SENAMI: Selective Non-Invasive Active Monitoring for ICS Intrusion Detection

William Jardine
MWR InfoSecurity
UK
william.jardine
@mwrinfosecurity.com

Sylvain Frey
Security Lancaster
Lancaster University, UK
s.frey@lancaster.ac.uk

Benjamin Green
Security Lancaster
Lancaster University, UK
b.green2@lancaster.ac.uk

Awais Rashid
Security Lancaster
Lancaster University, UK
a.rashid@lancaster.ac.uk

ABSTRACT

Current intrusion detection systems (IDS) for industrial control systems (ICS) mostly involve the retrofitting of conventional network IDSs, such as SNORT. Such an approach is prone to missing highly targeted and specific attacks against ICS. Where ICS-specific approaches exist, they often rely on passive network monitoring techniques, offering a low cost solution, and avoiding any computational overhead arising from actively polling ICS devices. However, the use of passive approaches alone could fail in the detection of attacks that alter the behaviour of ICS devices (as was the case in Stuxnet). Where active solutions exist, they can be resource-intensive, posing the risk of overloading legacy devices which are commonplace in ICSs. In this paper we aim to overcome these challenges through the combination of a passive network monitoring approach, and selective active monitoring based on attack vectors specific to an ICS context. We present the implementation of our IDS, SENAMI, for use with Siemens S7 devices. We evaluate the effectiveness of SENAMI in a comprehensive testbed environment, demonstrating validity of the proposed approach through the detection of purely passive attacks at a rate of 99%, and active value tampering attacks at a rate of 81-93%. Crucially, we reach recall values greater than 0.96, indicating few attack scenarios generating false negatives.

Keywords

Industrial Control Systems; Intrusion Detection Systems; Active Monitoring

1. INTRODUCTION

Industrial Control Systems (ICS) are networks of devices responsible for the monitoring, control, and automation of a physical process. Applied across a range of sectors, from water treatment to energy generation and oil refinery, some ICSs are considered part of national critical infrastructure. Historically, ICS were deployed on networks isolated from the Internet, however, more and more ICS are now found on public IP addresses, as demonstrated by various reports such as project SHINE [13]. Systems that are not directly accessible are also under threat, as shown by a range of high-profile attacks such as Stuxnet, the German Steel Mill disaster, and the recent attack on the Ukrainian power grid [8, 15, 21].

The increasing severity and likelihood of network-based attacks has resulted in the deployment of Intrusion Detection Systems (IDS) across most large-scale IT infrastructures [5]. However, within the context of ICS, we see a lack of bespoke IDS offerings. A number of commercial offerings involve retrofitting existing network intrusion detection systems (NIDS), e.g., SNORT, to ICS settings [1]. While this approach has its benefits, sophisticated attacks often exploit specific characteristics and patterns of operation within ICS environments, that do not necessarily manifest as network anomalies [12, 16].

A number of works have attempted to address this limitation through the design of ICS-specific IDSs. For example, [9, 12, 22] are based on passive monitoring of network traffic to and from PLCs. Passive monitoring techniques induce minimal overhead and require little adaptation of the monitored networks. However, targeted attacks that covertly alter the behaviour of a compromised PLC are hard to detect using passive monitoring only, as they leave little to no traces on the network. On the other hand, active monitoring of PLC variables allows us to detect tampering even if the attack is not visible on the network [2, 19]. However, this comes at the cost of additional polling on the target PLC, which can become computationally expensive on resource-constrained legacy systems.

In this paper, we investigate a hybrid IDS exploiting the best of both worlds. We demonstrate that a number of advanced ICS-specific attacks can be detected through a largely passive solution, with additional selective active monitoring of highly contextualised PLC variables. We refer to this active monitoring as *selective, non-invasive, active monitoring*, where polling of unnecessarily high number of or

CPS-SPC'16, October 28 2016, Vienna, Austria

ISBN 978-1-4503-4568-2/16/10. . . $15.00

DOI: http://dx.doi.org/10.1145/2994487.2994496

contextually redundant values is avoided. We demonstrate the implementation of our approach in the form of SENAMI: a selective, non-invasive, active monitoring IDS for systems based on Siemens S7 devices. We evaluate the effectiveness of SENAMI in Lancaster's comprehensive ICS testbed [10, 20]. Our evaluation demonstrates that SENAMI is able to detect purely passive attacks at a rate of 99%, and active value tampering attacks at a rate of 81-93%. Crucially, we reach recall values greater than 0.96, indicating few attack scenarios generating false negatives.

The novel contributions of this paper are as follows:

- We introduce the notion of selective, non-invasive, active monitoring as a means for detection of targeted attacks aimed at covertly altering the behaviour of PLCs, without introducing noticeable computational overhead and while controlling the network overhead.
- We provide a practical implementation of this concept in SENAMI, an IDS for Siemens S7 control systems.
- We demonstrate the effectiveness of such an approach through evaluation of SENAMI against a range of attack scenarios in a realistic water treatment testbed.

The remainder of this paper is structured as follows. Section 2 discusses existing research on intrusion detection within the context of ICS, highlighting the need for selective, non-invasive, active monitoring. Section 3 discusses four key attack scenarios, motivating the use of our proposed approach. Section 4 presents the implementation of SENAMI. Section 5 describes the results of evaluating SENAMI against a range of attacks in our ICS testbed. Section 6 reflects on the results of the evaluation, discussing the strengths and weaknesses of selective, non-invasive, active monitoring, and limitations of SENAMI's implementation. Section 7 concludes the paper and identifies directions for future work.

2. BACKGROUND AND RELATED WORK

Mahan et al. [16] discuss the pros and cons of adapting existing NIDS vs. crafting ICS-specific intrusion detection systems, highlighting the need for the latter. McLaughlin et al. [17] introduce the notion of *security enclaves*, groupings of systems with common security policies, as a basis for cost-effectively combining a range of existing general purpose solutions. They highlight the lack of specificity as a key problem of such a general purpose reuse approach.

Gonzalez and Papa [9] propose an IDS for passive monitoring of Modbus. Intrusions are detected through the inspection of captured Modbus packets, and maintaining a table of states for the PLCs and other devices monitored by the system. This approach is based on the often fixed and predictable patterns of behaviour within ICS networks.

Strohmeier et al. [22] discuss passive intrusion detection for unmanned planes. They describe a method of measuring the Received Signal Strength (RSS) of messages, as an airplane's signal will vary during flight, while an attacker is likely to remain stationary, and therefore have a constant RSS. This approach applies machine learning techniques to the characterisation of normal level variation.

Caselli et al. [4] discuss the limitations of existing NIDS to detect semantic attacks similar to Stuxnet. They highlight a specific attack based on legitimate messages that may be sent out of normal sequence or at malicious times and would nevertheless be accepted as valid event sequences by the system and existing NIDS. They go on to propose an IDS to support sequence-aware intrusion detection (S-IDS).

McLaughlin and McDaniel [18] discuss SABOT, a tool that generates semi-automated attack vectors for PLCs without knowledge of the PLC's internals. The attacker provides a high-level description of the control system's behaviour, including declaration of plant devices, and a set of properties describing the behaviour of each. SABOT then downloads (in ICS parlance *uploads*[1]) and decompiles the PLC's logic bytecode and seeks to use the aforementioned descriptions to generate a mapping between the identified variables and the (attacker-defined) devices they control (e.g., pump, valve, etc.). This provides enough information about the internal state of the PLC so that malicious code attacks can be launched against those PLC-internal variables identified as weak.

The aforementioned passive solutions can be used to spot generally suspicious behaviour as well as more specific attack vectors, like uploads of PLC logic code. However, such approaches cannot readily detect attacks aimed at altering the behaviour of PLCs. Nicholson et al. [19] propose an active monitoring approach to detect attacks by monitoring the values of a PLC's internal variables. This provides an effective PLC-based detection of attacks, albeit at the expense of monitoring a large number of internal variables. The core concern with this approach relates to the age of devices and computational resource available in ICS environments. Devices can be decades old, running protocols designed as early as the 1970s. Therefore, an intensive monitoring approach can be computationally expensive and risks overloading legacy or resource-constrained devices [2].

Our approach aims to leverage the lack of computational overhead seen through passive traffic analysis, while enriching it with selective, active monitoring of a few highly contextualised PLC variables. This selective, non-invasive, active monitoring is aimed at honing the specificity of intrusion detection, while reducing the computational overhead on devices, hence eliminating the need for scheduling [2].

3. ATTACK SCENARIOS

In this work, we focus on the detection of attacks involving an individual PLC in a local network, as depicted in Figure 1. We assume that an attacker has already entered the network (preventing such network intrusions is out of the scope of this paper). We do not consider the case where the attacker would have physical access to the PLC: such a situation opens up a whole new range of attacks including firmware compromise and physical tampering, which are out of our scope. For similar reasons, we also do not take insider threats into account. We consider four scenarios, and for each of these, we discuss the theoretical applicability of passive and active monitoring techniques.

3.1 Scenario 1: Reconnaissance

An attacker getting a foothold in a network will likely perform reconnaissance scans to identify potential targets, using, for instance, plcscan [6]. Although such scans are generally harmless, detecting them – through passive monitor-

[1] In ICS terminology, a download of logic code is referred to as an upload. We will follow this convention from here onwards.

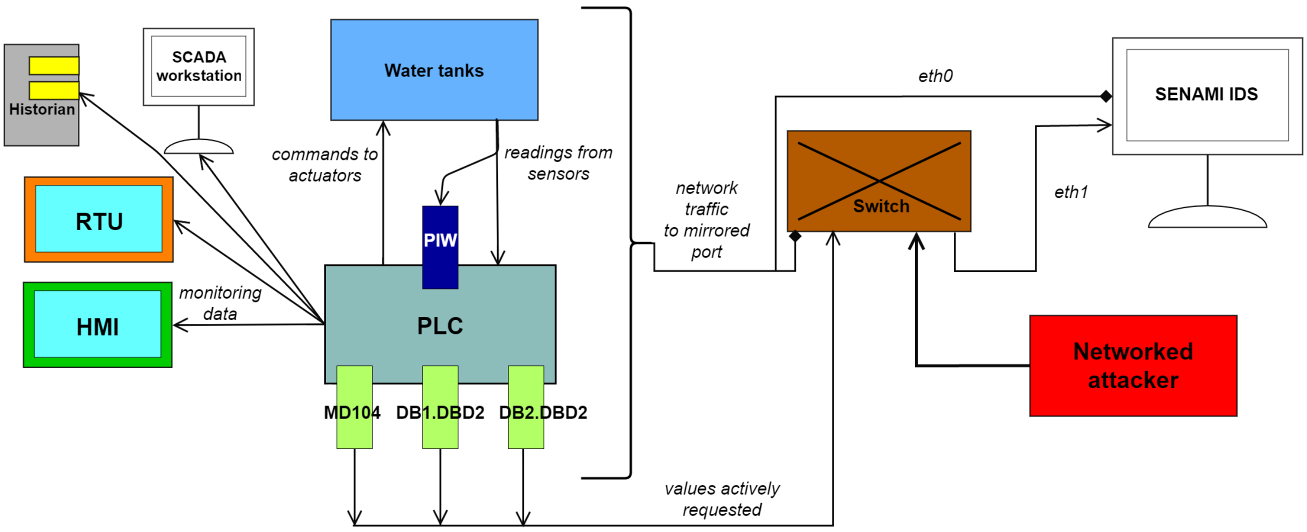

Figure 1: The infrastructure under attack.

ing of network activity – is key to identifying compromised assets and stopping the attacker's lateral movement in the network.

3.2 Scenario 2: Denial of Service

Denial of Service (DoS) attacks comprise flooding a target with traffic. In this paper, we also consider a variant attack using a fuzzer [3]. Interestingly, DoS techniques can also be used as a decoy to mask an attacker's true intentions. An attacker who can generate large amounts of traffic into the ICS network, for instance, via a compromised machine, can then generate endless logs that would dissimulate behaviour that might otherwise seem suspicious (e.g., writing specific PLC values to undermine processes, or reporting back system states, cf. scenarios 3 and 4).

DoS attacks themselves are trivial to detect via passive monitoring due to the amount of traffic they generate. Even in the presence of a decoy DoS, a network-based attack – cf. scenarios 1, 3 and 4.1 – would still generate noise in the network. Specifically, if the IDS had knowledge of the times at which legitimate packets usually arrive, even a small number of packets from a legitimate IP at anomalous times can be detected. Arguably, such a task would be difficult to perform manually under a decoy DoS.

3.3 Scenario 3: Exfiltration of logic code

Uploading logic code from a PLC to an engineering workstation is a legitimate action. ICS engineers will often upload control logic to their workstations when making alterations to process operations. These alterations can be minor (e.g., altering control timers), or substantial (e.g., modifying critical process thresholds). As highlighted by [18], control logic can be utilised by an attacker for a variety of reasons, including general reconnaissance and malware generation.

Irrespective of the attacker's purpose, uploading control logic from a PLC will generate network traffic, and a passive IDS should be able to detect this. With proper heuristic comparison of the expected quantity, IP addresses, and timing of this traffic, it should be possible to detect whether this action is expected or not.

3.4 Scenario 4: Tampering with monitoring

An attacker may choose to tamper with the values being returned to devices responsible for remote alarm management or data analytics, such as a Remote Terminal Unit (RTU) and a Historian database, as shown in Figure 1. This can blind or alter an operator's perception of physical process states, hiding malicious activity, potentially leading to catastrophic failures [7]. For example, an attacker may alter the PLC's behaviour causing physical process change beyond defined safe limits, while simultaneously adjusting values returned to the RTU and Historian, blinding remote operators to such changes. Tampering of this nature was applied in the case of Stuxnet.

We consider three forms of tampering with the values stored inside the PLC (scenarios where the attacker would tamper with the logic of the PLC are out of scope):

- **Scenario 4.1**: via the network, the attacker overrides monitoring values through write packets that can be detected by passive monitoring. The write packets would appear to come from a suspicious source (i.e. an IP that is not supposed to issue such packets) or at a suspicious time.

- **Scenario 4.2**: from a compromised PLC itself, its logic overriding the monitoring values – in which case, only active monitoring would be applicable, since no network traffic is generated.

- **Scenario 4.3**: from a compromised PLC again, but for very brief periods of time ("*partial tampering*"). This reduces the effectiveness of the attack, due to the refresh rate of the Data Block values. However, if carried out persistently, it can cause damage to the process underway and to the PLC itself, while being more difficult to detect via active monitoring.

4. SENAMI: SELECTIVE, NON-INVASIVE, ACTIVE MONITORING IDS

An overview of SENAMI is shown in Figure 2. As shown in the figure, SENAMI focuses on the Siemens S7 protocol and the Siemens SIMATIC S7 series PLCs. The reasons behind our focus on the S7 protocol are twofold. Firstly, our choice was driven by the real-world testbed environment described in [10, 20], configured to replicate PLC logic practices we have previously observed in real-world ICS networks. Secondly, Siemens S7 PLCs were targeted by Stuxnet [15], providing a realistic attack scenario to replicate with regards to scenario 4 in Section 3.

It is worth noting there are two distinct versions of the S7 protocol, applied across varying SIMATIC S7 PLCs. SENAMI is concerned with the standard legacy version, used in the majority of S7 control systems. This version of the S7 protocol is identified by the protocol ID "0x32" in the 62nd byte of the S7 packet [14]. The work described in this paper is, therefore, transferable across all Siemens PLCs using this same S7 protocol version. The principle of selective, non-invasive, active monitoring is applicable to the general field of ICS security. However, specific implementations will be required to support additional protocols.

The following sections discuss SENAMI's implementation of passive and active monitoring in more detail. One point which should be noted is SENAMI's ability to only monitor one PLC at a time, discarding packets where neither the source nor destination address matches that of the PLC. To monitor a wider network, multiple instance of SENAMI could be launched, with relevant PLC addressing applied to each instance. Such a PLC-centric approach makes it possible for SENAMI to scale to a complex ICS environment. For instance, in an environment with 6 PLC stations, 6 instances of SENAMI can be run, each fine-tuned to the specific behaviour of each station.

4.1 Packet capturing and parsing

The passive monitoring components of SENAMI require a means by which capturing and parsing of S7 packets can be achieved. The S7 protocol (much like other popular ICS protocols) operates over TCP, specifically over port 102. Within the TCP packet, the S7 packet is further encapsulated inside TPKT and COTP packets, with the S7 packet being given in the COTP PDU [14].

S7 is a request-response protocol, where requests are made to the PLC, and responses are served back, indicated by the ROSCTR field in the S7 header. There are multiple function codes specifying the purpose of these request or response packets. The function codes SENAMI is concerned with monitoring are Read, Write, and the logic code uploads (Start Upload, Upload, and End Upload). A packet will also list a series of *parameters*, areas of memory from which it is either requesting values, or to which it is writing values. As discussed in Section 4.2, SENAMI's passive component is concerned with various properties of these packets.

Non-S7 packets also need to be considered. For example, TCP Denial of Service attacks could be launched against the PLC's network interface. Given that S7 packets will account for the majority of traffic within the scope of SENAMI, for simplification, we do not parse such non-S7 packets on a protocol-by-protocol basis. Instead, SENAMI classifies all non-S7 packets as one type {*"Other"*} to be monitored.

These parsed and processed packets are passed onto the heuristic (passive) analysis component of SENAMI.

4.2 Passive network traffic analysis

SENAMI utilises a set of heuristics to characterise a normal baseline for traffic within the ICS network. These heuristics are derived from a representative period of network traffic. Note that, while powerful passive NIDS exist, our focus here is on highlighting the general shortcomings of a NIDS for ICS, particularly with regards to more advanced attacks and when considering more general-purpose IDSs. Therefore, we incorporate a generalised passive IDS within SENAMI in order to demonstrate the effectiveness of a combined active and passive approach. As shown in Figure 2, there are four types of heuristic analyses:

- Quantity anomaly detection
- Time anomaly detection
- IP anomaly detection
- PLC logic code upload detection

The baseline for these heuristics is established by first running a packet capture over a representative segment of network traffic (for instance, 2 hours). The capture generates a file of aggregated data: IPs, packet count, time of arrival, etc. sorted by function code. This data is then analysed by an expert user (i.e. an ICS operator, preferably one with some familiarity with how the IDS works) who uses it to fill a reference configuration file for the IDS to use as a baseline.

4.2.1 *Quantity anomaly detection*

The set of identified heuristics are concerned with the number of packets that arrive in a given time period between 2 points, for example, the number of S7 packets with the Write function code that arrive between the PLC and a particular workstation in 30 seconds.

With sufficient network traffic on which to define heuristic baselines, SENAMI can establish a reasonably robust estimate of which quantity represents a normal traffic for each packet type, and which thresholds define low and high traffics. The module then tags each time period with the corresponding traffic levels: Normal, Low or High.

Aggregating the traffic over set intervals, rather than marking individual packets, helps circumvent the potential problem of unmanageably large logs. This would, for example, help avoid attack scenario 2 in Section 3, whereby an attacker launches a DoS attack to mask a more targeted attack.

4.2.2 *Time anomaly detection*

One common feature of certain ICS processes is that they can occur at relatively fixed, predictable times. For example, in the water industry there may be an otherwise unusually high number of Write packets turning on pumps every day at roughly 9pm, accounting for increased demand within the supplied area. If, however, a similar increase was observed at 4am, it could be seen as unusual. This extra heuristic analysis allows more nuance in the marking of quantity; it allows the system to decide if a given quantity marking is in itself usual or unusual for the time period at which it is occurring. This heuristic is, therefore, applied to a certain time interval (e.g., the interval 20:30 might encapsulate all

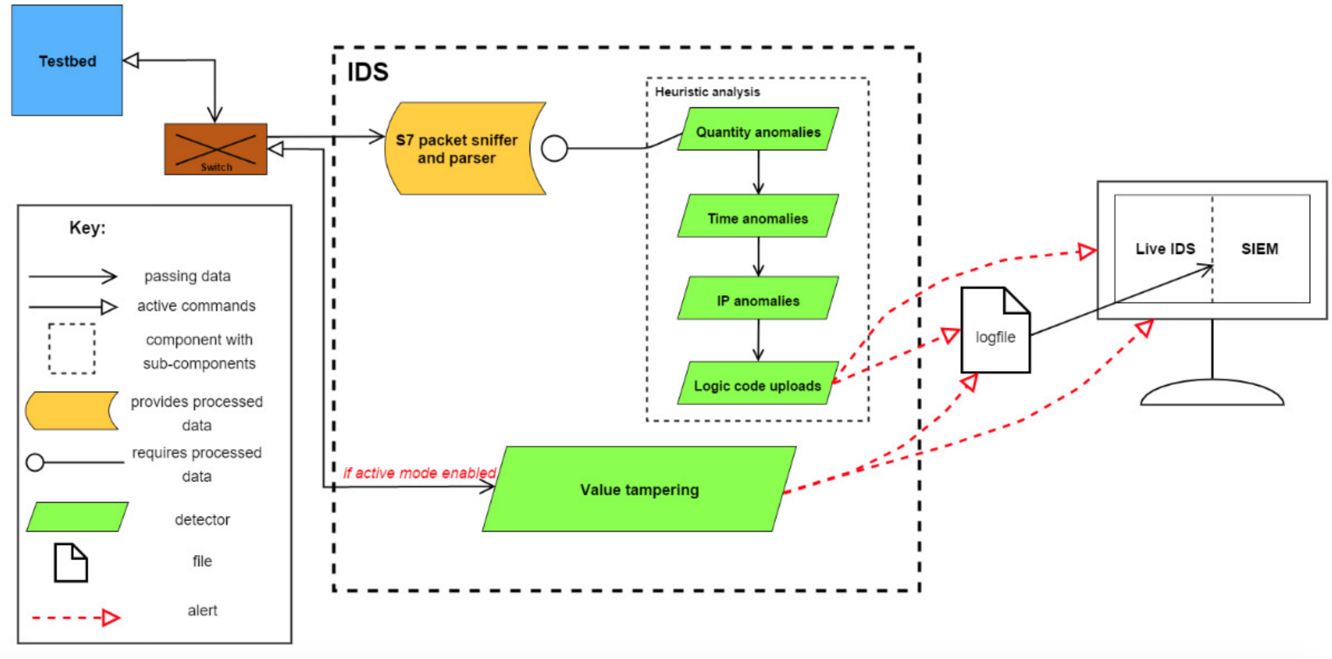

Figure 2: Overview of SENAMI

times between 20:30 and 20:39), as opposed to the exact time at which a packet occurs.

The aggregated packet information from the quantity anomaly detection stage (marked as either Low, Normal or High quantity) is then compared against heuristics for time. The system can allocate a suspicion level, based around whether this quantity is usual for the given time interval.

In our current implementation of SENAMI, a more basic proof-of-concept version of this idea is given. For the sake of testing the solution, it is not feasible to observe time anomalies on a 24 hour cycle. The system, therefore, looks at 5 minute intervals within an hour (00, 05, 10, etc.) and decides whether this behaviour is normal for that time. This is done per packet type and per IP address endpoint.

4.2.3 IP anomaly detection

As previously noted, each instance of SENAMI is only concerned with communications to and from the individual PLC it monitors, as opposed to communications between all devices on the network. Similarly to how there are time periods during which we expect certain packets, and their associated quantities, there exists a set of IP addresses from which the PLC expects packets. In an ICS environment, communicating components are largely always on; components are unlikely to change IP addresses. As such, communication from a device that the system has not previously observed would be considered suspicious. Likewise, communication from an IP address already on the network, which does not normally Write to a PLC, could indicate a potential attack.

4.2.4 PLC logic code upload detection

Upload of PLC logic code is a potentially dangerous reconnaissance step undertaken by an attacker [18]. However, it also forms a legitimate action that an ICS engineer would carry out. It is, therefore, important to characterise whether or not such uploads are anomalous. For example, if a logic upload comes from an IP address that is not normally associated with this action, it would be considered an alert. This does have the potential for false positives however. If an ICS engineer in charge of maintaining PLC logic is doing a legitimate upload, but has recently changed their IP address, this would flag an alert. However, this over-sensitivity is necessary to detect potentially covert attack vectors. Furthermore, simple passive heuristics can detect such attacks at their earliest ingress, as the code is uploaded to an attacking node, while remaining computationally inexpensive. This is in contrast with some active monitoring approaches [2], which require the mapping of device memory, allowing for monitoring of changes in logic and values, resulting in high computational expense.

4.3 Active monitoring for value tampering

SENAMI takes a *selective* approach, actively monitoring only three PLC variables in our initial proof of concept. The selected variables have a context-specific purpose that make their monitoring useful to the detection of value tampering. As illustrated in Figure 3, the variables monitored are the 104th Double Word (MD104), the 3rd Double Word of Data Block 1 (DB1), and the 3rd Double Word of Data Block 2 (DB2). MD104 is chosen as a reference point, as it can be reasonably assumed that it is a legitimate value – unless an attacker tampers physically with the PLC, which is out of the scope of this paper. DB1 and DB2 are monitored since these two hold key values that are used by the rest of the infrastructure for observation and control purposes.

4.3.1 Inside the PLC

An analogue input (4-20mA) value is taken from a hardware input card and stored in a Peripheral Input Word (PIW). This raw analogue input value is converted to a more meaningful digital range (0-100), and is stored in MD104.

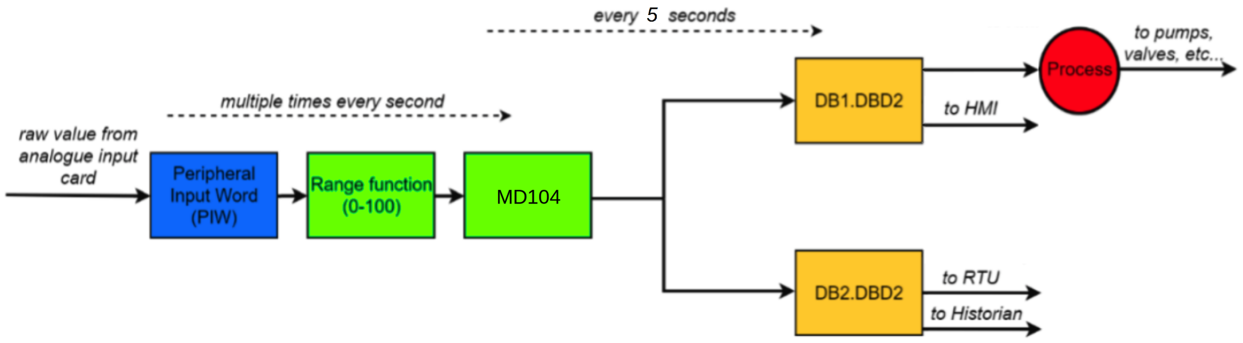

Figure 3: Diagram of internal PLC configuration, where far left represents direct raw input from the PLC's input card and far right indicates executed actions. The dotted arrows represent the rate at which the highlighted section refreshes.

This operation takes place multiple times per second. It is reasonable to assume the integrity of MD104 as the true input value. The value is overwritten too frequently for an attacker to noticeably tamper with it without either (1) getting direct physical access to the PLC and tampering with its firmware or (2) overriding the PLC's logic, which are two attack scenarios that we consider out of scope for this study.

The MD104 value is moved into both DB1 and DB2. DB1 is a block of memory used by the PLC to determine what action to take with regards to physical process operations. DB2 provides a single centralised monitoring source, used to inform the local RTU and Historian of current process states. The RTU will analyse this data, and where necessary, relay critical alarms to remote monitoring centres. Operators will interpret generated alarms and respond accordingly.

The value from MD104 is copied into DB1 and DB2 every 5 seconds. This delay is applied due to the number of processes managed by the PLC. As a quicker refresh rate is not required, resources can be reserved for additional functions, avoiding any computational overload. However, the refresh of values is now suitably slow for an attacker to achieve successful manipulation.

This separation of MD104 into two distinct locations in DB1 and DB2 presents a vulnerability, as an attacker can overwrite DB1 with a malicious Write function to, for example, fill a water tank beyond safe limits, while overwriting DB2 with a value indicating that the water level is normal. The remote operator would then see a falsified value, and be unaware of the real process state.

This practice of using two separate values is for a separation of duties, with distinct areas allocated for operational process decision making, and simple monitoring. This can also act as a safety feature. Due to the number of devices accessing operational data, a single misconfigured component inadvertently writing to a DB, rather than reading from it, would only impact monitoring, leaving the critical process unaffected. What may appear a significant security oversight at first glance, is, in actuality, a design choice derived from safety concerns and centralisation of external requests from monitoring devices.

4.3.2 *Selective active monitoring*

The active monitoring component of SENAMI consists of checking the difference in values between MD104 (what should actually be happening), DB1 (what is actually happening) and DB2 (what operators are being told is happening): any difference would reveal a tampering. It is important to note that this detection relies on the assumption that the attacker has not had physical access to the PLC: in case the firmware were compromised, the value in MD104 itself (i.e. our reference point) could be tampered with.

As such, SENAMI can actively request values frequently. To avoid any potential strain on the PLCs, SENAMI can be configured with polling intervals (e.g., every 5 seconds). The active monitoring component of SENAMI has, therefore, a limited disruptive impact on its target by limiting both the number of monitored variables (3 out of several dozens) and the frequency of this monitoring (every few seconds). However, it is also important to note this scenario is based on a case study in [11], monitoring only one critical operational value. Should two operational values require monitoring, we would need to compare six values in total.

For additional completeness, our future work (see section 6.3.3) will seek to include the RAW PIW (see Figure 3) value as part of this comparison.

5. EVALUATION

5.1 Overview of the Testbed

Two existing works provide a comprehensive overview of Lancaster University's ICS testbed[10, 20]. For reference, field site three, as described in [20], provides the platform for experimentation here.

SENAMI's passive element requires visibility of traffic passing to and from the monitored PLC. This is achieved through use of port mirroring on a network switch, responsible for interconnecting all devices within the network. In addition to passive monitoring, SENAMI also requires an appropriate connectivity for its active monitoring component. This is achieved through the use of a secondary connection between SENAMI and a standard port (non-mirrored) on the same network switch (cf. Figure 1).

5.2 Experiments and results

We evaluated the effectiveness of SENAMI with regards to the detection of various ICS attacks. Based on the attack scenarios established in section 3:

- Scenario 1: reconnaissance (passive threat)

- Scenario 2: DoS (passive threat)
- Scenario 3: logic exfiltration (passive threat)
- Scenario 4.1: tampering via the network (active and passive threat)
- Scenario 4.2: tampering via PLC compromise (active threat)
- Scenario 4.3: partial tampering via PLC compromise (active threat)

Specifically, we ran the following experiments:

- Session 1: Passive detection of passive threats (scenarios 1, 2, 3 and 4.1)
- Session 2: Passive detection of combined passive and active threats (scenarios 1, 2, 3, 4.1 and 4.2)
- Session 3: Active detection of active threats (scenarios 4.1 and 4.2)
- Session 4: Active detection of more advanced active threats combined with passive threats (scenarios 2, 4.1 and 4.3)

For each of these experiments, we present the True Negative (TN), True Positive (TP), False Negative (FN) and False Positive (FP) numbers. From these we compute the corresponding Recall or True Positive Rate (R = TPR = $\frac{TP}{TP+FN}$), Precision (P = $\frac{TP}{TP+FP}$), False Positive Rate (FPR = $\frac{FP}{FP+TN}$) and F-Measure (F = $2\frac{R.P}{R+P}$). Table 1 shows the results of all sessions, Figure 4 summarises the Precision, Recall and F-Measure and Figure 5 presents a receiver operating characteristic (ROC) plot showing the TPR against the FPR.

5.2.1 *Passive detection of passive threats*

These experiments included only passive threats (unauthorised writes, suspicious logic uploads, fuzzing, DoS attacks, etc.), as well as legitimate actions mimicking general network activity.

Session 1.1.

The activity carried out in this session included:

- Scenario 3: suspicious and legitimate logic code uploads.
- Scenario 4.1: writes coming either from legitimate or suspicious IPs, at either legitimate or suspicious times.

Session 1.2.

The activity carried out in this session was similar to that of session 1.1, with the inclusion of more non-S7 attack traffic:

- Scenario 1: use of plcscan [6] for reconnaissance.
- Scenarios 2 and 3: suspicious logic uploads while carrying out either a DoS attack or both a DoS and a fuzzing attack.
- Scenario 3: suspicious and legitimate logic code uploads.
- Scenario 4.1: tampering via write packets coming either from legitimate or suspicious IPs, at either legitimate or suspicious times.

The False Negative generated here was due to the use of plcscan coming from a legitimate IP address, and being a lightweight enough activity that it did not generate a high quantity of traffic to warrant an alert. Therefore, it went undetected.

5.2.2 *Passive detection of combined passive and active threats*

Next, SENAMI (running in passive mode) is tested against a combination of passive and active threats. These are tested for both attackers with suspicious IPs and more covert ones with legitimate IPs.

Session 2.1.

The activity carried out in this session included:

- Scenario 2: DoS and fuzzing attacks.
- Scenarios 2 and 3: logic code uploads while fuzzing.
- Scenario 4.1: tampering via write packets coming from either legitimate or suspicious IPs, at either legitimate or suspicious times.
- Scenarios 2 and 4.1: value tampering and a DoS attack to attempt to disguise partial value tampering.

Due to the use of suspicious IPs, an alert is generated when carrying out the value tampering. However, it is flagged only for its suspicious IP and high number of Write packets, effectively discarding it as an S7 DoS attack, and missing the true, more potentially dangerous, intentions behind it. Therefore, the alert generated for this should be considered a False Positive. Likewise, partial tampering cannot be reliably detected, and is sometimes flagged as a High alert (as a DoS), sometimes as a Medium alert, and sometimes remains entirely undetected.

Session 2.2.

The activity carried out in this session mirrored that of session 2.1, with the additional introduction of legitimate IP addresses in place of the suspicious ones used for value tampering.

A number of False Negatives arise here from the now somewhat legitimised traffic of an attacker, highlighting a fundamental challenge in that it can be difficult to identify malicious insiders (not in the scope of this paper). This demonstrates the requirement for strict observation. Again, value tampering was often either ignored or misclassified as a DoS-like alert (sometimes even as low an alert level as Medium).

5.2.3 *Active detection of active threats*

Operating in active mode, these evaluation sessions focused on SENEMI's ability to detect active value tampering. Passive attacks are included in the following sessions. However, due to the already established strength of passive detection on passive threats, they are not the core focus.

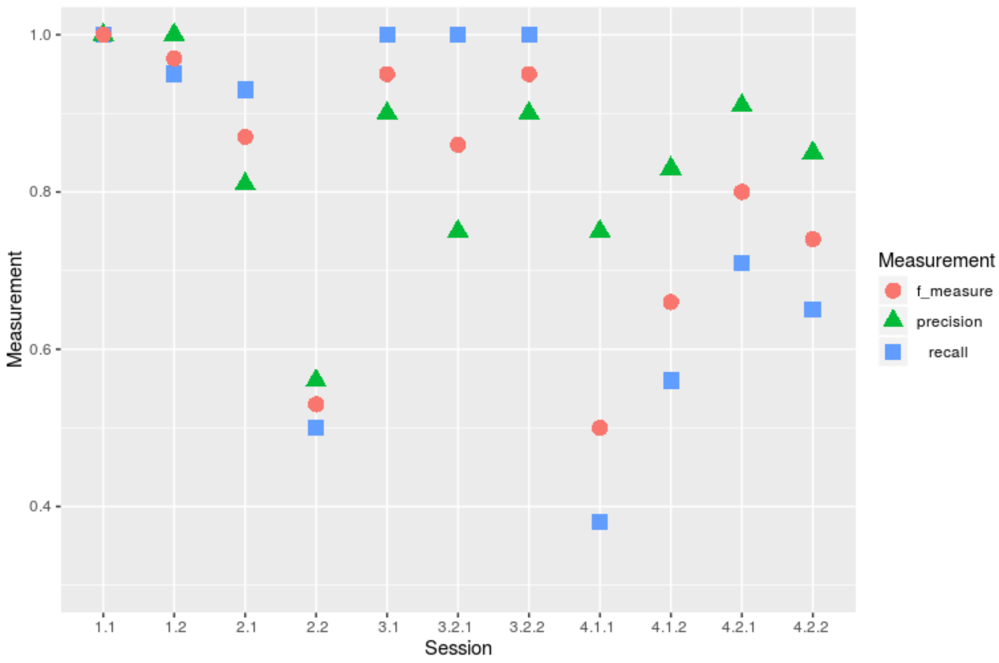

Figure 4: Summary of Recall, Precision and F-Measure for all experiments. Note the truncated Y axis.

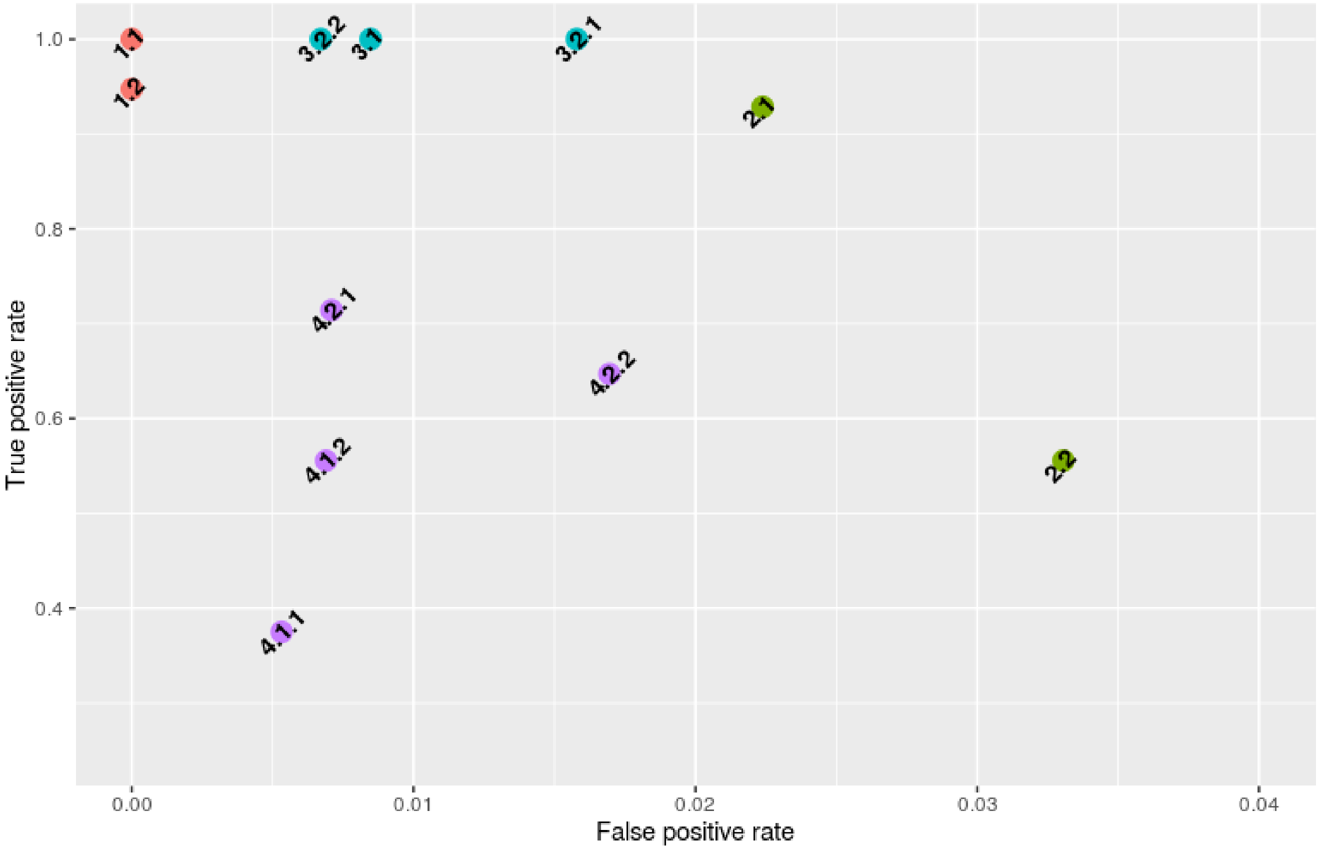

Figure 5: ROC values (TPR against FPR) for all sessions. Note the truncated X and Y axis.

Session 1.1	Predicted Negative	Predicted Positive
Negative Cases	TN = 119	FP = 0
Positive Cases	FN = 0	TP = 9

Session 1.2	Predicted Negative	Predicted Positive
Negative Cases	TN = 217	FP = 0
Positive Cases	FN = 1	TP = 18

Session 2.1	Predicted Negative	Predicted Positive
Negative Cases	TN = 131	FP = 3
Positive Cases	FN = 1	TP = 13

Session 2.2	Predicted Negative	Predicted Positive
Negative Cases	TN = 117	FP = 4
Positive Cases	FN = 5	TP = 5

Session 3.1	Predicted Negative	Predicted Positive
Negative Cases	TN = 117	FP = 1
Positive Cases	FN = 0	TP = 9

Session 3.2.1	Predicted Negative	Predicted Positive
Negative Cases	TN = 187	FP = 3
Positive Cases	FN = 0	TP = 9

Session 3.2.2	Predicted Negative	Predicted Positive
Negative Cases	TN = 148	FP = 1
Positive Cases	FN = 0	TP = 10

Session 4.1.1	Predicted Negative	Predicted Positive
Negative Cases	TN = 187	FP = 1
Positive Cases	FN = 5	TP = 3

Session 4.1.2	Predicted Negative	Predicted Positive
Negative Cases	TN = 144	FP = 1
Positive Cases	FN = 4	TP = 5

Session 4.2.1	Predicted Negative	Predicted Positive
Negative Cases	TN = 140	FP = 1
Positive Cases	FN = 4	TP = 10

Session 4.2.2	Predicted Negative	Predicted Positive
Negative Cases	TN = 116	FP = 3
Positive Cases	FN = 6	TP = 11

Table 1: Session results.

Session 3.1.

The activity carried out in this session included:

- Scenario 4.1: unauthorised writes at an unusual time
- Scenario 4.1: unauthorised writes at a normal time
- Scenario 4.1: multiple instances of value tampering

All of this was carried out from a suspicious IP address.

Session 3.2.

Attackers are likely to compromise, and therefore have access to, a legitimate IP address. The same actions as session 3.1 were carried out, but from legitimate IPs. These tests were carried out multiple times, to demonstrate the changeable – but overall reliable – detection of this active attack from legitimate IP addresses. Values from two of the tests are shown in Table 1 as session 3.2.1 and 3.2.2.

5.2.4 *Active detection of advanced active threats*

The final evaluation scenario introduced less typical, more selective, active threats. Specifically, we evaluated SENAMI against a compromised PLC whose control logic would override its own values (scenario 4.2). In the second session, the attacker combines a decoy DoS attack attempting to mask value tampering activities, while using partial tampering to avoid the timed detection window of SENAMI.

Session 4.1.

This evaluation session included partial tampering (scenario 4.3). Partial tampering is difficult to detect, as demonstrated by the results shown. Due to the unpredictable nature of detection for this attack, the scores of two tests are shown in Table 1.

Session 4.2.

The final evaluation scenario involved detecting a mix of full value tampering (scenario 4.1), and partial value tampering (scenario 4.3), while also under a DoS attack (scenario 2). This test also introduced the idea of using a large scale value tampering attack itself as a DoS, to mask a more selective and lightweight writing attack to a specific area. The results are given as 2 tests shown in Table 1.

The full value tampering attack was reliably detected (leading to the generally high Precision score). Partial tampering was only detected around a third of the time in these tests (leading to the lower Recall score).

The value tampering disguised by a DoS was detected quite simply. Using full-scale value tampering as a DoS-like attack encompassed the selective writes carried out at the same time, and meant no alerts were generated for the latter. A Critical alert would be generated for the offending IP, while its true intention may remain hidden. This is a weakness in the active monitoring. However, it still draws a considerable amount of attention to the attacker. Furthermore, launching such an attack requires a number of assumed factors: a compromised legitimate IP, a high level of insider knowledge on the working of both the ICS network and of the IDS itself.

6. DISCUSSION

6.1 Significance of results

SENAMI achieves its best results, with a high recall and precision (F-Measure > 0.97, cf. Figure 4), when using passive detection techniques against passive threats (sessions 1.1 and 1.2). The passive component of SENAMI, therefore, demonstrates that an entire range of threats can be countered with a high confidence via completely unintrusive, passive monitoring techniques.

Active detection of active threats (sessions 3.1, 3.2.1 and 3.2.2) showcase no True Negatives (perfect recall of 1.0 in each session, cf. Figure 4) and a fair Precision. The active component of SENAMI can, therefore, systematically detect active threats, at the cost of occasional False Positives. In comparison, the detection of active threats with passive monitoring (sessions 2.1 and 2.2) is less effective (F-Measures of 0.87 and 0.53 respectively, cf. Figure 4).

The results of sessions 4.* show the cost of SENAMI to attackers: more sophisticated attacks (partial tampering, combined tampering, and decoy DoS) are necessary to avoid detection. While we acknowledge there is room for improvement in SENAMI, even in its prototype form, the additional investment required from an attacker's perspective is significant.

6.2 Consilience of passive and active monitoring

Our experience with SENAMI highlights that neither active nor passive monitoring can address the challenges of ICS intrusion detection in isolation. The notion of selective, non-invasive, active monitoring is borne out of the need for combining these approaches so that the low computational overheads of a passive approach can be fully harnessed while enriching it with active monitoring of a small set of contextualised variables that do not overload legacy PLCs or compromise real-time properties of an ICS. One interesting avenue to explore is that of more advanced passive monitoring solutions, for example, those based on safe value ranges (or patterns of such ranges) and whether such an approach can further reduce the amount of active monitoring required to mitigate advanced attacks, such as the ones in sessions 4.*.

6.3 Future Work

A number of questions remain open for exploration.

6.3.1 *How selective should active monitoring be?*

The choice of polling rates (e.g., 5 seconds) and monitored values (MD104, DB1, DB2) in SENAMI are somewhat arbitrary and rooted in our knowledge of ICS architectures and well-known attacks. These choices heuristically determine how selective SENAMI is: the less intensive the monitoring, the less strain is put on the target PLC, the more likely an attack could be missed a priori. In this paper we present good detection rates while imposing no visible strain on the target PLC, compared to existing active approaches [19]. However, a stronger theoretical background, backed up by additional experiments, is necessary to precisely determine the trade-offs in terms of precision vs. computational and network overhead in the choice of polling rates and monitored values. In future work, we will explore the influence of variable polling rates and values monitoring on the efficiency and effectiveness of SENAMI.

6.3.2 How is the approach affected by different technologies?

At the moment, SENAMI focuses only on SIMATIC PLCs using the S7 protocol. The influence of the network stack and the PLCs themselves on the reliability of SENAMI is an open question. Lancaster's testbed offers a wide variety of alternative constructors and protocols [10]: future work will investigate similar scenarios in different technological contexts.

6.3.3 PIW Monitoring

To provide wider, more comprehensive applicability and accuracy, the active component should include monitoring of RAW PIW values. The reasons for this are twofold. Firstly, as previously noted, the MD value could in theory be subject to tampering. We believe tampering of the RAW PIW value would likely pose a greater challenge to attackers, especially where subtle modifications were the end goal. Secondly, the practice of data separation based on function (DB1 for process decision making, and DB2 for monitoring) may not be applied in other scenarios, whereas the use of conversions between RAW values and meaningful REAL values is common place. Therefore, through the comparison of PIW against MD and/or DB addressing, our approach would be more widely applicable.

7. CONCLUSION

In this paper, we presented SENAMI, an IDS featuring both passive monitoring and selective, non-invasive, active monitoring. SENAMI avoids introducing computational and network overhead by polling a limited number of contextualised variables from its target. The combination of active and passive monitoring allows the detection of a range of attacks, including combined attacks involving decoys, as demonstrated by a number of experiments.

Although the approach taken in SENAMI constitutes a step forward in active intrusion detection compared to the state of the art, our experience with SENAMI's implementation and its evaluation has also uncovered a number of future research directions. The choice of polling rates and target values is key to performing efficient yet effective intrusion detection. Our future work will focus on refining the current heuristics. The diversity of target equipments is another challenge to be overcome: Lancaster's testbed diversity will be exploited in that regard as we refine and expand the implementation of SENAMI to cover a wider range of devices and protocols.

8. SOURCE CODE FOR SENAMI

The SENAMI source code is freely accessible on github: https://github.com/WilliamJardine/SENAMI.

Acknowledgments

This work is supported in part by the EPSRC/CHIST-ERA grant: DYPOSIT: Dynamic Policies for Shared Cyber-Physical Infrastructures under Attack (EP/N021657/1).

9. REFERENCES

[1] Snort. https://www.snort.org/. [Online: accessed August 1st 2016].

[2] W. Abbas, A. Laszka, Y. Vorobeychik, and X. D. Koutsoukos. Scheduling intrusion detection systems in resource-bounded cyber-physical systems. In *Proceedings of the First ACM Workshop on Cyber-Physical Systems-Security and/or PrivaCy, CPS-SPC 2015, Denver, Colorado, USA, October 16, 2015*, pages 55–66, 2015.

[3] R. Antrobus, S. Frey, B. Green, and A. Rashid. Simaticscan: Towards a specialised vulnerability scanner for industrial control systems. In *Proceedings of the 4th International Symposium on ICS & SCADA Cyber Security Research (ICS-CSR 2016)*, ICS-CSR 2016, 2016.

[4] M. Caselli, E. Zambon, and F. Kargl. Sequence-aware intrusion detection in industrial control systems. In *Proceedings of the 1st ACM Workshop on Cyber-Physical System Security, CPSS 2015, Singapore, Republic of Singapore, April 14 - March 14, 2015*, pages 13–24, 2015.

[5] R. Di Pietro and L. V. Mancini. *Intrusion detection systems*, volume 38. Springer Science & Business Media, 2008.

[6] Digital Bond. PLCScan, 2016.

[7] S. Frey, A. Rashid, A. Zanutto, J. Busby, and K. Szmagalska-Follis. On the role of latent design conditions in cyber-physical systems security. In *Proceedings of the 2nd International Workshop on Software Engineering for Smart Cyber-Physical Systems, Austin, Texas*, pages 43–46. ACM, 2016.

[8] B. für Sicherheit in der Informationstechnik. Die lage der it-sicherheit in deutschland 2014. https://www.bsi.bund.de/DE/Publikationen/Lageberichte/lageberichte_node.html. [Online: accessed May 29th 2015].

[9] J. González and M. Papa. Passive scanning in modbus networks. In *Critical Infrastructure Protection, Post-Proceedings of the First Annual IFIP Working Group 11.10 International Conference on Critical Infrastructure Protection, Dartmouth College, Hanover, New Hampshire, USA, March 19-21, 2007*, pages 175–187, 2007.

[10] B. Green, S. Frey, A. Rashid, and D. Hutchison. Testbed diversity as a fundamental principle for effective ICS security research. In *Proceedings of the First International Workshop on Security and Resilience of Cyber-Physical Infrastructures (SERECIN)*, pages 12–15. Lancaster University Technical Report SCC-2016-01, 2016.

[11] B. Green, M. Krotofil, and D. Hutchison. Achieving ics resilience and security through granular data flow management. In *2nd ACM Workshop on Cyber-Physical Systems Security and Privacy (CPS-SPC), Austria, Vienna, October 28, 2016*, 2016.

[12] D. Hadziosmanovic, R. Sommer, E. Zambon, and P. H. Hartel. Through the eye of the PLC: semantic security monitoring for industrial processes. In *Proceedings of the 30th Annual Computer Security Applications Conference, ACSAC 2014, New Orleans, LA, USA, December 8-12, 2014*, pages 126–135, 2014.

[13] Infracritical. Project SHINE findings report. http://www.slideshare.net/BobRadvanovsky/project-shine-findings-report-dated-1oct2014, 2014. Last accessed 12 April 2016.

[14] A. Kleinmann and A. Wool. Accurate modeling of the siemens S7 SCADA protocol for intrusion detection and digital forensic. *JDFSL*, 9(2):37–50, 2014.

[15] R. Langner. To kill a centrifuge. http://www.langner.com/en/wp-content/uploads/2013/11/To-kill-a-centrifuge.pdf.

[16] R. E. Mahan, J. D. Fluckiger, S. L. Clements, C. Tews, J. R. Burnette, C. A. Goranson, and H. Kirkham. Secure data transfer guidance for industrial control and SCADA systems. Pacific Northwest National Lab (PNNL) Report, http://www.pnnl.gov/main/publications/external/technical_reports/PNNL-20776.pdf, 2011. Last accessed 4 January 2016.

[17] K. McLaughlin, S. Sezer, P. Smith, Z. Ma, and F. Skopik. PRECYSE: cyber-attack detection and response for industrial control systems. In *2nd International Symposium for ICS & SCADA Cyber Security Research 2014, ICS-CSR 2014, 11-12 September 2014, St. Pölten, Austria*, 2014.

[18] S. E. McLaughlin and P. McDaniel. SABOT: specification-based payload generation for programmable logic controllers. In *the ACM Conference on Computer and Communications Security, CCS'12, Raleigh, NC, USA, October 16-18, 2012*, pages 439–449, 2012.

[19] A. Nicholson, H. Janicke, and A. Cau. Position paper: Safety and security monitoring in ICS/SCADA systems. In *Proceedings of the 2nd International Symposium on ICS & SCADA Cyber Security Research 2014*, ICS-CSR 2014, pages 61–66, UK, 2014. BCS.

[20] B. Paske, B. Green, D. Prince, and D. Hutchison. Design and Construction of an Industrial Control System Testbed. In *PGNET*, pages 151–156, 2014.

[21] SANS. Analysis of the cyber attack on the ukrainian power grid. https://ics.sans.org/media/E-ISAC_SANS_Ukraine_DUC_5.pdf. [Online: accessed August 1st 2016].

[22] M. Strohmeier, V. Lenders, and I. Martinovic. Intrusion detection for airborne communication using PHY-Layer information. In *Detection of Intrusions and Malware, and Vulnerability Assessment - 12th International Conference, DIMVA 2015, Milan, Italy, July 9-10, 2015, Proceedings*, pages 67–77, 2015.