Hidden-Non-Malicious-Dummies for Evaluation of Defense Mechanisms of Industrial Control System against Steganographic Attacks

Robert Altschaffel, Stefan Kiltz, Jana Dittmann
Otto-von-Guericke University of Magdeburg
Magdeburg, Germany
e-mail: firstname.lastname@iti.cs.uni-magdeburg.de
Tom Neubert, Laura Buxhoidt, Claus Vielhauer
Brandenburg University of Applied Sciences
Brandenburg (Havel), Germany
e-mail: firstname.lastname@th-brandenburg.de
Mathias Lange, Rüdiger Mecke
Magdeburg-Stendal University of Applied Sciences
Magdeburg, Germany
e-mail: firstname.lastname@h2.de

***Abstract*—Cyber-Security in Industrial Control Systems (ICS) is a topic of growing relevance. Attack scenarios include the exfiltration of critical process data, the infiltration of commands and the manipulation of the controlled physical processes. Machine learning based detection mechanism are employed against these attacks. However, such machine learning based approaches rely on training data. This paper addresses two core challenges with regards to such machine learning approaches: 1. the required training data containing such attacks is usually difficult to obtain and 2. information about the detection rates is necessary in order to deploy the mechanisms for detection in a fashion benefiting security incident management. As such, this paper discusses an approach to generate such training data containing hidden non-malicious-dummy data representing attacks for five different attack scenarios, means to ensure that these dummies do not negatively affect the system under test, different strategies for injection and detection. This synthetically generated facility-specific data is then used for evaluating the usefulness of such machine learning detection approaches in ICS security management.**

Keywords-SCADA; hidden-non-malicious-dummy; cyber-security

I. Introduction

Industrial Control Systems (ICS) are under a rising threat from cyber-attacks, as shown by the trends identified in [1]. ICS processes directly affect the physical world (hence, they are often referred to as cyber-physical systems). An attack on the security of an ICS might compromise the safety of the surrounding facility, staff, bystanders or even those dependent on the services of the facility. As discussed in [1], Advanced and Persistent Threat actors (APTs) play a significant role in this threat scenario.

APTs are able to use advanced techniques including steganographic means to facilitate illicit communication flows (e.g., to infiltrate malicious payload, to exfiltrate data about the facility or for outright command&control of deployed malware). Such steganographic means are included in the MITRE ATT&CK Matrix [2] under the technique *Data Obfuscation: Steganography*. Another example from Desktop IT is the use of (primitive) steganographic means in the the widely-spread malware campaign SteganoAmor (see [3]).

Commonly, machine learning based detection mechanics are employed to counter attacks on ICS. Such approaches rely on the presence of training data including those of cyber-attacks in order to create models able to discern between legitimate network behavior and attacks.

Obtaining such training data faces different challenges (e.g., the data in itself contains critical information about the facility and is hence not made available to outsiders, the data is always facility-specific, some facilities might not have monitored any cyber attacks).

Hence, means to create 'known-bad' training data without compromising a given facility are necessary to support detection mechanisms. As such, this paper discusses an approach to generate such training data containing hidden non-malicious-dummy data representing attacks. Furthermore, the marking of these hidden-non-malicious dummies in a way to prevent any harm during testing procedures is discussed.

This paper furthermore explores the use of an Open Source Machine Learning suite to protect against attacks using steganographic means. The generation of a facility-specific training data set as well as the training of models. These models and their application is then evaluated.

This paper is structured as follows: After this introduction, Section II provides an overview on ICS terminology, communication protocols addressed within this paper and steganograpy employed in ICS. Section III describes the various scenarios of non-malicious-dummies and as well as their injection and their detection. Section IV discusses the creation of a test set and how it can be used to evaluate a detected approach. V discusses the results of the evaluation. Section VI provides the conclusion and a discussion on how to mark hidden non-malicious-dummies in the future to ensure that no facilities are damaged during security tests.

II. State-of-the-Art

This section provides a brief background on Industrial Control Systems (ICS) in general, the network protocol OPC UA commonly used in ICS, the terminology of steganography and the conjunction of ICS and steganograpy.

A. Industrial Control Systems (ICS)

ICS govern industrial processes. They encompass sensors (to measure the physical world), actuators (to affect the physical world), computing units (to calculate the intended control signals for the actuators based on sensor readings and operator input) and networking enabling all the communication required between these components. The computing units are generally known as Programmable Logic Controllers (PLCs). Other components also employed within the context of ICS comprise Human-Machine-Interfaces (HMI) that are used by operators to access sensor readings or to affect the actuators.

Various terms are used to describe the domain of ICS: Operational Technology (OT) or SCADA (Supervisory Control and Data Acquisition) describe functions and parts common in ICS. Field device is another term often used for ICS components located in a production field.

To enhance descriptive accuracy, we use the Purdue Enterprise Reference Architecture (PERA) [4]. The PERA describes the system hierarchy common to ICS in six levels. The exact definition and naming of these levels shifted over the years, but [5] identifies the following levels: *Level 0 - Process* (sensors and actuators involved in the basic manufacturing process), *Level 1 - Basic Control* (controllers that direct and manipulate the manufacturing process), *Level 2 - Area Supervisory Control* (Cell/Area zone runtime supervision and operation) (incl. operator interfaces or alarms), *Level 3 - Site Manufacturing Operations and Control*, *Level 4 - Site Business Planning and Logistics* (basic business administration tasks), and *Level 5 - Enterprise* (centralized IT systems and functions). These levels are grouped into specific zones. Levels 0, 1 and 2 are grouped into the *Cell/Area Zone*. Levels 0, 1, 2 and 3 represent the *Manufactoring Zone*. Levels 4 and 5 comprise the *Enterprise Zone*. The use of these levels of the control hierarchy enables a more accurate description than the terms ICS, OT or SCADA. In this paper, we are concerned with the levels 0-2 (The *Cell/Area Zone*).

B. OPC Unified Architecture Industrial Control Systems (OPC UA)

Within the *Manufacturing Zone*, the use of ICS-specific communication protocols is common. One of these protocols is the OPC Unified Architecture (OPC UA) protocol. OPC UA is an open standard (including an open source reference implementation) to facilitate the communication between various ICS components. TCP/IP is often used as a foundation for the network connection [6].

OPC UA follows a client/server-model. A range of clients connects to a specific OPC UA server. On *Control Hierarchy Level 1*, the OPC UA server is usually provided by a computing unit (the PLC) with the sensors and actuators connected as clients to this server.

C. Steganography in ICS

According to the recent work of [7]: "*Steganography is the art and science of concealing the existence of information transfer and storage*". Steganography has several subdomains such as: *text steganography*, *digital media steganography*, *file system steganography* and *cyber-physical systems steganography*. Each of these exemplary subdomains has different characteristics and requirements. The relevant subdomain for steganography in ICS is cyber-physical systems steganography and is characterized by a limited channel capacity and ICS-specific network protocols. There is usually a lower amount of available data for a potential embedding in ICS networks compared to traditional IT networks. Additionally, transmitted network packets are usually significantly smaller in ICS since only few (sensor) values or meta-data are transferred. ICS-specific network protocols, like Modbus TCP or OPC UA, are often encapsulated in TCP/IP (or other transport protocols). This creates the opportunity to utilize the data fields of the ICS-specific protocols in addition to TCP/IP protocol headers [8]. A further domain-specific characteristic is that the ICS-specific payload is transmitted unencrypted in many or at least some cases, because ICS are often considered closed networks.

From the attackers point of view, the embedding of hidden information can be realized by steganographic techniques (e.g, manipulating network packets payload by altering time intervals, time stamps or sensor values on least significant digits in specific selected packets). The attackers goal is that the packets seem inconspicuous for a potential warden (e.g., intrusion detection system) observing the network traffic.

A unified definition of terms and their applicability in steganographic context is provided in [7].

D. Steganograhic Attack Vectors in ICS

Since the last decade, stealthy malware or information hiding based malware is increasingly used by attackers, confirmed for example by the attack vectors presented in [2]. The well-known Stuxnet-Attack [9] proves that attackers use information hiding techniques to compromise ICS or cyber-physical systems since the last decade. During the attack, lnk-files have been utilized as cover data and in-memory code injections have been used to conceal the attack. Further attacks with stealthy malware on ICS, like the Ukrainian [10] and the Indian power grid attack [11], show that attacks on ICS and other cyber-physical systems are more and more common.

Basically, stealthy malware uses steganographic techniques to embed and inject or extract data in ICS. Therefore, attacker use stealthy malware to stay undetected for as long as possible to establish command and control channels (e.g., to trigger malfunctions or to exfiltrate confidential data).

Additional relevant attack vectors for information hiding based malware in ICS are discussed in [12].

III. HIDDEN-NON-MALICIOUS-DUMMIES

The central aim of this paper is to provide 'known-bad' training data of cyber-attacks containing steganographic means without compromising a given facility. These training data encompasses the network communication of an ICS. Hence, we face some general conceptional requirements for this data:

- **The data must be facility-specific**: Facilities come in diverse configurations, each leading to a different base line communication behavior. The inclusion of different sensors and actors leads to a differences in communication behavior, which could lead to the detection of anomalous behavior by using non-facility-specific data even if no attack and no steganographic communication is present at all.
- **The data must be attack-specific**: A dummy can have specific properties or requirements due to the category of an attack (see Section III-A).
- **The resulting data must not trigger any damage to the facility in question**: A dummy must never damage or destroy a target system.
- **The data must contain steganographic communication**: As we aim to evaluate a security measure's capability of detecting attacks containing steganographic communication, the inclusion of steganographic communication is necessary.

We term such training data as containing *hidden-non-malicious-dummies*. Our terminology comprises two parts: The non-malicious-dummies itself, and the hiding mechanism (provided by steganographic means).

A. The Non-Malicious-Dummy

The non-malicious-dummy is the message itself transmitted by stenographic means. As stated before, this message must not trigger harm to the specific facility. On the other hand, the message should mimic properties of messages used in cyber-attacks. Different types of attacks might affect the requirements for these non-malicious-dummies.

These attacks generally fall under the following broad categories:

- **Infiltration**: Data is infiltrated into the ICS. This data would include malicious commands, malicious binary code or manipulated documentation.
- **Exfiltration**: Confidential data is exfiltrated from the ICS into another network. This typically includes data like network scan information or process data used as reconnaissance for follow-up attacks or lateral movement in the scope of a complex cyber-attack scenario (e.g., as described by the Cyber Kill Chain [13]). Other potential targets include source code, binary objects or construction documents.
- **Command and Control** (C&C, C2): general command&control communication involving a two-way communication (e.g., queries and results).

These categories motivate five scenarios for the use of specific types of non-malicious-dummy messages. These scenarios aim at covering a broad range of these potential categories:

- **Scenario$_1$: Plain text documents**; this could be relevant during an exfiltration. This would include automatically generated text or placeholder text (e.g., Lorem Ipsum)
- **Scenario$_2$: Multimedia Files**, those could also be relevant mostly for exfiltration scenarios. This comprises placeholder files in standard document formats (e.g., JPEG for images, PDF for documents).
- **Scenario$_3$: Binary Codes**; this could be relevant for exfiltration or infiltration scenarios. This includes binary files, which include the common headers for the respective architecture but do not cause any malicious execution.
- **Scenario$_4$: C&C Control Commands**; this could be relevant for a Command and Control scenario. It includes a list of common unspecific command words used in the context of controlling deployed malicious software in plain text, e.g., *START*, *SET*, *TRANSFER*. These will have no function without deployed malware on the communication partners.
- **Scenario$_5$: Control Commands & Sensor readings**; this could be relevant in all categories. This scenario requires the greatest knowledge of the ICS in question since it encompasses sensor readings and control commands that do not affect the ICS in question (e.g., sensor readings from sensors not present within the ICS).

Furthermore, it must be ensured that the non-malicious-dummies do not cause any harm to the systems tested. This can be supported by marking the non-malicious-dummy, which is discussed in VI.

B. Injection of hidden non-malicious-dummies

In this section, we specify how hidden non-malicious-dummies without malicious effect can be injected into realistic, legitimate network traffic cover data without causing damage to the ICS. We identify the following approaches:

- **Direct injection:** For the direct injection of hidden non-malicious-dummies into the running network traffic of an ICS a corrupted setup with a "man-in-the-middle" (MitM) is required and should be modeled as a non-malicious simulation in a research lab. In this injection method, the non-malicious simulation can use scripts that select, intercept, modify and then forward selected network packets.
- **Injection through network recording:** Based on the Synthetic Steganographic Embedding (SSE) concept presented in [14], recorded network traffic from ICS can be subsequently modified synthetically. The SSE-concept offers the possibility to embed hidden information everywhere in recorded network cover data with a fast embedding pace near real time. The SSE-concept has two synthetic embedding options. Synthetic Embedding Option A (SEO$_A$) focuses on a high embedding pace and SEO$_B$ on a more comfortable and easier to handle embedding, due to access to structural elements of a network packet.

In the evaluation of this work, the steganographic hidden non-malicious-dummies are injected into the cover data through network recording using SEO$_A$ as part of the SSE-concept.

C. Detection of hidden non-malicious-dummies

Several detection approaches [8], [15], [16] of steganographic techniques used by attackers have been elaborated. A general overview of potential defense mechanisms for steganographic

network data is introduced in [17]. Additionally, an extended analysis testbed for steganographic network data to evaluate detection and defense mechanisms is presented in [8]. Machine learning driven approaches based on handcrafted feature spaces with as much discriminatory power as possible are well-suited for the detection of steganographic network data because they offer the opportunity for a comprehensive and explainable classification of samples (i.e., steganographic network data). In our evaluation (see Section IV), we train a classifier based on an existing handcrafted feature space to distinguish between steganographic network data samples with embedded hidden non-malicious-dummies and cover network data samples.

IV. Conceptual Approach for an Evaluation Setup for Hidden-non-malicious-dummies

In this section, we present our evaluation setup including our evaluation goals and metrics (Section IV-A), our evaluation data with our laboratory setup and a potential attack vector (Section IV-B), our exemplary steganographic embedding method which is used encode the hidden non-malicious-dummies into cover data and the resulting captured evaluation data (Section IV-C). Additionally, the detection approach for our evaluation is briefly outlined (Section IV-D).

A. Evaluation Goals and Metrics

In our evaluation, the goal is to determine if a selected state-of-the-art detection approach (Section IV-D) is able to detect samples with exemplary embedded hidden non-malicious-dummies and if the approach can distinguish between a sample with embedded hidden non-malicious-dummies and an unaltered cover data sample. To achieve our goals, we split our evaluation data with a 5-fold cross validation (Figure 1). For the determination of the performance of the detection approach we use the following well-known forensic success- and error rates:

- **True Positive Rate** (**TPR**, number of correctly classfied altered samples (with embedded dummy) in relation to number of all altered samples),
- **True Negative Rate** (**TNR**, number of correctly classfied unaltered samples in relation to number of all unaltered samples),
- **False Miss Rate** (**FMR**, number of incorrectly classfied altered samples in relation to number of all altered samples) and
- **False Alarm Rate** (**FAR**, number of incorrectly classfied unaltered samples in relation to number of all unaltered samples).

The determined classification performance of the used detection approach is presented in Section V.

B. Evaluation Data including Laboratory Setup and Attack Vector

For the evaluation of the detectability of exemplary hidden non-malicious-dummies, an uncompromised laboratory ICS setup is required to record uncompromised network traffic, i.e., cover data. As discussed in Section III-B), we use this cover

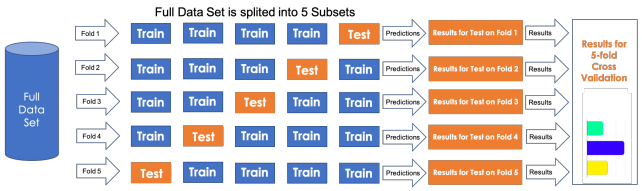

Figure 1. Generic description of the 5-fold cross validation process

data for the embedding of the introduced hidden non-malicious-dummies. We inject the dummies afterwards into the cover data recording with the previously mentioned SSE-concept from [14] without the need to compromise our laboratory ICS setup. To record uncompromised cover data, we build an ICS setup with multiple components, visualized in Figure 2.

This CP-Lab is a customizable educational system that integrates current Industry 4.0 standards. The system is made up of two islands, each consisting of four different modules (module describes a combination of base module, application module and Siemens TOUCH HMI) (see Figure 2). The two islands are physically connected by the transport robot called "ROBOTINO". This robot transports the pallet carriers between the two islands. The base module consists of a control cabinet with the control technology for the conveyor belt and the application module, which has task-specific components, e.g., a press, a drill, a magazine and even a Festo UR-5 robot. Only "The Branch" module is controlled by a FESTO PLC (one for each island). The others have a SIMATIC ET 200 Open Controller with a CPU 1515 SP. Each module is connected to a Siemens SCALANCE XB008 switch via PROFINET [**pn**]. The switches are in turn connected with each other via a central switch, which bundles all data traffic between the host computer and the system. This allows each module to communicate with each other. The OPC-UA server, a web store and a Manufacturing Execution System (MES) are located on the host computer. The host computer focuses on controlling a single production line or section, while the MES controls the entire production process. Both systems require data acquisition, process control and visualization to improve production. After start-up, the system is in automatic mode. In this operating mode, the conveyor belts move the pallet carriers permanently. During this time, all modules communicate permanently with the host computer, transmitting the position of the pallet carriers to the host computer and waiting for new commands, such as an order triggered by the MES. No SIEM systems exist in this test setup during the test period in order to document the effects of attacks on an unprotected system and to generate test data. For documentation purposes a switch with a mirror port connecting the host computer and the modules is integrated. The switch mirrors all traffic to a mirror port where it is recorded using a notebook running Wireshark [**wireshark**].

Our recorded cover data REC_{Cover} capture from this setup is roughly 25 minutes long and includes 4.961 relevant OPC UA packets (9.922 packets in total). In a potential attack scenario, the server is corrupted via a supply-chain attack and responds to

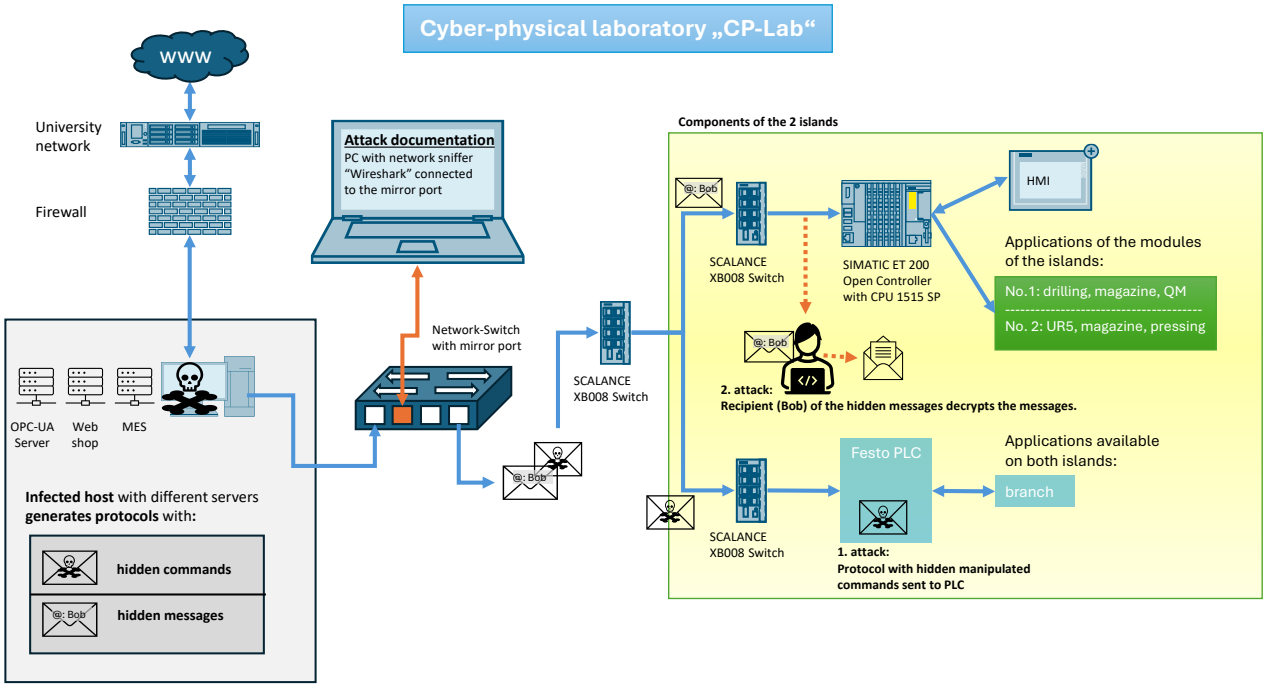

Figure 2. CP-Lab Setup for the recording of the uncompromised cover data. The communication path in the attack scenario used for evaluation is highlighted.

specific requests with timing delays, which embeds the hidden-malicious-dummy. An accomplice can decode the embedded message by accessing the mirror port of the switch, see Figure 2. For our evaluation, we duplicate REC_{Cover} to obtain the exact same data into which we embed our hidden non-malicous dummies. We embed synthetically created non-malicious data into the recorded data with the SSE-concept, introduced in [14], for a fast and easy embedding near real time afterwards, without risking a corruption of the deployed hardware. We embed the following exemplary dummy message: "*Set Valv1 390 Sleep 5 Set Valv1 400 Sleep 5 Set Valv1 405 Set Valv2 200 Sleep 15 Set Valv1 390 Sleep*" representing **Scenario$_5$: Control Commands & Sensor readings**. The resulting recording with included steganographic hidden-non-malicious-dummies is called REC_{Stego} and has the same number of packets as REC_{Cover}. The next subsection describes the steganographic embedding method used to inject the non-malicious data into the network data.

C. Steganographic Embedding Method for use in our scenario

To embed the hidden non-malicious-dummy into the network data, a state-of-the-art steganographic embedding method from [18] is used. This method utilizes network packet timestamps to embed hidden information. In this work, we use the protocol-specific OPC UA timestamp for an embedding. In the initial method from [18], the microsecond digits of a timestamp were altered (Example: $T_i = 08:00:00.123\mathbf{456}789$) to embed the hidden information. In this work, we have to adjust the approach due to unavailability in our setup's recording. We use the the digits in the millisecond range for embedding (Example: $T_i = 09:00:00.\mathbf{123}000000$), because they represent the three least significant values in the timestamp. The embedding methods embeds a bitstream into the data, which can be converted afterwards into an ASCII-message. For the embedding three consecutive OPC UA server timestamps (read requests) are modified (T_i , T_{i+1} , T_{i+2}). For timestamp T_i the first millisecond digit position ms_1 is modified, for T_{i+1} the second millisecond digit ms_2 and for T_{i+2} the third one ms_3. The following three timestamps of this component stay untouched (T_{i+3} , T_{i+4}, T_{i+5}) to ensure more unobtrusiveness. The approach uses the digit '4' to embed bit = 0 and digit '9' to embed bit=1. This means in three consecutive timestamps the following digit positions are altered into '4' or '9' to embed bit = 0 or bit = 1: $T_i = ms_1$, $T_{i+1} = ms_2$ and $T_{i+2} = ms_3$. For more details, see [18].

D. Detection Approach and resulting Data Set for Evaluation

For the detection of the hidden non-malicious-dummies, we build a logistic model tree classifier with WEKA 3.8 [19], which uses a handcrafted feature space from [15]. The feature space performs a frequency analysis of occurrence for the digits *0* to *9* on selected digit positions and a selected number of packets. In this work, we perform the frequency analysis on the OPC UA timestamps of the server packets on the millisecond digit positions ms_1, ms_2 and ms_3. Thus, we determine 10 features for each of the three digit positions with values ranging from 0.0 to 1.0. This results in a 30-dimensional feature space with the addition of a label for each vector, i.e., sample ('cover' or 'stego'). With this feature extractor, we iterate through relevant OPC UA server packets in both data sets REC_{Cover}

and REC_{Stego}. We extract a feature vector after analyzing 20 relevant packets. As introduced in Section IV-B, we have 4.961 relevant packets per recording, which results in 248 samples for both recordings REC_{Cover} and REC_{Stego}. In Section V, the evaluation results are presented.

V. Evaluation Results

As introduced in Section IV-A, we perform a 5-fold cross validation with our selected detection approach (Section IV-D) to determine if the approach is able to distinguish between cover data samples (unaltered) and steganographic data samples with embedded hidden non-malicious-dummies. The performance of the approach is measured with TPR, TNR, FMR and FAR. The resulting rates are visualized in Figure 3. The classification results of the approach are presented in Table I.

TABLE I. Confusion Matrix of 5-fold cross validation

Classification Results of Detection Approach		
classified as $->$ actual (248)	REC_{Cover}	REC_{Stego}
REC_{Cover}	**214**	34
REC_{Stego}	26	**222**

The approach is able to detect 222 out of 248 samples with embedded hidden non-malicious-dummies. This results in TPR = 0.895 and FMR = 0.105. Additionally, the detection approach classifies 214 of 248 unaltered cover data samples correctly, this leads to TNR = 0.862 and FAR = 0.138. Overall the approach has an accuracy of 0.879, derived from 436 of 496 correctly classified samples. For an initial detection approach the performance is decent but should be improved in future work, e.g., with a novel feature space. Especially, a better FAR would be critical for a real world application.

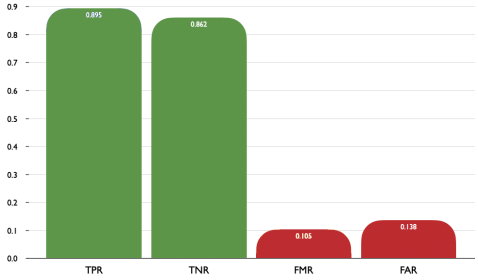

Figure 3. TPR, TNR, FMR and FAR of detection approach determined with 5-fold cross validation

VI. Conclusion and Future Work

This paper serves as starting point for the creation of a comprehensive definition of dummies that can be used to test and simulate attacks in ICS networks with the goal to improve detection and reaction of this threat.

However, it must be ensured that these dummies must not have any harmful effect on networks to prevent possible misuse. For this, it is necessary to establish a corresponding standard in future work. Such a standard should specify which requirements hidden-non-malicious-dummies must meet, as well as a systematic classification of such dummies, the definition and agreement of suitable protection mechanisms, the definition of areas of application and the identification of relevant user groups. In addition, potential extensions to the dummy definition that have not yet been addressed in this article must be considered.

The requirements formulated in this work have shown that the design of the dummies depends largely on their specific properties and purposes. These properties can differ significantly from one another, which means that the respective dummies also pose different potential hazards. Therefore it seems plausible to define different hazard classes for dummies.

The classification of the messages in hidden communication used by attackers can also support the attribution of real attacks against ICS. The motivation of some of proposed communication scenarios comes from the analysis of attack scenarios occurring in theoretical considerations as well as practical cases investigated during the work in the project ATTRIBUT [20]. For this project, measures to identify the Communication Scenario pursued by an attacker are an aspect of future work.

Marking of the hidden-non-malicious-dummies

One of the most important focal points of future work will be the inclusion of suitable protection mechanisms. Protection mechanisms have to be defined in close coordination with existing standards for security information and event management (SIEM) systems - both in the context of ICS and for enterprise/business IT. This is necessary to the risk of misuse or misappropriation of the dummies.

Any individual using so-called hidden-non-malicious-dummies, or test dummies in general, must commit - provided they act without malicious intent - to explicitly marking the protocols they use as dummies. While this approach may initially seem contradictory, particularly in the context of hidden channel attacks, it should be understood as a preventive security measure intended to mitigate the risk of such dummies being misused for offensive purposes.

For testing purposes, monitoring systems can be configured to ignore specific markers or tags. This enables the evaluation of relevant attack characteristics and their potential impact - as well as the detectability of hidden-non-malicious-dummies - without compromising the integrity of the testing environment.

Various protocol-marking methods are already established in enterprise IT. One such method is tagging, as implemented in tagged VLANs according to IEEE 802.1Q, where additional fields are inserted into the Ethernet protocol's data section to carry VLAN-specific tags. Another approach is labeling, such as through Quality of Service (QoS) labels, which allow certain protocols to be prioritized in network traffic. These mechanisms could also be applied to hidden-non-malicious-dummies by defining standardized procedures to identify and distinguish misused dummies within network environments.

Furthermore, protocol-level marking—such as IP header marking—offers an additional option. For instance, the Type of Service (ToS) field in the IP header could be extended to

introduce, define, and standardize a new service type labeled "DUMMY," thereby enabling a consistent and identifiable classification of such dummy traffic.

The range of possible measures to prevent misuse is significantly broader than those outlined in this context. There remains a considerable need for continued research and discourse on how, and to what extent, protective mechanisms can, should, and must be implemented in practice.

At the same time, it is important to acknowledge that absolute protection against misuse can never be fully guaranteed. Nevertheless, the residual risk can be substantially reduced through the conscious selection and implementation of appropriate safeguards—provided that researchers remain aware of their ethical responsibilities and adhere to the principles of responsible cyber-security research.

ACKNOWLEDGMENT

This work has been performed in the scope of the project "SYNTHESIS - Synthetically 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 motivation for the Communication Scenarios originates from considerations for the project ATTRIBUT which 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.

Author shares: All authors: fundamental discussion. RA: formalization of non-malicious dummies and injection methods. TN & LB: conception of the detection approach, implementation of the embedding code, injecting of the non-malicious dummies & experimental evaluation. TN & CV: General conception of SSE, embedding scheme & the evaluation methodology. ML: Creation on recordings; considerations on marking of resulting test sets.

REFERENCES

[1] Dragos, Inc., "2025 OT/ICS Cybersecurity Report", 2025, Accessed: Sep. 15, 2025. [Online]. Available: https://hub.dragos.com/hubfs/312-Year-in-Review/2025/Dragos-2025-OT-Cybersecurity-Report-A-Year-in-Review.pdf?hsLang=en.

[2] MITRE, "MITRE ATT&CK: Techniques - Data Obfuscation: Steganography", Apr. 2025, Accessed: Sep. 15, 2025. [Online]. Available: https://attack.mitre.org/techniques/T1001/002/.

[3] A. Badaev and K.Naumova, "Steganoamor campaign: Ta558 mass-attacking companies and public institutions all around the world", *https://www.ptsecurity.com/ww-en/analytics/pt-esc-threat-intelligence/steganoamor-campaign-ta558-mass-attacking-companies-and-public-institutions-all-around-the-world/*, 2024.

[4] T. Williams, "An overview of pera and the purdue methodology", *https://link.springer.com/content/pdf/10.1007/978-0-387-34941-1_8.pdf*, 1996.

[5] Rockwell Automation, "Converged plantwide ethernet (cpwe) design and implementation guide", *https://literature.rockwellautomation.com/idc/groups/literature/documents/td/enet-td001_-en-p.pdf*, 2011.

[6] OPC-Foundation, "Unified architecture", 2008, Accessed: Sep. 15, 2025. [Online]. Available: https://opcfoundation.org/about/opc-technologies/OPC%20UA/.

[7] S. Wendzel et al., "A generic taxonomy for steganography methods", *Association for Computing Machinery; ACM 1557-7341/2025/4-ART*, Jul. 2025. DOI: https://doi.org/10.1145/3729165.

[8] T. Neubert, E. Schueler, H.Ullrich, L. Buxhoidt, and C. Vielhauer, "Extended analysis, detection and attribution of steganographic embedding methods in network data of industrial controls systems", *International Journal on Advances in Security, ISSN:1942-2636 online: https://www.thinkmind.org/library/Sec/Sec_v18_n12_2025/sec_v18_n12_2025_10.html*, 2025.

[9] D. Kushner, "The real story of stuxnet", *https://spectrum.ieee.org/the-real-story-of-stuxnet, last access: 19/09/2024*, 2013.

[10] R. M. Lee, M. J. Assante, and T. Conway, "Analysis of the cyber attack on the ukrainian power grid", SANS Institute, Tech. Rep., 2016.

[11] Dragos, Inc., "Assessment of reported malware infection at nuclear facility", 2019, Accessed: Sep. 15, 2025. [Online]. Available: https://www.dragos.com/blog/industry-news/assessment-of-reported-malware-infection-at-nuclear-facility/.

[12] T. Neubert and C. Vielhauer, "Kill chain attack modeling for hidden channel attack scenarios in industrial control systems", *21st IFAC World Congress, Berlin, Germany, July 11-17, Submission 1475*, 2020.

[13] Lockheed Martin, "Https://www.lockheedmartin.com/en-us/capabilities/cyber/cyber-kill-chain.html", Apr. 2025, Accessed: Sep. 15, 2025. [Online]. Available: https://www.lockheedmartin.com/en-us/capabilities/cyber/cyber-kill-chain.html.

[14] T. Neubert, B. Peuker, L. Buxhoidt, E. Schueler, and C. Vielhauer, "Synthetic embedding of hidden information in industrial control system network protocols for evaluation of steganographic malware", *Tech. Report, arXiv, https://doi.org/10.48550/arXiv.2406.19338*, 2024.

[15] T. Neubert, A. J. C. Morcillo, and C. Vielhauer, "Improving performance of machine learning based detection of network steganography in industrial control systems.", *In the Proceedings of 17th International Conference on Availability, Reliability and Security (ARES 2022), Article No.: 51, pp. 1 - 8, August 23– 26, 2022, Vienna, Austria. ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3538969.3544427*, 2022.

[16] M. Massimo Guarascio, M. Marco Zuppelli, N. Nunzio Cassavia, G. Manco, and L. Caviglione, "Detection of network covert channels in iot ecosystems using machine learning", *In Proceedings of Italian Conference on Cybersecurity, Rome, Italy, https://api.semanticscholar.org/CorpusID:253270269*, 2021.

[17] L. Caviglione, "Trends and challenges in network covert channels countermeasures", *DOI: 10.3390/app11041641*, 2021.

[18] T. Neubert, C. Kraetzer, and C. Vielhauer, "Artificial steganographic network data generation concept and evaluation of detection approaches to secure industrial control systems against steganographic attacks", *In The 16th International Conference on Availability, Reliability and Security (ARES 2021), August 17–20, 2021, Vienna, Austria. ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3465481.3470073*, 2021.

[19] M. Hall, "The weka data mining software: An update.", *In SIGKDD Explorations*, 2009.

[20] ATTRIBUT, "Project ATTRIBUT", Accessed: Sep. 15, 2025. [Online]. Available: https://omen.cs.uni-magdeburg.de/itiamsl/english/attribut/attribut.html.