How ntopng monitors IEC 60870-5-104 traffic – ntop

Archived: 2026-04-05 18:33:19 UTC

Busy times for OT analysts.

Last month the number of known OT (operational technology) malware increased from five to seven. First malware discovered is Industroyer2 which was caught in the Ukraine. As nowadays popular, security companies name the malware they discover. That is why for the second malware two names were assigned, Incontroller or Pipedream. This malware was discovered before it was deployed.

Industroyer2 [1] is an evolution of Industroyer1, first seen in 2014. Both variants are targeting the electrical energy sector, specifically in Ukraine. As the malware is using commands out of the industrial protocol IEC-60870-5-104, traffic looks like legit communication as during normal operation.

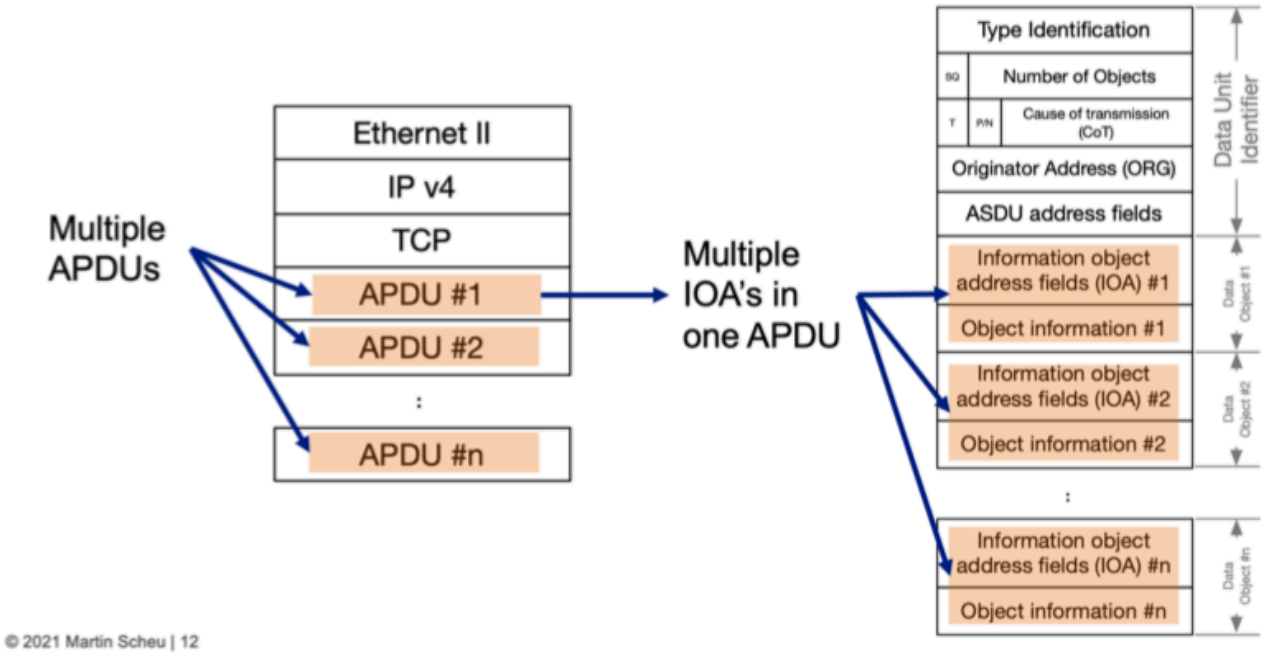
Incontroller [2] is a new set of malware components, targeting the LNG sector in the US. Similar to Industroyer2, pipedream is using popular industrial protocols like OPC-UA and ModbusTCP. Further more it uses build in functionality from engineering tools made to interact with OT devices like PLCs (process logic controller).

Both malware show clearly that the criminals behind have evolved and do understand OT protocols and are able to use build in functionality of legit software engineering tools like CODESYS.

What has not changed over the last years, is that SCADA systems still control like “fire and forget”. A command is sent from the control system server to the client in the field. The client translates the event into a physical action, like open or close a circuit breaker. The command source is not verified by the client. Translated back to the network traffic, it means that one packet containing a command is enough to disrupt the complete industrial process or power distribution.

Industroyer2 uses IEC-104, the short version for IEC 60870-5-104. IEC-104 is widely used in European energy sector and as well in utilities sector like water or waste water treatment.
A characteristic of many industrial protocol is, that even though the protocol is standardised, its implementation can vary between manufacturers or even system integrators. Meaning IEC-104 implemented by Hitachy Energy differs from how it is implemented by Siemens. Operators are familiar with it, but for network security monitoring it can be a challenge.

Further difficulty for monitoring is, that one packet transporting the IEC-104 payload can have multiple IEC-104 data messages, called APDU’s. Therefore traditional signature based detection on the TCP payload does not work. The payload needs to be parsed in order to understand what type of command each APDU contains:



Since the discovery of Industroyer2 in early of April 2022 until now, several reports analysing the malware were published. They contain information how the malware is working, captured network data and most of them contain recommendation how to deal with such type of malware. Having a closer look at the recommendations, or nowadays called actionable items, they are high level items. Example:

- "Use anomaly detection tools to detect any irregular activity occurring in OT environments"
- "Employing network segmentation to separate sensitive applications (e.g. PCN) from other network parts via firewalls"
- "Monitor East-West ICS networks with ICS protocol aware technologies"

Not much actionable in my point of view or a whole set of commercial products need to be in place. Products which for most SMEs are not suitable to operate. I am therefore looking for ways how to detect the malware with minimal effort.

Let's have a look at the environment. SCADA networks are highly deterministic. Means who is talking to whom and how, i.e. command and control patterns, is repeatable. For IEC-104 it means the same type and sequence of IEC-104 commands can be found in normal operation in time periods over a day or over weekdays and weekend days.

Example 1, time period 2 working days and one night, 36 hours:

TypeID	**Type**	**Description**	**Number of Occurrences**
13	M_ME_NC_1	Measured value, short floating point number	1'184'834

30	M_SP_TB_1	Single-point information witd time tag CP56Time2a	2
103	C_CS_NA_1	Clock synchronisation command	1

The only command sent was for time synchronisation of the client.

Comparing operations data with the available malware data, the different behaviour of the malware becomes visible:

iec60870_104 and tcp.dstport == 2404

No.	Time	Protocol	Length	Info
4	09:57:56.388263	IEC 60870-5-104	50	<- U (TESTFR act)
8	09:57:57.777530	IEC 60870-5-104	50	<- U (STARTDT act)
12	09:57:59.168074	IEC 60870-5 ASDU	60	<- I (0,0) ASDU=3 C_IC_NA_1 Act IOA=0
16	09:58:00.966451	IEC 60870-5-104	50	<- S (1)
18	09:58:02.152117	IEC 60870-5 ASDU	60	<- I (1,1) ASDU=3 C_SC_NA_1 Act IOA=130202
22	09:58:03.949065	IEC 60870-5 ASDU	60	<- I (2,1) ASDU=3 C_SC_NA_1 Act IOA=160921
26	09:58:05.777491	IEC 60870-5 ASDU	60	<- I (3,1) ASDU=3 C_SC_NA_1 Act IOA=160923
30	09:58:07.574706	IEC 60870-5 ASDU	60	<- I (4,1) ASDU=3 C_SC_NA_1 Act IOA=160924
34	09:58:09.371291	IEC 60870-5 ASDU	60	<- I (5,1) ASDU=3 C_SC_NA_1 Act IOA=160925
38	09:58:11.027459	IEC 60870-5 ASDU	60	<- I (6,1) ASDU=3 C_SC_NA_1 Act IOA=160927
44	09:58:12.684070	IEC 60870-5 ASDU	60	<- I (7,1) ASDU=3 C_SC_NA_1 Act IOA=160928
48	09:58:14.480261	IEC 60870-5 ASDU	60	<- I (8,1) ASDU=3 C_SC_NA_1 Act IOA=190202
52	09:58:16.277435	IEC 60870-5 ASDU	60	<- I (9,1) ASDU=3 C_SC_NA_1 Act IOA=260202
58	09:58:18.075091	IEC 60870-5 ASDU	60	<- I (10,1) ASDU=3 C_SC_NA_1 Act IOA=260901
62	09:58:19.871360	IEC 60870-5 ASDU	60	<- I (11,1) ASDU=3 C_SC_NA_1 Act IOA=260902
68	09:58:21.669480	IEC 60870-5 ASDU	60	<- I (12,1) ASDU=3 C_SC_NA_1 Act IOA=260903
72	09:58:23.465049	IEC 60870-5 ASDU	60	<- I (13,1) ASDU=3 C_SC_NA_1 Act IOA=260904
76	09:58:25.277649	IEC 60870-5 ASDU	60	<- I (14,1) ASDU=3 C_SC_NA_1 Act IOA=260905
80	09:58:27.074758	IEC 60870-5 ASDU	60	<- I (15,1) ASDU=3 C_SC_NA_1 Act IOA=260906
84	09:58:28.870927	IEC 60870-5 ASDU	60	<- I (16,1) ASDU=3 C_SC_NA_1 Act IOA=260907
90	09:58:30.684995	IEC 60870-5 ASDU	60	<- I (17,1) ASDU=3 C_SC_NA_1 Act IOA=260908
94	09:58:32.480503	IEC 60870-5 ASDU	60	<- I (18,1) ASDU=3 C_SC_NA_1 Act IOA=260909
98	09:58:34.277877	IEC 60870-5 ASDU	60	<- I (19,1) ASDU=3 C_SC_NA_1 Act IOA=260910
102	09:58:36.090148	IEC 60870-5 ASDU	60	<- I (20,1) ASDU=3 C_SC_NA_1 Act IOA=260911
106	09:58:37.887052	IEC 60870-5 ASDU	60	<- I (21,1) ASDU=3 C_SC_NA_1 Act IOA=260912
110	09:58:39.684105	IEC 60870-5 ASDU	60	<- I (22,1) ASDU=3 C_SC_NA_1 Act IOA=260914
114	09:58:41.480858	IEC 60870-5 ASDU	60	<- I (23,1) ASDU=3 C_SC_NA_1 Act IOA=260915
118	09:58:43.277663	IEC 60870-5 ASDU	60	<- I (24,1) ASDU=3 C_SC_NA_1 Act IOA=260916
122	09:58:45.074448	IEC 60870-5 ASDU	60	<- I (25,1) ASDU=3 C_SC_NA_1 Act IOA=260918
130	09:58:46.871366	IEC 60870-5 ASDU	60	<- I (26,1) ASDU=3 C_SC_NA_1 Act IOA=260920
134	09:58:48.668731	IEC 60870-5 ASDU	60	<- I (27,1) ASDU=3 C_SC_NA_1 Act IOA=290202
138	09:58:50.464988	IEC 60870-5 ASDU	60	<- I (28,1) ASDU=3 C_SC_NA_1 Act IOA=338501
142	09:58:52.293361	IEC 60870-5 ASDU	60	<- I (29,1) ASDU=3 C_DC_NA_1 Act IOA=1401
146	09:58:54.090015	IEC 60870-5 ASDU	60	<- I (30,1) ASDU=3 C_DC_NA_1 Act IOA=1402
150	09:58:55.886795	IEC 60870-5 ASDU	60	<- I (31,1) ASDU=3 C_DC_NA_1 Act IOA=1403
154	09:58:57.699431	IEC 60870-5 ASDU	60	<- I (32,1) ASDU=3 C_DC_NA_1 Act IOA=1404
158	09:58:59.496114	IEC 60870-5 ASDU	60	<- I (33,1) ASDU=3 C_DC_NA_1 Act IOA=1301
162	09:59:01.292947	IEC 60870-5 ASDU	60	<- I (34,1) ASDU=3 C_DC_NA_1 Act IOA=1302
166	09:59:03.089872	IEC 60870-5 ASDU	60	<- I (35,1) ASDU=3 C_DC_NA_1 Act IOA=1303
170	09:59:04.886824	IEC 60870-5 ASDU	60	<- I (36,1) ASDU=3 C_DC_NA_1 Act IOA=1304
174	09:59:06.684084	IEC 60870-5 ASDU	60	<- I (37,1) ASDU=3 C_DC_NA_1 Act IOA=1201
178	09:59:08.480310	IEC 60870-5 ASDU	60	<- I (38,1) ASDU=3 C_DC_NA_1 Act IOA=1202
182	09:59:10.277783	IEC 60870-5 ASDU	60	<- I (39,1) ASDU=3 C_DC_NA_1 Act IOA=1203
186	09:59:12.074263	IEC 60870-5 ASDU	60	<- I (40,1) ASDU=3 C_DC_NA_1 Act IOA=1204
190	09:59:13.871518	IEC 60870-5 ASDU	60	<- I (41,1) ASDU=3 C_DC_NA_1 Act IOA=1101
194	09:59:15.669191	IEC 60870-5 ASDU	60	<- I (42,1) ASDU=3 C_DC_NA_1 Act IOA=1102
198	09:59:17.465027	IEC 60870-5 ASDU	60	<- I (43,1) ASDU=3 C_DC_NA_1 Act IOA=1103
206	09:59:19.277313	IEC 60870-5 ASDU	60	<- I (44,1) ASDU=3 C_DC_NA_1 Act IOA=1104
214	09:59:23.152341	IEC 60870-5-104	50	<- U (STOPDT act)

Source

The malware is sending command after command to the client device (ASDU=3), iterating through the IOAs. Kind of similar like checking different ports on a host and trying to login.

TypeID	Type	Description
100	C_IC_NA_1	Interrogation command
45	C_SC_NA_1	Single command
46	C_DC_NA_1	Double command

From defender point of view, we obviously can not block port 2404, neither the commands used by the malware, as one or all commands are used for normal operation by the control system itself.

But looking at the TypeID transition, the malware differentiates from legitimate traffic:

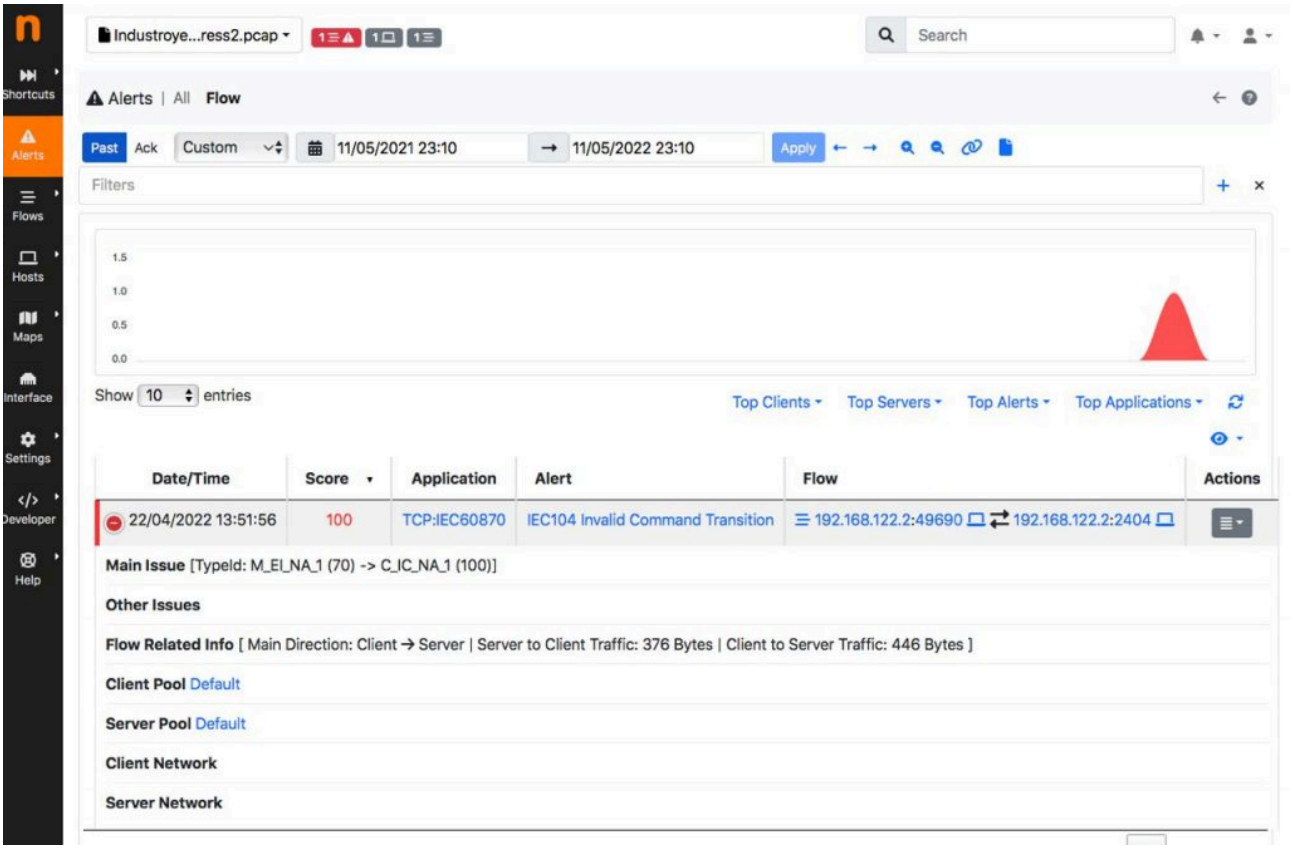
Transitions	Normal Operations Traffic	Malware
M_ to M_	> 1000	0
M_ to C_ or C_ to M_	> 0 && < 10	0
C_ to C_	0	> 10

In ntopng, three detection mechanism are build in:

- IEC Unexpected TypeID. As used TypeIDs are known by the operator, this check monitors for unknown or not allowed TypeIDs and alerts them.
- IEC Invalid Transition. In this check TypeID transitions are recorded over a predefined time period, the IEC60870 Learning Period, found under Settings / Preferences / Behaviour Analysis. An alert is generated, if a unknown TypeID transition is detected.
- IEC Invalid Command Transition is as well checking for transitions, but specifically transitions of commands. If the amount of command to command transition exceeds a threshold, an alert is generated.

All three Check can be found in the Flow Checks.

For “IEC Invalid Transition” ntopng needs a learning period in order to track transitions. Default is set to 6 hours, but most likely a longer learning period is necessary, e.g. 2 days.



Source: https://www.ntop.org/cybersecurity/how-ntopng-monitors-iec-60870-5-104-traffic/