Introduction to Security in Cyber-Physical Systems

Chris Dean
Department of Computer Science
University of North Carolina at Chapel Hill
Chapel Hill, USA
cmdean@cs.unc.edu

Martin Meng
Department of Computer Science
University of North Carolina at Chapel Hill
Chapel Hill, USA
martinmq@live.unc.edu

***Abstract*—The rise of new technologies always brings on security problems. As computers have been more and more integrated into the physical system, particularly control systems. This paper is intended to provide an introduction to some security problems that have been seen and some solutions that have been proposed for those problems. In particular we will focus on security vulnerabilities that are specific to control system, which is to say not general cyber security issues such as building secure networks.**

***Index Terms*—Cyber Security, Control Systems, Cyber-Physical Systems, Embedded Cyber-Physical Systems**

I. Introduction

Cyber-physical systems (CPS) represent systems that are the intersection of computational elements and physical processes connected by distributed networks [1, 2]. CPS have a wide-spread range of applications such as smart cities, energy delivery, water distribution, and Internet-of-Things devices [1, 3]. Therefore, it is of great importance to maintain the security of CPS. However, the increased scales and functionalities of CPS make it progressively more difficult to keep the systems secure. New methods and strategies are needed to main the security features of CPS given the increased complexity of the systems.

CPS consist of two part: embedded systems and control systems. Embedded systems combine software and hardware in order to perform tasks for or within a larger system. Control systems consist of a physical plant and a controller which ensure certain behavior of the physical plant. We assume that the control systems discussed in this paper are feedback control systems. The output of the system is controlled using the states of the system as feedback signals that determine the inputs to the system. In this paper, we emphasize on the security of feedback control systems.

The security of control systems is measured by the CIA triad in traditional computer security: Confidentiality, Integrity, and Availability [4]. Confidentiality considers keeping the secret information secret, ensuring that the data of the system remains unknown to unauthorized users. Integrity corresponds to ensuring the trustworthiness of the data, making sure the data is not modified by malicious party. Availability concerns the accessibility of the data or functionality of a system.

An attack of a control system can be classified by the resources needed to corrupt the system with respect to the CIA triad. The required resources of an attack can also be described by a triad consisting of CPS model knowledge, disclosure resources, and disruption resources [3].

CPS model knowledge represents the amount of the knowledge obtained by the attacker about the models of the plant, controller, and anomaly detector. Obtaining CPS model knowledge violates the principle of confidentiality; the model knowledge should be kept secret but it is known by the attacker.

Disclosure resource corresponds to the amount of resources needed by the attacker to read and store data in the communication channels of the system. For example, the number of channels used to read and store data in Figure 1 is the disclosure resources in a typical attack. The utilization of disclosure resources violates the confidentiality principle in CIA triad as well; the attacker uses this resource to obtain the secret information transmitted in the system.

Disruption resource relates to the resource required by the attacker to write or drop data in the communication channel. Disruption resource corresponds to the number of channels used to write or drop data as shown in Figure 1. The ability to write data to the system violates the principle of integrity; the feature of dropping system data violates the availability principle. An example of an attack utilizing disruption resource is the classical denial-of-service-attack [5, 6]. Such attack uses the method of sending overwhelmed request to certain functionality of the system and thus blocks the requests of such functionality sent by the system itself. In this attack, the transmitted data of system, i.e. the request of the system to such functionality, gets dropped.

After classifying the attacks to control systems by the required resources, we further specify an attack by its attack policy [3]. An attack policy specifies the modified version of system output and control input, i.e. the values of $\bar{y}$ and $\bar{u}$ in Figure 1, as functions of previous and current system outputs and control inputs. Formally, an attack policy is defined as follows:

$$\bar{y}[k] = \phi_y(Y_{[0,k]}, U_{[0,k]})$$
$$\bar{u}[k] = \phi_u(Y_{[0,k]}, U_{[0,k]})$$

where

$$X_{[0,k]} \equiv \{x[0], x[1], ..., x[k]\}$$

denotes the collection of data x[t] from time 0 to time k for $X = \{Y, U\}$. ϕ_x is an attacker-defined function of inputs $Y_{[0,k]}$ as well as $U_{[0,k]}$ and output $\bar{x}[k]$ for $x = \{y, u\}$.

Outline: In this section, we have identified control systems as the subject of our discussion; we have provided measurements from the literature of computer security, as well as a description framework of CPS attacks. In the next section, we formulate several common models of control systems to serve as the subjects of our following discussion. In section III, we discuss several types of CPS attack and fit them into our attack description framework. We then introduce two state-of-the-art detection methods in section IV. We describe and make an effort to address several open problems in section V. We conclude in section VI.

II. Background

A. Stochastic LTI Systems

For some of the attacks the systems are modeled as Stochastic LTI Systems. An LTI system is a linear time invariant system, which means it is modeled as a linear set of equations. However, it is not expected that a system will reach an exact state value given a particular input, but will problematically reach a state within a given distribution. This means analysis of the system must take into account a wide range of states into which a system might end up. In this case we would prescribe a probability that system constraints get violated rather than a binary yes or no. In particular these systems are also modeled as discrete time systems

These systems follow a slightly modified set of equations for modelling the system. They are as follows:

$$\begin{cases} x(k+1) = Ax(k) + Bu(k) + w(k) \\ y(k) = Cx(x) + Du(k) \end{cases}$$

In this case w(k) is a stochastic random variable following the appropriate distribution. A, b, C, and D are all the same from a non stochastic system.

B. Networked Cyber-Physical Systems

A useful model of a networked CPS can be seen in Fig. 1. In this case we will have a plant that outputs a reading to a Read and Store node which then is forwarded through out attack policy to a Write/Drop node which makes a choice to either pass on the message or not. This message passes through a traditional feed forward/feedback control system and also an anomaly detector. for an output u the output is passed back through the network which gives through the attack policy and on to the plant.

Attacks on networked systems are some of the most prevalent attacks present today. Traditionally there would be many things that an attacker can do to compromise a networked system, such as dropping all the packets or modifying them to make them invalid. However, given that this paper will focus on attacks on cyber-physical systems there has to be distinguishing traits that specify these attacks for a CPS rather than a general networked system.

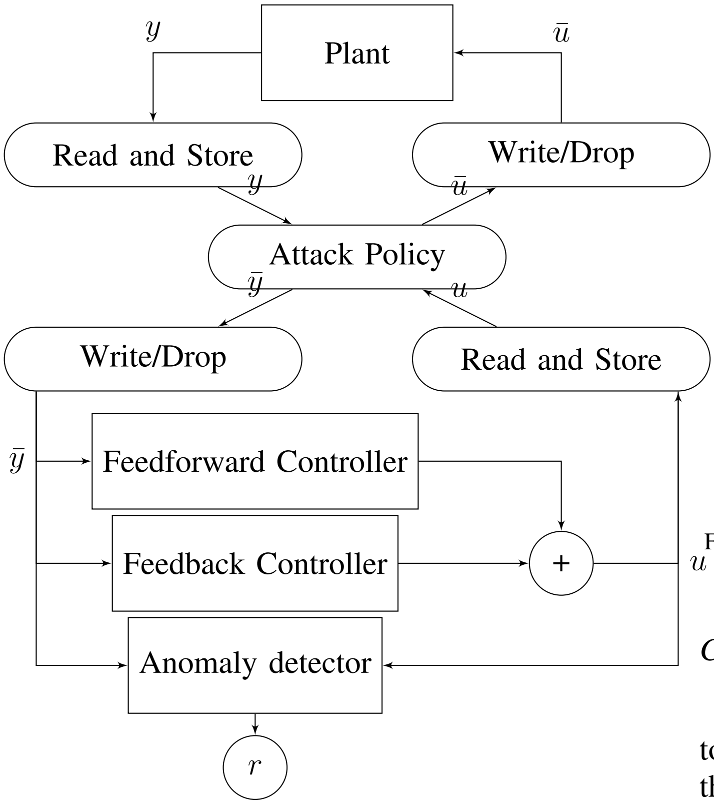

Fig. 1. Model for a networked Cyber-Physical System

Concretely an attack policy will not be able to simply drop all the packets (except in the case of a denial of service attack) and the anomaly detector will be able to detect anomalies with some amount of accuracy. It is definitely worth noting that while the anomaly detector needs to be able to detect anomalies soon after they happen, the anomaly detector is **not** subjected to hard or soft real-time constraints, and for some intents and purposes could be considered an offline calculation. The specifics of the anomaly detector's abilities are more specified depending on the attack model. For these type of attacks we are generally interested in what damage can be done by injecting fake messages into the network, dropping some of the legitimate messages, modifying the data of legitimate messages into fake messages.

With this threat model we can intercept messages going to and from the plant, which means we can modify not only the sensor readings but also install our own actuation into the plant to get desired behaviors.

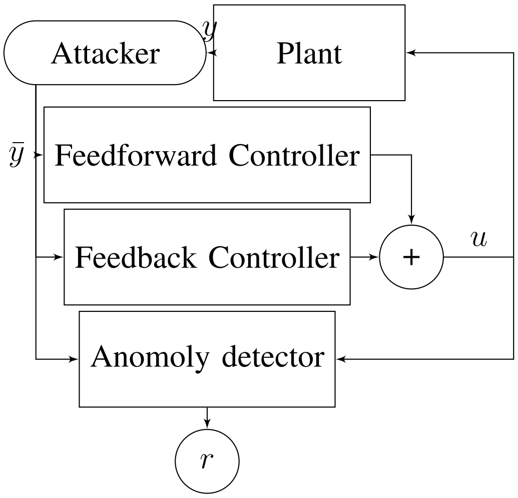

Fig. 2. Model for Sensor compromise in a Cyber-Physical System

C. Sensor Tampering

Another threat model for an attacker it the ability to tamper with the sensor output of the plant. In this model, the attacker only has control over the readings that are sent back to the controller, not general network access. It can be modeled as in figure 2. In this case we abstract away the message sending and receiving over the network, as in the common case the actually messaging system is not particularly relevant to the attack, just the actual values of the sensor data being falsely reported.

III. Attacks Against Cyber-Physical Systems

A. Eavesdropping Attacks

Eavesdropping attacks are attacks on networked CPS designed to violate confidentiality. Say a CPS is handling sensitive information such as user data. In an eavesdropping attack a listener on the network would monitor the network traffic and try to pull confidential information from the system. While this is not an attack that necessarily would be specific to cyber-physical systems, there may be ways to mitigate attacks that are specific to CPS. Additionally, given the timing critical nature of control systems, traditional defenses such as cryptography can often incur too much of an overhead to be feasible in a cyber-physical system.

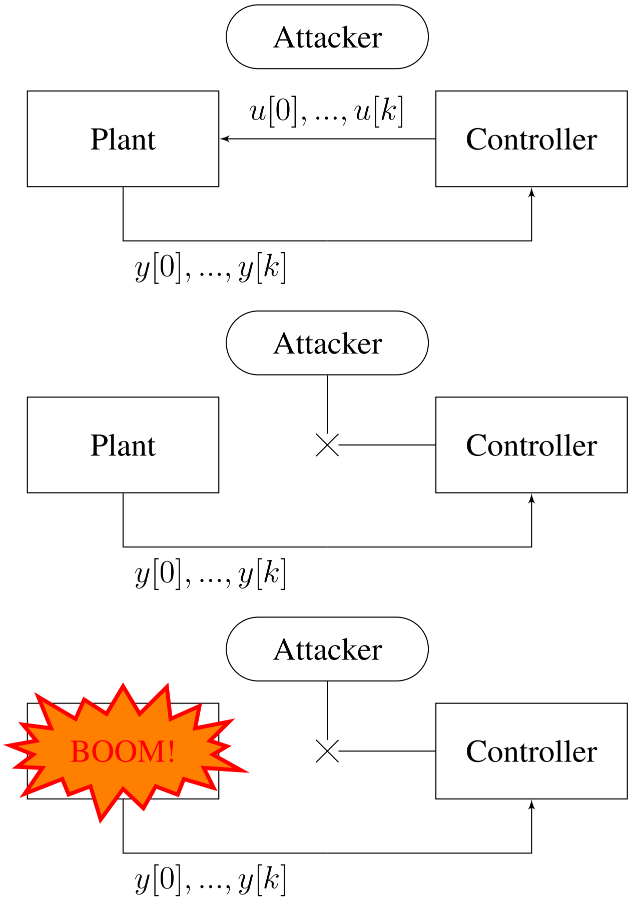

Fig. 3. At the top of this figure you see a normally function CPS. In the middle the attacker blocks control signals to the plant which causes the malfunction in the last part of the figure.

B. *Denial of Service Attacks*

A denial of service (DOS) attack on a system is an attack designed to disrupt the availability of a system. In this case a denial of service attack would be any attack designed to prevent the system from functioning normally. Similar to eavesdropping attacks this type of attack is definitely not specific to CPS. In this case examples of DOS attacks would be dropping all packets on the network, which would effectively prevent the plant from being controlled, or scrambling the contents of packets so control signals have random values, which could also prevent the plant from being controlled. Figure 3 shows an attacker blocking traffic to the plant causing malfunction. This would be a DOS attack.

C. *Replay Attacks*

[2] did an analysis of replay attacks in cyber-Physical systems. In this work a replay attack is described as such:

> ”In order to inject an exogenous control input without being detected, the attacker hijacks the sensors, observes and records their readings for a certain amount of time, and repeats them afterwards while carrying out his attack.”

Conceptually this makes a replay attack very simple. By replaying legitimate readings an anomaly detector would have trouble distinguishing between legitimate and illegitimate readings from the system sensors. While the attack is ongoing, the plant is not being controlled by the controller and thus might drift outside its constraints due to the uncontrolled nature of the attacks. An example of this can bee seen in figure 4. Note that in figure 4 the lines simply represent network connections.

This attack can be achieved in either the networked based model or the sensor tampering model. In the sensor based model the attacker simply simulates previously read sensor outputs when asked for sensor values. In the network based model messages are modified to contain the contents of previously read messages where appropriate. This is one of the most common attacks on CPS as it is hard to detect and correct for without violating the constraints of the system. While technically replay attacks are forms of injection attacks, they are a very specific and well studied form of the attack, as well as being very commonly seen in actual attacks (although they may be used in conjunction with other forms of injection attacks).

D. *Injection Attacks*

Injection attacks are a more general class of attack than replay attack, which involve injecting messages into the system to disrupt the plant’s behavior. With replay attacks we were specifically worried about an attacker injecting false sensor readings into the system, in an injection attack we worry about all messages that could be introduced into the system. This includes messages sent by an attacker to the pretending to be the controller to the plant.

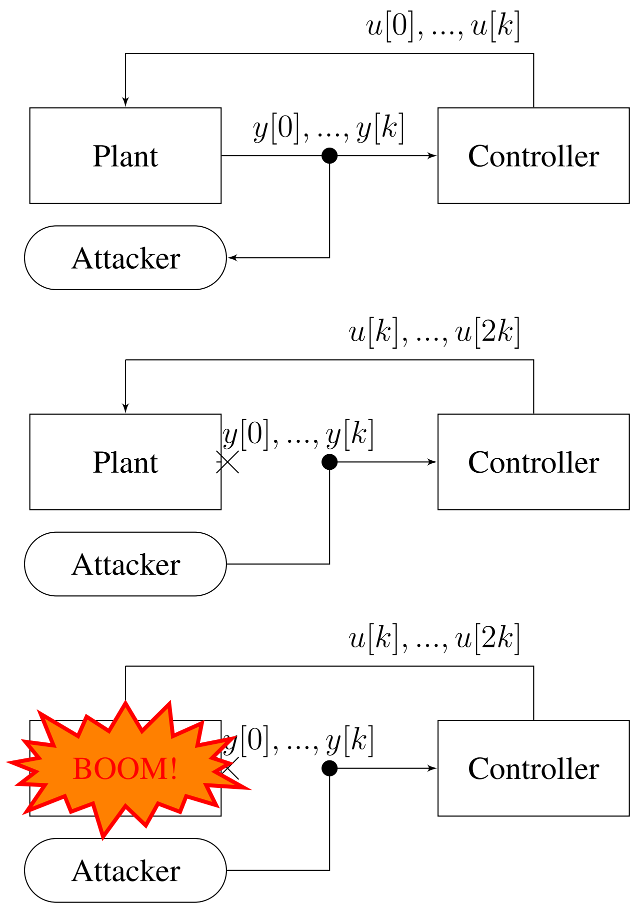

Fig. 4. At the top of the diagram you can see the attacker simply recording sensor outputs. In the middle diagram the attacker is replaying previously read states to the controller which is causing the explosion in the third diagram. Please note that the explosion is simply a stand in for the plant violating its constraints and is in no way guaranteed by a replay attack.

This type of attack is obviously a much stronger security threat than a simple replay attack, so techniques used in analysis are more complex and must take more into account. In an injection attack an attacker may have control over a system much closer to what is seen in figure 1. That is to say any or all messages on the system may be affected by whatever attack policy is in place. However, unlike a specific replay attack, an injection attack might not be focused on disrupting the system without being noticed, which is to say they may be just targeting the system to disrupt or destroy it and the damage done might not, from the attacker's perspective, need to be done stealthily.

IV. Detection of Attacks

A. Authentication Signals

According to [2] how well constrained systems function in the presence of replay attacks is not a well studied, however causing control systems to run in a dangerous way, which is to say only violating constraints not necessarily causing system instability, can cause lots of damage to most physical systems.

The solution [2] purposes is known as an authentication signal. In a probabilistically modeled system there would be an expected distribution in the variation of readings from the system to the expected state of the system. An authentication signal would a shift in the actuation signal such that the distribution of states of the outputs of the system would shift in a known way.

Over time an anomaly detector would gather data about the distribution of system states and use a statistical test known as a χ^2 test to construct its anomaly detector, known as a χ^2 failure detector. The essence of this detector is that it can accumulate the variations from the norm of the system and in a statistically meaningful way say whether or not these differences follow a given distribution or not.

The authentication signal is a way we can use these tests to detect replay attacks. When an authentication signal is applied the way the system should react to the signal can be accounted for by the anomaly detector and then the χ^2 test can be applied to test if the system outputs are appropriately adjusted to the authentication signal. The following is the state space model for the system:

$$\begin{cases} x(k+1) = Ax(k) + Bu(k) + w(k) \\ y(k) = Cx(x) + Du(k) + \alpha(u) \end{cases}$$

This is a variation on the LTI system model with $\alpha(k)$ being the authentication signal at time k.

One thing to notice is that the authentication signal must change over time. In the case that it doesn't then a replay attack would give the expected output and the anomaly detector would not be able to distinguish it from the true system output

One last consideration must be made when using authentication signals. Depending on the dynamics of the system a small perturbation to the system might cause the system to violate constraints or

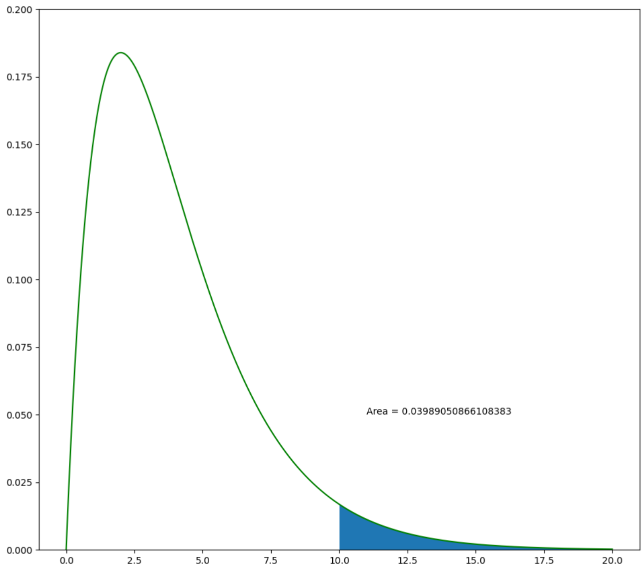

Fig. 5. An example χ^2 distribution. If the test had a value of 10, then that would correspond to a probability of about 4%, which may be anomalous on many systems.

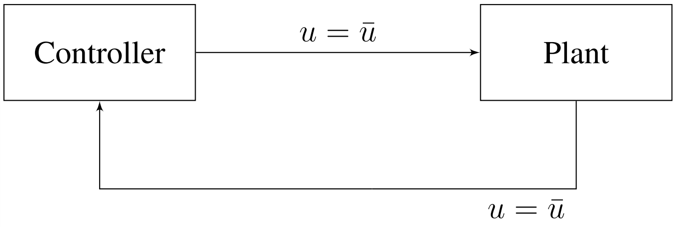

Fig. 6. A traditional control system with no encoding.

become unstable. For this reason full analysis of the system with the authentication inputs must be done in order to ensure with high probability that constraints are satisfied. Also given some systems, sensors may not be high quality enough to pick up small perturbations in the environment's state, so authentication signals simply do not have enough of an effect on the system to be used to effectively monitor for replay or other injection style attacks.

B. Two Way Coding

Two way coding is a system of mathematical functions that when applied to messages in a system would cause any attacker to have a distorted view of the system and would help to detect any sort of injection or message tampering.

A two way coding is defined as [7]

$$\begin{bmatrix} q(t) \\ y(t) \end{bmatrix} = M \begin{bmatrix} u(t) \\ v(t) \end{bmatrix}$$

where

$$M = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$$

Additionally we need

$$ad \neq 0, ad - bc \neq 0$$

Which guarantees that M has an inverse M^{-1}. Also let

$$\begin{bmatrix} \bar{a} & \bar{b} \\ \bar{c} & \bar{d} \end{bmatrix} = M^{-1} = \frac{1}{ad - bc} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$$

In two way coding a signal from each direction of the path is applies to the message to encode and decode it. This is different from one way coding in which is signal is only applied in the direction the message is travelling.

Special matrices are calculated to give the certain properties. These properties should provide a few things. First, we want to make the system resilient against attacks so it would be nice if the coding obscured system messages. Secondly it would be nice to be able to detect when messages were tampered with, and final it would be useful to have a way of correcting tampered with messages.

With certain matrices we can give these guarantees for some attacks. While some lucky injections may have certain mathematical properties, for most cases injections will be able to be caught and corrected if using two way coding. In all cases however two way coding is able to provide some level of confidentiality and only allow an attacker to gain a skewed view of the system.

V. Open Problems

A. Traffic Control

A lot of the biggest open problems in cyber-physical system security is building secure and safe traffic control systems [8]. There are a few open problems in this area. First you have the open problem of building these systems such that they

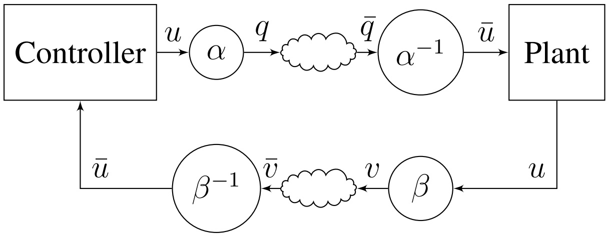

Fig. 7. A control system with one way encoding. The clouds represent traffic going over the network.

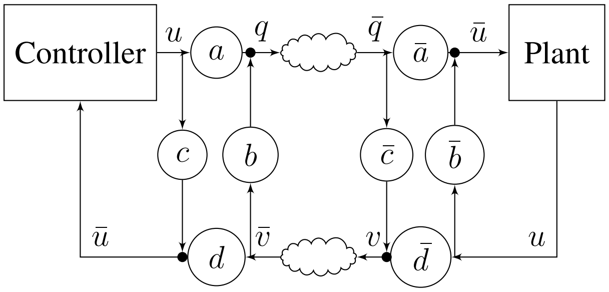

Fig. 8. A control system with two way encoding.

are extremely resistant to tampering and very robust against sensor failures or tampering. However, some other the larger problems come with building networked smart cities that can optimize traffic flow on the fly. Given the scale of these networks, there is obviously the problem of building these networks to be fault tolerant and also tamper resistant, as any tampering could cause major traffic accidents.

This seems to be one of the biggest open problems in CPS security, and also seems to garner much public attention, as almost everyone believes that traffic should be better. However, not all the problems in this area are necessarily control problems, there is a good overlap in CPS security and some of the problems in networked traffic control.

B. Smart Grid

Smart grid research is also a very active are of research for cyber-physical systems [9]. In these cases there are open problems in controlling the flow and storage of energy across a large area. A lot of the problems in this are more Similar to traffic control a lot of the security problems come down to designing such a large fault tolerant network. In this case there are also a lot of problems that come with accurate metering.

Although this is not necessarily as well known as traffic control, this is a very important research are for future energy consumption. In short, it is possible that with a large smart grid, significant amounts of energy could be conserved just through more precise control techniques control techniques. However, this is also a hard solution to implement because of the high cost of replacing current power grib infrastructure.

C. Hardware Security vs CPS security

In this section, we discuss the similarities and differences of hardware security and CPS security. To the best of our knowledge, such comparison has not been made in previous literature. Our comparison is not complete and we would like to call for further research on the intersection of these two fields. We assume in this section that the CPS model is discrete liner time invariant. We further assume that computation delay and communication delay are negligible. A typical model of such discrete system is as follows:

$$\begin{aligned} x[k+1] &= \phi x[k] + \Gamma u[k] \\ y[k] &= Cx[k] \\ u[k] &= Kx[k] + Fr \end{aligned}$$

where x[t], y[t], and u[t] denote the state, the output, and the control input of the system at time t, respectively. K is the feedback gain, F is the feedforward gain, and r is the reference of the system, which is a constant. ϕ is a matrix $\in \mathrm{R}^n \times \mathrm{R}^n$, Γ is a matrix $\in R^n \times 1$, and C is a matrix $\in 1 \times \mathrm{R}^n$.

Hardware consists of two basic building blocks: Sequential Logic and Combinational Logic. The definitions of them and the similarities between them and CPS model are given below.

On one hand, Sequential Logic is time-dependent: the state at current cycle is a function of present input and states in the previous cycles. This is similar to defining the state in the CPS model. The state at time k+1 is defined as a function of the state at time k and the control input at time k. Therefore, it can be formulated as a special case of Sequential Logic.

On the other hand, Combinational Logic is time-independent: it is only a function of inputs or states at the current cycle; it does not need to know information about previous states and previous inputs. This is similar to how the output and the control input are produced in the control system discussed previously. The current output is only dependent on current states, and the control input is also a function of current states. Thus, we can formulate the output and control input as Combinational Logic in hardware.

After identifying the similarities, we focus on the difference between CPS security and hardware security that may block us from reducing the CPS security problem to a hardware security problem. The main difference is that there is a program executed by the hardware processor while a control system does not execute a specific code; instead, a formula or function is used to calculate the control input in each cycle.

To potentially overcome this obstacle, we propose to design the control system as a hardware design and run a dummy program consisting of an infinite loop only on the design. If such formulation is correct, we can apply standard verification techniques in the literature of hardware security; for example, a recursive strategy to apply symbolic execution in hardware literature [10] can be used to verify a control system.

We have made a simple comparison between hardware security and CPS security. The similarities and differences are addressed, and a potential method of reducing CPS security problem to the verification problem in hardware security is proposed. However, further thinking has to be made and further research on this topic is strongly needed and encouraged.

VI. Conclusion

In this paper, we have provided an introduction to CPS security. In section I, we have introduced a framework to describe and measure the attacks to CPS. In section II, two common models of control systems are formulated as the subject matter for the following discussion. In section III, four typical types of CPS attacks are formulated and the detection methods are introduced in section IV. In section V, we discuss three open problems for future research directions.

References

[1] R. Romagnoli, S. Weerakkody and B. Sinopoli, ”A Model Inversion Based Watermark for Replay Attack Detection with Output Tracking,” 2019 American Control Conference (ACC), Philadelphia, PA, USA, 2019, pp. 384-390.

[2] Mehdi Hosseinzadeh, Bruno Sinopoli, Emanuele Garone: Feasibility and Detection of Replay Attack in Networked Constrained Cyber-Physical Systems. Allerton 2019: 712-717

[3] M. S. Chong. H. Sandberg, and A. M. Teixeira, “A tutorial introduction to security privacy for cyber-physical systems,” in Proc. 18th ECC, 2019, pp. 968-978.

[4] M. Bishop, Computer Security: Art and Science. Professional, 2002.

[5] S. Amin, A. A. Cárdenas, and S. S. Sastry, “Safe and secure networked control systems under denial-of-service attacks,” in In- ternational Workshop on Hybrid Systems: Computation and Control. Springer, 2009, pp. 31–45.

[6] C. De Persis and P. Tesi, “Input-to-state stabilizing control under denial-of-service,” IEEE Transactions on Automatic Control, vol. 60, no. 11, pp. 2930–2944, 2015.

[7] S. Fang, K. H. Johansson, M. Skoglund, H. Sandberg, and H. Ishii, “Two-way coding in control systems under injection attacks: From at- tack detection to attack correction,” arXiv preprint arXiv:1901.05420, 2019.

[8] R. Eini and S. Abdelwahed, ”Distributed Model Predictive Control for Intelligent Traffic System,” 2019 International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), Atlanta, GA, USA, 2019, pp. 909-915.

[9] P. Kumar, Y. Lin, G. Bai, A. Paverd, J. S. Dong and A. Martin, ”Smart Grid Metering Networks: A Survey on Security, Privacy and Open Research Issues,” in IEEE Communications Surveys Tutorials, vol. 21, no. 3, pp. 2886-2927, thirdquarter 2019.

[10] R. Zhang, C. Deutschbein, P. Huang and C. Sturton, ”End-to-End Automated Exploit Generation for Validating the Security of Processor Designs,” 2018 51st Annual IEEE/ACM International Symposium on Microarchitecture (MICRO), Fukuoka, 2018, pp. 815-827.