Destructive Attacks Detection and Response System for Physical Devices in Cyber-Physical Systems

KABIRI, Peyman <http://orcid.org/0000-0001-5143-0498> and CHAVOSHI, Mahdieh

Published version

KABIRI, Peyman and CHAVOSHI, Mahdieh (2019). Destructive Attacks Detection and Response System for Physical Devices in Cyber-Physical Systems. In: 2019 International Conference on Cyber Security and Protection of Digital Services (Cyber Security). IEEE.

Destructive Attacks Detection and Response System for Physical Devices in Cyber-Physical Systems

*Submission to Cyber Security 2019 (Cyber Security)

Peyman Kabiri
Department of Computing
Sheffield Hallam University
S1 1WB, Sheffield, UK
Email: p.kabiri@shu.ac.uk

Mahdieh Chavoshi
School of Computer Engineering
Iran University of Science and Technology
16846-13114, Tehran, Iran
Email: chavoshi.mahdieh@gmail.com

***Abstract*— Nowadays, physical health of equipment controlled by Cyber-Physical Systems (CPS) is a significant concern. This paper reports a work, in which, a hardware is placed between Programmable Logic Controller (PLC) and the actuator as a solution. The proposed hardware operates in two conditions, i.e. passive and active. Operation of the proposed solution is based on the repetitive operational profile of the actuators. The normal operational profile of the actuator is fed to the protective hardware and is considered as the normal operating condition. In the normal operating condition, the middleware operates in its passive mode and simply monitors electronic signals passing between PLC and Actuator. In case of any malicious operation, the proposed hardware operates in its active mode and both slowly stops the actuator and sends an alert to SCADA server initiating execution of the actuator's emergency profile. Thus, the proposed hardware gains control over the actuator and prevents any physical damage on the operating devices. Two sample experiments are reported in which, results of implementing the proposed solution are reported and assessed. Results show that once the PLC sends incorrect data to actuator, the proposed hardware detects it as an anomaly. Therefore, it does not allow the PLC to send incorrect and unauthorized data pattern to its actuator. Significance of the paper is in introducing a solution to prevent destruction of physical devices apart from source or purpose of the encountered anomaly and apart from CPS functionality or PLC model and operation.**

Keywords*— *Cyber-Physical System Security; IoT Security; Physical Threats in IoT; Physical damage control

I. INTRODUCTION

A Cyber Physical System (CPS) includes two main parts: cybernetic system and physical system. The CPS requirements and its structure makes it challenging to make it secured. Security challenges related to CPS include:

1) Device constraints such as energy, memory and computing resources, and device type.

2) The software deficiencies such as embedded heterogeneous applications, operating systems, diversity of CPSs in their security requirements and sensitivity ratio.

3) The network deficiencies such as node mobility, scalability, real-time operation and dynamic linkage.

Complicated relation between cybernetic system and physical system is also an important challenge. From security point of view, physical systems are often at risk from their cybernetic vulnerabilities. This is due to the growing interaction between physical and cyber systems in CPSs.

Due to the expansion of CPSs and their usage in sensitive areas such as medical systems, water, gas and power grid networks, intelligent transport systems and smart homes the operational safety of CPSs is important.

For example, Stuxnet is one of the most complex threats introduced during the recent years. The final goal of the Stuxnet is to reprogram Industrial Control Systems (ICS) by modifying codes stored on the PLCs, hide the attacker as a legitimate entity within the system and to interact with the system components. In addition, in the Stuxnet case, data mainstream from controller to actuators was changing, without being detected. Moreover, Stuxnet targets hardware components such as PLCs and Distributed Control Systems (DCSs) [1].

This paper presents a protection method against self-destruction apart from special functionality of the CPS. Once under attack, the proposed method prevents any inappropriate changes in controller parameters and sensor values. Therefore, terminating its destructive operation, it maintains system within a secure state of operation. Goal of the proposed method is to prevent CPSs from harming itself. Increasing device lifetime is not an intention for the proposed method.

Intrusion detection systems are divided to misuse (signature-based) detection systems and anomaly (behaviouroral-based) detection systems. In anomaly detection approach, normal system behaviour is defined first, and then all other behaviours considered as abnormal [2].

The proposed solution applies predefined constraints on the data stream to protect the actuator and its subsystems from destruction. The proposed system is a prevention system as well as a response system, it prevents the defected PLC to send data to the actuator. It has an anomaly and specification-based functionality.

In this work, a middleware is introduced between PLC and actuators, i.e. physical devices. First, a pattern is introduced as a working envelope using the controller parameters, authorized values and other sensitives for each device, i.e. normal behaviour pattern. The system administrator saves the normal behaviour pattern on the middleware at the initialization stage. Later, the secured working envelope is used as a basis to analyse safety of the CPS operation, during which, the CPS is monitored continuously. If an unauthorized value is recognized, the solution will immediately send an alert to the associated Supervisory Control and Data Acquisition (SCADA) system.

For example, at normal operating condition, the middleware traces transmission of the signals between PLC and actuators in a passive manner; it analyses signals send to

the PLC in accordance to a predefined secured working envelope. Whenever an unauthorized value is received, the middleware is activated automatically. Once activated, it slowly stops the actuator to prevent damage caused by a sudden stop. It sends alerts to the SCADA system and the SCADA system executes the emergency profile.

Thus, the proposed solution monitors and prevents not only external and internal attacks but also inappropriate functionality derived from unintentional and insider wrong doings that target physical devices directly or indirectly.

II. Related Work

Related works in cyber physical systems security, clarify that, the most of research follow the same common security solutions as in traditional IT networks [3]. Some of them engage in using authentication methods, access control, key management and encryption algorithms [4]. They propose an optimized solution for CPSs considering memory and computational resource constraints. In fact, they focus on requirements, processing time constraint and new diverse attacks on CPSs [5]. A category of them focuses on information flow [6] and routing protocols [7]. They optimize current protocols and propose solutions to improve the operation considering dynamic nodes and scalability in CPSs. Among all the researches in CPSs security field, the most of them present CPSs security challenges and requirements [8, 9]. They propose security models and frameworks referring to CPS attacks and vulnerabilities that are currently recognized [10, 11]. A majority of these works propose a method to CPSs attack detection using double closed-loop security control structure [12] or design an anomaly detection approach based on zone partition for the Industrial CPSs [13]. Hu et al. [14], propose a detection technique against stealthy attacks that can keep themselves undetected by following the expected behaviour of the system closely. Some consider a coding method for the sensor outputs to detect stealthy false data injection attacks [15]. Wang et al. [16], report a First Difference Aware Machine Learning (FDML) classifier to detect Time Synchronization attack (TS attack). Although these security solutions are necessary to design CPS's security framework, a new protective security solution is also required. Constraints, challenges and high accessibility in CPSs are important issues in this area.

III. The Proposed method

The proposed solution tries to provide a secure operating condition for physical devices in CPSs. The secured system stays within the safe operating condition even if an attack occurs and authentication methods fail or IDSs and other security tools are bypassed. Thus, high accessibility, confidentiality and graceful degradation in CPSs will improve.

The idea behind this approach is to protect the physical device against shocks caused by rapid change in momentum (impulse) of the equipment. This is how one can damage shafts, gears or gearboxes. In general, cyber-physical systems are designed to withstand exerted force within their operating condition. Their lifetime is negatively affected if they operate slightly out of their designated boundary. Significantly distancing from the designated operating condition and they can be damaged (broken). The proposed approach intends to protect cyber-physical systems against this time of vulnerability. In another example, asynchrony in operation of different actuators in a conveyer belt will cause malfunction or may even damage the system.

Protecting the system, individual actuator must be monitored and protected. Stopping one actuator may affect other parts of the system. Inevitability other parts of the system must shutdown once there is a malfunction in one part. This is why SCADA has to be informed and a central system has to monitor the middleware to correlate alerts and responses.

The speed of operation in middleware is an important issue. However, in many industrial systems, the required response time is easily achievable using the currently commercially available hardware. This is why this is not of much concern in the reported work.

As it is depicted in Figure 1 and Figure 2, the middleware is located between PLC and actuator. Usually PLC is programmed (or reprogrammed) to control the actuator operations. During the start-up process, actuator's control parameters are initialized in the middleware by the system administrator. In the normal operating condition, the middleware acts as a transparent device. Whenever the PLC sends data towards the actuator, the middleware receives electronic signals from PLC and sends them towards actuator. Here the middleware acts as a repeater and passes PLC generated signals to the actuator.

The middleware analyses signals generated by the PLC. Monitoring the controller parameters, the middleware is activated whenever it recognizes an anomaly. Initially, the middleware stops the actuator slowly and then, it sends an alert to SCADA system. SCADA runs emergency profile for actuators. Because of this operating condition, the PLC is tagged as a defected PLC. Furthermore, the PLC is left out from the working cycle until its error is corrected and the PLC is recovered.

Initially, the administrator defines the emergency profile related to each actuator operating in a safe operating condition. In addition, in the proposed experiment, the middleware treats PLC and actuator as black boxes. Therefore, the architecture and functionality of the proposed approach is not dependent on the functionality or implementation of the various CPSs in use. Generally, it analyses and monitors the sensitive system parameters within the system. Hence, the proposed experiments are designed free from CPSs functionalities and models. Since the Fault Detection and Response System (FDRS) is monitoring and applying corrections to the inputs to the actuators, its job is a repetitive task. Considering industrial production lines carrying-out repetitive jobs, middleware needs minimum configuration and knowledge of its subsystems. As mentioned earlier, system may need a shutdown if a fault is detected in any of its subsystems. This does not need the controller to hold any knowledge about the system.

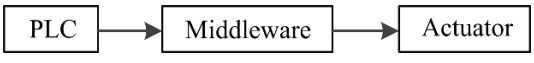

Fig. 1. Proposed solution structure.

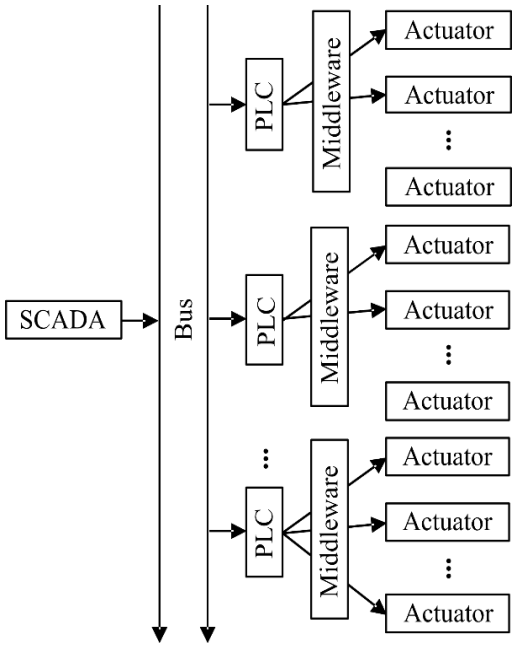

Fig. 2. Proposed solution structure.

Anomaly can occur by either intentional or unintentional reasons such as:

1) Internal or external attacks that either directly or indirectly target physical devices for destruction.

2) Bug or error in programming code within the PLC.

3) Device failure or elapsed lifetime

4) Unpredictable conditions

This paper presents an experiment to examine proposed solution in voltage variant cases.

A. Incompatible Voltage varient experiment I

The goal of this experiment is to evaluate the proposed solution during an incompatible voltage variation. Let the PLC generate sequences of numbers as voltage values with disordered distances (random numbers or predefined numbers). The generated numbers such as motor revolute speeds (per minutes) are sent to the actuator. In addition, there are two controller parameters: Tolerable threshold (T_t) and Critical threshold (C_t). As the generated numbers could be equal to or greater than the T_t only for *n* times within a Sampling Time Window (STW). In fact, *n* is a sensitivity factor for the fault detection that shows how many times passing the threshold can be tolerated before a fault is announced. Minimum number within the sequence is Start (S) and maximum number is End (E). E and S are set to specify minimum and maximum acceptable numbers.

In this experiment, T_t and C_t numbers are configured on the middleware prior to starting the experiment. During the normal operating conditions, the PLC generates numbers within acceptable boundaries and sends them towards the actuator. The middleware, like a repeater, passively monitors the transmitted numbers. The middleware generates numbers meeting one of the following cases once any internal or external attacks on the PLC or any bug and/or error codes in PLC programming is detected:

1) Generated number is equal to or more than the T_t for n samples within a single STW.

2) Generated number is equal to or more than the C_t for n samples within a STW.

In these conditions (Figure 3), the middleware will not send the unacceptable number generated by the PLC towards the actuator. Instead, it sends an alert to the SCADA and SCADA runs the emergency profile related to the actuator. Therefore, the proposed system prevents attacker from destructing and damaging the hardware. Table I presents the first case as an example.

Categorization of different boundaries of values within the sequence of numbers used in the experiment is presented in Figure 4. Generated numbers fall into three categories. Numbers smaller than the T_t are considered as authorized numbers. Numbers within the [T_t C_t) boundary are

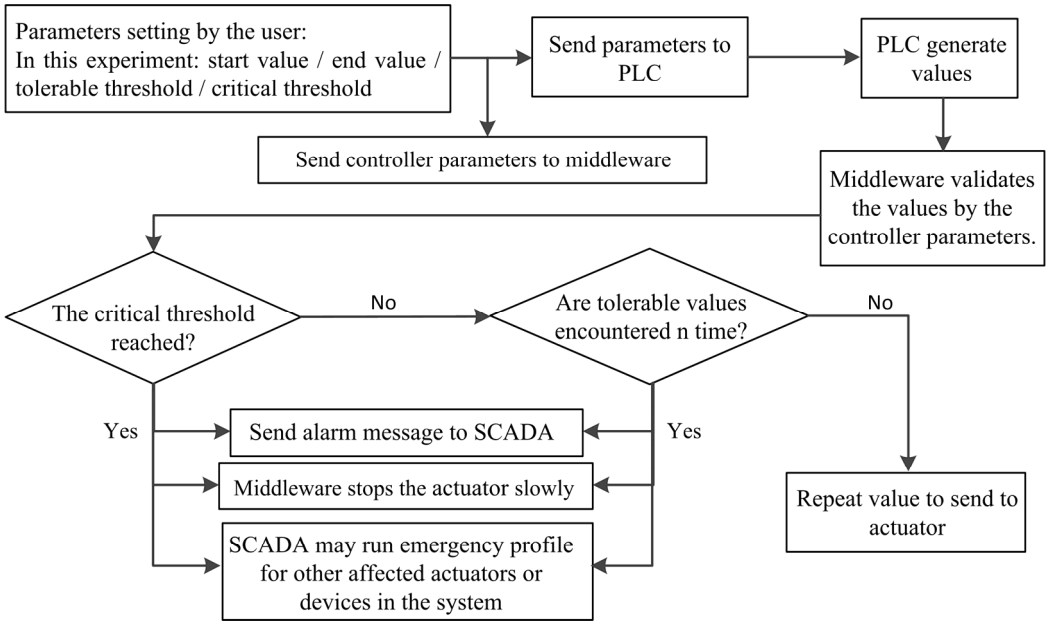

Fig. 3. Program flow chart.

TABLE I. FIRST EXPERIMENT PRIMITIVE DATA

Parameter	Value(s)
S	10
E	65
T_t	30
C_t	38
n	3
STW	20
Generated numbers	11, 15, 19, 12, 32, 28, 30, 16, 13, 25, 10, 31, 17, 30, 34, 22, 18, 36, 40, 23

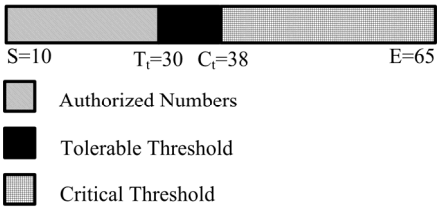

Fig. 4. Categorization of the values within the sequence of possible to generate numbers.

categorized as tolerable threshold. Numbers equal to and greater than the C_t are in critical threshold (must be avoided).

Let, normally 5 numbers are sent from PLC towards actuator in a STW (each STW is about 20ms apart in the experiment). Therefore, as in (1), the first STW includes following five numbers from the set of generated numbers:

$$STW_1 = 11, 15, 19, 12, \mathbf{32} \quad (1)$$

The first STW does not meet any conditions. Because, the collected samples in STW_1 are not equal to or more than the critical threshold. In this experiment, it is assumed that n=3 and it is acceptable to visit three numbers equal to or more than the tolerable threshold in every STW. There is only one number, i.e. 32, in STW_1 that is more than the tolerable threshold. Therefore, the STW_1 samples are acceptable and are transferred from the middleware to actuator.

As presented in (2), the second STW (STW_2) does not meet any fault conditions.

$$STW_2 = 28, \mathbf{30}, 16, 13, 25 \quad (2)$$

Because, the collected samples in STW_2 are not equal to or more than the critical threshold. There is only one number, i.e. 30, in STW_2 that equals to the tolerable threshold.

Whereas, it is assumed that *n*=3 in the experiment. Thus, it is acceptable to visit three numbers equal to or more than the tolerable threshold in every STW. Therefore, the STW_2 samples are acceptable. Hence, these samples are transmitted from the middleware to the actuator.

In the third STW (STW_3), as in (3), the second condition is not met since the collected samples in STW_3 are not equal to or more than the critical threshold.

$$STW_3 = 10, \mathbf{31}, 17, \mathbf{30}, \mathbf{34} \quad (3)$$

However, there are three samples, i.e. 31, 30, 34, in STW_3 that are equal to or more than the tolerable threshold. In the experiment, it is assumed that *n*=3, therefore, any recurrences less than three occurrences will be within tolerable threshold of any STW. Thus, the first condition is met in (3) and the middleware is activated as soon as 34 is generated in the STW_3 by the PLC.

Initially, the middleware stops the motor slowly (or holds its speed) and then it sends an alert to the SCADA. Later, SCADA runs emergency profile for actuators. In this operating condition, the PLC is tagged as a defected PLC. In fact, no numbers are received from the PLC for the analysis and transmission to actuator. Furthermore, the PLC is left out from the working cycle until error correction and PLC recovery is carried-out.

In the fourth STW (STW_4), as in (4), the PLC generates the number 40.

$$STW_4 = 22, 18, \mathbf{36}, \underline{\mathbf{40}}, 23 \quad (4)$$

Although the number 40 is more than the critical threshold, and agrees with the second condition, but no samples in the STW_4 is considered. Due to the condition occurred in STW_3, the PLC is tagged as a defected PLC. In addition, the PLC is left out from the working cycle until PLC operation is corrected.

B. Incompatible Voltage varient experiment II

Examining the second case, the third and the fourth STW samples are replaced with one another. Reported results in Table II are derived from the second experiment.

The collected samples in the first and the second STWs are like STW_1 and STW_2 in the first experiment as in (5) and (6).

$$STW_1 = 11, 15, 19, 12, \mathbf{32} \quad (5)$$

$$STW_2 = 28, \mathbf{30}, 16, 13, 25 \quad (6)$$

In the third STW (STW_3), as in (7), as soon as the PLC generates the value 40, the middleware analyses this number before sending it towards the actuator. Since 40 is more than the critical threshold, the middleware will be activated. Therefore, the middleware starts with stopping the motor slowly and sending an alert to the SCADA. Consequently, SCADA runs the emergency profile for the actuators. In this condition, the PLC is tagged as a defected PLC.

$$STW_3 = 22, 18, \mathbf{36}, \underline{\mathbf{40}}, 23 \quad (7)$$

Furthermore, the PLC is left out from the working cycle until error correction and PLC recovery is performed. As in

TABLE II. SECOND EXPERIMENT PRIMITIVE DATA

Parameter	Value(s)
S	10
E	65
T_t	30
C_t	38
n	3
STW	20
Generated numbers	11, 15, 19, 12, 32, 28, 30, 16, 13, 25, 22, 18, 36, 40, 23, 10, 31, 17, 30, 34

(7), it is necessary to note that 36 is an acceptable number.

Although this number is more than the tolerable threshold, it can be ignored since it is the first occurrence of this condition in the STW.

When the PLC generates number 34 in the fourth STW (STW_4), as in (8). This indicates that the first condition is met, and an alert can be generated.

$$STW_4 = 10, \mathbf{31}, 17, \mathbf{30}, \mathbf{34} \quad (8)$$

The alert generation condition is met since there are three numbers equal to or more than the tolerable threshold in STW_4. However, the PLC is tagged as a defected PLC because of the anomaly occurred in STW_3. Thus, no numbers in the STW_4 can be considered.

As shown in the experiment, the middleware recognizes any incompatible changes in voltage value.

In these experiments, anomalous pattern is used to generate out of range values. For example, following patterns can be considered:

1) *Operations outside of working envelope.*
2) *Irregular and unauthorized number of occurrences of authorized behaviours*
3) *Unacceptable duration of the commanding signals.*

It is imperative to place the middleware on a dedicated network isolated from the PLCs networks or subnets. The proposed functionality for the middleware is close to an Intrusion Detection and Response Systems (IDRS) and its architecture is again very similar, i.e. hierarchical architecture. A dedicated secure network can be responsible for the communication between them. In this way, a group of middleware can act as a team controlled by a central command that can find possible correlations between their activities considering information received from them.

IV. Experimental Setup

The experimental results in the simulated operation conditions successfully followed our expectations.

Schematic for experimental setup is presented in Figure 5. In this paper, Arduino [17] MEGA R3-2560 [18] generates the input numbers, as the controller's output data. In other words, it is considered as a Simulated PLC (SPLC). In addition, Arduino Leonardo [18] plays role of the proposed middleware. Therefore, the middleware monitors the controller's output data. This decision was because Arduino is commonly used as an easily available and affordable price board. Moreover, complexity of the controller's processing is such that Arduino is a suitable choice. For future works, the use of PLC instead of Arduino is considered.

Resistive ladder is used to convert digital values to their analogue equivalents. In the real-world this part is replaced by a Digital-to-Analog Convertor (ADC).

V. Results

The reported experiments show that once the working envelope for the PLCs are defined, the middleware considers them as the acceptance criteria for output data sent from the PLCs to the actuators. Whenever any incompatible data is received, the middleware acts as a barrier and prevents the incompatible data to be transmitted to the actuators. This low-level prevention approach eliminates any concerns and further need for tracking events looking for risks at the physical devices level. In this way, the risk of damaging physical devices can be reduced or even eliminated.

In this experiment, there is a noise problem in converting analogue signals to their digital equivalents. An error rate of 1 or 2 errors per 20 transmissions occurs in the repetitive experimentation. The experiment considers 6 bits as signal outputs. In another set of experiments, choosing larger step size and increase in the signal-to-noise ratio, the error rate

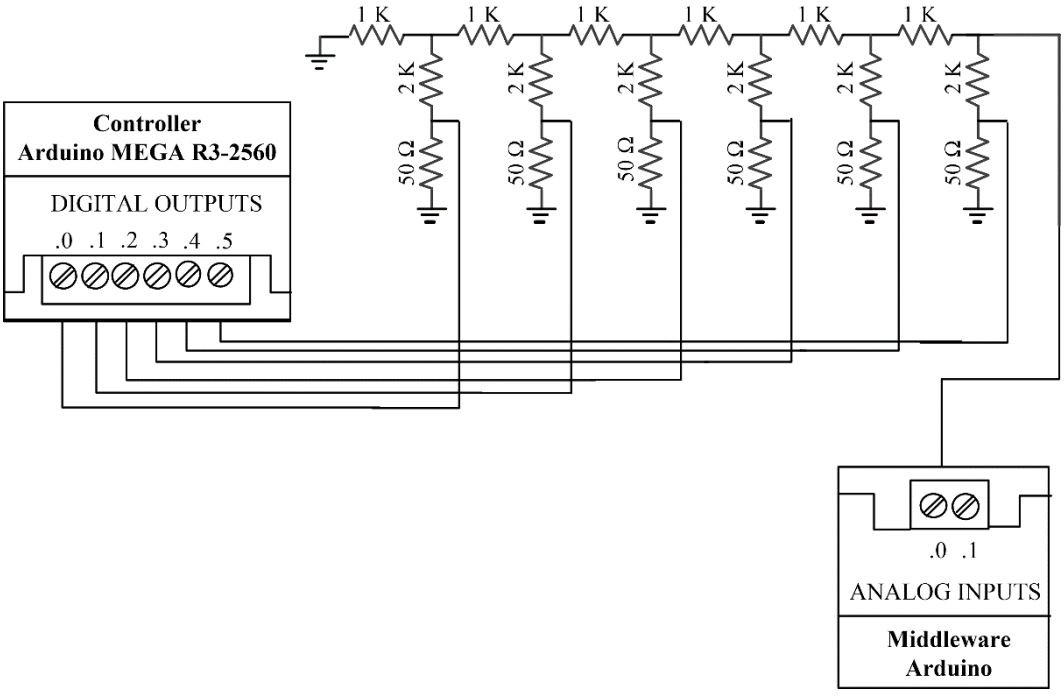

Fig. 5. Schematic for experimental setup.

can be greatly reduced or eliminated.

In the reported experiments, an Arduino Leonardo is used as a middleware. The added middleware may increase the total response time. However, output signals this delay might be negligible for low rate changes in the Arduino.

The main goal of the reported work is to prevent any destructive attacks on physical devices in CPS to guarantee business continuation. The middleware can be designed as a hardcoded device. In this approach, the middleware device can be implemented with PROM or EPROM replacing the RAM. As mentioned earlier, as a great advantage for this approach is its independency from CPS's functionality, PLC Model and their operations.

VI. CONCLUSIONS

In the experiments, it is shown that the proposed solution can prevent destructive attacks on physical devices. The solution is an FDRS and responses to an attack once the attack is detected. In other words, using the proposed approach, the middleware slowly stops the actuator to prevent damage due to a sudden stop and sends an alert to the SCADA system. Consequently, SCADA executes the emergency profile and leaves out the defected PLC from the working cycle and most probably with shutdown the system.

The proposed solution has some drawbacks such as additional cost and delay since it uses a middleware. In comparison to PLC's CPU cycle time, the response time in the proposed solution is short enough to affect actuators functionality. Hence, the system delay (that is due to the middleware) is negligible. Therefore, cost of the proposed solution is mainly the pricewise cost of the added middleware.

VII. FUTURE WORKS

In future work, the proposed solution will be examined in cases where frequency of the digital pulse is changing. In this approach, conditions where speed of the targeted stepper motors change will be monitored.

In the proposed solution, the key point is the data transition rate between middleware and actuators. Especially, during an attack, in which, sending timely response is of great importance. For future work, it is proposed to use boards such as FPGA to design the middleware, which, can reduce middleware delay.

In this paper, designing add-on hardware was proposed. However, designing add-on software could be a proposal for the future works.

In the reported work, the presented experiment (Figure 1 and Figure 2), a dedicated middleware is considered for each PLC. However, as for future work, it is possible to plan a common middleware to monitor the multiple PLCs. The middleware hardware should be fast enough to handle all the PLCs in its subnet within an acceptable time window.

ACKNOWLEDGMENT

The reported work is result of Ms. Chavoshi's work on her M.Sc. project dissertation under supervision of Dr. Peyman Kabiri at Iran University of Science and Technology where she successfully graduated.

REFERENCES

[1] A. Nourian and S. Madnick, "A Systems Theoretic Approach to the Security Threats in Cyber Physical Systems Applied to Stuxnet," *IEEE Transactions on Dependable and Secure Computing,* vol. 15, no. 1, pp. 2-13, 2018.

[2] S. Kourki Nejat and P. Kabiri, "An Adaptive and Cost-Based Intrusion Response System," *Cybernetics and Systems,* vol. 48, no. 6-7, pp. 495-509, October 2017.

[3] L. Vegh, "Cyber-physical systems security through multi-factor authentication and data analytics," in *2018 IEEE International Conference on Industrial Technology (ICIT)*, 2018, pp. 1369-1374.

[4] O. Kocabas, T. Soyata, and M. K. Aktas, "Emerging Security Mechanisms for Medical Cyber Physical Systems," *IEEE/ACM Transactions on Computational Biology and Bioinformatics,* vol. 13, no. 3, pp. 401-416, 2016.

[5] L. A. Maglaras and J. Jiang, "Intrusion detection in SCADA systems using machine learning techniques," in *2014 Science and Information Conference*, 2014, pp. 626-631.

[6] J. Liu *et al.*, "Secure Autonomous Cyber-Physical Systems Through Verifiable Information Flow Control," presented at the Proceedings of the 2018 Workshop on Cyber-Physical Systems Security and PrivaCy, Toronto, Canada, 2018.

[7] L. Wallgren, S. Raza, and T. Voigt, "Routing Attacks and Countermeasures in the RPL-Based Internet of Things," *International Journal of Distributed Sensor Networks,* vol. 9, no. 8, p. 794326, August 2013.

[8] A. Riahi, E. Natalizio, Y. Challal, N. Mitton, and A. Iera, "A systemic and cognitive approach for IoT security," in *2014 International Conference on Computing, Networking and Communications (ICNC)*, 2014, pp. 183-188.

[9] I. Ahmad, M. K. Zarrar, T. Saeed, and S. Rehman, "Security Aspects of Cyber Physical Systems," in *2018 1st International Conference on Computer Applications & Information Security (ICCAIS)*, 2018, pp. 1-6.

[10] F. Pasqualetti and Q. Zhu, "Design and Operation of Secure Cyber-Physical Systems," *IEEE Embedded Systems Letters,* vol. 7, no. 1, pp. 3-6, 2015.

[11] A. Humayed, J. Lin, F. Li, and B. Luo, "Cyber-Physical Systems Security-A Survey," *IEEE Internet of Things Journal,* vol. 4, no. 6, pp. 1802-1831, 2017.

[12] H. Ge, D. Yue, X. Xie, S. Deng, Y. Yang, and Y. Ji-quan, "Double closed-loop NCSs modeling for security control and a defense framework design," in *2017 36th Chinese Control Conference (CCC)*, 2017, pp. 7777-7782.

[13] J. Yang, C. Zhou, S. Yang, H. Xu, and B. Hu, "Anomaly Detection Based on Zone Partition for Security Protection of Industrial Cyber-Physical Systems," *IEEE Transactions on Industrial Electronics,* vol. 65, no. 5, pp. 4257-4267, 2018.

[14] Y. Hu, H. Li, H. Yang, Y. Sun, L. Sun, and Z. Wang, "Detecting stealthy attacks against industrial control systems based on residual skewness analysis," *EURASIP Journal on Wireless Communications and Networking,* vol. 2019, no. 1, p. 74, March 19 2019.

[15] F. Miao, Q. Zhu, M. Pajic, and G. J. Pappas, "Coding Schemes for Securing Cyber-Physical Systems Against Stealthy Data Injection Attacks," *IEEE Transactions on Control of Network Systems,* vol. 4, no. 1, pp. 106-117, 2017.

[16] J. Wang, W. Tu, L. C. K. Hui, S. M. Yiu, and E. K. Wang, "Detecting Time Synchronization Attacks in Cyber-Physical Systems with Machine Learning Techniques," in *2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS)*, 2017, pp. 2246-2251.

[17] (As visited on June 2018). *Arduino - Home*. Available: www.arduino.cc

[18] (As visited on June 2018). *Products - Home*. Available: https://www.arduino.cc/en/Main/Products