Cobalt Strike: Using Known Private Keys To Decrypt Traffic – Part 2

blog.nviso.eu/2021/10/27/cobalt-strike-using-known-private-keys-to-decrypt-traffic-part-2/

October 27, 2021

h.



Blogpost series: <u>Cobalt Strike: Decrypting Traffic</u> *We decrypt Cobalt Strike traffic using one of 6 private keys we found.*

In this blog post, we will analyze a Cobalt Strike infection by looking at a full packet capture that was taken during the infection. This analysis includes decryption of the C2 traffic.

If you haven't already, we invite you to read part 1 first: Cobalt Strike: Using Known Private Keys To Decrypt Traffic – Part 1.

For this analysis, we are using capture file <u>2021-02-02-Hancitor-with-Ficker-Stealer-and-Cobalt-Strike-and-NetSupport-RAT.pcap.zip</u>, this is one of the many malware traffic capture files that Brad Duncan shares on his web site <u>Malware-Traffic-Analysis.net</u>.

We start with a minimum of knowledge: the capture file contains encrypted HTTP traffic of a Cobalt Strike beacon communicating with its team server.

If you want to know more about Cobalt Strike and its components, we highly recommend the following blog post.

First step: we open the capture file with Wireshark, and look for downloads of a full beacon by stager shellcode.

Although beacons can come in many forms, we can identify 2 major categories:

- 1. A small piece of shellcode (a couple of hundred bytes), aka the stager shellcode, that downloads the full beacon
- 2. The full beacon: a PE file that can be reflectively loaded

In this first step, we search for signs of stager shellcode in the capture file: we do this with the following display filter: *http.request.uri matches* "/....\$".

2021-02-02-Hancitor-with-Ficker-Stealer-and-Cobalt-Strike-and-NetSupport-RAT.pcap

<u>F</u> il	e <u>E</u> dit	<u>V</u> iew	<u>G</u> o	<u>C</u> apture	<u>A</u> nalyze	<u>S</u> tatist	tics Telephon	<u>y</u> w	<u>/</u> ireless <u>T</u> ools <u>H</u> e	lp						
		۲	010	🗙 🔁	۹ 🗢 🖻) 🖻 .	Ŧ 🕹 📃 🛛		0,0,0, 🎹							
	http.requ	iest.uri i	matches	"/\$"												
No.		Time					Source		Destination	Protocol	Length	Stream index	Info	1		
	6034	2021	-02-02	17:24:	31 ,70 5199		10.2.2.101		192.254.79.71	HTTP	251	129	GET	/EbHm HTTP/1.1		
>	Frame (5034:	251 b	vtes on	wire (200	98 bit	ts), 251 byt	es ci	aptured (2008 bi	ts)						
				·), Dst: Netgear	· · · · · · · · · · · · · · · · · · ·	(20:e5	:2a:b6:93:f	1)			
>	Interne	et Pro	otocol	Version	n 4, Src:	10.2.	2.101, Dst:	192	.254.79.71							
						Port	:: 49801, Ds	t Por	rt: 8080, Seq: 1	, Ack: 1	, Len:	197				
~				r Proto											Fig	gure 1:
				/1.1\r\r	-											
						ible;	MSIE 9.0; N	lindo	ows NT 6.1; WOW6	4; Tride	nt/5.0;	FunWebProd	ucts))\r\n		
				79.71:80												
				ep-Alive												
	Cach	ne-Con	ntrol:	no-cach	ne/r/n											
	\r\r	ı														
	[Fu]	ll red	uest l	URI: htt	tp://192.2	54.79	.71:8080/Eb	lm]								
	[HT1	FP req	uest 1	1/1]												
	[Res	sponse	e in fr	rame: 67	703]											
_																

packet capture for Cobalt Strike traffic

We have one hit. The path used in the GET request to download the full beacon, consists of 4 characters that satisfy a condition: the bytevalue of the sum of the character values (aka checksum 8) is a known constant. We can check this with the tool <u>metatool py</u> like this:

@NVISO_Labs C:\Demo>echo http://example.com/EbHm | metatool.py url8 URL: http://example.com/EbHm path: EbHm checksum: URI_CHECKSUM_INITW / CS x86 (0x5c) @NVISO_Labs C:\Demo>

Figure 2: using metatool.py

More info on this checksum process can be found <u>here</u>. The output of the tool shows that this is a valid path to download a 32-bit full beacon (CS x86). The download of the full beacon is captured too:

🥖 2021-0																							
	2-02-Han	citor-wi	th-Ficke	r-Steale	er-and	l-Cobal	t-Strike	-and-Ne	tSuppo	ort-RA	T.pcap									_			×
<u>F</u> ile <u>E</u> dit	<u>V</u> iew	<u>G</u> o <u>(</u>	apture	<u>A</u> naly	ze	<u>S</u> tatisti	s Tel	ephon <u>y</u>	<u>W</u> ire	less	<u>T</u> ools	<u>H</u> elp											
	•	010	2	۹ 🤃	•	🖭 👔	- U		Ð		. 🏨												
http						_				-											× -		+
No.	Time						Source		D	estina	tion		Protocol	Length	Stream in	dev	Info					-	
	4 2021-0	22-02	17.24.3	31 705	199		10.2.2	0 101	-		54.79.		HTTP	251	Sucanni			EbHm HT	TP/1	1			_
	5 2021-0							.78.68			2.101		нттр	1317				1.1 200		-			
	7 2021-0			-			10.2.2				.252.4		нттр	278				format		нттр/	/1.1		
	1 2021-0							.252.4			2.101		нттр	239				1.1 200				1)	
	3 2021-0			-			192.25	54.79.7	1 10	0.2.1	2.101		HTTP	714				1.1 200		(· /	
675	9 2021-0	02-02	17:24:3	,024	763		10.2.2	2.101	19	92.2	54.79.	71	нттр	429				otj HTT					
676	5 2021-0	02-02	17:24:3	34,341	254		192.25	54.79.7	1 1	0.2.3	2.101		нттр	168		132	HTTP/	1.1 200	ОК				
683	5 2021-0	02-02	17:25:0	97,682	258		10.2.2	2.101	1	85.1	00.65.	29	тср	65		134	49806	→ 80 [I	PSH,	ACK]	Seq=36	5	~
> Etherr > Interr	et II, et Prot	Src: N	letgear /ersion	_b6:93	3:f1 rc: 1	(20:e	5:2a:1 4.79.3	b6:93:f 71, Dst	1), D : 10.	st: 2.2.	Hewlet 101	-	:47:ae)						
> Etherr > Interr > Transm > [156 F	net II, net Prot nission Reassemb	Src: M cocol N Contro oled TC	Vetgear Version 1 Prot 2P Segm	b6:93 0 4, S0 cocol, 0ents	3:f1 rc: 1 Src	(20:e 192.25 Port:	5:2a:1 4.79.7 8080	b6:93:f 71, Dst , Dst P	1), D : 10. ort:	st: 2.2. 4980	Hewlet 101 1, Sec	ttP_1 q: 20	2:47:ae	ck: 198	, Len:	660		(1460),	, #60	59(14	60),#	6066	948
 Ethern Intern Transm [156 F Hypert 	net II, net Prot nission	Src: M cocol N Contro oled TC ansfer	Vetgear Version D1 Prot IP Segm Protoc	b6:93 0 4, S0 cocol, 0ents	3:f1 rc: 1 Src	(20:e 192.25 Port:	5:2a:1 4.79.7 8080	b6:93:f 71, Dst , Dst P	1), D : 10. ort:	st: 2.2. 4980	Hewlet 101 1, Sec	ttP_1 q: 20	2:47:ae	ck: 198	, Len:	660		(1460),	, #60	59(14	60),#	6066	948)
 Ethern Intern Transm [156 F Hypert 	net II, net Prot nission Reassemb cext Tra	Src: M cocol N Contro oled TC ansfer	Vetgear Version D1 Prot IP Segm Protoc	b6:93 0 4, S0 cocol, 0ents	3:f1 rc: 1 Src	(20:e 192.25 Port:	5:2a:1 4.79.7 8080	b6:93:f 71, Dst , Dst P	1), D : 10. ort:	st: 2.2. 4980	Hewlet 101 1, Sec	ttP_1 q: 20	2:47:ae	ck: 198	, Len:	660		(1460),	, #60	59(14	60), #	6066	948
 Etherr Interr Transm [156 F Hypert Data (net II, net Prot nission Reassemb cext Tra	Src: M cocol N Contro oled TC ansfer	Vetgear Version D1 Prot IP Segm Protoc	b6:93 0 4, S0 cocol, 0ents	3:f1 rc: 1 Src	(20:e 192.25 Port:	5:2a:1 4.79.7 8080	b6:93:f 71, Dst , Dst P	1), D : 10. ort:	st: 2.2. 4980	Hewlet 101 1, Sec	ttP_1 q: 20	2:47:ae	ck: 198	, Len:	660		3(1460),	, #60	59(14	60), #	6066	9(948)
<pre>> Etherr > Interr > Transm > [156 F > Hypert > Data (</pre>	net II, net Prot nission Reassemb cext Tra 208452	Src: M cocol N Contro oled TC ansfer bytes)	Vetgear Version DI Prot Protoc	b6:9: 4, Si cocol, eents col	3:f1 rc: 1 Src (2085	(20:e 192.25 Port: 571 by	5:2a:1 4.79. 8080 tes):	b6:93:f 71, Dst , Dst F #6053(1), D : 10. ort: 1460)	st: 2.2. 4980 , #6	Hewlet 101 1, Sec 054(14	ttP_1 q: 20 460),	c:47:ae 7912, A #6055(ck: 198	, Len:	660		;(1460),	, #60	59(14	60), #	6066	ð(948) >
<pre>> Etherr > Interr > Transm > [156 F > Hypert > Data (00000070</pre>	et II, net Prot ission ceassemb cext Tra 208452 34 35	Src: N cocol N Contro bled TC ansfer bytes) 32 0d	letgear /ersion Dl Prot Protoc	b6:9: 0 4, 50 cocol, 0 ents col	3:f1 rc: 1 Src (2085	(20:e 192.25 Port: 571 by	5:2a:1 4.79.7 8080 tes):	b6:93:f 71, Dst , Dst P #6053(1), D : 10. ort: 1460) a e2	st: 2.2. 4980 , #6	Hewlet 101 1, Seq 054(14	ttP_1 q: 20 460),	c:47:ae 7912, A #6055(ck: 198	, Len:	660		;(1460),	, #60	59(14	60), #	6060)(948 > ^
<pre>> Etherr > Interr > Transm > [156 F > Hypert > Data (00000070 000000000</pre>	et II, net Prot nission Reassemb ext Tra 208452 34 35 81 eb	Src: N cocol V Contro oled TC msfer bytes) 32 0d 2b 5d	letgear /ersion)l Prot Protoc Protoc 0 0a 0d 8b 55	b6:9: 4, Si cocol, eents col 0a fo 00 83	3:f1 rc: 1 Src (2085 (2085 (2085 (2085 (2085) (2085	(20:e 192.25 Port: 571 by 3 04 0 5 04 8	5:2a:1 4.79.7 8080 tes): 0 00 0 b 45 0	b6:93:f 71, Dst , Dst F #6053(1), D : 10. vort: 1460) a e2 0 83	4980 , #6	Hewlet 101 1, Sec 054(14	ttP_1 q: 20 460),	:47:ae 7912, A #6055(.B:. :1	ck: 198	, Len:	660		(1460),	, #60	59(14	60), #	6066	>
<pre>> Ethern > Intern > Intern > Transm > [156 F > Hypert > Data (00000070 00000080 00000080 00000080</pre>	et II, et Prot ission leassemb ext Tra 208452 34 35 81 eb c5 04 83 e8	Src: N cocol V Contro oled TC msfer bytes) 32 0d 2b 5d 55 8b 04 31	letgear /ersion)l Prot Protoc Protoc 0 0a 0d 8b 55 4d 00 c9 39	b6:9: 0 4, Si cocol, ments col 0 a fo 00 8 31 d1 c8 74	3:f1 rc: 1 Src (2085 (2085 (2085 (2085 (2085 (2085)	(20:e 192.25 Port: 571 by 3 04 0 5 04 8 9 4d 0 2 eb e	5:2a:1 4.79.7 8080 tes): 0 00 0 b 45 0 0 31 0 8 5a f	b6:93:f 71, Dst , Dst F #6053(00 42 3 00 31 d ca 83 c ff e2 e	1), D : 10. ort: 1460) a e2 0 83 5 04 8 d0	4980 , #6	Hewlet 101 1, Sec 054(14 2	ttP_1 q: 20 460),	:47:ae 7912, A #6055(ck: 198	, Len:	660		(1460),	, #60	59(14	60), #	6066)(948) >
<pre>> Ethern > Intern > Intern > Transm > [156 F > Hypert > Data (000000070 000000000 000000000 00000000</pre>	et II, et Prot ission leassemb ext Tra 208452 34 35 81 eb c5 04 83 e8	Src: N cocol V Contro oled TC msfer bytes) 32 0d 2b 5d 55 8b 04 31	letgear /ersion)l Prot Protoc Protoc 0 0a 0d 8b 55 4d 00 c9 39	b6:9: 0 4, Si cocol, ments col 0 a fo 00 8 31 d1 c8 74	3:f1 rc: 1 Src (2085 (2085 (2085 (2085 (2085 (2085)	(20:e 192.25 Port: 571 by 3 04 0 5 04 8 9 4d 0 2 eb e	5:2a:1 4.79.7 8080 tes): 0 00 0 b 45 0 0 31 0 8 5a f	b6:93:f 71, Dst , Dst F #6053(00 42 3 00 31 d ca 83 c	1), D : 10. ort: 1460) a e2 0 83 5 04 8 d0	4980 , #6	Hewlet 101 1, Sec 054(14 2 +].U U.M.1.	ttP_1 q: 20 460),	.B:.	ck: 198	, Len:	660		(1460),	, #60	59(14	60), #	:6066	9(948) > ^
<pre>> Ethern > Intern > Intern > Transm > [156 F > Hypert > Data (00000070 00000080 00000080 00000080</pre>	at II, at Prot ission Reassemb cext Tra 208452 34 35 81 eb c5 04 83 e8 ff ff	Src: N cool V Contro oled TC insfer bytes) 32 0d 2b 5d 55 8b 04 31 ff 5f	letgear /ersion)l Prot Protoc Protoc 0 0a 0d 8b 55 4d 00 c9 39	b6:9: 0 4, 50 0 0 50 0 1 0 8 3 1 0 0 8 5 1 0 0 0 8 5 1 0 0 0 8 5 1 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	3:f1 rc: 1 Src (2085 (2085 (2085 (2085 (2085 (2085)	(20:e 192.25 Port: 571 by 04 0 04 8 04d 0 2 eb e 197 c	5:2a:1 4.79.7 8080 tes): 0 00 0 b 45 0 0 31 0 8 5a f	b6:93:f 71, Dst , Dst F #6053(00 42 3 00 31 d ca 83 c ff e2 e	1), D : 10. ort: 1460) a e2 0 83 5 04 8 d0	4980 , #6	Hewlet 101 1, Sec 054(14 2 +].U U.M.1. .1.9.t	ttP_1 q: 20 460),	.B:.	ck: 198	, Len:	660		(1460),	, #60	59(14	60), #		>
<pre>> Etherr > Interr > Transm > [156 F > Hypert > Data (00000070 00000080 00000000 00000000 Frame (71*</pre>	at II, at Prot ission Reassemb cext Tra 208452 34 35 81 eb c5 04 83 e8 ff ff	Src: N cocol V Contro bled TC ansfer bytes) 32 0d 2b 5d 55 8b 04 31 ff 5f Reass	Version Protoco Protoco Magnetic Protoco Magnetic Magneti	b6:9: 0 4, 50 0 0 50 0 1 0 8 3 1 0 0 8 1 0 0 8 1 0 0 8 1 0 0 8 1 0 0 8 1 0 0 1 0 0 8 1 0 0 1 0 0 0 1 0 0 0 0	3:f1 rc: 1 Src (2085 (2085 (2085 (2085 (2085 (2085)	(20:e 192.25 Port: 571 by 04 0 04 8 04d 0 2 eb e 197 c	5:2a:1 4.79.7 8080 tes): 0 00 0 b 45 0 0 31 0 8 5a f	b6:93:f 71, Dst , Dst F #6053(00 42 3 00 31 d ca 83 c ff e2 e	1), D : 10. ort: 1460) a e2 0 83 5 04 8 d0	4980 , #6	Hewlet 101 1, Sec 054(14 2 +].U U.M.1. .1.9.t	ttP_1 q: 20 460),	.B:.	ck: 198 1195),	, Len: #6057(:	660 L460),	#6058	(1460),			60), #		>

Figure 3: full beacon download

And we can extract this download:

_)2-Hancitor-with <u>V</u> iew <u>G</u> o <u>C</u> a		Statistics	Telephony			lelp				
<u>O</u> pen		Ctrl+O	E 👔		•	R. Q. 🎹					
Open <u>R</u> e	ecent	,									
Merge			s	ource	De	stination	Pro				
Import f	from Hex Dump		1	0.2.2.101	19	2.254.79.71	HTT				
Close		Ctrl+W	8	.209.78.68	3 10	.2.2.101	нтт				
				0.2.2.101		.21.252.4	HTT				
<u>S</u> ave		Ctrl+S		3.21.252.4		.2.2.101	HTT				
Save <u>A</u> s	5	Ctrl+Shift+S		92.254.79.		.2.2.101	HTT				
F 1 A A				0.2.2.101 92.254.79.		2.254.79.71 .2.2.101	НТ1 НТ1				
File Set		•		0.2.2.101		5.100.65.29	TCF				
Export	Specified Packets			0.2.2.101		5.100.65.29	TCF	Figure 4	: expor	HTTF	obj
	Packet Dissection		. 1	0.2.2.101		5.100.65.29	TCF	, ,			
			1	0.2.2.101	18	5.100.65.29	TCF				
	Packet <u>B</u> ytes	Ctrl+Shift+X	1	0.2.2.101	18	5.100.65.29	TCF				
Export F	PDUs to File			0.2.2.101		5.100.65.29	TCF				
Export 7	TLS Session Keys.		1	0.2.2.101		5.100.65.29	TCF				
Export (Objects	,	•	DICOM		5.100.65.29	TCF				
-	,			HTTP		5.100.65.29	TCF				
<u>P</u> rint		Ctrl+P		2		5.100.65.29 5.100.65.29	TCF TCF				
0.1		01.0		MF		2.100.05.29 2.254 70 71					
Quit	(NO: /14 UVIE	Ctrl+Q	-	5MB		ured (5712 b					
		tgear_b6:93:f1	-	TFTP		st: HewlettP					
Tasaani		Postorel Con	192.254	.79.71, Ds	st: 10.2	2.2.101	20701				~
Wireshar	rk · Export · HTTP	Destanal Car	192.254 Deet	.79.71, Ds	st: 10.2	2.2.101	20701	Content Type		nt-Types	×
Wireshar	rk · Export · HTTP	Destanal Car	192.254	9000 D-+	ze	Filename	20201	Content Type			
Wireshar Fext Filter:	rk · Export · HTTP ostname	object list Content Type	Dark.	Si	Danta (Filename	20201	Content Type			; ~
Wireshar ext Filter: Packet Ho 12128 192	rk • Export • HTTP ostname 12.254.79.71:8080	object list	Seet .	Si 10	ze ×	10001 C	20201	Content Type	- All Cont		; ~
Wireshar Text Filter: [Packet Ho 12128 192 13055 192	rk · Export · HTTP ostname 2.254.79.71:8080	object list Content Type application/octet-s	stream stream	Si 10 10	ze 40kB	Filename ptj	20201	Content Type			; ~
Wireshar ext Filter: [Packet Ho 12128 19/ 13055 19/ 13891 19/ 1859 pre	ostname 12,254,79,71:8080 12,254,79,71:8080 12,254,79,71:8080 12,254,79,71:8080 12,254,79,71:8080 12,254,79,71:8080 12,254,79,71:8080	object list Content Type application/octet-s application/octet-s text/html	stream stream	5000 Det Si 10 10 93	ze 40kB 40kB 33kB	Filename ptj	30301	Content Type			; ~
Wireshar ext Filter: [Packet Ho 12128 19/ 13055 19/ 13891 19/ 1859 pre 11175 19/	rk · Export · HTTP ostname 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080	object list Content Type application/octet-s application/octet-s text/html application/octet-s	stream stream stream stream	Since Det Since Si	ze 40kB 40kB 33kB 11kB 36kB	Filename ptj ptj ptj fotage.php ptj	202001	Content Type	 : All Contr		; ~
Wireshar ext Filter: [Packet Ho 12128 19; 13055 19; 13891 19; 13891 19; 1859 pro 11175 19; 6355 bo	rk · Export · HTTP ostname 2.254.79.71:8080 2.254.79.71:8080 2.254.79.71:8080 emierpt.co.uk 2.254.79.71:8080 obcatofreddin	object list Content Type application/octet-s application/octet-s ext/hml application/octet-s application/octet-s	stream stream stream stream stream	Sin 10 10 10 10 10 10 10 10 10 10 10 10 10 1	ze 40kB 40kB 33kB 11kB 96kB 73kB	Filename ptj ptj ptj footage.php ptj 6lavfdk.exe	202001	Content Type	 All Contr		; ~
Turner Wireshar 'ext Filter: [Packet Hc 12128 19/ 13055 19/ 13891 19/ 1859 previous 11175 19/ 6355 bo 6703 19/	rk · Export · HTTP ostname 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 remierpt.co.uk 12.254.79.71:8080 bocatofreddin 12.254.79.71:8080	object list Content Type application/octet-s application/octet-s text/html application/octet-s application/octet-s application/octet-s	stream stream stream stream stream stream	Sin 10 10 93 75 27 20 20	ze 40kB 40kB 3kB 11kB 36kB 73kB 38kB	Filename ptj ptj footage.php ptj folavfdk.exe EbHm		Content Type	 All Contr		; ~
Text Filter: [Packet Hot 12128 19/ 13055 19/ 13891 19/ 1859 provide 11175 19/ 6355 boo 66703 19/ 82323 19/	rk · Export · HTTP ostname 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 emierpt.co.uk 12.254.79.71:8080 bocatofreddin 12.254.79.71:8080 12.254.79.71:8080	Content Type application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s	stream stream stream stream stream stream	Sin 10 10 93 79 78 27 20 75 20 75	ze 40kB 40kB 33kB 95kB 95kB 98kB 98kB	Filename ptj ptj footage.php ptj footage.php ptj 6lavfdk.exe EbHm ptj			- All Cont		; ~
Text Filter: [Packet Hot 12128 19/ 13055 19/ 13891 19/ 1859 prov 11175 19/ 6355 boo 6703 19/ 82323 19/ 9723 19/	rk · Export · HTTP ostname 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080	Content Type application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s	stream stream stream stream stream stream stream	Since Dett Si 10 10 93 75 75 75 20 75 10	ze 40kB 40kB 33kB 11kB 56kB 73kB 56kB 73kB 56kB 73kB 56kB 18kB	Filename ptj ptj footage.php ptj folavfdk.exe EbHm ptj submit.php?id		267	- All Cont		; ~
Toron Image: Construction of the second	rk · Export · HTTP ostname 22.254.79.71:8080 22.254.79.71:8080 22.254.79.71:8080 22.254.79.71:8080 22.254.79.71:8080 22.254.79.71:8080 22.254.79.71:8080 22.254.79.71:8080	object list Content Type application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s	stream stream stream stream stream stream stream stream	5000 Det 51 10 10 93 75 75 27 20 75 10 38	ze 40kB 40kB 33kB 11kB 36kB 373kB 88kB 38kB 38kB 38kB 376 bytes	Filename ptj ptj ptj footage.php ptj folavfdk.exe EbHm ptj submit.php?id submit.php?id	=242569	267 267	- All Cont		; ~
Text Filter: [Packet Hc 12128 19; 13055 19; 13891 19; 13895 19; 13895 19; 1355 bo 6703 19; 82323 19; 9723 19; 85020 19; 86029 19;	rk · Export · HTTP ostname 2.254.79.71:8080 2.254.79.71:8080 2.254.79.71:8080 2.254.79.71:8080 2.254.79.71:8080 2.254.79.71:8080 2.254.79.71:8080 2.254.79.71:8080 2.254.79.71:8080	object list Content Type application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s	stream stream stream stream stream stream stream stream stream	5000 Det Si 10 10 93 75 27 20 75 10 38 34	ze 40kB 40kB 33kB 31kB 36kB 378kB 38kB 38kB 38kB 376 bytes 112 bytes	Filename ptj ptj ptj fotatge.php ptj 6lavfdk.exe EbHm ptj submit.php?id submit.php?id submit.php?id	=242569 =242569	267 267 267	 : All Contr		; ~
Torest Filter: [] Packet Hc 12128 19; 13055 19; 13055 19; 13859 prc 11175 19; 6355 bo 6703 19; 82323 19; 9723 19; 85020 19; 85020 19; 85020 19; 85029 19; 13928 19;	rk · Export · HTTP ostname 2.254.79.71:8080 2.254.79.71:8080 2.254.79.71:8080 0:2254.79.71:8080 0:254.79.71:8080 0:254.79.71:8080 2:254.79.71:8080 2:254.79.71:8080 2:254.79.71:8080	object list Content Type application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s	stream stream stream stream stream stream stream stream stream stream	Sin 10 10 10 10 10 10 10 10 10 10 10 10 10 1	ze 440kB 440kB 440kB 43kB 11kB 56kB 73kB 56kB 73kB 58kB 58kB 58kB 58kB 59kB 776 bytes 112 bytes 176 bytes	Filename ptj ptj footage.php ptj 6lavfdk.exe EbHm ptj submit.php?id submit.php?id submit.php?id submit.php?id	=242569 =242569 =242569	267 267 267 267 267	 : All Contr		; ~
Text Filter: [Packet Hc 12128 19; 13055 19; 13891 19; 1859 prc 11175 19; 6355 bo 6703 19; 85020 19; 85020 19; 85020 19; 85020 19; 85020 19; 85021 19; 85721 19;	rk · Export · HTTP ostname 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080	object list Content Type application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s	stream stream stream stream stream stream stream stream stream stream	Sin 10 10 10 93 75 27 20 75 10 32 34 22 22 22	ze 400kB 400kB 400kB 400kB 400kB 33kB 73kB 73kB 73kB 90kB 76 bytes 176 bytes 176 bytes 176 bytes 172 bytes	Filename ptj ptj footage.php ptj 6lavfdk.exe EbHm ptj submit.php?id submit.php?id submit.php?id submit.php?id submit.php?id	=242569 =242569 =242569	267 267 267 267 267	 : All Contr		; ~
Text Filter: [Packet Hc 12128 19; 13055 19; 13891 19; 1859 prc 11175 19; 6355 bo 6703 19; 85020 19; 85020 19; 85020 19; 85020 19; 85020 19; 85021 19; 85721 19;	rk · Export · HTTP ostname 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080	object list Content Type application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s	stream stream stream stream stream stream stream stream stream stream	Sin 10 10 10 10 10 10 10 10 10 75 10 10 38 34 22 22 15	ze 440kB 440kB 440kB 43kB 11kB 56kB 73kB 56kB 73kB 58kB 58kB 58kB 58kB 59kB 776 bytes 112 bytes 176 bytes	Filename ptj ptj footage.php ptj 6lavfdk.exe EbHm ptj submit.php?id submit.php?id submit.php?id submit.php?id submit.php?id	=242569 =242569 =242569	267 267 267 267 267	 All Contr		; ~
Text Filter: [Packet Hc 12128 19/ 13055 19/ 13055 19/ 13891 19/ 1859 pri 11175 19/ 6355 bo 6703 19/ 82021 19/ 9723 19/ 86099 19/ 13928 19/ 85721 19/ 82721 19/ 82721 19/ 82721 19/ 82721 19/ 82721 19/ 82721 19/ 8271 19/ 8271 19/ 6915 19/	rk · Export · HTTP ostname 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080	object list Content Type application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s	stream stream stream stream stream stream stream stream stream stream	Sin 10 10 10 10 10 10 10 10 10 10 10 10 10 1	ze 40kB 40kB 40kB 40kB 33kB 11kB 56kB 33kB 56kB 33kB 576 bytes 112 bytes 112 bytes 112 bytes 552 bytes	Filename ptj ptj footage.php ptj 6lavfdk.exe EbHm ptj submit.php?id submit.php?id submit.php?id submit.php?id submit.php?id	=242569 =242569 =242569	267 267 267 267 267	– All Contr		; ~
Toron i Wireshar ['ext Filter: [Packet Hc 12128 19; 13055 19; 13891 19; 13859 prc 11175 19; 6355 bo 6703 19; 82323 19; 9723 19; 85020 19; 85020 19; 85721 19; 82027 19; 6915 7712	rk · Export · HTTP ostname 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080	object list Content Type application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s	stream stream stream stream stream stream stream stream stream stream	5000 Det 51 10 10 10 10 10 75 75 27 20 75 10 38 34 22 22 15 14 14 14	ze 400kB 400kB 33kB 96kB 96kB 98kB 98kB 98kB 98kB 98kB 98kB 99kB 976 bytes 776 bytes	Filename ptj ptj footage.php ptj 6lavfdk.exe EbHm ptj submit.php?id submit.php?id submit.php?id submit.php?id submit.php?id	=242569 =242569 =242569	267 267 267 267 267	- All Contr		; ~
Text Filter: [Packet Hc 12128 19; 13055 19; 13055 19; 13851 19; 1859 prc 11175 19; 6355 bo 6703 19; 85020 19; 85020 19; 85020 19; 82027 19; 82721 19; 82027 19; 6915 7712 7913 7917	rk · Export · HTTP ostname 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080	object list Content Type application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s	stream stream stream stream stream stream stream stream stream stream	Since Det Si 10 10 10 93 75 27 20 75 10 0 38 34 22 22 15 14 14 14 14	ze 440kB 440kB 440kB 43kB 73kB 73kB 73kB 73kB 73kB 73kB 73kB 7	Filename ptj ptj footage.php ptj 6lavfdk.exe EbHm ptj submit.php?id submit.php?id submit.php?id submit.php?id submit.php?id	=242569 =242569 =242569	267 267 267 267 267	- All Contr		; ~
Toron Image: Construct of the second se	rk · Export · HTTP ostname 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080	object list Content Type application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s	stream stream stream stream stream stream stream stream stream stream	Sin 10 10 10 10 10 10 10 10 10 10 10 10 10 1	ze 400kB 400kB 400kB 400kB 400kB 400kB 73kB 73kB 73kB 73kB 73kB 73kB 73kB 73	Filename ptj ptj footage.php ptj 6lavfdk.exe EbHm ptj submit.php?id submit.php?id submit.php?id submit.php?id submit.php?id	=242569 =242569 =242569	267 267 267 267 267	- All Contr		; ~
Toron Image: Construct of the second se	rk · Export · HTTP ostname 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080	object list Content Type application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s	stream stream stream stream stream stream stream stream stream stream	Sin 10 10 10 10 10 10 10 10 10 10 10 10 10 1	ze 400kB 400kB 400kB 400kB 400kB 400kB 400kB 73kB 73kB 73kB 73kB 73kB 73kB 73kB 73	Filename ptj ptj footage.php ptj 6lavfdk.exe EbHm ptj submit.php?id submit.php?id submit.php?id submit.php?id submit.php?id	=242569 =242569 =242569	267 267 267 267 267	 : All Contr		; ~
Text Filter: [Packet Hc 12128 19; 13055 19; 13891 19; 13855 bo 6703 19; 82323 19; 95020 19; 85020 19; 85020 19; 85721 19; 85027 19; 6915 7712 7913 7917 7922 7927 7932 19;	rk · Export · HTTP ostname 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080	object list Content Type application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s	stream stream stream stream stream stream stream stream stream stream	5000 0-4 51 10 10 10 10 10 99 75 75 27 20 75 10 38 34 22 22 25 11 14 14 14 14 14 14	ze 440kB 440kB 33kB 11kB 96kB 33kB 96kB 376 bytes 112 bytes 1252 bytes 120 bytes 160 bytes 160 bytes 160 bytes 160 bytes 160 bytes 160 bytes	Filename ptj ptj footage.php ptj 6lavfdk.exe EbHm ptj submit.php?id submit.php?id submit.php?id submit.php?id submit.php?id	=242569 =242569 =242569	267 267 267 267 267	 : All Contr		; ~
Text Filter: [Packet Hc 12128 19; 13055 19; 13891 19; 1355 bo 6703 19; 82323 19; 9723 19; 85020 19; 85021 19; 82023 19; 80099 19; 13928 19; 82021 19; 6915 7712 7913 7917 7922 7927 7932 7937	rk · Export · HTTP ostname 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080	object list Content Type application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s	stream stream stream stream stream stream stream stream stream stream	Since Det Si Si 10 10 10 93 75 27 20 75 10 38 34 22 27 10 38 34 22 22 15 14 14 14 14 14 14 14 14 14	ze 440kB 440kB 33kB 33kB 33kB 33kB 33kB 33kB 33kB 3	Filename ptj ptj footage.php ptj 6lavfdk.exe EbHm ptj submit.php?id submit.php?id submit.php?id submit.php?id submit.php?id	=242569 =242569 =242569	267 267 267 267 267	 : All Contr		; ~
Text Filter: Packet Ho 12128 199 13055 199 13055 199 13891 192 13859 prer 11175 192 6355 bo 6703 199 82323 192 192 13928 192 85020 192 85020 192 86099 192 86099 192 192 7913 7917 7922 7917 7932 7937 7937 7937	rk · Export · HTTP ostname 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080 12.254.79.71:8080	object list Content Type application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s application/octet-s	stream stream stream stream stream stream stream stream stream stream	Since Det Si Si 10 10 93 75 27 20 75 10 38 34 22 22 15 14 14 14 14 14 14 14 14 14 14	ze 440kB 440kB 440kB 33kB 73kB 86kB 73kB 86kB 73kB 86kB 73kB 73kB 80kB 73kB 80kB 73kB 736 bytes 12 bytes 12 bytes 152 bytes 160 bytes 160 bytes 160 bytes 160 bytes 160 bytes 160 bytes 160 bytes 160 bytes 160 bytes	Filename ptj ptj footage.php ptj 6lavfdk.exe EbHm ptj submit.php?id submit.php?id submit.php?id submit.php?id submit.php?id	=242569 =242569 =242569	267 267 267 267 267	 : All Contr		; ~

ort HTTP objects

Help

Save Save All Preview Close

Figure 5: selecting download

EbHm for saving

Wireshark · Save Object As	×
$\leftarrow \rightarrow \checkmark \uparrow$ $\square \rightarrow$ This PC \rightarrow Boot (C:) \rightarrow demo \checkmark \boxed{O} Search demo	
Organize 🔻 New folder	?
 This PC 3D Objects Desktop Desktop Documents Downloads Music Pictures Videos Secturent SRECYCLE.BIT Config.Msi 	
File name: EbHm.vir	~
Save as type: All Files (*)	~
Hide Folders Save Canc	el

Figure 6: saving selected download to disk

Once the full beacon has been saved to disk as EbHm.vir, it can be analyzed with <u>tool 1768.py</u>. 1768.py is a tool that can decode/decrypt Cobalt Strike beacons, and extract their configuration. Cobalt Strike beacons have many configuration options: all these options are stored in an encoded and embedded table.

Here is the output of the analysis:

@NVISO Labs	>
NVISO_Labs C:\Demo>1768.py EbHm.vir	
wviso_labs C:\Demo≻i/68.py EDHm.vir ∵ile: EbHm.vir	
corkey(chain): 0x94f45fff	
length: 0x032e0033	
Config found: xorkey b'.' 0x00000000	0v00007fof
0x0001 payload type	0x0001 0x0002 0 windows-beacon http-reverse http
0x0002 port	
x0003 sleeptime	0x0002 0x0004 60000
x0004 maxgetsize	0x0002 0x0004 1048576
x0005 jitter	0x0001 0x0002 0
x0007 publickey	0x0003 0x0100 30819f300d06092a864886f70d010101050003818d0030818902818100a738cde75f1fbb1c18646c377e03
	05a95c26170bf908105ad7fa4bbccfa798632261bed9870f975f20794e1fe499523d71f08a56cae0315bfde3d6c8a16386b03b7a
	a73b9fb7c3fb4d7a088e323f07618656ecd83595fa5f82361302030100010000000000000000000000000
	000000000000000000000000000000000000000
0 Has known private key	
x0008 server,get-uri	0x0003 0x0100 '192.254.79.71,/ptj'
x000e SpawnTo	0x0003 0x0010 (NULL)
k001d spawnto x86	0x0003 0x0040 '%windir%\\syswow64\\rundll32.exe'
x001e spawnto x64	0x0003 0x0040 '%windir%\\sysnative\\rundll32.exe'
x001f CryptoScheme	0×0001 0×0002 0
x001a get-verb	0x0003 0x0010 'GET'
001b post-verb	0x0003 0x0010 'POST'
x001c HttpPostChunk	0x0002 0x0004 0
x0025 license-id	0x0002 0x0004 0 trial or pirated? - Stats uniques -> ips/hostnames: 380 publickeys: 218
x0026 bStageCleanup	0x0001 0x0002 0
x0027 bCFGCaution	0x0001 0x0002 0
x0009 useragent	0x0003 0x0100 'Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.0; WOW64; Trident/5.0)'
x000a post-uri	0x0003 0x0040 '/submit.php'
x000b Malleable_C2_Instructions	0x0003 0x0100 '\x00\x00\x00\x04'
x000c http_get_header Cookie	0x0003 0x0200
000d http post header	0x0003 0x0200
&Content-Type: application/octet-s	
id	
0036 HostHeader	0x0003 0x0080 (NULL)
(0032 UsesCookies	0x0001 0x0002 1
<pre><0023 proxy_type</pre>	0x0001 0x0002 2 IE settings
x003a	0x0003 0x0080 '\x00\x04'
(0039	0x0003 0x0080 '\x00\x04'
(0037	0x0001 0x0002 0
x0028 killdate	0x0002 0x0004 0
x0029 textSectionEnd	0x0002 0x0004 0
x002b process-inject-start-rwx	0x0001 0x0002 64 PAGE_EXECUTE_READWRITE
x002c process-inject-use-rwx	0x0001 0x0002 64 PAGE_EXECUTE_READWRITE
x002d process-inject-min_alloc	0x0002 0x0004 0
x002e process-inject-transform-x86	0x0003 0x0100 (NULL)

Figure 7: extracting beacon configuration

Let's take a closer look at some of the options.

First of all, option 0x0000 tells us that this is an HTTP beacon: it communicates over HTTP.

It does this by connecting to 192.254.79[.]71 (option 0x0008) on port 8080 (option 0x0002).

GET requests use path /ptj (option 0x0008), and POST requests use path /submit.php (option 0x000a)

And important for our analysis: there is a known private key (Has known private key) for the public key used by this beacon (option 0x0007).

Thus, armed with this information, we know that the beacon will send GET requests to the team server, to obtain instructions. If the team server has commands to be executed by the beacon, it will reply with encrypted data to the GET request. And when the beacon has to send back output from its commands to the team server, it will use a POST request with encrypted data.

If the team server has no commands for the beacon, it will send no encrypted data. This does not necessarily mean that the reply to a GET request contains no data: it is possible for the operator, through profiles, to masquerade the communication. For example, that the encrypted data is inside a GIF file. But that is not the case with this beacon. We know this, because there are no so-called malleable C2 instructions in this profile: option 0x000b is equal to 0x00000004 -> this means no operations should be performed on the data prior to decryption (we will explain this in more detail in a later blog post).

Let's create a display filter to view this C2 traffic: http and ip.addr == 192.254.79[.]71

	ß	O I	× C	• ج ا		s Telephony			elp						
lo.		o.addr == 192 Time	. 254. 79.	/1		Source	Destir	nation	Protocol	Length	Stream index	Info			
			2 17:24	4:31,70519		10.2.2.101		254.79.71	HTTP	251		GET /E	bHm HT	TP/1.1	
				4:33,73348		192.254.79.71		.2.101	HTTP	714		HTTP/1			
+	6759	2021-02-0	2 17:24	4:34,02476	3	10.2.2.101	192.	254.79.71	HTTP	429	132	GET /p	tj HTT	P/1.1	
_	6765	2021-02-0	2 17:24	4:34,34125	1	192.254.79.71	10.2	.2.101	HTTP	168	132	HTTP/1	.1 200	ОК	
	8056	2021-02-0	2 17:2	5:34,54644	3	10.2.2.101	192.	254.79.71	HTTP	429	139	GET /p	tj HTT	P/1.1	
	8058	2021-02-0	2 17:2	5:34,83399		192.254.79.71	10.2	.2.101	HTTP	168	139	HTTP/1	.1 200	ок	
	8260	2021-02-0	2 17:20	5:35,03820	3	10.2.2.101	192.	254.79.71	HTTP	429	146	GET /p	tj HTT	P/1.1	F : 0
	8262	2021-02-0	2 17:20	5:35,36546	7	192.254.79.71	10.2	.2.101	HTTP	168	146	HTTP/1	.1 200	ок	Figure 8:
	8707	2021-02-0	2 17:2	7:35,58384	5	10.2.2.101	192.	254.79.71	HTTP	429	150	GET /p	tj HTT	P/1.1	
	8709	2021-02-0	2 17:2	7:35,79931	3	192.254.79.71	10.2	.2.101	HTTP	168	150	HTTP/1	.1 200	ок	
	8776	2021-02-0	2 17:28	3:35,97509	3	10.2.2.101	192.	254.79.71	HTTP	429	155	GET /p	tj HTT	P/1.1	
	8778	2021-02-0	2 17:20	3:36,22248) (192.254.79.71	10.2	.2.101	HTTP	168	155	HTTP/1	.1 200	ок	
	8840	2021-02-0	2 17:29	9:36,45278	7	10.2.2.101	192.	254.79.71	HTTP	429	158	GET /p	tj HTT	P/1.1	
	8842	2021-02-0	2 17:29	9:36,77992	3	192.254.79.71	10.2	.2.101	HTTP	168	158	HTTP/1	.1 200	ок	
	8954	2021-02-0	2 17:30	0:37,03211	L	10.2.2.101	192.	254.79.71	HTTP	429	165	GET /p	tj HTT	P/1.1	
	8956	2021-02-0	2 17:30	0:37,33962	2	192.254.79.71	10.2	.2.101	HTTP	168	165	HTTP/1	.1 200	ОК	
	9045	2021-02-0	2 17:3	L:37,50388)	10.2.2.101	192.	254.79.71	HTTP	429	169	GET /p	tj HTT	P/1.1	
	9047	2021-02-0	2 17:3	L:37,77645	L	192.254.79.71	10.2	.2.101	HTTP	168	169	HTTP/1	.1 200	ок	
	9095	2021-02-0	17.3	0.38 03318	1	10 2 2 101	192	25/ 70 71	нттр	//20	172	GET /m	ні нтт	D/1 1	

beacon download and HTTP requests with encrypted Cobalt Strike traffic

This displays all HTTP traffic to and from the team server. Remark that we already took a look at the first 2 packets in this view (packets 6034 and 6703): that's the download of the beacon itself, and that communication is not encrypted. Hence, we will filter these packets out with the following display filter:

http and ip.addr == 192.254.79.71 and frame.number > 6703

This gives us a list of GET requests with their reply. Remark that there's a GET request every minute. That too is in the beacon configuration: 60.000 ms of sleep (option 0x0003) with 0% variation (aka jitter, option 0x0005).

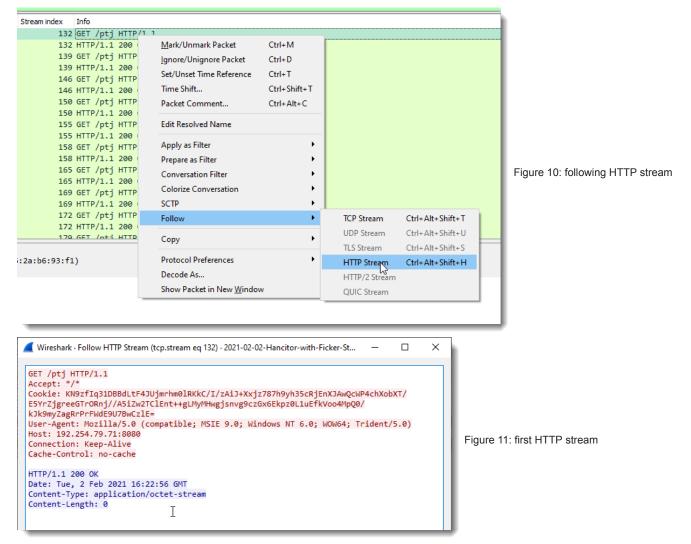
2021-02-02-Hancitor-with-Ficker-Stealer-and-Cobalt-Strike-and-NetSupport-RAT.pcap

 File
 Edit
 Vireless
 Icols
 Help

 Image: Image:

h	http and ip.addr == 192.254.79.71 and frame.num	oer > 6703						
о.	Time	Source	Destination	Protocol	Length	Stream index	Info	1
Þ	6759 2021-02-02 17:24:34,024763	10.2.2.101	192.254.79.71	HTTP	429	132	GET /ptj HTTP/1.1	
-	6765 2021-02-02 17:24:34,341254	192.254.79.71	10.2.2.101	HTTP	168	132	HTTP/1.1 200 OK	
	8056 2021-02-02 17:25:34,546440	10.2.2.101	192.254.79.71	HTTP	429	139	GET /ptj HTTP/1.1	
	8058 2021-02-02 17:25:34,833999	192.254.79.71	10.2.2.101	HTTP	168	139	HTTP/1.1 200 OK	
	8260 2021-02-02 17:26:35,038203	10.2.2.101	192.254.79.71	HTTP	429	146	GET /ptj HTTP/1.1	
	8262 2021-02-02 17:26:35,365467	192.254.79.71	10.2.2.101	HTTP	168	146	HTTP/1.1 200 OK	
	8707 2021-02-02 17:27:35,583846	10.2.2.101	192.254.79.71	HTTP	429	150	GET /ptj HTTP/1.1	
	8709 2021-02-02 17:27:35,799318	192.254.79.71	10.2.2.101	HTTP	168	150	HTTP/1.1 200 OK	Figure 9: H
	8776 2021-02-02 17:28:35,975093	10.2.2.101	192.254.79.71	HTTP	429	155	GET /ptj HTTP/1.1	r iguro o. ri
	8778 2021-02-02 17:28:36,222489	192.254.79.71	10.2.2.101	HTTP	168	155	HTTP/1.1 200 OK	
	8840 2021-02-02 17:29:36,452787	10.2.2.101	192.254.79.71	HTTP	429	158	GET /ptj HTTP/1.1	
	8842 2021-02-02 17:29:36,779923	192.254.79.71	10.2.2.101	HTTP	168	158	HTTP/1.1 200 OK	
	8954 2021-02-02 17:30:37,032111	10.2.2.101	192.254.79.71	HTTP	429	165	GET /ptj HTTP/1.1	
	8956 2021-02-02 17:30:37,339622	192.254.79.71	10.2.2.101	HTTP	168	165	HTTP/1.1 200 OK	
	9045 2021-02-02 17:31:37,503880	10.2.2.101	192.254.79.71	HTTP	429	169	GET /ptj HTTP/1.1	
	9047 2021-02-02 17:31:37,776451	192.254.79.71	10.2.2.101	HTTP	168	169	HTTP/1.1 200 OK	
	9095 2021-02-02 17:32:38,033184	10.2.2.101	192.254.79.71	HTTP	429	172	GET /ptj HTTP/1.1	
	9097 2021-02-02 17:32:38,346333	192.254.79.71	10.2.2.101	HTTP	168	172	HTTP/1.1 200 OK	
	0370 2021-02-02 17+33+38 622804	10 2 2 101	102 254 70 71	нттр	120	179	GET /nti HTTD/1 1	-
	Frame 6759: 429 bytes on wire (3432							1
E	Ethernet II, Src: HewlettP_1c:47:ae	(00:08:02:1c:47:ae), Dst: Netgear_	b6:93:f1	(20:e5	:2a:b6:93:f1	L)	

requests with encrypted Cobalt Strike traffic We will now follow the first HTTP stream:



This is a GET request for /ptj that receives a STATUS 200 reply with no data. This means that there are no commands from the team server for this beacon for now: the operator has not issued any commands at that point in the capture file.

Remark the Cookie header of the GET request. This looks like a BASE64 string: KN9zflq31DBBdLtF4JUjmrhm0IRKkC/I/zAiJ+Xxjz787h9yh35cRjEnXJAwQcWP4chXobXT/E5YrZjgreeGTrORnj//A5iZw2TClEnt++gLMyMHwgjsnvg

That value is encrypted metadata that the beacon sends as a BASE64 string to the team server. This metadata is RSA encrypted with the public key inside the beacon configuration (option 0x0007), and the team server can decrypt this metadata because it has the private key. Remember that some private keys have been "leaked", we discussed this in our <u>first blog post in this series</u>.

Our beacon analysis showed that this beacon uses a public key with a known private key. This means we can use tool <u>cs-decrypt-metadata.py</u> to decrypt the metadata (cookie) like this:

@NVISO_Labs	-	×
VISO_Labs C:\Demo>cs-decrypt-metadata.py KN9zfIq31DBBdLtF4JUjmrhm0lRKkC/I/zAiJ+Xxjz787h9yh35cRjEnXJAwQcWP4chXobXT/E lEnt++gLMyMHwgjsnvg9cz6x6Ekpz0L1uEfkVoo4MpQ0/kJk9myZagRrPrFWdE9U7BwCzlE= crypted metadata: KN9zfIq31DBBdLtF4JUjmrhm0lRKkC/I/zAiJ+Xxjz787h9yh35cRjEnXJAwQcWP4chXobXT/E5YrZjgreeGTrORnj//ASiZw x6Ekpz0L1uEfkVoo4MpQ0/kJk9myZagRrPrFWdE9U7BwCzlE=		
crypted: ader: 0000beef		
tasize: 0000005d w key: caeab4f452fe41182d504aa24966fbd0		
eskey: 3342f45e6e2f71f5975c998600b11471		
nackey: 7142dd70f4ec320badac8ca246a9488f arset: 04e4 ANSI Latin 1; Western European (Windows)		
nset_one 0155 Cell United States		
: 644d8e4 105175268		
1: 1c7c 7292 t: 0		
1: 10		
2: 0		
-3: 19042 -4: 0		
5: 1988364896		
6: 1988359504		
ernal IPv4: 10.6.9.111 Id: b'DESKTOP-021RU7A'		
Id b maxwell.carter'		
eld: b'svchost.exe'		
VISO_Labs C:\Demo>		

Figure 12: decrypting beacon metadata

We can see here the decrypted metadata. Very important to us, is the raw key: caeab4f452fe41182d504aa24966fbd0. We will use this key to decrypt traffic (the AES adn HMAC keys are derived from this raw key).

More metadata that we can find here is: the computername, the username, ...

We will now follow the HTTP stream with packets 9379 and 9383: this is the first command send by the operator (team server) to the beacon:



Figure 13: HTTP stream with encrypted

command

Here we can see that the reply contains 48 bytes of data (Content-length). That data is encrypted:

											- 20	21 0	2 01				ith-Fick					
0000000	47	45	54	20	2f	70	74	6a	20	48	54	54	50	2f	31	2e				TP/1.		
00000010	31	Ød	0a	41	63	63	65	70	74	3a	20	2a	2f	2a	Ød	0a	1A	ссер	t: '	*/*		
00000020		6f							4b	4e	39	7a	66	49	71	33	Cook	ie:	KN9:	zfIq3		
00000030	31	44	42	42	64	4c	74	46	34	4a	55	6a	6d	72	68	6d	1DBB	dLtF	4JU	jmrhm		
00000040	30	6c	52	4b	6b	43	2f	49	2f	7a	41	69	4a	2b	58	78	Ølrk	kC/I	/zAi	iJ+Xx		
00000050		7a								33							jz78	7h9y	h350	cRjEn		
0000060	58	4a	41	77	51	63	57	50	34	63	68	58	6f	62	58	54	XJAw	QcWP	4ch)	KobXT		
00000070		45							72	65	65	47	54	72	4f	52	/E5Y	rZjg	ree(GTrOR		
0800000		6a							77	32	54	43	6c	45	6e	74	nj//	A5iZ	w2T(ClEnt		
00000090	2b	2b	67	4c	4d	79	4d	48		67							++gL	МуМН	wgjs	snvg9		
000000A0	63	7a	47	78	36	45	6b	70	7a	30	4c	31	75	45	66	6b	czGx	6Ekp	z0L1	luEfk		
000000B0		6f							2f	6b	4a	6b	39	6d	79	5a	Voo4	MpQ0	/kJł	k9myZ		
000000C0		67								45							agRr	PrFW	dE9l	J7BwC		
000000D0								73									zlE=	Us	er-A	Agent		
00000E0	3a	20	4d	6f	7a	69	6c	6c	61	2f	35	2e	30	20	28	63	: Mo	zill	a/5.	.0 (c		
00000F0	6f	6d	70	61	74	69	62	6c	65	Зb	20	4d	53	49	45	20	ompa	tibl	e; /	ISIE		
00000100	39	2e	30	Зb	20	57	69	6e	64	6f	77	73	20	4e	54	20	9.0;	Win	dows	s NT		
00000110	36	2e	30	Зb	20	57	4f	57	36	34	Зb	20	54	72	69	64	6.0;	WOW	64;	Trid		
00000120	65	6e	74	2f	35	2e	30	29	0d	0a	48	6f	73	74	3a	20	ent/	5.0)	Ho	ost:		
00000130	31	39	32	2e	32	35	34	2e	37	39	2e	37	31	3a	38	30	192.	254.	79.7	71:80		
00000140	38	30	Ød	0a	43	6f	6e	6e	65	63	74	69	6f	6e	3a	20	80	Conn	ecti	ion:		
00000150								69									Кеер	-Ali	ve.	.Cach		
00000160	65	2d	43	6f	6e	74	72	6f	6c	Зa	20	6e	6f	2d	63	61	e-Co	ntro	1: 1	no-ca		
00000170	63	68	65	Ød	0a	Ød	0a										che.					
00000	000	48	54	54	50	21	31	. 2e	31	20	32	2 30	30	20	4f	4b	Ød	нттр,	/1.1	200	OK.	
00000	010	Øa	44	61	. 74	65	5 3a	20	54	75	65	5 2 c	20	32	20	46	65	.Date	e: T	ue,	2 Fe	
00000	020	62	20	32	30	32	2 31	. 20	31	36	5 3a	33	32	3a	30	31	20	b 202	21 1	6:32	:01	
00000	030	47	40	54	0d	0a	43	6f	6e	74	65	5 6e	74	2d	54	79	70	GMT.	.Con	tent	-Тур	
00000	040	65	5 3a	20	61	. 70	70	6c	69	63	61	74	69	6f	6e	2f	6f	e: ap	ppli	cati	on/o	
00000	050	63	3 74	65	74	20	73	74	72	65	61	6d	00	0a	43	6f	6e	ctet	-str	eam.	.Con	
00000	060	74	65	6 6e	74	20	40	65	6e	67	74	68	3a	20	34	38	Ød	tent	-Len	gth:	48.	
00000	070	Øa	00	1 Øa																		
00000	073	f9) c7	' cc	ef	ce	47	' a0	62	42	2 26	5 7b	17	e4	2f	c7	30		.G.b	B&{.	./.0	
00000	083	a4	e e	85	co	af	80	: Øb	42	bs	0	: d7	dg	a1	. 5e	6b	40		В		.^k@	
00000	093	19	0 ø	d6	f9	34	l da	67	85	53	8 01	b2	50) b3	ec	13	ee	4	4.g.	SP		

Figure 14: hexadecimal view of HTTP stream

with encrypted command

Encrypted data like this, can be decrypted with tool <u>cs-parse-http-traffic.py</u>. Since the data is encrypted, we need to provide the raw key (option -r caeab4f452fe41182d504aa24966fbd0) and as the packet capture contains other traffic than pure Cobalt Strike C2 traffic, it is best to provide a display filter (option -Y http and ip.addr == 192.254.79.71 and frame.number > 6703) so that the tool can ignore all HTTP traffic that is not C2 traffic.

This produces the following output:

@NVISO_Labs		- 0	×
	ic.py -r caeab4f452fe41182d504aa24966fbd0 -Y "http and ip.addr == 192.254.79.71 and fr ler-and-Cobalt-Strike-and-NetSupport-RAT.pcap more	ame.number >	670
Packet number: 9707 HTTP response Timestamp: 1612283536 20210202-163216 Data size: 19 Command: 53 UNKNOWN Arguments length: 11 b'\x00\x00\x00\x00\x00\x00\x00\x00\x03* MD5: c0747fd03245897c597c4ba783e0ebb7			
Packet number: 9723 HTTP request http://192.254.79.71:8080/submit.php?id Counter: 2 Callback: 22 TODO b`\x00\x00\x00\xba`	=242569267		
C:\Users\maxwell.carter\Documents* D 0 01/28/2021 02:15:53 D 0 01/28/2021 02:15:53 F 1330688 07/22/2018 05:45:52 F 691049 07/22/2018 02:20:50 F 610179 07/22/2018 02:20:50 F 206336 12/02/2018 02:21:02 F 206336 12/02/2016 23:11:02 F 14688 12/02/2016 23:11:02 F 14688 12/02/2018 05:53:02 F 14688 12/02/2018 05:53:02 F 224879 07/22/2018 05:53:02 F 521454 07/22/2018 05:39:26 F 413895 07/22/2018 02:23:28 F 521454 07/22/2018 02:23:28 F 413895 07/22/2018 02:23:54 F 199168 07/22/2018 05:52:14 F 202085 07/22/2018 05:53:45	13-Using_Powerpoint.ppt 160197-airport-template-16x9.pptx 160228-photo-template-16x9.pptx 160283-hook-template-16x9.pptx 1625.ppt 1625_example.jpg 19-adp.doc 2016_election_final_report_06-14-18_0.pdf 2018-calendar.xlsx 301FRN.pdf 4Q-report.pptx activity-costs-tracker1.xlsx AID1420-17.doc amp-a0034857.pdf antFableEng.pps APAFormat.doc		

Figure 15: decrypted commands and results

Now we can see that the encrypted data in packet 9383 is a sleep command, with a sleeptime of 100 ms and a jitter factor of 90%. This means that the operator instructed the beacon to beacon interactive.

Decrypted packet 9707 contains an unknown command (id 53), but when we look at packet 9723, we see a directory listing output: this is the output result of the unknown command 53 being send back to the team server (notice the POST url /submit.php). Thus it's safe to assume that command 53 is a directory listing command.

There are many commands and results in this capture file that tool cs-parse-http-traffic.py can decrypt, too much to show here. But we invite you to reproduce the commands in this blog post, and review the output of the tool.

The last command in the capture file is a process listing command:

@NVISO_Labs			_	×
Packet number: 8 HTTP response Timestamp: 16122 Data size: 12 Command: 32 LIS Arguments lengt b'\x00\x00\x00\x00 MD5: def5c8e080	286770 2 [_PROCES th: 4 \xce'			^
Packet number: & HTTP request http://192.254.7 Counter: 28 Callback: 22 TOD b`\x00\x00\x00\x00\x00	79.71:80 00	80/submit.php?id=242569267		
[System Process]	1	0 0		
System 0	4			
Registry	4	108		
smss.exe	4	384		
csrss.exe	488	500		
wininit.exe	488	580		
csrss.exe	572	588		
winlogon.exe	572	680		
services.exe	580	728		
lsass.exe	580	748		
svchost.exe	728	860		
fontdrvhost.exe		896		
fontdrvhost.exe		964		
svchost.exe	728	992		
svchost.exe	728	412		
dwm.exe 680	984			
svchost.exe	728	1096		
svchost.exe	728	1108		
svchost.exe	728	1176		
svchost.exe	728	1184		
svchost.exe	728	1192		
svchost.exe	728	1284		
svchost.exe	728	1292		
svchost.exe	728	1300		
svchost.exe	728	1464		
svchost.exe	728	1476		
svchost.exe	728	1504		
svchost.exe	728	1556		
svchost.exe	728	1588		~

Figure 16: decrypted process listing command and result **Conclusion**

Although the packet capture file we decrypted here was produced more than half a year ago by Brad Duncan by running a malicious Cobalt Strike beacon inside a sandbox, we can decrypt it today because the operators used a rogue Cobalt Strike package including a private key, that we recovered from VirusTotal.

Without this private key, we would not be able to decrypt the traffic.

The private key is not the only way to decrypt the traffic: if the AES key can be extracted from process memory, we can also decrypt traffic. We will cover this in an upcoming blog post.

About the authors

Didier Stevens is a malware expert working for NVISO. Didier is a SANS Internet Storm Center senior handler and Microsoft MVP, and has developed numerous popular tools to assist with malware analysis. You can find Didier on <u>Twitter</u> and <u>LinkedIn</u>.

You can follow NVISO Labs on Twitter to stay up to date on all our future research and publications.