2022 The 22nd International Conference on Control, Automation and Systems (ICCAS 2022)
BEXCO, Busan, Korea, Nov. 27~Dec. 01, 2022

A Resiliency Coordinator Against Malicious Attacks for Cyber-Physical Systems

Yongsoon Eun[1]*, Jaegeun Park[1], Yechan Jeong[2], Daehoon Kim[1], and Kyung-Joon Park[1]

[1]Department of Electrical Engineering and Computer Science, DGIST,
Daegu, 42988, Korea ({yeun,jaegeun2, dkim, kjp}@dgist.ac.kr)
[2] Research and Development Division, Hyundai Motor Company,
Hwaseong, 18280, Korea (yjeong@hyundai.com) * Corresponding author

Abstract: Resiliency of cyber-physical systems (CPSs) against malicious attacks has been a topic of active research in the past decade due to widely recognized importance. Resilient CPS is capable of tolerating some attacks, operating at a reduced capacity with core functions maintained, and failing gracefully to avoid any catastrophic consequences. Existing work includes an architecture for hierarchical control systems, which is a subset of CPS with wide applicability, that is tailored for resiliency. Namely, the architecture consists of local, network and supervision layers and features such as simplex structure, resource isolation by hypervisors, redundant sensors/actuators, and software defined network capabilities. Existing work also includes methods of ensuring a level of resiliency at each one of the layers, respectively. However, for a holistic system level resiliency, individual methods at each layers must be coordinated in their deployment because all three layers interact for the operation of CPS. For this purpose, a resiliency coordinator for CPS is proposed in this work. The resiliency coordinator is the interconnection of central resiliency coordinator in the supervision layer, network resiliency coordinator in the network layer, and finally, local resiliency coordinators in multiple physical systems that compose the physical layer. We show, by examples, the operation of the resiliency coordinator and illustrate that RC accomplishes a level of attack resiliency greater than the sum of resiliency at each one of the layers separately.

Keywords: Cyber-Physical Systems, Malicious Attacks, Resiliency Coordinator, Hierarchical Control Systems.

1. INTRODUCTION

Cyber-physical systems (CPSs) refer to architectures where physical and computing components are tightly integrated by wire/wireless communication networks [1–3]. Almost all modern complex systems, such as power grid, metros, airplanes, and vehicles that involve the use of computers fall into this description. An essential element of the CPS is that through sensors and actuators, the cyber components, i.e., computers *do* interact with physical systems. Such arrangements have been developed over an extended period of time, beginning well before the term CPS was coined. The trend has only been accelerated with the advancement of communication, automatic control, and real-time computing technologies.

From the very idea of computers affecting the physical systems such as cars and airplanes, a critical concern has been computer malfunctions due to various reasons. Even a seemingly minor miscalculation in the computing unit, e.g., inconsistent units used for computing variables, may result in a significant failure for the connected physical systems. Apart from innocent malfunctions or faults, miscalculations induced by adversaries with malicious intent have been one of the major concerns in the past decade. The concern became a reality as shown by the incident of Stuxnet in 2010 [4, 5]. Specifically, Stuxnet is a computer virus-like pieces of code that targets supervisory control and data acquisition systems (SCADA) of the Iranian nuclear facility. Stuxnet intrudes on the SCADA system through its networked communication and manipulates the output of programmable logic controllers (PLCs) leading to the malfunctioning of the nuclear facility. Moreover, while changing the output of PLCs, the Stuxnet hides the physical damage to the main control center by feeding the previously recorded data, which is obtained by recording data received from PLCs to the SCADA's monitoring systems. It has been observed that conventional security solutions, e.g., cryptography and intrusion detection systems, are insufficient to protect the CPS from such attacks. Other incidents followed [6, 7].

The term *resilience* CPS implies systems that tolerate a certain level of malicious external attacks or internal faults, that secure core functions if full operation is not possible, and that degrade gracefully when inevitable [1–3, 8–11]. Clearly, resiliency becomes more important as CPS becomes more complicated with the increasing number of interconnected components because each one of them becomes a potential target by attacks [12, 13]. Furthermore, the wire/wireless communication network can be exploited by adversaries to maliciously manipulate the data exchanged between components or compromise computing systems.

Recent work of [11] substantiates the idea of resilient CPS by narrowing down the considered CPS to what is known as hierarchical control systems and by proposing an architecture tailored for resiliency by design. In [11], the considered CPS consists of three layers: physical layer where multiple agents (physical plant and feedback controllers) belong to, supervisory layer where central supervising functions reside for involved agents, and network layer responsible for communication between the two layers. See Figure 1. It is pointed out that existing metro systems, to-be intelligent transportation systems where cars and infrastructure are connected for semi- and/or full autonomous operation, and a fleet of drones

978-89-93215-25-0/22/$31.00 ©ICROS

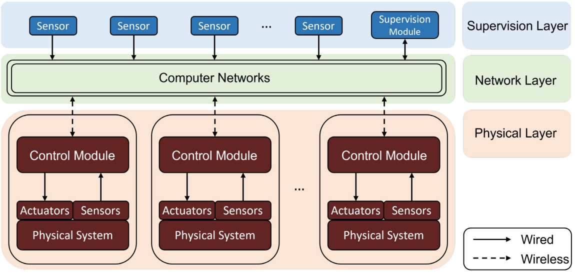


Fig. 1. An instantiation of CPS: Hierarchical control systems [11].

deployed for civil and military missions all fall in the class of hierarchical control systems.

The tailoring for resiliency of Figure 1, proposed in [11], is shown in Figure 2. Here, features to support resiliency at each one of the layers are proposed: agents in the physical layers adopt simplex structure [14] where high assurance (HA) controller and high performance (HP) controller co-exist. For the computing hardware where HP resides, hypervisors or the likes exist for computing resource isolation in case of failures. Also, redundant sensors/actuators are considered to support resilient operation with appropriate algorithms. Network layer is designed with software defined network capabilities, so that attack resilient algorithms such as network level physical property based anomaly detection are implemented with flexibility. Supervisory layer collects operation data from all agents, which inherently provides a level of information redundancy that can be exploited if necessary. In addition, [11] illustrates several resiliency algorithms residing in each one of the three layers and demonstrates resilient operation scenarios.

What this paper proposes is resiliency coordinator (RC) for the system shown in Figure 2, which fully exploits the resilient architecture designed in from the outset. It enables attack detection and toleration in a coordinated way by the three layers so that attack scenarios that are not able to be defended without the coordination are able to be tolerated. This is the main contribution of the work: we propose Resiliency Coordinator, a software that runs on the architecture of 2 and fully exploits the features for ensuring the resiliency from a holistic point of view.

The outline of the paper is as follows: Review of the related work is given in Section 2, the construction and operation of Resiliency Coordinator are given in Section 3, illustrating examples are discussed in Section 4, and finally Section 5 formulates the conclusions.

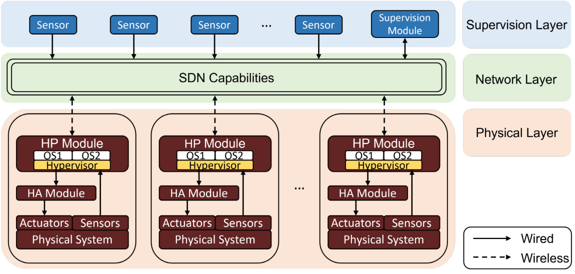


Fig. 2. Attack-resilient CPS architecture proposed in [11] for hierarchical control systems.

2. RELATED WORK

Research for resilience enhancement of CPS has received attention for a long time. Reference [15] proposes a hierarchical security architecture and emphasizes the need for resilience algorithms considering the interdependency between layers, which are introduced in [16] and [17]. Reference [16] proposes a cross-domain security analysis framework for novel cross-domain attack models and attack detection methods, and a cross-layer framework for CPS relying on the connections between the cryptography with control theory and leveraging game theory is introduced in [17]. Reference [12] introduces a cross-layer security framework for smart grid, where the concept of mobile agents are responsible for interlayer communication and data exchange tasks to enable cross-layer interaction between layers in real-time. However, the validation of cross-layer security via simulation is not performed. Reference [18] proposes contract-based hierarchical resilience management for CPS, where the hierarchical architecture of resilience managers based on contracts allows multiple components to flexibly respond to a failure of one component. This framework, however, deals with only component failures. In [19], a distributed multi-agent framework for resilient CPS is proposed. This framework enhances the resilience for complex CPS, consisting of a diversity of distributed physical devices, in the context of heterogeneous communication networks. Especially, master agents in such a multi-agent framework that are responsible for guaranteeing that subordinate agents are working properly provide the resilience of CPS from a holistic perspective. However, they do not deal with situations where the system is gracefully degraded when inevitable.

3. RESILIENCY COORDINATOR

3.1 Purpose

The resiliency coordinator is a piece of software that runs on the architecture to support attack detection and response to attack in a systematic and coordinated way. It does book keeping of when an attack occurs and how long it persists and does book keeping of which countermeasures (e.g., algorithms for resiliency) are already deployed or reserved for future use. Most importantly, it coordinates the deployment of multiple countermeasures from different layers to tolerate those attacks that are not tolerable without coordination.

It has been pointed out that resilient CPS is capable of tolerating a certain level of malicious external attacks or internal faults, capable of securing core functions if full operation is not possible, and capable of degrading (failing) gracefully when inevitable. All three capabilities are of system level behaviors: success or failure of attack detection and resilient algorithm deployment in a single layer is insufficient to determine which of the three behaviors the system shall operate by in the given circumstances. Such decision making is feasible by the role of RC. RC maintains the holistic view of countermeasure deployment at each layer and decides which behavioral mode the CPS shall operate in. Detailed descriptions are given in the subsequent subsections.

3.2 Construction and operation

The resiliency coordinator is the interconnection of central resiliency coordinator (CRC) in the supervision layer, network resiliency coordinator (NRC) in the network layer, and finally, local resiliency coordinators (LRCs) in multiple physical systems that compose the physical layer. Figure 3 shows the presence of CRC, NRC, and LRC with the communication between them. At this point, the communication between NRC and CRC, and between multiple LRCs and CRC are assumed to be secure by some means.

The three types of resiliency coordinators continuously monitor attacks on each layer with attack detection response (ADR) tables. An instantiation of the ADR table for LRC is shown in Figure 4. A row is assigned for an attack and a column is assigned for an algorithm equipped in the agent where LRC resides. Capabilities of an algorithm for an attack are described by 'Detection' if the algorithm is able to detect the said attack, by 'Response' if the algorithm is able to counteract the attack or nullify the effect, and by 'Detection & Response' if the algorithm does both. Figure 4 lists five attacks and three algorithms as an abstracted example. Algorithm 1 is able to detect Attack 1 and counteract, is also able to only detect Attack 2, but is irrelevant with Attacks 3 to 5. Algorithm 2 detects Attack 3, and Algorithm 3 can react to Attack 3. No countermeasure exists for Attacks 2, 4 and 5.

We point out that this way of maintaining attack detection and response capability in tabular form is scalable. When new attacks are discovered or speculated, new rows are added to the ADR table. New algorithms add corresponding columns to the table and the capabilities are listed where the row and column meet.

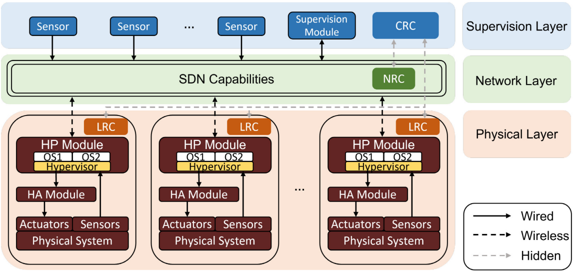
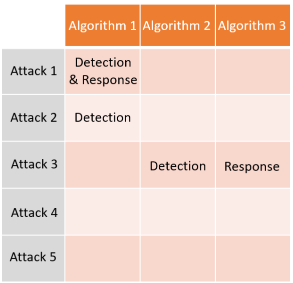


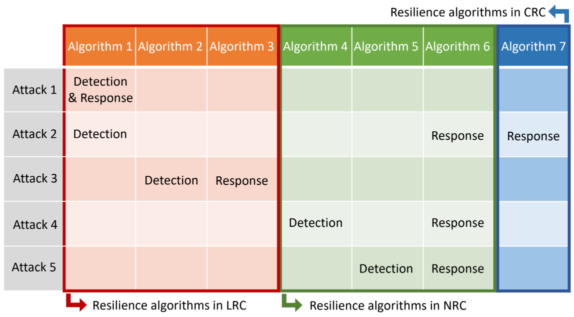
Fig. 3. Local resiliency coordinators (LRCs), network resiliency coordinator (NRC), central resiliency coordinator (CRC) and their connections shown in the attack-resilient CPS architecture.



	Algorithm 1	Algorithm 2	Algorithm 3
Attack 1	Detection & Response		
Attack 2	Detection		
Attack 3		Detection	Response
Attack 4			
Attack 5			

Fig. 4. An instantiation of ADR table for LRC.

Network resiliency coordinator maintains an ADR table of its own, similar to what LRC does. In contrast, the CRC maintains its own ADR together with ADR tables for NRC and LRCs so that coordinated algorithm deployment is possible. See Figure 5 as an example. Here, ADR table for LRC is color coded by light orange, that for NRC is coded by light green and the ADR table for CRC is in light blue for visual distinction. Clearly, it now provides a holistic view that Attack 2 can be tolerated if LRC and NRC work in coordination to deploy Algorithm 1 and Algorithm 5, respectively. Alternatively, Algorithm 1 in LRC and Algorithm 7 in CRC may be coordinated to counteract Attack 2.



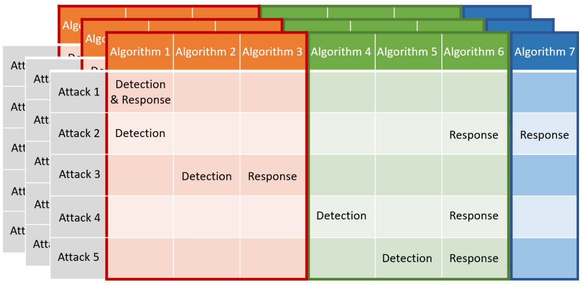
Resilience algorithms in CRC

	Algorithm 1	Algorithm 2	Algorithm 3	Algorithm 4	Algorithm 5	Algorithm 6	Algorithm 7
Attack 1	Detection & Response						
Attack 2	Detection					Response	Response
Attack 3		Detection	Response				
Attack 4				Detection		Response	
Attack 5					Detection	Response	

Resilience algorithms in LRC
Resilience algorithms in NRC

Fig. 5. An instantiation of ADR table for CRC

It should be pointed out that there are as many LRCs as the number of local agents with embedded controllers. Hence, in reality, the ADR table CRC maintains may look like the one in Figure 6. This is simply a graphical illustration for the case with multiple LRCs. The layers of NRC and CRC in the figure are not operating independently, which does not appear clearly in this graphical representation. The actual software implementation may take an appropriate measures to connect all of them in coherent manner.



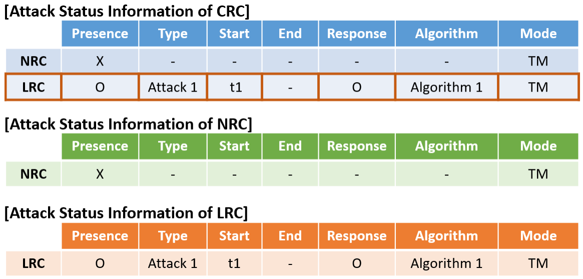
	Algorithm 1	Algorithm 2	Algorithm 3	Algorithm 4	Algorithm 5	Algorithm 6	Algorithm 7
Attack 1	Detection & Response						
Attack 2	Detection					Response	Response
Attack 3		Detection	Response				
Attack 4				Detection		Response	
Attack 5					Detection	Response	

Fig. 6. An instantiation of ADR table for CRC in the case of multiple LRCs.

CRC monitors the situation for attack detection and response in all the layers. LRC and NRC periodically send information on detecting and responding to attacks to the CRC through hidden secure communication networks. When being informed that the LRC or the NRC could not respond to a detected attack, the CRC finds the proper response algorithm in another layer and request the corresponding resiliency coordinator to activate the selected algorithm.

We further describe the behavior of the resiliency coordinator using an example of the ADR table for CRC in Figure 5. As shown in Figure 5, the LRC can detect and respond to Attack 1 and Attack 3. Similarly, NRC can detect and respond to Attack 4 and Attack 5. However, the LRC can detect Attack 2 but cannot respond to it. Therefore, when detecting Attack 2, LRC notifies the CRC that Attack 2 is detected but is not responded to. Then, CRC searches the countermeasures of Attack 2 in its ADR table and CRC requests NRC to activate Algorithm 6, or CRC activates Algorithm 7 by itself.

Figure 7 shows an instantiation how LRC, NRC, and CRC record, respectively, the state at a moment. CRC is updated by NRC that no attack is detected in the network layer, while updated by LRC that Attack 1 is present starting from time t_1, and the attack is responded upon by Algorithm 1.



[Attack Status Information of CRC]

	Presence	Type	Start	End	Response	Algorithm	Mode
NRC	X	-	-	-	-	-	TM
LRC	O	Attack 1	t1	-	O	Algorithm 1	TM

[Attack Status Information of NRC]

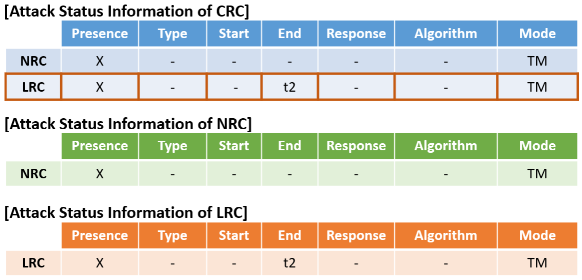
	Presence	Type	Start	End	Response	Algorithm	Mode
NRC	X	-	-	-	-	-	TM

[Attack Status Information of LRC]

	Presence	Type	Start	End	Response	Algorithm	Mode
LRC	O	Attack 1	t1	-	O	Algorithm 1	TM

Fig. 7. An instantiation of attack status information for RC when reacting to Attack 1.

Figure 8 illustrates similar information but at a different time. Here Attack 1 shown in Figure 7 is no longer present and three status table at each layer reflects the change.



[Attack Status Information of CRC]

	Presence	Type	Start	End	Response	Algorithm	Mode
NRC	X	-	-	-	-	-	TM
LRC	X	-	-	t2	-	-	TM

[Attack Status Information of NRC]

	Presence	Type	Start	End	Response	Algorithm	Mode
NRC	X	-	-	-	-	-	TM

[Attack Status Information of LRC]

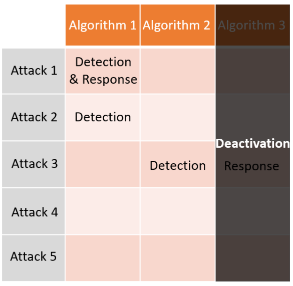
	Presence	Type	Start	End	Response	Algorithm	Mode
LRC	X	-	-	t2	-	-	TM

Fig. 8. An instantiation of attack status information update for RC when Attack 1 disappeared at $t = t_2$.

At the right end of the tables in Figures 7 and 8 show the mode. In these particular cases, the mode is written as TM which stands for tolerate mode. The modes are explained in the subsequent subsection.

3.3 Three modes: tolerate, reduced capacity and graceful degradation

As alluded in Section 3.1, resilient CPS operates in three different modes. They are tolerate mode (TM), reduced capacity mode (RCM), and graceful degradation mode (GDM). When the system operates in TM, attacks may be present. However, some algorithms are active and counteract the effect of the existing attack. The whole functionality is maintained. The system goes into RCM, for instance, the HP module in the Figure of 3 is corrupted and the local embedded system must operate with HA module. In this case, some of the algorithms that are implemented in HP become not available. Such situation is shown in Figure 9, where a column is marked with 'Deactivation'.



	Algorithm 1	Algorithm 2	Algorithm 3
Attack 1	Detection & Response		Deactivation
Attack 2	Detection		
Attack 3		Detection	Response
Attack 4			
Attack 5			

Fig. 9. An instantiation of ADR table for LRC in RCM.

Finally, if majority of the algorithms are not available, and the system functionality is severely affected, CRC puts the system in GDM mode to initiate gradual termination of the operation. NRC and LRCs can change the operation mode of themselves according to the situation, but the change of the operation mode to the GDM can be determined only by CRC.

All resiliency coordinators are in the TM at first. If any countermeasure becomes unavailable after responding to an attack, the resiliency coordinator that has the corresponding countermeasure changes the operation mode to the RCM. If even one of three resiliency coordinators change to RCM, all resiliency coordinators change the operation mode to the RCM. Also, in the RCM, the resiliency coordinator ensures that the target system is in a minimally safe state, then requests the system inspection. In the case of train control systems, for instance, the resiliency coordinator moves a train to the nearest platform and requests to check the train. Meanwhile, in the case where the CRC changes the operation mode to the GDM because it cannot respond to an attack, the system is gracefully degraded to avoid catastrophic system injury and accident.

4. ILLUSTRATIONS

Suppose that either a sensor or an actuator in one of the agents is compromised by the adversary. The nature of the compromise could be a damage exerted on the instrumentation, or malware executed in the local embedded computing device so that internal network communication between sensor/actuator to the feedback controller is corrupted. Another possible attack is jamming on sensors so that without physical destruction or damage, and temporarily hinders the operation of the sensor. Several algorithms that detect sensor attacks have been proposed in [20–22] and actuator attacks in [23, 24] as candidates for the algorithms. In the scenario described, LRC alone is able to decide which algorithm to deploy and counteract.

Another scenario for NRC is link failure in the network. Then, communication between agents and supervisory layer that uses the link is hindered. Upon detection, NRC alone is able to counteract by an appropriate algorithm, e.g., [25].

An example of coordinate countermeasure is as follows. Suppose an agent is unable to operate because its embedded controller is compromised by malicious code. Upon recognizing the situation by the update from LRC, CRC may start to initiate resetting the controller by applying a measures such as hot-patching [26] of the corresponding functionality. In this way, the malicious code is successfully removed for the time being.

Another example may be that adversary connects to a particular access point (AP) in the network and begin corrupting information exchange for those agents that are connected to the network layer through the AP. CRC may provide a coordinated response in the following way. First, it has LRC to change, again by hot-patching or the likes, the communication code so that the agent connects to a different AP. Next, it has NRC to re-route the link so that the compromised AP is removed in the communication route. NRC maintains to monitor the AP to determine if the attack is persisting or discontinued. This information is sent to CRC to update the holistic view of the system capabilities.

5. CONCLUSIONS

This paper has proposed resiliency coordinator that operates on the attack resilient hierarchical control system architecture to fully exploits the individual features to achieve system level resiliency. The resiliency coordinator maintains the holistic status of the system including presence of attacks if detected, either they are responded or tolerated by which countermeasures. It enables coordination of countermeasures, and able to determine which operation mode is adequate for a given situation. In sum, the proposed resiliency coordinator is a key operational software in order to achieve the maximum level of resiliency for the whole system. Multiple examples are given to illustrate the benefit of the resiliency coordinator. Actual implementation of the resiliency coordinator is future work, and in fact under way for communications-based train control systems testbed [27] and scale truck platooning testbed [28], respectively.

ACKNOWLEDGEMENT

This work was partly supported by Institute for Information & communications Technology Promotion (IITP) grant funded by the Korea government (MSIP) (No.2014-0-00065, Resilient Cyber-Physical Systems Research).

REFERENCES

[1] R. Rajkumar, I. Lee, L. Sha, and J. Stankovic, “Cyber-physical systems: The next computing revolution,” in *Proc. 47th Design Automation Conf. (DAC)*, 2010, pp. 731-736.

[2] K.-D. Kim and P. R. Kumar, “Cyber-physical systems: A perspective at the centennial,” *Proc. IEEE,* vol. 100, pp. 1287-1308, May 2012.

[3] K.-J. Park, R. Zheng, and X. Liu, “Cyber-physical systems: Milestones and research challenges,” *Computer Communications*, vol. 36, no. 1, pp. 1-7, 2012.

[4] R. Langner, “Stuxnet: Dissecting a cyberwarfare weapon,” *IEEE Security and Privacy*, 9 (3), pp. 49–51, 2011.

[5] N. Falliere, L. O. Murchu, and E. Chien, “W32.stuxnet dossier,” *White paper*, Symantec Corp., Security Response, vol. 5, no. 6, 2011.

[6] A. Teixeira, A. Saurabh, H. Sandberg, and K. H. Johansson, “Cyber security analysis of state estimators in electric power systems,” *in Proc. of the IEEE Conference on Decision and Control*, pp. 5991–5998, 2010.

[7] A. Wright, “Hacking cars,” *Communications of the ACM*, 54 (11), pp. 18–19, 2011.

[8] T. Sanislav, S. Zeadally, and G. D. Mois, “A cloud-integrated, multi-layered, agent-based cyber-physical system architecture,” *Computer*, vol. 50, no. 4, pp. 27-37, 2017.

[9] H. A. Müller, “The rise of intelligent cyber-physical systems,” *Computer*, vol. 50, no. 12, pp. 7-9, 2017.

[10] H. Sandberg, S. Amin, and K. H. Johansson, “Cyberphysical security in networked control systems: An introduction to the issue,” *IEEE Control Syst. Mag.*, vol. 35, no. 1, pp. 20-23, 2015.

[11] Y. Won, B. Yu, J. Park, I.-H. Park, H. Jeong, J. Baik, et al., “An attack-resilient CPS architecture for hierarchical control: A case study on train control systems,” *Computer*, vol. 51, no. 11, pp. 46-55, Nov. 2018.

[12] M. Farag, M. Azab, and B. Mokhtar, “Cross-layer security framework for smart grid: Physical security layer,” *IEEE PES Innovative Smart Grid Technologies Europe (ISGT Europe)*, pp. 1-7, 2014.

[13] E. Bou-Harb, W. Lucia, N. Forti, S. Weerakkody, N. Ghani, and B. Sinopoli, “Cyber meets control: A novel federated approach for resilient cps leveraging real cyber threat intelligence,” *IEEE Commun. Mag.,* vol. 55, no. 5, pp. 198-204, 2017.

[14] L. Sha, “Using simplicity to control complexity,” *IEEE Software*, 18 (4), pp. 20–28, 2001.

[15] Q. Zhu, C. Rieger, and T. Basar, “A hierarchical security architecture for cyber-physical systems,” in *Proc. 2011 4th Int. Symp. Resilient Control Systems*, Boise, ID, USA, 2011, pp. 1520.

[16] S. R. Chhetri, J. Wan, and M. A. Al Faruque, ”Cross-domain security of cyber-physical systems,” in *Proc. 2017 22nd Asia South Pacific Design Autom. Conf.*, 2017, pp. 200-205.

[17] Q. Zhu and Z. Xu, *Cross-layer design for secure and resilient cyber-physical systems: A decision and game theoretic approach*, Springer Nature, 2020.

[18] M. S. Haque, D. Jun, X. Ng, A. Easwaran, and K. Thangamariappan, “Contract-based hierarchical resilience management for cyber-physical systems,” *Computer*, vol. 51, pp. 56-65, 2018.

[19] F. Januario, A. Cardoso, and P. Gil, “A distributed multi-agent framework for resilience enhancement in cyber-physical systems,” *IEEE Access*, vol. 7, pp. 31342-31357, 2019.

[20] C. Lee, H. Shim, and Y. Eun, “On redundant observability: From security index to attack detection and resilient state estimation,” *IEEE Transactions on Automatic Control*, 64(2), pp. 775-782, 2018.

[21] Y. Jeong and Y. Eun, “A Robust and Resilient State Estimation for Linear Systems”, *IEEE Transactions on Automatic Control*, 67(5) pp. 2626-2632, 2021.

[22] G. Na and Y. Eun, “Active Probing Signal-Based Attack Detection Method for Autonomous Vehicular Systems,” *In Proc. of the 20th International Conference on Control, Automation and Systems*, pp. 53-59, 2020.

[23] G. Na and Y. Eun, “Actuator fault detection for unmanned ground vehicles considering friction coefficients”, *Sensors*, 21 (22), p. 7674, 2021.

[24] G. Na and Y. Eun, “Actuator Fault Detection for Unmanned Ground Vehicles using Unknown Input Observers,” *In Proc. of the 21st International Conference on Control, Automation and Systems*, pp. 97-103, 2021.

[25] H.–S. Park, S. Lee, and K.–J. Park, “Wireless SDN Self-Recovery for Unmanned Swarm Cyber-Physical Systems”, *In Proc. of the 21st International Conference on Control, Automation and Systems*, pp. 87-90, 2021.

[26] H. Jeong, J. Baik, and K. Kang, “Functional level hot-patching platform for executable and linkable format binaries,” *in Proc. IEEE International Conference on Systems, Man, and Cybernetics*, pp. 489–494, 2017.

[27] Y. Won, B. Yu, J. Park, I.-H. Park, H. Jeong, J. Baik, K. Kang, I. Lee, K.-J. Park, Y. Eun, “KRS-DGIST: a resilient CPS testbed for radio-based train control: WiP abstract”, *Proc. of the 9th ACM/IEEE International Conference on Cyber-Physical Systems*, pp. 339-340. 2018.

[28] H. Lee, J. Park, C. Koo, J.-C. Kim, and Y. Eun, “Cyclops: Open Platform for Scale Truck Platooning,” in *Proc. of the IEEE International Conference on Robotics and Automation*, 2022.