Information Hiding Detection in Industrial Control Systems Statistical Analysis in Modbus TCP/IP

Robert Altschaffel
Advanced Multimedia and Security Lab
Otto-von-Guericke University
Magdeburg, Germany
email: Name.Surname@iti.cs.uni-magdeburg.de

Lennox Lingk
Advanced Multimedia and Security Lab
Otto-von-Guericke University
Magdeburg, Germany
email: Name.Surname@iti.cs.uni-magdeburg.de

Jana Dittmann
Advanced Multimedia and Security Lab
Otto-von-Guericke University
Magdeburg, Germany
email: Name.Surname@iti.cs.uni-magdeburg.de

Abstract—Hidden Communication is a technique increasingly employed by advanced attackers. Attacks performed by such advanced attackers on Industrial Control Systems (ICS) also recently gained relevance. This paper aims at increasing the security of ICS against attacks employing hidden communication. The detection of hidden communication is a necessary foundation to prevent non-legitimate communication within a network – potentially one used within a critical infrastructure. Besides detection, the attribution of such an advanced attack is useful to enhance future security. Therefore, we explore means to detect hidden communication in ICS using statistical methods. We demonstrate an approach based on heuristic methods and show a proof of concept for Modbus Messaging on Transmission Control Protocol/Internet Protocol (Modbus TCP/IP) including the successful evaluation with 37 network captures for ICS.

Keywords-Communication; Steganography; Attribution.

I. Introduction

Hidden Communication is a technique increasingly employed by advanced attackers; see e.g., the description of the technique Data Obfuscation: Steganography in the MITRE ATT&CK Matrix [1] or the widespread SteganoAmor campaign [2]. Also, Industrial Control Systems (ICS) are a relevant target surface.

The detection of hidden channels is a necessary prerequisite to prevent non-legitimate communication within a network – potentially one used within a critical infrastructure. The detection has to address varying embedding parameters and scenarios. Therefore, we use this paper to show how non-legitimate communication within an ICS can be detected using statistical methods.

Besides detection, the attribution of such an advanced attack is useful to enhance future security. Therefore, we explore means to identify the embedding parameters used by an attacker while hiding communication within the network traff c of an ICS. We demonstrate an approach based on heuristic methods and show a proof of concept.

The paper is structured as follows: Section II provides background information about stegomalware, steganographic terminology, and some information on ICS and Modbus Messaging on Transmission Control Protocol/Internet Protocol (Modbus TCP/IP) addressed in this paper. Section III provides an overview of a concept for the investigation of the use of statistic methods to detect hidden communication in ICS traff c including the essential creation of a test setup. Section IV describes the proof of concept of a statistical approach for the detection of steganographic messages in ICS traff c, while Section V describes the evaluation of the proof of concept. Section VI provides a summary and a discussion of limitations of the presented approach.

II. Background

This section provides a brief background on some concepts used within the scope of this paper: the terminology of hidden communication including its potential use in malicious software, ICS and the protocol Modbus TCP/IP, which is commonly employed in such systems.

A. Stegomalware and steganographic terminology

Stegomalware is a composite word of steganography and malware (which is in turn a composite word of malicious and software). Therefore, stegomalware is malicious software that uses steganographic means to hide some communication, be it the initial download of the malicious software or for command and control (C2).

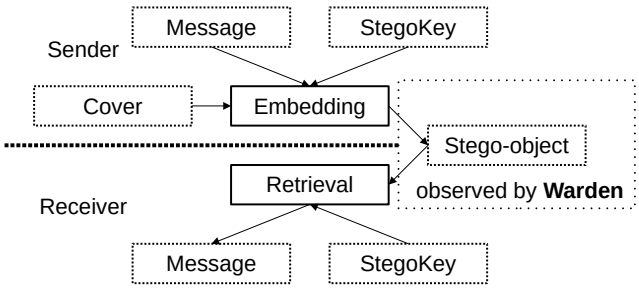

Figure 1. Communication using a steganographic channel.

The general procedure of communication can be seen in Figure 1. On the sender side, a message is embedded into a cover object by using a steganographic key. The embedding parameters vary depending on the channel and generally include information about embedding position, encoding, start or stop sequences. The resulting stego-object is then transferred to the receiver of the message and might be subject to investigation by a warden, the generic term for any security measure trying to detect hidden communication within a cover object. The receiver then uses the knowledge of the steganographic key to obtain the message.

B. Industrial Control Systems (ICS)

ICS control industrial processes. They consist of sensors that measure the physical world, actuators that manipulate the physical world and computing units that calculate, which manipulations achieve the industrial objective based on the readings provided by sensors. These computing units are commonly known as Programmable Logic Controllers (PLCs). The overall system relies on the communication between the components, which is performed using ICS-specific technologies and protocols. Some ICS protocols use Ethernet as transmission medium with Modbus TCP/IP being an example.

C. Modbus TCP/IP

The Modbus protocol is widely used in ICS. Modbus TCP/IP is an adaptation for the use of Ethernet as a carrier. Modbus/TCP uses a client/server model. The protocol itself is quite simple. Modbus/TCP uses a client/server model. The protocol itself has a simple structure as shown in Figure 2, consisting of the Mobdbus Application Protocol (MBAP) Header and the Protocol Data Unit (PDU). The MBAP Header consists of a Transaction Identifier (a sequence number used to tie requests and responses together), a Protocol Identifier (a static field), a length entry and the Unit ID that is used to identify remote peripheral devices that might be attached to a server. The PDU contains a function code, as well as data field of varying length.

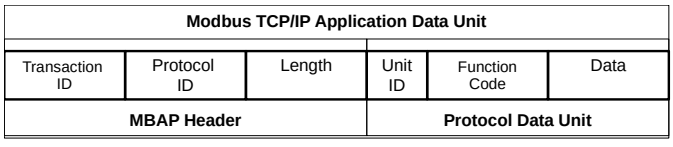

Figure 2. Modbus TCP/IP Application Data Unit.

D. Modbus TCP/IP and hidden channels

The possibility to use hidden channels in Modbus TCP/IP communications has been explored in [4] by using an extended taxonomy of network information hiding patterns (see [5]) and exploring their application in the specific case of Modbus TCP/IP. It identifies 14 theoretical patterns and evaluated their capacity as well as requirements such as whether they could be implemented on the server and/or the client-side. Furthermore, four patterns are implemented for testing.

The storage pattern *S6 Reserved/Unused* is of special relevance to this paper. It uses the Unit ID field in the Modbus TCP/IP protocol to embed up to one byte of data.

E. Attribution and embedding parameters

Attribution relies on identifying properties of the attacker. These properties include the capabilities of the attacker, the tools and techniques used as well as the parameters used. These parameters include the embedding parameters used by a steganographic method.

F. Usage of statistical methods in computer security

Statistical methods have been used in the scope of computer security in various forms. They act as a foundation for pattern recognition. A primary benefit of using statistical methods in this approach is the fact that approach can be easily explained and understood; especially in contrast to complex machine learning algorithms where the field of explainability currently develops into a relevant research field. Explainability is fundamental for attribution since it allows to comprehend the reasoning over the result of a algorithm.

III. Concept

We want to show that statistic methods can be used to detect hidden communication in ICS traffic as a first step. As the second step we aim to obtain the embedding parameters used by steganographic methods in specific scenarios.

This process is described in detail in Section IV. However, to evaluate the applicability of such an approach, a data set of network recordings with and without steganographic embeddings is necessary.

In this work, we use steganographic embeddings performed within Modbus TCP/IP based on the results presented in [4].

Our approach uses a test set of Modbus TCP/IP traffic without steganographic embedding (*Cover*). Then, we perform steganographic embeddings on these Modbus TCP/IP recordings (*Steganographic Embeddings*) with varying embedding parameters. Finally, we analyse the Modbus TCP/IP traffic in order to detect steganographic embedding and to obtain the embedding parameters. Network recordings in the pcap-file format were used.

A. Cover Data Set: Modbus TCP/IP traffic

As a foundation, the publicly available data sets *MB-Base-1* [6] and *MB-Base-2* [7] are used. Schneider Modicon PLCs were used to create these two data sets. These Schneider PLCs act as client and server, respectively. Data is transferred cyclically between client and server.

MB-Base-1 [6] consists of two subsets. The subset *MB-Base-1-1* consists of 9 recordings of the communication between one client and two servers for about 10 minutes, each. The number of transferred data fields and the cycle time are varied among the recordings. The subset *MB-Base-1-1* contains two recordings of the communication between one client and one server for 10 and 70 minutes, respectively.

MB-Base-2 [7] consists of three recordings of the communication between one client and one server for 2 hours, each.

The number of transferred data fields and the cycle time are varied among the recordings.

B. Steganographic Embeddings: Storage Channel S6

The steganopgrahic pattern S6 from [4] was re-implemented in the tool timeembedder [8]. The goal was to enable a batch processing of various embeddings to create a broad data set by varying the embedding parameters and the embedded message. This implementation of the pattern S6 uses five embedding parameters:

- **startCode** Numeric start code for the embedding process in the packets that mark the start of the message embedding, Length: 3
- **endCode** Numeric end code for the embedding that marks the end of the embedded message, Length: 3
- **skipSize** Number of skipped packets between embeddings
- **oneCode** Numeric code for the embedding of a 1, e.g., 121 could be a code for 1, Length: 3
- **zeroCode** Numeric code for the embedding of a 0, e.g., 122 could be a code for 0, Length: 3

The storage channel S6 uses the field Unit ID of the MBAP header. In addition, this implementation fills the Unit ID with random values to avoid the suspicion that any non-zero values obviously belong to a steganographic transmission.

We use the data set *MB-Embed-1* [9], which was created using this algorithm and are publicly available. In total, this data sets contain 3 captures with steganographic embedding using the pattern S6. The data set also contain additional captures using other steganographic patterns described in [4] that are irrelevant for this paper.

C. Additional data sets for evaluation of detection performance

Due to the general handling of the field Unit ID in the MBAP header, further data sets are required to evaluate the performance of detecting the steganographic embedding. These data sets must mimic the behaviour of the changed Unit ID entries without embedding a steganographic message. Thus, a data set of 11 captures with the changed Unit ID behaviour but without steganographic embedding has been created.

However, traffic captures from general ICS were also used to evaluate the approach presented in this paper. A test set of Modbus TCP/IP network captures was compiled from public sources in [10] (it contains network captures from [11], [12] and [13]). This publicly available test set contains 8 network captures (see [14]).

IV. Statistical analysis to detect steganographic embedding using the storage channel S6 in Modbus TCP/IP

This section describes how steganographic embedding using the storage channel S6 in Modbus TCP/IP can be detected. This detection could be performed by a warden residing on the network of the Modbus TCP/IP communication. At first, a preprocessing is necessary in order to extract the Unit ID fields from the network recordings. Then, statistical analysis is performed following some assumptions about the network recordings, which are discussed in the following.

A. Preprocessing

As a first step, a preprocessing is necessary in order to extract the Unit IDs from the network recordings. The network recordings were available in the pcap-file format (see [15] for specification). The file format contains a file header and the packet records, which include the network packets and timing information.

The tool NWD [16] strips the file header and detects Modbus TCP/IP packets in network recordings and outputs timing information, as well as the MBAP headers, including the field Unit ID that is used by the S6 steganographic pattern.

B. Statistical assumptions

There are some underlying assumptions for the statistical analysis based on the behaviour of hidden channels in network streams:

- A start sequence (StartCode in the implementation of the pattern S6 used in this work; see Section III-B) can only appear to a very limited extend within a given network capture; the retrieval relies on the presence of such start sequence.
- The end sequence (EndCode) should be rare since random occurrences of the end sequence after the start sequence would disrupt the retrieval by cutting a message short.
- Between the start and end sequence, the sequences encoding the message (oneCode and zeroCode) will be overrepresented.

These assumptions hold true whether there is one embedded message or multiple messages using the same embedding key within a given network recording. For the proof of concept presented here, we assume only one embedded message per given network recording. Thus, a start sequence will only occur once.

C. Statistical Analysis

Based on these assumptions, the potential start sequence is identified by processing the network recording and searching for the least occurring instances of the Unit ID. This can be done by creating a histogram over the entire network recording and picking the rarest instance of Unit ID. These are the potential candidates for the start and end sequence.

In the next step, permutations of the specific pairings of start and end sequences are created and checked whether these pairings appear in said ordering within the data set.

If a suspected start sequence - end sequence segment is identified within the network capture, the next step is to extract the segment between these two markers for further statistical analysis.

It is assumed that the sequences encoding the message are overrepresented in the segment between start sequence and end sequence. Therefore, a histogram over the segment is

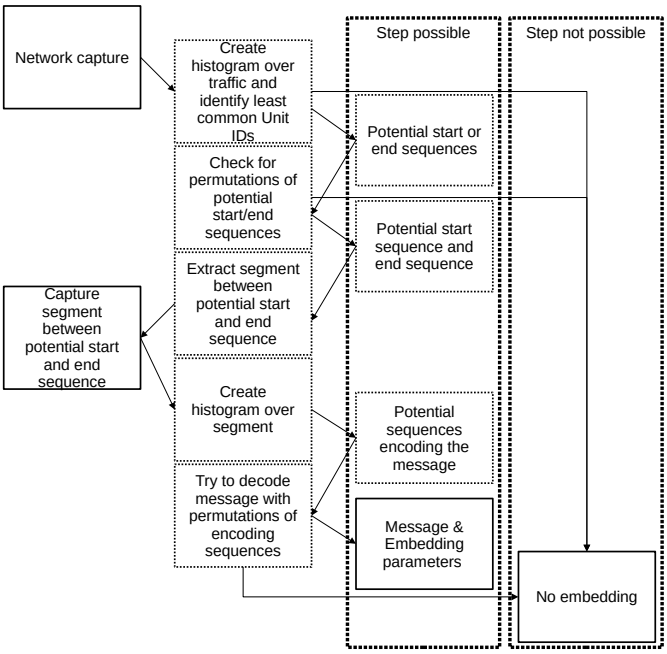

Figure 3. Process for the statistical analysis to find embeddings using the S6 steganographic pattern.

calculated to identify the most common occurrences of Unit IDs as candidates for oneCode and zeroCode, respectively.

At this point, the suspected oneCode and zeroCode can be used to try to retrieve the message. As such, any occurrence of the suspected codes is translated to a zero or a one, respectively. As it is not possible to identify, which of the candidates is the oneCode or the zeroCode at this point, both permutations have to be tried in order to check whether they lead to the retrieval of the message.

In addition, the distance between sequences identified as either oneCode or zeroCode within the segment can be used to calculate the SkipSize from the distance between the respective segments. At this step, the entire embedding key is obtained.

The entire process is visualized in Figure 3.

V. Evaluation

A proof of concept was created from the approach described in Section IV within the tool uidhist.pl [17] for evaluation.

The captures described in Section III-B and Section III-C were used for the evaluation of the detection mechanisms. In total, 37 network captures were used for the evaluation of the detection approach. All instances of the embedding were successfully detected with no false positives. An overview on the evaluation results can be seen in Table I.

Dataset MB-Embed-1 contained recordings with embedded steganographic messages using the S6 steganographic patterns. In these cases, the embedding was identified and the correct embedding parameters were obtained by the use of the presented approach. A screenshot of the exemplary output of the approach can be seen in Figure 4. The embedding parameters were correctly identified as shown in Table II.

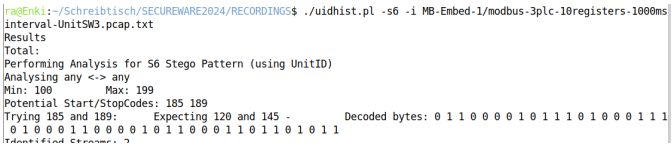

Figure 4. Screenshot from the application of the modified *uidhist.pl* on capture 18. The identified embedding parameters are visible as output.

In the case of the other evaluation data, no steganographic messages were embedded and none were detected. In the case of captures with few packets, potential candidates for the start and end sequence were detected (e.g., with captures 24 and 29). This is due to the fact that these captures do not contain enough packets for the statistical distribution to fully develop. However, in these cases no message could be detected and hence, the no embedding was detected.

This shows that the present approach enables promising results with the data sets used within this research.

VI. Conclusion and Future Work

This paper shows the viability of using statistic means to detect steganographic communication in ICS communications on the example of Modbus TCP/IP traffic. An approach for detection was presented and evaluated with 29 network captures showing promising results. In addition, the identification of embedding parameters was shown to be possible using the same approach. It was successfully evaluated with the network recordings that contained embedded steganographic messages. A clear limitation is that the approach is tied to the specific embedding pattern and to very similar patterns. Similar detection mechanisms can be devised for other parts of the MBAP header or for parts of other protocols. Another limitation is the use of our own data set for testing and evaluation in this work.

Future work is focused on the application of the proposed approach to other ICS protocols and steganographic patterns. In terms of ICS protocols, mainly Open Platform Communications Unified Architecture (OPC UA) and MQ Telemetry Transport (MQTT) are of interest due to their widespread use. So far, the application to MQTT seems promising. In terms of other steganographic patterns, the application to User-data Value Modulation and Reserved/Unused (S10 in [4]) forms the next goal in our research.

Acknowledgment

The work contributed by Lennox Lingk [6], [7], [8], [9], as well as the work on the usage of statistical measures in order to identify parameters of steganographic embedding has been performed in the research project ATTRIBUT (https://omen.cs.uni-magdeburg.de/itiamsl/deutsch/projekte/attribut.html). This work has been supported by the Agentur für Innovation in der Cybersicherheit GmbH. The Agentur für Innovation in der Cybersicherheit GmbH did not interfere in the research process and its results.

The tool NWD [16] was used in an older version created in the scope of the project “SYNTHESIS - Synthetically

TABLE I
RESULTS OF THE EVALUATION OF THE DETECTION APPROACH AGAINST THE VARIOUS DATA SETS.

Number	Filename	Dataset	Duration	Modbus Pakets	Embedding	
					present	detected
1	modbus-3plc-1registers-200msinterval	MB-Base-1	10:02	6024	✗	✗
2	modbus-3plc-5registers-200msinterval	MB-Base-1	10:00	6011	✗	✗
3	modbus-3plc-10registers-200msinterval	MB-Base-1	10:01	6020	✗	✗
4	modbus-3plc-1registers-500msinterval	MB-Base-1	10:05	2420	✗	✗
5	modbus-3plc-5registers-500msinterval	MB-Base-1	10:01	2048	✗	✗
6	modbus-3plc-10registers-500msinterval	MB-Base-1	10:03	2416	✗	✗
7	modbus-3plc-1registers-1000msinterval	MB-Base-1	10:04	1212	✗	✗
8	modbus-3plc-5registers-1000msinterval	MB-Base-1	10:46	1296	✗	✗
9	modbus-3plc-10registers-1000msinterval	MB-Base-1	9:22	1124	✗	✗
10	modbus-2plc-10registers-1000msinterval-70mins	MB-Base-1	1:10:23	33792	✗	✗
12	modbus-2plc-10registers-1000msinterval	MB-Base-1	10:02	4832	✗	✗
13	10_registers_2h_1000ms	MB-Base-2	2:03:00	95762	✗	✗
14	120_registers_2h_1000ms	MB-Base-2	2:12:22	1932985	✗	✗
15	240_registers_2h_1000ms	MB-Base-2	2:35:17	2266247	✗	✗
16	modbus-3plc-1registers-200msinterval-UnitSW1	MB-Embed-1	10:02	6024	✓	✓
17	modbus-3plc-5registers-500msinterval-UnitSW2	MB-Embed-1	10:01	2024	✓	✓
18	modbus-3plc-10registers-1000msinterval-UnitSW3	MB-Embed-1	9:22	1124	✓	✓
19-29	modbus-3plc-1registers-...					
19	...200msinterval-RandomUID-NoMessage	Created	10:02	6024	✗	✗
20	...200msinterval-RandomUID-NoMessage-1	Created	10:02	6024	✗	✗
21	...500msinterval-RandomUID-NoMessage-1	Created	10:05	2420	✗	✗
22	...500msinterval-RandomUID-NoMessage-2	Created	10:05	2420	✗	✗
23	...500msinterval-RandomUID-NoMessage-3	Created	10:00	6011	✗	✗
24	...1000msinterval-RandomUID-NoMessage-3	Created	10:04	1212	✗	✗
25	...200msinterval-RandomUID-NoMessage-4	Created	10:00	6011	✗	✗
26	...1000msinterval-RandomUID-NoMessage-6	Created	10:46	1296	✗	✗
27	...200msinterval-RandomUID-NoMessage-7	Created	10:01	6020	✗	✗
28	...500msinterval-RandomUID-NoMessage-8	Created	10:03	2416	✗	✗
29	...1000msinterval-RandomUID-NoMessage-9	Created	9:22	1124	✗	✗
30	eth2dump-clean-0,5h_1	[11]	0:30:00	35430	✗	✗
31	eth2dump-clean-1h_1	[11]	1:30:00	72150	✗	✗
32	eth2dump-clean-6h_1	[11]	6:00:00	427842	✗	✗
33	mb	[12]	1:30	55800	✗	✗
34	run8	[13]	1:00	72186	✗	✗
35	run11	[13]	1:00	72489	✗	✗
36	run1_3RTU_2s	[13]	1:00	305932	✗	✗
37	run1_6RTU	[13]	1:00	134690	✗	✗

TABLE II
EMBEDDING PARAMETERS OBTAINED BY THE DETECTION APPROACH IN THE CAPTURES THAT CONTAINED EMBEDDINGS.

Number	Filename	StartCode		EndCode		OneCode/ZeroCode	
		used	found	used	found	used	found
16	modbus-3plc-1registers-200msinterval-UnitSW1	185	185	189	189	120; 145	120; 145
17	modbus-3plc-5registers-500msinterval-UnitSW2	185	185	189	189	120; 145	120; 145
18	modbus-3plc-10registers-1000msinterval-UnitSW3	185	185	189	189	120; 145	120; 145

generated data segments with hidden malicious code functions for safety analysis in nuclear control technology" with the grant number FKZ: 1501666A which is funded by the Federal Ministry for the Environment, Nature Conservation, Nuclear Safety and Consumer Protection (BMUV). The tool uidhist.pl [17] which was used as a basis for the evaluation of the detection approach also originates from this project.

In addition, we would like to thank Dr-Ing. Stefan Kiltz for his support with formatting as well as Stefan Seidlitz for presenting the results at the conference.

REFERENCES

[1] MITRE ATT&CK, "Techniques - Data Obfuscation: Steganography ," https://attack.mitre.org/techniques/T1001/002/, 2020, [retrieved: October, 2024].

[2] A. Badaev and K.Naumova, "SteganoAmor campaign: TA558 mass-attacking companies and public institutions all around the world,"

Positive Technologies, https://www.ptsecurity.com/ww-en/analytics/pt-esc-threat-intelligence/steganoamor-campaign-ta558-mass-attacking-companies-and-public-institutions-all-around-the-world/, 2224.

[3] J. Dittmann, C. Kraetzer, J. Alemann, and B. Birnbaum, "Forensic Trace Analysis for MP3 based Stego-Malware: Exemplary Study for Stego-Algorithm and Capacity Attribution to derive YARA Rules for Malware Identification," Proceedings of the 2024 ACM Workshop on Information Hiding and Multimedia Security, pp. 101–112, https://dl.acm.org/doi/10.1145/3658664.3659641, 2024.

[4] K. Lamshoeft and J. Dittmann, "Assessment of Hidden Channel Attacks: Targetting Modbus/TCP," 21st IFAC World Congress, vol. 53, pp. 529–551, https://www.sciencedirect.com/science/article/pii/-S240589632030536X, 2020.

[5] W. Mazurczyk, S. Wendzel, and K. Cabal, "Towards deriving insights into data hiding methods using pattern-based approach," 13th International Conference on Availability, Reliability and Security, pp. 1-10, https://dl.acm.org/doi/10.1145/3230833.3233261, 2018.

[6] MB-Base-1 (Modbus-Schneider-Basis), https://datasets-amsl.cs.uni-magdeburg.de/index.php/s/YEpM7Nx6FPjdWJe, released 05.06.2024, [retrieved: October, 2024].

[7] MB-Base-2 (Modbus-Schneider-Basis), https://datasets-amsl.cs.uni-magdeburg.de/index.php/s/YEpM7Nx6FPjdWJe, released 05.06.2024, [retrieved: October, 2024].

[8] L. Lingk, *timeembedder*, released 26.07.2024, [retrieved: October, 2024].

[9] MB-Embed-1 (Modbus-Schneider-Embed), https://datasets-amsl.cs.uni-magdeburg.de/index.php/s/YEpM7Nx6FPjdWJe, released 05.06.2024, [retrieved: October, 2024].

[10] *[GERMAN]* R. Altschaffel, S. Kiltz, K. Lamshöft, and J. Dittmann, "ICS/OT-Sicherheit: Evaluation und Validierung der Erkennungsleistung von Stego-Malware in industriellen Steuernetzwerken mittels Synthese und Simulation," [Translation: ICS/OT Security: Evaluation and Validation of the Detection Performance against Stegomalware in ICS using Synthesis and Simulation], Kongressdokumentation zum 20. Deutschen IT-Sicherheitskongress des BSI, pp 333-348, 2024.

[11] I. Frazão, P. H. Abreu, T. Cruz, H. Araújo, and P. Simões, "Denial of Service Attacks: Detecting the frailties of machine learning algorithms in the Classification Process," in 13th International Conference on Critical Information Infrastructures Security (CRITIS 2018), ed. Springer, Kaunas, Lithuania, September 24-26, 2018, Springer series on Security and Cryptology , 2018. DOI: 10.1007/978-3-030-05849-4_19

[12] https://github.com/ITI/ICS-pcap/tree/master [retrieved: October, 2024].

[13] A. Lemay and J. M. Fernandez, "Providing SCADA network data sets for intrusion detection research," in 9th Workshop on Cyber Security Experimentation and Test (CSET 16), 2016; https://www.usenix.org/-conference/cset16/workshop-program/presentation/lemay [retrieved: October, 2024]

[14] R. Altschaffel, "Modbus Fingerprints - BSI Paper," https://gitti.cs.uni-magdeburg.de/raltschaffel/modbus-fingerprints-bsi-paper [retrieved: October, 2024]

[15] Network Working Group, "PCAP Capture File Format," https://www.ietf.org/archive/id/draft-gharris-opsawg-pcap-01.html, 2020, [retrieved: October, 2024].

[16] R. Altschaffel and R.Mecke , *nwd* version 0.2, https://gitti.cs.uni-magdeburg.de/raltschaffel/nwd/-/blob/23ae26a3b699b10d0d1df2f9d23d3e2c9c668bac/nwd.c, [retrieved: October, 2024].

[17] R. Altschaffel, *uidhist.pl* version 1.52, https://gitti.cs.uni-magdeburg.de/raltschaffel/nwd, [retrieved: October, 2024].