electronics

MDPI

Article

Testing Commercial Intrusion Detection Systems for Industrial Control Systems in a Substation Hardware in the Loop Testlab

Jon-Martin Storm [1,2,*], Siv Hilde Houmb [3], Pallavi Kaliyar [3], Laszlo Erdodi [3] and Janne Merete Hagen [1,2]

[1] Department of Informatics, University of Oslo, 0373 Oslo, Norway; jannehag@ifi.uio.no
[2] The Norwegian Water Resources and Energy Directorate, 0301 Oslo, Norway
[3] Department of Information Security and Communication Technology, Norwegian University of Science and Technology, 2815 Gjøvik, Norway
* Correspondence: jonmartp@uio.no

Abstract: Industrial Control Systems (ICS) are increasingly integrated with Information Technology (IT) systems, blending Operational Technology (OT) and IT components. This evolution introduces new cyber-attack risks, necessitating specialized security measures like Intrusion Detection Systems (IDS). This paper presents our work on both developing an experimental protocol and conducting tests of various IDS types in a digital substation hardware in the loop (HIL) testbed, offering insights into their performance in realistic scenarios. Our findings reveal significant variations in IDS effectiveness against industrial-specific cyber-attacks, with IT-specific IDSs struggling to detect certain attacks and changing testlab conditions affecting the assessment of ICS-specific IDSs. The challenges faced in creating valid and reliable evaluation metrics underscore the complexities of replicating operational ICS conditions. This research enhances our understanding of IDS effectiveness in ICS settings and underscores the importance of further experimental research in HIL testlab environments.

Keywords: smart grid; cyber security; industrial control system; intrusion detection system; SCADA; digital substation; IEC 60870-5-104; IEC 61850

check for updates

Citation: Storm, J.-M.; Houmb, S.H.; Kaliyar, P.; Erdodi, L.; Hagen, J.M. Testing Commercial Intrusion Detection Systems for Industrial Control Systems in a Substation Hardware in the Loop Testlab. *Electronics* **2024**, *13*, 60. https://doi.org/10.3390/electronics13010060

Academic Editor: Aryya Gangopadhyay

Received: 23 November 2023
Revised: 11 December 2023
Accepted: 19 December 2023
Published: 21 December 2023

Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

While power grids play an increasingly crucial role in modern society's infrastructure, their reliance on specialized Information Technology (IT), known as Operational Technology (OT), Industrial Control Systems (ICS), or Supervisory Control and Data Acquisition (SCADA), becomes more critical. These systems enable engineers to monitor and control various elements of the power system from a central location. However, the trend toward greater sensor integration, cloud computing, and remote access has led to a more profound intertwining with general IT systems. This integration does not only elevate the functionality and efficiency of these systems but also significantly increases their vulnerability to cyber-attacks. Consequently, there is an urgent need to re-evaluate and enhance the cyber security measures for these critical infrastructures. Their security requirements are transforming in response to these changes, necessitating ongoing updates and improvements.

Cyber threats are evolving, with specific increases in cyber-attacks targeting ICS. Stuxnet destroyed more than 200,000 machines in 14 Iranian facilities, including the Natanz uranium-enrichment plant [1]. Many analyses claim Stuxnet damaged almost one-fifth of Iran's nuclear centrifuges, consequently delaying the Iranian uranium enrichment program [2]. The Stuxnet attack also highlighted the critical role of ICS domain knowledge in cyber-attacks.

The Ukraine cyber-attacks in 2015 and 2016 targeted the Ukraine Power grid to disrupt grid operations. The 2015 cyber-attack employed the BLACKENERGY3 [3] malware, targeting the distribution level and disrupting 30 substations [4]. The attack left approximately 225,000+ customers without power for 1–6 h [5]. The BlackEnergy3 malware, an advanced

version of the BlackEnergy2 malware (Attacks related to the BlackEnergy malware are attributed to the Russian Sandworm group that has targeted numerous industries in cyber and espionage operations), was initially used to access the corporate network. The attacker then pivoted to the SCADA network [6], denying operators SCADA system interaction while remotely opening circuit breakers and taking substations offline [7].

The 2016 attack utilized the CRASHOVERIDE malware, also known as Industroyer [8,9]. The malware directly exploited vulnerabilities in industrial protocols and disrupted a transmission level substation, which caused an outage in Kyiv for about one hour [10]. According to Dragos, the CRASHOVERIDE malware is not unique to any configuration or vendor [11]. It leverages network communication and grid operation knowledge, making it an immediate threat to European, Middle Eastern, and Asian grid systems. In 2022, Ukraine faced another cyber-attack using an updated version of the Industroyer malware, known as "Industroyer2", specifically designed to compromise the country's energy sector [12]. This sophisticated cyber-attack on a critical infrastructure underscores the advanced capabilities of cyber adversaries in exploiting vulnerabilities in ICSs.

Some authors claim that a combination of security measures could have either prevented the cyber-attacks mentioned or reduced their consequences [13]. Even a basic security level outlined in the IEC 62443 standard family [14] might have prevented the 2015 Ukraine attack [15]. An important part of these measures is the capability to detect an attack.

Detecting cyber-attacks in ICS environments requires insights into the activities within the system itself, including the associated networks, communication protocols, hosts, controllers, sensors, and other ICS components. While detecting cyber-attacks in ICS environments is crucial, bridging this need with effective solutions presents a challenge. Currently, the market offers only a limited number of commercial solutions, like intrusion detection systems (IDS), equipped to handle these complex requirements. Moreover, there is a noticeable gap in implementing ICS-specific methods, as suggested by recent research [16].

1.1. Motivation and Contribution

Cyber-attack detection on ICS must occur early to prevent physical damage and process disruptions like power outages. While ICS IDSs are designed for this purpose, their adoption has been slow. This can partly be due to the lack of proven effectiveness in realistic environments and the insufficient implementation of ICS-specific methods [16]. Consequently, evaluating both commercial and open-source ICS IDSs in a controlled, realistic environment becomes essential.

It is vital to undestand the differences in detection capabilities between traditional IT-specific IDSs and those designed specifically for ICS. To be able to evaluate different IDS types, we propose running a controlled experiment on a Hardware In the Loop (HIL) lab. This approach mirrors the implementation challenges in production environments, emphasizing the need for a robust experiment protocol for setup validation and fair comparison.

We have formulated three research questions:

1. How effective are IT-specific IDSs in detecting IT-related attacks within industrial control systems?
2. To what extent do ICS-specific IDSs struggle to detect novel, tailor-made attacks targeted at industrial control systems?
3. What are the inherent challenges in creating valid and reliable evaluation metrics for IDS performance in a live testlab environment?

Our research contributes to the field of ICS security by providing empirical insights from extensive experiments in a Digital Substation (DS) HIL testlab. In addition to our primary research, we have also made a contribution by developing an experimental protocol that forms the foundation of these experiments. Together, these contributions advance both the theoretical understanding and practical application of IDS in the realm of ICS security.

1.2. Organization

This paper is structured as follows. Section 2 presents an overview of industrial control systems. Section 3 presents an overview of intrusion detection systems, and Section 4 presents other research into evaluating IDSs in IT and OT. Section 5 discusses the tools and experiment methods. Section 6 presents the results. Section 7 includes the discussion. Section 8 contains conclusions and further work.

2. Industrial Control Systems in Power Systems

An ICS is a specialized IT system used to control a physical process. The National Institute of Standards and Technology (NIST)'s Guide to ICS Security defines ICS as: "An ICS consists of combinations of control components (e.g., electrical, mechanical, hydraulic, pneumatic) that act together to achieve an industrial objective (e.g., manufacturing, transportation of matter or energy)" [17], p. 2-1. An ICS is a system used to control a physical process that often entails vendor-specific and COTS (Commercial-off-the-shelf) components.

Power systems' primary function is to generate and transfer electric energy to consumers. A power system encompasses one or more generating sources and transmission lines operated under joint management to supply electricity [18]. Power system operations are enabled by specialized IT often referred to as OT, ICS, or SCADA. These systems allow engineers to monitor and control the power system's multiple locations, generators, and substations from a central terminal. A power system ICS is commonly composed of both digital and analog components like Remote Terminal Units (RTU), Programmable Logic Controllers (PLC), Intelligent Electronic Devices (IED), Human Machine Interfaces (HMIs), Breakers, Sensors, and various specialized software. All these components communicate in real time through industrial communication protocols.

Security Issues in the IEC 60870-5-104 Protocol

IEC 60870-5-104 (IEC/104) [19] is a communication protocol between SCADA systems and RTUs in power systems. It is an extension of IEC 60870-5-101 [20] and is widely used in the European power grid. Table 1 outlines key characteristics of the IEC/104 protocol.

Table 1. Overview of the IEC 60870-5-104 (IEC/104) protocol.

Part	Description
Transport protocol	TCP/IP or OSI stack
Data format	Binary
Transmission mode	Asynchronous
Application layer protocol	Application Service Data Units (ASDUs)
ASDU types	Information, Supervisory, Control
Object addresses	Two-level hierarchy: Type Identifiers and Object Identifiers
Function codes	43 function codes for various control and monitoring tasks

Despite its widespread use, IEC/104 has several vulnerabilities attackers can exploit. Some of these vulnerabilities include:

- Lack of authentication: The protocol does not provide robust authentication mechanisms, making it easy for attackers to impersonate legitimate users.
- Lack of encryption: The protocol does not provide encryption, making it easy for attackers to eavesdrop on the communication and steal sensitive information.
- Buffer overflow: The protocol is susceptible to buffer overflow attacks, where an attacker can exploit a vulnerability in the protocol to inject malicious code into a system.
- Denial of Service attacks: The protocol is susceptible to Denial of Service attacks, where an attacker can flood a system with traffic to disrupt its operation.

3. Cybersecurity Detection Controls

3.1. Intrusion Detection System

Multiple measures, known as controls, are needed to achieve a sufficient level of cybersecurity. Companies need organizational controls, such as an Information Security Management System (ISMS) and risk assessment; technical protection controls, such as firewalls and hardening; response controls, such as emergency preparedness and event handling; and recovery controls, such as backups. Another essential control involves detective measures, which ensure the detection of unwanted events and enable a timely response to them. NIST defines detection as one of the basic cybersecurity activities [21], emphasizing it as a core function in their cybersecurity framework.

IDSs are the most commonly implemented detection tools in IT security. IDSs monitor system data using various techniques to generate alerts on possible intrusions. Common data sources include network data and host data, while prevalent analysis techniques comprise rule-based, anomaly-based, and behavior-based methods [22–25].

3.1.1. Network-Based

In most cases, network-based monitoring is performed by passively sniffing traffic at Layer 2 and Layer 3 switches or by means of dedicated in-line appliances. Network traffic data is gathered locally on each network segment and often forwarded to a centralized data analysis engine. The analysis engine can be localized inside the ICS and as part of the enterprise network. The analysis is often performed using deep packet inspection to dissect traffic from serial and Ethernet control network equipment.

3.1.2. Host-Based Monitoring

In most cases, host-based monitoring in an ICS is performed using non-intrusive end-point monitoring software, such as agents, scripts, and built-in tools such as Syslog or similar. Host-based monitoring is conducted directly on the ICS endpoints, including HMIs, engineering stations, and other critical system components. Data is gathered locally at each end-point and, in many cases, forwarded to an analysis engine, either localized inside the ICS or centralized at the plant. Monitoring and aggregating events across plants requires data to be forwarded to the enterprise network or cloud services.

3.1.3. Signature or Rule-Based

Signature-based or rule-based detection compares observed events with signatures or rules to identify intrusions. Anything matching the signature or rule will create an alert [22].

3.1.4. Anomaly-Based IDS

Anomaly-based detection compares observed events with what is considered normal for the system to identify intrusions. What is considered normal is usually a recorded profile or baseline of the system during normal operations. Anything deviating from the baseline will create an alert [22].

3.2. IDS for ICS

Detection is vital in ICSs; however, it can be more challenging to implement, with fewer commercial tools available [13]. The real-time nature of most ICS demands that any implemented IDS have minimal to no impact on the data flow in the network. Availability is the most important security goal for ICSs, as lack of information can have long-term consequences for the physical process. Any intrusive actions, such as an intrusion protection system, can be dangerous. An ICS IDS also needs to be able to parse industrial protocols to detect protocol-specific attacks. Most hosts in an ICS cannot have agents installed or send security logs to a centralized location [13,26].

NIST's guide to ICS security presents that the security architecture of an ICS must incorporate mechanisms to monitor, log, and audit activities occurring on various systems

and networks to be able to validate that no policy violations or cyber incidents have hindered the operation of the system. The guide mentions signature-based and anomaly-based detection as valuable options for monitoring in ICS [17].

IEC 62443 is a standards family focusing on ICS security [14]. The standards address end-users (asset owners), system integrators, security practitioners, and control systems manufacturers responsible for manufacturing, designing, implementing, or managing ICSs. IEC 62443-2-1, ref. [27] which was first published in 2010 and has since been updated and is now circulated for formal approval, requires that:

"The asset owner shall ensure that the organization periodically uses manual or automated processes to discover and address:

- Undocumented and unauthorized devices and software connected to or communicating within the ICS.
- Undocumented or unauthorized network traffic.
- Undocumented vulnerabilities in the ICS.
- Other security anomalies and non-conformities ".

4. Related Work

4.1. Lack of Testing

Storm et al. [16] surveyed 239 research papers on the use of process data and features of ICSs in intrusion detection. The study revealed an increasing trend in published papers addressing detection capabilities in industrial control systems, with a rise in published papers from 2011. Two hundred and six papers (206) addressed important features of ICSs in their proposed detection methods. However, only 38 papers covered testing in a live test environment, none covered testing on a system in operation, and none covered comparison to existing commercial IDS solutions for ICS. The authors suggested research into implementing and verifying proposed detection methods in both live test environments and systems in operation and testing existing commercial IDSs for ICSs.

In one paper [28], Giraldo et al. presented a survey of 50 papers based on physics-based anomaly detection. The authors explored the limitations, challenges, location of detected attacks, detection methods, and validation methods of the different approaches. The authors found that only seven papers out of 50 validated their approaches on a testbed, and cybersecurity-related works used various non-standard models. In contrast, the control and power grid works use standardized models to model their physical systems.

4.2. Examples of Tested IDS in ICS

In 2008, Oman and Phillips [29] showed the performance of a customized event monitoring and IDS through a SCADA/sensor testbed. During their test runs, they noticed improvements in the security and auditing performance of the system. In 2010, Carcano et al. [30] proposed a state-based IDS to detect complex attacks in a SCADA architecture. The authors claim that their prototype IDS can detect the attacks that interfere with the SCADA system's behavior by using a rule-based language concept for field device states and Modbus Signatures. The proposed IDS prototype analyses the Modbus packets using two detection techniques: (1) single packet signature and (2) keeping track of the state of the industrial system. These techniques help the IDS identify a set of misbehaving commands that can lead a system to a critical state. The authors compared their IDS with Snort and find that their IDS can detect complex attacks that Snort does not identify.

In 2013, Lin et al. [31] proposed a Bro [32] specification-based Intrusion Detection Framework (IDF). Bro is a runtime network traffic analyzer, and the authors used Bro and built a parser to support the DNP3 protocol in SCADA systems. The proposed IDF observes and analyzes all DNP3 events and their semantic information. The IDF creates and designs its electric grid security policies to perform analysis based on semantic information. IDF also includes protocol validation policies, ensuring that the network traffic follows predefined patterns in their communication. The authors tested their IDF in multiple

scenarios and found that the IDF caused a large number of alerts when malformed network packets were found.

After this, in 2014, Yang et al. [33] proposed a Specific Intrusion Detection System (SIDS) for Power Networks. SIDS combines distributed IDS technology, behavioral monitoring, and security enclaves to make SCADA systems more secure. The authors claimed that SIDS monitors smart grids and is compatible with SIEM technology. SIDS analysed different attributes and the behavior of the SCADA protocols in order to detect various known and unknown attacks in the power grid. SIDS was implemented and tested on a realistic substation testbed with different cyber-attack scenarios.

Later, in 2016, Cruz et al. [34] proposed a Distributed Intrusion Detection System (DIDS) for SCADA industrial control systems. DIDS was developed for the CockpitCI [35] architecture using domain-specific knowledge like operation, functionality, management, and integration. DIDS leverages the benefits of both signature-based and rogue threat detection. DIDS was validated using an especially designed hybrid testbed of an electrical distribution grid SCADA system.

Recently, in 2018, Vivek et al. [36] presented a security evaluation of two (Bro [32], and Snort [37]) anomaly-based IDSs in a smart grid SCADA environment. This work presented the performance of both IDSs in terms of detection rate and latency in the alert packets by applying time-based rules on network traffic to identify malicious packets in the network. The authors used Iowa State University's PowerCyber testbed to compare the IDSs during attack implementation and detection. The author's results suggested that Bro performed better than Snort regarding latency and detection rate. However, the authors did not detail the experimentation protocol used for the performance analysis of the two IDSs.

4.3. Controlled Experiments and Experiment Protocols

Kitchenham et al. [38] proposed preliminary guidelines for conducting empirical research in software engineering, focusing on controlled experiments. The authors presented a framework for planning, conducting, analyzing, and reporting software engineering experiments, emphasizing the importance of proper experimental design, data collection, and analysis. The guidelines provide a foundation for researchers to develop rigorous and replicable experimental protocols in software engineering.

Jedlitschka et al. [39] presented guidelines for reporting software engineering experiments to improve empirical research quality, transparency, and reproducibility. The authors discussed the importance of clearly reporting the experimental context, design, data collection and analysis methods, results, and conclusions. They provided a structured reporting template that can be used to document and communicate the results of software engineering experiments.

Table 2 summarizes the essential aspects of related works pertinent to this paper.

Table 2. Summary of related work.

Authors	Year	Research Methodology	Relevant Results
Giraldo et al. [28]	2018	Systematic review	Lack of validation of anomaly detection on a testbed
Storm et al. [16]	2021	Systematic review	Lack of verification of ICS (Industrial Control System) IDSs (Intrusion Detection System) in live test environments
Oman and Phillips [29]	2008	Conceptual ICS IDS validation on a testbed	The ICS IDS detected critical events and configuration errors
Carcano et al. [30]	2012	Testlab experiment comparing IDSs	The ICS-specific IDS detected complex attacks
Lin et al. [31]	2013	Testlab experiment	The developed ICS-specific IDS detected ICS-specific attacks

Table 2. *Cont.*

Authors	Year	Research Methodology	Relevant Results
Yang et al. [33]	2014	Testbed experiment to validate ICS IDS	ICS IDS detected multiple ICS-specific attacks
Cruz et al. [34]	2016	Hybrid testbed experiment	The ICS-specific IDS detected ICS-specific attacks
Vivek et al. [36]	2018	Testbed experiment	The modified IDS detected ICS-specific attacks
Kitchenham et al. [38]	2002	Review	Research guidelines
Jedlitschka et al. [39]	2008	Survey and interviews	Experiment reporting guidelines

5. Materials and Methods

5.1. Controlled Experiment

We aimed to evaluate and compare ICS IDSs regarding their detection precision and classification, which requires a high degree of control. We chose to conduct a controlled experiment to ensure all IDSs undergo testing in a consistent and repeatable manner and eliminate biases from uncontrolled variables.

According to Wohlin [40], a controlled experiment is an empirical inquiry that manipulates some input factors or variables of the studied setting based on randomization and measures the effect on outcome variables. To conduct a controlled experiment, researchers must sufficiently manage all other variables and the environment. This control enables statistical analysis to determine the impact of changes in input variables on the output variables. Experiments give researchers a high degree of control over the study and also a high degree of control over collecting measurements.

A controlled experiment needs a controlled environment where variables are known, a way to measure the variables, and a way to ensure the experiment's validity. We achieved this by carefully scoping and planning both the environment and the operation of the experiment. Implementing proper experiment protocols is essential to ensure the smooth operation of the experiment and to validate that the environment is under control.

5.2. Testlab

We used a DS HIL Testbed, controlled detection architecture, IDS tools, and specific attack scenarios for our controlled environment.

For the immediate environment, we used an existing HIL testbed, the DS enclave located at the Institute for Energy Technology (IFE) in Halden, Norway [41]. The DS enclave includes hardware and software for one substation breaker bay, including a protection relay. The topology of the testbed is illustrated in Figure 1.

The figure depicts the DS enclave infrastructure for a Norwegian Digital Substation, aligning with the IEC 62443 standard. It features a layered design and incorporates both station and process buses in accordance with the IEC 61850 standard series [42]. The enclave router acts as a demilitarized zone (DMZ). This DMZ separates IT and OT environments. The infrastructure incorporates remote access software for secure connectivity. It also utilizes virtualization platforms and container technologies. These technologies support computing, storage, and networking. The design is tailored for flexibility in simulating hardware- and software-based attack scenarios. The infrastructure is designed for efficient data capture and analysis, with provisions for addressing challenges such as large PCAP file management and resource limitations. The DS enclave has a central logging server for collecting network traffic and Syslogs. This server enables detailed data analysis and tracking of attacks. The setup supports multiple detection systems, with network traffic directed to hardware appliances and software-based IDS.

The DS enclave is equipped with IECTest, a specialized software conforming to the IEC/104 standard, which we utilized to emulate the operations of a control center [43]. This versatile tool allowed script-based programming, enabling the automation of specific com-

mands, such as the opening and closing circuit breakers, to be executed at predetermined times within the experiment's duration. Each experimental run employed the identical script to ensure consistency and replicability of results, maintaining uniformity across all test scenarios.

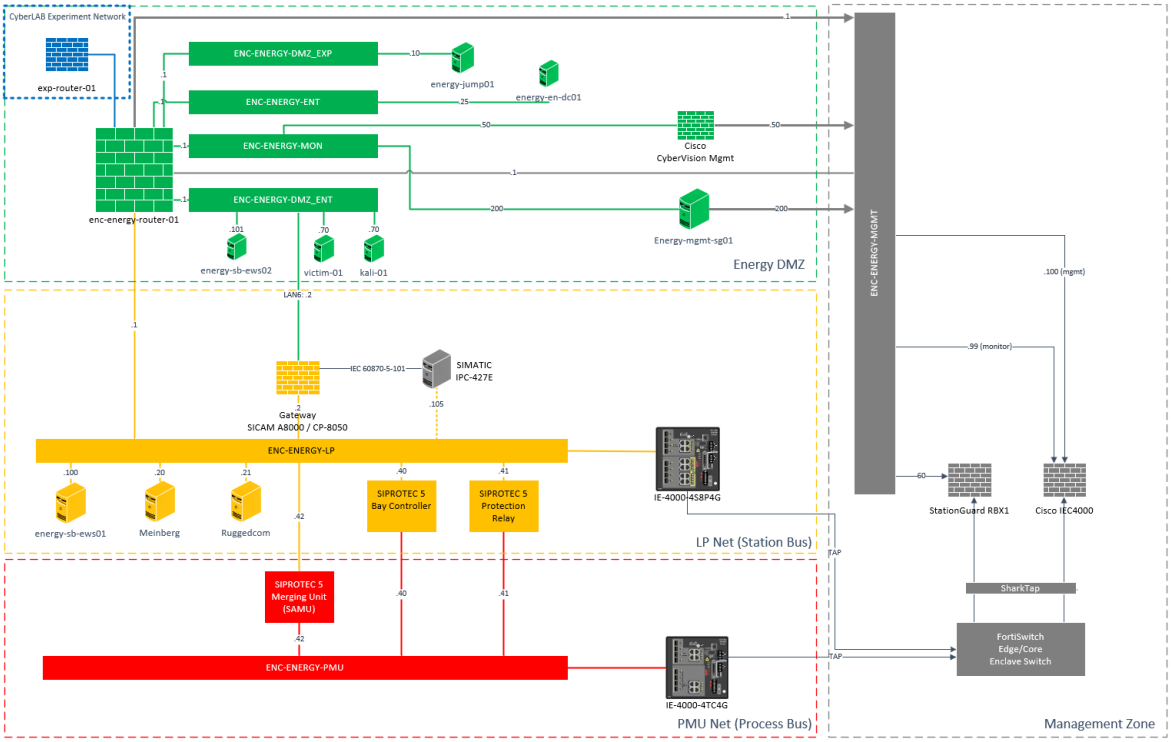


Figure 1. The Digital Substation (DS) Hardware In the Loop (HIL) testbed infrastructure, illustrating the connection of each DS level to the tested IDSs through a management zone. Note: In the figure, '.42' and similar expressions are used as shorthand for IP addresses, indicating partial addresses where full addresses are not specified or required.

5.3. Detection Architecture

We set up all the IDSs with access to the same information so they could "see" the same events. All network traffic going through the edge router, the process bus switch, and the station bus switch were mirrored and fed into the IDS sensors. We also installed sensors for host-based IDS with the same privileges on the same hosts.

To remove bias in the tools used by the IDSs, we configured them up to send all alerts to the same Security Information Event Management (SIEM) system using the Syslog protocol.

In testing and creating an experiment protocol for ICS IDS evaluation, we used three commercial and two open-source (free) IDSs. Cisco Cyber Vision (https://www.cisco.com/c/en/us/products/security/cyber-vision/) (accessed on 10 October 2023) (IDS-A), Omicron Stationguard (https://www.omicronenergy.com/en/products/stationguard/) (accessed on 10 October 2023) (IDS-B), Secure-NOK SNOK (https://www.securenok.com/our-products/) (accessed on 10 October 2023) (IDS-C), Suricata (https://suricata.io/) (accessed on 10 October 2023) (IDS-D) and Wazuh (https://wazuh.com/) (accessed on 10 October 2023) (IDS-E). A, B, and C are commercial, and D and E are free and open source. A, B, and D are network-based, E is host-based, and C is a hybrid. B, D, and E are rule-based, A is both anomaly and signature-based, and C is behavior-based. For our experiment, the most interesting difference between the five IDSs is how they can be categorized based on the environments they target.

- Environment-focused IDSs: Cisco CyberVision, Omicron StationGuard, and Secure-Nok SNOK are specifically designed for OT environments and industrial control

systems. They are tailored to address the unique requirements and protocols used in OT networks.

- General-purpose IDSs: Suricata and Wazuh are versatile solutions that can be deployed in various environments, including IT and OT networks. They offer a broad range of detection capabilities suitable for different use cases.

In addition to this, they can be categorized by their detection capabilities and deployment options.

- Detection capabilities: Cisco CyberVision, Omicron StationGuard, and Secure-Nok SNOK focus on monitoring OT-specific protocols, while Suricata and Wazuh offer broader detection capabilities suitable for a variety of network protocols and traffic patterns.
- Deployment options: Cisco CyberVision, Omicron StationGuard, and Secure-Nok SNOK are commercial solutions with vendor-specific support and deployment options. Suricata and Wazuh are open-source projects, offering greater flexibility for customization and integration with other security tools.

5.4. Attack Scenarios

To assess the precision and attack classification capabilities of the IDS, we selected a set of validated cyber-attacks [43] for implementation in the testbed.

In the 2015 Ukraine power grid attack, attackers initially gained access to electrical breakers and then executed a denial of service strategy to lock out the operators [4]. We employed a carefully selected set of validated attacks to mimic this attack sequence on a smaller scale. Our approach included phases of active reconnaissance (Attack 1), followed by the opening of breakers (Attacks 2 and 3), and culminating with a series of denial of service attacks (Attacks 4 and 5). The objective was to evaluate the IDSs in a setting that mirrors real-world conditions, where an attacker typically utilizes various attack methods rather than a single, isolated approach. The adversary profile was a competent, motivated, skilled attacker who has established a foothold through privilege escalation and pivoting. We scripted the five attacks to run in a fifteen-minute window, constituting one experiment run. Figure 2 illustrates the timeline of our executed attacks, with further details provided in Table 3.

Table 3. Table of attacks utilized for IDS evaluation during the experiment [43].

ID	Attack Type	Attack Aim	Attack Technique
1	Active Reconnaissance	Obtaining details (e.g., inputs and outputs) about the devices and items in use with information object addresses	Sending out interrogation commands with packet injection
2	Operation failure	Opening the breaker if it is closed and closing the breaker if it is opened in order to cause malfunction in the operation	Opening/Closing the breaker with packet injection/spoofed controlling station IP
3	Operation failure	Opening the breaker if it is closed and closing the breaker if it is opened to cause a malfunction in the operation	Opening/Closing breaker with ARP (Address Resolution Protocol) poisoning
4	Denial of Service	Poison the ARP table of the controlling station and the (Remote Terminal Unit) RTU to send all packets to the wrong device	ARP poisoning without packet forward
5	Denial of Service	Disable communication between the RTU and the control station, sending spoofed RESET right after the connection re-established between the controlling station and the RTU	ASDU RESET flood with packet injection/controlling station source IP spoofing

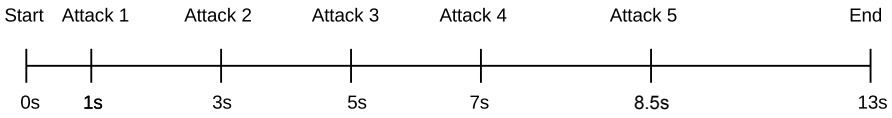


Figure 2. Timeline of attacks in a single experiment run.

5.5. Experiment—Steps and Quality Assurance

Our experiment protocols can be seen in Tables 4 and 5. We needed to be explicit in using two protocols: one for setup to verify that we had a controlled environment and one for running the experiment. Secondary protocols for running the attacks were also created but are outside the scope of this paper and not presented.

Table 4. IDS setup—detailed steps and quality assurances.

Step	Detailed Steps	QA Task
1	Startup HIL testbed and all other IDSs	
2		Confirm HIL testbed running
3		Confirm other IDSs running
4		Confirm SIEM (Security Information and Event Management) running
5	Check IDS config/baseline	
6		Confirm correct config/baseline
7	Run lab for 30 min	
8	Check for unintended alerts	
9		Confirm no unintended alerts in SIEM
10	Run one detectable attack and check response	
11		Confirm alert in SIEM

Table 5. IDS experiment—detailed steps and quality assurance.

Step Number	Detailed Steps	QA Task
1	Run lab without attack for 1 min	
2	Check for errors in output	
3		Confirm no errors in output
4	Note down experiment start time (ISO 8601)	
5	Run attack script	
6		Confirm attack script has been run
7	Run lab without attack for 4 min	
8	Note down experiment end time (ISO 8601)	
9	Check for errors in output	
10		Confirm no errors in output

5.6. Analysis

In our case we were looking to verify the effectiveness of the IDSs we compare and evaluate [44]. The confusion matrix is the most commonly used in such cases, shown in Figure 3.

- True Positive (TP): Attacks/intrusions that are successfully detected
- False Positive (FP): Normal events that are wrongly classified as attacks
- True Negative (TN): Normal events that are classified as such
- False Negative (FN): Attacks/intrusions that are not detected

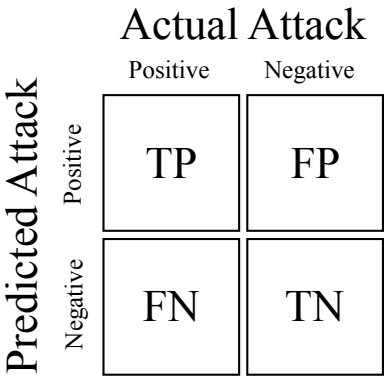


Figure 3. A confusion matrix illustrating the relationship between true positives, true negatives, false positives, and false negatives.

As the effectiveness of an IDS is best measured by combining the terms in the confusion matrix, other numeric values are usually calculated based on these terms [44]. Since we did not evaluate the network level values, using metrics like the classification rate (CR), which depends on correctly classified instances and the total number of instances, was not ideal [45]. Instead, we chose to use the detection rate (*DR*) as our primary metric, which is also relevant at our aggregated level. We compared the detected attacks gathered in the SIEM with the attacks executed in each test:

$$DR = \frac{\text{Correctly detected attacks}}{\text{Total number of attacks}} = \frac{TP}{TP + FN} \quad (1)$$

We also separately considered whether the IDS could correctly classify an attack according to the ICS domain. This mainly addresses the issue that the attacks in Table 3 could be detected by IDSs as an attack but not necessarily classified correctly according to the attack type. For example, an IDS might have detected attack ID 3 as ARP poisoning but still fail to classify the attack as an operating failure or breaker opening.

6. Results

Table 6 presents the total number of alerts generated by each IDS and sent as Syslog messages to the SIEM system during each of the five experiment runs. The data reveals variability in the alert counts across different tests for each IDS, with IDS-E exhibiting the most significant fluctuations in alert numbers. Table highlights the differing sensitivities and response patterns of the IDSs under test conditions.

Table 6. Number of alerts sent by the different IDSs to the SIEM.

Eksperiment Run Number	IDS-A	IDS-B	IDS-C	IDS-D	IDS-E
1	7	15	20	4	2
2	14	21	30	7	27
3	13	20	32	6	24
4	17	24	30	3	24
5	17	16	27	4	4

Table 7 provides a comprehensive overview of the detection rates for different attacks by each IDS across all five experiment runs. It is expressed as a percentage, indicating the proportion of runs in which each IDS successfully detected and alerted an attack. This table highlights the effectiveness of each IDS in recognizing attacks, showcasing their strengths and weaknesses in attack detection within the experimental setup.

Table 7. Detected attacks overall. The percentage of the runs of the experiment in which an attack was detected, i.e., a Syslog message correctly alerting an attack was sent from the to the SIEM.

Attack ID	IDS-A	IDS-B	IDS-C	IDS-D	IDS-E
1	0%	100%	0%	40%	0%
2	0%	100%	100%	0%	0%
3	0%	100%	100%	0%	0%
4	0%	100%	100%	0%	0%
5	100%	100%	100%	0%	0%

Table 8 examines the ability of each IDS to correctly classify attacks within an ICS context, based on the alerts sent to the SIEM. The table categorizes the responses as 'Yes' or 'No,' indicating whether each IDS accurately identified the nature of the attack as per the classification provided in Table 3. Notably, Attack 4, as indicated by the footnote, is not specific to ICS, providing a context for interpreting the classification performance of the IDSs for this particular attack scenario.

Table 8. Correctly classified within the ICS domain. If the IDS sent a Syslog message to the SIEM, did it correctly classify the attack according to Table 3.

Attack ID	IDS-A	IDS-B	IDS-C	IDS-D	IDS-E
1	No	No	No	No	No
2	No	Yes	No	No	No
3	No	Yes	No	No	No
4 [1]	-	-	-	-	-
5	Yes	Yes	No	No	No

[1] Attack 4 is not an ICS-specific attack.

7. Discussion

7.1. IDS Performance in ICS Environments

In our study, the performance of IT-specific IDSs, particularly IDS-D and IDS-E, highlighted their limited effectiveness in detecting many ICS-specific attacks, aligning with expectations and existing research. Notably, IDS-D failed to identify ARP attacks (Attacks 3 and 4), which are IT-specific but impact the ICS at layer 2 of the OSI model. This failure is attributed to IDS-D being developed for higher OSI model layers, confirming the known limitations of IT-centric systems like Suricata in ICS environments. These systems often lack support for key ICS protocols essential for effective threat detection in SCADA networks [46]. However, this observation also suggests that IT-specific IDSs are only partially ineffective in ICS contexts, as they can detect specific IT-based attacks that traverse into ICS environments. This underscores the necessity for a layered security approach, combining IT and ICS-specific measures, to provide comprehensive protection against a broad spectrum of cyber threats. Our inclusion of an IT-centric IDS was to validate these limitations empirically, reaffirming the need for IDS solutions that are either inherently designed for or skillfully adapted to the unique requirements of ICS. This emphasizes the importance of a thoughtful and informed approach in selecting and configuring IDSs for robust security in specialized ICS environments.

7.2. Comparative Analysis of Rule-Based vs. Behavior-Based IDSs

In our experimental analysis, the effectiveness of IDS-A, as detailed in Table 7, was not as high as expected in detecting attacks. The extensive need for tuning and resetting in the lab might have significantly reduced the effectiveness of the IDSs by altering their baselines. This issue was not unique to IDS-A; we observed similar challenges across all ICS-specific IDSs, including IDS-B and IDS-C. The dynamic and continually changing testbed environment, with potentially outdated baselines, underscores a significant challenge in maintaining the effectiveness of ICS-specific IDSs within such variable environments. This highlights the need for continual adaptation and updating of IDS con-

figurations to stay aligned with evolving ICS environments and threat landscapes, emphasizing the complexity and expertise required for effective deployment and management of these systems.

Furthering our analysis, we observed that within the constraints of our testlab, the rule-based IDS-B outperformed the behavior-based IDSs, IDS-A and IDS-C. As shown in Table 8, only IDS-B correctly classifies attacks within the ICS domain. We attribute this notable performance difference primarily to the dynamic nature of our testlab, characterized by frequent changes that can significantly challenge behavior-based IDSs. These systems, particularly IDS-A and IDS-C, rely on machine learning algorithms and require stable and consistent environments for sufficient training and baseline establishment. The continual alterations in the testlab environment necessitated frequent updates to these baselines, impacting the accuracy and reliability of these behavior-based systems.

In contrast, the rule-based IDS-B, which operates based on predefined rules rather than learned behaviors, was less susceptible to such environmental variability. Its performance, therefore, remained more consistent and effective in detecting intrusions despite the changing conditions of the testlab. This observation underscores the importance of considering the operational environment and its stability when deploying IDSs, especially behavior-based ones that rely on machine learning algorithms for anomaly detection.

7.3. Challenges and Insights from IDS Testing in Dynamic Environments

In the broader context of evaluating IDSs within SCADA environments, other studies are noteworthy reference points [33,34]. However, our research expands upon this by evaluating a range of IDSs and implementing a thorough and transparent experimental protocol. This detailed documentation of our methodology significantly enhances the reproducibility and credibility of our findings. Moreover, the comprehensive nature of testing in a live lab environment, including all inherent system complexities and potential faults, offers a more realistic assessment of IDS performance in actual operational settings. This approach underscores the value of thorough system testing, which can reveal intricacies and challenges that might not be evident in more controlled or theoretical evaluations. Additionally, our focus on examining commercial IDSs caters to the industry's growing interest and need for practical, deployable solutions, moving beyond theoretical models to evaluate how these systems perform in real-world scenarios. This shift toward a more holistic and pragmatic evaluation approach is crucial for advancing the field of cyber security in SCADA and other critical infrastructure systems.

7.4. Development and Refinement of Experimental Protocol

Our experience with developing the experimental protocol revealed several challenges in creating a controlled environment for evaluating IDS performance. Each of the experiments conducted in the lab led to iterative improvements in the protocol. Initial experiments highlighted the necessity for an IDS setup protocol and modifications to address network and topology changes. Subsequent runs emphasized the importance of having a singular focus for each experiment and allocating time and resources to address setup issues. The reported experiment, with its specific attack scenarios and focused goal, demonstrated the effectiveness of the refined protocol in evaluating ICS IDS precision and classification. Our journey in developing this protocol underscores the complexity and dynamism of ICS environments and the critical need for robust, adaptable, and comprehensive experimental protocols to reliably assess IDS performance.

7.5. Limitations and Challenges of the Experimental Setup

While effective in comparing different IDS solutions, our experimental setup faced certain limitations. Notably, the absence of a control group, typically essential for baseline comparisons, might limit our ability to assess the absolute effectiveness of each IDS. However, in our context of comparative analysis, this limitation was mitigated as our focus was primarily on contrasting the performance of various systems.

The limited number of test runs could impact the experiment's robustness, potentially affecting our findings' generalizability. This is further argued by the fluctuation in alert numbers shown in Table 6. Nevertheless, these runs provided sufficient preliminary analysis, particularly given the controlled, hardware in the loop environment and the specific scenarios tested.

Changes to the testlab architecture shortly before the experiments may have influenced the tuning and effectiveness of the IDS. While not ideal, this situation presented a realistic scenario of IDSs adapting to evolving environments, offering valuable insights into their flexibility.

Lastly, the lack of a fully realized SCADA HMI in our setup might limit the applicability of our findings to real-world scenarios. However, this also allowed for a more focused examination of IDS performance in a controlled setting, which is crucial for the initial evaluation stages.

8. Conclusions and Further Work

Our experiments critically evaluated various IDS types in a realistic substation environment, including Network and Host-based IDSs and ICS-aware systems, revealing crucial insights into their effectiveness against cyber-attacks in ICS. This evaluation highlighted the strengths and weaknesses of both commercial and open-source IDSs, guiding their implementation in ICS. Surprisingly, IT-related attacks (Attack IDs 3 and 4) were not detected by IT-specific IDSs, indicating that these systems might only effectively identify IT-related ICS attacks with significant tuning.

Our primary aim was to determine the capability of ICS-specific IDSs in detecting tailor-made attacks on industrial control systems. However, the changing testlab environment hindered a conclusive assessment due to unstable baselines critical for behavior-based IDSs. In our study, the Omicron Stationguard, a rule-based IDS, showed the most consistent detection of tailor-made attacks in our testlab. However, it is important to note that the variable nature of the testlab may have influenced these results. Further research in a more stable setting is essential to verify these findings and better understand anomaly-based IDS effectiveness in industrial control systems.

Additionally, the creation of valid and reliable evaluation metrics for IDS performance faced challenges due to the dynamic nature of the testlab, the absence of a complete SCADA HMI, and limitations in test runs and time. These factors emphasize the complexities in replicating operational ICS conditions and assessing IDS effectiveness in such environments. Our study's adherence to a standardized protocol, building on the gaps identified by Giraldo et al. [28], enhances the reliability and applicability of our results, setting a foundation for future research to explore further and improve IDS performance in ICS.

We suggest broadening the range of IDS testing in ICS environments to better understand diverse system performances. Establishing a more stable and controlled testlab environment is crucial for enhancing the reliability of IDS evaluations, especially for behavior-based systems. Adding a complete SCADA HMI to our testlab will more accurately mimic operational conditions. We also recommend increasing the duration and number of test runs to thoroughly assess IDS effectiveness against a wider array of sophisticated and novel cyber threats.

Author Contributions: Conceptualization, J.-M.S. and S.H.H.; methodology, J.-M.S. and S.H.H.; software, P.K. and L.E.; formal analysis, J.-M.S.; investigation, J.-M.S., P.K., L.E. and S.H.H.; data curation, J.-M.S.; writing—original draft preparation, J.-M.S., S.H.H. and P.K.; writing—review and editing, J.-M.S., J.M.H., S.H.H., L.E. and P.K. All authors have read and agreed to the published version of the manuscript.

Funding: This work has received funding from the Research Council of Norway through the CybWin (Cybersecurity Platform for Assessment and Training for Critical Infrastructures-Legacy to Digital twin) project with project no. 287808. This work has received funding from the Norwegian Water Resources and Energy directorate through the efficient cybersecurity measures for industrial control systems project with project no. 80166.

Data Availability Statement: The data are not publicly available due to the sensitive and critical nature of the data.

Acknowledgments: We extend our heartfelt gratitude to the Institute for Energy Technology (IFE), Halden, for their invaluable contribution to the success of our experiments. Their commitment and expertise in managing the lab environment played a pivotal role in the execution and success of our research. Their support and dedication were instrumental in facilitating the comprehensive testing and analysis that form the core of our study. We sincerely appreciate their collaboration and their integral role in advancing our research objectives.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

ARP	Address Resolution Protocol
ASDU	Application Service Data Unit
COTS	Commercial-off-the-shelf
CR	Classification Rate
DS	Digital Substation
DR	Detection Rate
DNP3	Distributed Network Protocol 3
FN	False Negative
FP	False Positive
HIL	Hardware in the Loop
HMI	Human–Machine Interface
ICS	Industrial Control System
IED	Intelligent Electronic Device
IDS	Intrusion Detection System
ISMS	Information Security Management System
IT	Information Technology
OSI	Open Systems Interconnection
OT	Operational Technology
PLC	Programmable Logic Controller
RTU	Remote Terminal Unit
SIEM	Security Information Event Management
SCADA	Supervisory Control and Data Acquisition
TCP/IP	Transmission Control Protocol/Internet Protocol
TN	Trune Negative
TP	True Positive

References

1. IEEESpectrum. The Real Story of STUXNET. Available online: https://spectrum.ieee.org/the-real-story-of-stuxnet (accessed on 5 October 2022).
2. Chlela, M. Cyber Security Enhancement Against Cyber-Attacks On Microgrid Controllers. Ph.D. Thesis, McGill University Montréal, Montréal, QC, Canada. Available online: https://escholarship.mcgill.ca/concern/theses/1c18dh978 (accessed on 5 October 2022).
3. Fianance News. Hackers Attacked Prykarpattiaoblenerho, De-Energizing Half of the Region for 6 h. Available online: http://news.finance.ua/ua/news/-/366136/hakery-atakuvaly-prykarpattyaoblenergo-znestrumyvshy-polovynu-regionu-na-6-godyn (accessed on 5 October 2022).
4. SANS Blog. Confirmation of a Coordinated Attack on the Ukrainian Power Grid. Available online: https://ics.sans.org/blog/2016/01/09/confirmation-of-a-coordinated-attack-on-the-ukrainian-power-grid (accessed on 5 October 2022).
5. Cybersecurity & Infrastructure Security Agency. Cyber-Attack Against Ukrainian Critical Infrastructure. Available online: https://ics-cert.us-cert.gov/alerts/IR-ALERT-H-16-056-01 (accessed on 10 October 2022).
6. Whitehead, D.E.; Owens, K.; Gammel, D.; Smith, J. Ukraine cyber-induced power outage: Analysis and practical mitigation strategies. In Proceedings of the 2017 70th Annual Conference for Protective Relay Engineers (CPRE), College Station, TX, USA, 3–6 April 2017 ; pp. 1–8. ISSN: 2474-9753. [CrossRef]
7. Zetter, K. Inside the Cunning, Unprecedented Hack of Ukraine's Power Grid. Available online: https://www.wired.com/2016/03/inside-cunning-unprecedented-hack-ukraines-power-grid/ (accessed on 10 January 2022).

8. Bindra, A. Securing the power grid: Protecting smart grids and connected power systems from cyberattacks. *IEEE Power Electron. Mag.* **2017**, *4*, 20–27. [CrossRef]
9. Slowik, J. *Crashoverride: Reassessing the 2016 Ukraine Electric Power Event as a Protection-Focused Attack*; Dragos, Inc.: Hanover, MD, USA, 2019.
10. REUTERS. Ukraine's Power Outage Was a Cyber Attack: Ukrenergo. Available online: https://www.reuters.com/article/us-ukraine-cyber-attack-energy-idUSKBN1521BA, (accessed on 10 October 2022).
11. DRAGOS. CRASHOVERRIDE Analysis of the Threat to Electric Grid Operations. Available online: https://www.dragos.com/wp-content/uploads/CrashOverride-01.pdf (accessed on 10 October 2022).
12. Kozak, P.; Klaban, I.; Šlajs, T. Industroyer cyber-attacks on Ukraine's critical infrastructure. In Proceedings of the 2023 International Conference on Military Technologies (ICMT), Brno, Czech Republic, 23–26 May 2023; pp. 1–6. [CrossRef]
13. Knapp, E.D.; Langill, J.T. *Industrial Network Security, Second Edition: Securing Critical Infrastructure Networks for Smart Grid, SCADA, and Other Industrial Control Systems*, 2nd ed.; Syngress: Rockland, MA, USA, 2014. [CrossRef]
14. *IEC 62443 Series*; Industrial Communication Networks—Network and System Security. IEC: Geneva, Switzerland, 2023.
15. Hauet, J.P.; Bock, P.; Foley, R.; Françoise, R. Ukrainian Power Grids Cyberattack. In *Special Section: Ukrainian Power Grids Cyberattack*; International Society of Automation (ISA): Research Triangle Park, NC, USA, 2017. Available online: https://www.isa.org/intech-home/2017/march-april/features/ukrainian-power-grids-cyberattack (accessed on 15 February 2023).
16. Storm, J.M.; Hagen, J.; Toftegaard, Ø.A.A. A Survey of Using Process Data and Features of Industrial Control Systems in Intrusion Detection. In Proceedings of the 2021 IEEE International Conference on Big Data (Big Data), Orlando, FL, USA, 15–18 December 2021; pp. 2170–2177. [CrossRef]
17. Stouffer, K.; Lightman, S.; Pillitteri, V.; Abrams, M.; Hahn, A. *Guide to Industrial Control Systems (ICS) Security*; Technical Report NIST Special Publication (SP) 800-82 Rev. 2; National Institute of Standards and Technology: Gaithersburg, MD, USA, 2015. [CrossRef]
18. Institute of Electrical and Electronics Engineers. *IEEE Standard Dictionary of Electrical and Electronics Terms*, 3rd ed.; IEEE: New York, NY, USA, 1984; ISBN 0471807877.
19. *IEC 60870-5-104*; Telecontrol Equipment and Systems—Part 5-104: Transmission Protocols—Network Access for IEC 60870-5-101 Using Standard Transport Profiles. IEC: Geneva, Switzerland, 2006.
20. *IEC 60870-5-101*; Telecontrol Equipment and Systems—Part 5-101: Transmissionprotocols—Companion Standard for Basic Telecontrol Tasks. IEC: Geneva, Switzerland, 2003.
21. Barrett, M. *Framework for Improving Critical Infrastructure Cybersecurity Version 1.1*; National Institute of Standards and Technology: Gaithersburg, MD, USA, 2018. [CrossRef].
22. Scarfone, K.; Mell, P. *Guide to Intrusion Detection and Prevention Systems (IDPS)*; Technical Report NIST Special Publication (SP) 800-94; National Institute of Standards and Technology: Gaithersburg, MD, USA, 2007. [CrossRef]
23. Kozushko, H. Intrusion detection: Host-based and network-based intrusion detection systems. *Indep. Study* **2003**, *11*, 1–23.
24. Kemmerer, R.; Vigna, G. Intrusion detection: A brief history and overview. *Computer* **2002**, *35*, supl27–supl30. [CrossRef]
25. Pfleeger, C.P. *Security in Computing*, 4th ed.; Pearson: Upper Saddle River, NJ, USA, 2015; ISBN 9780134085050.
26. Knapp, E.D.; Samani, R. *Applied Cyber Security and the Smart Grid: Implementing Security Controls into the Modern Power Infrastructure*, 1st ed.; Elsevier/Syngress: Amsterdam, The Netherlands, 2013; ISBN 1-299-40881-8. [CrossRef]
27. *IEC 62443-2-1*; Industrial Communication Networks—Network and System Security—Part 2-1: Establishing an Industrial Automation and Control System Security Program. IEC: Geneva, Switzerland, 2010.
28. Giraldo, J.; Urbina, D.; Cardenas, A.; Valente, J.; Faisal, M.; Ruths, J.; Tippenhauer, N.O.; Sandberg, H.; Candell, R. A Survey of Physics-Based Attack Detection in Cyber-Physical Systems. *ACM Comput. Surv.* **2018**, *51*, 4. [CrossRef] [PubMed]
29. Oman, P.; Phillips, M. Intrusion detection and event monitoring in SCADA networks. In Proceedings of the 1st Annual IFIP International Conference on Critical Infrastructure Protection, Hanover, NH, USA, 19–21 March 2007; Goetz, E., Shenoi, S., Eds.; IFIPAICT; Springer: Boston, MA, USA, 2008; Volume 253, pp. 161–173. [CrossRef]
30. Carcano, A.; Fovino, I.N.; Masera, M.; Trombetta, A. State-Based Network Intrusion Detection Systems for SCADA Protocols: A Proof of Concept. In Proceedings of the 4th International Workshop on Critical Information Infrastructures Security, Fraunhofer Inst Intelligent Anal & Informat Syst, Bonn, Germany, 30 September–2 October 2009; Lecture Notes in Computer Science ; Rome, E., Bloomfield, R., Eds.; Springer: Berlin/Heidelberger, Germany, 2010; Volume 6027, pp. 138–150. [CrossRef]
31. Lin, H.; Slagell, A.; Di Martino, C.; Kalbarczyk, Z.; Iyer, R. Adapting bro into SCADA: Building a specification-based intrusion detection system for the DNP3 protocol. In Proceedings of the Eighth Annual Cyber Security and Information Intelligence Research Workshop, Oak Ridge, TN, USA, 8–10 January; Association for Computing Machinery: New York, NY, USA, 2013. [CrossRef]
32. The Bro Project. Bro Network Security Monitor. Available online: http://bro-ids.org (accessed on 10 January 2012).
33. Yang, Y.; McLaughlin, K.; Sezer, S.; Littler, T.; Im, E.G.; Pranggono, B.; Wang, H.F. Multiattribute SCADA-Specific Intrusion Detection System for Power Networks. *IEEE Trans. Power Deliv.* **2014**, *29*, 1092–1102. [CrossRef]
34. Cruz, T.; Rosa, L.; Proenca, J.; Maglaras, L.; Aubigny, M.; Lev, L.; Jiang, J.; Simoes, P. A Cybersecurity Detection Framework for Supervisory Control and Data Acquisition Systems. *IEEE Trans. Ind. Inform.* **2016**, *12*, 2236–2246. [CrossRef]
35. CockpitCI. CockpitCI FP7 Project (FP7-SEC-2011-1 285647). Available online: http://cockpitci.eu (accessed on 10 January 2012).

36. Singh, V.K.; Ebrahem, H.; Govindarasu, M. Security Evaluation of Two Intrusion Detection Systems in Smart Grid SCADA Environment. In Proceedings of the 2018 North American Power Symposium (NAPS), Fargo, ND, USA, 9–11 September 2018; IEEE: New York, NY, USA, 2018. [CrossRef]
37. Valli, C. SCADA Forensics with Snort IDS. In Proceedings of the WORLDCOMP, Security and Management, Las Vegas, NV, USA, 13–16 July 2009.
38. Kitchenham, B.A.; Pfleeger, S.L.; Pickard, L.M.; Jones, P.W.; Hoaglin, D.C.; El Emam, K.; Rosenberg, J. Preliminary guidelines for empirical research in software engineering. *IEEE Trans. Softw. Eng.* **2002**, *28*, 721–734. [CrossRef]
39. Jedlitschka, A.; Ciolkowski, M.; Pfahl, D. Reporting experiments in software engineering. In *Guide to Advanced Empirical Software Engineering*; Shull, F., Singer, J., Sjøberg, D.I.K., Eds.; Springer: London, UK, 2008; pp. 201–228. [CrossRef]
40. Wohlin, C.; Runeson, P.; Höst, M.; Ohlsson, M.C.; Regnell, B.; Wesslén, A. *Experimentation in Software Engineering*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2012. [CrossRef]
41. Jørgensen, P.A.; Waltoft-Olsen, A.; Houmb, S.H.; Toppe, A.L.; Soltvedt, T.G.; Muggerud, H.K. Building a Hardware-in-the-Loop (HiL) Digital Energy Station Infrastructure for Cyber Operation Resiliency Testing. In Proceedings of the 3rd International Workshop on Engineering and Cybersecurity of Critical Systems (EnCyCriS'22), Online, 16 May 2022; Association for Computing Machinery: New York, NY, USA, 2022. [CrossRef]
42. *IEC 61850 Series*; Communication Networks and Systems for Power Utility Automation. IEC: Geneva, Switzerland, 2023.
43. Erdodi, L.; Kaliyar, P.; Houmb, S.H.; Akbarzadeh, A.; Waltoft-Olsen, A.J. Attacking Power Grid Substations: An Experiment Demonstrating How to Attack the SCADA Protocol IEC 60870-5-104. In Proceedings of the 17th International Conference on Availability, Reliability and Security (ARES 2022), Vienna, Austria, 23–26 August 2022; Association for Computing Machinery: New York, NY, USA, 2022. [CrossRef]
44. Kumar, G. Evaluation metrics for intrusion detection systems-a study. *Int. J. Comput. Sci. Mob. Appl.* **2014**, *2*, 11–17.
45. Shah, S.A.R.; Issac, B. Performance comparison of intrusion detection systems and application of machine learning to Snort system. *Future Gener. Comput. Syst.* **2018**, *80*, 157–170. [CrossRef]
46. Wong, K.; Dillabaugh, C.; Seddigh, N.; Nandy, B. Enhancing Suricata intrusion detection system for cyber security in SCADA networks. In Proceedings of the 2017 IEEE 30th Canadian Conference on Electrical and Computer Engineering (CCECE), Windsor, ON, Canada, 30 April–3 May 2017; pp. 1–5. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.