Secure Cyber-Physical Systems: Current Trends, Tools and Open Research Problems

Anupam Chattopadhyay*†, Alok Prakash*, and Muhammad Shafique∓

*School of Computer Science and Engineering, Nanyang Technological University, Singapore
†School of Physical and Mathematical Sciences, Nanyang Technological University, Singapore
∓Institute of Computer Engineering, Vienna University of Technology (TU Wien), Austria
Email: {anupam, alok}@ntu.edu.sg, muhammad.shafique@tuwien.ac.at

***Abstract*— To understand and identify the attack surfaces of a Cyber-Physical System (CPS) is an essential step towards ensuring its security. The growing complexity of the cybernetics and the interaction of independent domains such as avionics, robotics and automotive is a major hindrance against a holistic view CPS. Furthermore, proliferation of communication networks have extended the reach of CPS from a user-centric single platform to a widely distributed network, often connecting to critical infrastructure, e.g., through smart energy initiative. In this manuscript, we reflect on this perspective and provide a review of current security trends and tools for secure CPS. We emphasize on both the design and execution flows and particularly highlight the necessity of efficient attack surface detection. We provide a detailed characterization of attacks reported on different cyber-physical systems, grouped according to their application domains, attack complexity, attack source and impact. Finally, we review the current tools, point out their inadequacies and present a roadmap of future research.**

I. Introduction

Cyber-Physical Systems (CPS) are widely proliferated from minuscule bio-implantable devices to ultra-complex infrastructures. The rapid advances in computing and communication technology have enabled the cyber world to seamlessly and intelligently interface with a distributed physical world. This growth also increased our reliance on the system, which, more often than not, has been designed without security as a design goal. Consequently, the threat of an attack in various forms and scale is more real than ever. In fact, recent cases of CPS attacks include city water pipeline [1] and pacemaker [2]. While the former represents industrial CPS attack, the latter constitute attacks on consumer and healthcare CPS. Undeniably, for both CPS designer and user, security has become a prime concern. Our paper addresses this challenge with a systematic approach.

A. State-of-the-Art Approaches

There have been several works that outlined the security issues of cyber-physical systems [3][4][5][6][7]. In the discussion of potential attacks [3] observes that traditional design of robust cyber-physical systems center around random faults [7] and not the ones that can be caused by a malicious attacker. As a result, the lack of an attack model that considers the dynamics of the physical system is pointed out. In [5], authors presented a general work flow of CPS and identified the attack surfaces on this work flow during sensing, computing, communicating or actuating. Further, they proposed the idea of sensing security and using context-dependent security measures. Similarly, industrial CPS, such as smart grid and critical infrastructures' security championed the idea of ensuring security from the perspective of control systems [7], [8], [9]. CPS is often characterized by strong real-time constraints that makes them vulnerable to timing-driven attack. It was shown in [4] that security considerations do have an adverse impact on the schedule and runtime. The issue of strict resource constraints and hard runtime deadlines were also pointed out by [3][4][5] and at a roundtable discussion [10]. It was suggested in [10] to work on a language, or a feature of it, to let the designers work on the security enhancement in synchronization with other requirements of CPS. Indeed, there has been little effort, notably from the control theory perspective [11], in considering *security as a design parameter for CPS from an early design phase*.

B. CPS Security: Why It's Different?

CPS is a distributed control system with strict timing constraints consisting of both physical and cyber components. The presence of the physical interface is what makes CPS security particularly challenging. Unlike a standalone IT system, security compromise in a CPS system leads to disastrous consequences. The differences are detailed in the following.

- *Physical Interface*: The sensor and actuator interface for a CPS marks the most trivial attack surfaces [18], which also distinguishes it from IT security. An attacker can exploit the physical interface to undermine the security of a CPS without actually needing to *break* the access control mechanism. In traditional IT security that could happen only if the data is transmitted through open network. Furthermore, the control network of CPS is overlaid through the physical interfaces.
- *Control System*: CPS execute on the premise of one or multiple underlying control network, which is often integrated with a physical sensor/actuator that is markedly different from traditional IT security viewpoint. A typical example of this is implantable medical devices, which collect user data, and trigger operations in case of abnormal vital parameters. Supervisory Control and Data Acquisition (SCADA) systems are integral part of modern industrial infrastructure. Not surprisingly, the vulnerabilities in this control network remains a sweet spot for cyber attacks [6] that keeps on growing due to internet-connected SCADA systems [19].
- *Availability*: The significance of the availability breach in CPS is much more severe than a standalone digital system. An example of this is power grid attack reported in 2015 [20]. Note that for industrial control systems, the availability attack increases the attack's economic impact

978-3-9815370-8-6/17/$31.00

TABLE I
A CONCISE LIST OF REPRESENTATIVE CPS ATTACKS

Domain	Application	Source	Manifested Security Violation	Attack Cost	Impact
Consumer	Healthcare	Wireless Network [12]	Eavesdropping, Compromised Key Attack	Medium	Critical
	Automotive	Various ECUs [13]	Eavesdropping, Compromised Key Attack, Man-in-the-middle, Denial of Service	Medium	Critical
	Smart Home	Digital locks [14], ZigBee devices[15]	Eavesdropping, Denial of Service, Confidentiality	Low	High
Infrastructure	Transportation	GPS Spoof [16]	Integrity	Low	Critical
	Energy	Computing systems [3]	Denial of Service, Forced Update Attack	High	Critical
Industrial	Manufacturing Control	Virus spread through USB devices [17]	Eavesdropping, Side Channel Attacks, Resonance Attacks	High	Critical

proportionally with the duration of unavailability. On the other hand, for implantable health devices, or autonomous vehicles, it could lead to kinetic attack.

- *Timing Constraint*: Hard and soft real-time constraints form an important aspect of CPS. The execution time between an event and its corresponding response can be dictated by a hard deadline, which, if missed, may lead to failure of the complete control flow. For example, industrial smart energy monitoring systems deploy circuit breakers to estimate under/over-current. In case of a delay in detecting the surge in the current, the grid can be physically damaged, eventually causing the entire system to fail. Similar use cases are also given in Smart Production systems.
- *Socio-Technical Model*: Information security only forms a part of the larger socio-technical system security. For CPS, in particular, industrial scale systems, it is not only sufficient to define the access control but also the social and economic impacts of the security breach. This problem is less manifested in a classical information security paradigm due to its limited exposure to the physical interfaces and constraints. For CPS, this becomes especially important due to the possibility of life-threatening situations that can arise due to a security breach. A detailed discussion of these issues deserves independent study and is not covered within the scope of this manuscript though, mentioned for the sake of completeness. Interested readers may refer to [21][22] for a detailed discussion.

Naturally, the complexity of secure CPS design is relatively manageable if one concentrates on one particular application scenario or use case. This enables the designer to model the control flow, sensor behavior, timing deadlines and the network infrastructure in a much more concrete fashion. Indeed, there has been a few detailed studies on security and corresponding countermeasures for CPS that center around a specific application domain. For example, [23] presents a detailed and comprehensive analysis of different attack surfaces of an automotive. Similar studies for security of smart grid [24], [11], additive manufacturing[25], [26] and implantable medical devices [27], [28], [29] have been reported. Several representative CPS attacks are listed in the Table I.

We relate to such studies whenever appropriate in this manuscript though, we mainly highlight the generic CPS security frameworks. Further, it should be noted that, though the socio-economic aspects of secure CPS is an important domain to be addressed, we restrict our manuscript to the discussion of technical aspects only.

Organization

The rest of this manuscript is organised as following. In Section II, a generic structure of CPS is presented, for which diverse modeling approaches are discussed. Section III presents a categorization of CPS attacks along with a discussion on existing CPS security standards, while Section IV discusses different existing strategies for attack identification. The idea of security-aware design and automation has received some research momentum in recent times. These are described in the Section V. Section VI concludes the paper with an agenda of future research tasks.

II. CYBER PHYSICAL SYSTEM: GENERIC STRUCTURE

CPS consists of the following components - Sensor, Actuator, Computing, Storage and Communication. The combination of several components from this set forms a distributed control system. For each of these components, diverse kinds of attacks are feasible, of which, few representative attacks are shown in the Fig. 1.

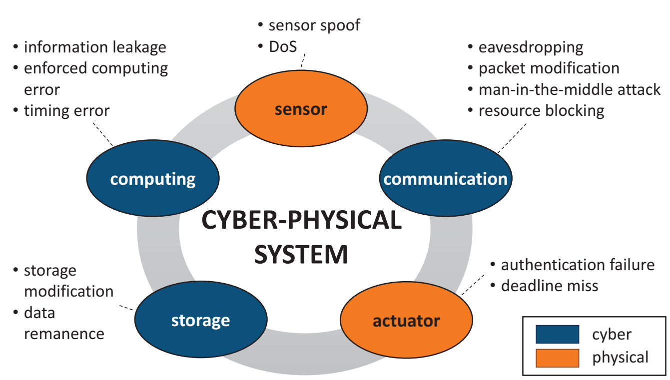

Fig. 1. Abstraction of CPS and Vulnerabilities

CPS Modeling Frameworks

Based on this generic structure, the CPS system can be modeled using the two kinds of approaches, block diagrams and equation-based object oriented languages. These modeling approaches are supported by efficient tools for e.g., formal verification and optimized code generation. An excellent survey of CPS modeling frameworks is available at [30], while the complexity of CPS modeling is discussed in [31]. Note that the modeling frameworks are based on formalisms such as, timed/hybrid automata, hierarchical state machines, differential equations and dataflow [30]. For example, VHDL/Verilog programming languages adhere to timed automata, or discrete event formalism.

Block Diagrams: Block diagrams are frequently used to describe such systems since they can be used represent both discrete and continuous events. MATLAB Simulink®[1] and Ansys[2] are examples of commercial tools that use differential equations and discrete time difference equations for continuous and discrete events respectively. Scicos[3] from Scilab environment provides an open source alternative to Simulink. For the physical components of CPS especially to efficiently solve partial differential equations, COMSOL[4] can be used, which can be integrated to Simulink. Another prominent modeling language for CPS is Architecture Analysis & Design Language (AADL), which has its roots in Avionics systems modeling. Recent literature reports the extension AADL to model the physical components [32]. An open-source simulator that supports diverse range of models is presented in [33].

Equation-based Object Oriented (EOO): As opposed to the block diagram specifications, EOO languages are *non-causal*, which implies that the flow of information is not specified beforehand. Modelica is an example of an EOO language that can be used to model and simulate both the cyber and physical portions of CPS. Another example of non-causal EOO language is Functional Hybrid modeling (FHM) in which functions are used to model the composition. The main advantage of EOO languages over Block diagrams is that the topology of the real physical systems and the models is similar that makes model reuse easier. Modelica can be integrated with other third-party components in a complex system via standardized Functional Mock-up Interface (FMI). Such an effort for smart grid modeling is reported in [34].

III. CPS Attacks and Security Standards

In this section, we present a classification of CPS attacks. Furthermore, a detailed discussion on the current CPS security standards is also presented.

A. Attack Definition and Classification

An attack towards a CPS is a deviation from an anticipated/specified execution flow, which breaches either of these security objectives - *Confidentiality* of data/execution flow, *Integrity* and *Authenticity* of command and message, and *Availability* of a functional system. The attack can originate anywhere in the system, and even though it does not manifest into an externally visible impact, it is considered an (foiled) attack. One may use the CPS modeling frameworks for an attack simulation, where the socio-technical attack sources are restricted to purely technical origins only.

Note that, the aspect of IP piracy is sometimes integrated with security analysis since, IP piracy relies on reverse engineering techniques. Reverse engineering is a useful step for confidentiality breach via side-channel analysis, inserting Trojan hardware, and piggybacking malware.

B. Security Standards for CPS

It is a well-understood fact that perfect security is extremely difficult, if not an impossible goal. Nevertheless, to minimise the effect of cybersecurity incidents, one may try to increase the cost of an attack and reduce the impact of it, in case of eventuality. To facilitate the comparison between different standards and also allow a holistic view of security, we define the Complexity-Impact Ratio (CIR), which is expressed as:

$$CIR = \frac{complexity_{attack}}{impact_{attack}},$$

where the metric of complexity and attack can be defined in terms of duration, monetary cost, manpower or a combination of these. Current cyber security standards provide standards for information security and critical infrastructure security. Well-known information security standards like ISO/IEC 27002 [35] provide the basic template, which is adopted by different national standards. It includes recommendations for managing critical assets, information, intervention of humans, access control, underlying cryptographic primitives among others.

Security of critical infrastructure is governed by standards such as North American Electric Reliability Corporation's (NERC) Critical Infrastructure Protection (CIP) Cyber Security Standards. CIP [36] standard mandates an electronic security perimeter for critical assets, and further puts in recommended practices for identification and recovery plans for cybersecurity incidents. Of late, there is also a rise in attacks particularly against IoT devices, e.g., IP cameras used in smart home networks fall prey to this. There is a recent draft of recommendations from NIST [37] to caution against such attacks but, these are yet to be standardized. Clearly, *there is an urgent need for defining CPS security standard that combines the recommendations from both of these with a goal to maximise the CIR*, and incorporate new attacks in a watchlist. In the following, such recommendations are made.

- *Infrastructure and Industrial Control Systems:* First, Increasing digitization of infrastructure via e.g., smart manufacturing requires adoption of digital security standards to infrastructural CPS. Second, physical side-channels, e.g., fault injection attacks, Trojans, information leakages has to be considered. Exemplarily 3D printers, considered to be a mainstay of smart manufacturing, are susceptible to side-channel attacks [26]. Third, manufacturing of components for large-scale infrastructure needs to adhere to secure supply chain management principles.
- *Consumer CPS:* First, the modern age of communication paved way for Internet-of-Things (IoT), which caters to a consumer with distributed but, connected devices, running on separate protocols. Security standards need

[1] https://ch.mathworks.com/products/simulink/
[2] http://www.ansys.com
[3] http://www.scicos.org/
[4] https://www.comsol.com/

to adapt for distributed, ad-hoc and cross-protocol environments. Second, due to the perceived lower impact for consumer devices, the protection measures are relatively lightweight. However, for consumer devices that has significant autonomy and criticality, e.g., wearable and implanted health devices, the CIR is low [38]. A quantifiable CIR threshold definition will be of much practical importance.

These concerns, which are often exacerbated by the CPS manufacturers' inability to address the security challenges, led to an exemption of copyright [39]. Essentially, this allows the software in an Electronic Control Unit (ECU) to be examined by the owner of a CPS to determine flaws and mitigate it, if identified, independently. Such an approach paves way for security auditing, with eventual growth of security-as-a-service (please refer subsection V-C for more details) in CPS. Along the same lines, Food and Drug Administration (FDA) in US has released guidance for management of devices in case of cyber security threat [40].

IV. Modeling the Attack Flow

A systematic attack flow modeling is only possible by adopting a reference system model and identifying the attack sources as well as attack manifestation path therein. This approach, in contrast to in-depth domain-specific evaluations like [23], allows one perform an early-stage vulnerability analysis. This kind of technique is commonplace in malware detection, dubbed in early literature as attack tree [41]. In a more elaborate effort, a system-wide flow of information is captured in a so called *taint graph* [42]. For a given source of anomaly, the taint propagates in a directed graph.

A. Control Attack

Clearly, this analysis is inadequate to capture the complex nature of attacks in a CPS. In [6], authors viewed the attacks from two perspectives - from the implementation model, e.g., network/device/storage attacks, and from the control theory viewpoint. In the first case, the attacks are again classified into several categories. First, software attacks, which include OS attack, buffer overflow attack and database attack. Second, the attacks on the communication stack across different layers. From the control theory perspective, the attacks are divided into deadline miss attack, Denial-of-Service (DoS) attack, unauthenticated actuator control attack and sensor spoofing attack. This classification, though introduced with primarily SCADA network in mind, remains highly relevant and can easily be applied to identify the attack surfaces in another domain, e.g., Controller Area Network (CAN) bus in an automotive.

B. Deadline Violation Attack

Another method of identifying attack sources is presented in [43]. A specific example is shown in the Fig. 2. The proposed Attack Sequence Diagram (ASD) illustrate the attack propagation from an event (caused by the attacker) leading to a deadline miss. Here, the deadline miss for task A is caused by a change in the priority during the *load_priority* event.

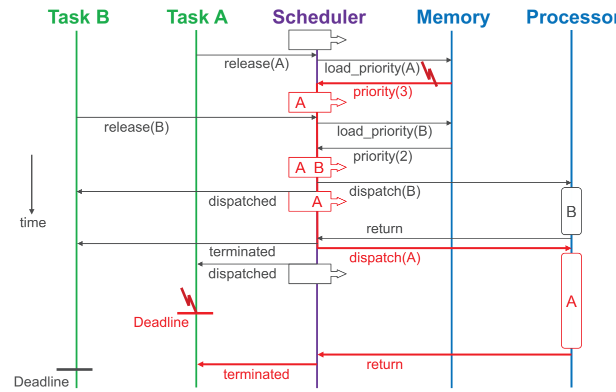

Fig. 2. Attack Simulation using Attack Sequence Diagram (ASD) [43]

C. Sensor/Actuator Attack

Similarly, for identifying the physics-based attacks via the sensors/actuators, particularly for industrial control systems, two different strategies are pointed out in [7]. These are termed, *stateful* and *stateless* anomaly detection tests.

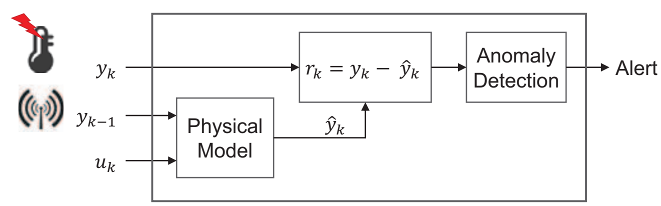

Fig. 3. Anomaly Detection in a Physical System [7]

The key idea is to predict the next sensor/actuator outcome on the basis of previous observations, as shown in the Fig. 3, where y_k and y_{k-1} denotes the sensory data, u_k denotes the control command and $\hat{y}_k$ denotes the predicted sensory data. When the new observation is significantly different, an alert is generated. The key challenge in this approach is to minimize false positives, which is achieved by accurate modeling of the physical behavior, e.g., patient insulin infusion pattern, electricity consumption pattern and GPS values for unmanned aerial vehicles [44]. For an intelligent attacker, it is possible to spoof the sensor data in such a way that it mimics the expected outcome including high-level protocol specifications. Hence, a lot of research work has been reported that studied passive and active monitoring [45], distributed/centralized attack detection filters [11] and domain-specific accurate physical modeling. An excellent survey of such attacks and countermeasures are presented in [7]. It is interesting to observe the similarity of such attack detection with behavioral anomaly prediction in software security and concurrent error detection methods for fault injection attack in hardware [46].

D. Attack Source Classification

We propose the following categorization of the attack sources in a CPS. For each of these attack sources, the attack can be mounted by an adversary through *cyber* (malicious software/hardware trojan) or *physical* (information leakage, fault injection) means. For each case, we highlight some sample attacks.

Control/Data flow: This kind of attack can be mounted in communication network, e.g., Modbus/TCP protocol stack or SCADA network, or in the computing platform. This could be carried out through malware or physical fault injection. A control flow attack at the computing device (e.g., programmable logic controller) would be modification of the instruction control flow.

Storage: For a cyber attack, the storage can be made unavailable through DoS attack, or a false data can be injected through SQL injection attack. If the attacker gets physical access to the storage, sensitive information can be read through data remanence attack.

Sensor/Actuator: Manipulating a sensor/actuator potentially represents a hazardous attack source, first, due to the difficulty in monitoring its behavior and second, due to its immediate impact in the physical world with lowered possibility to contain the attack within the system. Such attacks can be non-invasive and potentially hazardous as exemplified in [47].

For every attack category, an abstract view from system-level or a detailed view from the implementation level is admissible. This adjustable abstraction eventually facilitates development of security-aware design flows. Further note that, these attack sources can be exploited to eventually trigger an erroneous execution in the control network, sensor/actuator spoofing, unavailability of critical resource, a timing misalignment or a combination of these failures.

V. Security-aware CPS Design Flow

Compared to the advances in vulnerability identification and countermeasure propositions, the research in security-aware CPS design is still at a nascent phase. This is partly due to the fact that security threats are often hard to model at high-level abstractions. The most common approach for security-aware design flow is to include a feedback loop at high-level design flow, which simulates/emulates attacks [48] and validates the efficiency of the countermeasures. On the other hand, the recommendations from NIST [37] is that - *"there must be a level of confidence in the feasibility and correctness-in-concept, philosophy, and design, regarding the ability of a system to function securely as intended"* - indeed, it calls for a design approach that ensures secure execution.

A. Building Trusted CPS from Untrusted Components

The complex design flow of CPS heavily relies on multiple different tool flows, including a complex supply-chain management, manufacturing and testing process. Specifically, the heterogeneous nature of CPS components often include library of available IPs as shown in the Fig. 4. There is a large body of work to rule out hardware Trojan insertion [49][50] and mitigate attacks via Design-For-Testability (DFT) constructs [51][52]. In contrast, system-level integration of untrusted components or malicious design modification at higher level of abstraction are few and rare.

B. Reverse Engineering

In the following Fig. 5, different design abstractions are shown with some of them marked red using the Y-Chart. The marked abstractions are demonstrated to be target of

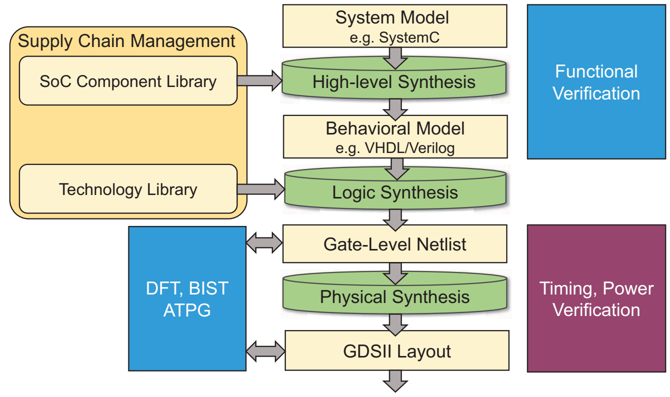

Fig. 4. Electronic Design Automation Flow for Digital Systems

reverse engineering. Also note that, these abstractions are exchanged as soft/hard IPs among vendors. Reverse engineering plays a crucial role in security-aware CPS design flow for several reasons. First, it is directly related to IP piracy. Second, by preventing reverse engineering, it also reduces the possibility of Trojan hardware insertion. Finally, several side-channel attacks heavily depends on the identification of physical location for different computing/storage blocks. For RTL constructs, design obfuscation [49] is proposed to combat reverse engineering, which is yet to be ported to CPS.

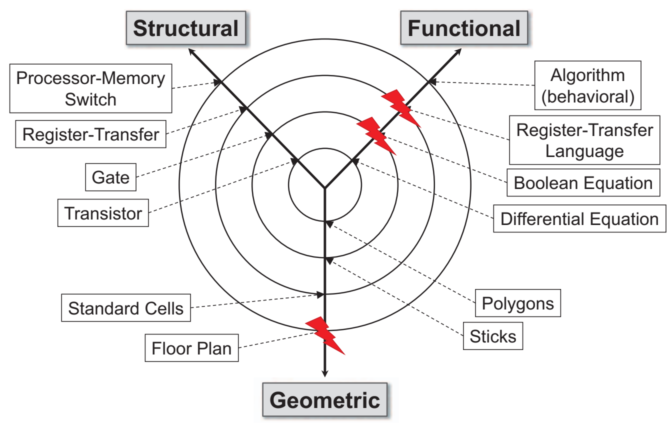

Fig. 5. Reverse Engineering at Different Abstractions

C. Security-As-A-Service (SECaaS) for CPS

There is already a multibillion-dollar industry that is serving diverse industries the service of security, e.g., for internet security, data security and cloud security. The challenge of secure CPS has to be partly addressed by similar Security-As-A-Service (SECaaS) model for CPS. Already available examples for this includes, security auditing of CAN bus in an automotive network [48] and rapid detection of attack in a large-scale wireless sensor network [53]. In fact there are several commercial offerings, which target sector-specific security, e.g., Argus[5] for automotive security and Bitdefender[6] Box for IoT security.

[5]https://argus-sec.com/
[6]http://www.bitdefender.com

VI. Conclusion and Roadmap

This paper presented an overview of state-of-the-art secure CPS design and execution, including recent trends and tools to combat the continuously growing threats. The problem is going to be increasingly relevant due to the rapid adoption of technologies like IoT, smart home/automotive/energy and in general deeper interaction between the cyber and physical world. We presented a brief survey of CPS attacks, recommendations for CPS security standards and followed up with a discussion of attack flow modeling. A classification of attack types and attack sources/surfaces is also presented. Finally, the challenges related to secure CPS design flow are reviewed.

Plethora of open research problems in secure CPS design do exist. Some of these are highlighted in the following.

1) How to include security as a quantifiable design metric in early-stage CPS modeling frameworks?
2) How to model complex and hybrid attack flows?
3) How the attack surface and attack flow be combined to determine the Complexity Impact Ratio (CIR)?
4) How to develop a Design-For-Security-Testing (DFS) construct for static and dynamic security auditing?
5) How to create a comprehensive test infrastructure with the notion of *security coverage*?

References

[1] Verizon Data Breach Digest. http://www.verizonenterprise.com/resources/reports/rp_data-breach-digest_xg_en.pdf.
[2] M. Moe. Go Ahead, Hackers. Break My Heart. https://www.wired.com/2016/03/go-ahead-hackers-break-heart/.
[3] A. Cardenas *et al.*, "Challenges for securing cyber physical systems," in *Workshop on future directions in cyber-physical systems security*, 2009.
[4] F. Pasqualetti *et al.*, "Design and operation of secure cyber-physical systems," *IEEE ESL*, 2015.
[5] E. K. Wang *et al.*, "Security issues and challenges for cyber physical system," in *GREENCOM-CPSCOM*, 2010.
[6] B. Zhu *et al.*, "A taxonomy of cyber attacks on scada systems," in *ITHINGSCPSCOM*, 2011.
[7] D. I. Urbina *et al.*, "Survey and new directions for physics-based attack detection in control systems," 2016. [Online]. Available: https://dx.doi.org/10.6028/NIST.GCR.16-010
[8] S. Huang *et al.*, "Cyber-physical system security for networked industrial processes," *Springer IJAC*, 2015.
[9] D. Gollmann *et al.*, "Cyber-physical systems security: Experimental analysis of a vinyl acetate monomer plant," in *ICCPS*, 2015.
[10] S. Peisert *et al.*, "Designed-in security for cyber-physical systems," *IEEE Security Privacy*, 2014.
[11] F. Pasqualetti *et al.*, "Attack detection and identification in cyber-physical systems," *IEEE Transactions on Automatic Control*, 2013.
[12] N. Leavitt, "Researchers fight to keep implanted medical devices safe from hackers," *Computer*, 2010.
[13] K. Koscher *et al.*, "Experimental security analysis of a modern automobile," in *SP*, 2010.
[14] The Terrible Security of Bluetooth Locks. http://hackaday.com/2016/08/08/the-terrible-security-of-bluetooth-locks/.
[15] Researchers exploit ZigBee security flaws that compromise security of smart homes. http://www.networkworld.com/article/2969402/microsoft-subnet/researchers-exploit-zigbee-security-flaws-that-compromise-security-of-smart-homes.html.
[16] M. L. Psiaki *et al.*, "Protecting gps from spoofers is critical to the future of navigation," *IEEE Spectrum*, 2016.
[17] Virus Attacks Siemens Plant-Control Systems. http://www.wsj.com/articles/SB10001424052748703954804575381372165249074.
[18] M. Harris. (2015) Researcher hacks self-driving car sensors. http://spectrum.ieee.org/cars-that-think/transportation/self-driving/researcher-hacks-selfdriving-car-sensors.
[19] (2015) Dell security annual threat report. https://software.dell.com/docs/2015-dell-security-annual-threat-report-white-paper-15657.pdf.
[20] (2016) "analysis of the cyber attack on the ukrainian power grid". https://ics.sans.org/media/E-ISAC_SANS_Ukraine_DUC_5.pdf.
[21] R. Anderson *et al.*, *Measuring the Cost of Cybercrime*, 2013.
[22] K. Y. Lam, "IoT Security: Cybersecurity from IT to OT," in *Navigating the Digital Age: Singapore*, 2016.
[23] S. Checkoway *et al.*, "Comprehensive experimental analyses of automotive attack surfaces," in *SEC*, 2011.
[24] W. Wang *et al.*, "Survey cyber security in the smart grid: Survey and challenges," *Comput. Netw.*, 2013.
[25] M. Backes *et al.*, "Acoustic side-channel attacks on printers," in *USENIX Security*, 2010.
[26] M. A. Al Faruque *et al.*, "Acoustic side-channel attacks on additive manufacturing systems," in *Proceedings of the 7th International Conference on Cyber-Physical Systems*, 2016.
[27] J. Sametinger *et al.*, "Security challenges for medical devices," *Commun. ACM*, 2015.
[28] W. Burleson *et al.*, "Design challenges for secure implantable medical devices," in *DAC*, 2012.
[29] D. Halperin *et al.*, "Pacemakers and implantable cardiac defibrillators: Software radio attacks and zero-power defenses," in *SP*, 2008.
[30] D. Broman *et al.*, "Viewpoints, formalisms, languages, and tools for cyber-physical systems," in *MPM*, 2012.
[31] P. Derler *et al.*, "Modeling cyber-physical systems," *Proceedings of the IEEE*, 2012.
[32] E. Ahmad *et al.*, "Hybrid annex: An aadl extension for continuous behavior and cyber-physical interaction modeling," *Ada Lett.*, 2014.
[33] E. A. Lee *et al.*, "Modeling and simulating cyber-physical systems using cyphysim," in *EMSOFT*, 2015.
[34] J. Vaubourg *et al.*, *Multi-agent Multi-Model Simulation of Smart Grids in the MS4SG Project*, 2015.
[35] (2013) Iso/iec 27002:2013 standard. http://www.iso.org/iso/catalogue_detail?csnumber=54533.
[36] Nerc critical infrastructure protection standard. http://www.nerc.com/pa/Stand/Pages/CIPStandards.aspx.
[37] R. Ross *et al.*, "Systems security engineering: Considerations for a multidisciplinary approach in the engineering of trustworthy secure systems," 2016. [Online]. Available: http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-160.pdf
[38] A. Newitz. (2006) The rfid hacking underground. https://www.wired.com/2006/05/rfid-2/.
[39] Library of Congress, Copyright Office, "Exemption to Prohibition on Circumvention of Copyright Protection Systems for Access Control Technologies", [Docket No. 2014?07]. http://copyright.gov/fedreg/2015/80fr65944.pdf.
[40] (2016) Postmarket management of cybersecurity in medical devices. http://www.fda.gov/downloads/MedicalDevices/DeviceRegulationandGuidance/GuidanceDocuments/UCM482022.pdf.
[41] C. Salter *et al.*, "Toward a secure system engineering methodolgy," in *NSPW*, 1998.
[42] H. Yin *et al.*, "Panorama: Capturing system-wide information flow for malware detection and analysis," in *CCS*, 2007.
[43] A. Easwaran *et al.*, "A systematic security analysis of real-time cyber-physical systems," in *ASP-DAC*, 2017.
[44] A. J. Kerns *et al.*, "Unmanned aircraft capture and control via gps spoofing," *J. Field Robot.*, 2014.
[45] K. R. Davis *et al.*, "Power flow cyber attacks and perturbation-based defense," in *SmartGridComm*, 2012.
[46] W. He *et al.*, "Bypassing parity protected cryptography using laser fault injection in cyber-physical system," in *CPSS*, 2016.
[47] Y. Shoukry *et al.*, "Non-invasive spoofing attacks for anti-lock braking systems," in *CHES*, 2013.
[48] A. Lebrun and J.-C. Demay. (2016) Canspy: a platform for auditing can devices. https://www.blackhat.com/docs/us-16/materials/us-16-Demay-CANSPY-A-Platorm-For-Auditing-CAN-Devices.pdf.
[49] R. S. Chakraborty *et al.*, "Security against hardware trojan attacks using key-based design obfuscation," *Springer JETTA*, 2011.
[50] T. F. Wu *et al.*, "TPAD: hardware trojan prevention and detection for trusted integrated circuits," *CoRR*, 2015.
[51] J. Lee *et al.*, "A low-cost solution for protecting ips against scan-based side-channel attacks," in *VTS*, 2006.
[52] Y. Huang *et al.*, "Trace buffer attack: Security versus observability study in post-silicon debug," in *VLSI-SoC*, 2015.
[53] Y. Shoukry *et al.*, "Sound and complete state estimation for linear dynamical systems under sensor attacks using satisfiability modulo theory solving," in *ACC*, 2015.