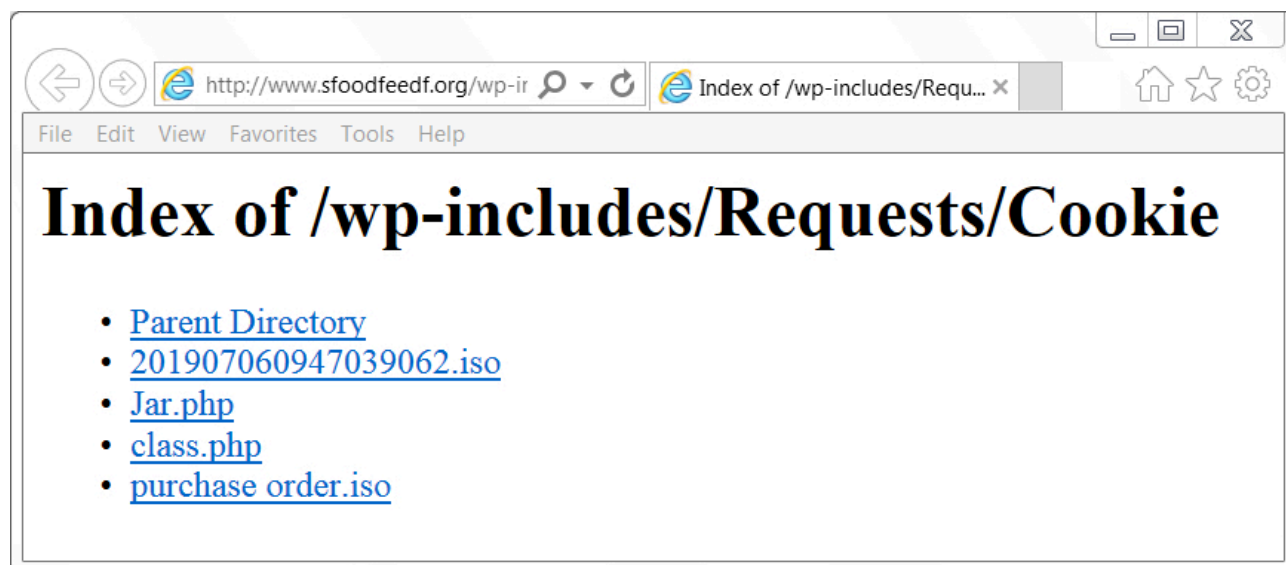


Recent AZORult activity - SANS Internet Storm Center

By SANS Internet Storm Center

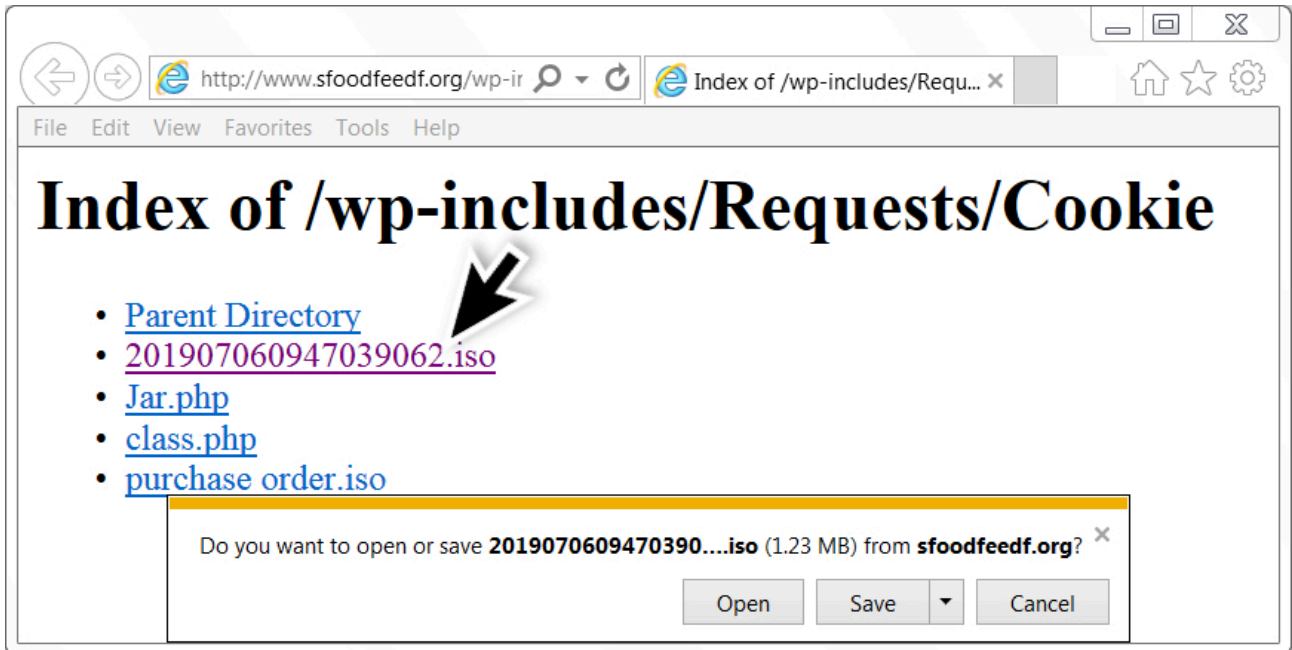
Archived: 2026-04-05 21:35:36 UTC

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-includes/Requests/Cookie/)

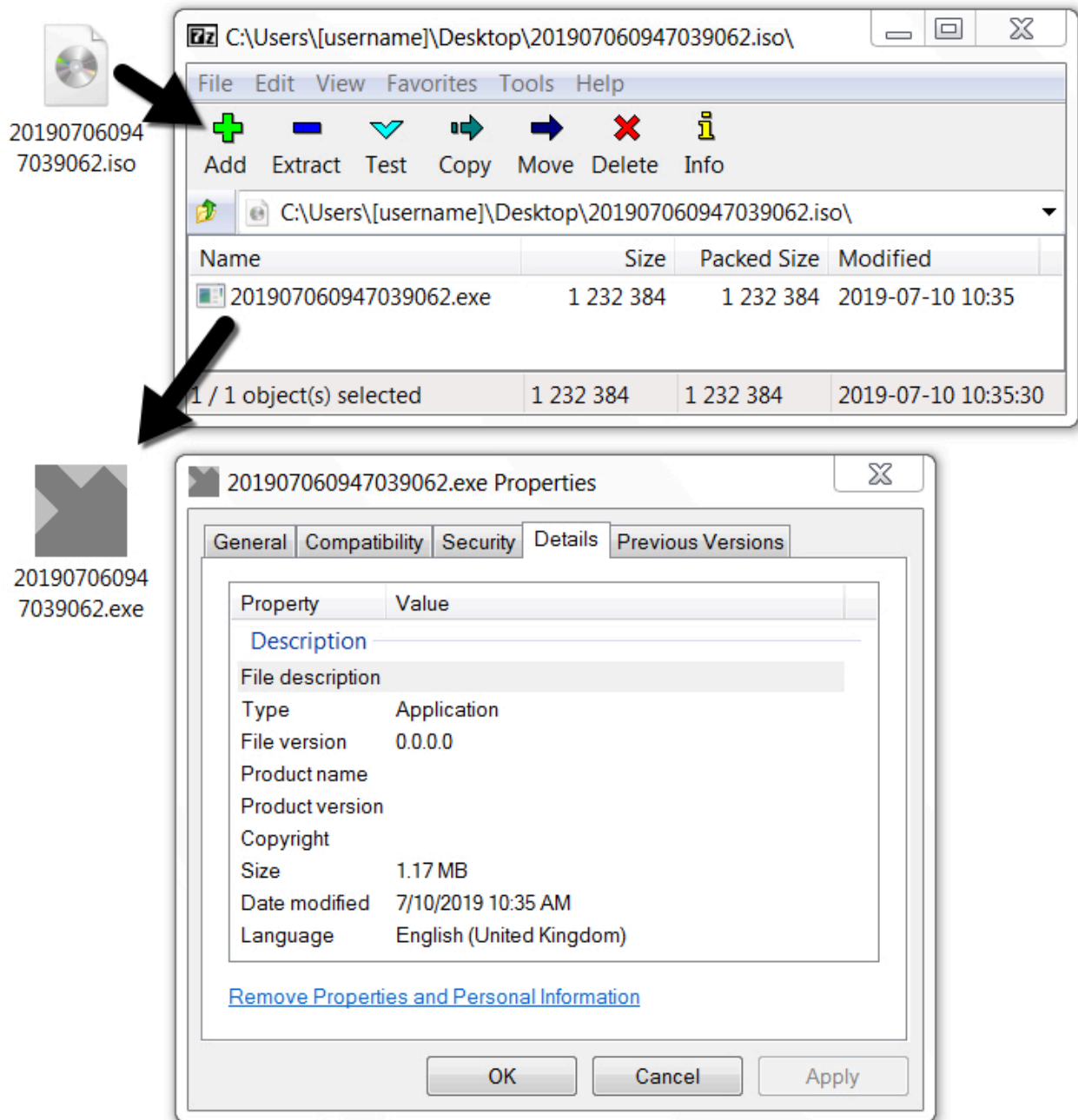


Shown above: The open directory at sfoodfeedf[.]org.

@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.



Shown above: Getting the other ISO file.



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

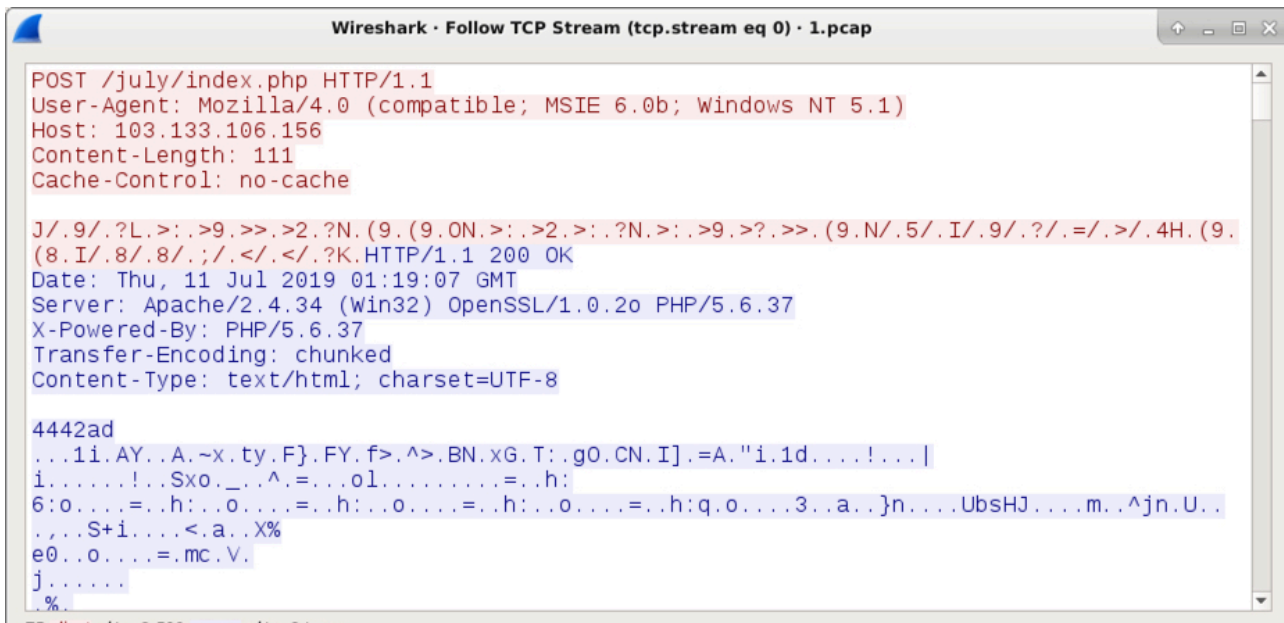
In previous AZORult infections in my lab, the malware usually deleted itself after an initial exfiltration of data. This one repeatedly did callback traffic, and there was a .vbs file made persistent on my infected Windows host during the infection. This is apparently a more recent variant of AZORult 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.

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

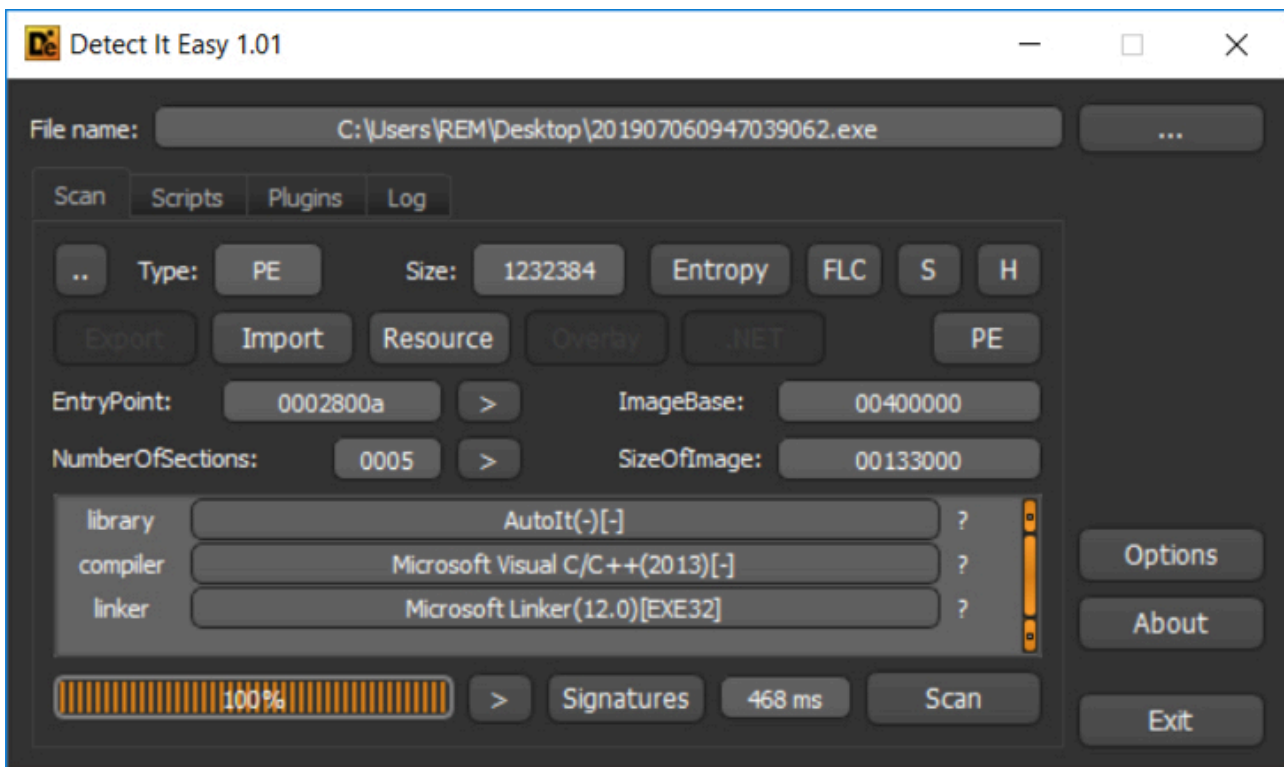
Shown above: Traffic from the infection filtered in Wireshark.

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

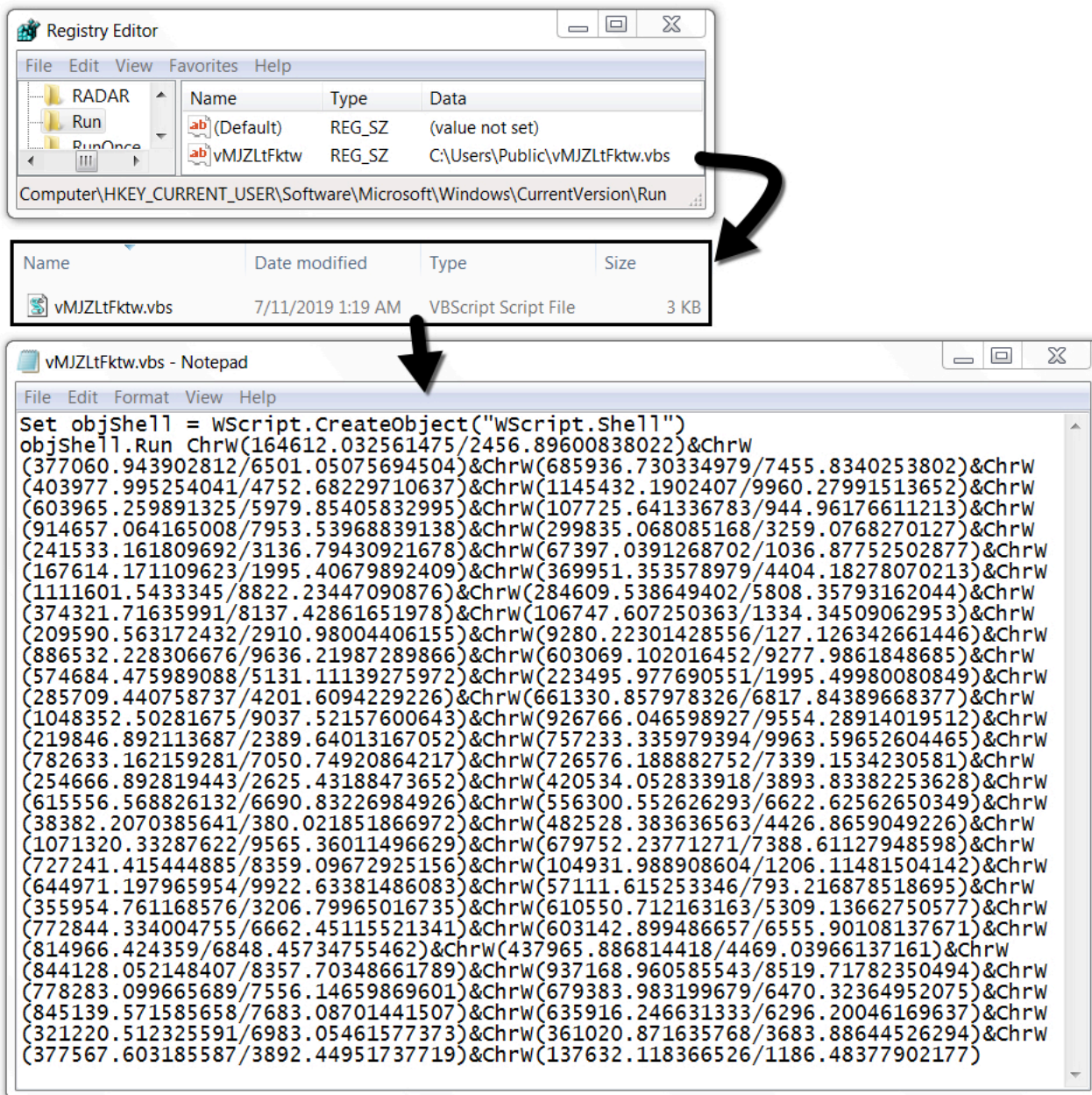
Shown above: TCP conversations from my infected Windows host.



Shown above: An example of the AZORult callback traffic.



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



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

Source: <https://isc.sans.edu/diary/25120>