Design and Realization of Testbeds for Security Research in the Industrial Internet of Things

Nils Ole Tippenhauer

Abstract Research on the (cyber) security of industrial control systems requires holistic understanding of practical systems in the field. In particular, important differences to IT security scenarios are related to industrial networking protocols and programming languages such as ladder logic. Arguably, access to realistic testbeds with physical process and related controls would enable researchers to understand the scenarios better, to develop attacks, and test countermeasures. While the implementation of such testbeds presents significant investments and efforts, the implementation process itself is often not discussed in literature. In this chapter, we discuss the design and realization of such industrial control system testbeds for security research. In particular, we discuss a process in which testbeds are designed by security researchers to resemble existing (and future) plants, and are then implemented by commercial system integrators using industry's best practises. As use cases, we provide details on design decisions, cost, and outcomes for three testbeds established at the Singapore University of Technology and Design.

Keywords Testbed design · ICS security · IIoT security · Applied security

1 Introduction

The discovery of Stuxnet demonstrated to the public that advanced attacks on Industrial Control Systems (ICS) exist in practice [3, 43], and motivated academic researchers to investigate security aspects of ICS and the future Industrial Internet of Things (IIoT). While reliability and safety issues are well understood by the engineering community, the introduction of cyber-security threats and solutions in ICS and the IIoT is still challenging for several reasons. Traditional (proprietary) industrial protocols feature no security mechanisms to guarantee confidentiality,

N. O. Tippenhauer (✉)
CISPA Helmholtz Center for Information Security, Saarbrücken, Germany
e-mail: tippenhauer@cispa.saarland

C. Alcaraz (ed.), *Security and Privacy Trends in the Industrial Internet of Things*,
Advanced Sciences and Technologies for Security Applications,
https://doi.org/10.1007/978-3-030-12330-7_14

authenticity, and freshness of exchanged traffic [12, 30]. Access controls for functionality of the protocols are usually non-existent, and the assumption is that anyone on the local network is trusted. The wide-spread use of legacy devices in existing ICS prevents fast introduction of protocols with modern security concepts. Legacy devices cannot be patched or upgraded without major investments, and risks to breaking working production systems, leading to major costs due to interruptions.

For that reason, academic research often focuses on passive attack detection measures to be introduced to complement existing systems [17, 26]. In that setting, the diversity of (proprietary) industrial protocols is challenging for the introduction of an Intrusion Detection System (IDS), as only limited documentation of many protocols is openly available. If active countermeasures are designed, their effects on operations in the IIoT need to be tested in a holistic manner, in real-world systems.

In addition, the actual physical process presents novel challenges and opportunities for research on attacks that leverage physical interconnections, and aim to achieve physical damage as impact [45]. For example, attacks in which sensor values are spoofed can have serious consequences on the control stability of a physical process [28], but investigations into that topic require detailed physical process models or real test systems.

The outlined challenges motivate the construction of realistic IIoT testbeds. The testbeds would contain both (scaled-down) physical processes, industrial control, and potentially the Supervisory Control And Data Acquisition Systems (SCADA) part of IIoT. Among other things, such testbeds would allow academics to (i) understand state-of-the-art industrial protocols and devices, (ii) experiment on attacks and countermeasures leveraging physical processes, and (iii) develop holistic detection and defense systems, and to test them with actual attacks.

Although such testbeds require considerable resources to set up and maintain, strong interest into IIoT security by funding agencies and the industry has resulted in construction of several by now (see a comprehensive survey in [22]). Example research results at top (computer science) security venues resulting such testbeds are [9, 14, 41]. While the implementation of such testbeds presents significant investments and efforts, the implementation process itself is often not discussed in literature.

In this chapter, we discuss the design and implementation process for IIoT security testbeds, based on experiences gained through the construction of three such testbeds at the Singapore University of Technology and Design (SUTD).[1] We provide a general discussion of our approach, together with details on the testbeds as use cases, lessons learned, and specific recommendations for future projects.

This chapter is structured as follows. In Sect. 2, we summarize the motivation to set up a testbed, and provide an introduction to components of IIoT. We present a general approach for design, specification and implementation of testbeds in Sect. 3. In Sect. 4, we discuss three testbeds at SUTD as use cases. Application of testbeds

[1] The author was employed at SUTD during most work for this chapter, and involved in design and use of all three testbeds.

for research, advantages and disadvantages, and lessons learned are discussed in Sect. 5, and related work is summarized in Sect. 6. The chapter is concluded in Sect. 7.

2 Industrial Control System Testbeds

In this section, we discuss motivations for the implementation of IIoT testbeds, design alternatives and trade-offs, and general architectures.

2.1 Motivation for Testbeds

There are three main motivations to construct a testbed for security research on Industrial Control Systems:

- **Discovery**: Discover knowledge on the subject that cannot be learned from simulations (without investing more in simulations than the real system).
- **Demonstration**: Demonstrate applicability of research findings to make them more convincing for researchers, funding agencies, and end users.
- **Education**: Leverage the testbed to educate students, researchers, stakeholders.

Components of an IIoT testbed could be real, emulated, or simulated (and a hybrid system mixing those types of components). For example, a testbed could contain only real components: Industrial devices, a real physical process, together with the SCADA system required to operate everything. Instead of a real process, high fidelity process emulators could also be used, for example Real Time Digital power Systems (RTS) in the domain of power systems [34]. Such emulation systems do real-time simulations of complex electromagnetic transients in distributed power systems, and are able to represent the simulated system state through analog or digital interfaces to other real industrial devices. If the overall system is mostly emulated (or simulated) and includes only few real devices, the setup is usually referred to as *hardware-in-the-loop*. If less precision is required, process or system stages can be simulated using domain specific software, or standard tools such as Python or Matlab. Simulations can be offline, or connected to other simulated systems in real-time. Typically, simulated systems would not exchange real network traffic with industrial protocols any longer, but use other (less overhead) messaging mechanisms between the systems components (direct network sockets, Message Queuing Telemetry Transport/MQTT).

The decision which components are required to be real, which should be emulated, and what can be simulated is certainly one of the most important decisions when designing a testbed for security research. A number of trade-offs have to be considered depending on the individual project scope, domain, and intended research:

- **Adaptability**: Cyber-Physical testbeds that contain real devices and real processes will require more effort to reconfigure and adapt to different settings. Fully simulated processes can likely be changed more quickly by updating the process topology. Real Programmable Logic Controllers (PLCs) will have to be reprogrammed with new control logic, while simulated systems might adapt automatically.
- **Fidelity**: Testbeds with real (although scaled down) processes and real devices allow to take observations that are expected to be closest to operational systems. Systems that are planned and implemented by industrial vendors are expected to resemble commercial systems better.
- **Observability**: Experimental work will likely require data collection, and potentially manipulations of physical processes and control logic. Data collection from real systems can leverage the SCADA and historian systems (if available), but introduction of additional sensors will cost money and potentially disrupt the process. The design of the system should have observability of the process in mind, and potentially allow for later addition of sensors if required. While data collection from simulations and emulations is usually easier, the data itself if expected to carry less details and surprising results.
- **Safety**: Simulated and emulated testbeds have less or no risk of damaging components, and endangering human lives. On the other hand, attack that threaten devices (or even humans) cannot be faithfully replicated in such systems.

We note that in [36], related requirements are stated (in a non-security context): Fidelity, repeatability, measurement accuracy, safe execution of tests. As noted in [22], repeatability and accuracy are likely less important in a security context (as it can be assumed that attacks/defenses need to work in a range of operating conditions). The observability we discussed is related to accuracy, but extends the concept to cover additional sensors and insights from the running system. The authors of [22] also note that a minority of existing testbeds provides arguments why they faithfully represent setups from industry.

For our testbeds at SUTD, we decided to design and implement fully functional testbeds without emulation of simulation. We made this choice to ensure high fidelity, to enable use of testbeds for other educational and training use, and in order to develop solutions that could directly be translated to industry. In terms of research goals, we planned to leverage the fully functional physical process to explore process-based attack detection and novel attacks, including physical interactions between process stages. To achieve diversity in physical processes, protocols, and vendors, we decided to build several testbeds, that could be interconnected. For the remainder of this chapter, we discuss similar setups.

2.2 Abstract IIoT Model

We start with an informal definition of IIoT testbeds. In general, cyber-physical systems contain a "cyber" part (i.e., communications, controllers, digital components), and a "physical" part (i.e., the physical process to be controlled, the actuators and sensors). An IIoT testbed can cover one or both of those areas. In the following, we refer to testbeds just covering the physical process as *physical testbeds*, testbeds that are covering only cyber components as *cyber testbeds*, and testbeds that cover both as *cyber-physical testbeds*. In this chapter, we focus on IIoT testbeds as sub-class of general CPS testbeds, but many aspects should also be generally applicable for other types of CPS testbeds.

2.3 General Structure of IIoT Testbeds

A general structure of IIoT testbeds is as follows (see Fig. 1): One (or more) physical process segments (which can be separate, or connected physically), the corresponding sensing and control devices, together with the required networking. To simplify discussion, we only refer to major components in the figure. A historian (a database to keep historical values of sensor measurements and actuator states in the IIoT) and SCADA (for supervisory control, monitoring by operators, and manual control) are responsible for overall control and monitoring. In individual process segments, local Human-Machine-Interfaces (HMI) can be found (that allow operators to directly interact with local PLCs and other components), together with the Programmable Logic Controller (PLC) that performs the distributed local real-time control. The PLCs are connected to the sensors and actuators through analog connections or field-buses.

Industrial Networking As shown in Fig. 1, the networking is often segmented into areas to accommodate logical and physical proximity, aggregated through *higher level* networks to forward the data. In the industry, the hierarchy of network segments (from local field-buses to SCADA and office networks) is often referred to as *Purdue Architecture* [44]. The lower levels carry sensor readings and direct commands to actuators using protocols such as Modbus/TCP, Ethernet/IP, and GOOSE. Physically, the protocols were traditionally spoken over bus architectures, with a trend in recent years to run legacy protocols over Ethernet and IP, e.g., with minimal changes as payload of a TCP connection (as in the case of Modbus/TCP). An in-depth introduction of related protocols can be found in [12].

The choice of industrial protocol spoken on the lower levels of the industrial control system often determines which vendors can be considered for individual components such as PLCs, as not all products support all protocols. As largest common denominator, Modbus/TCP is often chosen, as the protocol is so simple and common that it is supported by most devices. For future IIoT applications, publish/subscribe protocols such as Message Queuing Telemetry Transport/MQTT

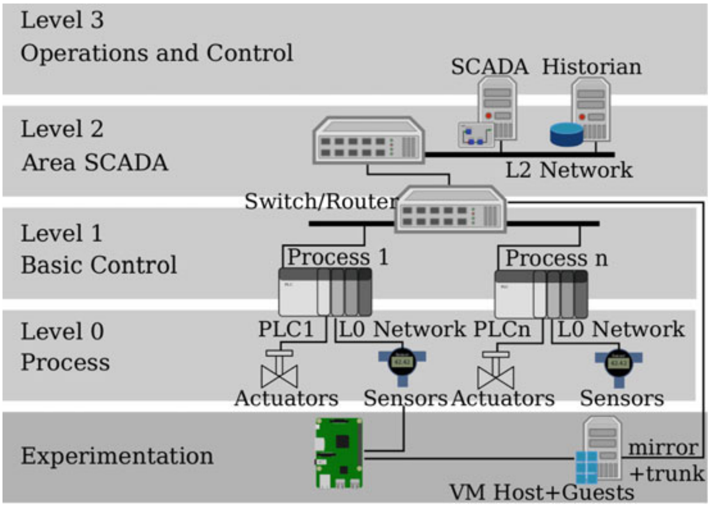

Fig. 1 General networking structure of our Industrial Control System testbeds

are also often discussed. Protocols used in the testbed will also depend on the application domain, as the type of industry and physical process (e.g., electric power) determines the standardization organization (for legacy reasons). As result, testbeds usually only use few different protocols, and cannot cover a large range. To mitigate that issue, testbeds could implement discrete sections with processes from different industries [20].

Physical Process Physical processes are diverse, and different domains such as electric power, water and oil processing, water and gas distribution, transportation and manufacturing have vastly different characteristics. In general, the control system interacts with the physical process through sensors (that observe the current state of the process), and actuators (that manipulate the physical state of the process). Actuation of the process will result in change of the state of the process, and that change will be observable through the sensors. Unlike digital systems, in which the spread of information can be precisely measured and contained, in physical processes different process segments and components can interact in a multitude of ways, from subtle changes (e.g., due to environmental temperature change) to more direct ways (e.g., exchange of product material).

Experimentation Platform While the testbed aims to replicate practical scenarios, additional components will be required to instrument the testbed (i.e., additional sensors, traffic taps), and to conduct active experiments (traffic manipulators, process manipulators). We refer to this second system as *experimentation platform* (in [20], dedicated networking for experiments is called a *management network*). The experimentation platform can be used to collect additional sensor measurement (that are not directly required by or used in the control process), obtain ground truth measurements in case of practical attacks that manipulate traffic, and to deploy countermeasures that are intended to be unreachable by an attacker. As such, the experimentation platform itself is usually assumed to be non-existent from the perspective of the attacker. In addition, the experimentation platform itself should never actively interfere or influence the normal operations of the system. We provide more details of such an experimentation platform when discussing the testbed in Sect. 4.1.

3 Design and Realization of Industrial Control Systems

In this section, we discuss the overall process to design and build an IIoT testbed, based on experience gained by constructing three large testbeds at SUTD (explained in more details in Sect. 4). In the following, we assume that the testbed is planned to be sufficiently large to contain (parts of) closed physical processes, and estimated cost will be more than 100,000 USD.

3.1 Preparations

Acquisition of Funding As the design and implementation of any testbed will require significant effort in manpower, it is likely that funding sources will have to be sought for before starting detailed planning. In particular, the overall time-frame of the testbed construction should be considered at this stage. As we will discuss later, the *time to operate* (i.e., time from starting to write a tender specification, to opening the testbed) was at least 12 months for the testbeds at SUTD, and additional time will be required to establish appropriate tool-chains to use the testbeds productively in research. As such, grants and deliverable time-lines should allow for at least 18 months of delay until results can be obtained. Ideally, construction of testbeds themselves will be seen as deliverables and milestones in the related grants. From experience, it pays to reserve larger contingency sums for costs related to the testbeds, to ensure that the overall system can be finished in face of unplanned costs.

At SUTD, funding for all three testbeds was sought before the procurement and implementation process was started, based on initial cost approximations. After funding was secured, the testbeds were then designed in more detail to work out specification documents. Using those specification documents, a public tender was called for each testbed, asking related system integrators and manufacturers to submit bids on construction of the whole system (or parts of the system).

Prerequisites In addition to funding sources and related projects, we briefly want to highlight other requirements that might be overlooked. In general, a testbed will present potential hazards to humans and assets, so such risks need to be addressed and mitigated. Sufficient permanent space has to be found for the testbed, together with infrastructure to enable continuous operations (depending on the testbed, for example power supplies, water supplies, waste water disposal, sprinkler systems, and even network connectivity). If industrial processes are replicated, it can be expected that components with heavy weight are required. In that case, it has to be ensured that the floor's load capacity will not be violated by larger components, and that leakages can be contained. Noise emanations from the testbed should not interfere with other users of nearby spaces. Heat produced in the testbed should be appropriately addressed through ventilation and air conditioning.

We also found that testbeds should be designed with sufficient free space in mind in order to allow tours for visitors, and ideally have demonstrations in mind when designing setup of HMIs, monitors, and similar. Researchers naturally want to work close to the testbed when running experiments, so ideally seating in a nearby room (isolated from heat and noise generated by the testbed) should be available. In our testbeds, we also found that large windows connecting the researcher seating and testbeds were quite helpful for visual inspections of the current state of process (e.g., during attacks, when data reported was unreliable).

3.2 Testbed Specification for Tender Process

We recommend to draft the testbed design including physical process and cyber components by a committee formed involving faculty members (who are going to be the main users) and admin staff (who will support the procurement process). Based on those initial ideas, we then suggest to hire a consultant and involve stakeholders from the industry to work out further details. In particular, we found that the physical process specification has to be quite detailed to allow for a smooth tender process. For example, a fair evaluation of the tender bids requires that all bidders will propose roughly the same system, so ambiguities in the specifications should be avoided. The tender specification should also mention specific standards and protocols to use in the industrial control, if such requirements exist from the research side. We found that the physical process specification and the specification of the main control system were relatively straight forward to discuss with the vendors. More attention had to be paid to specify requirements related to the experimentation platform, and instrumentation related to the research nature of the testbed. As an example, we requested the vendors to add wireless networking options in our testbeds, and in general segment networks more than strictly required to "simply make it work". In the end, additional work by us was required to really integrate the networking setup with our general back-end, and enable seamless manipulation and monitoring of traffic at the different levels of the network. On average, the technical specification for the tenders at SUTD were around 45 pages long, and significant effort was spent on those documents.

3.3 Tender and Construction

Tender process We assume that your local policies mandate a public tender process to ensure appropriate spending of funding (as it was the case in Singapore). As part of the tender process, the specifications are published, and bidders have a time window to submit bids (usually, several weeks). In addition, tender briefings and on-site visits will be held. After bids are submitted by interested companies, they will be reviewed by a committee, and requests for clarifications (or a second tender stage) will be asked if required. For our testbeds at SUTD, time from publishing of the tender spec to awarding the tender was 90 days (on average).

The tender specification documents outline operational requirements such target flow rate of the process, total power to be generated, safety requirements, requirements on the documentation, and similar. In addition, evaluation criteria were outlined, together with and overview of the evaluation process. For our testbeds, the technical tender specification had an average length of 45 pages.

Construction process The average time to build the testbeds (from day of awarding, to official opening after the user acceptance test) was around 9 months. During that time, manpower was required on our side to liaise with the vendors to answer questions during the implementation, and supervise the progress. At the end, a comprehensive user acceptance test was conducted in each case, in which the fulfillment of the requirements from the tender specification was verified and certified.

We also found it useful to require the vendor to provide training on the software and hardware used in the testbed for our researchers. In addition, control strategies used to program the PLCs had to be documented in a manual, together with details on the implementation such as comprehensive listing of all *tags* (i.e., variables) used in the system, comments on the protocols used, user-names and passwords, and similar. For future projects, we would recommend to add requirements on specifications of physical process parameters and relations between physical features measured (if possible). Ideally, suitable process simulation software would be available to complement the actual physical testbed, and to allow for easier simulation of the overall system.

4 Case Studies: SWaT, WADI, EPIC

In this section, we introduce the three testbeds at SUTD (see Fig. 2), and compare them on a number of dimensions relating to their design and construction, manufacturers and protocols, and outcomes. The testbeds are Secure Water Treatment (SWaT), Water Distribution (WADI), and Electric Power and Intelligent Control (EPIC). The testbeds all have fully functional (but scaled down) physical processes, controlled by state-of-the-art industrial controls (implemented by a commercial system integrator).

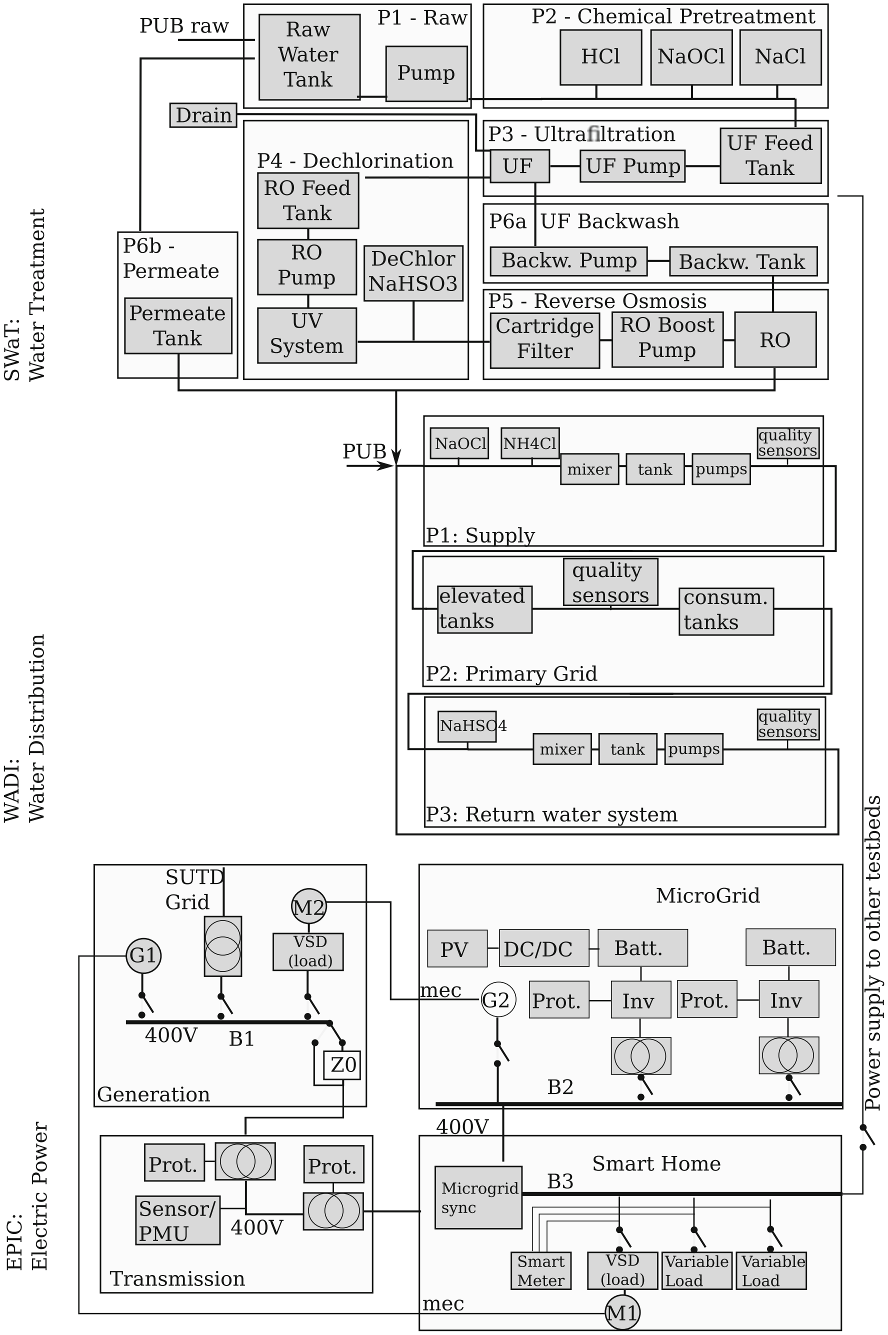

Fig. 2 Summary of physical process stages in the three use-case testbeds at SUTD. The water treatment testbed can be used to feed the water distribution testbed, and the power testbed is designed to be able to supply the other testbeds if needed for experiments on cascading effects

Table 1 Key information on design, procurement (costs approximate), and operation of SUTD testbeds

Name	Domain	Segments	Approx. cost	Opening	Time to open
SWaT	Water treatment	Chemical treatment, ultra-filtration, UV-dechlorination, reverse osmosis	750k USD	March 2015	15 months
WADI	Water distribution	Chemical treatment, gravity-fed distribution boosted distribution	750k USD	July 2016	12 months
EPIC	Electric power	Generation, transmission, micro-grid, smart home/consumption	750k USD	May 2017	21 months

Table 2 Summary of manufacturers and industrial protocols of main control devices used in the testbeds. RTU refers to Remote Terminal Unit, PLC refers to Programmable Logic Controller

Testbed	Manufacturer	Type	Model	Count	Protocols
SWaT	Allen Bradley	PLC	ControlLogix 1756	12	Ethernet/IP
WADI	National instruments	PLC	RIO-cRIO 9068	3	LOGOS (NI LabView)
WADI	Schneider electric	RTU	SCADAPack 334	2	Modbus/TCP
EPIC	Wago	PLC	750-8202	5	MMS (ISO 9506)
Skid 1	Allen Bradley	PLC	750-8202	3	Ethernet/IP
Skid 2	National instruments	PLC	cRIO 9063	1	LOGOS
Skid 2	Siemens	PLC	SIMACTIC S7-1500	1	S7
Skid 3	Schneider electric	PLC	CPU 651 50	1	Modbus/TCP

We start by summarizing their domain, the approximate cost to procure and construct, and the *time to open* (time from starting to write the specification, to officially opening the testbed) in Table 1. Outcomes from the testbeds are discussed in Sect. 5.

As each testbed covers a different physical process domain and was purchased separately, devices and their manufacturers differ between them. Table 2 summarizes main manufacturers, type, and models of the industrial devices used in the testbeds.

4.1 Secure Water Treatment Testbed

The SWaT Testbed was the first to be constructed at SUTD. Its construction was motivated by a collaboration with the Public Utility Board (PUB) in Singapore, who is operating (among other things) the water distribution and treatment processes for households and commercial customers in Singapore. The SWaT testbed was designed to resemble a modern water treatment process, which is leveraging chemical treatment and reverse osmosis membrane filters. More details on the SWaT testbed are available in [29].

Design and Procurement The physical process of SWaT is designed as distributed control of a interconnected plant, with six distinct process stages. The process stages include raw water, chemical pre-treatment, ultrafiltration, dechlorination, reverse osmosis, and backwash/permeate. As the names suggest, the physical process contains both hydraulic components (valves, pumps, tanks), and chemical treatment (using HCl, NaOCl, NaCl). The physical process design was lead by a consultant from the water treatment industry, together with few faculty members to ensure that ideal conditions for experimentation would be provided.

Preparation of the design specification for the tender was started around February 2014, with the main tender call published in August 2014. The overall technical design specification for the tender covered around 50 pages. The tender was awarded in October 2014 after several rounds of meetings with bidders and subsequent clarifications.

Experimentation platform We soon realized that practical attacks and attack detection in the fieldbus (level 0) communications requires devices in the network (in the case of SWaT, the device level ring between PLCs and the remote Input/Output Units (RIOs) and actuators). Such devices would require at least three networking interfaces: One that is connected to the *experimentation platform* (e.g., to control the devices, and aggregate data), and two interfaces to bridge the device level ring network without disturbing the normal communication flow. To minimize cost and space requirements, we are mostly using Raspberry Pi devices for that task, equipped with a wireless interface (to connect to the experimentation platform), and two wired Ethernet adapters (to bridge the device level ring). While establishment of a bridge for the ring network is straight forward with Linux networking (i.e., using *bridge-utils*), traffic manipulation is also possible, but requires additional work. As traffic manipulation usually requires parsing first, we implemented a framework using libnetfilter-queue and scapy with custom protocol support.[2]

In addition, we adapted the networking of the industrial system to allow for better monitoring of the traffic, and connection of additional devices. To achieve this, the central industrial switch (Moxa EDS-316) connecting most devices of level 1 and 2 in the network was replaced with a 24-port configurable switch with management interface (HP E3800). The switch allowed us to set up four monitoring ports, which were then directly connected to IDSs, and several servers that act as general Virtual Machine (VM) hosts. The VM hosts are 2U servers with processors that feature a larger number of logical cores, and sufficient RAM to operate around 10 VMs on each host. The VM hosts are connected through Virtual Local Area Network (VLAN) trunk connections to the switch (i.e., with attached 802.1Q headers), which

[2] See the swat-assault-crawler source at [40].

allows guests on the VM hosts to access any VLAN on the main switch. We then migrated the SCADA server, engineering workstation, and historian (all running on Windows OS) to be guests on the VM hosts. The VM host for Windows guests is also running Windows (to allow for easier management of licenses), while other research and experimentation guests are usually running on Linux, on an Ubuntu VM host. This setup allowed a very simply backup solution for important machines, ensuring that they could be restored after experimentation if required. In addition, sending the mirroring port traffic to VM hosts allowed to supply a larger number of VM guests with mirroring traffic in real-time, which would otherwise have been more challenging due to restrictions on the number of mirroring ports on the main switch.

Challenges As SWaT was our first testbed, we had to develop a range of tools to collect data from the testbed, and perform attacks. A main challenge in that context is the lack of open source tool support for the main protocol used in the testbed, Ethernet/IP, and the interface to the historian. For both, we had to implement our own software stack, before we were able to fully run experiments. In addition, reprogramming the PLCs with the industrial Studio 5000 software had to be learning through training sessions organized by the vendor. Related to this, our researchers had to be trained on the IEC 61131-3 compliant languages such as ladder logic and functional block diagrams.

Understanding and simulating the physical process was another challenge. Although we specified the physical process in detail in our tender specification, our work on attacks and detection mechanisms required detailed understanding of process dynamics. The related equations were not provided by the vendor and had to be derived by us. As the SWaT process contains physical and chemical aspects, finding an overall process model that covers all aspects was not possible. We currently model hydrological and chemical processes separately. Parameters for the simulations were derived from the process description in the manual by the vendor, and by matching models to observations.

4.2 Water Distribution Testbed

The WADI testbed focuses on a replication of a geographically distributed and centrally controlled water distribution network, for example of a city. The testbed features three main process stages: (i) Water storage and transmission grid, (ii) Consumers, and (iii) Water return (required in testbed to return "consumed" water to water storage). Each process stage is individually controlled by PLCs. The testbed is set up physically next to SWaT, in a room of approximately 80 m^2 size, and designed for a throughput of 10 US gallons of water per minute.

A major consideration in the design of WADI was the realistic simulation of gravity-based water pressure in the network. To achieve effects similar to the ones experienced in real systems, the water tanks in WADI are mounted at different heights, pipes with varying diameters are used, and booster pumps are available. In addition, a custom leak simulation setup was designed and implemented as part of WADI. The leak simulation allows to divert a defined percentage of water from the main distribution pipe that lowers pressure and volume of available water.

As WADI simulates geographically distributed centrally controlled system, it contains a number of remote terminal units and PLCs (NI compactRIO) that aggregate data from local sensors, and transmit that data to a central SCADA system. The transmission link for that communication can be switched between (simplistic) Ethernet communication, and 3G-based wireless communication. As such, a range of different attack and defense scenarios can be investigated. In WADI, the PLCs and RTU are connecting to most sensors and actuators directly. Where needed, Modbus/TCP is used for communications between RTUs and the SCADA, and the National Instruments proprietary Logos protocol between the PLCs and the SCADA. For further details on WADI, we refer to [2].

Procurement The WADI technical specification document was a bit shorter, due to a simpler physical process and control design. The tender was published in September 2015, and awarded in December 2015. The testbed was opened after 7 months in July 2016.

Challenges Similarly to SWaT, the PLCs (National Instrument CompactRio) and SCADA in WADI communicate with a proprietary protocol. In this case, it is what we assume to be the Logos Real-Time Protocol (related to LabVIEW), for which we did not find open source libraries that would support parsing or creation. We are currently working on a simulation model for the full physical process in the EPANET software, but face issues with matching the simulations with data from the testbed. In particular, several sensors in the actual system drift over time, which is challenging to account for when comparing data from real and simulation system.

4.3 Electric Power and Intelligent Control Testbed

The EPIC testbed consists of four process segment: Three-phase generation segment, transmission segment, micro-grid segment, and a smart home segment. Together, the generation, transmission, and smart home segments emulate a utility grid network. In addition, the micro-grid segment consists of photo-voltaic (PV) generation, transformers, battery-based storage, and protection devices. The testbed is briefly introduced in [37], with particular focus on cyber components.

Physical Process The generation part consists of three motorized generators (powered by the SUTD power grid). In particular, the SUTD grid is used to power the motors M2 and M3, which are mechanically linked to generators G2 and G3 in the micro-grid. Generator G1 is powered when M1 is powered through the smart home load bus.

In the transmission portion, a transformer is used to step down the voltage from the distribution levels to transmission levels. While in real systems, electricity is normally transmitted at high voltages in order to reduce losses, EPIC uses lower voltage for safety reasons. For slight changes in load, tap changes in the transformer adjust the voltages. The smart home segment consists of variable resistive, capacitive, and inductive loads. The loads are connected to the smart home section by a bus tie. The motor M1 is used as load. This motor can operate the generator G1, and provide power to the extra loads in the system.

Control System In general, the communication infrastructure and control system in EPIC is similar to that in WADI. In particular, it represents a spatially distributed but centrally controlled system. The individual process stages each have one PLC or RTU connected to the central SCADA system. The PLC in the generation process controls the different ways to generate power for the system, and collects data from the sensors and fault detection components. The PLC in the smart home section is used to control the variable loads from the SCADA. In addition, the smart home has several smart meters connected to the loads, which report the usage to the Advanced Metering Infrastructure (AMI).

Procurement The design specification for EPIC had approximately 45 pages, and was published as part of the tender call in September 2015. The winning bid of the tender was selected and awarded in January 2016. Construction of the testbed took about 16 months, in particular due to delays related to the more complex process (and safety considerations).

4.4 Training Skids

In addition to the three testbeds with full physical process, we also purchased several *training skids*. The training skids are usually employed in industry for staff training on PLC and RTU programming, and consist of 1–3 PLCs from different product lines of the same manufacturer, together with IO components, and dummy sensors and actuators. The dummy sensors are not connected to actual physical processes, but can be manipulated by users easily. For example, a thermometer is measuring environmental temperature, and its measured value can easily be changed by touching it. Other sensors are simple switches, that can be actuated by the user and provide binary signals to the attached PLCs. Similarly, the dummy actuators can be used to give feedback to the user, e.g., by lighting up lamps or activating a simple motor.

The training skids allow the researchers to interact with the devices without fearing to break the overall process, and to become familiar with the respective programming environments. In addition, we found that the skids provide good variety of the devices for traffic capture and vulnerability testing. Overall, while the price of a single training skid is significantly less than a full testbed (around 100k USD), we also found that their usefulness for most of our direct experimental research work is limited. As such, they are a good platform to familiarize new staff with the environment, but do not offer the functionality that more complete testbeds provide.

5 Discussion

We now discuss how the introduced testbeds were used for research, options for commercialization, data collection and sharing, remote access and collaboration, and other lessons learned.

5.1 Testbed Applications for Research

Since their opening, the testbeds have been leveraged in a significant number of publications authored by researchers from SUTD (to date, more than 50 conference and workshop papers, and around 5 journal publications). A number of quite productive research directions developed out of the testbeds, for example the experimental investigation of attacks, and collection of data samples during normal and attack scenarios, together with related attack detection schemes.

A semi-public dataset with traces of normal operations and several attacks in the SWaT testbed was collected in 2016, and made available on request online. To date, access to the dataset has been requested by (and granted to) more than 115 international researchers and teams. Details on the dataset can be found on the website[3] and in the companion publication [18].

For the other two testbeds (WADI and EPIC), similar datasets and companion papers are planned for the future. We feel that the existence of such common dataset will benefit the community and enable collective work towards better attack detection mechanisms with comparable performance evaluations. While similar datasets can be generated by use of physical process simulators (e.g., to host attack detector competitions [39]), datasets from real testbeds will have more realistic noise, artifacts, and the possibility to combine network and process data.

Attack Traces Research on security of IIoT faces the challenge that very little information on actual attacks are publicly available. In particular, no details network traffic captures or historian traces of actual attacks are published (to the best of our knowledge). While a number of attacks are prominently cited in academic works (e.g., Stuxnet [43], the Maroochi shire attack [38], the Aurora Attack [45]), those attacks are only described on a certain level of abstraction. As result, there is no common attack to help design or evaluate proposed countermeasures.

[3] Available online at [25].

IIoT testbeds can help with collection of traces of attacks to enable comparison of countermeasures, and drive the design process of protection mechanisms. In particular, there are four options to generate such traces: (i) Attacks on the physical process in which the attacker has control over HMI or SCADA can be conducted by simply overriding normal plant control manually on the HMI or SCADA, (ii) Attacks in which the attacker manipulates traffic in real-time can be conducted by using suitable devices that are placed as man-in-the-middle (e.g., as part of the experimentation platform discussed earlier), (iii) (trusted) Third parties could be asked to perform (unknown) attacks while traffic and historian data is recorded, and (iv) the testbed could be exposed to untrusted third parties (similar to a honeypot). Out of those options, we used (i)–(iii) to generate attack data from our testbeds, further details can be found in publications such as [1] (for (i)), [42] (for (ii)), and [5] (for (iii)). Option (iv) (attacks by untrusted third parties/honeypot) was not considered so far, as the risk of physically damaging the plant was considered too high, the likelihood of successful attacks by third parties was considered too low, and recovery of the system was estimated as too challenging.

Overall, we believe that generating such datasets is one of the key advantages of similar testbeds, and a promising way to provide contributions to the community. To the best of our knowledge, only few other such datasets have so far been made available. A collection of power system, gas and water system and energy management system data was made available by Oak Ridge National Laboratories (ORNL).[4]

Security Competitions The SWaT testbed was also used as centerpiece for two *SWaT Security Showdown* (S3) competitions held at SUTD in 2016 and 2017. In those competitions, 5–6 international teams of students, academic researchers, and industrial professionals were invited to develop and demonstrate attacks on the SWaT testbed. Teams were invited directly (in 2016), or had to qualify (in 2017). During an on-line stage teams had to demonstrate basic capabilities in IIoT security in a Capture-The-Flag (CTF)-like event. Afterwards, successful teams were invited to Singapore, and given around one day of time in the lab to prepare, and then had to demonstrate their attacks. Attacks were scored based on a formula that awarded impact and control over the physical process, while at the same time using weaker attacker models (i.e., only access over the network). More details on the event in 2016 can be found in [5], and related reports on the iTrust website [24].

Countermeasures We used the testbeds to test various countermeasures we developed, among them solutions integrated in PLCs, traffic monitoring on the fieldbus [42], and SCADA systems [16]. In addition, we were able to host a number of commercial platforms in the testbeds for extended periods of time, which (among other things) enabled potential government customers to become more familiar with the products. It also enabled the system vendors to demonstrate their capability in customizing the product to specific settings, and their support for proprietary

[4] Available online, at [31].

protocols. Most countermeasures (academic and commercial) were also evaluated as part of the S3 competitions (see above).

5.2 Commercialization

As discussed, testbeds represent significant investments in terms of money and effort. In addition, continued operations require funding for maintenance and replacement of consumables, and salaries for operational and admin staff. Testbeds can also be expected to operate over a longer time-frame, as the technology itself will likely still be relevant 10 years later (given that industrial technology in the field can have much longer lifetime).

In order to make such testbeds self-sustainable, either long-term funding through grants has to be acquired (ideally, 10+ years), multiple sequential grants have to be acquired (which is susceptible to unexpected delays in new grants, or failure to secure funding), or other funding sources have to be found. In particular, charging for access to the testbeds (for training or research purposes) for third parties promises to mitigate the funding problem. To the best of our knowledge, this practice is not yet widely established (a trial is currently running at SUTD). While charging will likely bring in additional funding, it will also require a professionalized management of resources and technical support, likely moreso than purely academic research environments usually provide.

5.3 Data Collection and Sharing

One of the advantages of real-world IIoT testbeds is the opportunity to collect traffic and process data over long time frames, under different operating conditions, and including artifacts such as the industrial protocol headers, temporal features, and noise. Collection of a dataset that unites traffic and historian process data, ideally with detailed labels and context information, would provide a very valuable source for applications such as machine-learning enabled attack detection, and verification of simulation and emulation environments.

Unfortunately, we found it challenging to collect such datasets in the SUTD testbeds for various reasons. Testbeds initially did not have appropriate means to capture relevant network traffic, extraction of process data from the historian required use of industrial software (e.g., manual queries for each individual tag, use of Windows-only development kits with libraries), and integration of both types of data together with labels requires suitable frameworks to be developed. While partial dataset have been semi-publicly release by SUTD (essentially on request), those datasets cover only historian data or network traffic, and not both. Labeling of the datasets is also only coarse, as periods are labeled as under attack, or normal traffic.

For future testbeds, we would recommend to leverage a central data collection and labeling platform, which combines data such as network traffic from various taps, historian data, information such as log-files produced, and enables to easily label the samples (perhaps semi-automatically based on process state data). This central platform could be used by all researchers to obtain and share data, and minimize engineering effort related to data acquisition.

5.4 Remote Access and Federation

While the IIoT network part of the network will likely not be directly connected to the Internet (not even indirectly through a NAT), other network segments will likely have Internet access, e.g., for researchers. In our testbeds, we use open source network applications such as pfSense [33] to provide NATing and firewalling. The central pfSense host is directly connected to our Internet uplink on a public IP.

Remote access to testbeds is often asked for by collaborators at SUTD. In general, remote access to the experimentation platform and IIoT network is not difficult to achieve. We use VPN server features of the pfSense application to allow remote users to connect with a VPN client, which will typically give them network-layer access to a special VLAN which also contains selected virtual machine guests. The VM guests are then used by our external collaborators to collect traffic or interact with selected other network segments. Based on virtual networking setup of the host the VM guests can have any number of virtual interfaces, in any of our VLANs. External collaborators can also connect to our HMI and SCADA system if allowed in our configuration, and can thus directly influence the physical process. Due to the related safety risks, we require on-site collaborators to be present for such experiments.

The setup would technically also allow to create link-layer bridges over VPN with remote sites, to integrate remote testbeds and SCADA with our testbeds. So far, the engineering effort, security and safety concerns have prevented us from setting up such systems in practice.

5.5 Lessons Learned

We conclude the discussion now with a quick summary of lessons learned from the process of building and operating our testbeds.

Network specification vs. implementation We found that it was relatively challenging to ensure that the final network design and implementation followed our specification closely enough to allow for easy experimentation and research. In all three testbeds, the original networking designs had to be revised several times during implementation. Most likely this is influenced by the fact that protocols used and data exchanged are influenced by control logic and devices used, and in normal industrial settings the vendor has more freedom to implement the network as deemed suitable. We recommend to at least stay in touch with the vendor on network design decisions to ensure that the resulting platform will be suitable for research.

Staff We found that permanent engineering staff is essential to support the testbeds. The staff does not only need to take care of maintenance of the process, but also acts as support for the researchers during their work. The engineer can oversee physical experiments to ensure no damage is done unintentionally, and support the development of tools to interact with the system. Of course, this job profile asks for a person with industrial engineering and programming experience, who is often hard to find. The networking setup and virtualization environments also benefit from a permanent staff member who is familiar with the setup. We have one permanent engineer position for each testbed. In addition, several admin staff members help with admin side of testbeds and collaborations.

Size and Complexity of Physical Process We found that the sheer size and complexity of the first testbed (SWaT)—while allowing for interesting experimental setups—made it hard to fully model the process in order to theoretically analyze it. Ideally, the testbed needs a mix of small closed control loops and process segments, and some larger and more complex ones. In practice, we found that in many of our publications, we used simple process stages such as the raw water tank (a tank with a pump, level sensor, several valves and flow meters). The simplicity of those process segments allowed us to more easily model the system, and spend less time explaining the setup in the publications. After more experience on systems and solutions was gained, larger process segments could be used.

6 Related Work

A number of general surveys on ICS security provide a good overview of challenges and current work [30, 46]. In [46], the authors review ICS intrusion detection and prevention systems. In [30], the authors review the general cybersecurity challenges and state of the art in defense in ICS networks. In particular, the authors also discuss advantages of hardware-in-the-loop testbeds. Work on ICS security is often focusing on power systems due to seminal works in that area [28]. An example for work in other domains is [8], in which is railway ICS security is discussed.

Testbeds A number of surveys provide an overview of ICS testbeds in different application domains and research fields [3, 17, 22]. While a full review of related testbeds is out of scope, we would like to mention a number of relevant ICS testbeds used for security research [6, 13, 20, 32]. In [32], the authors present a SCADA Security Laboratory and Power and Energy Research laboratory, which features several physical process segments with a RTU unit in each, and a central HMI to control the segments. The physical process segments are more diverse than the ones covered in the testbeds at SUTD, but are much smaller and less complex. In [21], the PowerCyber testbed at Iowa State University is introduced, which features RTS simulation of the physical process, and emulation of WAN communication. Full virtualization of testbeds is discussed in [35], with PLC and RTU components simulated in Python, and communicating over virtual network interfaces. The resulting testbed is also able to communicate with physical industrial devices. In [15], another framework is presented, which combines real-time simulation of physical processes (using Simulink), and emulation of cyber-components using Emulab. In [13], the authors discuss hybrid ICS testbeds for security research with special focus on emulation. In [10], an approach based on network emulation, process simulation, and SDN is presented in the context of power systems. A framework leveraging virtualized network emulation, device simulation using Linux guests, and an abstract physical process interface was proposed in [4]. In [19], the authors experimentally explore attacks on a vinyl acetate monomer plant by leveraging a fully simulated chemical process (the Tennessee Eastman challenge Process [11]). The process has also been made available as open source.[5]

Automotive ICS In this work, we mostly discussed testbeds related to utilities such as water and power, and plants such as they are used in manufacturing. A related research area is the one of automotive security. In that context, the threatened system is itself a commercially available product, which means that it is possible to just buy or rent the system *off the shelf*. Several high impact research works have done this, e.g., [7, 27]. To the best of our knowledge, rental of equipment for other ICS research has not been used otherwise, but could also be a promising avenue for systems that are relatively self-contained.

7 Conclusion

In this chapter, we provided an overview of motivations, design considerations, and alternatives for IIoT testbeds for security research. We focus on cyber-physical testbeds that combine a scaled-down physical process with real industrial devices such as PLCs, SCADA, sensors, and actuators. The design and implementation of such testbeds requires considerable resources and effort, and we provide and overview of the overall process, estimates on documentation and time required (based on our experiences in constructing three such testbeds).

Testbeds are long-term investments and will not only be used for single research projects. To show options for different usage scenarios, we summarize how our testbeds have been used so far, and what our plans for future use is. We also

[5] Available online at [23].

comment on general academic results so far, and point to promising directions to increase impact (e.g., through release of more extensively prepared datasets).

Overall, we conclude that the availability of cyber-physical testbeds provides unique opportunities for applied research, education, and outreach activities. Results and insights from such testbeds cannot be obtained from simulation and emulation alone. Given the required investments, suitable long-term funding and man-power is required to fully utilize those opportunities.

Acknowledgements Many people were involved at SUTD in the design, procurement, and operation of the testbeds. Most importantly, Aditya Mathur was responsible for the initial vision and overall leadership of the project. Ivan Lee, Mark Goh, and Angie Ng from iTrust additionally supported the process from the admin side. Writing of the SWaT process requirement documents was supported by Giedre Sabaliauskaite, while the WADI process was designed together with Stefano Galelli. Gareth Thomas acted as consultant on both water testbeds. The three testbeds were funded by grants awarded by the National Research Foundation of Singapore, the Singaporean Ministry of Defence, and the International Design Center at SUTD.

References

1. Adepu S, Mathur A (2016) Distributed detection of single-stage multipoint cyber attacks in a water treatment plant. In: Proceedings of the ACM ASIA Conference on Computer and Communications Security (ASIACCS). ACM, pp 449–460
2. Ahmed CM, Palleti VR, Mathur A (2017) WADI: a water distribution testbed for research in the design of secure cyber physical systems. In: Proceedings of the Workshop on Cyber-Physical Systems for Smart Water Networks (CySWATER), Apr 2017. ACM, pp 25–28
3. Antón SD, Fraunholz D, Lipps C, Pohl F, Zimmermann M, Schotten HD (2017) Two decades of scada exploitation: a brief history. In: Proceedings of the IEEE Conference on Application, Information and Network Security (AINS), Nov 2017, pp 98–104
4. Antonioli D, Tippenhauer NO (2015) MiniCPS: a toolkit for security research on CPS networks. In: Proceedings of Workshop on Cyber-Physical Systems Security & Privacy (SPC-CPS), co-located with CCS, Oct 2015
5. Antonioli D, Ghaeini HR, Adepu S, Ochoa M, Tippenhauer NO (2017) Gamifying ICS security training and research: design, implementation, and results of S3. In: Proceedings of Workshop on Cyber-Physical Systems Security & Privacy (SPC-CPS), co-located with CCS, Nov 2017
6. Candell R, Stouffer K, Anand D (2014) A cybersecurity testbed for industrial control systems. In: Process Control and Safety Symposium, International Society of Automation, Houston
7. Checkoway S, McCoy D, Kantor B, Anderson D, Shacham H, Savage S, Koscher K, Czeskis A, Roesner F, Kohno T (2011) Comprehensive experimental analyses of automotive attack surfaces. In: Proceedings of the USENIX Security Symposium
8. Chen B, Schmittner C, Ma Z, Temple WG, Dong X, Jones DL, Sanders WH (2015) Security analysis of urban railway systems: the need for a cyber-physical perspective. In: Koornneef F, van Gulijk C (eds) Computer safety, reliability, and security. Springer, Cham, pp 277–290
9. Chen Y, Poskitt CM, Sun J (2018) Learning from mutants: using code mutation to learn and monitor invariants of a cyber-physical system. arXiv preprint arXiv:1801.00903
10. Dong X, Lin H, Tan R, Iyer RK, Kalbarczyk Z (2015) Software-defined networking for smart grid resilience: opportunities and challenges. In: Proceedings of the 1st ACM Workshop on Cyber-Physical System Security. ACM, pp 61–68
11. Downs JJ, Vogel EF (1993) A plant-wide industrial process control problem. Comput Chem Eng 17(3):245–255

12. Galloway B, Hancke GP et al (2013) Introduction to industrial control networks. IEEE Commun Surv Tutor 15(2):860–880
13. Gao H, Peng Y, Jia K, Dai Z, Wang T (2013) The design of ICS testbed based on emulation, physical, and simulation (EPS-ICS testbed). In: Proceedings of the Conference on Intelligent Information Hiding and Multimedia Signal Processing, pp 420–423, Oct 2013
14. Garcia L, Brasser F, Cintuglu MH, Sadeghi A-R, Mohammed O, Zonouz SA (2017) Hey, my malware knows physics! attacking PLCs with physical model aware rootkit. In: Proceedings of the Annual Network & Distributed System Security Symposium (NDSS), Feb 2017
15. Genge B, Siaterlis C, Fovino IN, Masera M (2012) A cyber-physical experimentation environment for the security analysis of networked industrial control systems. Comput Electr Eng 38(5):1146–1161
16. Ghaeini H, Tippenhauer NO (2016) HAMIDS: hierarchical monitoring intrusion detection system for industrial control systems. In: Proceedings of Workshop on Cyber-Physical Systems Security & Privacy (SPC-CPS), Co-located with CCS, Oct 2016
17. Giraldo J, Sarkar E, Cardenas AA, Maniatakos M, Kantarcioglu M (2017) Security and privacy in cyber-physical systems: a survey of surveys. IEEE Design Test 34(4):7–17
18. Goh J, Adepu S, Junejo KN, Mathur A (2016) A dataset to support research in the design of secure water treatment systems. In: International Conference on Critical Information Infrastructures Security (CRITIS). Springer, pp 88–99
19. Gollmann D, Gurikov P, Isakov A, Krotofil M, Larsen J, Winnicki A (2015) Cyber-physical systems security: experimental analysis of a vinyl acetate monomer plant. In: Proceedings of the ACM Workshop on Cyber-Physical System Security. ACM, pp 1–12
20. Green B, Lee A, Antrobus R, Roedig U, Hutchison D, Rashid A (2017) Pains, gains and PLCs: ten lessons from building an industrial control systems testbed for security research. In: Proceedings of the USENIX Workshop on Cyber Security Experimentation and Test (CSET). USENIX Association
21. Hahn A, Ashok A, Sridhar S, Govindarasu M (2013) Cyber-physical security testbeds: architecture, application, and evaluation for smart grid. IEEE Trans Smart Grid 4(2):847–855
22. Holm H, Karresand M, Vidström A, Westring E (2015) A survey of industrial control system testbeds. In: Proceedings of Nordic Conference (NordSec), pp 11–26
23. Isakov A. Damn vulnerable chemical process – Tennessee eastman. https://github.com/satejnik/DVCP-TE
24. iTrust. Centre for research in cyber security. https://itrust.sutd.edu.sg/
25. iTrust. ICS testbed datasets. https://itrust.sutd.edu.sg/research/dataset/
26. Jardine W, Frey S, Green B, Rashid A (2016) SENAMI: selective non-invasive active monitoring for ICS intrusion detection. In: Proceedings of the ACM Workshop on Cyber-Physical Systems Security and Privacy (CPS-SPC), New York. ACM, pp 23–34
27. Koscher K, Czeskis A, Roesner F, Patel S, Kohno T, Checkoway S, McCoy D, Kantor B, Anderson D, Shacham H et al (2010) Experimental security analysis of a modern automobile. In: 2010 IEEE Symposium on Security and Privacy (SP). IEEE, pp 447–462
28. Liu Y, Ning P, Reiter MK (2011) False data injection attacks against state estimation in electric power grids. ACM Trans Inf Syst Secur (TISSEC) 14(1):13
29. Mathur A, Tippenhauer NO (2016) SWaT: a water treatment testbed for research and training on ICS security. In: Proceedings of Workshop on Cyber-Physical Systems for Smart Water Networks (CySWater), Apr 2016
30. McLaughlin S, Konstantinou C, Wang X, Davi L, Sadeghi AR, Maniatakos M, Karri R (2016) The cybersecurity landscape in industrial control systems. Proc IEEE 104(5):1039–1057
31. Morris TH. Industrial control system (ICS) cyber attack datasets. https://sites.google.com/a/uah.edu/tommy-morris-uah/ics-data-sets
32. Morris T, Srivastava A, Reaves B, Gao W, Pavurapu K, Reddi R (2011) A control system testbed to validate critical infrastructure protection concepts. Int J Crit Infrastruct Prot 4(2):88–103
33. pfSense. open source firewall. https://www.pfsense.org/
34. Real time digital simulator. www.rtds.com/index/index.html

35. Reaves B, Morris T (2012) An open virtual testbed for industrial control system security research. Int J Inf Secur 11(4):215–229
36. Siaterlis C, Garcia AP, Genge B (2013) On the use of emulab testbeds for scientifically rigorous experiments. IEEE Commun Surv Tutor 15(2):929–942
37. Siddiqi A, Tippenhauer NO, Mashima D, Chen B (2018) On practical threat scenario testing in an electric power ICS testbed. In: Proceedings of the Cyber-Physical System Security Workshop (CPSS), Co-located with ASIACCS, June 2018
38. Slay J, Miller M (2007) Lessons learned from the maroochy water breach. Springer, Boston
39. Taormina R, Galelli S, Tippenhauer NO, Salomons E, Ostfeld A, Eliades DG, Aghashahi M, Sundararajan R, Pourahmadi M, Banks MK, Brentan BM, Campbell E, Lima G, Manzi D, Ayala-Cabrera D, Herrera M, Montalvo I, Izquierdo J, Luvizotto E Jr, Chandy SE, Rasekh A, Barker ZA, Campbell B, Shafiee ME, Giacomoni M, Gatsis N, Taha A, Abokifa AA, Haddad K, Lo CS, Biswas P, Pasha M, Fayzul K, Kc B, Somasundaram SL, Housh M, Ohar Z (2018) The battle of the attack detection algorithms: disclosing cyber attacks on water distribution networks. J Water Res Plann Manag 144(8):04018048
40. Urbina D. The swat assault crawler. https://github.com/scy-phy/swat
41. Urbina D, Giraldo J, Cardenas AA, Tippenhauer NO, Valente J, Faisal M, Ruths J, Candell R, Sandberg H (2016) Limiting the impact of stealthy attacks on industrial control systems. In: Proceedings of the ACM Conference on Computer and Communications Security (CCS), Oct 2016
42. Urbina D, Giraldo J, Tippenhauer NO, Cárdenas A (2016) Attacking fieldbus communications in ICS: applications to the SWaT testbed. In: Proceedings of Singapore Cyber Security Conference (SG-CRC), Jan 2016
43. Weinberger S (2011) Computer security: is this the start of cyberwarfare? Nature 174:142–145
44. Williams TJ (1992) The Purdue enterprise reference architecture, a technical guide for CIM planning and implementation. Isa, Research Triangle Park
45. Zeller M (2011) Myth or reality—does the aurora vulnerability pose a risk to my generator? In: Proceedings of Conference for Protective Relay Engineers. IEEE, pp 130–136
46. Zhu B, Sastry S (2010) SCADA-specific intrusion detection/prevention systems: a survey and taxonomy. In: Proceedings of the Workshop on Secure Control Systems (SCS), vol 11