Security Development Lifecycle for Cyber-Physical Production Systems

Matthias Eckhart*†, Andreas Ekelhart*†, Arndt Lüder‡, Stefan Biffl§, and Edgar Weippl*†‖
*Christian Doppler Laboratory "SQI", Vienna, Austria, †SBA Research, Vienna, Austria, §TU Wien, Vienna, Austria,
‡Otto von Guericke University, Magdeburg, Germany, ‖St. Pölten University of Applied Sciences, St. Pölten, Austria
{matthias.eckhart, stefan.biffl, edgar.weippl}@tuwien.ac.at, andreas.ekelhart@sba-research.org, arndt.lueder@ovgu.de

***Abstract*—As the connectivity within manufacturing processes increases in light of Industry 4.0, information security becomes a pressing issue for product suppliers, systems integrators, and asset owners. Reaching new heights in digitizing the manufacturing industry also provides more targets for cyber attacks, hence, cyber-physical production systems (CPPSs) must be adequately secured to prevent malicious acts. To achieve a sufficient level of security, proper defense mechanisms must be integrated already early on in the systems' lifecycle and not just eventually in the operation phase. Although standardization efforts exist with the objective of guiding involved stakeholders toward the establishment of a holistic industrial security concept (e.g., IEC 62443), a dedicated security development lifecycle for systems integrators is missing. This represents a major challenge for engineers who lack sufficient information security knowledge, as they may not be able to identify security-related activities that can be performed along the production systems engineering (PSE) process. In this paper, we propose a novel methodology named *Security Development Lifecycle for Cyber-Physical Production Systems (SDL-CPPS)* that aims to foster security by design for CPPSs, i.e., the engineering of smart production systems with security in mind. More specifically, we derive security-related activities based on (i) security standards and guidelines, and (ii) relevant literature, leading to a security-improved PSE process that can be implemented by systems integrators. Furthermore, this paper informs domain experts on how they can conduct these security-enhancing activities and provides pointers to relevant works that may fill the potential knowledge gap. Finally, we review the proposed approach by means of discussions in a workshop setting with technical managers of an Austrian-based systems integrator to identify barriers to adopting the SDL-CPPS.**

***Index Terms*—Cyber-physical production systems, information security, security development lifecycle, security by design**

I. Introduction

The advent of the fourth industrial revolution, also known as Industry 4.0, has led to a rapid proliferation of cyber-physical production systems (CPPSs). One of the key characteristics of CPPSs is their advanced connectivity, enabling a continuous data exchange that is required for a variety of Industry 4.0 applications (e.g., predictive maintenance). However, the increased connectivity of production systems also causes the attack surface to expand; thus, making them attractive targets for cyber attacks. This issue is exacerbated if information security aspects have not been considered in the design phase, as it may result in inherently insecure CPPSs being engineered. Apparently, this is a widespread problem in the industrial sector, since a report published by Dragos, Inc. [1] indicates that 64% of the patches released in 2017 for vulnerabilities in industrial control systems (ICSs) (presumably including CPPSs) do not completely fix the issues as a result of the systems' insecure design. In other words, even if security weaknesses in CPPSs have been discovered, it seems that they cannot be eliminated via patches in a significant number of systems, leaving them potentially exploitable until end of life. Thus, the security of CPPSs must be taken into account throughout the entire lifecycle, especially in the engineering phase. Failure to do so could have devastating consequences, as cyber attacks against CPPSs may cause significant damage to machinery and can even harm human health.

In the software industry, the integration of security-related activities into the software development process is already well established, given that seminal works (e.g., [2], [3]) that introduce methodologies for the development of secure software have been published more than 10 years ago. It seems that the automation industry lags behind in this respect, which can be partly attributed to the lack of research on designing security methodologies applicable to production systems engineering (PSE). Although existing security development lifecycles for software (e.g., [2], [3]) and hardware (e.g., [4]) are certainly relevant to PSE, they do not take the full breadth of engineering disciplines that are involved in the development of production systems into account.

Thus far, only a few works [5]–[7] discuss security-improved development approaches for cyber-physical systems (CPSs). The methodologies proposed in [5]–[7] have been designed in a rather generic way, making them applicable to various CPS applications. However, the flexibility comes at the expense of being less relevant to PSE. As a consequence, engineers may not be able to derive appropriate security-activities to be performed along the PSE process. Furthermore, although the parts 2-4 [8], 3-2 [9], and 3-3 [10] of the IEC 62443 standard provide general guidance to systems integrators, these documents do not specify a security development lifecycle tailored to PSE, leaving them without the necessary support to embed security-improving activities into the integration phase.

The COMET center SBA Research (SBA-K1) is funded within the framework of COMET — Competence Centers for Excellent Technologies by BMVIT, BMDW, and the federal state of Vienna, managed by the FFG. This research was further funded by the FFG under the industrial PhD program (grant no. 874644). Moreover, the financial support by the Christian Doppler Research Association, the Austrian Federal Ministry for Digital and Economic Affairs and the National Foundation for Research, Technology and Development is gratefully acknowledged.

This paper attempts to fill this gap by synthesizing existing security approaches discussed in (i) established security standards and guidelines, and (ii) scientific works in order to describe a PSE process that treats security as a 'first-class citizen'. Although we view the safe operation of CPPSs as fundamental, we do not consider dedicated activities for ensuring safety in our work, allowing us to focus on the security aspects. Yet, we are well aware of the relationship between security and safety, i.e., achieving a sufficient security level also contributes in meeting high safety requirements.

The contributions of this paper are twofold. First, we conduct a comprehensive literature review to identify related works that cover security methodologies and concepts that may be applicable to PSE. Second, based on the literature analysis, we propose a novel methodology named *Security Development Lifecycle for Cyber-Physical Production Systems (SDL-CPPS)* that extends the PSE process by additional security-related activities. Moreover, we review SDL-CPPS via workshop-style discussions with technical managers employed by a major systems integrator based in Austria.

The remainder of this paper is organized as follows. First, Section II provides background information on PSE, ICS security standards and guidelines, and related work. Section III introduces the SDL-CPPS. Potential barriers to adopting the SDL-CPPS are discussed in Section IV. Finally, Section V concludes the paper by summarizing the findings of this work.

II. Background & Related Work

Before presenting the SDL-CPPS, we briefly describe the PSE process that systems integrators undergo when providing integration services to asset owners. Moreover, we discuss standards and guidelines applicable to PSE and review methodologies that have been proposed in existing works.

A. Production Systems Engineering (PSE)

PSE is undertaken by systems integrators as part of plant engineering and on behalf of asset owners, as it focuses on the engineering of a single CPPS. Although the details of PSE processes vary depending on the characteristics of the CPPS to be engineered, these processes tend to share the same high-level structure [11]. Based on (i) discussions with stakeholders (technical managers and engineers) involved in PSE processes for steel mills, (ii) the VDI/VDE 3695-1 [12] and VDI 2206 [13] documents, and (iii) a description of engineering workflows [11], [14], we derive a generic, high-level view of PSE (cf. Fig. 1). The PSE process can be divided into five phases, viz., (i) preparation, (ii) basic engineering, (iii) detailed engineering, (iv) integration, and (v) installation and ramp-up. These phases may overlap in time [12] and require the close collaboration of multiple disciplines [15]. Since the security-improving activities need to be seamlessly integrated into the engineering workflow, the PSE process depicted in Fig. 1 serves as a foundation for the SDL-CPPS.

B. ICS Security Standards & Guidelines Applicable to PSE

Standardization efforts in the area of ICS security have resulted in a great body of documents being published in the past years. In the following, we briefly describe selected publications and explain their relevance to CPPS integrators.

IEC 62443 is a series of standards that aim to address security issues of ICSs, which we consider a superset of CPPSs. The parts of this series are divided into four categories, viz., (i) general, (ii) policies and procedures, (iii) system, and (iv) component. Thus, the series covers security aspects that are relevant for product suppliers, systems integrators, and asset owners [16]. In particular, the parts 2-4 [8], 3-2 [9], and 3-3 [10] provide guidance for systems integrators [16]. Part 2-4 [8] defines a set of requirements that systems integrators may offer to asset owners as part of integration or maintenance services. This set comprises requirements from a broad spectrum of security areas and range from solution staffing to backup/restore. Part 3-2 [9] introduces an approach to assess security risks of ICSs, serving as a basis for partitioning the system into zones and conduits (i.e., segmenting assets based on security requirements) and, in further consequence, determining the security level target (SL-T) for each of them. The rationale behind this approach is that risks pertaining to assets within the same zone and conduit may be mitigated with countermeasures that provide a common level of security; hence, these assets can be subsumed under one SL-T [9]. Part 3-3 [10] specifies countermeasures, in connection with system requirements, that fall into five tiers of effectiveness. In this way, the standard defines capability security levels (SL-Cs), which components can provide in order to meet desired security levels (SL-Ts). In addition to parts 2-4 [8], 3-2 [9], and 3-3 [10], systems integrators may utilize the secure product development lifecycle defined in part 4-1 [16]. This lifecycle focuses on the development and maintenance of secure products that are intended to be integrated into ICSs. If systems integrators also develop automation products in-house, they can be considered as product suppliers as well, meaning that the implementation of this lifecycle is worthwhile. Along with the development of secure products, however, there is a need for a dedicated lifecycle focusing on the secure systems integration, which justifies the relevance of the SDL-CPPS.

The VDI/VDE 2182 guideline defines in sheet 1 [17] a risk-based approach that can be applied by product suppliers, systems integrators, and operators for implementing security measures. Sheets 2.{1–3} and 3.{1–3} of the guideline demonstrate how the defined approach can be applied by all three parties. Sheet 4 [18] is of particular importance for the work at hand, since it provides guidance on establishing the principles *Secure by Default*, *Security by Design*, *Security by Implementation*, and *Security by Deployment* for automation components and ICSs. Since the recommendations provided in this document are not tailored to PSE in order to make them widely applicable, systems integrators can implement SDL-CPPS supplementary to this guideline.

The NIST SP 800-82 [19] guide describes various techniques for securing ICSs. Due to its broad scope, systems integrators can use this guide as a basis for implementing certain security-enhancing measures (e.g., designing a secure ICS architecture). However, this guide does not fully address

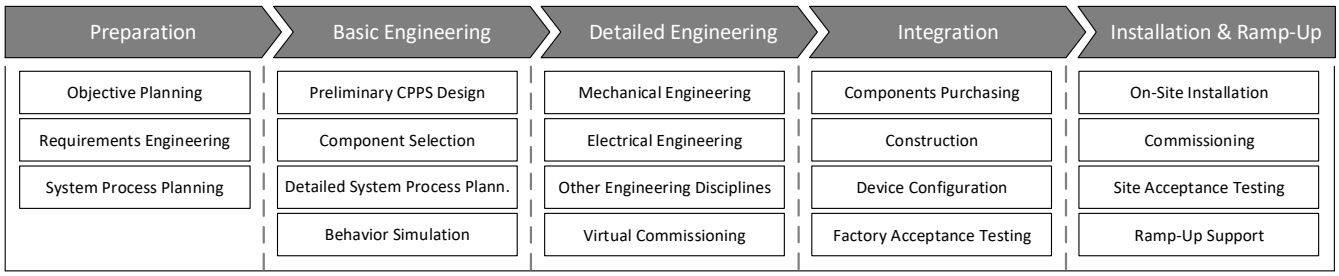

Fig. 1. A high-level view of the production systems engineering process based on [11]–[14]

the security concerns that systems integrators of CPPSs may have (e.g., in the context of automation software engineering), leaving ample room for improvement.

C. Related Methodologies

As indicated in Section I, methodologies for developing secure software have received considerable attention in the past. Microsoft's *Security Development Lifecycle (SDL)* [3] comprises security practices that can be performed by stakeholders of the software development process. A similar concept was proposed by McGraw [2] named *software security touchpoints*.

Besides software-centric security methodologies, there are also a few lifecycles with different scopes. For example, the NIST SP 800-64 [20] document focuses on information systems as a whole, while the lifecycle proposed by Khattri et al. [4] targets hardware technologies. To the best of our knowledge, [5]–[7] are the only works that provide security development approaches specifically in the context of CPSs. Schmittner et al. [5] analyze existing safety and security lifecycles, identify common activities among them and design a combined lifecycle, including both safety and security activities. In [6], Al Faruque et al. introduce a framework that focuses on the design of secure control systems for CPSs. Sun et al. [7] propose a security-enhanced design flow for CPSs, whose activities are related to the specification of security requirements, threat modeling, security design, and security verification. Since the approaches described in these works are rather generic and can be used across CPS industry sectors, the suggested activities (e.g., threat modeling, safe and secure software development) only scratch the surface of what CPPS integrators could undertake to build these systems from the ground up to be secure.

III. The Security Development Lifecycle for Cyber-Physical Production Systems

CPPSs function in the 'cyber' as well as 'physical' world and, ipso facto, must be protected against attacks emanating from both domains. Thus, we imposed the requirement that the SDL-CPPS needs to include security-improving activities that mitigate (i) cyber-to-cyber, (ii) cyber-to-physical, (iii) physical-to-physical, and (iv) physical-to-cyber attacks (cf. the CPSs attack taxonomy described in [21]). Although the physical domain introduces additional complexity that needs to be addressed (e.g., the expanded attack surface and the potential impact of attacks), physical properties can also be leveraged for designing effective technical countermeasures.

In total, we identified 14 groups of security-improving activities for PSE that have shown to be effective. These groups, including the activities that can be performed as part of the PSE steps, are summarized in Table I. It is worth mentioning that the SDL-CPPS is cyclic in nature, meaning that security efforts should not end when the CPPS is put into operation, but rather be maintained until the system's end of life (e.g., for providing ongoing security support or when retrofitting) and adopted in subsequent projects. Moreover, note that the presented list of activities provides a solid foundation for establishing security within PSE, but may not be exhaustive. In the following, we explain the SDL-CPPS in greater detail.

A. Security Governance

Security is a cross-cutting concern that affects all parts of a CPPS and, as a consequence, all parties (i.e., vendors, systems integrators, and asset owners) and stakeholders (from engineers to the CEO) involved in the system's lifecycle. To ensure that an organization's security risks are sufficiently addressed, *Policies & Procedures* must be designed and enforced, forming the basis of a security program [8], [19]. Since they must be in line with the organization's risk appetite and business needs (e.g., for meeting tender or legal requirements), the management needs to take an active part in this endeavor. Designing the security policies and procedures is not a separate activity, but rather closely intertwined with *Security Risk Management* [19]. Following the risk-based nature of the IEC 62443 series and the VDI/VDE 2182 guideline, the SDL-CPPS imposes that systems integrators have to manage security risks along the PSE process. The rationale behind this is that security risks pertaining to the developed CPPS need to be mitigated in a cost-effective manner, which is fundamental for both the systems integrator and asset owner. While systems integrators ought to take adequate proactive measures against consequences that may affect themselves (e.g., liability claims due to insecure CPPSs, industrial espionage), they also need to provide asset owners with a basis for ongoing security assessments. In particular, considering the in-depth system knowledge that integrators have, they are in the position to estimate consequences of attacks and express their impact in

quantitative terms (e.g., assessments concerning physical damages, business interruption, impact propagation, reduced safety level). The results of the security risk assessments intended for asset owners can be included in the *Documentation* along with a description of the CPPS's security features and the procedures that need to be implemented to ensure the security throughout the operational and end-of-life phase. Furthermore, it is fundamental that the systems integrator implements an organization-wide *Security Training* program to improve the overall security awareness and teach PSE stakeholders their respective security responsibilities as per the SDL-CPPS.

B. Security Planning

In the beginning of the preparation phase, *Security Objectives* need to be defined. These objectives are specific to the project at hand, concern organizational security measures, and are driven by the organization's security program as well as strategic (business) objectives. Furthermore, a high-level security risk assessment needs to be conducted in order to derive *Preliminary Security Requirements* for the CPPS to be developed. This security risk assessment is based on the general requirements of the CPPS and ought to give a preliminary view on threats and the resulting required security level. For instance, potential physical damages (e.g., owing to the plant layout), the product that the CPPS will manufacture and the CPPS's capabilities (e.g., involving assembly cells with cobots) can be determining factors in establishing initial security risk levels.

C. Secure CPPS Architecture

Similar to security planning, designing the architecture of the CPPS needs to be accompanied by a security risk assessment, which can be supported by threat modeling. Based on basic engineering artifacts detailing the preliminary CPPS structure, security risks pertaining to assets can be analyzed from a system, network, and physical perspective, albeit to a limited extent. For instance, block diagrams or piping and instrumentation diagrams (P&IDs) of manufacturing or process control systems are both valuable sources for conducting this assessment. The identified and analyzed security risks are then used to close potential architectural security gaps and to refine the *Security Requirements* defined during security planning. To give an example of the assessment scope at this stage of the PSE process, consider the design of a flexible manufacturing system (FMS) with a line or loop layout that involves multiple workstations that are connected to a central control component. Given its structure, a compromise of the central control component or a single workstation can affect all transported parts and all subsequent processing steps. Thus, it may be worth considering to compartmentalize the control of workstations, to introduce inspection stations for quality control (and detecting the consequences of malice acts early), or even to change the FMS layout to minimize potential damages on materials. After defining the security requirements of the CPPS, they need to be considered by engineers when performing the *Selection of Secure Components* for integration. However, due to insufficient offers from product suppliers, technical requirements or cost constraints, inherently insecure components may still be selected for integration. Thus, known security weaknesses in the selected components ought to be assessed and considered in the security risk assessment. After gaining a profound understanding of the components' security capabilities and the potential security risks to the CPPS, *Zone Segmentation* as per the IEC 62443-3-2 [9] needs to be performed. In essence, the assets and planned communication paths are grouped into zones and conduits based on their risks and characteristics (e.g., functionality) [9]. By applying this established security concept, adequate compensating security measures can be designed, and common security policies and controls for each zone and conduit can be enforced, ensuring that the desired security requirements are met [9], [10].

D. Secure Electrical Design

The electrical engineering step needs to be augmented with security activities in order to mitigate physical layer attacks. In particular, engineers need to obtain a *Secure Cabling Design* that not only protects against interference (e.g., caused by electromagnetic attacks) but also reduces accessibility for unauthorized personnel (e.g., to mitigate wiretapping) [19]. This security activity is strongly linked to the development of a *Secure Control Cabinet Design*. The control cabinet houses the components required for controlling physical processes and can therefore be considered as an attractive target for attacks. Thus, physical security measures, protections against electromagnetic influences, and a systematic cable routing (e.g., to facilitate inspections carried out for detecting physical backdoors) must be in place. Furthermore, *Physical Side Channel Protection* mechanisms can be designed that capture the device's power consumption or electromagnetic emission in order to detect anomalies during execution [22], or even to monitor the program control flow [23]. However, being able to leverage physical side channels for detecting cyber attacks also means, by implication, that valuable information (e.g., control logic, parametrization) may be obtained by adversaries if they are able to install sensors near these devices.

E. Security-Aware I&C Engineering

Instrumentation and control (I&C) engineers integrate instrumentation technology and develop control systems, which both constitute vital parts of every CPPS. The design of *Resilient Control Systems* represents a fundamental proactive security activity that these engineers can perform. According to Rieger et al. [24], a resilient control system is capable of maintaining state awareness (e.g., knowing that it moves into an undesirable state) and an acceptable level of performance (i.e., remaining within the boundaries of normal operation, inter alia, in terms of process stability) while being under attack. Based on this definition, the authors of [24] derive the two areas *state awareness* and *resilient control design*, which lie both in the I&C engineering field. To mitigate the loss of state awareness (e.g., due to compromised sensor nodes or communication links), redundancy and the use of

TABLE I
THE SECURITY DEVELOPMENT LIFECYCLE FOR CYBER-PHYSICAL PRODUCTION SYSTEMS

Activity	Objective Planning	Req. Engineering	Sys. Process Plann.	Prel. CPPS Design	Comp. Selection	Det. Sys. Pro. Plann.	Electrical Eng.	I&C Engineering	Software Eng.	Network Eng.	IT Engineering	Safety Engineering	Comp. Purchasing	Construction	Device Config.	FAT	On-Site Install.	Commissioning	SAT
	Preparation			Basic Eng.			Detailed Engineering						Integration				Install.		
Security Governance																			
Policies & Procedures	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
Security Risk Management	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
Documentation	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
Security Training	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
Security Planning																			
Security Objectives	●	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Preliminary Security Requirements	–	●	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Secure CPPS Architecture																			
Security Requirements	–	–	–	●	●	●	–	–	–	–	–	–	–	–	–	–	–	–	–
Selection of Secure Components	–	–	–	–	●	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Zone Segmentation	–	–	–	●	●	●	–	–	–	–	–	–	–	–	–	–	–	–	–
Secure Electrical Design																			
Secure Cabling & Control Cabinet Design	–	–	–	–	–	–	●	–	–	–	–	–	–	–	–	–	–	–	–
Physical Side Channel Protection	–	–	–	–	–	–	●	●	–	–	–	–	–	–	–	–	–	–	–
Security-Aware I&C Engineering																			
Resilient Control Systems	–	–	–	–	–	–	–	●	–	–	–	●	–	–	–	–	–	–	–
Sensor Fingerprinting & Watermarking	–	–	–	–	–	–	–	●	–	–	–	–	–	–	–	–	–	–	–
Secure Automation Software																			
Secure Coding Practices	–	–	–	–	–	–	–	●	●	–	–	–	–	–	–	–	–	–	–
Static Code Analysis	–	–	–	–	–	–	–	●	●	–	–	–	–	–	–	–	–	–	–
Secure Network Design																			
Network Segmentation	–	–	–	●	–	–	●	●	–	●	●	–	–	–	–	–	–	–	–
Industrial Wireless Security	–	–	–	–	–	–	–	–	–	●	–	–	–	–	–	–	–	–	–
Secure Remote Access	–	–	–	–	–	–	–	–	–	●	●	–	–	–	–	–	–	–	–
Technical Security Controls																			
Malware & Data Protection	–	–	–	–	–	–	–	–	●	●	●	–	–	–	–	–	–	–	–
Intrusion Detection & Prevention Systems	–	–	–	–	–	–	–	●	–	●	●	–	–	–	–	–	–	–	–
Operational Security Support																			
Asset Management	–	–	–	●	●	–	●	●	●	●	●	●	●	–	●	–	–	–	–
Patch Management	–	–	–	●	●	–	–	●	●	●	●	–	–	–	●	–	–	–	–
Testbed	–	–	–	●	●	●	●	●	●	●	●	●	●	●	●	–	–	–	–
Contingency Planning	–	–	●	●	–	●	●	●	●	●	●	●	–	–	●	–	–	–	–
Configuration Management	–	–	–	–	●	●	–	–	●	●	●	–	●	–	●	–	–	–	–
Incident Response Planning	–	–	–	●	–	●	–	–	●	●	●	●	–	–	●	–	–	–	–
Secure Setup																			
Security-Aware Configuration	–	–	–	–	–	–	–	–	–	–	–	–	–	–	●	–	–	–	–
Access Control	–	–	–	–	–	–	–	–	–	–	–	–	–	–	●	–	–	–	–
Supply Chain Security	–	–	–	–	●	–	–	–	–	–	–	–	●	●	–	–	–	–	–
Security Validation	–	–	–	–	–	●	●	●	●	●	●	●	–	–	–	●	–	–	●
Insider Threat Mitigation	–	●	●	●	–	●	–	–	●	●	●	–	–	–	●	–	●	●	–
Physical Security Planning	–	●	●	●	●	●	●	●	–	●	●	●	–	●	–	–	●	–	–

data fusion techniques are considered to be proper measures [25]. Although receiving states reflecting the (alleged) physical system behavior supports the controller in reacting to attacks, state awareness by itself does not guarantee that the controller is indeed able to maintain adequate normalcy in the face of threats [25]. Thus, a secure estimation and control strategy needs to be designed that detects malicious behavior and takes corrective action, aiming to achieve a full recovery. In recent years, researchers have shown an increased interest in designing estimators and controllers that accomplish this objective. For instance, Fawzi et al. [26] show that a reconstruction of the system states is only possible if less than half of the sensors were to be attacked, albeit they do not consider the use of a sensor data fusion technique. In addition, the authors design a local control loop that increases system resilience by allowing to correct a certain number of malicious states, provided that this local controller is not under attack. The findings of research conducted in this area do not only provide significant advances in making ICSs, such as CPPSs, more resilient to attacks, but are also invaluable for risk assessments. Interested readers are also referred to the security-aware control system design framework proposed by

Al Faruque et al. [6].

Two activity areas contributing to the resilience of CPPSs that I&C engineers can engage in is *Sensor Fingerprinting & Watermarking*. Hardware imperfections of sensors that arise naturally during fabrication can be leveraged to obtain fingerprints from the noise of sensor readings, providing the means to uniquely identify them and therefore also allow to detect physical attacks [27]. Physical watermarking, on the other hand, aims at authenticating the physical dynamics for detecting integrity attacks (e.g., replay attacks). For instance, as demonstrated in [28], this can be done by first injecting noise into the system and then examining the output for traces of the noisy input. In this context, the potential trade-off between security and performance must be carefully considered.

F. Secure Automation Software

Considering the importance of software security, this group by itself deserves its own lifecycle. As discussed in Section II-C, several software security methodologies have already been proposed and we therefore refer readers to existing literature. However, there are still some differences between automation and typical IT (business) applications in the context of software security that are worth pointing out. In particular, far too little attention has been paid to *Secure Coding Practices* and *Static Code Analysis* tools for IEC 61131-3 programming languages. The work by Valentine [29] aims to remedy this situation by providing an in-depth description of security design patterns for mitigating software vulnerabilities in programmable logic controller (PLC) programs, which have been derived from a comprehensive vulnerability analysis that even resulted in a taxonomy. While traditional software security principles (e.g., input validation) are also applicable in the realm of languages for control applications, the vulnerability taxonomy discussed in [29] shows that particular attention must be paid to logic errors, duplicate objects installed, unused objects, and hidden jumpers. Moreover, the author presents in [29] a static code analysis tool that automatically detects vulnerabilities as per the taxonomy and recommends suitable design patterns for mitigation. In addition to the use of static code analysis tools, periodic logic validation mechanisms can be implemented that aim to detect differences between the code running on the PLC and a trusted, correct version of it stored on a protected server [30]. This countermeasure has been proposed to mitigate *ladder logic bombs (LLBs)*, i.e., malicious control logic that has been implanted and deeply hidden in PLC code by adversaries [30].

G. Secure Network Design

The increased connectivity of CPPSs can be considered as one of their main characteristics. Yet, for the most part, industrial network protocols lack fundamental security features [31], leaving CPPSs vulnerable if compensating measures are missing. Furthermore, the developed CPPS needs to be securely integrated into the plant's industrial network architecture. In this context, *Network Segmentation* is a critical security measure that needs to be performed when designing the CPPS. Depending on the nature of the engineered system, this security activity may concern multiple levels of the architecture (e.g., from process to supervisory control) and therefore needs to be performed in close coordination with relevant stakeholders of the plant engineering process. Furthermore, it is worth pointing out that network and zone segmentation are considered as distinct concepts (cf., for instance, [31]). While network segmentation aims to partition the industrial network into smaller ones, the concept of zones and conduits as per the IEC 62443-3-2 [9] aims to establish groups of assets on the basis of their required security level [31] (cf. Section III-C). However, a carefully designed segmented network can constitute a solid foundation for zoning [31].

Another activity that falls within the context of achieving a secure network design is *Industrial Wireless Security*. CPPSs engineers may need to integrate wireless technologies, for example, due to mobility requirements or cost constraints. Several wireless communication protocols (e.g., WirelessHART) may be used by a variety of industrial components (e.g., sensors), each of which has its own nuances in terms of technical features and hence requires tailored security measures. Ensuring the security of wireless networks is particularly critical, as adversaries merely need to be in proximity of wireless signals to launch attacks against the wireless-enabled CPPSs. Similarly, it is of utmost importance that engineers take adequate measures to *Secure Remote Access*, due to the fact that it significantly increases the CPPS's attack surface.

H. Technical Security Controls

Following relevant standards or guidelines (e.g., NIST SP 800-82 [19]), this group comprises security activities that aim to equip the CPPS with technical security controls that protect against and respond to threats. More specifically, *Data Protection* mechanisms must be put in place to secure data (e.g., the process history) in transit as well as at rest. While software engineers may have a greater flexibility with regards to implementing data protection mechanisms in automation applications, especially for those to be deployed in upper levels of the automation pyramid, network and IT engineers are generally restricted to the security features of the devices selected for integration. Similarly, the software or hardware constraints of devices (e.g., SCADA systems) need to be considered when designing *Malware Protection* strategies, detailing the deployment of antivirus software [19]. Note that compatibility issues or general recommendations concerning the use of antivirus software may also be provided by vendors of industrial components [19].

The design of *intrusion detection and prevention systems (IDSs/IPSs)* constitutes another crucial security activity to fend off or at least detect cyber attacks against CPPSs. Since engineers maintain profound knowledge about the benign behavior that the systems they develop should exhibit, they can leverage this know-how to design behavior-specification-based IDSs. IDSs that apply this detection technique are based on a formal model that defines the correct behavior of the system and alert if the observed behavior diverges from the model, making

these IDSs powerful but generally effortful to create [32]. However, these models may already be implicitly present as a result of engineering activities and can therefore be directly used without manual effort [33], [34]. Physics-based IDSs represent another powerful way of detecting attacks, as they employ models based on the physical properties of systems to determine inconsistencies between the observed and expected systems' behavior [35]. It is worth highlighting that there are evident synergies between the identified security-activities that fall within the realm of I&C engineering (cf. Section III-E) and the design of physics-based IDSs, which may be leveraged as part of the control system development. Deploying IPSs in CPPSs requires careful consideration of availability and performance requirements, especially in lower levels of the automation pyramid [31]. Still, (semi-)automatically responding to threats, such as detaching infected automation cells (to prevent spreading of malware), or even fully preventing intrusions constitutes a building block of the SDL-CPPS.

I. Operational Security Support

The activities in this group aim to provide security support to asset owners for the operation of the CPPSs. Due to space constraints, we cannot give a thorough explanation for each of them. However, we want to stress the potential impact that they have on the security of CPPSs, provided that they are closely coordinated with asset owners. For instance, establishing a baseline for the purpose of *Asset Management* and designing discovery strategies for automated solutions (e.g., whether, when and how active scanning is viable) supports the management of CPPSs during the operation phase.

J. Secure Setup

During the integration phase, a *Security-Aware Configuration* of devices needs to be conducted in order to reduce the CPPSs' attack surface. This includes, inter alia, activating the security features of devices (e.g., protecting PLC programs) and hardening. Furthermore, setting up proper authentication as well as authorization mechanisms and managing user accounts are just a few of the tasks required for attaining a secure setup, which generally fall within the scope of *Access Control*.

K. Remaining Security Activities

The remainder of the SDL-CPPS is made up of groups whose activities can be applied in various phases of PSE.

Supply Chain Security involves the management of security risks that may arise in the supply chain of systems integrators. For instance, this may include the definition of security requirements for vendors, conducting assessments to determine whether the selected vendors indeed fulfill them, and establishing traceability in the supply chain.

Security Validation needs to be performed by engineers together with security professionals at multiple stages of PSE. While engineering artifacts have to undergo regular security reviews during basic and detailed engineering, security testing needs to be conducted in the integration and installation phase.

PSE projects are generally undertaken by large teams, some of which may even be affiliated with other organizations (e.g., subcontractors). Thus, there are typically a high number of stakeholders who are in the position to do harm to engineering projects or the resulting CPPSs. Consequently, proper measures related to *Insider Threat Mitigation* are required, particularly in PSE steps in which systems integrators may not be able to sufficiently exercise control over personnel (e.g., on-site installation via subcontractors). Insider threats pertaining to PSE projects may be mitigated by adopting recommended security practices from the nuclear security community, such as enforcing strict security procedures, access compartmentalization, or surveillance [36].

Finally, careful *Physical Security Planning* is required for several steps of PSE in order to mitigate physical security threats against the CPPS. The scope of this security activity ranges from identifying proper locations for placing the CPPS within the facility to making individual physical assets inaccessible for unauthorized personnel. Similarly to mitigating insider threats, CPPS engineers can build upon the knowledge of and recommended methods from the nuclear security community. For instance, pathway analysis can be applied in order to identify where to install detection systems and place physical barriers [37].

IV. Discussion

We conducted a workshop with three technical managers of an Austrian-based systems integrator to discuss potential hurdles toward the establishment of the SDL-CPPS. In the following, we summarize the most interesting findings.

a) Rapid Adoption is Challenging: The workshop participants stated that their clients currently deal with the implementation of security measures for the most part. However, they assume that this responsibility will gradually shift from asset owners to systems integrators within the next few years. Given that industrial engineers typically have only minimal security know-how and that engineering steps follow established procedures, the participants expect that achieving a viable security training program and adjusting engineering workflows will be a lengthy and expensive process. To gain momentum, they suggested to focus on those security activities that are inexpensive and easy to implement.

b) Lack of Adequate Tool Support: Tools that support engineers in performing the security activities of the SDL-CPPS can be considered as implementation accelerators. For instance, integrating static code analysis tools into the build pipeline typically incurs minimal effort to set up and may already yield a significant security improvement. However, adequate security-improving tools for PSE appear to be lacking.

c) SDL-CPPS as an Enabler, not an Impediment for PSE: The participants repeatedly expressed their concerns regarding justifying the additional costs incurred by performing the activities of the SDL-CPPS. The discussions gave rise to the notion that achieving more secure CPPSs may also yield quality improvements. For instance, designing resilient control systems evidently has a positive effect on the availability. This line of argumentation may not only encourage engineers to

mind security aspects, but also serve as a selling point for upper management and clients.

V. Conclusion

In this paper, we have presented a novel security-improved engineering process specifically for CPPSs, named SDL-CPPS. Adopting the SDL-CPPS may lead to a security-by-design engineering approach and, as a consequence, yield more secure and robust CPPSs. We conducted a workshop with technical managers to determine barriers to adoption of the SDL-CPPS. In particular, the discussions showed that further research and development is needed concerning tool support that would enable realizing "quick wins" in the journey toward the full implementation of the SDL-CPPS. On a final note, considering that existing plants are also modernized, we aim to investigate how the SDL-CPPS can be adapted to support security-aware retrofitting efforts.

References

[1] Dragos, Inc., "Industrial control vulnerabilities: 2017 in review," Dragos, Inc., techreport, Mar. 2018.

[2] G. McGraw, *Software security: building security in*. Addison-Wesley Professional, 2006, vol. 1.

[3] M. Howard and S. Lipner, *The Security Development Lifecycle*. Redmond, WA, USA: Microsoft Press, 2006.

[4] H. Khattri, N. K. V. Mangipudi, and S. Mandujano, "HSDL: A security development lifecycle for hardware technologies," in *2012 IEEE International Symposium on Hardware-Oriented Security and Trust*, June 2012, pp. 116–121.

[5] C. Schmittner, Z. Ma, and E. Schoitsch, "Combined safety and security development lifecylce," in *2015 IEEE 13th International Conference on Industrial Informatics (INDIN)*, July 2015, pp. 1408–1415.

[6] M. Al Faruque, F. Regazzoni, and M. Pajic, "Design methodologies for securing cyber-physical systems," in *Proceedings of the 10th International Conference on Hardware/Software Codesign and System Synthesis*, ser. CODES '15. Piscataway, NJ, USA: IEEE Press, 2015, pp. 30–36.

[7] C. Sun, J. Ma, and Q. Yao, "On the architecture and development life cycle of secure cyber-physical systems," *Journal of Communications and Information Networks*, vol. 1, no. 4, pp. 1–21, Dec 2016.

[8] IEC 62443-2-4, "Security for industrial automation and control systems – part 2-4: Security program requirements for iacs service providers," *International Standard, First Edition, International Electrotechnical Commission, Geneva*, vol. 1.1, 2017.

[9] IEC 62443-3-2, "Security for industrial automation and control systems – part 3-2: Security risk assessment and system design," *International Standard, Draft, International Electrotechnical Commission, Geneva*, vol. 1, 2018.

[10] IEC 62443-3-3, "Industrial communication networks – network and system security – part 3-3: System security requirements and security levels," *International Standard, First Edition, International Electrotechnical Commission, Geneva*, vol. 1, 2013.

[11] A. Lüder and N. Schmidt, *AutomationML in a Nutshell*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2017, pp. 213–258.

[12] VDI/VDE 3695-1, "Sheet 1: Engineering of industrial plants - evaluation and optimization - fundamentals and procedure," Berlin, 2010.

[13] VDI 2206, "Design methodology for mechatronic systems," Berlin, 2004.

[14] A. Strahilov and H. Hämmerle, *Engineering Workflow and Software Tool Chains of Automated Production Systems*. Cham: Springer International Publishing, 2017, pp. 207–234.

[15] S. Biffl, D. Gerhard, and A. Lüder, *Introduction to the Multi-Disciplinary Engineering for Cyber-Physical Production Systems*. Cham: Springer International Publishing, 2017, pp. 1–24.

[16] IEC 62443-4-1, "Security for industrial automation and control systems – part 4-1: Secure product development lifecycle requirements," *International Standard, First Edition, International Electrotechnical Commission, Geneva*, vol. 1, 2018.

[17] VDI/VDE 2182-1, "Sheet 1: IT-security for industrial automation - general model," Berlin, 2011.

[18] VDI/VDE 2182-4, "Sheet 4: IT-security for industrial automation - recommendations for the implementation of security properties for components, systems and equipment," Berlin, 2018.

[19] K. Stouffer, V. Pillitteri, S. Lightman, M. Abrams, and A. Hahn, "Guide to industrial control systems (ICS) security," *NIST special publication*, vol. 800, no. 82r2, Jun 2015.

[20] R. Kissel, K. Stine, M. Scholl, H. Rossman, J. Fahlsing, and J. Gulick, "Security considerations in the system development life cycle," *NIST special publication*, vol. 800, no. 64r2, Oct 2008.

[21] M. Yampolskiy, P. Horvath, X. D. Koutsoukos, Y. Xue, and J. Sztipanovits, "Taxonomy for description of cross-domain attacks on CPS," in *Proceedings of the 2Nd ACM International Conference on High Confidence Networked Systems*, ser. HiCoNS '13. New York, NY, USA: ACM, 2013, pp. 135–142.

[22] C. Aguayo Gonzalez and A. Hinton, "Detecting malicious software execution in programmable logic controllers using power fingerprinting," in *Critical Infrastructure Protection VIII*, J. Butts and S. Shenoi, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2014, pp. 15–27.

[23] Y. Han, S. Etigowni, H. Liu, S. Zonouz, and A. Petropulu, "Watch me, but don't touch me! contactless control flow monitoring via electromagnetic emanations," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS '17. New York, NY, USA: ACM, 2017, pp. 1095–1108.

[24] C. G. Rieger, D. I. Gertman, and M. A. McQueen, "Resilient control systems: Next generation design research," in *2009 2nd Conference on Human System Interactions*, May 2009, pp. 632–636.

[25] C. G. Rieger, "Notional examples and benchmark aspects of a resilient control system," in *2010 3rd International Symposium on Resilient Control Systems*, Aug 2010, pp. 64–71.

[26] H. Fawzi, P. Tabuada, and S. Diggavi, "Secure estimation and control for cyber-physical systems under adversarial attacks," *IEEE Transactions on Automatic Control*, vol. 59, no. 6, pp. 1454–1467, June 2014.

[27] C. M. Ahmed and A. P. Mathur, "Hardware identification via sensor fingerprinting in a cyber physical system," in *2017 IEEE International Conference on Software Quality, Reliability and Security Companion (QRS-C)*, July 2017, pp. 517–524.

[28] Y. Mo, S. Weerakkody, and B. Sinopoli, "Physical authentication of control systems: Designing watermarked control inputs to detect counterfeit sensor outputs," *IEEE Control Systems Magazine*, vol. 35, no. 1, pp. 93–109, Feb 2015.

[29] S. E. Valentine, "PLC code vulnerabilities through SCADA systems," Ph.D. dissertation, University of South Carolina, Columbia, SC, USA, 2013.

[30] N. Govil, A. Agrawal, and N. O. Tippenhauer, "On ladder logic bombs in industrial control systems," in *Computer Security*, S. K. Katsikas, F. Cuppens, N. Cuppens, C. Lambrinoudakis, C. Kalloniatis, J. Mylopoulos, A. Antón, and S. Gritzalis, Eds. Cham: Springer International Publishing, 2018, pp. 110–126.

[31] E. D. Knapp and J. T. Langill, *Industrial Network Security: Securing critical infrastructure networks for smart grid, SCADA, and other Industrial Control Systems*. Syngress, 2014.

[32] R. Mitchell and I.-R. Chen, "A survey of intrusion detection techniques for cyber-physical systems," *ACM Comput. Surv.*, vol. 46, no. 4, pp. 55:1–55:29, Mar. 2014.

[33] M. Eckhart and A. Ekelhart, "Towards security-aware virtual environments for digital twins," in *Proceedings of the 4th ACM Workshop on Cyber-Physical System Security*, ser. CPSS '18. New York, NY, USA: ACM, 2018, pp. 61–72.

[34] ——, "A specification-based state replication approach for digital twins," in *Proceedings of the 2018 Workshop on Cyber-Physical Systems Security and PrivaCy*, ser. CPS-SPC '18. New York, NY, USA: ACM, 2018, pp. 36–47.

[35] J. Giraldo, D. Urbina, A. Cardenas, J. Valente, M. Faisal, J. Ruths, N. O. Tippenhauer, H. Sandberg, and R. Candell, "A survey of physics-based attack detection in cyber-physical systems," *ACM Comput. Surv.*, vol. 51, no. 4, pp. 76:1–76:36, Jul. 2018.

[36] World Institute for Nuclear Security (WINS), "Managing internal threats," techreport 3.4, Mar. 2019, version 2.1.

[37] ——, "Modelling and simulation for nuclear security," techreport 4.7, Mar. 2019, version 2.1.