE\b196b287-bab4-101a-b69c-00aa00341d07\b
Q1@
P1RauapXhd
uqeUBcVFI
PhhGexkRLW
J1dDmxEE
WvS-GOZO
owzTHEZQ
aeD1sRiMr
gqXRhsAgne
HavqXXfBTM
NfqaS1xvDV
1P0q1hTulu1
DEjbTXNoAV
Q2MeuYBrfs
yTOMhD0T0x
EOFOhC
VBfrnefCK
1cCtTsZxD
M1BFnAOYD
oyZRZLlshH
```

I can see that the strings at the start of the file are heavily obfuscated, making it difficult to interpret. We can see human-readable strings just over the halfway point of the text file.



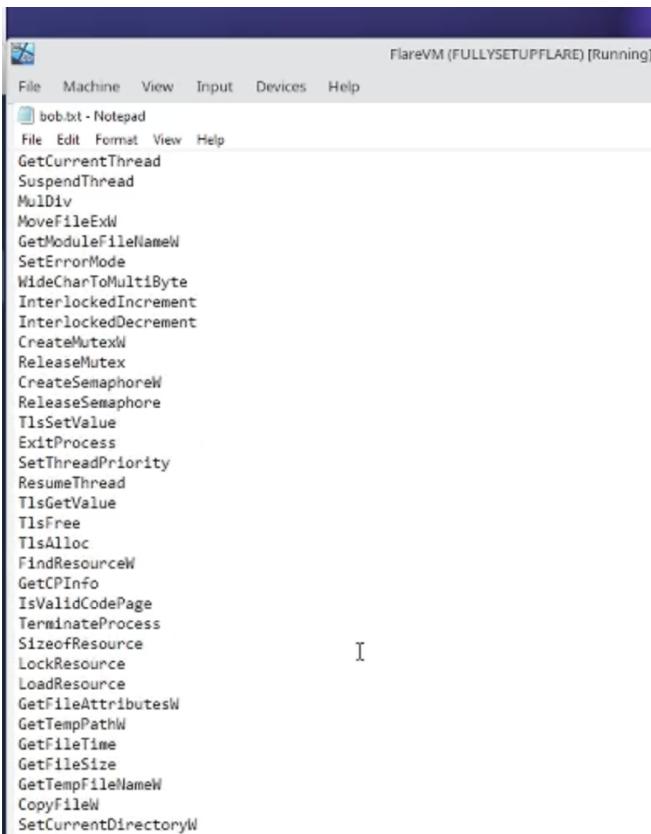
```
FlareVM (FULLYSETUPFLARE) [Running] - Oracle VirtualBox  
bob.txt - Notepad  
File Edit Format View Help  
zpaAudyCSUU  
oC01scLoaG  
nfb4DXdfGo  
l1cvfxHhs1  
nBxs0eVnb5  
qB0tMqCTAE  
jBULQrbtEw  
gWBskZX0e  
sd bJAifDM  
zxEcIfJzkp  
ASZsMSesOT  
XicavvhBF  
XxyEEYQBzn  
GMvyYgBNPo  
ZMKU1BsScr  
NL1nSEDpOM  
RFZGUnXmRG  
jjPlmDfni1  
XU YUavVaW  
zkgxcbIDNA  
NoPcZGEcLu  
vTIOOGfsy1  
FBjKdNovig  
UnoAXFx_wv  
BFzkquJmKu  
f1iiiiiiiiiiiiiiiiiiiiiiiiiiiiiiii  
V1VkpZpJ_y  
NnBoPjaRUn  
f1iiiiiiiiiiiiiiiiiiiiiiiiiiiiiiii  
tpcKdseInV  
f1iiiiiiiiiiiiiiiiiiiiiiiiiiiiiiii  
VirtualAlloc  
kernel32  
MMxajjvLBuk  
WbaodZLQ8c  
FFFOSOYbH  
<
```

The repeated 'fiiiiii' string appears to serve as a placeholder or anti-analysis technique. Can see function VirtualAlloc, which is used for memory allocation and is often linked with shellcode loading or unpacking routines.

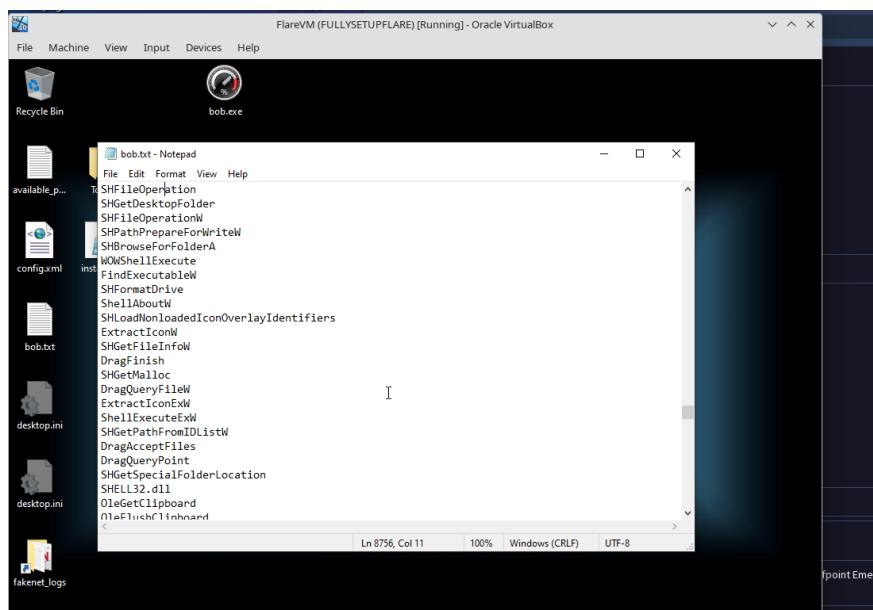


```
FlareVM (FULLYSETUPFLARE) [Running] - Oracle VirtualBox  
bob.txt - Notepad  
File Edit Format View Help  
KhtPbcpcGmR  
GetLastError  
Sleep  
LoadLibraryA  
GetProcAddress  
GetModuleHandleW  
IsValidLocale  
GetOverlappedResult  
CommConfigDialogW  
lstrcmplA  
WriteConsoleOutputA  
DeleteTimerQueueTimer  
SetHandleCount  
GetSystemTimeAsFileTime  
GlobalGetAtomNameW  
EnumSystemCodePagesA  
GetNamedPipeHandleStateA  
CompareStringW  
GetProcessAffinityMask  
ReadConsoleOutputCharacterW  
SetMessageWaitingIndicator  
GetProfileIntA  
Process32FirstW  
GetPrivateProfileSectionA  
FlushConsoleInputBuffer  
GetVersion  
GetModuleHandleA  
MultiByteToWideChar  
GetVersionExW  
CreateFileW  
Writefile  
GetEnvironmentVariableW  
GetSystemTime  
GetCurrentProcessId  
FindNextFileW  
FindClose  
<
```

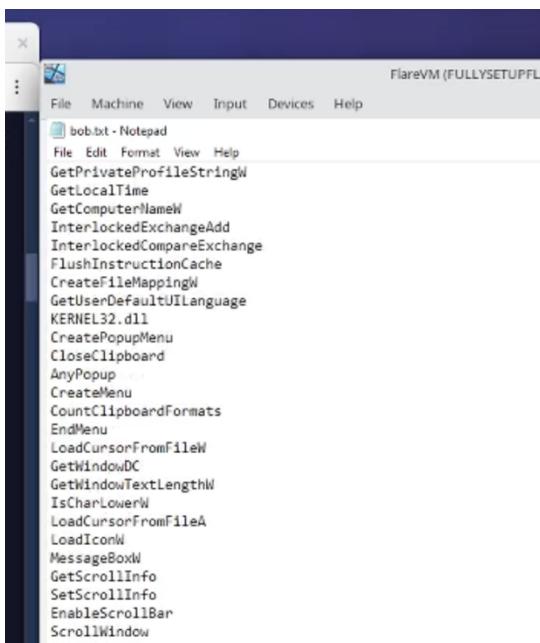
The GetLastError function retires most recent error code for debugging. The Sleep function is often used for anti-analysis, introducing delays to evade analysis. LoadLibraryA and GetProcAddress are frequently used for dynamic function resolution so the malware can load DLLs and resolve pointers at runtime. Can also see CreateFile and WriteFile, which suggests file system interaction.



I can see thread-related functions: GetCurrentThread, SuspendThread, and ResumeThread, which allow malware to control thread execution. CreateMutexW and TLSAlloc, implying synchronization to manage multiple threads. Memory-focused functions; LoadResource, SizeofResource and LockResource - points toward unpacking/loading payloads during runtime.

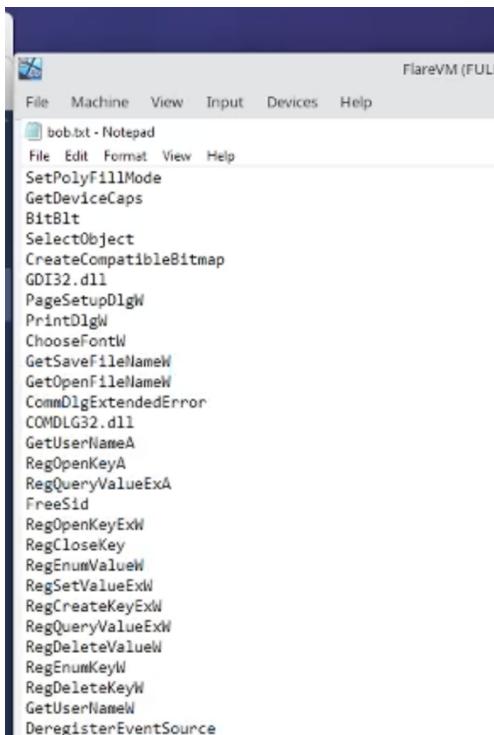


Can see functions ShellExecuteW, ShellExecuteExW, and WOWShellExecute - provides malware with ability to execute additional programs or scripts.



```
FlareVM (FULLYSETUPFL)
File Machine View Input Devices Help
bob.txt - Notepad
File Edit Format View Help
GetPrivateProfileStringW
GetLocalTime
GetComputerNameW
InterlockedExchangeAdd
InterlockedCompareExchange
FlushInstructionCache
CreateFileMappingW
GetUserDefaultUILanguage
KERNEL32.dll
CreatePopupMenu
CloseClipboard
AnyPopup
CreateMenu
CountClipboardFormats
EndMenu
LoadCursorFromFileW
GetWindowDC
GetWindowTextLengthW
IsCharLowerW
LoadCursorFromFileA
LoadIconW
MessageBoxW
GetScrollInfo
SetScrollInfo
EnableScrollBar
ScrollWindow
```

GUI-related functions, CreatePopupMenu, CreateMenu, ScrollWindow and IsWindowVisible, indicates attempts to trick the user or display fake system messages. Functions CloseClipboard and CountClipboardFormats suggest the malware attempting to monitor/interact with clipboard contents.



```
FlareVM (FUL)
File Machine View Input Devices Help
bob.txt - Notepad
File Edit Format View Help
SetPolyFillMode
GetDeviceCaps
BitBlt
SelectObject
CreateCompatibleBitmap
GDI32.dll
PageSetupDlgW
PrintDlgW
ChooseFontW
GetSaveFileNameW
GetOpenFileNameW
CommDlgExtendedError
COMDLG32.dll
GetUserNameA
RegOpenKeyA
RegQueryValueExA
FreeSid
RegOpenKeyExW
RegCloseKey
RegEnumValueW
RegSetValueExW
RegCreateKeyExW
RegQueryValueExW
RegDeleteValueW
RegEnumKeyW
RegDeleteKeyW
GetUserNameW
DeregisterEventSource
```

Registry-related functions such as RegOpenKeyA, RegOpenKeyExA, RegCreateKeyExW, RegSetValueExw and RegDeleteKey.

```

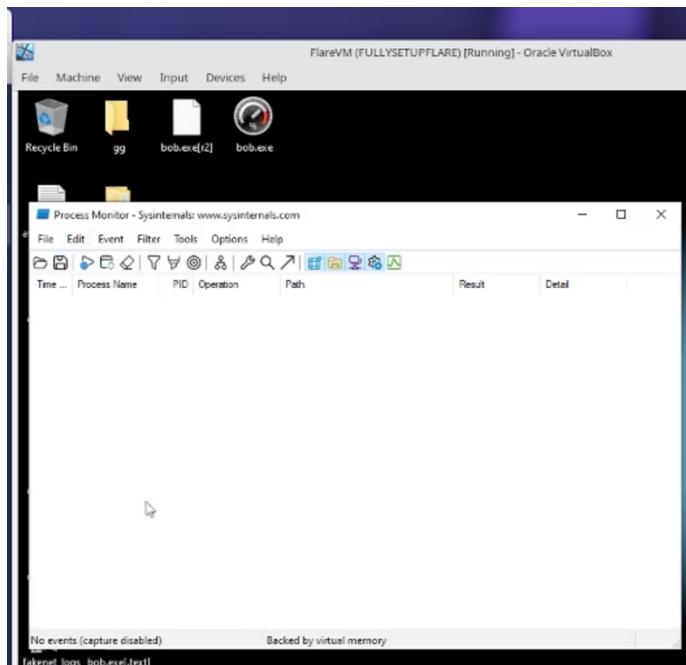
bob.txt - Notepad
File Edit Format View Help
Width
Height
Caption
Default
Enabled
ModalResult
TabOrder
TButton
btnCancel
Left
Top
Width
Height
Caption
Cancel
ModalResult
TabOrder
TOpenDialog
CommandLineDialog
Filter [
+Executable (*.exe)|*.exe|Any file (*.*)|*.*
FilterIndex
Left
Top
BorderIcons
biSystemMenu

```

The +Executable function suggests malware may invoke fake/real-life dialog to trick users into selecting a .exe (based on GUI aspect) for execution or programmatically selecting .exe files for injection/execution.

3.2. Basic Dynamic Analysis

Setup and Running Malware



FlareVM (FULLYSETUPFLARE) [Running] - Oracle VirtualBox

File Input Devices Help

Recycle Bin

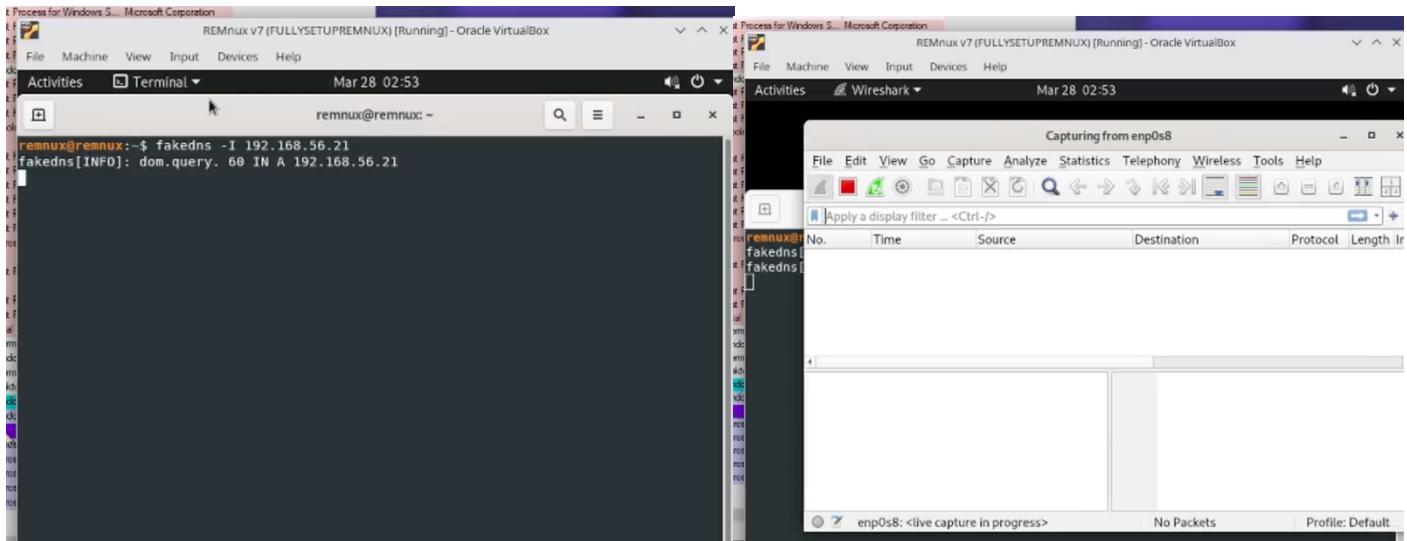
Process Explorer - Sysinternals www.sysinternals.com [C:\VR473\arman] (Administrator)

File Options View Process Find Users Help

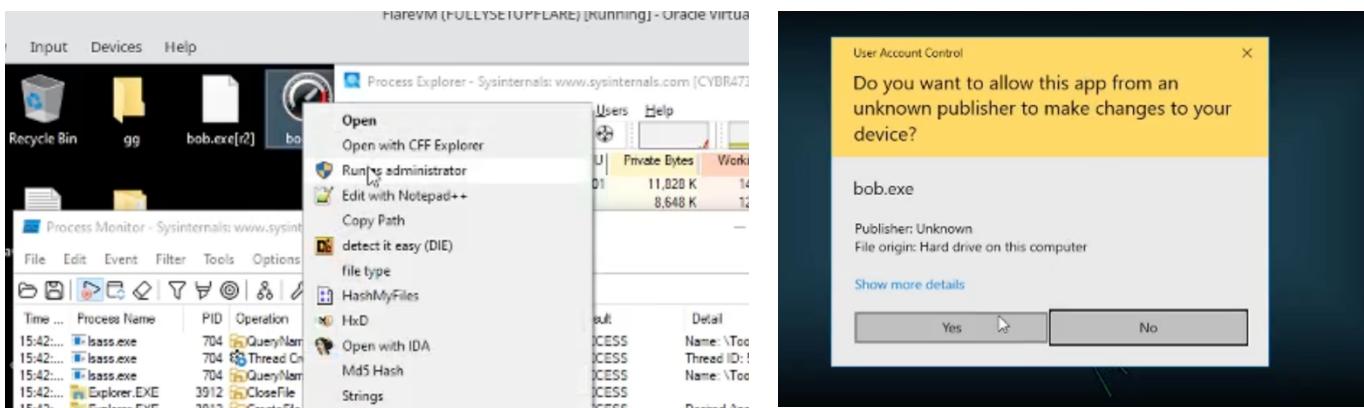
Process	CPU	Pinned	Working Set	PID	Description	Company Name
System Idle Process	84.21	40 K	19,382 K	108		
System	< 0.01	64 K	12 K	4	n/a	
Interrupts	0.77	0 K	0 K			
sms.exe		328 K	345 K	332		
console		1,248 K	1,848 K	440		
cacls.exe	< 0.01	1,356 K	2,388 K	540		
wmirev.exe		1,032 K	912 K	564		
services.exe	1.15	2,968 K	4,848 K	684		
svchost.exe		6,256 K	11,696 K	828	Host Process for Windows S...	Microsoft Corporation
StartMenuExperience	14.672 K	32,104 K	2712	2712		
RuntimeBroker.exe	5,192 K	6,716 K	3356	3356	Runtime Broker	Microsoft Corporation
SearchLoop.exe	94,464 K	129,108 K	4108	4108	Search application	Microsoft Corporation
Sup...		34,020 K	55,308 K	4440	Runtime Broker	Microsoft Corporation
RuntimeBroker.exe		11,092 K	25,516 K	2424	Process Monitor	Sytemate - www.syntate...
promote.exe		19,680 K	38,316 K	4972	Sytemate Process Explorer	Sytemate - www.syntate...
process.exe	7,404 K	12,104 K	5608	5608	Runtime Broker	Microsoft Corporation
RuntimeBroker.exe	3,864 K	3,040 K	5736	5736	Application Frame Host	Microsoft Corporation
TextInputHost.exe	6,836 K	10,908 K	6080	6080		Microsoft Corporation
shellExperienceHost...	11,624 K	30,744 K	1636	1636	Windows Shell Experience H...	Microsoft Corporation
RuntimeBroker.exe	5,916 K	14,576 K	1604	1604	Runtime Broker	Microsoft Corporation
dhost.exe	4,688 K	7,652 K	5856	5856	COM Sumguate	Microsoft Corporation
WmiPrvSE.exe	2,016 K	8,096 K	5928	5928	WMI Provider Host	Microsoft Corporation
svchost.exe	4,968 K	7,524 K	952	952	Host Process for Windows S...	Microsoft Corporation
taskhost.exe	< 0.01	27,948 K	40,020 K	1164	Host Process for Windows S...	Microsoft Corporation
shot.exe	4,752 K	14,216 K	3400	3400	Shell Infrastructure Host	Microsoft Corporation
taskhostw.exe	5,528 K	9,240 K	3496	3496	Host Process for Windows T...	Microsoft Corporation
MonikerDgUpdate...	1,692 K	3,552 K	3532	3532	Microsoft Edge Update	Microsoft Corporation
cmd.exe	1,844 K	292 K	2552	2552	Windows Command Processor	Microsoft Corporation

fakenet_logs_bob.exe[2].txt

First launched Process Monitor and Process Explorer to monitor system processes in real-time.

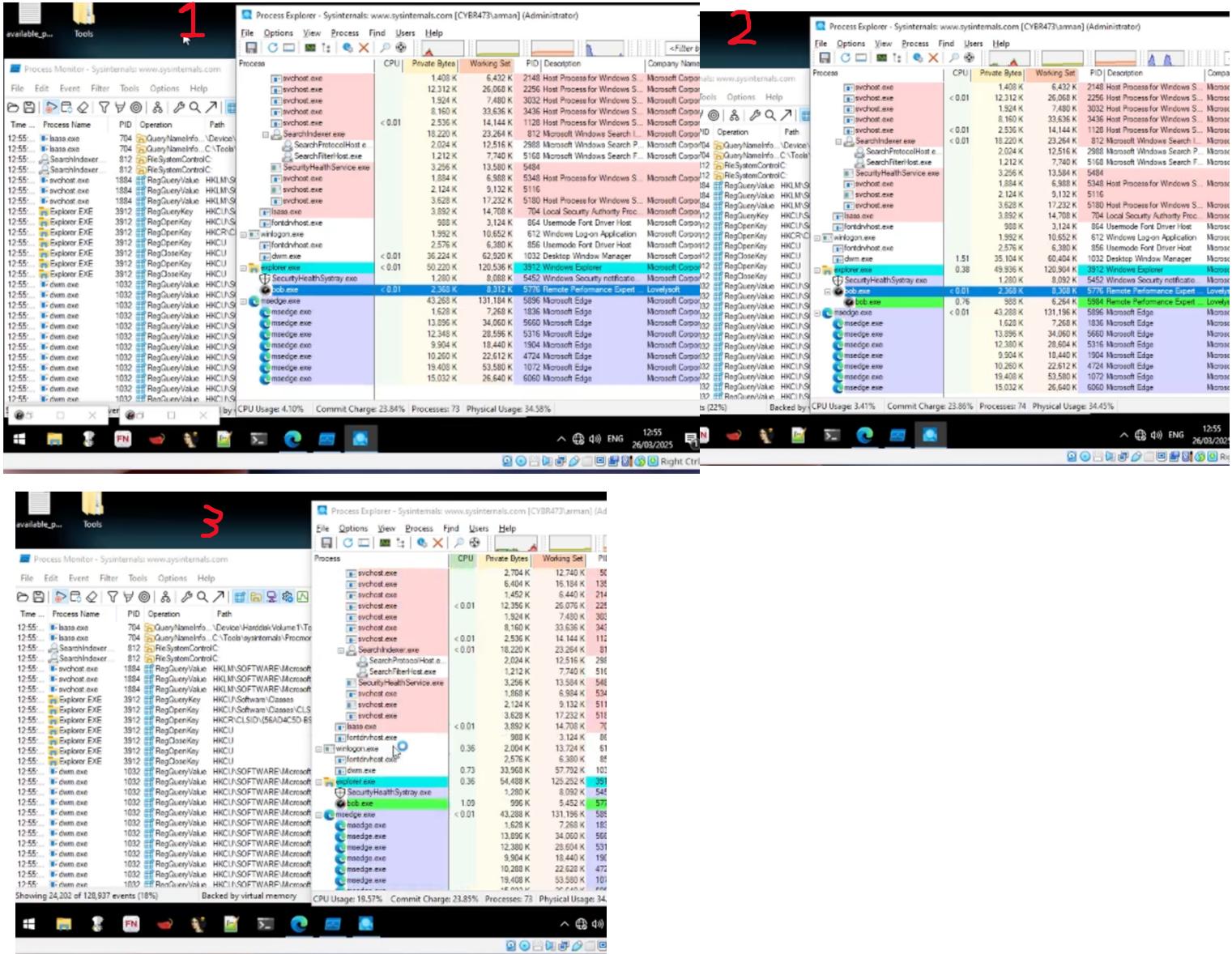


Setup a FakeDNS server to respond to DNS requests and Wireshark to listen to network traffic on enp0s8.

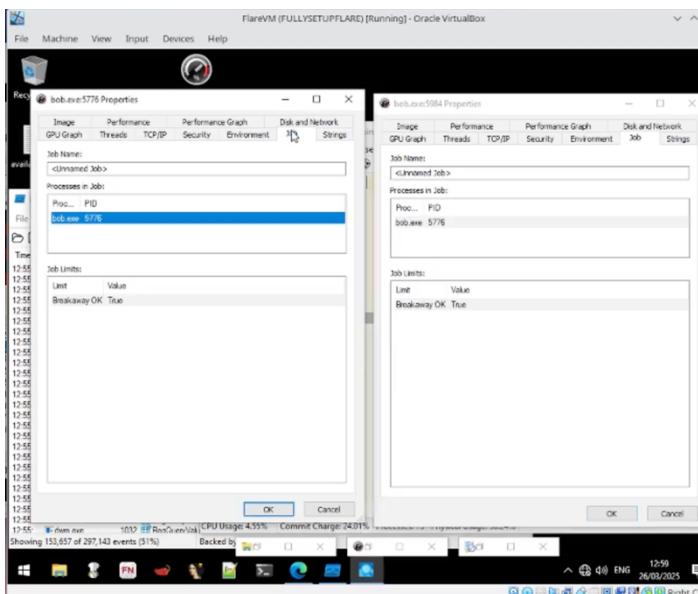
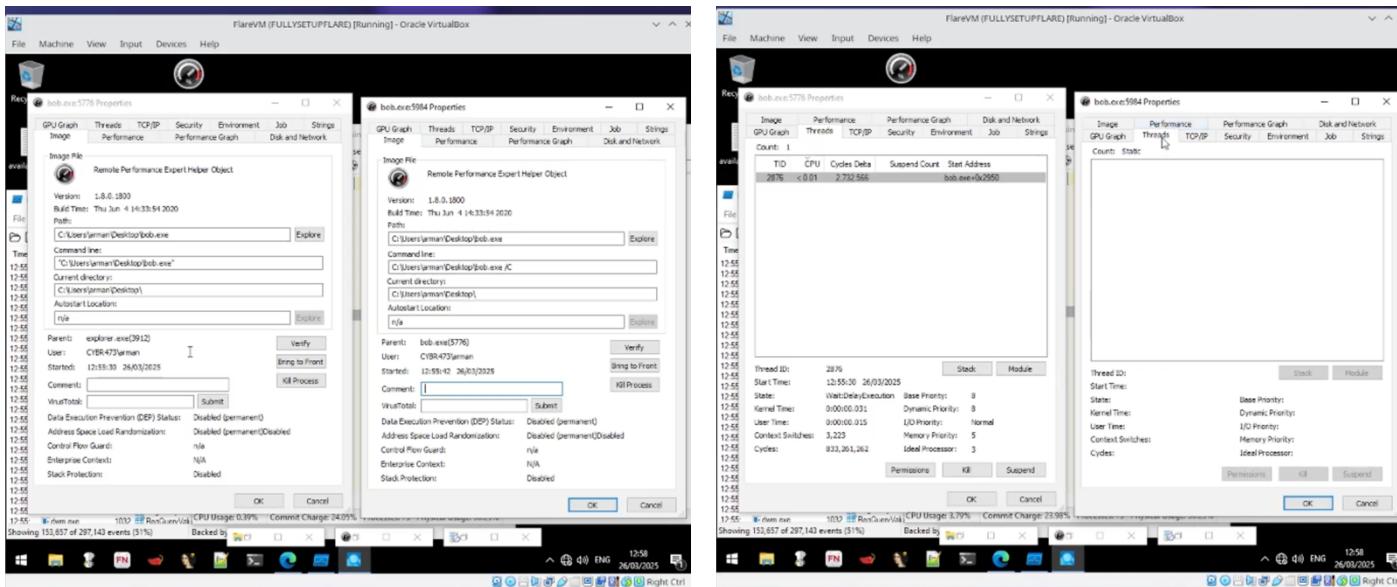


Executed bob.exe with administrative privilege.

Process Explorer



After executing bob.exe, bob.exe (PID 5576) process starts. Shortly after, a child process called bob.exe (PID 5984) is created. After that, the child process disappears, and the parent process (PID: 5776) is left. This behaviour could suggest process manipulation or injection to confuse analysis tools or evade detection.



Child process (5984) started with a command-line argument (bob.exe /C). Child process appears to execute its payload due to its short lifespan and absence of threads after termination. Also runs same job PID (5776) as parent, suggesting parent remains running inheriting the same job object as the child. May indicate process hollowing or parent-injection techniques, where child is carrying out malware actions, making the parent appear less suspicious.

Process Monitor

This screenshot shows the Process Monitor interface with the following details:

- File Path:** FlareVM (FULLYSETUPFLARE) [Running] - Oracle VirtualBox
- Selected Tab:** Process Monitor - Sysinternals: www.sysinternals.com
- Table Headers:** Time, Process Name, PID, Operation, Path, Result, Detail
- Log Data (Excerpt):**

Time	Process Name	PID	Operation	Path	Result	Detail
12.55...	bob.exe	5776	Process Start		SUCCESS	Parent PID: 3912, ...
12.55...	bob.exe	5776	Thread Create		SUCCESS	Thread ID: 2875
12.55...	bob.exe	5776	Load Image	C:\Windows\Prefetch\BOB.EXE-07379ECf	SUCCESS	Image Base: 0x400...
12.55...	bob.exe	5776	Load Image	C:\Windows\System32\vtdl.dll	SUCCESS	Image Base: 0x771...
- Bottom Status:** Showing 7,876 of 297,143 events (2.6%) Backed by virtual memory
- Control Flow Guard:** n/a
- Enterprise Context:** n/a

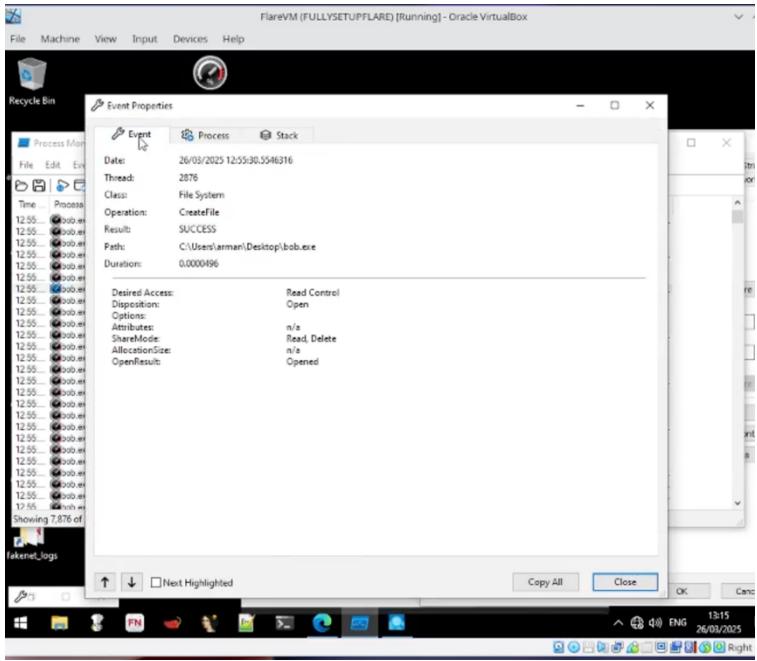
Start of parent process indicated by; Process Start, Thread Create, Load Image. I can see registry operations, RegOpenKey, RegQueryValue, and RegCloseKey, indicating system information gathering and possibly anti-analysis checks.

This screenshot shows the Process Monitor interface with the following details:

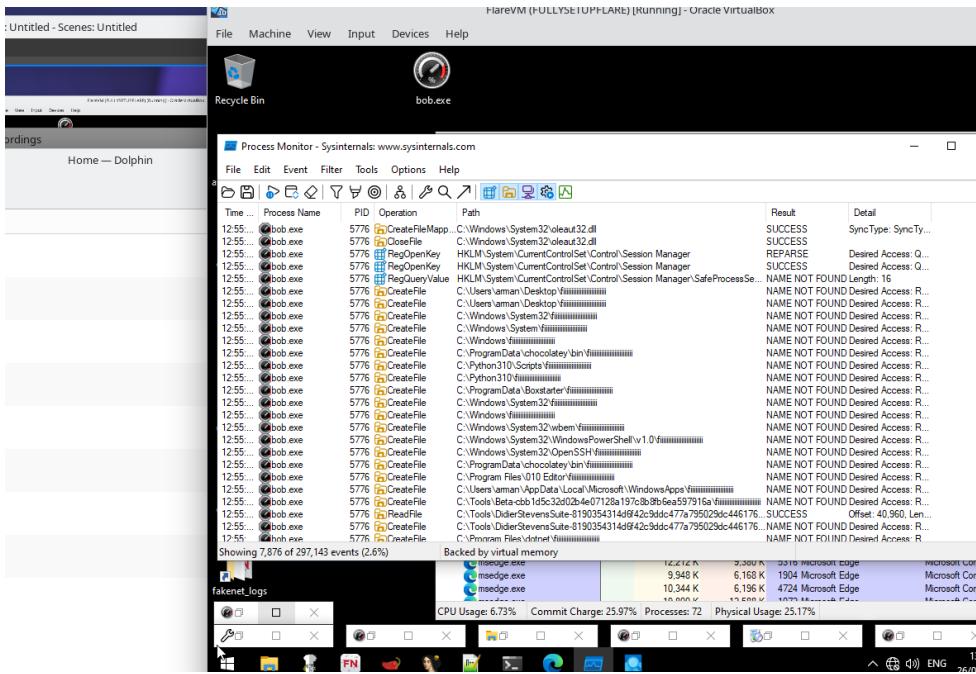
- File Path:** FlareVM (FULLYSETUPFLARE) [Running] - Oracle VirtualBox
- Selected Tab:** Process Monitor - Sysinternals: www.sysinternals.com
- Table Headers:** Time, Process Name, PID, Operation, Path, Result, Detail
- Log Data (Excerpt):**

Time	Process Name	PID	Operation	Path	Result	Detail
12.55...	bob.exe	5776	CreateFile	C:\Windows\System32\apcelp.dll	SUCCESS	Desired Access: R...
12.55...	bob.exe	5776	CreateFile	C:\Windows\System32\apcelp.dll	SUCCESS	FILE LOCKED W...
12.55...	bob.exe	5776	CreateFileMapping	C:\Windows\System32\apcelp.dll	SUCCESS	Sync Type: Sync Ty...
12.55...	bob.exe	5776	CreateFileMapping	C:\Windows\System32\apcelp.dll	SUCCESS	Sync Type: Sync Ty...
12.55...	bob.exe	5776	Load Image	C:\Windows\System32\apcelp.dll	SUCCESS	Image Base: 0x731...
12.55...	bob.exe	5776	CreateFile	C:\Windows\System32\apcelp.dll	SUCCESS	Desired Access: R...
12.55...	bob.exe	5776	RegOpenKey	HKEY_LOCAL_MACHINE\Software\Microsoft\Windows NT\CurrentVersion\AppCompatFlags	SUCCESS	Desired Access: Q...
12.55...	bob.exe	5776	RegQueryValue	HKEY_LOCAL_MACHINE\Software\Microsoft\Windows NT\CurrentVersion\AppCompatFlags	SUCCESS	NAME NOT FOUND Length: 20
12.55...	bob.exe	5776	RegCloseKey	HKEY_LOCAL_MACHINE\Software\Microsoft\Windows NT\CurrentVersion\AppCompatFlags	SUCCESS	Desired Access: Q...
12.55...	bob.exe	5776	CreateFile	C:\Users\leman\Desktop\bob.exe	SUCCESS	Desired Access: R...
12.55...	bob.exe	5776	WriteFile	C:\Users\leman\Desktop\bob.exe	BUFFER OVERFL...	Information: Owner
12.55...	bob.exe	5776	CreateFile	C:\Users\leman\Desktop\bob.exe	SUCCESS	Information: Owner
12.55...	bob.exe	5776	CreateFile	C:\Windows\System32\vtdl.dll	SUCCESS	Desired Access: R...
12.55...	bob.exe	5776	WriteFile	C:\Windows\System32\vtdl.dll	BUFFER OVERFL...	Information: Owner
12.55...	bob.exe	5776	CreateFile	C:\Windows\System32\vtdl.dll	SUCCESS	Information: Owner
12.55...	bob.exe	5776	CreateFile	C:\Windows\System32\kernel32.dll	SUCCESS	Desired Access: R...
12.55...	bob.exe	5776	WriteFile	C:\Windows\System32\kernel32.dll	BUFFER OVERFL...	Information: Owner
12.55...	bob.exe	5776	CreateFile	C:\Windows\System32\kernel32.dll	SUCCESS	Information: Owner
12.55...	bob.exe	5776	WriteFile	C:\Windows\System32\kernel32.dll	BUFFER OVERFL...	Information: Owner
12.55...	bob.exe	5776	CreateFile	C:\Windows\System32\kernel32.dll	SUCCESS	Desired Access: R...
12.55...	bob.exe	5776	QuerySecurityFile	C:\Windows\System32\kernel32.dll	BUFFER OVERFL...	Information: Owner
12.55...	bob.exe	5776	CloseFile	C:\Windows\System32\kernel32.dll	SUCCESS	Information: Owner
- Bottom Status:** Showing 7,876 of 297,143 events (2.6%) Backed by virtual memory
- Control Flow Guard:** n/a
- Enterprise Context:** n/a
- Stack Protection:** Disabled

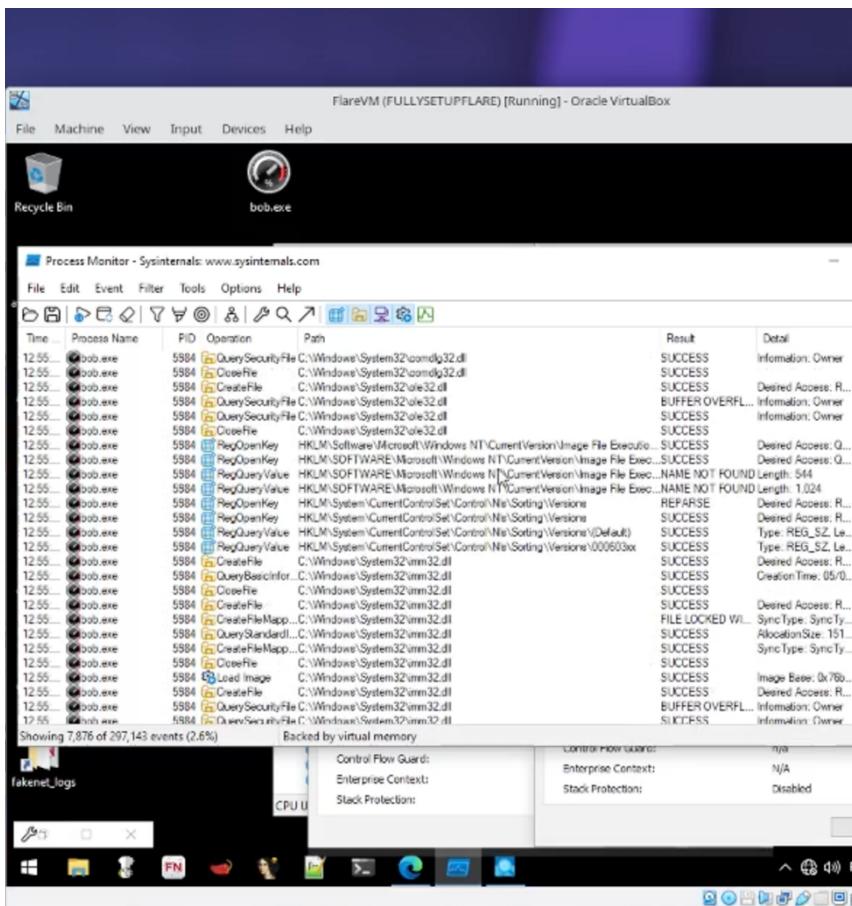
Can see file operations, CreateFile, WriteFile, and DeleteFile, indicating malware interacting with the file system, creating or modifying files and possibly cleaning up to avoid detection.



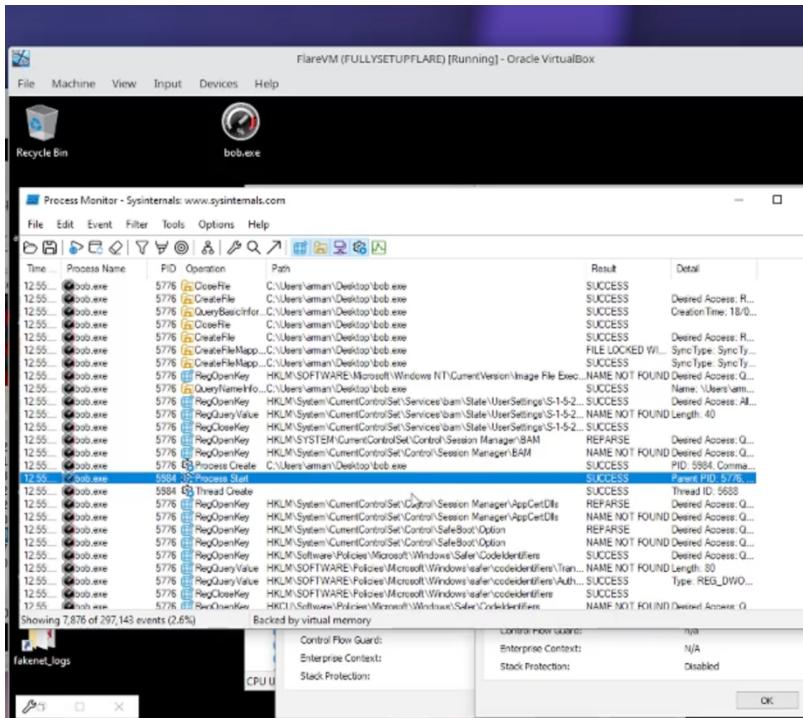
Can see parent process (PID: 5776) create child process (PID: 5984).



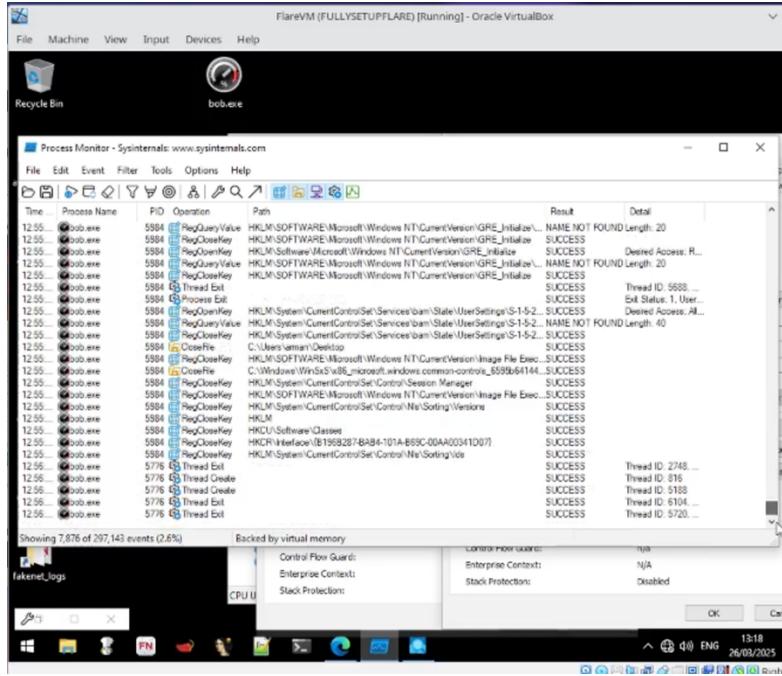
Can see CreateFile process, containing the repetitive string identified earlier, "fiiiiiiiiii". Suggests malware may use this as markers or dummy filenames as a possible anti-analysis technique to check existence of sandbox-specific directories. NAME NOT FOUND indicates the malware is scanning the environment before proceeding.



Parent process (PID: 5776) finishes off making/creating child bob.exe. The child process (PID: 5984) starts its process.

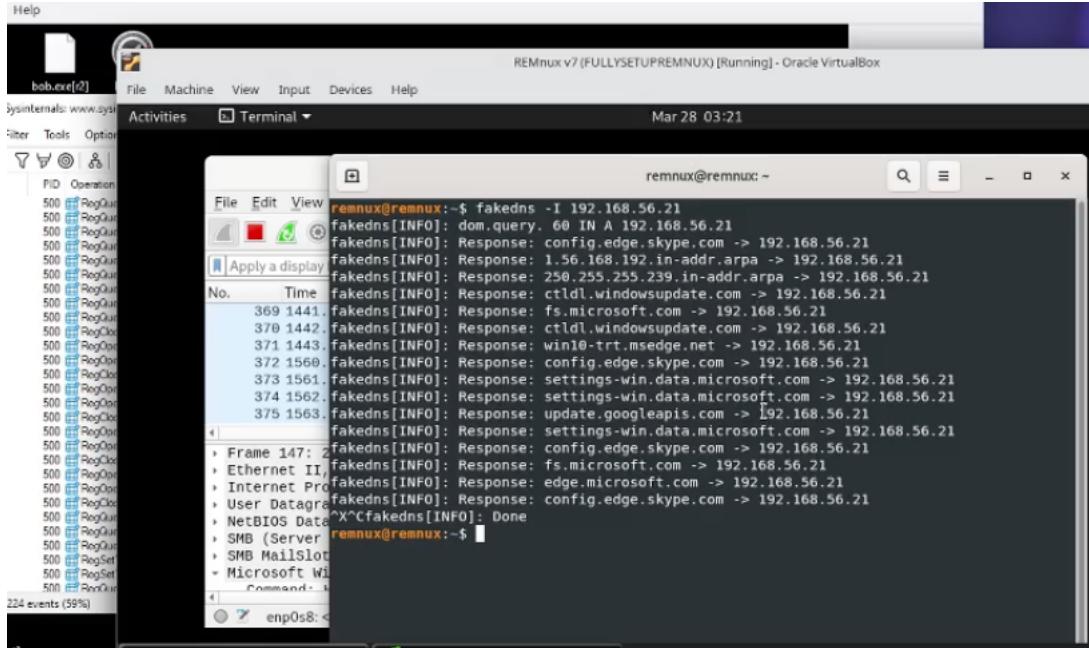


The child process performs similar actions with registry (RegOpenKey, RegQueryValue) and file processes (CreateFile, CloseFile).



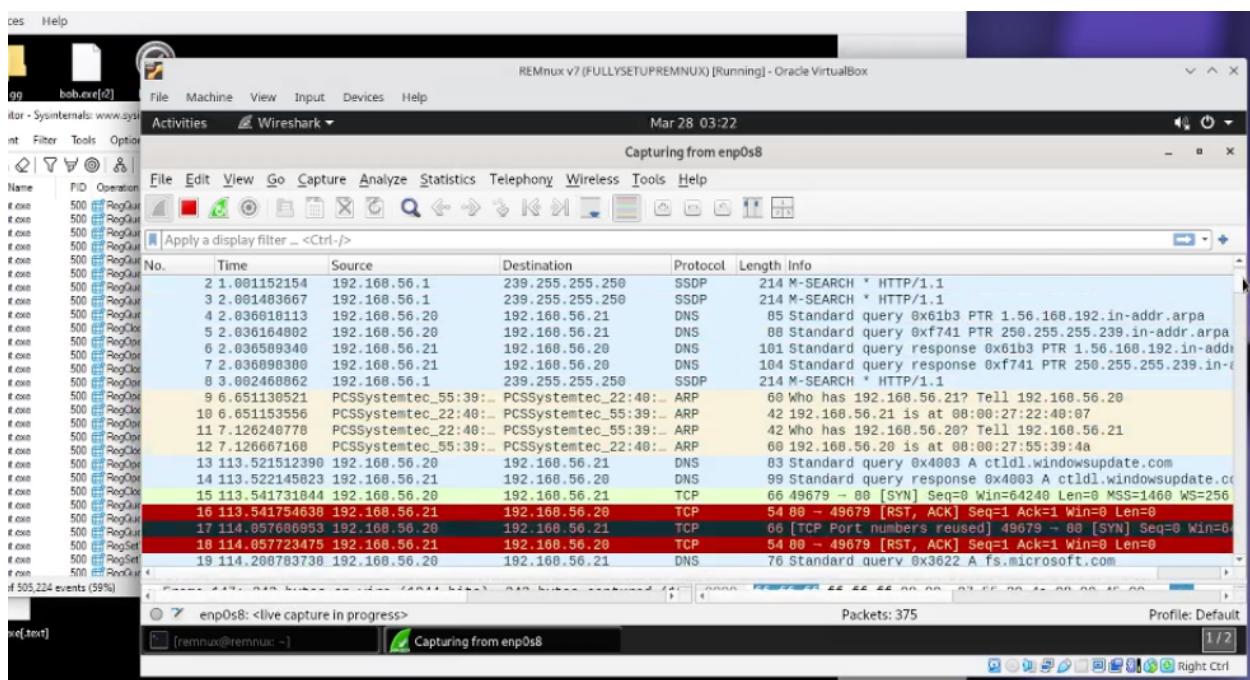
Child process (PID: 5984) performs majority of the actions in the malware. Can see the child's process finishing and exiting its process (Process Exit). After, we can see the parent process finishing its thread as well.

FakeDNS



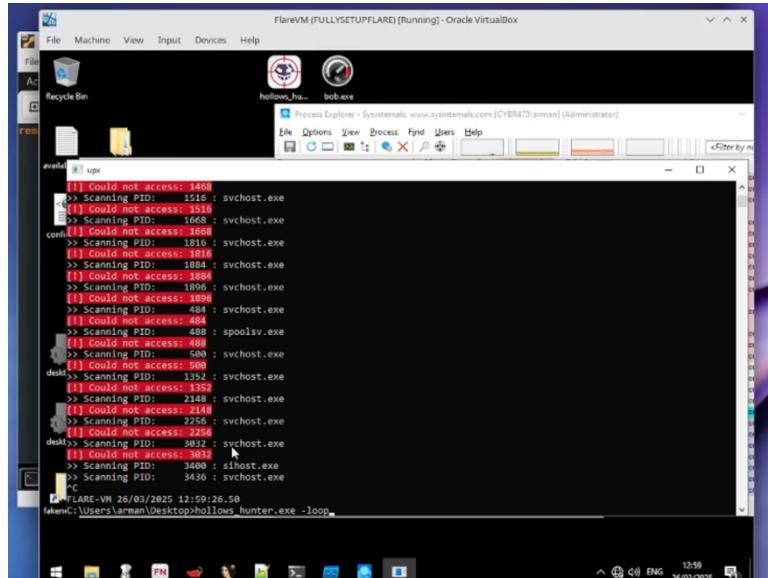
I can see FakeDNS making DNS requests for FlareVM and possibly the malware. Domains are being queried; however, no prior evidence was identified on these requests being made by the malware.

Wireshark

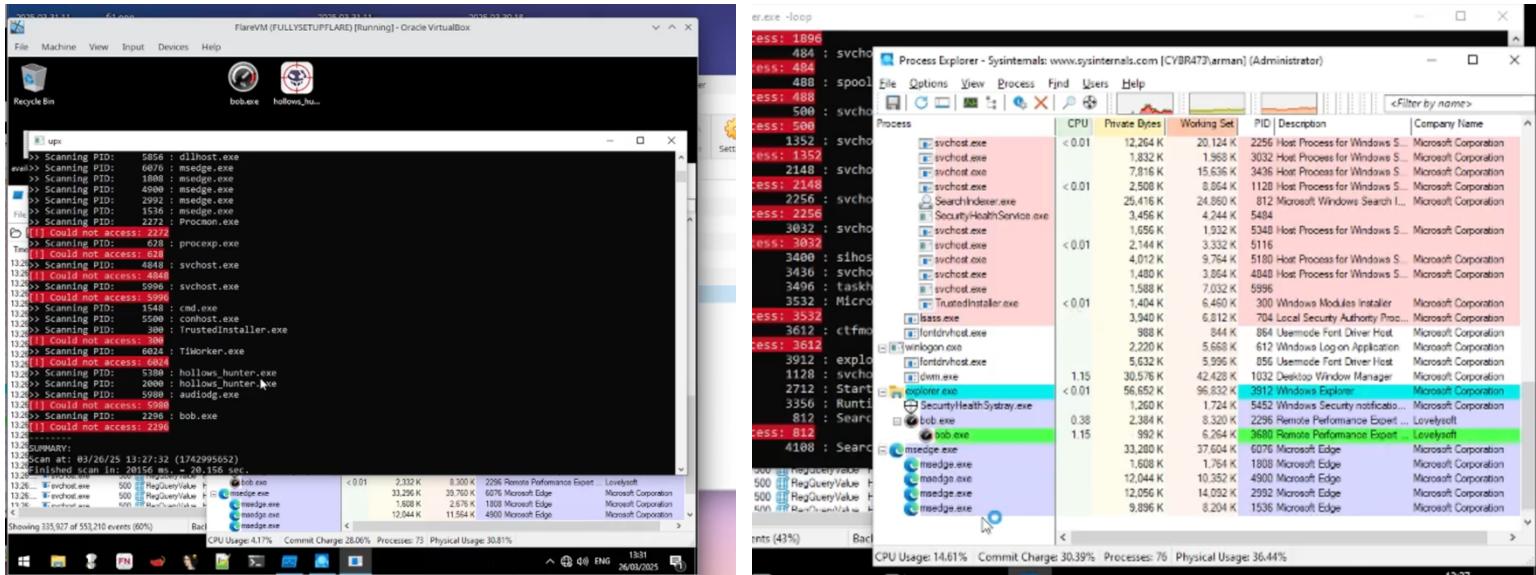


A successful connection didn't appear after the DNS request, as no follow-up HTTP request or data transfer was visible. Unclear whether request was from normal processes or malware due to lack of network-based identifiers analyzing the malware. There are limitations regarding observing/understanding network activities and malware processes using only basic dynamic analysis techniques. Applying advanced dynamic analysis techniques such as OllyDbg to observe TLS connections would help understand malware processes further.

Hollows Hunter



Ran hollows hunter on a loop, as there is no indication when malware has finished running, using command line: hollows-Hunter.exe -loop.



Hollows Hunter failed to access parent process (PID: 2296) and pick up child process (PID: 3680), suggesting the malware could possibly be using evasion techniques like handle manipulation to avoid detection during process scanning.

4. Lab Submissions

- Basic Static Analysis:
<https://vstream.au.panopto.com/Panopto/Pages/Viewer.aspx?id=adf7b2a9-e11a-4338-98cb-b2b101697306>
- Basic Dynamic Analysis:
<https://vstream.au.panopto.com/Panopto/Pages/Viewer.aspx?id=a4755c3c-3d6e-4b02-bc04-b2b1016d133d>

5. Word Count Clarification

Word count is 2982 for sections 1 – 3, which is the main report on the malware, excluding title and sections 4 – 6.

6. References and Tools

6.1. Tools Used For Static Analysis

- VirusTotal (<https://www.virustotal.com>)
- CFF Explorer (<https://cff-explorer.com>)
- PEiD (<https://github.com/wolfram77web/app-peid>)
- Detect it Easy (<https://github.com/horsicq/Detect-It-Easy>)
- 7-Zip (<https://www.7-zip.org>)
- PE-Bear (<https://github.com/hasherezade/pe-bear>)

- Dependency Walker (<https://www.dependencywalker.com>)
- Notepad (<https://apps.microsoft.com/detail/9msmlrh6lzf3?hl=en-US&gl=US>)

6.2. Tools Used For Dynamic Analysis

- Process Explorer (<https://learn.microsoft.com/en-us/sysinternals/downloads/process-explorer>)
- Process Monitor (<https://learn.microsoft.com/en-us/sysinternals/downloads/procmon>)
- FakeDNS (<https://github.com/SocialExploits/fakedns/tree/main>)
- Wireshark (<https://www.wireshark.org>)
- Hollows Hunter (https://github.com/hasherezade/hollows_hunter)

6.3. References/Websites

- ChatGPT (Helped in understanding and clarification of malware analysis behaviour, processes and patterns)
- https://www.youtube.com/watch?v=1Kb6tee2eJl&ab_channel=screeck (Used to help build lab environment setup for live malware analysis).
- <https://www.cisa.gov/news-events/alerts/2017/05/12/indicators-associated-wannacry-ransomware> (Used to understand malware HTTP kill chain process)
- <https://learn.microsoft.com/en-us/windows/win32/dlls/dynamic-link-libraries> (Microsoft documentation to understand processes and DLL purposes)