Industroyer2 IEC-104 Analysis

netresec.com/

April 25, 2022

Erik Hjelmvik

,

Monday, 25 April 2022 10:35:00 (UTC/GMT)

The Industroyer2 malware was hardwired to attack a specific set of electric utility substations in Ukraine. It seems to have been custom built to open circuit breakers, which would effectively cut the power from the substation.

After connecting to an RTU in a substation the malware would immediately start changing the outputs at specific addresses without first having to enumerate which IOAs that were available on the targeted device. This custom-built malware seems to know what IOAs to use at each station, as well as what type of output each specific IOA controls.

UPDATE 2022-04-26

Upon popular demand we've decided to release three PCAP files with IEC-104 traffic from our own sandbox execution of the Industroyer2 malware. Please feel free to use these capture files to verify our findings using any tool of your choice. The capture files can be downloaded from here:
https://www.netresec.com/files/Industroyer2-Netresec.zip

UPDATE 2022-04-29

A PNG image in the original CERT-UA security alert #4435 turned out to actually include the IOAs targeted by the non-public 108_100.exe Industroyer2 version. The IOAs disclosed in CERT-UAs alert have now been included in this blog post as well.

Backstory

I was looking at a public sandbox execution of a presumed Industroyer2 malware sample two weeks ago. At first glance the malware sample, which was named “40_115.exe”, didn't do much. It just printed the text below to the console and then terminated the process.

19:46:06:0106> T281 00006800
19:46:06:0247> RNM 0015
19:46:06:0294> 10.82.40.105: 2404: 3
19:46:06:0294> T65 00006800
19:46:06:0341> 10.82.40.105 M68B0 SGCNT 44
19:46:06:0497> RNM 0015
19:46:06:0544> T113 00006800
19:46:06:0544> 192.168.122.2: 2404: 2
19:46:06:0544> 192.168.122.2 M68B0 SGCNT 8
19:46:06:0591> RNM 0015
19:46:06:0653> 192.168.121.2: 2404: 1
19:46:06:0700> 192.168.121.2 M68B0 SGCNT 16
19:46:21:0747> 192.168.122.2 M6812
19:46:21:0747> 10.82.40.105 M6812
19:46:21:0794> 192.168.121.2 M6812

I later noticed that it also sent TCP SYN packets to three different RFC1918 addresses, but never received a response.

Source	Destination	Protocol	Info
192.168.2.3	10.82.40.105	TCP	49719 → 2404 [SYN]
192.168.2.3	192.168.122.2	TCP	49720 → 2404 [SYN]
192.168.2.3	192.168.121.2	TCP	49721 → 2404 [SYN]

Image: Wireshark showing Industroyer2 trying to reach TCP port 2404

TCP port 2404 is used by the SCADA protocol IEC 60870-5-104, also known as IEC-104, which is primarily used to monitor and control electricity transmission and distribution systems. IEC-104 is also the only Industrial Control System (ICS) protocol implemented in Industroyer2 according to ESET. The previous version of Industroyer, which was used to cut the power in Ukraine in 2016, additionally supported the IEC 61850 and OPC DA protocols according to the CRASHOVERRIDE report from Dragos.

Industroyer2's IEC-104 client didn't receive any SYN+ACK response in the sandbox execution I was looking at, so I couldn't tell what it was trying to do. I therefore decided to set up my own sandbox with a built-in IEC-104 server (also known as a slave, RTU or IED). My sandbox execution confirmed that Industroyer2 was indeed trying to communicate with these three IP addresses using IEC-104. I also noticed that it was very specific about which outputs (or IOAs) it wanted to access on those servers in order to turn these outputs either ON or OFF.

Station Address 1 at 192.168.121.2

The Industroyer2 malware spawned three separate threads when started, one thread for each IEC-104 server to contact. The malware would communicate with all three servers in parallel if all of them were available. However, in order to simplify my analysis I decided to only respond to one of the IPs at a time, starting with IP address 192.168.121.2.

The thread that connected to IP address 192.168.121.2 toggled all outputs between 1250 and 1265 to OFF at Station Address 1 (also known as “ASDU address” or “common address”). Judging from the command type used (ID 46 with short pulse duration) these outputs likely control circuit breakers, which are used to disconnect the power from an electric utility substation.

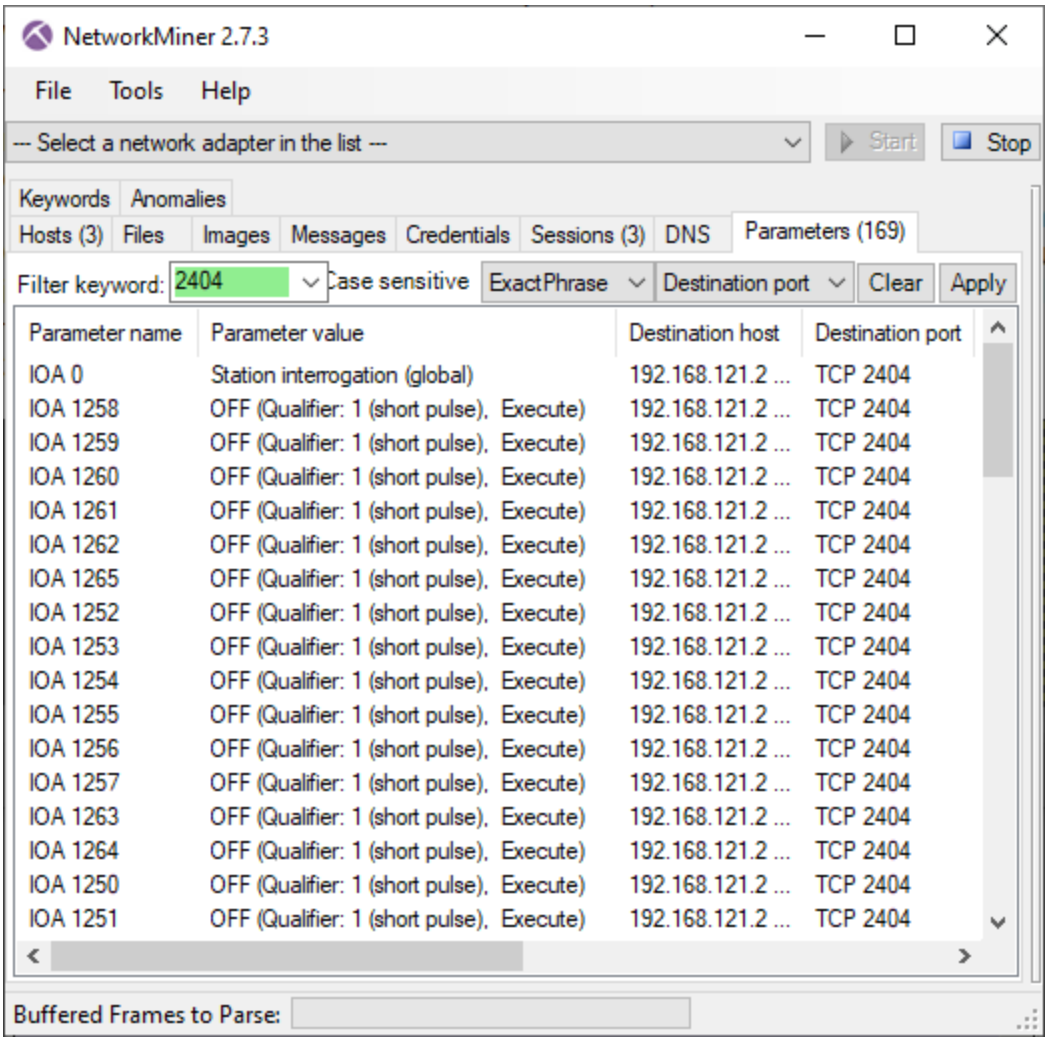

Image: PCAP file with IEC-104 traffic to 192.168.121.2 in NetworkMiner

Station Address 2 at 192.168.122.2

On 192.168.122.2 the malware targeted station address 2, where it toggled outputs between 1101 and 1108 to OFF.

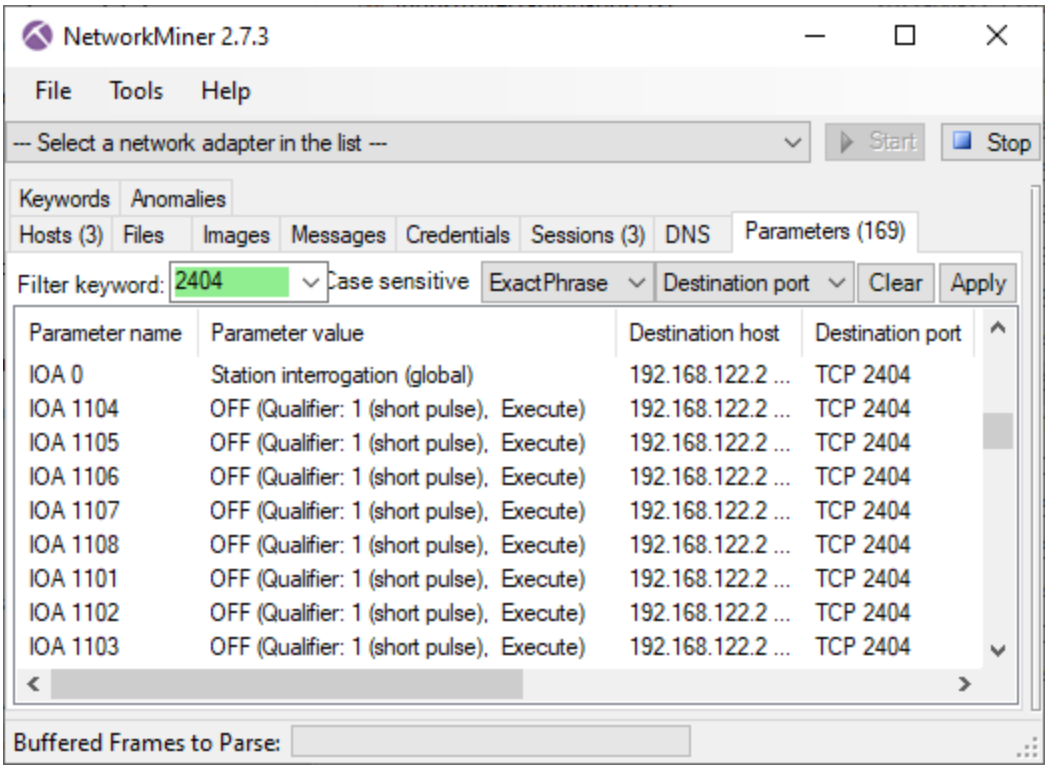

Image: PCAP file with IEC-104 traffic to 192.168.122.2 in NetworkMiner

Station Address 3 at 10.82.40.105

The malware toggled a great deal of outputs on 10.82.40.105, which had station address 3. But in contrast to the other stations, many of these outputs were toggled to the “ON” state rather than “OFF”.

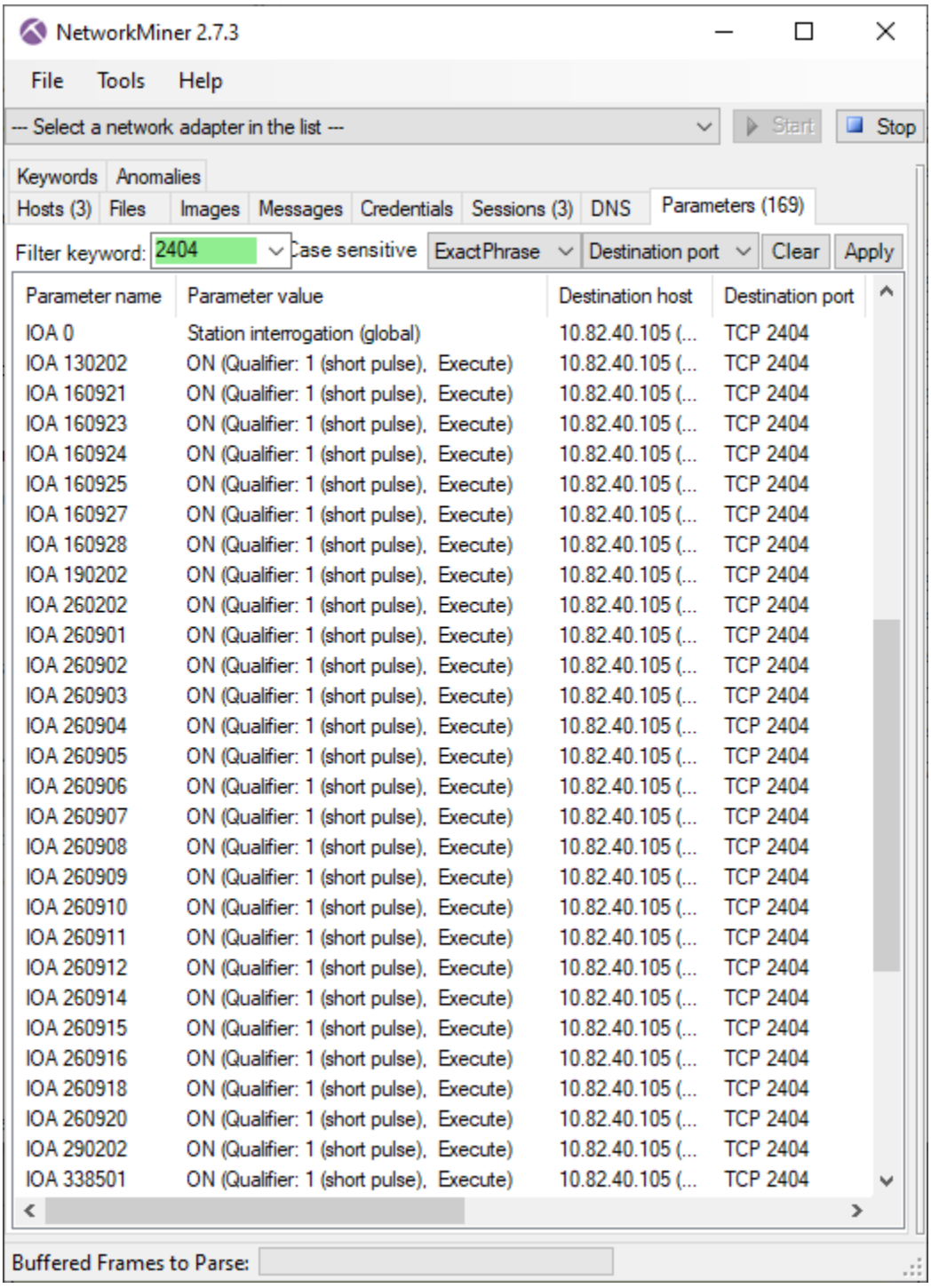

Image: PCAP file with IEC-104 traffic to 10.82.40.105 in NetworkMiner

Yet, after setting those outputs to “ON” it proceeded with setting outputs to “OFF” for several other IOAs on station address 3.

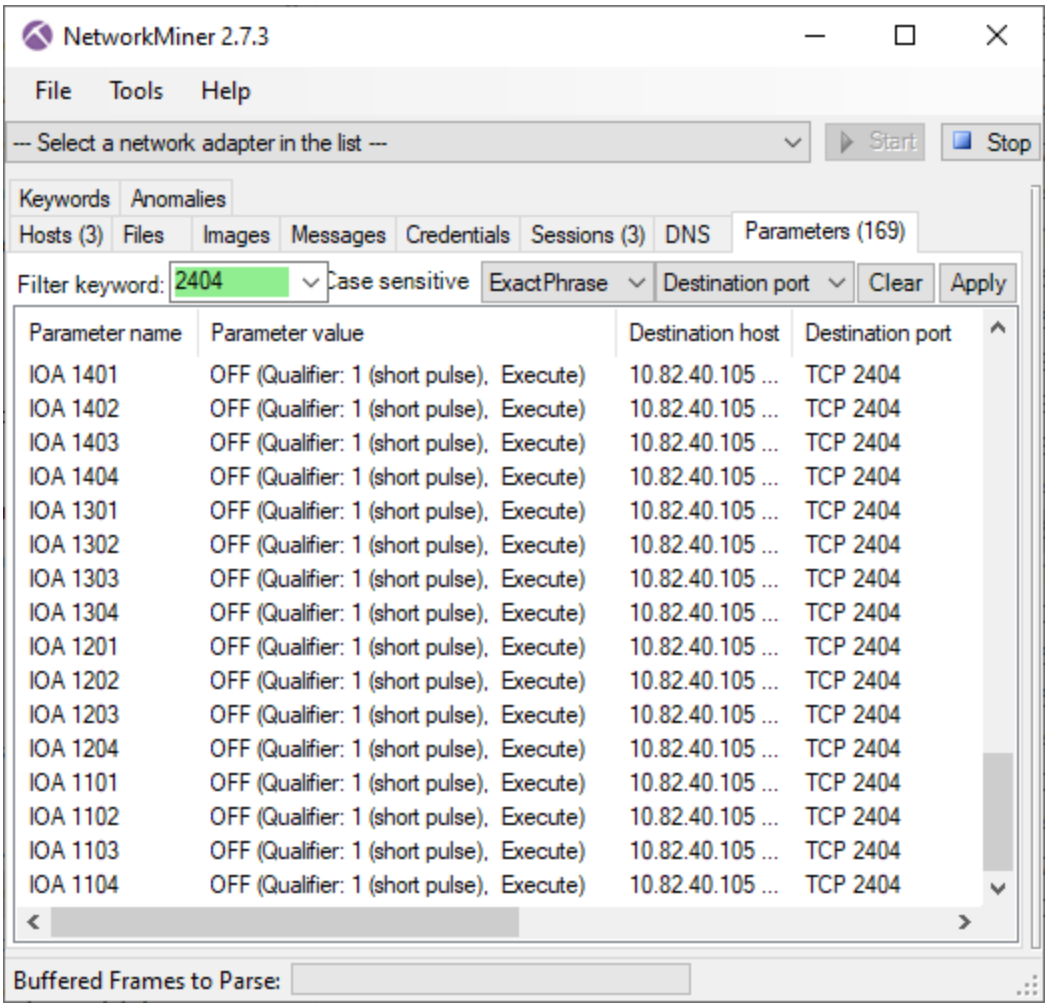

Image: PCAP file with IEC-104 traffic to 10.82.40.105 in NetworkMiner

In each thread Industroyer2 paused for approximately 3 seconds between each accessed IOA. This delay seems to have been hard coded since the malware didn't seem to care whether or not the IEC-104 server responded with an OK message, such as ACT or ACTTERM, or an error message, like “unknown common address of ASDU”. Each thread would simply proceed with setting an IOA every 3 seconds no matter what the server responded.

The specific order in which the IOAs were accessed was also very deterministic, the exact same sequence of IOAs was used every time. I verified this behavior by running the malware multiple times as well as by comparing my results to an execution of the same sample on a different sandbox (thanks for the PCAP Joe and Dan).

What Did the Attackers Know?

The fact that the malware toggled these specific outputs, rather than just randomly turning outputs ON or OFF, indicates that the threat actors had technical knowledge about the specific substation(s) they were attacking. Not only did the attackers know the IP addresses, station addresses and IOAs of each targeted output. They also knew what ASDU Type ID to

use for each respective output. For IOA 1101 to 1404 the Type ID 46 was used (also known as "double command" or C_DC_NA_1) while for IOAs from 130202 and above it used Type ID 45 (also known as “single command” or C_SC_NA_1).

As you can see in the previous screenshots, NetworkMiner nicely parses and presents the IEC-104 commands issued by Industroyer2. But I noticed that the malware also printed all sent and received commands to the console when executed. For example, the following output was printed to the console by the Industroyer2 thread communicating with station address 2 on 192.168.122.2:

```
11:51:56:0163> T65 00006800
11:51:56:0201> RNM 0003
11:51:56:0241> 192.168.122.2: 2404: 2
11:51:56:0267> 192.168.122.2 M68B0 SGCNT 8
11:51:56:0297> 192.168.122.2 M6813
```

The string “192.168.122.2: 2404: 2” above reveals that “2404” is the target port and “2” is the station address. The “SGCNT 8” string additionally tells us that there were 8 outputs to be toggled on that station. The other two stations had SGCNT 16 and 44.

The malware also printed very detailed information about each sent and received IEC-104 command, such as in the example below where the output at IOA 1104 was successfully turned off at station address 2 (here referred to as “ASDU:2”).

```
MSTR ->> SLV 192.168.122.2:2404
  x68 x0E x02 x00 x08 x00 x2E x01 x06 x00 x02 x00 x50 x04 x00 x05

  I |Length:16 bytes | Sent=x1 | Received=x4
  ASDU:2 | OA:0 | IOA:1104 |
  Cause: (x6) | Telegram type: (x2E)

MSTR <<- SLV 192.168.122.2:2404
  x68 x0E x08 x00 x04 x00 x2E x01 x47 x00 x02 x00 x50 x04 x00 x05

  I |Length:16 bytes | Sent=x4 | Received=x2
  ASDU:2 | OA:0 | IOA:1104 |
  Cause: (x47) | Telegram type: (x2E)
```

Note that the Type ID values were also logged to the console by Industroyer2, but it used the term “Telegram type” instead of “Type ID”.

Static Analysis

The following three Unicode strings can be found in the 40_115.exe binary:

> 10.82.40.105 2404 3 0 1 1 PService_PPD.exe 1 "D:\OIK\DevCounter" 0 1 0 0 1 0 0 44 130202 1 0 1 1 1 160921 1 0 1 1 2 160923 1 0 1 1 3 160924 1 0 1 1 4 160925 1 0 1 1 5 160927 1 0 1 1 6 160928 1 0 1 1 7 190202 1 0 1 1 8 260202 1 0 1 1 9 260901 1 0 1 1 10 260902 1 0 1 1 11 260903 1 0 1 1 12 260904 1 0 1 1 13 260905 1 0 1 1 14 260906 1 0 1 1 15 260907 1 0 1 1 16 260908 1 0 1 1 17 260909 1 0 1 1 18 260910 1 0 1 1 19 260911 1 0 1 1 20 260912 1 0 1 1 21 260914 1 0 1 1 22 260915 1 0 1 1 23 260916 1 0 1 1 24 260918 1 0 1 1 25 260920 1 0 1 1 26 290202 1 0 1 1 27 338501 1 0 1 1 28 1401 0 0 0 1 29 1402 0 0 0 1 30 1403 0 0 0 1 31 1404 0 0 0 1 32 1301 0 0 0 1 33 1302 0 0 0 1 34 1303 0 0 0 1 35 1304 0 0 0 1 36 1201 0 0 0 1 37 1202 0 0 0 1 38 1203 0 0 0 1 39 1204 0 0 0 1 40 1101 0 0 0 1 41 1102 0 0 0 1 42 1103 0 0 0 1 43 1104 0 0 0 1 44

> 192.168.122.2 2404 2 0 1 1 PService_PPD.exe 1 "D:\OIK\DevCounter" 0 1 0 0 1 0 0 8 1104 0 0 0 1 1 1105 0 0 0 1 2 1106 0 0 0 1 3 1107 0 0 0 1 4 1108 0 0 0 1 5 1101 0 0 0 1 6 1102 0 0 0 1 7 1103 0 0 0 1 8

> 192.168.121.2 2404 1 0 1 1 PService_PPD.exe 1 "D:\OIK\DevCounter" 0 1 0 0 1 0 0 16 1258 0 0 0 1 1 1259 0 0 0 1 2 1260 0 0 0 1 3 1261 0 0 0 1 4 1262 0 0 0 1 5 1265 0 0 0 1 6 1252 0 0 0 1 7 1253 0 0 0 1 8 1254 0 0 0 1 9 1255 0 0 0 1 10 1256 0 0 0 1 11 1257 0 0 0 1 12 1263 0 0 0 1 13 1264 0 0 0 1 14 1250 0 0 0 1 15 1251 0 0 0 1 16

After having analyzed the IEC-104 traffic from the binary it's obvious that this is the IEC-104 configuration that has been hard-coded into the binary. For example, the substring “10.82.40.105 2404 3” in the first Unicode string refers to the IP, port and station number of the first target.

The “16 1258 [...]” section in the third Unicode string above tells us that there are 16 outputs configured for station address 1, where the first one to be set is at IOA 1258. Thus, we can easily verify that all accessed IOAs on all three stations were hard-coded into the binary.

Additional Substations Targeted

The malware sample I've analyzed has the following properties:

- Filename: 40_115.exe
- MD5: 7c05da2e4612fca213430b6c93e76b06
- SHA1: fdeb96bc3d4ab32ef826e7e53f4fe1c72e580379
- SHA256: d69665f56ddef7ad4e71971f06432e59f1510a7194386e5f0e8926aea7b88e00
- Compiled: 2022-03-23 10:07:29 UTC

But there is an additional Industroyer2 sample called “108_100.exe” (MD5 3229e8c4150b5e43f836643ec9428865), which has been mentioned by ESET as well as CERT-UA. I haven't been able to access that binary though, so I don't yet know which IP addresses it was designed to target. However, a few screenshots [1] [2] [3] published by ESET reveal that the 108_100.exe malware sample was hard coded to access 8 different

station addresses, 5 of which were on the 10.0.0.0/8 network and 3 on the 192.168.0.0/16 net. An image in CERT-UA's alert #4435 from April 12 reveals the targeted IOAs for these 8 stations.

Targets hard-coded in 108_100.exe ordered by station address:

- SA#1, 192.x.x.x, 12 IOAs (1101-1104, 1201-1204, 1301-1304)
- SA#2, 10.x.x.x, 12 IOAs (1101-1104, 1201-1204, 1301-1304)
- SA#3, 192.x.x.x, 18 IOAs (1103-1104, 1201-1204, 1301-?, 38601-38607)
- SA#4, 10.x.x.x, 34 IOAs (16501, 16603, 26502, 38507-38513, 38519-38524 and more...)
- SA#5, 192.x.x.x, 10 IOAs (1101-1103, 1201-1204, 1301-1303)
- SA#6, 10.x.x.x, 8 IOAs (1101-1104, 1201-1204)
- SA#7, 10.x.x.x, 8 IOAs (1101-1104, 1201-1204)
- SA#8, 10.x.x.x, 8 IOAs (1101-1104, 1201-1204)

We can compare those station addresses, IP addresses and IOAs to the ones targeted by the 40_115.exe sample, which was analyzed in this blog post.

- SA#1, 192.168.121.2, 16 IOAs (1250-1265)
- SA#2, 192.168.122.2, 8 IOAs (1101-1108)
- SA#3, 10.82.40.105, 44 IOAs (1101-1104, 1201-1204, 1301-1304, 1401-1404, 130202, 160921-160928, 190202, 260202, 260901-260920, 290202, 338501)

There doesn't seem to be any overlap across the two sets (except for possibly station address 1 which is on the 192.x.x.x network in both configs but has different IOAs). This indicates that the 108_100.exe Industroyer2 version was hard coded to attack a different set of targets than the 40_115.exe sample that I've analyzed.

More ICS blog posts from Netresec

If you'd like to find our earlier work in the field of ICS/SCADA security, then check out these (slightly older but still very relevant) blog posts:

- 2011: Monitor those Control System Networks!
- 2012: SCADA Network Forensics with IEC-104
- 2014: Full Disclosure of Havex Trojans
- 2014: Observing the Havex RAT
- 2015: From 4SICS with ICS PCAP Files

Posted by Erik Hjelmvik on Monday, 25 April 2022 10:35:00 (UTC/GMT)

Tags: #IEC-104 #60870-5-104 #ICS #ICS #SCADA #PCAP