CoFacS – Simulating a Complete Factory to Study the Security of Interconnected Production

Stefan Lenz*, David Schachtschneider*, Simon Jonas*, Liam Tirpitz‡, Sandra Geisler‡, Martin Henze*§
*_Security and Privacy in Industrial Cooperation_, RWTH Aachen University, Germany
‡_Data Stream Management and Analysis_, RWTH Aachen University, Germany
§_Cyber Analysis & Defense_, Fraunhofer FKIE, Germany
{lenz, schachtschneider, jonas, henze}@spice.rwth-aachen.de · {tirpitz, geisler}@dbis.rwth-aachen.de

Abstract—**While the digitization of industrial factories provides tremendous improvements for the production of goods, it also renders such systems vulnerable to serious cyber-attacks. To research, test, and validate security measures protecting industrial networks against such cyber-attacks, the security community relies on testbeds to simulate industrial systems, as utilizing live systems endangers costly components or even human life. However, existing testbeds focus on individual parts of typically complex production lines in industrial factories. Consequently, the impact of cyber-attacks on industrial networks as well as the effectiveness of countermeasures cannot be evaluated in an end-to-end manner. To address this issue and facilitate research on novel security mechanisms, we present CoFacS, the first COmplete FACtory Simulation that replicates an entire production line and affords the integration of real-life industrial applications. To showcase that CoFacS accurately captures real-world behavior, we validate it against a physical model factory widely used in security research. We show that CoFacS has a maximum deviation of 0.11% to the physical reference, which enables us to study the impact of physical attacks or network-based cyber-attacks. Moreover, we highlight how CoFacS enables security research through two cases studies surrounding attack detection and the resilience of 5G-based industrial communication against jamming.**

Index Terms—**Industrial control systems, security testbed, factory simulation**

I. Introduction

The digitization of factories is the foundation for efficient and adaptable production of goods [1]. However, the resulting increase in connectivity exposes complex industrial control systems with numerous interacting components to cyberattacks [2]. Attacks targeting factories and production lines can, e.g., halt the production, damage components, or even physically harm workers, posing a great risk to companies around the globe.

However, researching, testing, and evaluating novel methods to secure factories and production lines cannot be performed on actual live systems, due to strict availability and safety requirements [3]. Therefore, the security community has to rely on testbeds [3]. Although *physical testbeds* (e.g., [4]) provide the highest degree of realism, they are often hard to access, inflexible, and expensive to build [3]. Moreover, it is impractical to examine attacks that might irreversibly damage expensive equipment of the physical testbed. In contrast, *virtual testbeds* provide a high degree of flexibility and accessibility, but must be validated against a real-life system to provide sufficiently realistic results. As recent surveys show [3], there exist a variety of virtual testbeds of industrial factories for security research (e.g., [5]–[7]). However, all publicly available testbeds that specifically focus on factories capture only a single step of typically involved production lines [6]–[9]. Thus, these testbeds do not adequately capture the complexity of modern factories, including the interaction of different production systems, such as the transferal from one production step to the next, their timings, or additional physical properties.

To address this gap and thus lay the foundation for comprehensively studying the security of factories, we propose *CoFacS*; a simulation of an industrial factory that covers a complete production line, from arrival of raw material in the factory to the completion of the finished product. To realistically and comprehensively cover the properties of a real factory, CoFacS not only accurately captures the underlying physical processes but also simulates all control components such as Programmable Logic Controllers (PLCs), a Supervisory Control and Data Acquisition (SCADA) system, as well as the corresponding industrial network.

To ensure accurate behavior of all testbed components, we provide a comprehensive validation of CoFacS. We utilize the Fischertechnik Learning Factory 4.0 [10] as *physical* reference, which provides a lab-scale replica of a complete production process. We choose this reference model, since it is already successfully utilized in other research [11]–[14] and generally available to other researchers for reproducibility.

Contributions. With the goal to enable research, testing, and evaluation of novel mechanisms to strengthen the security of industrial factories, we present the following contributions.

1) We provide CoFacS, a comprehensive, freely-available, virtual testbed of an industrial factory, simulating physical processes, PLC logic, network communication, and SCADA application of a complete production line (Sec. III).
2) We validate CoFacS's accuracy against the Fischertechnik Learning Factory 4.0 [10], a realistic physical factory simulation often used in security research (Sec. IV).
3) We show the usability of CoFacS for security research by evaluating the behavior of the simulation and the physical reference model under different attack scenarios (Sec. V).
4) We further exemplify CoFacS's versatility as an enabler for security research, by studying the resilience of 5G-based production system a wireless production system utilizing a real 5G channel and CoFacS's capability to research attack detection approaches (Sec. VI).

Author's version of a paper accepted for publication in Proceedings of the 2025 IEEE 50th Conference on Local Computer Networks (LCN).

Availability Statement. As a novel virtual factory, we provide full access to the code base of CoFacS[1] and the exemplary attacks, encouraging researchers to use our testbed for their own security research. To enable reproducibility, we also make all results underlying our evaluation available.

II. The Need for Complete Factory Simulation

To motivate the need for complete factory simulation covering all steps of a production process, we first provide a short introduction to Industrial Control Systems (ICSs) and testbeds (Sec. II-A). Subsequently, we analyze the current landscape of testbeds, highlighting the gap in existing virtual testbeds (Sec. II-B).

A. Industrial Control Systems & Testbeds

Industrial Control Systems (ICSs) monitor and control complex physical processes, e.g., in factories, and thus serve as the backbone of digitized production. The components that make up an ICS are typically grouped into field bus, control, and supervisory level [15].

The *field bus level* comprises all devices directly interacting with the physical process, such as sensors and actuators. In the execution of a control loop (i.e., one cycle of the ICS logic), these devices forward the process state to the *control level* and in turn receive commands to interact with the physical process. To this end, controllers (e.g., PLCs) respond to the current process state as captured by sensors according to their control logic and send commands back to actuators. Complex manufacturing processes necessitate the use of multiple sensors and actuators, each demanding prompt responses to ensure the safe control of the underlying physical process. Consequently, ICSs are composed of several PLCs, with each PLC responsible for controlling a specific part of the system, such as an individual manufacturing step. Additionally, the controllers send updates to devices on the *supervisory level*. These supervisory devices include, e.g., a SCADA system, to facilitate direct human interaction to monitor the state of the physical process and perform higher-level control such as which parts to produce.

Since such ICSs which control highly critical processes have strong availability and safety requirements [3], testing of novel security measures cannot be performed on live systems. Thus, researchers rely on replicas of ICSs, which represent industrial processes in physical, virtual (e.g., a digital simulation), or hybrid testbeds as a combination of the two [3]. Although *physical* testbeds allow the most realistic representation of ICSs, they suffer from high costs and inflexibility, due to set up times, which also cause scalability issues. Additionally, physical testbeds often provide high barriers for accessibility. In contrast, *virtual* testbeds provide high accessibility and flexibility, since researchers only need means to execute the testbed (e.g., a simulation) and experiments can be done quickly. However, this abstraction entails the disadvantage of possibly providing less accurate results. Thus, researchers must thoroughly validate virtual testbeds to ensure correct results. Then, validated virtual testbeds allow testing of additional scenarios that would, e.g., be too dangerous to perform in a physical testbed [3].

Consequently, validated virtual testbeds are crucial to enable various security research. For example, simulations enable studying trends of modern industry such as wireless 5G communication. As jamming attacks potentially cause serious damage to physical components (e.g., by rendering the ICS unable to react to emergency scenarios), virtual testbeds allow their execution in a safe environment. Thus, they facilitate the research of new security-trends for safety-reliant production systems. Likewise, validated virtual testbeds with their ability to gather data such as network traffic captures, enable the study of further defense mechanisms such as Intrusion Detection Systems (IDSs). These systems utilize the predictability of ICSs to detect attacks in communication patterns (e.g., timings [16], packet sequences [17]) or the physical process (e.g, [18]).

> *Take Away:* ICSs serve as the backbone of modern digitized production. Due to availability and safety requirements, testing security mechanisms for ICSs is challenging on live-systems. Thus, researchers utilize testbeds to gain insight into such systems. As virtual testbeds provide the highest degree of flexibility and accessibility without damaging physical components, they are a good solution to gain deep insights into ICS.

B. Related Work on ICS Testbeds

Given these benefits of (virtual) testbeds for ICS security research, we now analyze already available testbeds. As a recent survey comprehensively summarizes the landscape of ICS testbeds [3], we specifically focus on research leveraging the Fischertechnik Learning Factory 4.0 [10] and virtual testbeds related to our approach. We summarize these works in Table I based on their *scenario*, the support of *network*, *physical*, *control logic*, and *SCADA* simulation, their *accessibility*, and whether they realize a *complete factory simulation*.

Physical and Hybrid Testbeds. We discuss physical and hybrid testbeds utilizing the Fischertechnik hardware, showing its viability for researching modern production environments. LICSTER [11] is a hybrid testbed to simulate control logic and SCADA application based on individual Fischertechnik components. Gardiner et al. [20] present a physical ICS testbed including the Fischertechnik model factory [10] to research guidelines for building testbeds. They confirm the usefulness of the Fischertechnik factory for setting up new testbeds due to high fidelity. Similarly, Foley et al. [19] apply the Fischertechnik factory to building a hybrid model of a complete production line. Although these works show the benefits of the Fischertechnik factory [10] for security research, they all require a physical model factory, limiting accessibility and flexibility due to costs and potential damage to physical components.

Virtual Testbeds. Focussing on virtual testbeds for ICS and specifically production lines, we observe a variety of research.

Research on virtual testbeds focuses on simulations to safely research the impact of cyber-attacks in *power grids* [5], [21]–[25]. While the physical process of power grids is clearly

[1] We provide code and results under *github.com/RWTH-SPICe/CoFacS*

TABLE I

THE CURRENT LANDSCAPE OF HYBRID AND VIRTUAL ICS TESTBEDS FOR INTERCONNECTED PRODUCTION INCLUDES A VARIETY OF DIFFERENT APPROACHES. WE COMPARE (1) SCENARIO, (2) COMPLETENESS AS A FACTORY SIMULATION, (3) NETWORK EMULATION, (4) PHYSICAL SIMULATION, (5) SCADA SIMULATION, (6) CONTROL LOGIC SIMULATION, AND (7) THE ACCESSIBILITY, HIGHLIGHTING THE NEED FOR A COMPLETE VIRTUAL TESTBED TO ENABLE COMPREHENSIVE SECURITY RESEARCH FOR INTERCONNECTED PRODUCTION.

Type	Name	Scenario	Complete Factory Simulation	Network Emulation	Physical Simulation	SCADA Simulation	Control Logic Simulation	Accessibility
hybr.	LICSTER [11]	Factory	◐	●	○	●	●	●
hybr.	Foley et al. [19]	Factory	?	●	○	◐	○	●
phys.	Gardiner et al. [20]	Factory	◐	◐	◐	●	○	○
virtual	Davis et al. [21]	Power Grid	—	●	●	●	●	●
	Koganti et al. [22]	Power Grid	—	○	○	●	●	○
	RICS-el [23]	Power Grid	—	●	●	●	●	○
	Singh et al. [24]	Power Grid	—	?	●	●	●	○
	TasSCS [25]	Power Grid	—	●	●	●	●	○
	Wattson [5]	Power Grid	—	●	●	●	●	●
	Alves et al. [26]	Gas Pipeline	—	●	●	●	●	○
	Morris et al. [27]	Gas Pipeline	—	●	●	●	●	○
	SCADAVT-A [28]	Water Pipeline	—	●	●	●	●	○
	DVCP [8]	Chemical Plant	○	◐	●	●	●	●
	VTET [9]	Chemical Plant	○	●	●	●	●	○
	GRIFCS [6]	Chemical Plant	○	●	●	●	●	●
	ICSSIM [7]	Factory	○	●	●	●	●	●
	Sala et al. [13]	Factory	◐	?	●	●	●	○
	CoFacS (this paper)	*Factory*	●	●	●	●	●	●

○: No support/access ◐: Supported, but not provided/access limited ●: Provided/full access ?: Unclear

distinct from modern multi-step production lines in factories, those works highlight important aspects of virtual testbeds: authentically capturing the physical process [24], replicating realistic traffic patterns and network architectures [22], [25], fully capturing ICS assets [23], and thoroughly validating against a physical reference [5]. To research *gas pipeline* security, two virtual testbeds replicate lab-scale physical models to create authentic representations [4], [26] demonstrating the advantages w.r.t. validation. Likewise, a virtual testbed simulating a *water pipeline* [28] highlights the need for virtual simulations to safely research attack scenarios with potentially devastating consequences. Similarly, simulations of a *chemical plant* [6], [8], [9] highlight this necessity with catastrophic outcomes such as an exploding reactor.

Moving towards *factory* simulation, ICSSIM [7] replicates a bottle filling plant to simulate a "robust" process that can quickly recover from a fault state enabling researchers to test several attack scenarios without waiting for the process to recover. However, by focussing on only one process step, ICSSIM misses the interaction between different components. Finally, to test the ability of students to create a digital shadow of a complex production line, Sala et al. [13] utilize the Fischertechnik Learning Factory 4.0 [10]. Not focussing on security, they neither create an authentic simulation of a complete factory including network communication nor evaluate the authenticity and accuracy of their simulation. Still, they show the general suitability of the Fischertechnik factory as a baseline to create a complete factory simulation.

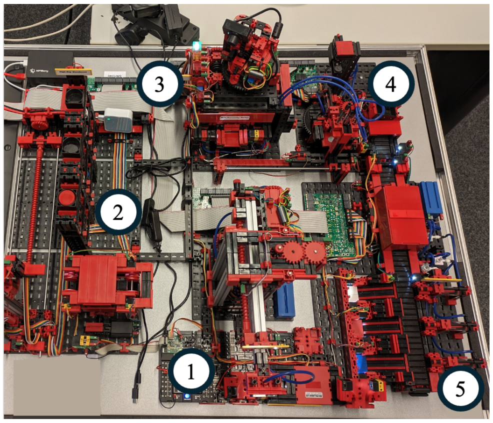

Fig. 1. The Fischertechnik Learning Factory 4.0 is a lab-scale replica of a complete production line covering (1) a vacuum gripper, (2) a high-bay warehouse, (3) a furnace, (4) a mill, and (5) a sorting station. It serves as a widely-used reference architecture for (security) research [11], [13], [19], [20].

Take Away: There is no virtual ICS testbed covering a complete production process *and* fully replicating all aspects of modern digitized production (i.e., physical process, control logic, SCADA, and network). To address this gap, we propose CoFacS, a *complete* virtual factory to facilitate the study of interconnected production.

III. THE COFACS COMPLETE FACTORY SIMULATION

To develop, test, and evaluate novel security measures for interconnected production, we propose CoFacS, the first simulation of a complete factory. To facilitate accessibility and reproducibility [29], as well as to validate our simulation, we utilize the Fischertechnik Learning Factory 4.0 [10] as a physical reference. This factory, as shown in Fig. 1, encompasses a complete production process from the delivery of raw material to completion of the product, including process control logic, network communication, and a SCADA system as a table-sized physical model. Additionally, the Fischertechnik factory can be bought as a ready-to-use package or as individual components [11] without requiring intensive setup. Therefore, we choose this model as a physical reference for our simulation, enabling reproducibility but also flexibility to place individual, physical components in a hybrid factory testbed.

As a virtual model, CoFacS provides an accurate simulation of all physical components, the process logic, and SCADA monitoring of the factory. To serve its purpose as a security testbed, CoFacS also precisely emulates the industrial network, allowing for the integration for real applications and attack tools in this network. Furthermore, CoFacS allows extracting artifacts for further analysis such as the sensor and actuator states or recordings of the network traffic.

To introduce our testbed, we first describe the individual components and the production process of the Fischertechnik Learning Factory 4.0 reference factory (Sec. III-A), before we detail the technical realization of CoFacS (Sec. III-B).

A. The Simulated Physical Production Line

The modeled process of the Fischertechnik Learning Factory 4.0 [10] consists of five different components: a *vacuum gripper (VG)*, a *high-bay warehouse*, a *multi-processing unit (MPU)*, comprising a *furnace* and a *mill*, and a *sorting station* (Fig. 1). The process itself mimics a complete production line by using small "cylinders" (i.e., miniature hockey pucks) in red, white, or blue as the material and end-product. These cylinders then pass each component, replicating the arrival, storage, manufacturing and finally sorting of a product.

Step 1 – Material Input & Vacuum Gripper. Once a new ("raw") cylinder arrives at factory, the *VG* transports the material to the high-bay warehouse to store until needed to complete an order. To do so, the *VG* uses its arm with a suction mechanism to pick up new cylinders (Fig. 1-1). The VG senses the rotation angle, horizontal, and vertical position of the rotating arm, and transmits these values to the PLC. The PLC controls the electric motor of the VG, to reach pre-defined destinations (e.g., delivery and pickup station).

Step 2 – High-Bay Warehouse. The *high-bay warehouse* (Fig. 1-2) stores raw material (i.e., recently delivered cylinders) in a high-bay rack until they are needed for the production process (i.e., ordered through the SCADA application). To this end, a cantilever picks up a cylinder at the drop-off location of the VG and transports it to a free spot in the high-bay rack. The factory keeps track of each stored cylinder using a triple of X and Y coordinates and the color of the cylinder. Once a product order arrives through the SCADA system, the warehouse "unloads" a cylinder in the requested color by performing these actions in reverse order, placing the cylinder at the VG's pick-up location.

Step 3 & 4 – Multi-Processing Unit. The multi-processing unit (MPU) replicates the production steps of a production line with two parts, a *furnace* and a *mill*, producing a "finished" product ready for delivery. The first part of the MPU, the *furnace* (Fig. 1-3), bakes the raw cylinder. To this end, the VG transports a cylinder from the warehouse to a platform in front of the furnace. Then, the platform transports the cylinder into the chamber of the furnace, where the cylinder resides for a per-order customizable period of time. The duration of the firing process can be customized from the SCADA interface. To indicate the firing process of a cylinder, the furnace controls an LED light that is activated for the particular duration. Once this process is complete, the cylinder leaves the firing chamber and the MPU transfers the cylinder to the *mill* (Fig. 1-4), the second and final step of the production process. This step simulates finishing the product in, e.g., a CNC machine. To replicate this behavior, the physical reference factory actuates an electrical motor that turns a gear wheel above the current cylinder. Finally, a piston pushes the now finished cylinder onto a conveyor of the last component, the *sorting station*.

Step 5 – Sorting & Delivery. After being processed in the MPU, the finished cylinder arrives on the conveyor belt of the *sorting station* (Fig. 1-5). The sorting stations purpose is to organize finished cylinders based on their colors for the delivery. Thus, the cylinder first passes through a light-barrier, which activates a door to the color-sensing enclosure. Within this light-proof enclosure, an LED creates a bright flash. Then, a sensor compares the reflection of the cylinder to the baseline thus receiving a distinct value for each of the three colors (i.e, red, white, or blue). Following the reading, the cylinder passes through another light-barrier, this time activating a timer. Finally, based on the timer, a piston activates to sort the cylinders into bays, with each bay designated for a specific color, thus completing the production process.

Take Away: The considered production process consists of five distinct components, each with an own purpose, requiring different control logic, sensors, actors, and timings, adaptable to specific demands of individual orders. Additionally, to produce a finished product, the components must correctly work together. To be able to capture the interplay of these components, CoFacS has to thoroughly simulate the entire production process.

B. Realizing a Complete Factory Simulation

To build CoFacS and correctly capture the complexity of the physical reference model, we partition CoFacS into four components: the *physical*, *logical*, and *SCADA simulation*, and *network emulation*, which we introduce in the following. Fig. 2 visualizes the structure, all components, and the network configuration of CoFacS.

Physical Simulation. The physical simulation replicates the physical behavior of the five components (i.e., VG, warehouse, furnace, mill, and sorting station) as well as the cylinders. To this end, we utilize Discrete Event Simulation (DES) to represent the physical state of the factory. Assuming that the current state directly results from a previous state, DES simulates time as discrete steps where an event occurs. We base our choice on the fact that PLCs also operate on polling cycles and thus control the physical state of the process in fixed time intervals, which fits the simulation behavior of DES nicely. Furthermore, this choice enables CoFacS to save computational resources, allowing execution in light-weight environments.

More specifically, we simulate the physical properties of the production line components using SimPy [30], a Python-based DES framework, capturing all necessary variables to execute each component. We create one process for each component within SimPy's environment to recreate the respective physical behavior. To keep track of the cylinders that are present in the factory, each process monitors a list of which cylinders to handle. Additionally, processes may trigger event for other components, such as the VG triggering a light barrier. We recreate the VG's movement using three variables, for the horizontal, vertical positions, and for the rotation angle. Additionally, we represent the cylinders as files within CoFacS's codebase, which eases their creation and deletion, thus enabling physical attacks, such as forcibly removing cylinders during production. To parameterize our simulation, we recorded the behavior of the physical testbed during complete production cycles, fixing color, storage position, and processing times.

Logical Simulation. The logical simulation layer, i.e., the simulation of the control logic, consists of five PLCs, one for each of the production components. With this design choice, we allow for adjustments in the control logic, whether simulating benign adaptation of the process or acting as an attacker. Additionally, it enables scalability of the simulation, by allowing the addition of more components. We implement control programs for all five PLCs in the IEC-61131-3 compliant OpenPLC framework [31] and the structured text programming language. Furthermore, OpenPLC enables users to exchange the PLC logic with any IEC-61131-3 compliant code allowing for testing of real-world logic in a safe environment. To accurately control the physical hardware, we set the default PLC cycle time to that of the physical reference (i.e., 20 ms). Additionally, CoFacS includes OpenPLC's web-interface, which enables adaptation of the PLC logic during live operation of the virtual factory.

SCADA Simulation. The SCADA application performs higher-level supervision and control of our virtual factory by forwarding order-specific parameters (e.g., the firing time of an individual cylinder) to the PLCs. To set such parameters, users can directly interact with the SCADA application, ordering new products, adapting the workflow, or monitoring the status of the factory. We implement this application using Node-RED [32], an open-source SCADA system, which is also used in real-life industrial applications and in the Fischertechnik Learning Factory 4.0 [10]. Furthermore, Node-RED provides a web-interface enabling detailed real-time monitoring, interactive ordering, and automation of the virtual factory.

Network Emulation. As a part of an accurate depiction of a complete factory, CoFacS provides an emulation of the network communication. We emulate connections between the SCADA and the PLCs (connected via a network switch), and between the PLCs and the physical simulation of their respective production line components (Fig. 2). To achieve authentic traffic patterns, we utilize ModbusTCP [33], the most commonly used communication protocol for industrial networks [3]. To emulate the communication channel, we use Containernet [34], a "containerized" fork of the Mininet [35] simulator, which enables the emulation of crucial properties for industrial deployments, e.g., latency [36] or bandwidth [37], which we utilize to emulate physical distance between the SCADA and PLC layers. Additionally, using containers enables the flexible integration of additional industrial components and applications if needed. Furthermore, this choice allows replacing the emulated network with real-world network components by deploying each container on a different physical host. We exemplify how this can be utilized in Section VI-B.

Take Away: We present CoFacS a *complete* virtual factory, consisting of all components (i.e., physical process, control logic and SCADA monitoring) also present in modern interconnected production lines. To achieve high realism, CoFacS utilizes industrial protocols and frameworks as found in real-world deployments.

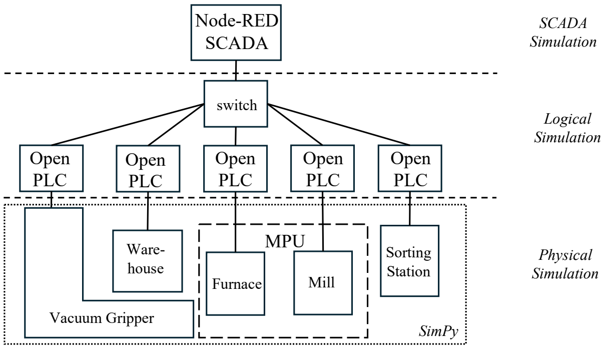

Fig. 2. CoFacS is a complete factory simulation replicating physical components, PLC control logic with IEC-61131-3 compliant OpenPLC code, a fully usable Node-RED SCADA application, and an authentic network emulation.

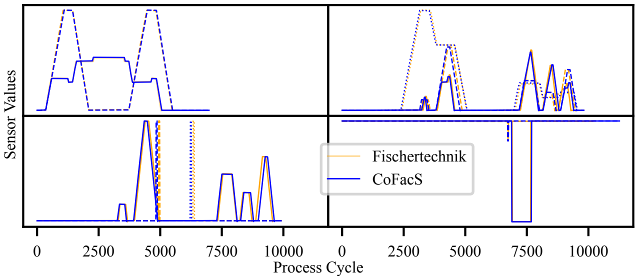

Fig. 3. We showcase the correctness of CoFacS (blue) by comparing the behavior during a complete production cycle with the Fischertechnik factory (orange). The sensor values (different line styles) of all components, i.e., the warehouse (upper left), the VG (upper right), the MPU (lower left), and the sorting station (lower right), substantially overlap during the process execution, showing correct behavior of CoFacS.

IV. Testbed Accuracy & Resource Requirements

To verify the accuracy of CoFacS, we compare its behavior to a physical Fischertechnik Learning Factory 4.0 [10]. Concretely, we compare all components during a complete production process and study the VG as the most complex component of the factory in more detail. We evaluate on fresh data collected *after* CoFacS has been parameterized to avoid overfitting.

Evaluation Setup. To demonstrate that CoFacS is a lightweight testbed, we conduct all experiments on a resource-constrained Raspberry Pi 5 equipped with a 2.4 GHz quad-core ARM CPU and 4 GB of RAM. We chose this hardware due to its low cost and accessibility, making CoFacS available to researchers and students worldwide without requiring powerful and expensive equipment. Furthermore, the limited resources provide an ideal foundation for assessing the resource requirements of our simulation.

Example Production Process. We demonstrate CoFacS's correct behavior by conducting an example production process. We simulate the arrival of a red cylinder, which is stored in the $(1, 1)$ position of the warehouse and then processed for 1 s in the MPU before being passed from the sorting station to the delivery bay. Fig. 3 shows the respective sensor values (distinguished by line style) over time (measured in process cycles) for each component of the Fischertechnik factory (orange) and CoFacS (blue). We observe major overlap of all curves, indicating nearly identical behavior of both factories. Most notably, the shape of all curves is identical and the only deviations result from timings, e.g., the "shift" along the x-Axis for the VG's (Fig. 3 upper right). These results indicate that CoFacS mimics the physical reference factory accurately.

Detailed Validation. To further illustrate that CoFacS accurately represents the Fischertechnik factory, we comprehensively analyze the behavior of the VG as the most complex component. We measure the VG's vertical (Fig. 4 top) and horizontal (Fig. 4 middle) position and the rotational angle (Fig. 4 bottom) during four runs of the physical reference factory and compare them to four simulation runs of CoFacS. We divide our measurements into two scenarios: *arrive*, where the VG transports a cylinder from material delivery to the warehouse, and *order*, where the VG transports a cylinder from the warehouse to the MPU. Fig. 4 visualizes the minimum and maximum values of the sensors for the physical (orange) and virtual (blue) factory. Again, we observe a substantial overlap of the sensor values. To examine the maximum deviation of the individual sensors' value, corresponding to a deviating movement of the VG, we measure the relative distance between the curves from the physical Fischertechnik factory and CoFacS. Beginning with the vertical sensor, we observe a relative deviation of 0.00 % in the arrive and 0.05 % in the order scenario. The horizontal sensor exhibits a maximum deviation of 0.10 % and 0.03 %, and the rotation sensor a maximum deviation of 0.11 % and 0.03 % for the respective scenarios. These results show that CoFacS replicates the behavior of the physical reference factory *very* closely. Therefore, CoFacS serves as an accurate virtual testbed to study attacks without risking damage to real components and provides further insight into the security of interconnected production.

Resource Requirements. By performing all experiments on a Raspberry Pi 5, we already show the rather low resource requirements of CoFacS. During the execution of the simulation, we measure approximately 40 % CPU usage and 1 GB of memory usage. These modest requirements result not only in *scalability* of CoFacS (e.g., by being able to simulate more components on more powerful hardware), but they also provide high *accessibility*, since low-cost hardware suffices to run CoFacS.

> *Take Away:* Our results show that CoFacS achieves a maximum deviation of 0.11 % from the physical reference factory, thus providing a realistic representation of a complex factory environment. As these results were obtained on rather low-end hardware, CoFacS positions itself as an efficient, flexible, accessible and thus widely applicable testbed.

V. Analysis of Exemplary Attacks on CoFacS

After validating CoFacS, we use it to conduct exemplary attacks on the *physical process* to demonstrate that our testbed accurately reflects behavior under anomalous conditions. Additionally, we carry out *network* attacks that cannot be performed on the physical testbed due to the risk of damaging expensive components. These attacks are intended to showcase the versatility of CoFacS as a security testbed. Particularly, by simulating a complete factory, CoFacS enables the study of attack impact on multiple production components.

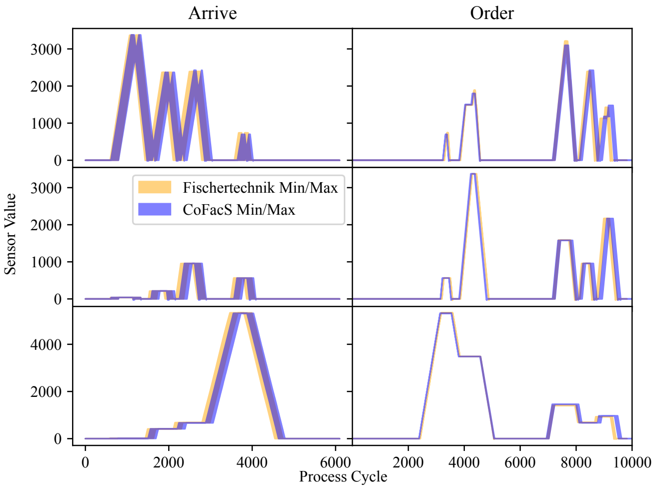

Fig. 4. To validate the behavior of CoFacS, we compare the VG's movement in the *arrive* (left) and *order* (right) scenario. We show the minimum and maximum value for the Fischertechnik Learning Factory 4.0 (orange) and CoFacS (blue) and the *vertical* (top), *horizontal* (middle), and *rotation* (bottom) sensor. We observe a maximum deviation of 0.11 % for all sensors, demonstrating the accuracy of CoFacS.

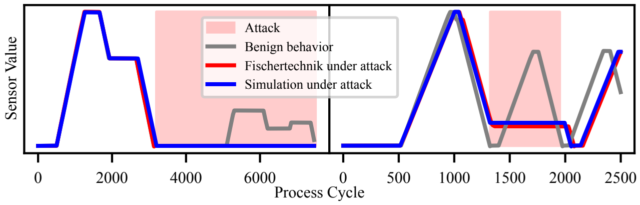

Fig. 5. To validate that CoFacS accurately replicates anomalous behavior from the physical reference, we perform two low-impact physical attacks on both testbeds. To this end, we remove a cylinder during the execution of the MPU (left) and block the vacuum gripper while transporting a cylinder (right).

Physical Attacks. In a first step, we perform two low-impact physical attacks (i.e., attacks that do not damage the equipment) on the Fischertechnik factory *and* CoFacS to verify that our testbed behaves similar to the real system also under anomalous process states. As a first attack, we *remove the cylinder* after it has been processed by the furnace. We realize this attack by physically picking up the cylinder from the delivery bay of the MPU (physical testbed) and by deleting the corresponding file which simulates a cylinder (CoFacS). In Fig. 5 (left), we depict the sensor readings of the MPU's transport system. Under normal behavior (gray) the MPU transports the cylinder from the furnace to the mill, and then pushes the finished product onto the sorting station's conveyor belt, visualized by the two peaks in Fig. 5. In contrast, if the attacker physically removes the cylinder, it cannot be transferred to the sorting station after

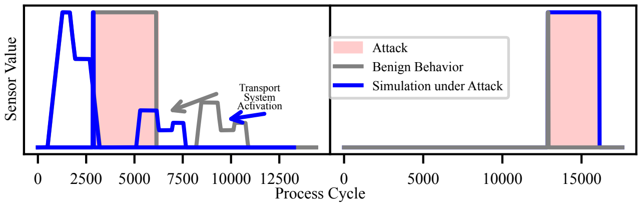

Fig. 6. By performing two command injection attacks on both components of the MPU (i.e., furnace (left) and mill (right)), we verify that CoFacS behaves as expected in scenarios we cannot study in a real setup.

completing the milling step, visualized by the transport system remaining at 0 after the initial peak. Both the physical testbed (red) and CoFacS (blue) exhibit identical behavior.

Similarly, for the second attack, we *physically block the vacuum gripper*, which results in the VG halting and then resuming movement after being released. Fig. 5 (right) shows the reading of the rotation angle of the VG, and we again observe nearly identical behavior for the Fischertechnik factory (red) and CoFacS (blue), which is clearly distinct from normal behavior (gray). The only deviation between the behavior of both testbeds stems from halting the VG at different process cycles in the physical reference model.

Network Attacks. Making use of CoFacS as a virtual testbed, we execute two *command injection* attacks on the components of the MPU which we could not launch against the reference testbed. In this attack, we model a compromised PLC enabling the attacker to change the system behavior at will. To cause maximum damage to the production, the attacker chooses to increase the activation times of furnace and mill. In Fig. 6 (left), we see the impact of this attack through the increased firing time of the furnace. Furthermore, this attack causes a delay in the activation of the MPU's transport system. Similarly, we observe the increased processing time of the mill (Fig. 6 right) compared to CoFacS's benign behavior. Additionally, this attack causes a delayed activation of the piston which pushes the "finished" cylinder to the sorting station. In a real system, these attacks could have drastic consequences such as the destruction of materials or even potential fires.

By conducting these attacks, we demonstrate that CoFacS enables studying handcrafted protocol-level attacks on specific factory components, posing a significant threat to real production systems [38].

Utilizing Existing Attack Tools. To enable the study of security measures against known threats, CoFacS also provides the possibility to execute existing attack tools, such as *nmap* or *Metasploit* [39]. For example, we can launch attacks against the SCADA simulation by utilizing the (currently 331) HTTP exploits present in the Metasploit library over the host machine's localhost interface. These allow us, e.g., to spy for unencrypted login credentials or active scanning attacks, which often serve as the first step when attacking ICSs. Therefore, CoFacS can also promote research of security measures against known attacks, further demonstrating its versatility.

> *Take Away:* Not only does CoFacS precisely replicate the behavior of the physical reference testbed also under attack conditions, but it also enables us to study the impact of attacks that cannot be performed against the real testbed, showcasing CoFacS's versatility. Additionally, CoFacS enables the execution of existing security tools such as Metasploit.

VI. Security Research Enabled by CoFacS

To further demonstrate how CoFacS enables security research of interconnected production, we conduct case studies on intrusion detection (Sec. VI-A) and the resilience of a 5G-enabled factory against jamming attacks (Sec. VI-B). Hence, we not only show CoFacS's broad applicability but also lay the foundation for further research ideas that our testbed enables.

A. Intrusion Detection in Interconnected Production

Intrusion Detection Systems (IDSs) allow for timely detection of ongoing attacks [40], [41]. As threats against modern ICSs not only target the communication but also the physical process (cf. Sec.V), it is crucial to deploy mechanisms to detect anomalies in both dimensions. To show how CoFacS can facilitate such research, we examine the performance of IDSs in threat scenarios targeting these dimensions.

Experimental Setup. To conduct these experiments, we create a dataset consisting of benign behavior and attack data. We capture the benign behavior (i.e., training data for anomaly-based IDSs) by recording CoFacS's network traffic using Wireshark. When gathering the attack data (i.e., a command injection attack and a Modbus scan attack), we label all attack timings to generate an accurate ground truth. We transcribe the Modbus traffic from CoFacS into the protocol-agnostic *IPAL* framework [40] to evaluate various state-of-the-art IDSs on the dataset. More specifically, we deploy two of the *process-aware* SIMPLE detectors [18], which monitor minimum and maximum or consistent timings of process states, as well as the two *communication-based* IDSs Inter-Arrival Time (IaT) [16], which monitors timing consistency of network packets, and DTMC [17], monitoring packet sequences.

Results. Figure 7 visualizes the alerts of the IDSs in reference to the ground truth (top, red bars). All IDSs can detect the injection attacks (Fig. 7, 1 & 2), as the additional packet not only causes a change in the process state (MinMax and SteadyTime), but also sufficient deviation in the network patterns for IaT and DTMC to notice the attack. As expected, the Modbus Scan attack (Fig. 7, 3) impacts communication and both communication-based IDSs (IaT and DTMC) detect this attack. In contrast, IDSs monitoring whether physical process variable stay within bounds (MinMax) cannot detect such an attack. Interestingly, however, while also only focussing on process variables, SteadyTime still detects (albeit delayed) that monitored process registers are updated later due to the attack. These results demonstrate how a complete factory simulation can provide novel insights into the performance

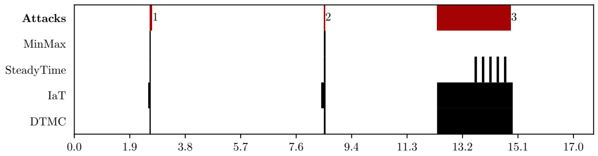

Fig. 7. The process-aware IDSs *MinMax* and *SteadyTime* [18], along with the communication-based IDSs *IaT* [16] and *DTMC* [17], are capable of detecting command injection attacks (1 & 2) against CoFacS. However, not all IDSs reliably detect the Modbus-Scan Attack (3) highlighting the need to capture multiple dimensions of modern production systems which CoFacS enables.

of IDSs. Consequently, this case study highlights the need for a testbed that accurately captures all aspects of a modern factory to adequately study IDS approaches.

B. Resilience of 5G-based Factories Against Jamming Attacks

To match the increasing demands for flexibility and mobility for modern production systems, analyzing the capability of wireless communication in industrial setting is essential [42], [43]. In this regard, we show how CoFacS can enable research on the resilience of 5G-based communication to attacks on the wireless medium in modern factories. To this end, we leverage the adaptability of CoFacS to replace single links in the emulated factory network with *real 5G communication*.

Experimental Setup. We deploy the simulation of the SCADA system and the control logic simulation on separate (physical) hosts (Fig. 8). Another host runs the 5G system using Open5GS [44] as the core network and srsRAN [45] as the radio access network. Connected to this host is an Ettus USRP B210 Software-Defined Radio (SDR) that serves as the antenna of the 5G base station (i.e., the cell tower of a 5G network). To connect the logical and physical simulation to the 5G network, we use a commercial-off-the-shelf smartphone as the User Equipment (UE). Finally, a second Ettus USRP B210 SDR serves as the attacker performing the *jamming attack* using the CleverJam [46] framework.

Results. We verify the successful deployment of CoFacS over 5G by observing the behavior of the production process. Similar to the fully virtual deployment, we can place new cylinders and observe the expected behavior from the PLCs and physical simulation when ordering a cylinder through the SCADA interface. However, when activating the jammer, we observe that no further communication via the 5G link is possible. Thus, the communication between SCADA and PLCs is disrupted rendering monitoring, ordering of goods and adapting the production process (e.g., the firing time) impossible. Thus, there is a need to develop and evaluate resilience measures for such attacks, for which CoFacS provides the ideal basis.

> *Take Away:* As demonstrated by two exemplary case studies, CoFacS enables a wide range of security research by comprehensively covering physical and networking behavior. Additionally, CoFacS allows producing artifacts for detailed studies and is adaptable to also incorporate real components such as an actual 5G network.

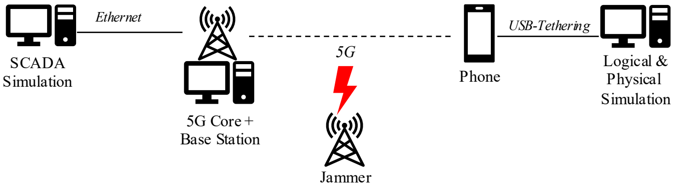

Fig. 8. We demonstrate CoFacS ability to enable security research beyond virtual simulation by deploying a real 5G wireless communication between SCADA (left) and logical simulation (right). We utilize this setup to study the impact of jamming attacks against industrial 5G communication.

VII. Conclusion

In this paper, we present CoFacS, the first complete factory simulation of an *entire* production process to facilitate security research for interconnected production and SCADA applications in an end-to-end manner. By comprehensively simulating the physical processes, PLC logic, network communication, and SCADA application of a complete production line, and also precisely emulating the industrial network, CoFacS enables the execution of real applications and especially attack tools. Through comparison with a physical reference, the Fischertechnik Learning Factory 4.0 [10], we show that CoFacS accurately captures the behavior of all components of the physical production line both under normal conditions and attack situations. Consequently, we can apply CoFacS to perform attacks such as malicious command injection, which cannot be performed against the physical testbed due to the risk of inflicting permanent physical damage. Additionally, we can extract data to evaluate security mechanisms in detail such as IDSs. Furthermore, the flexibility of our testbed allows us to exchange the emulated network with physical components enabling us to study the resilience of 5G communication in production systems against jamming attacks.

To enable follow-up work and thus spark further research to strengthen the security of interconnected production, we make CoFacS freely and openly available to the research community [47]. Therefore, CoFacS creates an ideal environment to evaluate research on industrial networks such as efficient transport security for industrial communication [48], classifying encrypted traffic [49], object security for industrial data [50], software defined networking in industrial settings [51], or scheduling algorithms for federated learning [52]. As such, with CoFacS, we lay the foundation for comprehensively studying the security of modern interconnected factories and thus ultimately improve their security.

Acknowledgments

Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy – EXC-2023 Internet of Production – 390621612 and the Foundation for Innovation in Higher Education (Stiftung Innovation in der Hochschullehre), Freiraum project RealistICS.

References

[1] J. Pennekamp, R. Glebke, M. Henze, T. Meisen, C. Quix, R. Hai, L. Gleim, P. Niemietz, M. Rudack, S. Knape, A. Epple, D. Trauth, U. Vroomen, T. Bergs, C. Brecher, A. Bührig-Polaczek, M. Jarke, and K. Wehrle, "Towards an Infrastructure Enabling the Internet of Production," in *ICPS*, 2019.

[2] E. D. Knapp, "Industrial Cybersecurity History and Trends," in *Industrial Network Security*, 2024.

[3] M. Conti, D. Donadel, and F. Turrin, "A Survey on Industrial Control System Testbeds and Datasets for Security Research," *IEEE Commun. Surv. Tutorials*, vol. 23, no. 4, 2021.

[4] T. Morris, R. Vaughn, and Y. S. Dandass, "A Testbed for SCADA Control System Cybersecurity Research and Pedagogy," in *CSIIRW*, 2011.

[5] L. Bader, M. Serror, O. Lamberts, Ö. Sen, D. Van Der Velde, I. Hacker, J. Filter, E. Padilla, and M. Henze, "Comprehensively Analyzing the Impact of Cyberattacks on Power Grids," in *IEEE EuroS&P*, 2023.

[6] D. Formby and M. Rad, "Lowering the Barriers to Industrial Control System Security with GRFICS," in *USENIX ASE*, 2018.

[7] A. Dehlaghi-Ghadim, A. Balador, M. H. Moghadam, H. Hansson, and M. Conti, "ICSSIM - A Framework for Building Industrial Control Systems Security Testbeds," *Computers in Industry*, vol. 148, 2023.

[8] M. Krotofil and J. Larsen, "Rocking the Pocket Book: Hacking Chemical Plants for Competition and Extortion," in *DEFCON*, 2015.

[9] Y. Xie, W. Wang, F. Wang, and R. Chang, "VTET: A Virtual Industrial Control System Testbed for Cyber Security Research," in *SSIC*, 2018.

[10] Fischertechnik GmbH, "Fischertechnik Learning Factory 4.0." [Online]. Available: https://www.fischertechnik.de/en/industry-and-universities/technical-documents/simulate/training-factory-industry-4,-d-,0-24v

[11] F. Sauer, M. Niedermaier, S. Kießling, and D. Merli, "LICSTER – A Low-cost ICS Security Testbed for Education and Research," in *ICS-CSR*, 2019.

[12] L. Malburg, J. Grüger, and R. Bergmann, "An IoT-enriched Event Log for Process Mining in Smart Factories," *arXiv preprint arXiv:2209.02702*, 2022.

[13] R. Sala, F. Pirola, and G. Pezzotta, "On the development of the Digital Shadow of the Fischertechnik Training Factory Industry 4.0: An educational perspective," *Procedia C.S.*, 2023.

[14] W. Alsabbagh, S. Amogbonjaye, Chaerin Kim, and P. Langendörfer, "Pirates of the MQTT: Raiding IIoT Systems with a Rogue Client," 2024, preprint.

[15] T. J. Williams, "The Purdue Enterprise Reference Architecture," 1994.

[16] C.-Y. Lin, S. Nadjm-Tehrani, and M. Asplund, "Timing-Based Anomaly Detection in SCADA Networks," in *CRITIS*. Springer, 2018.

[17] B. Ferling, J. Chromik, M. Caselli, and A. Remke, "Intrusion Detection for Sequence-Based Attacks with Reduced Traffic Models," in *Measurement, Modelling and Evaluation of Computing Systems*. Springer International Publishing, 2018.

[18] K. Wolsing, L. Thiemt, C. v. Sloun, E. Wagner, K. Wehrle, and M. Henze, "Can Industrial Intrusion Detection be Simple?" in *European Symposium on Research in Computer Security*. Springer, 2022.

[19] S. N. Foley, F. Autrel, E. Bourget, T. Cledel, S. Grunenwald, J. Rubio Hernan, A. Kabil, R. Larsen, V. M. Rooney, and K. Vanhulst, "Science Hackathons for Cyberphysical System Security Research: Putting CPS Testbed Platforms to Good Use," in *PrivaCy*, 2018.

[20] J. Gardiner, B. Craggs, B. Green, and A. Rashid, "Oops I Did it Again: Further Adventures in the Land of ICS Security Testbeds," in *CPS-SPC*, 2019.

[21] C. M. Davis, J. E. Tate, H. Okhravi, C. Grier, T. J. Overbye, and D. Nicol, "SCADA Cyber Security Testbed Development," in *NAPS*, 2006.

[22] V. S. Koganti, M. Ashrafuzzaman, A. A. Jillepalli, and F. T. Sheldon, "A Virtual Testbed for Security Management of Industrial Control Systems," in *MALWARE*, 2017.

[23] M. Almgren, P. Andersson, G. Björkman, M. Ekstedt, J. Hallberg, S. Nadjm-Tehrani, and E. Westring, "RICS-el: Building a National Testbed for Research and Training on SCADA Security," in *CRITIS*, 2019.

[24] P. Singh, S. Garg, V. Kumar, and Z. Saquib, "A Testbed for SCADA Cyber Security and Intrusion Detection," in *SSIC*, 2015.

[25] M. Mallouhi, Y. Al-Nashif, D. Cox, T. Chadaga, and S. Hariri, "A Testbed for Analyzing Security of SCADA Control Systems (TASSCS)," in *ISGT*, 2011.

[26] T. Alves, R. Das, and T. Morris, "Virtualization of Industrial Control System Testbeds for Cybersecurity," in *ICSS*, 2016.

[27] T. Morris, Z. Thornton, and I. Turnipseed, "Industrial Control System Simulation and Data Logging for Intrusion Detection System Research," in *Southeast. Cyb. Sec. Summit*, 2015.

[28] A. Almalawi, Z. Tari, I. Khalil, and A. Fahad, "SCADAVT - A framework for SCADA security testbed based on virtualization technology," in *38th Annual IEEE Conference on Local Computer Networks*, 2013.

[29] R. Uetz, C. Hemminghaus, L. Hackländer, P. Schlipper, and M. Henze, "Reproducible and Adaptable Log Data Generation for Sound Cybersecurity Experiments," in *ACSAC*, 2021.

[30] Team SimPy, "SimPy," https://simpy.readthedocs.io/en/latest/index.html, 2023.

[31] T. R. Alves, M. Buratto, F. M. De Souza, and T. V. Rodrigues, "OpenPLC: An Open Source Alternative to Automation," in *GHTC*, 2014.

[32] OpenJS Foundation & Contributors, "Node-RED," https://nodered.org/, 2024.

[33] A. Swales *et al.*, "Open Modbus/TCP Specification," *Schneider Electric*, 1999.

[34] M. Peuster, H. Karl, and S. van Rossem, "MeDICINE: Rapid Prototyping of Production-Ready Network Services in Multi-PoP Environments," in *NFV-SDN*, 2016.

[35] B. Lantz, B. Heller, and N. McKeown, "A Network in a Laptop: Rapid Prototyping for Software-defined Networks," in *HotNets*, 2010.

[36] J. Hiller, M. Henze, M. Serror, E. Wagner, J. N. Richter, and K. Wehrle, "Secure Low Latency Communication for Constrained Industrial IoT Scenarios," in *LCN*, 2018.

[37] M. Henze, J. Hiller, S. Schmerling, J. H. Ziegeldorf, and K. Wehrle, "CPPL: Compact Privacy Policy Language," in *WPES*, 2016.

[38] E. D. Knapp, "OT Attack and Defense Lifecycles," in *Industrial Network Security*, 2024.

[39] Team Metasploit, "Metasploit," https://www.metasploit.com/, 2025.

[40] K. Wolsing, E. Wagner, A. Saillard, and M. Henze, "IPAL: Breaking up Silos of Protocol-dependent and Domain-specific Industrial Intrusion Detection Systems," in *Proceedings of the 25th International Symposium on Research in Attacks, Intrusions and Defenses*, 2022.

[41] K. Wolsing, E. Wagner, L. Lux, K. Wehrle, and M. Henze, "GeCos Replacing Experts: Generalizable and Comprehensible Industrial Intrusion Detection," in *USENIX Security*, 2025.

[42] S. Michaelides, S. Lenz, T. Vogt, and M. Henze, "Secure Integration of 5G in Industrial Networks: State of the Art, Challenges and Opportunities," *Future Generation Computer Systems*, 2025.

[43] S. Michaelides, D. Eguiguren Chavez, and M. Henze, "Poster: Towards an Automated Security Testing Framework for Industrial UEs," in *IEEE EuroS&P*, 2025.

[44] S. Lee, "Open5GS," https://open5gs.org/, 2025.

[45] Team srsRAN, "srsRAN," https://www.srsran.com/, 2025.

[46] @jhonnybonny, "CleverJam," https://github.com/jhonnybonny/CleverJAM, 2025.

[47] S. Lenz, D. Schachtschneider, S. Jonas, L. Tirpitz, S. Geisler, and M. Henze, "CoFacS," https://github.com/RWTH-SPICe/CoFacS, 2025.

[48] J. Bodenhausen, S. Mangel, T. Vogt, and M. Henze, "Bidirectional TLS Handshake Caching for Constrained Industrial IoT Scenarios," in *LCN*, 2025.

[49] C. Dao, V. Tong, N.-T. Hoang, H.-A. Tran, and T. X. Tran, "Enhancing Encrypted Traffic Classification with Deep Adaptation Networks," in *LCN*, 2023.

[50] M. Henze, R. Hummen, R. Matzutt, D. Catrein, and K. Wehrle, "Maintaining User Control While Storing and Processing Sensor Data in the Cloud," *IJGHPC*, 2013.

[51] N. S. Bülbül, D. Ergenç, and M. Fischer, "SDN-based Self-Configuration for Time-Sensitive IoT Networks," in *LCN*, 2021.

[52] A. Taïk, H. Moudoud, and S. Cherkaoui, "Data-Quality Based Scheduling for Federated Edge Learning," in *LCN*, 2021.