A Deep Dive into Medusa

Ransomware

Prepared by: Vlad Pasca, Senior Malware &

Threat Analyst



SecurityScorecard.com info@securityscorecard.com

Tower 49 12 E 49th Street Suite 15-001 New York, NY 10017 1.800.682.1707

Table of contents

Table of contents	1
Executive summary	2
Analysis and findings	2
Indicators of Compromise	18
Appendix	19

Executive summary

Medusa ransomware <u>appeared</u> in June 2021, and it became more active this year by launching the "Medusa Blog" containing data leaked from victims that didn't pay the ransom. The malware stops a list of services and processes decrypted at runtime and deletes the Volume Shadow Copies.

The files are encrypted using the AES256 algorithm, with the key being encrypted using an RSA public key. The ransomware deletes itself after the file encryption is complete. The extension of the encrypted files is changed to ".MEDUSA".

Analysis and findings

We will analyze a ransomware sample that our Professional Services team found in a Medusa Ransomware engagement. We can't share the malware hash to protect the client's confidentiality.

The ransomware can run with one of the following parameters: "-d", "-f", "-i", "-k", "-n", "-p", "-s", "-t", "-v", "-w", and "-V". If the "-v" parameter is not specified, the process calls the ShowWindow function to hide the current window (0x0 = **SW_HIDE**):

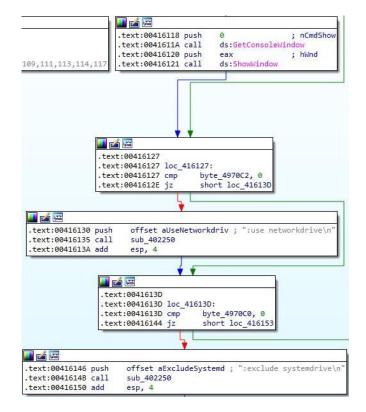


Figure 1

When running with the "-v" parameter, the malware displays multiple strings in the command line window using WriteFile, as shown in Figure 2.

EIP	0080560 0080560 0080560 0080561 0080561 0080561 0080561 <	8 50 9 56 A 8D85 0 50 1 57 2 FF15	FCEBFFFF	push ea	d,dword ptr s			eax:"start\r\n" eax:"start\r\n"	<pre>Cefault (stdcall) 1: [esp] 000000A 2: [esp+4] 00754 3: [esp+4] 00754 4: [esp+C] 00754 5: [esp+10] 0000</pre>	188 "start\r\n" 008 184
	ds:[00B19208			<kernel32.w< th=""><th>riteFile></th><th>[x=] Locals</th><th>Struct</th><th></th><th>00754160 000000A 00754164 0075418</th><th></th></kernel32.w<>	riteFile>	[x=] Locals	Struct		00754160 000000A 00754164 0075418	
Address				00 00 00 00	ASCII start	1		^	00754164 0075418 00754168 0000000 0075416C 0075418 00754170 0000000	B 4



The malicious process creates multiple anonymous pipes via a function call to CreatePipe:

Address He			ASCII	11	1	00754A94 00754AF8 00754A98 00000000
Dump 1	💭 Dump 2 🛛	Dump 3 🛛 🛄 Dump 4	💷 Dump 5 🛛 🧶 Watch 1	🛛 🖉 Struc	t	00754A8C 00754B0C 00754A90 00754B08
(3))	ls:[00B19018 <ma 190A malware.exe</ma 		<kernel32.createpipe></kernel32.createpipe>			
	• <	1113 10308100	carr anora per dost	der eucer (pez)	>	5: [esp+10] 51B05D78
TP	• 00AB4900 • 00AB490A	C785 8CEFFFFF 000 FF15 1890B100	000 mov dword ptr ss: call dword ptr ds:[op-1074],0		3: [esp+8] 00754AF8 4: [esp+C] 00000000
	 00AB48EC 00AB48F6 		000 mov dword ptr ss: e 000 mov dword ptr ss: e		c:'\f'	1: [esp] 00754B0C 2: [esp+4] 00754B08
	 00AB48D8 00AB48E2 	C785 98EFFFFF 000	0000 mov dword ptr ss: e 0000 mov dword ptr ss: e	pp-1068 .0		Default (stdcall) 🔻 5 🔹 🗌 Unlo
	00AB48D7	50	push eax			<
	 00AB48C7 00AB48CD 	8D85 9CEFFFFF	lea eax, dword ptr s	: [ebp-1064]		x87TW_4 3 (Empty) x87TW_5 3 (Empty) x87TW 6 3 (Empty) x87TW 7 3 (Empty)
	 00AB48BC 00AB48C6 	C785 A0EFFFFF 000	0000 mov dword ptr ss:[e	pp-1060],0		x87TW_0 3 (Empty) x87TW_1 3 (Empty) x87TW_2 3 (Empty) x87TW_3 3 (Empty)
	 00AB48B5 00AB48B6 	50 8D85 98EFFFFF	lea eax, dword ptr s	: ebp-1068		x87TagWord FFFF
	00AB48AE	C745 FC 00000000	mov dword ptr ss:[e			ST(7) 0000000000000000 x87r7 Empty
	 00AB48A6 00AB48A8 	6A 00 8D85 88EFFFFF	push 0 lea eax,dword ptr s	Tohn 1070		ST(6) 00000000000000000 x87r6 Empty

Figure 3

The SetHandleInformation routine is used to make the pipes inheritable by child processes (0x1 = HANDLE_FLAG_INHERIT):

EIR	00AB491 00AB492 00AB492 00AB492 00AB492	0 6A 01 2 FFB5 8 FFD6		push 0 push 1 push du call es	word ptr ss:	[ebp-1064]		 1: [esp] 0000013C 2: [esp+4] 00000001 3: [esp+8] 00000000 4: [esp+C] 51805D78 5: [esp+10] 00000000
		eInformatio						007554200 0000013C
Dump 1	Dump 2	Dump 3	Dump 4	Dump 5	💮 Watch 1	[x=] Locals	Struct	00754A94 00000001
Address He	v				ASCTT	1		00754A98 00000000

Figure 4

The malware creates a PowerShell process using the CreateProcessA API (0x08000000 = **CREATE_NO_WINDOW**):

319	00A8499B 00A849A5 00A849A5 00A849A6 00A849AF 00A849B2 00A849B2 00A849B2 00A849B2 00A849S2 000	8180 6CEFFFFF 000 50 6A 00 6A 00 6A 00 6A 00 6A 00 FFB5 3CEFFFFF 6A 00 6C:0F1385 44EFFFF 6C:0F1385 44EFFF 6C:0F1385 44EFF 6C:0F1385 44EFFF 6C:0F1385 44EFFF 6C:0F1385 44EFFF 6C:0F1385 44EFFF 6C:0F1385 44EFFF 6C:0F1385 44EFFF 6C:0F1385 44EFFF 6C:0F1385 44EFF 6C:0F1385 44EFF 6C:0F1385 44EFFF 6C:0F1385 44EFFF 6C:0F1385 44EFFF 6C:0F1385 44EFFF 6C:0F1385 44EFFF 7C:0F1385 44EFF 7C:0F1385 44EFF 7C:0F1385 44EFFF 7C:0F1385 44EFFF 7C:0F1385 44EFFF 7C:0F1385 44EFFF 7C:0F1385 44EFFF 7C:0F1385 44EFFF 7C:0F1385 44EFFF 7C:0F1385 44EFFF 7C:0F1385 44EFFF 7C:0F1385 44EFFF 7C	push eax push 0 push 0 push 0 push 1 push 1 push 0 push dword push dword 1 movups Xmw FF mov1pd qwor. FF mov1pd qwor. FF mov1pd qword p fF mov1pd qword 2	ptr ss: ebp-10C4 ord ptr ss: ebp-10G d ptr ss: ebp-10G d ptr ss: ebp-10A d ptr ss: ebp-10A d ptr ss: ebp-103 d ptr ss: ebp-103 d ptr ss: ebp-103	05 8], xmm0 30], xmm0 34], xmm0 45], xmm0 44], xmm0 35], xmm0 35], xmm0 44	[ebp-10C4]:"powershe	st(4) 00000000 st(5) 000000000 st(6) 000000000 st(7) 000000000 st(7) 000000000 st(7) 000000000 st(7) 000000000 st(7) 000000000 st(7) 00000000 st(7) 00000000 st(7) 000000000 st(7) 0000000000 st(7) 000000000000000000000000000000000000) x87TW_3 3 (Empty)) x87TW_5 3 (Empty)) x87TW 7 3 (Empty) ▼ 5 0 Univ 3FO "powershell -Comman 300
	Is:[00B1903C <	nalware.&CreateProces	sA>]= <kerne132.c< th=""><th>reateProcessA></th><th></th><th></th><th></th><th></th></kerne132.c<>	reateProcessA>				
ana ana ang ang ang ang ang ang ang ang	9FE malware.e	<pre>ke:\$149FE #13DFE</pre>			100			
ana ana ang ang ang ang ang ang ang ang	Dump 2	ke:\$149FE #13DFE	📖 Dump 5 🛛 🤴	Watch 1 [x=] Locals	Struct		00754A74 0000000 00754A78 00755BF0 00754A7C 0000000) "powershell -Command

The malicious process reads data from the pipe containing the above process output using ReadFile:

	<pre>OOAB4A3E OOAB4A3E OOAB4A3E OOAB4A3E FFD5 AOEFFFFF Call call call call call call call call</pre>					ss:[ebp-101 [ebp-1060]		>	<pre>c Default (stdcal) 1: [esp] 00000144 2: [esp+4] 00754858 3: [esp+8] 0001000 4: [esp+c] 00754814 5: [esp+10] 00000000</pre>	> ▼ 5 호 Unloci
Ump 1	Dump 2	Ump 3	Dump 4	Ump 5	👹 Watch 1	[x=] Locals	Struct		00754A88 00000144 00754A8C 00754B58	
Address He	ex	00.00.00	00.00.00.00	00.00.00.00	ASCII		for the first	^	00754A90 00001000 00754A94 00754B14	

Figure 6

The sample retrieves the firmware table from the raw SMBIOS firmware table provider using the GetSystemFirmwareTable routine (see Figure 7).

	 00AB5317 00AB5319 			push 0 push 0				1:	[esp] 52			
IP	00AB5310	68 42	4D5 35 2	push 0 push 52 call ec				 2: 3: 4: 5:	[esp+4] [esp+8] [esp+C]	00000000 00000000 00000000 51805D9)	
	32.GetSysten			Dump 5	👹 Watch 1	[x=] Locals	Struct		755754 52 755758 00			

Figure 7

CryptStringToBinaryA is used to decode the RSA public key from Base64 (0x7 = **CRYPT_STRING_ANY**):

edi= <crypt32.cryptstringtobinarya> .text:00AAIA0D ma]ware.exe:\$1A0D #E0D</crypt32.cryptstringtobinarya>	1 1ware.8390F8	8390F8:"BEGIN F	Control Co
Image Image <th< th=""><th>UBLIC (EYM IIICgKC4QEA0a20A XX24H8GDpRkytnaD F)Ft0xp01epM0vV 09e556445K4RRHwQ +Z0dEzn0AJ0efGI 0dkwE1spt1bfUW00 A3WWc4UMW48GTF35 S10RN48GTF35 S1</th><th>^</th><th>00754820 008390F8 malware.008390F8 00754824 0000000 00754828 0000007 00754826 0091A340 00754826 0091A340 00754830 00754868 00754838 0000000 00754838 0000000 00754836 0000000 00754840 00915504 00754846 00915504 00754850 00915504 00754858 00919888 00754856 00919888 00754868 0091000 00754868 0091987 00754868 00919888 00754868 0091000 00754868 00919888 00754868 00910000 00754868 00919888 00754869 00919888 00754868 00910000 00754868 00919888 00754869 00754924 00754870 00754924</th></th<>	UBLIC (EYM IIICgKC4QEA0a20A XX24H8GDpRkytnaD F)Ft0xp01epM0vV 09e556445K4RRHwQ +Z0dEzn0AJ0efGI 0dkwE1spt1bfUW00 A3WWc4UMW48GTF35 S10RN48GTF35 S1	^	00754820 008390F8 malware.008390F8 00754824 0000000 00754828 0000007 00754826 0091A340 00754826 0091A340 00754830 00754868 00754838 0000000 00754838 0000000 00754836 0000000 00754840 00915504 00754846 00915504 00754850 00915504 00754858 00919888 00754856 00919888 00754868 0091000 00754868 0091987 00754868 00919888 00754868 0091000 00754868 00919888 00754868 00910000 00754868 00919888 00754869 00919888 00754868 00910000 00754868 00919888 00754869 00754924 00754870 00754924

Medusa ransomware decodes a structure of the **RSA_CSP_PUBLICKEYBLOB** type by calling the CryptDecodeObjectEx function (0x10001 = **X509_ASN_ENCODING** | **PKCS_7_ASN_ENCODING**, 0x13 = **RSA_CSP_PUBLICKEYBLOB**):

OQAAIA81 GA 00 push 0 OQAAIA83 GA 00 push 0 push 0	ord ptr ss:[ebp-1020]	x87TW 6 3 (Fmpty) x87TW 7 3 (Fmpty) x87TW 7 3 (Fmpty) x87TW 7 3 (Fmpty) x87TW 7 3 (Fmpty) > Default (stdcall) 1: [esp1 00010001 2: [esp+4] 0001001 3: [esp+4] 0001001 4: [esp-C] 0000101 5: [esp+10] 00000000
Image Image <th< th=""><th>MIUU-a .xx5.1a w1.we)loolsec.#s *.5ito.uAo.Adusc 0(co.gP.woE w3.2ce5.xmm" BAAXO10.0.W.r tpFV_SEP *0.fe+OW.w1,4.A} p.#,5:1.0y.oI ueyGD_TiK</th><th>0075481C 00010001 00754820 00000013 00754820 000010E 00754820 000010E 00754820 0000000 00754820 0000000 00754820 0000000 00754830 000754830 00754830 0000000 00754830 0000000 00754830 0000000 00754830 0000000 00754830 0000000 00754834 0031504 00754834 0031554 00754835 00315564 00754836 0031880 00754836 0031888 00754856 0031888 00754856 0031888 00754856 000314 00754868 000014 00754868 00075485</th></th<>	MIUU-a .xx5.1a w1.we)loolsec.#s *.5ito.uAo.Adusc 0(co.gP.woE w3.2ce5.xmm" BAAXO10.0.W.r tpFV_SEP *0.fe+OW.w1,4.A} p.#,5:1.0y.oI ueyGD_TiK	0075481C 00010001 00754820 00000013 00754820 000010E 00754820 000010E 00754820 0000000 00754820 0000000 00754820 0000000 00754830 000754830 00754830 0000000 00754830 0000000 00754830 0000000 00754830 0000000 00754830 0000000 00754834 0031504 00754834 0031554 00754835 00315564 00754836 0031880 00754836 0031888 00754856 0031888 00754856 0031888 00754856 000314 00754868 000014 00754868 00075485

Figure 9

The process imports the RSA public key from a key BLOB using BCryptImportKeyPair:

	 00AA1BAE 00AA1BB0 00AA1BB6 00AA1BB7 	6A 00 FFB5 CCEFFF 57 FFB5 C8EFFF	push e	word ptr ss:[ek di word ptr ss:[ek			edi:"RSA1"	Contraction of the second seco		×8/1W / 3 (EmpTV) ▼ 5 🖨 🗌 Unlo
	00AA18BD 00AA18C2 00AA18C4 00AA18C7 <	68 <u>C077B200</u> 6A 00 FF70 1C FF15 <u>3892B1</u> alware.&BCrypt	push m push 0 push d	alware.B277CO word ptr ds:[ea word ptr ds:[<&	ax+1C] &BCryptImp	portKeyPair>]	3277CO:L"RSAPUBLI	3: [esp+8] 4: [esp+C]	0000000 008277C 009155F	0 L"RSAPUBLICBLOB"
Dump 1	Dump 2	Dump 3	ump 4 💭 Dump 5	👹 Watch 1	[x=] Locals	Struct		00754B20 0 00754B24 0	00000000	
091AEB0 00 091AEC0 F3	53 41 31 00 08 00 00 00 00 00 E2 1F 06 0E 94	0 00 00 01 00 4 64 CA D9 DA	00 00 00 01 00 00 01 D1 AD A8 03 15 0C 52 62 3E DD 31)>Ý1		^	00754B28 0 00754B2C 0 00754B30 0 00754B34 0 00754B38 0	009155F4 0091AEA0 0000011B	malware.00B277C0 "RSA1"
0091AEE0 11 0091AEF0 64 0091AF00 31 0091AF10 23 0091AF20 30 0091AF20 40 0091AF40 F6 0091AF40 F6 0091AF50 A4 0091AF50 78 0091AF90 F0 0091AF90 F0	F0 AB EC C3 7- C0 4D 6C A6 D2 61 FC 19 31 7- 73 B2 86 24 22 43 93 8A F5 00 D2 91 98 D5 77 CA 57 20 08 32 22 20 42 C4 47 72 2C 9A 94 92 13 B2 D6 60 84 97 CD FA 93 60 8- 07 DD DF EF A8	4 4C E7 D0 0A 58 7E E5 A8 7 E9 29 5D 44 1 1 74 30 14 75 C 6 D2 3F 04 8 4F F4 1A 10 8 8B 5A 09 06 7 58 D3 AD 0D 2 12 62 FD 46 0 C8 28 F0 57 1 23 88 73 AC 4 A2 79 36 FT C2	CE B& E3 99 38 45 89 39 E7 C6 22 87 A0 0D EF 58 A7 94 DB EC 33 45 C7 06 C1 30 9D C2 F5 F5 C3 A3 2A DC FC 88 67 91 0C 50 1E B& 15 40 24 92 B& 6C 31 D4 16 D5 AD 57 56 5F 24 E8 50 82 86 B8 14 A5 5E C0 A1 97 0E F2 FF 97 57 3F EF A1 48 05 81 D2 9A DD 41 CF A3 AB AB AB AB AB AB	. ð«ìAtL¢D. 9¢ dAM1;Ú[~à. i 1 aŭ.iwė)]DØi3) #s ² .\$Ito.uAo. 9 c. ô.40?.Af ² 8 eO.0{0ô.g. 0ÈW .3.Ze\$ #" BAGXÓ.1Ô. 2.rbýFV_\$ 5.°O.£+ØW.3] A}.P#s ³ . 0ÌL¢S67.?i	4". ×ς. Αδύ Ου. Ρ.» Ρ.» δ.»m δ.w δ.w δ.v. λ4. δy. λ4. δy. μΑΪ			0075483C (00754844 (00754844 (00754848 (00754848 (00754848 (0075485 (0075488 (0075488 (0075486 (0075486 (0075486 (0075486 (0075486 (0075486 (007548 (007548 (007548 (007548 (007548 (00754 (007754 (00774 (00774 (0077	00000000 009155D4 009155D4 009155D4 009155F4 0000118 0091AEB8 0091AEA0 0091AEA0 0091AD70 00000114 0000010E 00000206	"RSA1"

A list of file extensions that will be skipped is decrypted using the XOR operation with 0x2E: ".dll", ".exe", ".lnk", and ".MEDUSA". The ransomware also decrypts a list of services and processes that will be stopped:

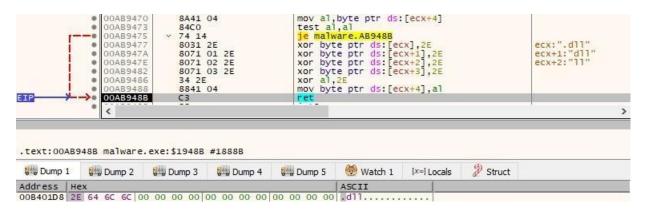


Figure 11

	<			
	00AB943A	C3	ret	
i		72 F6	jb malware.AB9430	
	00AB9435	83F8 15	cmp eax,15	
	00AB9434	40	inc eax	
	00AB9430	803408 2E	xor byte ptr ds:[eax+ecx],2E	
	00AB9425	0F1F40_00	nop dword ptr ds:[eax].eax	ECX. ACTONTS VS
	00AB9425 00AB9429	66:0FEFC8 0F1109	<pre>pxor xmm1, xmm0 movups xmmword ptr ds:[ecx], xmm1</pre>	ecx: "Acronis VS
	00AB941E	0F280D 2091B200	movaps xmm1, xmmword ptr ds: [B29120]	
	00AB9419	B8 10000000	mov eax,10	
	00AB9416	0F1001	movups xmm0, xmmword ptr ds:[ecx]	ecx: "Acronis VS
		74 24	je malware.AB943A	100
	00AB9410	8079 14 00	cmp byte ptr ds:[ecx+14],0	

.text:00AB943A malware.exe:\$1943A #1883A

Ump	1.5	-	Dur	mp 2			Dum	р 3	Ę	D 💭	ump	4	Ę	D	ump	5	🥘 Watch	n 1	[x=] L	ocals	Struct	
Address	Hex														ASCII							
00B4057C 00B4058C																						



The entire list of processes and services to terminate can be found in the Appendix.

The malware obtains the number of milliseconds that have elapsed since the system was started:

and		00AB63B4	8B1D 2C90B100	mov ebx,dword ptr ds:[<&GetTickCount64>]	
EIP		00AB63BA	FFD3	call ebx	1
		nosnenne l	68 84358300	anal galance parrow	Internet Hereiteren
		<			>
ebx= <ket< td=""><td>rnel32.0</td><td>GetTickCount</td><td>64></td><td></td><td></td></ket<>	rnel32.0	GetTickCount	64>		

Figure 13

The ransomware stops the target services using the "net stop" command and the target processes using the "taskkill" command:

ac→ dword_ptr_ds:[i	00A8498D 00A84993 00A84993 00A84993 00A849A5 00A849A5 00A849A5 00A849A8 00A849A8 00A849A8 00A849A8 00A84983 00A84983 00A84983 00A84983 00A84983 00A84985 00A8495 00A845 00A8	50 50 50 50 66:0F1385 6CEFFFFF 50 50 6A 00 6A 00 66:0F1385 44EFFFFF 66:0F1385 44EFFFFF 66:0F1385 54EFFFFFF 66:0F1385 54EFFFFF 66:0F1385 54EFFFFF 66:0F1385 54EFFFFF 66:0F1385 54EFFFFF 66:0F1385 54EFFFFF 66:0F1385 54EFFFFFF 66:0F1385 54EFFFFFF 66:0F1385 54EFFFFFF 66:0F1385 54EFFFFF 66:0F1385 54EFFFFF 66:0F1385 54EFFFFF 66:0F1385 54EFFFFF 66:0F1385 54EFFFFF 66:0F1385 54EFFFFF 66:0F1385 54EFFFFF 66:0F1385 54EFFFFF 66:0F1385 54EFFFFF 66:0F1385 54EFFFFF 66:0F1385 54EFFFFF 66:0F1385 54EFFFFF 66:0F1385 54EFFFFF 66:0F1385 54EFFFFF 66:0F1385 54EFFFFFF 66:0F1385 54EFFFFFFFFF 66:0F1385 54EFFFFFF 66:0F1385 67:0FFFF 66:0F1385 67:0FFFF 66:0F1385 67:0FFFF 66:0F1385 67:0FFFF 66:0F1385 67:0FFFF 66:0F1385 67:0FFFF 66:0F1385 67:0FFFFF 67:0FFFF 67:0FFFFF 67:0FFFFF 67:0FFFF 67:0FFFFF 67:0FFFF 67:0FFFFF 67:0FFFF 67:0FFFFF 67:0FFFF 67:0FFFF 67:0FFFF 67:0FFFF 67:0FFFF 67:0FFFF 67:0FFFF 67:0FFFF 67:0FFFF 67:0FFFF 67:0FFFFF 67:0FFFFF 67:0FFFFF 67:0FFFFF 67:0FFFFF 67:0FFFFF 67:0FFFFF 67:0FFFFFFFFFFFFF 67:0FFFFFFF 67:0FFFFFFFFFFFFFFF 67:0FFFFFFFFF	push eax push 0 push 0 push 5000000 push 1 push 0 push 0 push 0 push 0 movlpd qword ptr movlpd qword ptr movlpd qword ptr movlpd qword ptr movlpd qword ptr movlpd qword ptr sovlpd qword ptr sovlpd qword ptr sovlpd qword ptr social dword ptr social qword ptr social dword ptr social qword qword qword qtr social quoties qt social qt	s: [ebp-1094],10 [ebp-1094],10 r s: [ebp-1064] r s: [ebp-108 s: [ebp-108 s: [ebp-108 s: [ebp-108 s: [ebp-108 s: [ebp-108]] s: [ebp-108] [ebp-100], s: [ebp-108]]	05 9], xmm0 1, xmm0 1, xmm0 1, xmm0 1, xmm0 1, xmm0 1, xmm0 1, xmm0 1, xmm0 1, xmm0	[ebp-10C4]:"net s 44:'D'	<pre>ST(1) 00000000000 ST(2) 0000000000 ST(3) 00000000000 ST(3) 00000000000 ST(5) 400580000000 ST(7) 3FFFE x87Tw_0 3 (Empty) x87Tw_4 3 (Empty) x87Tw_4 3 (Empty) x87Tw_4 3 (Empty) x87Tw_4 3 (Empty) x87Tw_4 3 (Empty) x87Tw_6 3 (E</pre>)
	Dump 2 🛛 💭 Dui	mp 3 💭 Dump 4 🕻	📖 Dump 5 🛛 👹 Watch	1 [x=] Locals	Struct			"net stop \"Acronis VS!
Address Hex			ASCII			^	00754824 00000000 00754828 00000000	
00755988 6E 65 00755998 73 20 007559A8 2F 79	74 20 73 74 6	F 70 20 22 41 63 72 0 72 6F 76 69 64 65	72 22 20 S VSS Pr	ovider"			0075482C 00000001 00754830 08000000	

Figure 14

OOAB498C OOAB498D OOAB498D OOAB498D OOAB4993 OOAB4993 OOAB499A OOAB49A5 OOAB49A6 OOAB49A6 OOAB49A6 OOAB49A7 OOAB49A7 OOAB49A7 OOAB49A7 OOAB49A7 OOAB4982 OOAB4982 OOAB4982 OOAB4982 OOAB4982 OOAB492C4 OOAB495C4 OOAB495C OOAB455C OOAB455C	BD85 40EFFFFF 1ea 66:0F1385 6CEFFFFF movi 818D 6CEFFFF movi 50 pusi 6A 00 6A 00 pusi 6A 00 761 50 CEFFFF pusi 6A 6A 00 pusi 66:0F1385 42EFFFF movi 66:0F1385 44EFFFFF movi 66:0F1385 54EFFFF movi 66:0F1385 54EFFFFF movi 66:0F1385 54EFFFFF movi 66:0F1385 54EFFFFF movi 66:0F1385 54EFFFFF movi 785 40EFFFFF movi 66:0F1385 54EFFFFF movi 785 40EFFFFF Movi 66:0F1385 54EFFFFF movi	<pre>text 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0</pre>	(ebp-10C4) (ebp-10C4) (smm0 xmm0 xmm0 xmm0 xmm0 44; 'D'	st(1) 0000000 st(2) 000000 st(3) 0000000 st(4) 000000 st(5) 4005800 st(6) 4002800 st(7) 3FFDC00 x87TagWord FF x87TW_0 3 (Em x87TW_2 3 (Em x87TW_2 3 (Em x87TW_4 3 (Em c Default (stdcall) 1: [esp 0000	pty) x87TW_1 3 (Empty) pty) x87TW_3 2 (Empty) pty) x87TW_5 3 (Empty) utv) x87TW 7 3 (Emptv) v5 ↓ Unlock 0000 755988 "taskkill /F /IM zool 000000
.text:00AB49FE malware.exe	Dump 3 U Dump 4 U Dump	5 🛞 Watch 1 [x=] Locals	Struct	0075481C 0000	
Address Hex	a comp of grap comp 1 grap comp	ASCII	<u>ac</u> 00 000	00754820 0075	0000
00755988 74 61 73 68 68 6 00755988 7A 6F 6F 6C 7A 2 00755988 70 45 22 0 2 00755988 00 00 00 00 00 0 00755988 00 00 00 00 00 0 00755908 00 00 00 00 00 00	9 6C 6C 20 2F 46 20 2F 49 4D 65 78 65 20 2F 54 00 22 27 79 00 00 2F 79 00 <td>20 Easkkill /F /IM 2F zoolz.exe /T." / 00 y.E" /y/y.y.y. 00</td> <td></td> <td>00754828 0000 00754820 0000 00754830 0800 00754834 0000 00754838 0000 00754836 0075 00754840 0075</td> <td>0001 0000 0000 0000 485 8</td>	20 Easkkill /F /IM 2F zoolz.exe /T." / 00 y.E" /y/y.y.y. 00		00754828 0000 00754820 0000 00754830 0800 00754834 0000 00754838 0000 00754836 0075 00754840 0075	0001 0000 0000 0000 485 8

The process deletes the Volume Shadow Copies using the vssadmin command, as highlighted below:

OOA8498C OOA8498D OOA8498D OOA8498D OOA8498D OOA8498A OOA849A5 OOA8498B OOA84982 OOA8498C OOA8498C	8085 40EFFFFF 66:0F1385 6CEFFFFF 8180 6CEFFFFF 0001000 50 6A 00 6A 00 6A 00 6A 01 6A 016A 01 6A	push o movlpd qword ptr s movlpd qword ptr ss: call dword ptr ds:	<pre>si [ebp-1094] xmm0 bp-1094] 100 ss [ebp-1058] xmm0 ss [ebp-1058] xmm0 si ebp-1084 xmm0 si ebp-1084 xmm0 si ebp-1084 xmm0 si ebp-1084 xmm0 si ebp-1084 xmm0 bp-1086 xmm0 si ebp-1086 xmm0</pre>	44: 'D'	<pre>S1(0) 00000000000000000 X87r1 Empty 0. ST(1) 0000000000000000000 X87r1 Empty 0. ST(3) 00000000000000000000 X87r3 Empty 0. ST(3) 000000000000000000 X87r4 Empty 0. ST(5) 4008800000000000000 X87r5 Empty 6. ST(6) 400880000000000000 X87r5 Empty 6. ST(6) 40088000000000000 X87r5 Empty 6. ST(7) 3FFDC000000000000 X87r5 Empty 6. X87TW_0 3 (Empty) X87TW_3 3 (Empty) X87TW_4 3 (Empty) X87TW_5 3 (Empty) X87TW_4 3 (Empty) X87TW_5 3 (Empty) X87TW_4 3 (Empty) X87TW_5 3 (Empty) X87TW_4 3 (Empty) X87TW 7 3 (Empty) X87TW 6 3 (Empty) X87TW 7 3 (Empty) X87TW_4 0000000 ≤: [esp+4] 0000000 5: [esp+4] 0000000 S: [esp+10] 00000001</pre>
dword ptr ds:[00B1903C <ma .text:00AB49FE ma]ware.exe .text:00AB49FE ma]ware.exe</ma 	:\$149FE #13DFE	= <kernel32.createpr Dump 5 💮 Watch 1</kernel32.createpr 	IX= Locals 🏾 🖉 Strue	ct	0075480C 00000000 007548E0 0083E9C8 malware.0083E9C8
Address Hex 0083E9C8 76 73 73 61 64 67 0083E9C8 53 68 61 64 67 0083E9E8 69 65 74 00 C7 00 0083E9F8 65 63 44 65 76 63 0083EA08 76 69 63 65 00 00 0083EA18 65 65 46 72 61 65	73 20 2F 61 6C 6C 20 21 0 00 80 42 61 63 68 75 70 63 65 4D 65 64 69 61 5 0 00 00 98 00 00 80 4D 63	F 71 75 Shadows /a 0 45 78 iet.ÇBad 3 65 72 ecDeviceMed 3 41 66 vice	ll /qu ckupEx diaSer McAf	^	007548E4 0000000 007548E5 0000000 007548EC 0000000 007548F4 0000000 007548F4 0000000 007548F4 0000000 007548F6 00754918

Figure 16

GetLogicalDriveStringsW is utilized to extract the valid drives in the system (Figure 17):

ETP	->• 00AB127 00AB127 •• 00AB127	8 57 FFD6		push ea push ed call es	i			× 4	2: [esp+4] 0090D230 3: [esp+8] 51B05FEC 4: [esp+C] 00000002 5: [esp+L] 00000050
esi= <kernel< th=""><th>32.GetLogic</th><th></th><th></th><th></th><th>_</th><th></th><th></th><th><u> </u></th><th>. [cspito] 0000000</th></kernel<>	32.GetLogic				_			<u> </u>	. [cspito] 0000000
Dump 1	Dump 2	Dump 3	Ump 4	Ump 5	🥘 Watch 1	[x=] Locals	2 Struct	0	0755984 00000009 0755988 0090D230

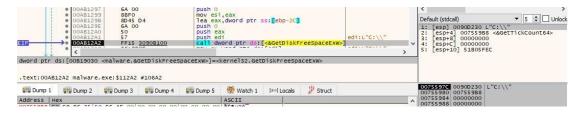
Figure 17

The executable retrieves the drive type via a function call to GetDriveTypeW:

dword ptr ds: [00B1291 dware, exc	push edi call dword ptr ds:[<&GetDriveTypew>] w>]= <kernel32.getdrivetypew></kernel32.getdrivetypew>	edi:L"C:\\"	3: [esp+8] 0000002 4: [esp+C] 00000050 5: [esp+10] 00755A58
Numeral States of State			00755988 0090D230 L"C:\\"

Figure 18

It extracts the amount of space that is available on the disk volume using the GetDiskFreeSpaceExW routine, as shown below:





The sample spawns two processes in order to resize the maximum amount of storage space used for shadow copy storage:

00AB498C 00AB498D 00AB499B 00AB499B 00AB499B 00AB499B 00AB499B 00AB499B 00AB499B 00AB493B 00AB49A5 00AB49A5 00AB49A6 00AB49A6 00AB49A6 00AB49A6 00AB49A6 00AB49A6 00AB49A6 00AB49A7 00AB49A7 00AB49A7 00AB49B1 00AB49B1 00AB49B5 00AB49C4 00AB49B5 00AB49C4 00AB49B6	[ebp-10C4]:"vssad	ST(1) 00000000000000000 x87r1 Empty 0 ST(2) 0000000000000000 x87r2 Empty 0 ST(2) 0000000000000000 x87r2 Empty 0 ST(3) 0000000000000000 x87r2 Empty 0 ST(4) 00000000000000000 x87r2 Empty 0 ST(4) 00000000000000000000 x87r4 Empty 0 ST(5) 400580000000000000 x87r5 Empty 8 ST(6) 400280000000000000 x87r6 Empty 8 ST(6) 400280000000000000 x87r6 Empty 8 ST(7) 3FFDC000000000000 x87r6 Empty 8 x87TagWord FFFF x87Tw_0 3 (Empty) x87Tw_3 3 (Empty) x87Tw_2 2 (Empty) x87Tw_3 5 (Empty) x87Tw 4 (Empty) x87Tw_3 5 (Empty) x87Tw 6 3 (Emptv) x87Tw 7 3 (Empty) x87Tw 4 (empty) x87Tw 7 3 (Empty) x87Tw 4 (empty) x87Tw 7 3 (Empty) x87Tw 4 (empty) x87Tw 7 3 (Empty) x87Tw 6 3 (Emptv) x87Tw 7 3 (Emptv) x87Tw 6 3 (Emptv) x87Tw 7 3 (Emptv)
.text:00AB49FE malware.exe:\$149FE #13DFE ## Dump 1 ## Dump 2 ## Dump 3 ## Dump 4 ## Dump 5	^	007548CC 00000000 007548D0 00755A88 007548D4 0000000 007548D4 0000000 007548D5 00000000 007548DC 0000000 007548E4 0000000 007548E4 0000000 007548E6 000754908 007548E0 00754970
Figure 20	[ebp-10C4]:"vssad 44:'D'	ST(1) 000000000000000000000000000000000000

007548CC 0000000 007548D0 00755A88 007548D4 0000000 007548D8 0000000 007548DC 00000001 007548E0 0800000 007548E4 0000000 007548E8 0000000

007548DC 007548E0 007548E4 007548E8 007548EC 00754908 007548E0 00754970

>

🛞 Watch 1 🛛 🕅 🖉 🖉

Figure 21

2 Struct

The ransomware enumerates the files on the drives using the FindFirstFileExW and FindNextFileW APIs:

.text:00AB49FE malware.exe:\$149FE #13DFE

Ump 1 Ump 2 Ump 3 Ump 4 Ump 5

<

dword ptr ds:[00B1903C <malware.&CreateProcessA>]=<kernel32.CreateProcessA>

 Address
 Hex
 ASCII

 00755A88
 76 73 73 61
 64 60 69 6E
 20 72 65 73
 69 7A 65 20
 Vssadmin resize

 00755A88
 76 73 73 61
 64 60 77 73 74
 67 76 75 67 16 76 52 02
 26 58 61 69 72
 Vssadmin resize

 00755A88
 6F 72 3D 43
 3A 20 2F 6F
 16 76 52 02
 27 60 61 0r=C: /on=C: /on=C:

"vssadmin resize shado

Denduit (succany [1: [esp] 00000000 2: [esp]+4] 00755AB8 "vssadmin resize shad 3: [esp]+8] 00000000 4: [esp]+0] 00000000 5: [esp]+10] 00000001

00AC04E2 00AC04E3 00AC04E3 00AC04E4 00AC04E5 00AC04E6 00AC04E6 00AC04E7 00AC04E8 €	56 56 51 50 FF15 <u>C490B100</u>	1	i x i x ord ptr ds:	[<&FindFirstFileExw>]	eax:L"C:\\\\#"	C > Default (stdcall) ▼ 5 • Unlock 1: [esp1 00755784 L"C:*" 2: [esp+4] 00755784 L"C:*" 4: [esp+6] 00755534 4: [esp+10] 0000000 5: [esp+10] 0000000
 s:[00B190C4 <ma 4E8 malware.exe</ma 	lware.&FindFirstFi \$204E8 #1F8E8	leExW>]= <kerr< th=""><th>1el32.FindFi</th><th>rstFileExW></th><th></th><th></th></kerr<>	1el32.FindFi	rstFileExW>		
 4E8 malware.exe		leExw>]= <kerr< td=""><td>00132.FindFi</td><td>rstFileExW> [x=]Locals 2 Struct</td><td></td><td>0755508 00755784 L"C:*" 0075550C 0000000</td></kerr<>	00132.FindFi	rstFileExW> [x=]Locals 2 Struct		0755508 00755784 L"C:*" 0075550C 0000000

The following files and directories will be skipped from encryption:

.rdata:00827888	aGSkpDir:			; DATA XREF: sub AB14A0+817to
.rdata:00827888				; sub AB14A0+83E↑o
.rdata:00B27B88		text	"UTF-16LE",	'g_skp_dir',0
.rdata:00B27B9C	aDesktopIni:			; DATA XREF: sub_AB14A0+97Fto
.rdata:00B27B9C		text	"UTF-16LE",	'desktop.ini',0
.rdata:00B27BB4	aThumbsDb:			; DATA XREF: sub_AB14A0+995†o
.rdata:00B27BB4		text	"UTF-16LE",	'Thumbs.db',0
.rdata:00B27BC8	aWindows:			; DATA XREF: sub_AB27F0+2941o
.rdata:00B27BC8		text	"UTF-16LE",	'Windows',0
.rdata:00B27BD8	aWindowsOld:			; DATA XREF: sub_AB27F0+456↑o
.rdata:00B27BD8		text	"UTF-16LE",	'Windows.old',0
.rdata:00B27BF0	aPerflogs:			; DATA XREF: sub_AB27F0+6261o
.rdata:00B27BF0		text	"UTF-16LE",	'PerfLogs',0
.rdata:00B27C02		align	4	
.rdata:00B27C04	aMsocache:			; DATA XREF: sub_AB27F0+7F6to
.rdata:00B27C04		text	"UTF-16LE",	'MSOCache',0
.rdata:00B27C16		align	4	
.rdata:00B27C18	aProgramFiles:			; DATA XREF: sub_AB27F0+9EEto
.rdata:00B27C18		text	"UTF-16LE",	'Program Files',0
.rdata:00B27C34	aProgramFilesX8:			; DATA XREF: sub_AB27F0+BBD1o
.rdata:00B27C34		text	"UTF-16LE",	'Program Files (x86)',0
.rdata:00B27C5C	aProgramdata:			; DATA XREF: sub_AB27F0+D8D10
.rdata:00B27C5C		text	"UTF-16LE",	'ProgramData',0

Figure 23

The GetFileExInfoStandard API is utilized to obtain attributes for a file or directory (0x0 = GetFileExInfoStandard):

IP	00AC065 00AC065 00AC065 00AC065	9 6A 00 8 50	CC90B100	push (push (push (call (D	<&GetFileA	ttributesE	eax:L"test\\file.zip	✓ 4:	[esp+8] 00F35384
1999 - 1997 -	is:[00B190CC			butesExW>]	= <kernel32.ge< th=""><th>tFileAttrib</th><th>utesExW></th><th>></th><th>5:</th><th>[cohiza] arrange</th></kernel32.ge<>	tFileAttrib	utesExW>	>	5:	[cohiza] arrange
Ump 1	Dump 2	Dump 3	Ump 4	Ump 5	🛞 Watch 1	[x=] Locals	2 Struct			35374 0125A3E0 L"test\\file.zip" F35378 00000000
ddeare lu	av				LACCTT	1			. 008	F3537C 00F35384

Figure 24

The ransom note called "!!!READ_ME_MEDUSA!!!.txt" is created in every traversed directory. It contains the victim's name and a 32-byte hash that should be used during the communication with the threat actor:

812	 0081009E 008100A0 008100A1 008100A4 008100A7 008100AA 008100AD 008100AD 	6A 00 50 FF75 18 FF75 0C FF75 1C FF75 14 FF75 08 FF15 BC	push push push push		ss: ebr	0+C 0+1C 0+14 0+8	Louis Ji	[ebp+8]:L"test\\	< De 1: 2:	fault (stdca [esp] [esp+4 [esp+8	II) 00F34F78 1 4000000 00000000 000F34D9	L"test\ 0 3	and statements of the statement of the s	 Unloci
	• <								> 5	[esp+1	000000	02		
	ds:[00B190BC <n 00B0 malware.e></n]= <kernel:< th=""><th>2.Create</th><th>ilew></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th></kernel:<>	2.Create	ilew>									
	00B0 malware.e>	(e:\$700B0 #0]= <kernel:< td=""><td></td><td></td><td>=] Locals</td><td>Struct</td><td></td><td></td><td></td><td>00F34F78</td><td>L"test</td><td>(\!!!RE/</td><td>D_ME_MEC</td></kernel:<>			=] Locals	Struct				00F34F78	L"test	(\!!!RE/	D_ME_MEC

IIIREAD_ME_MEDUSAIII - Notepad

\$\	\$\$\	\$\$\$\$\$\$\$	\$\$\$\$\$	\$\$\	\$\$\	\$\$\	\$\$\$\$	\$\$\	\$\$\$\$\$	\$\
\$\$\	\$\$\$	\$\$	\$\$	\$\$\	\$\$	\$\$	\$\$	\$\$\	\$\$	\$\$\
\$\$\$\	\$\$\$\$	\$\$	\$\$	\$\$	\$\$	\$\$	\$\$ /	1	\$\$ /	\$\$
\$\\$\$\\$	\$ \$\$	\$\$\$\$\$\	\$\$	\$\$	\$\$	\$\$	\\$\$\$\$	\$\$\	\$\$\$\$\$\$	\$\$
\$ \\$\$\$	\$\$	\$\$	\$\$	\$\$	\$\$	\$\$	1	\$\$\	\$\$	\$\$
\$ \\\$	/\$\$	\$\$	\$\$	\$\$	\$\$	\$\$	\$\$\	\$\$	\$\$	\$\$
\$ _	/ \$\$	\$\$\$\$\$\$\$\$	\$\$\$\$\$	\$\$	\\$\$\$\$	5\$	\\$\$\$\$	\$	\$\$	\$\$
	1	1	1	1	1	1	1	1		\mathbf{N}

WHAT HAPPEND?

1. We have PENETRATE your network and COPIED data.

* We have penetrated entire network including backup system and researched all about your data.

* And we have extracted terabytes of your important and valuable data and copied them to private cloud storage.

2. We have ENCRYPTED your files.

While you are reading this message, it means all of your files and data has been ENCRYPTED by world's strongest ransomware. All files have encrypted with new military-grade encryption algorithm and you can not decrypt your files. But don't worry, we can decrypt your files.

There is only one possible way to get back your computers and servers - CONTACT us via LIVE CHAT and pay for the special MEDUSA DECRYPTOR and DECRYPTION KEYs. This MEDUSA DECRYPTOR will restore your entire network, This will take less than 1 business day.

WHAT GUARANTEES?

We can post your data to the public and send emails to your customers.

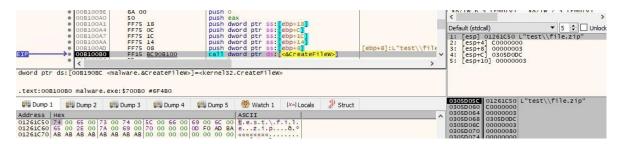
We have professional OSINTs and media team for leak data to telegram, facebook, twitter channels and top news websites. You can easily search about us.

You can suffer significant problems due disastrous consequences, leading to loss of valuable intellectual property and other sensitive information, costly incident response efforts, information misuse/abuse, loss of customer trust, brand and reputational damage, legal and regulatory issues. After paying for the data breach and decryption, we guarantee that your data will never be leaked and this is also for our reputation.

YOU should be AWARE!

Figure 26

The sample opens a target file by calling the CreateFileW API (0xC0000000 = **GENERIC_READ** | **GENERIC_WRITE**, 0x3 = **FILE_SHARE_READ** | **FILE_SHARE_WRITE**, 0x3 = **OPEN_EXISTING**, 0x80 = **FILE_ATTRIBUTE_NORMAL**):





GetFileType is utilized to retrieve the file type, as highlighted below:

	dword ptr ds:[00B1921C <mathrms dware.exe<="" th=""><th></th><th>push edi call dword ptr ds:[<&GetFileType>]]=<kernel32.getfiletype></kernel32.getfiletype></th><th>3: [esp+8 4: [esp+2 5: [esp+1</th><th>] 0305D198] 0000000 0] 00000000</th></mathrms>		push edi call dword ptr ds:[<&GetFileType>]]= <kernel32.getfiletype></kernel32.getfiletype>	3: [esp+8 4: [esp+2 5: [esp+1] 0305D198] 0000000 0] 00000000
--	--	--	--	-------------------------------------	--

Figure 28

The malicious process moves the file pointer of the target file via a function call to SetFilePointerEx (see Figure 29).

112	0080243C 0080243D 00802440 00802443 00802443 00802444	51 FF75 10 FF75 0C 50 FF15 <u>D490B100</u>	push ecx push dword ptr push dword ptr push eax call dword ptr		pinterEx>]	 Default (stdcall) 1: [esp] 00000 2: [esp+4] 000 3: [esp+8] 000 4: [esp+6] 030 5: [esp+10] 00	DFE4 000000 000000 05D150	E Unic
dword ptr o	ds:[00B190D4 <m< th=""><th>alware.&SetFilePoint</th><th>terEx>]=<kernel32.se< th=""><th>tFilePointerEx></th><th>r</th><th></th><th></th><th></th></kernel32.se<></th></m<>	alware.&SetFilePoint	terEx>]= <kernel32.se< th=""><th>tFilePointerEx></th><th>r</th><th></th><th></th><th></th></kernel32.se<>	tFilePointerEx>	r			
.text:00B02	2444 malware.ex	e:\$62444 #61844						
.text:00B0:		e:\$62444 #61844	📖 Dump 5 🛛 👹 Wat	ch 1 [x=] Locals	Struct	0305D134 0000 0305D138 0000		

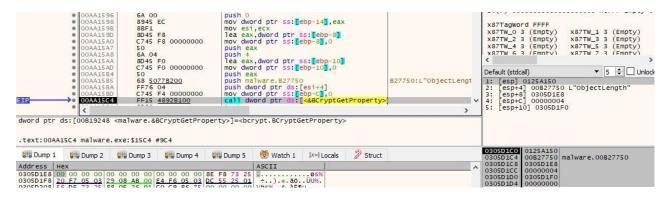
Figure 29

Each file is read by calling the ReadFile function:

EIP dword ptr (7587340 <		940FBD75 ReadFile>l=	1.4.1.2	ord ptr ds:[<	<&ReadFile) ReadFi	le v		10] 0305D0E 14] 0000000		
	34C0 kernel3	2.d11:\$234C0) #144C0 <re< th=""><th>adFile></th><th></th><th></th><th></th><th></th><th>02050084</th><th>00802057</th><th>atura</th><th>to malware.0080</th></re<>	adFile>					02050084	00802057	atura	to malware.0080
Dump 1	Dump 2	Dump 3	Dump 4	Dump 5	💮 Watch 1	[x=] Locals	2 Struct			00000FE4	ecurii	to marware.oobc
	FO AD BA O				ASCII .0.°.0.°.0.			^	0305D0C0 0305D0C4	0125D7B0 00001000 0305D0E4 00000000		

Figure 30

The BCryptGetProperty API is used to obtain the values of the "ObjectLength" and "BlockLength" properties for the CNG object:





IP	00AA1617 00AA1619 00AA1614 00AA1614 00AA1616 00AA1616 00AA1620 00AA1625 00AA1628 €	6A 00 50 6A 04 8D45 F4 50 68 <u>6C778200</u> FF76 04 FF15 <u>48928100</u>	push e push m push d	ax ux,dword ptr s ax alware.B2776C word ptr ds:[B2776C:L"BlockLength	<pre>< Default (std 1: [esp] 2: [esp+ 3: [esp+ 4: [esp+</pre>	dcall) dcall) dcall)	≑ □u
dword ptr	ds:[00B19248 ·	cmalware.&BCryptGetP	roperty>]= <bo< th=""><th>crypt.BCryptG</th><th>etProperty></th><th></th><th>-</th><th></th><th></th></bo<>	crypt.BCryptG	etProperty>		-		
	8	cmalware.&BCryptGetP exe:\$1628 #A28	roperty>]= <bo< th=""><th>crypt.BCryptG</th><th>etProperty></th><th></th><th>-</th><th></th><th></th></bo<>	crypt.BCryptG	etProperty>		-		
	1628 malware.			crypt.BCryptG	etProperty> [x=] Locals Ø Struct			0 0125A150 4 0082776C malware.00823	76C

The following 16 bytes represent the IV (initialization vector) that is the same for all files to be encrypted:

and the second second second second second	166C malware.	Dump 3	Uump 4	Dump 5	🛞 Watch 1	[x=] Locals	Struct	
and the second second second second second	166C malware.	EXE: \$100C #	1997 B				- COS	
eax=0124D29	00 ls:[esi+14]=[0							
	• <						02	3
	• 00AA166F	83C4 0		add esp				
IP	● 00AA1667 →● 00AA166C	E8 34D 8B46 1			alware.AEEEAC			
	 00AA1661 00AA1666 	50	39B300	push ea				
					7	ebp-C		

Figure 33

The AES chaining mode is set to cipher block chaining using the BCryptSetProperty routine:

	 00AA16AA 00AA16AC 00AA16AE 	6A 20	78200	push 0 push 20) lware.B2778		B27784:L"ChainingMoc	Default (st		₹ 5 😫 🗆 ι
P	• 00AA16B3 00AA16B8 • 00AA16B8 • 00AA16B8	68 <u>A47</u> FF76 0 FF15 <u>4</u>	7 <u>B200</u> 4 492B100	push ma push dw call dw	lware.8277A word ptr ds: word ptr ds:	4 [esi+4] [<&BCryptSetProper	B277A4:L"ChainingMoc	2: [esp 3: [esp 4: [esp] 0125A150 +4] 00B277A4 +8] 00B27784 +C] 00000020 +10] 0000000	
word ptr o	ds:[00B19244	<malware.&bc< th=""><th>CryptSetProp</th><th>perty>]= bci</th><th>rypt.BCryptS</th><th>SetProperty></th><th></th><th></th><th></th><th></th></malware.&bc<>	CryptSetProp	perty>]= bci	rypt.BCryptS	SetProperty>				
622	ds:[00B19244 16BB malware.			perty>]= <box< th=""><th>rypt.BCryptS</th><th>6 189</th><th></th><th></th><th></th><th></th></box<>	rypt.BCryptS	6 1 89				
640		exe:\$16BB #A		perty>]= <box< td=""><td>rypt.BCryptS</td><td>setProperty> [x=] Locals</td><td>ruct</td><td></td><td>4 0125A150 3 00B277A4 m</td><td>na]ware.008277A4</td></box<>	rypt.BCryptS	setProperty> [x=] Locals	ruct		4 0125A150 3 00B277A4 m	na]ware.008277A4

Figure 34

The malware creates a key object based on 32 bytes that were generated, which represent the AES256 key that is changing between iterations:



Address	He	¢															ASCII
0125E7C8	00	00	00	00	00	00	00	00	14	00	00	00	52	55	55	55	
0125E7D8	50	A1	25	01	FO	E7	25	01	00	00	00	00	00	00	00	00	Pi%.ðç%
0125E7E8	00	00	00	00	00	00	00	00	4A	02	00	00	4B	53	53	4D	JKSSM
0125E7F8	02	00	01	00	01	00	00	00	10	00	00	00	10	00	00	00	
0125E808	00	01	00	00	00	00	00	00	38	A2	25	01	00	00	00	00	8¢%
0125E818	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0125E828	20	00	00	00	4D	58	4C	22	BC	OF	64	66	88	OB	71	7B	MXL"%.dfq{
0125E838	DG	CE	7E	94	12	C6	D1	29	E6	DO	65	9D	3C	FA	9B	1D	ÖÎ~4Ň)æĐe.<ú
0125E848	99	8D	2C	CB	00	00	00	00	4D	58	4C	22	BC	OF	64	66	ËMXL"¼.df
0125E858	88	0B	71	7B	DG	CE	7E	94	12	C6	D1	29	E6	DO	65	9D	q{ÖÎ~4Ň)æĐe.
0125E868	3C	FA	9B	1D	99	8D	2C	CB	11	29	53	CC	AD	26	37	AA	<ú,Ë.)SÌ.&7ª
0125E878	26	2D	46	D1	FO	E3	38	45	9E	D7	DG	47	78	07	B 3	DA	&-FNOASE.xÖGx.*Ú
0125E888	44	FD	28	C7	DD	70	04	OC	42	DB	AD	OD	EF	FD	9A	A7	Dý(CÝpBÛïý.§
0125E898	C9	DO	DC	76	39	33	E4	33	8C	14	BF	84	F4	13	0C	5E	ÉDÜV93ā3¿.ô^
0125E8A8	BO	EE	24	99	6D	9E	20	95	4D	6C	87	31	A2	91	1D	96	°î\$.mMl.1¢
0125E8B8	GB	41	C1	EO	52	72	25	D3	8C	54	80	E2	78	47	8C	BC	kAAaRr%O.T. axG. 4
0125E8C8	C8	A9	AS	25	A5	37	88	BO	DF	AS	60	37	7D	39	7D	A1	È@ %¥7.°B 7}9};
0125E8D8	16	78	BC	41	44	0A	99	92	97	33	6E	AD	EF	74	E2	11	.x¼AD3n.itå.
0125E8E8	27	DD		34	82	EA		84	48				35	B4	42		

Firstly, the AES key is encrypted using the RSA public key via a call to BCryptEncrypt:

	 00A8088F 00A80891 00A80897 00A80898 00A80898 00A8084 00A8084 00A80847 00A808A7 00A808A7<th>6A 04 SD85 54D8FFFF SD85 54D8FFFF 8D85 10FEFFFF 50 6A 00 8D85 38D8FFFF 50 6A 20 8D45 C4</th><th>push eax push dword ptr lea eax,dword p push eax push 0 push 0 lea eax,dword p push eax push 20 lea eax,dword p</th><th>otr ss:[ebp-1F0] otr ss:[ebp-24C8]</th><th>[ebp-24C8]:L"SHA1"</th><th>ST(6) 00000000000000000000000 x87r5 Empty 0, ST(7) 0000000000000000000 x87r7 Empty 0, x87TagWord FFFF x87TW_0 3 (Empty) x87TW_1 3 (Empty) x87TW_2 3 (Empty) x87TW_3 3 (Empty) x87TW_4 3 (Empty) x87TW_5 3 (Empty) x87TW 4 3 (Empty) x87TW 7 3 (Fmoty) x87TW 4 3 (Empty) x87TW 7 3 (Fmoty) x87TW 5 3 (Empty) x87TW 6 3 (Empty) x87TW 7 3 (Empty)</th>	6A 04 SD85 54D8FFFF SD85 54D8FFFF 8D85 10FEFFFF 50 6A 00 8D85 38D8FFFF 50 6A 20 8D45 C4	push eax push dword ptr lea eax,dword p push eax push 0 push 0 lea eax,dword p push eax push 20 lea eax,dword p	otr ss:[ebp-1F0] otr ss:[ebp-24C8]	[ebp-24C8]:L"SHA1"	ST(6) 00000000000000000000000 x87r5 Empty 0, ST(7) 0000000000000000000 x87r7 Empty 0, x87TagWord FFFF x87TW_0 3 (Empty) x87TW_1 3 (Empty) x87TW_2 3 (Empty) x87TW_3 3 (Empty) x87TW_4 3 (Empty) x87TW_5 3 (Empty) x87TW 4 3 (Empty) x87TW 7 3 (Fmoty) x87TW 4 3 (Empty) x87TW 7 3 (Fmoty) x87TW 5 3 (Empty) x87TW 6 3 (Empty) x87TW 7 3 (Empty)
dword ptr o	00AB0885 00AB0886 00AB0886 00AB0885 00AB0885 4 ds:[00B19250 <m 00B5 ma]ware.ex</m 	50 8885 280BFFFF FF70 20 FF15 50928100 alware.&BCryptEncryp e:\$108BF #FCBF	push dword ptr call dword ptr	ds:[<&BCryptEncrypt>]	>	1: [esp] 0125C230 2: [esp+4] 0305F64 3: [esp+8] 00000020 4: [esp+C] 0305258 &L"SHA1" 5: [esp+10] 00000000
Dump 1	💭 Dump 2 🚦	Dump 3 🔛 Dump 4	💷 Dump 5 🛛 👹 Wate	ch 1 🛛 🗱 🕺 Str	ruct	0305D1E0 0125C230 0305D1E4 0305F6E4
0305F6F4 1 0305F704 5 0305F714 5 0305F724 4	D 5.8 4C 22 BC 0 2 C6 D1 29 E6 D 6 DE 7.3 25 01 0 4 FB 05 0.3 31 3 0 47 AB 00 00 0	F 64 66 8B 08 71 7B 0 65 90 3C FA 9B 1D 0 00 00 2C F7 05 03 5 B1 00 00 00 00 00 00 0 A0 OF AB 3C A5 01 6 20 01 48 3C A5 01 6 20 02 20 20 02 02 03	99 8D 2C CB .4N) adde 08 D2 05 03 VPs% 60 FB 05 03 Tû15± 40 47 AB 00 @G«	<û,Ĕ .,÷ò û		0305DIE8 00000020 0305DIE0 0305D258 4L"SHA1" 0305DIF0 00000000 0305DIF4 00000000 0305DIF8 0305F530 0305DIF6 0000100 0305D200 0305D274 0305D204 00000004

Figure 37

Address	He	ĸ														_	ASCII
0305F530	C4	C4	0E	59	06	12	C0	AE	65	16	95	60	65	09	7F	8C	AA.YAse e
0305F540	1F	00	45	B2	6D	E7	2C	59	BE	18	49	56	4E	BC	E6	05	E=mc, Y%. IVN%æ.
0305F550	5A	61	1A	12	BA	48	56	53	3B	71	6C	33	1E	FD	A2	OD	Za. °HVS;ql3.y¢.
0305F560	45	GC	42	64	61	7B	98	C6	37	CA	EF	54	07	2B	59	C5	ElBda{.47ÊïT.+YÂ
																	n. N=Rez'N. 1+.g
0305F580	96	CB	CA	A7	9E	F7	OC.	E2	5D	5A	BE	90	C6	71	B 9	2B	.EE§.÷. a]Z%.Aq'+
0305F590	01	DF	8D	46	11	AE	E9	6C	5A	77	56	90	43	B3	5F	48	.B.F.@élŹwV.C*K
																	»É.ØcOº**ë.ÕBÊ
0305F5B0	CC	98	CA	86	BA	BE	DD	ED	31	38	15	08	3B	33	7D	03	1.Ê.º¾Ý118;3}.
0305F5C0	09	EC	F1	87	2F	20	C3	BD	4B	88	F4	A7	82	7C	34	EE	.iñ./ ŽK.ô§. 41
0305F5D0	AF	A6	CB	85	D3	FC	D3	D8	50	86	2C	2E	F3	CE	C5	F2	
0305F5E0	22	AG	87	F8	CD	B1	6A	69	70	FC	E9	93	BC	87	AO	C6	".øl±jipüé.¼. Æ
0305F5F0	1E	OD	17	D2	B8	D7	EO	BA	6C	DC	3C	34	OE	41	B6	DA	Ò xaº10<4 A¶Ú
0305F600	54	B 3	A7	1A	95	9D	5A.	13	28	1D	40	23	D9	27	E2	8B	T*§Z.+.@#Ù'â.
0305F610	8A	BO	09	2A	AA	2A	C7	E6	9D	75	C9	90	69	96	41	E6	.°.**¢Çæ.uÉ.i.Aæ
0305F620	92	AF	32	04	67	B 7	68	7A	80	43	EO	20	7D	A7	30	02	. 2.g.hz.ca }§0.

Figure 38

The file content is encrypted using the AES256 algorithm, as highlighted in Figure 39.

	00AA1746 00AA1748 00AA1749 00AA1740 00AA174D 00AA174D 00AA174E 00AA174F 00AA1751	6A 00 57 50 FF75 F4 51 52 6A 00 50	push 0 push edi push eax push dword ptr push ecx push ecx push edx push eax	ss:[ebp-C]			≪ Default (stdca	B (Empty) B (Empty) B (Empty)	x87TW_3 3 (Empty) x87TW_5 3 (Empty) x87TW_5 3 (Empty) x87TW 7 3 (Empty) x87TW 7 3 (Empty) x87TW 7 3 (Empty)
		53 FF75 F8 FF15 5092B100 Tware.&BCryptEncrypt	<pre>push ebx push dword ptr call dword ptr >]=<bcrypt.bcryptl< pre=""></bcrypt.bcryptl<></pre>	ds: [<&BCrypt	Encrypt>]	× >	3: [esp+8 4: [esp+C] 0305E278] 00001000] 00000000 0] 0124D290	i.
	6 malware.exe	Dump 3 🛄 Dump 4	📖 Dump 5 🛛 🛞 Wa	tch 1 [x=] Loca	ls 🤌 Struct		0305B16C 0305B170	0305E278	
0305E288 B8 1 0305E298 GF 0 0305E2A8 3F E 0305E2B8 14 5 0305E2C8 AE 1	L3 ED AB 13 00 53 65 78 70 28 54 49 16 58 64 50 69 C0 72 58 51 15 61 17 F1	00 00 08 00 CA 88 1 78 53 28 00 08 00 C 65 78 65 E4 8D 78 7 61 83 06 45 45 89 8 40 61 61 43 CC 2E 5 12 05 37 08 59 1F	0 00 70 72i« C 94 C5 F5 ocexp. 2 D5 1A 54 ?%I.Xd E 02 2A 90 .PiÀr[6 A3 A2 C5 @a.ñ	xS+pr exeä%{ .ÅÕ аЕЕÕ.Т ЗааСÌ.^.*.		^	03058174 03058178 0305817C 03058180 03058184 03058188 03058188 03058180	00000000 0124D290 00000010 0305B1EC 00001000 0305B1E8	

The encrypted data is written back to the file by calling the WriteFile function:

	 00B05AA2 00B05AA4 00B05AA5 00B05AA6 00B05AA9 	6A 00 AB AB 8D45 D0 50	2	push 0 stosd stosd lea eax push ea	,dword ptr	ss:[ebp-24]			×87 <	TW 6 3 (Empty)	x87TW 3	S (Empty) 7 3 (Empty)
and a second second	00805AAA 00805AAD 00805AAF → 00805AAF < COB19208 <r< th=""><th>nalware.&Wr</th><th>8928100 iteFile>]=</th><th>push et push et call dv</th><th>vord ptr ds:</th><th></th><th>e)]</th><th>></th><th>2: 3: 4:</th><th>[esp] 00000FE4 [esp+4] 0305D278 [esp+8] 00001000 [esp+C] 0305D0C0 [esp+10] 0000000</th><th></th><th></th></r<>	nalware.&Wr	8928100 iteFile>]=	push et push et call dv	vord ptr ds:		e)]	>	2: 3: 4:	[esp] 00000FE4 [esp+4] 0305D278 [esp+8] 00001000 [esp+C] 0305D0C0 [esp+10] 0000000		
.LEX1:00805	AAF manware.ex	Ke: \$65AAF #	64EAF									
Dump 1	Ump 2	Dump 3	Dump 4	Dump 5	🤴 Watch 1	[x=] Locals	Struct			5D09C 00000FE4 5D0A0 0305D278		
Address He					ASCII			^		5D0A4 00001000 5D0A8 0305D0C0		
	B5 9F BB 0A	1B 92 35 9A			Gµ.»5.0]					5D0AC 00000000		

Figure 40

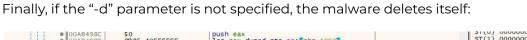
The ransomware appends the ".MEDUSA" extension to all encrypted files (see Figure 41).

ETTP	 00AB1164 00AB1165 00AB1165 	57	4090B100	push ea push ea	li	[<&MoveFilew	e	ax:L"test\' di:L"test\'	2:	[esp+8]	0305F278 2573DE56 01260E88	L"test\\	file.zip	.MEDU
	• <		<i></i>	1		Examover i ten	~		 5:	[esp+10]	75B6C9C0	<pre>kernel</pre>	32.GetTi	ckCou
aword ptr o	ds:[00B19040	<malware.&m< th=""><th>loveF1leW>]=</th><th><kernel32.m< th=""><th>oveFileW></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th></kernel32.m<></th></malware.&m<>	loveF1leW>]=	<kernel32.m< th=""><th>oveFileW></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th></kernel32.m<>	oveFileW>									
energen der er	ds:[00B19040 1166 malware.			≪kerne132.M	oveFileW>									

Figure 41

An encrypted file has the following structure: Encrypted file content + "MEDUSA" string + file length + Encrypted AES key with RSA + "Company identification hash" (Figure 42).

28:72	290h:	3E	A1	80	99	DE	3D	F1	98	2D	A2	AB	4D	64	0B	D3	6A	>;€™Þ=ñ~-¢«Md.Ój
28:72	AOh:	25	52	05	68	89	EC	9A	F7	D4	22	CB	B 8	9D	75	74	0E	%R.h‰ìš÷Ô"Ë,.ut.
28:72	BOh:	70	9B	AD	BB	1D	50	98	FF	DB	7D	42	5D	48	56	05	C9	p>-».P~ÿÛ}B]HV.É
28:72	2COh:	4D	45	44	55	53	41	00	00	BA	72	28	00	00	00	00	00	MEDUSAºr(
28:72	DOh:	00	02	00	00	00	00	00	00	C4	C4	0E	59	06	12	C0	AE	ÀÄ.YÀ®
28:72	EOh:	65	16	95	60	65	09	7F	8C	1F	00	45	B2	6D	E7	2C	59	e.•`eŒE²mç,Y
28:72	POh:	BE	18	49	56	4E	BC	E6	05	5A	61	1A	12	BA	48	56	53	%.IVN%æ.ZaºHVS
28:73	300h:	3B	71	6C	33	1E	FD	A2	OD	45	6C	42	64	61	7B	98	C6	;gl3.ý¢.ElBda{~Æ
28:73	10h:	37	CA	EF	54	07	2B	59	C5	F1	9A	4E	3D	52	E8	5A	B9	7ÊïT.+YÅñšN=RèZ¹
28:73	320h:	09	16	4E	8C	31	2B	84	67	96	CB	CA	A7	9E	F7	0C	E2	NŒ1+"g-Ëʧž÷.â
28:73]Z¾.Æq¹+.ß.F.®él
28:73	340h:	5A	77	56	90	43	B 3	5F	4B	BB	C9	8C	D8	63	30	94	8E	ZwV.C ³ _K»ÉŒØc0″Ž
28:73	350h:	BA	B3	2A	EB	0C	F5	42	CA	CC	98	CA	86	BA	BE	DD	ED	°³*ë.õBÊÌ~ʆ°¾Ýí
28:73	60h:	31	38	15	08	3B	33	7D	03	09	EC	F1	87	2F	20	C3	BD	18;3}ìñ‡/ ý
28:73																	and the second second	K^ô§, [41 ¦ËÓüÓØ
28:73																	69	P†,.óÎÅò"¦‡øĺ±ji
28:73																	BA	püé″¼‡ ÆÖ,×àº
28:73																	13	1Ü<4.A¶ÚŢ³§.•.Z.
28:73																		+.@#Ù'⋊°.*ª*Çæ
28:73										-	AF	32	04	67	B7	68	7A	.uÉ.i-Aæ' 2.g·hz
28:73	BDOh:	80	43	EO	20	7D	A7	30	02									€Cà }§0.



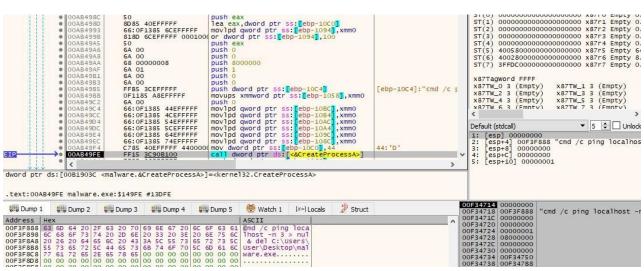


Figure 43

Medusa ransomware excludes the System folder from encryption by running with the "-f" parameter:

Command Prompt

C:\Users\User\Desktop>malware.exe -f -v
start
:exclude systemfolder
initial run powershell from predefined variable.
default key:0
preprocess
kill_services processes
kill services Acronis VSS Provider
kill_services Enterprise Client Service
kill_services Sophos Agent
kill_services Sophos AutoUpdate Service
kill_services Sophos Clean Service
kill_services Sophos Device Control Service
kill_services Sophos File Scanner Service
kill_services Sophos Health Service
kill_services Sophos MCS Agent
kill_services Sophos MCS Client
kill_services Sophos Message Router
kill_services Sophos Safestore Service
kill_services Sophos System Protection Service
kill_services Sophos Web Control Service
kill_services SQLsafe Backup Service
kill_services SQLsafe Filter Service
kill_services Symantec System Recovery
kill_services Veeam Backup Catalog Data Service

Figure 44

The malware can encrypt a specific folder using the "-i" parameter and load the RSA public key from a file mentioned in the "-k" parameter. The ransom note can be changed with a file mentioned in the "-t" parameter.

The C drive is not encrypted if it runs with the "-s" parameter, and the sample doesn't stop the target processes/services and doesn't delete the Volume Shadow Copies if the "-p" parameter is specified:

Command Prompt	
C:\Users\User\Desktop≻malware.exe -s -p start :exclude systemdrive	-v
:do not use preprocess :initial run powershell from predefined	variable
default key:0 preprocess :System	
encrypt system C:∖	
Total time:0	

Figure 45

The ransomware can execute a PowerShell script using the "-w" parameter, as highlighted in the figure below.



E112	00A8499C 00A8499D 00A8499D 00A8499B 00A849A5 00A849A5 00A849A5 00A849A5 00A849A5 00A849A5 00A849A5 00A849A5 00A849A5 00A849A5 00A849B2 00A849C 00A849C 00A849C 00A849C 00A849C 00A849C 00A849C 00A849C 00A849C	50 6A 00 6A 00 68 00000008 6A 01 6A 00 6A 00 6FB5 3CEFFFFF 0F1185 A8EFFFFF 6G:0F1385 4CEFFFFF 6G:0F1385 4CEFFFFF 6G:0F1385 5CEFFFFF 6G:0F1385 5CEFFFFF 6G:0F1385 574EFFFFFF	push eax lea eax,dword ptr ss: [ebp-109 or dword ptr ss: [ebp-1094],1 push eax push 0 push 8000000 push 1 push 0 push 1 push 0 push dword ptr ss: [ebp-108 mov1pd qword ptr ss: [ebp-108] mov1pd qword ptr ss: [ebp-108]	41, xmm0 00 05 81, xmm0 41, xmm0 41, xmm0 41, xmm0 41, xmm0 7, xmm0 44	[ebp-10C4]:"powershe	ST(1) 00000000000 ST(2) 0000000000 ST(3) 0000000000 ST(5) 0000000000 ST(5) 0000000000 ST(5) 00000000000 ST(7) 00000000000 X87Tw_0 3 (Empty) X87Tw_2 3 (Empty) X87Tw_4 3 (Empty) X87Tw_6 3 (Empty	10
.text:00AB4 .text:00AB4 .text:00A5F980 70 00A5F980 75 00A5F980 70 00A5F980 70 00A5F980 70	#9FE malware.exe #9FE Dump 2 ### 0 6F 77 65 72 73 0 6F 77 65 72 73 73 20 46 69 0 73 20 146 44 47 14 40 47 47	\$149FE #13DFE Dump 3 U Dump 4 U Dump 4 U Dump 3 U Dump 4	<pre>=<kernel32.createprocessa> Dump 5</kernel32.createprocessa></pre>	2 Struct	^	00A5492C 00A54930 00A54930 00A54934 00A54934 00A54938 0000000 00A54930 00A54930 00A54944 0000000 00A54944 0000000 00A54945 00A54958	"powershell -execution

The last parameter, "-V," displays the Medusa ransomware version:

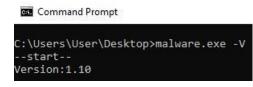


Figure 47

Indicators of Compromise

Medusa Ransom Note

!!!READ_ME_MEDUSA!!!.txt

Processes spawned

powershell -Command "& {}"

net stop <service name> /y

- taskkill /F /IM <process name> /T
- vssadmin Delete Shadows /all /quiet

vssadmin resize shadowstorage /for=C: /on=C: /maxsize=401MB

vssadmin resize shadowstorage /for=C: /on=C: /maxsize=unbounded

cmd /c ping localhost -n 3 > nul & del <Executable>

Appendix

List of services

"Acronis VSS Provider" "Enterprise Client Service" "Sophos Agent" "Sophos AutoUpdate Service" "Sophos Clean Service" "Sophos Device Control Service" "Sophos File Scanner Service" "Sophos Health Service" "Sophos MCS Agent" "Sophos MCS Client" "Sophos Message Router" "Sophos Safestore Service" "Sophos System Protection Service" "Sophos Web Control Service" "SQLsafe Backup Service" "SQLsafe Filter Service" "Symantec System Recovery" "Veeam Backup Catalog Data Service" "AcronisAgent" "AcrSch2Svc" "Antivirus" "ARSM" "BackupExecAgentAccelerator" "BackupExecAgentBrowser" "BackupExecDeviceMediaService" "BackupExecJobEngine" "BackupExecManagementService" "BackupExecRPCService" "BackupExecVSSProvider" "bedbg" "DCAgent" "EPSecurityService" "EPUpdateService" "EraserSvc11710" "EsgShKernel" "FA_Scheduler" "IISAdmin" "IMAP4Svc" "macmnsvc" "masvc" "MBAMService" "MBEndpointAgent" "McAfeeEngineService" "McAfeeFramework" "McAfeeFrameworkMcAfeeFramework" "McShield" "McTaskManager" "mfemms" "mfevtp" "MMS" "mozyprobackup" "MsDtsServer" "MsDtsServer100" "MsDtsServer110" "MSExchangeES" "MSExchangeIS" "MSExchangeMGMT" "MSExchangeMTA" "MSExchangeSA" "MSExchangeSRS" "MSOLAP\$SQL_2008" "MSOLAP\$SYSTEM_BGC" "MSOLAP\$TPS" "MSOLAP\$TPSAMA" "MSSQL\$BKUPEXEC" "MSSQL\$ECWDB2" "MSSQL\$PRACTICEMGT" "MSSQL\$PRACTTICEBGC" "MSSQL\$PROFXENGAGEMENT" "MSSQL\$SBSMONITORING" "MSSQL\$SHAREPOINT" "MSSQL\$SQL_2008" "MSSQL\$SYSTEM_BGC" "MSSQL\$TPS" "MSSQL\$TPSAMA" "MSSQL\$VEEAMSQL2008R2" "MSSQL\$VEEAMSQL2012" "MSSQLFDLauncher" "MSSQLFDLauncher\$PROFXENGAGEMENT" "MSSQLFDLauncher\$SBSMONITORING" "MSSQLFDLauncher\$SHAREPOINT" "MSSQLFDLauncher\$SQL_2008" "MSSQLFDLauncher\$SYSTEM_BGC" "MSSQLFDLauncher\$TPS" "MSSQLFDLauncher\$TPSAMA" "MSSQLSERVER" "MSSQLServerADHelper100" "MSSQLServerOLAPService" "MySQL80" "MySQL57" "ntrtscan" "OracleClientCache80" "PDVFSService" "POP3Svc" "ReportServer" "ReportServer\$SQL_2008" "ReportServer\$SYSTEM_BGC" "ReportServer\$TPS" "ReportServer\$TPSAMA" "RESvc" "sacsvr" "SamSs" "SAVAdminService" "SAVService" "SDRSVC" "SepMasterService" "ShMonitor" "Smcinst" "SmcService" "SMTPSvc" "SNAC" "SntpService" "sophossps" "SQLAgent\$BKUPEXEC" "SQLAgent\$ECWDB2" "SQLAgent\$PRACTTICEBGC" "SQLAgent\$PRACTTICEMGT" "SQLAgent\$PROFXENGAGEMENT" "SQLAgent\$SBSMONITORING" "SQLAgent\$SHAREPOINT" "SQLAgent\$SQL_2008" "SQLAgent\$SYSTEM_BGC" "SQLAgent\$TPS" "SQLAgent\$TPSAMA" "SQLAgent\$VEEAMSQL2008R2" "SQLAgent\$VEEAMSQL2012" "SQLBrowser" "SQLSafeOLRService" "SQLSERVERAGENT" "SQLTELEMETRY" "SQLTELEMETRY\$ECWDB2" "SQLWriter" "SstpSvc" "svcGenericHost" "swi_filter" "swi_service" "swi_update_64" "TmCCSF" "tmlisten" "TrueKey" "TrueKeyScheduler" "TrueKeyServiceHelper" "UI0Detect" "VeeamBackupSvc" "VeeamBrokerSvc" "VeeamCatalogSvc" "VeeamCloudSvc" "VeeamDeploymentService" "VeeamDeploySvc" "VeeamEnterpriseManagerSvc" "VeeamMountSvc" "VeeamNFSSvc" "VeeamRESTSvc" "VeeamTransportSvc" "W3Svc" "wbengine" "WRSVC" "MSSQL\$VEEAMSQL2008R2" "SQLAgent\$VEEAMSQL2008R2" "VeeamHvIntegrationSvc" "swi_update" "SQLAgent\$CXDB" "SQLAgent\$CITRIX_METAFRAME" "SQL Backups" "MSSQL\$PROD" "Zoolz 2 Service" "MSSQLServerADHelper" "SQLAgent\$PROD" "msftesql\$PROD" "NetMsmqActivator" "EhttpSrv" "ekrn" "ESHASRV" "MSSQL\$SOPHOS" "SQLAgent\$SOPHOS" "AVP" "kInagent" "MSSQL\$SQLEXPRESS" "SQLAgent\$SQLEXPRESS" "wbengine" "kavfsslp" "KAVFSGT" "KAVFS" "mfefire"

List of processes

"zoolz.exe" "agntsvc.exe" "dbeng50.exe" "dbsnmp.exe" "encsvc.exe" "excel.exe" "firefoxconfig.exe" "infopath.exe" "isqlplussvc.exe" "msaccess.exe" "msftesql.exe" "mspub.exe" "mydesktopqos.exe" "mydesktopservice.exe" "mysqld.exe" "mysqld-nt.exe" "mysqld-opt.exe" "ocautoupds.exe" "ocomm.exe" "ocssd.exe" "onenote.exe" "oracle.exe" "outlook.exe" "powerpnt.exe" "sqbcoreservice.exe" "sqlagent.exe" "sqlbrowser.exe" "sqlservr.exe" "sqlwriter.exe" "steam.exe" "synctime.exe" "tbirdconfig.exe" "thebat.exe" "thebat64.exe" "thunderbird.exe" "visio.exe" "winword.exe" "wordpad.exe" "xfssvccon.exe" "tmlisten.exe" "PccNTMon.exe" "CNTAoSMgr.exe" "Ntrtscan.exe" "mbamtray.exe"