On the use of watermark-based schemes to detect cyber-physical attacks

Jose Rubio-Hernan, Luca de Cicco, Joaquin Garcia-Alfaro

To cite this version:

Jose Rubio-Hernan, Luca de Cicco, Joaquin Garcia-Alfaro. On the use of watermark-based schemes to detect cyber-physical attacks. EURASIP Journal on Information Security, 2017, 2017, pp.8-1 - 8-25. 10.1186/s13635-017-0060-9 . hal-01701167

HAL Id: hal-01701167
https://hal.science/hal-01701167

Submitted on 5 Feb 2018

RESEARCH

On the use of Watermark-based Schemes to Detect Cyber-Physical Attacks

Jose Rubio-Hernan[1], Luca De Cicco[2] and Joaquin Garcia-Alfaro[1*]

*Correspondence: joaquin.garcia_alfaro@telecom-sudparis.com
[1] SAMOVAR, Telecom SudParis, CNRS, Université Paris-Saclay, 9 Rue Charles Fourier, 91000 Evry, France
Full list of author information is available at the end of the article

Abstract

We address security issues in cyber-physical systems (CPSs). We focus on the detection of attacks against cyber-physical systems. Attacks against these systems shall be handled both in terms of safety and security. Networked control technologies imposed by industrial standards already cover the safety dimension. However, from a security standpoint, using only cyber information to analyze the security of a cyber-physical system is not enough, since the physical malicious actions that can threaten the correct behavior of the systems, are ignored. For this reason, the systems have to be protected from threats to their cyber and physical layers. Some authors have handled replay and integrity attacks using, for example, physical attestation to validate the cyber process and detect the attacks, or watermark-based detectors which uses also physical parameters to ensure the cyber layers.

We reexamine the effectiveness of a stationary watermark-based detector. We show that this approach only detects adversaries that do not attempt to get any knowledge about the system dynamics. We analyze the detection ratio of the original design under the presence of new adversaries that are able to infer the system dynamics and evade the detector with high frequency. We propose a new detection scheme which employs several non-stationary watermarks. We validate the detection efficiency of the new strategy via numeric simulations and by running experiments on a laboratory testbed. Results show that the proposed strategy is able to detect adversaries using non-parametric methods, but it is not equally effective against adversaries using parametric identification methods.

Keywords: Cyber-Physical Security; Control Theory; Network Security; Networked-Control System; Critical Infrastructures; Attack Detection; Adversary Model; Cyber-Physical Adversary; Attack Mitigation

1 Introduction

In an effort of reducing complexity and costs, traditional industrial control systems are being upgraded with novel computing, communication and interconnection capabilities. Industrial control systems that close the loop through a communication network are hereinafter referred to as *cyber-physical systems*. The adoption of new communication capabilities comes at the cost of introducing new security threats that are required to be holistically handled, both in terms of safety and security (in the traditional ICT sense). The recently coined *cyber-physical security* term refers to the mechanisms that address this specific challenge [1].

The use of inadequate *cyber-physical security* mechanisms may have an adverse effect on a vast number of resources, including assets of private companies, government networks and mission critical infrastructures [2]. The associated costs, especially in terms of loss of business opportunities and the expenses for fixing the incidents, are expected to be reduced. As a consequence, the issue of the assessment of *cyber-physical security* mechanisms is a hot research topic.

In this paper, we address security in industrial control systems. Our focus is centered on integrity issues due to the interconnection between *cyber* and *physical* control domains in networked control systems. More specifically, we focus on the adaptation of physical-layer failure detection mechanisms (e.g., systems for the detection of faults and accidents) to handle, as well, attacks (e.g., replay and integrity attacks conducted by malicious adversaries). We extend the work proposed in [3] which reexamines the security of a specific scheme proposed by Mo *et al.* in [4, 5].

The Mo *et al.* scheme relies on the adaptation of a real-time failure detector based on a *linear time-invariant* model of the system. Built upon *Kalman filters* and *linear-quadratic regulators*, the scheme produces authentication watermarks to protect the integrity of physical measurements communicated over the cyber and physical control domains of a networked control system. Without the protection of the messages, malicious actions can be conducted to mislead the system towards unauthorized or improper actions and affect the availability of the system services.

We show that the Mo *et al.* detection scheme only works against some integrity attacks. We present two adversary models that can evade the Mo *et al.* detector. These adversaries are classified based on the algorithm used to obtain the knowledge of the system dynamics in order to carry out the attack (non-parametric [3] and parametric adversaries [6]).

Contributions – The main contributions of this paper are summarized as follows:

- We reexamine the effectiveness of the attack detector proposed in [4, 5] under new adversary models.
- We show detection weaknesses in [4, 5] under the new adversary models.
- Enhanced detector approaches against the new adversaries are presented and validated via numerical simulations and experiments carried out by using a real testbed.

Paper Organization – Section 2 provides the necessary background for the paper. Section 3 reviews the detector scheme in [4, 5], defines our adversary models and reexamines the security of the detector under the new adversary models. Section 4 adapts the detection scheme in [4, 5] to handle the uncovered limitations, and validates the resulting approach via numerical simulations. Section 5 presents experimental results based on a laboratory testbed. Section 6 discusses the results. Section 7 reviews some related work. Section 8 concludes the paper.

2 Background

2.1 Industrial Control Systems

We assume *Industrial Control Systems* built upon *Supervisory Control and Data Acquisition* (SCADA) technologies and *Industrial Control Protocols*. Such combinations are hereinafter denoted as *Networked Control Systems*. Some more information about these systems and protocols follows.

2.1.1 SCADA

General term that encompasses well-defined types of field devices, such as: (1) Master Terminal Units (MTUs) and Human Machine Interfaces (HMIs), located at the topmost layer and managing all communications; (2) Remote Terminal Units (RTUs) and Programmable Logic Controllers (PLCs), controlling and acquiring data from remote equipment and connecting with the master stations; (3) sensors and actuators.

The MTUs of a SCADA system are located at the control center of the organization. The MTUs give access to the management of communications, collection of data (generated by several RTUs), data storage, and control of sensors and actuators connected to RTUs. The interface to the administrators is provided via the HMIs.

RTUs are stand-alone data acquisition and control units. They are generally microprocessor-based devices that monitor and control the industrial equipment at the remote site. Their tasks are twofold: (1) to control and acquire data from process equipment (at the remote sites), and (2) to communicate the collected data to a master (supervision) station. Modern RTUs may also communicate between them (either via wired or wireless networks).

PLCs are small industrial microprocessor-based computers. Most significant differences with respect to an RTU are in size and capability. An RTU has more inputs and outputs than a PLC, and much more local processing power (e.g., to postprocess the collected data before generating alerts towards the MTU via the HMI). In contrast, PLCs are often represented by pervasive sensors with communication capabilities. PLCs have two main advantages over traditional RTUs: (1) they are general-purpose devices enforcing a large variety of functions, and (2) they are physically compact.

Sensors are monitoring devices responsible for retrieving measurement related to specific physical phenomena and feed them to the controller. Sensors typically convert a measured quantity to an electrical signal, which is later converted and stored as data. Sensors can be seen as the input function of a SCADA system. The data produced by sensors are sent to the upper layers via the RTUs and the PLCs. Finally, *actuators* are control devices, in charge of managing some external devices. Actuators translate control signals to actions that are needed to correct the dynamics of the system, via the RTUs and the PLCs.

2.1.2 Industrial Control Protocols

Protocols for industrial control systems must cover regulation rules such as delays and faults [7]. However, few protocols imposed by industrial standards provide security features in the traditional ICT security sense (e.g., confidentiality, integrity, etc.). Details about such ICT security capabilities of representative protocols follows.

Modbus – Created in the 70s by Modicon, an American company created in 1968 and absorbed by Schneider Electric. Nowadays, it is one of the most spread protocols, probably due to its simplicity and its free license. The Modbus protocol was initially conceived for serial communications. Since 1999, it has been adapted to work over TCP/IP as well. The use of Modbus over TCP/IP allows using SCADA components in heterogeneous environments (i.e., working over IP or serial networks). Moreover, it is possible to use gateways to convert Modbus/TCP to serial Modbus.

From a security standpoint, Modbus does not integrate traditional ICT protection features. For instance, in terms of availability, Modbus/TCP may use some function codes (e.g., ECO4: Server Failure, EC06: Server Busy) as the response of a query from an unavailability device. This way, a controller can point out to availability issues in the absence of responses from one or several devices, or if their responses are error codes. Error handling is performed at the application layer. The availability of a given equipment is also related to the implementation of the layers below Modbus (e.g., TCP/IP layers) and the nature of the media shared for the exchange of data.

The integrity of a Modbus message is validated using the TCP layer for Modbus/TCP or by adding a control field (Cyclic Redundancy Check or CRC) for Modbus/Serial. Nevertheless, without authentication of the message, malicious actions can modify the message and recalculate valid CRCs. This kind of validation must be seen only as a protection against transmission errors. Malicious modification of registers, e.g., time windows, is complex but possible. Replay attacks and, in general, integrity attacks, are also possible.

Finally, the Modbus protocol does not implement encryption of messages. Nevertheless, it is possible to implement encryption by encapsulating Modbus/TCP messages under TLS or IPsec tunnels. Confidentiality is not considered as a crucial property in industrial environments. The deployment of encryption solutions can be seen as detrimental given their complexity (e.g., Public Key Infrastructures, manual deployment of keys, etc.), since it may induce to unnecessary latency delays.

PROFINET – Suite of industrial protocols operating at different network layers, mainly used in Siemens products. For instance, PROFINET IO is an Ethernet-based protocol associated to the PROFINET suite. Implemented over TCP/IP layers, it allows real-time communication and self-configuration. All equipment implementing PROFINET IO must be certified by the PROFIBUS organization. This certification monitors compliance of software, data models and integrity in a PROFINET environment.

In 1999, the first security extension of PROFINET was released. Referred to as PROFIsafe, it leverages from PROFINET IO, acting as one of its upper layers. This allows its deployment over less secure networks maintaining acceptable error rate such as IEEE 802.11 and IEEE 802.15.1 standards, while ensuring high availability and backwards compatibility for legacy equipment. Legacy operations can yet use the standard layer, called Black channel, while other operations requiring safety properties, can use the new layers. Those new layers include elements such as continuity control, acknowledgment timeouts, peer authentication, integrity check CRCs, etc.

Åkerberg ans Björkman uncovered in [8] some flaws in the protocol routines associated to the generation of CRCs. Indeed, PROFIsafe meet standards where intentional attacks are not considered a risk. The protocol does not integrate cryptographic features. It only considers protection to cover from unintentional faults. It should not be considered a protection layer against cyber attacks.

Indeed, the PROFINET Safety Guide [9] indicates the use of VPNs whenever ICT security is required. It is important to emphasize that PROFIsafe has been designed to ensure safety and malfunction (e.g., transmission errors). However, it does not ensure security against intentional malicious acts.

DNP3 – Short for Distributed Network Protocol, version 3, DNP3 is a modern SCADA protocol that includes security extensions, often referred to as DNP3-SA (DNP3 Secure Authentication). DNP3-SA adds features to DNP3 such as protection against replay attacks by ensuring message integrity and authentication. The new features are implemented as new function codes of the original DNP3 protocol. In other words, they are defined at upper layers of the original DNP3 protocol suite, without modifying previous function codes, so that all legacy monitoring and diagnostic tools for DNP3 are still valid. This way, DNP3-SA is compatible with legacy devices that do not require from security support.

The DNP3-SA extensions are expected to be highly scalable. They shall allow changing the security algorithms, keys sizes, and others parameters to meet future conditions of state-of-the-art installations. Nevertheless, both DNP3 and DNP3-SA are relatively complex protocols. The DNP3-SA extensions are relatively young, and the first DNP3-SA products may present vulnerabilities. They do not provide protection against confidentiality attacks either. For the time being, DNP3-SA has not been largely deployed. Given the pace of industrial systems (whose upgrades are often superior to decades), it may take quite long before DNP3-SA is fully tested over large environments.

Ethernet/IP – Nowadays maintained by ODVA Inc. (formerly Open DeviceNet Vendors Association, Inc.) Ethernet/IP is an industrial protocol that relies on the use of CIP (Common Industrial Protocol) over standard Ethernet frames [10]. According to [11], Ethernet/IP was the most widely deployed protocol on industrial environments in 2009, with about 30% of actives nodes (about five million nodes).

Communications over Ethernet/IP can either be in an unconnected mode, using TCP/IP in a client/server model, or they can be in connected mode. In that case, resources are reserved to create a link between two users; UDP/IP and multicast transmissions are employed to make latency as small as possible to enforce real-time constraints. In fact, Ethernet/IP inherits from Ethernet all its security issues. Moreover, Ethernet/IP environments mainly use UDP/IP connections, and rarely use security mechanisms (in a traditional ICT sense), due to performance and real-time constraints [12]. Thus, we can consider some concerns such as: message hijacking, disclosure of communication and configuration details, injection of unauthorized frames, etc.

2.1.3 Networked Control Systems (NCSs)

NCSs are spatially distributed systems whose control loops are connected through communication networks. The communication network connects the different components of a traditional control system, i.e., the controller, sensors, and actuators. Examples include smart grids, smart vehicles, and water distribution systems. The use of a communication network to connect the different components of a control system adds more flexibility in the system and reduces the implementation cost of new installations. However, the use of a communication network to decentralize traditional control systems comes at the price of an increased control design complexity. For instance, the analysis and design of the overall system has also to deal with new theoretical challenges due to, for instance, loss of measurements and time-varying sampling [13]. The integration of the control system (often referred as *physical-space*) with the communication network (*cyber-space*) creates a new degree of interaction between these two domains [14].

Communication protocols used in traditional control systems are required to comply with the constrains imposed by industrial standards (e.g., to cover regulation roles such as delay and faults). Indeed, prominent industrial control protocols (e.g., Modbus, PROFINET and Ethernet/IP), are not designed to provide security from a traditional information or network perspective. However, current NCSs use these protocols over TCP/IP or UDP/IP communications (e.g., Modbus over TCP, PROFINET over TCP, Ethernet/IP over TCP or UDP). Although such combinations can provide some security elements at either their transport or network layers, this is not enough to ensure control-data protection. At the same time, traditional control systems come with already existing mechanisms to handle failures. Such mechanisms are expected to detect faults and avoid accidents. Nevertheless, classical mechanisms conceived to detect failures in control systems are not able to detect intentional actions from malicious adversaries holding enough knowledge about the systems. Cryptography can be used to ensure the integrity of control-data. This may partially solve the aforementioned problem. Without underestimating cryptographic solutions, in this paper we consider a complementary solution proposed by Mo *et al.* [4, 5] to protect the integrity of the physical sensor measurements by means of authentication watermarks. Such watermarks are enforced by adapting a real-time failure detector based on linear time invariant models of industrial control systems. This security solution allows to improve system security in cases where, e.g., cyber-adversaries bypass security measures at upper protocol layers. We present some representative examples in the following section.

2.2 Cyber-Physical Attacks

The use of communication networks and IT components in traditional control systems paves the way to new vulnerability issues. The attacks against these setups are named cyber-physical attacks. These threats may target physical processes through the network. In [15], Texeira *et al.* propose a taxonomy of cyber-physical attacks based on the resources of the adversaries. Such resources are mainly measured in terms of adversary knowledge (e.g., *a priori* knowledge of the adversary about the system and its security measures). Indeed, the knowledge of the adversary about the system is the main resource in order to build up complex attacks, as well as to make them undetectable. Based on the degree of the adversary knowledge, the attacks may succeed at violating system properties, such as availability and integrity, as well as to obtaining operational information about the system to make the attacks undetectable.

Figure 1 shows an adversary conducting a cyber-physical attack. The $\bigoplus$ symbol in the figure represents a *summing junction*, i.e., a linear element that outputs the sum of a number of input signals. In a nutshell, the adversary succeeds at modifying some plant measurements, by recording and replicating previous measurements corresponding to normal operation conditions. Then, the adversary modifies the control input u to affect the system state and disrupt normal operation conditions. If, on one hand, the adversary is not required to have the knowledge of the system process model, on the other hand access to all sensors (i.e., it has access to all components of the vector y) or insecure communication protocols is required to carry out a successful attack. This type of adversary is undetectable with a monitor detector which only verifies faulty measurements.

Adversaries conducting more powerful injection attacks are expected to modify some of the plant measurements, either by targeting the individual sensors or their communication channels in case of insecure communication protocols. Sophisticated variation of the attack in Figure 1 include (1) bias-injection cyber-physical attacks, in which the new data injected by the adversary corresponds to a bias from the legitimate data, with the aim of leading the system to wrong control decisions (e.g., to cause malfunction in the long-term); and (2) geometric-injection cyber-physical attacks, in which the bias is gradually injected. The attack may remain undetected when data compatible with the system dynamics are injected, potentially leading the system to irreversible damages [15]. Another undetectable attack is the covert attack, where the adversary knows perfectly the system model. This attack is defined in [16] and the author concludes that it is not possible to be detected.

Techniques to prevent the aforementioned cyber-physical attacks exist. In [17], Arvani *et al.* describe a signal-based detector method, using discrete wavelet transformations. Do *et al.* study in [18] strategies for handling cyber-physical attacks using statistical detection methods. Mo *et al.* propose in [4, 5] the use of watermark-based detection by adapting traditional failure detection mechanisms (e.g., detectors to handle faults and errors). In the following sections, we elaborate further on the watermark-based technique by Mo *et al.*, discuss about some security limitations and propose an improved technique.

3 Watermark-based Attack Detection

In [4, 5], a watermark-based strategy is proposed to detecting replay and injection attacks against cyber-physical systems. This section reviews the mechanism proposed in [4, 5] and assesses its performance when a new adversary model, that we name *cyber-physical adversary*, is employed. In particular, this section is organized as follows: in Section 3.1 we provide some necessary definitions and background concerning the class of control systems considered in this paper; Section 3.2 describes the attack detection scheme proposed in [4, 5]; in Section 3.3, we propose the cyber-physical adversaries; finally, in Section 3.4 we show methods that can be employed by the adversary to mislead the watermark-based detector. The notation used hereinafter is summarized in Table 1.

3.1 Definitions and Background

We consider plants of industrial control systems that can be mathematically modeled as discrete linear time-invariant (LTI) systems. It is worth mentioning that a mathematical model provides a rigorous way to describe the dynamical behaviour of a given system. In the following we denote with $X(z)$ the Z-transform of a signal x_t, where t is the time-discrete variable. Such class of systems can be described as follows:

$$zX(z) = AX(z) + BU(z) + W(z) \tag{1}$$

where $X(z)$ is the Z-transform of $x_t \in \mathbb{R}^n$, the vector of the state variables (or state) at the time step t, $U(z) = Z\{u_t\}$ is the Z-transform of $u_t \in \mathbb{R}^p$, the control signal, and $W(z)$ is the Z-transform of $w_t \in \mathbb{R}^n$, the process noise that is assumed to be a white noise with zero mean and variance Q, *i.e.* $w_t \sim N(0, Q)$. Moreover, $A \in \mathbb{R}^{n\times n}$ and $B \in \mathbb{R}^{n\times p}$ are respectively the *state* matrix and the *input* matrix.

A static relation maps the state $X(z)$ to the system output $Y(z)$, the Z-transform of $y_t \in \mathbb{R}^m$:

$$\begin{aligned} Y(z) &= CX(z) + V(z) \\ &= C(zI - A)^{-1}(BU(z) + W(z)) + V(z) \end{aligned} \tag{2}$$

where $C \in \mathbb{R}^{m\times n}$ is the output matrix. The value of the output vector $Y(z)$ represents the measurement produced by the sensors that is affected by a noise $V(z)$ that is the Z-transform of v_t assumed as a white noise with zero mean and variance R, *i.e.* $v_t \sim N(0, R)$.

For the class of systems defined above, a widely-used and effective control technique is the *Linear Quadratic Gaussian* (LQG) approach. The overall goal of an LQG controller is to produce a control law $U(z)$ such that a quadratic cost J, that is function of both the state x_t and the control input u_t, is minimized:

$$J = \lim_{n\to\infty} E\left[\frac{1}{n}\sum_{j=0}^{n-1}(x_j^T \Gamma x_j + u_j^T \Omega u_j)\right] \tag{3}$$

where Γ and Ω are positive definite cost matrices [19].

It is well-known that such a control problem has, under some unrestrictive technical conditions, an optimal solution that, thanks to the separation principle, is made of two components that can be designed independently:

1 a *Kalman filter* that, based on the noisy measurements, produces an optimal state estimation $\hat{X}(z)$ of the state $X(z)$;

2 a *Linear Quadratic Regulator* (LQR) that, based on the state estimation $\hat{X}(z)$, provides the control law $U(z)$ that solves the LQR problem (cf. Equation (3)).

Let us briefly illustrate how these two components are designed. The Kalman filter estimates the state as follows:

- Predict (*a priori*) system state $\hat{X}^{(-)}(z)$ and covariance:

$$\begin{aligned}\hat{X}^{(-)}(z) &= A\hat{X}z^{-1} + BUz^{-1}\\ P^{(-)} &= A(P^{(+)}z^{-1})A^T + Q\end{aligned}$$

- Update parameters and (*a posteriori*) system state and covariance:

$$\begin{aligned}K_f &= (P^{(-)}C^T)(CP^{(-)}C^T + R)^{-1}\\ \hat{X}^{(+)} &= \hat{X}^{(-)}(z) + K_f(Y(z) - C\hat{X}^{(-)}(z))\\ P^{(+)} &= (I - K_f C)P^{(-)}\end{aligned}$$

where K_f and $P^{(+)}$ denote, respectively, the Kalman gain and the *a posteriori* error covariance matrix, I is the identity matrix.

The optimal control law $U(z)$ provided by the LQR is a linear controller:

$$U(z) = L\hat{X}^{(+)}(z) \tag{4}$$

where L denotes the feedback gain of a linear-quadratic regulator (*LQR*) which minimizes the control cost (cf. Equation 3) and it is defined as follows (cf. [4, 5] for further details):

$$L = -(B^T SB + \Omega)^{-1}B^T SA,$$

with S being the matrix that solves the following discrete time algebraic Riccati equation:

$$S = A^T SA + \Gamma - A^T SB[B^T SB + \Omega]^{-1}B^T SA.$$

3.2 The χ^2 Detector

This section briefly describes the detection scheme proposed in [4, 5]. The procedure is applicable to discrete LTI plants controlled by a LQG controller as detailed in Section 3.1.

Before presenting the detection scheme, we provide a definition of the adversary model considered in [4, 5]:

Definition 3.1 An attacker that has the ability to eavesdrop all the messages containing the sensor outputs $Y(z)$ and to inject messages with an arbitrary signal $Y'(z)$ to conduct malicious actions is defined a *cyber adversary*.

Remark 3.2 Notice that the definition given above does not suppose that the attacker possesses (or makes attempts to gather) any knowledge about the system model.

We denote with $U^*(z)$ the output of the LQR controller given by Equation (4) and with $U(z)$ the control input that is sent to the plant (cf. Equation (1)). The idea is to superpose to the optimal control law $U^*(z)$ a watermark signal, $\Delta U(z) = Z\{\Delta u_t\}$, Z-transform of $\Delta u_t \in \mathbb{R}^p$ that serves as an authentication signal. Thus, the control input $U(z)$ is given by:

$$U(z) = U^*(z) + \Delta U(z) \tag{5}$$

The watermark signal is a Gaussian random signal with zero mean that is independent both from the state noise $W(z)$ and the measurement noise $V(z)$. Such an authentication watermark is expected to detect replay and integrity attacks modeled by the adversary defined above. Now that the optimal control law $U^*(z)$ is equipped with the authentication signal $\Delta U(z)$, a *detector* – physically co-located with the controller – can be designed having the goal of generating alarms when an attack takes place. Towards this end, [4, 5] propose to employ a χ^2 detector, a well-known category of real-time anomaly detectors classically used for fault detection in control systems [20], for the purpose of attack detection.

Figure 2 shows the overall control system equipped with the attack detector proposed in [4, 5].

An *alarm signal* g_t is computed based on the residues $r_t = Z^{-1}\{r(z)\}$, where $r(z) = Y(z) - C\hat{X}^{(-)}$ generated from the estimator. Then, g_t is compared with a threshold γ to decide whether the system is in a normal state. The threshold is tuned to minimize false alarms [4, 5]. The alarm signal g_t is computed as follows

$$g_t = \sum_{i=t-w+1}^{t} (r_i)^T \mathcal{P}^{-1}(r_i) \tag{6}$$

where w is the size of the detection window and $\mathcal{P} = (CPC^T + R)$ is the co-variance of an independent and identically distributed (i.i.d.) Gaussian input signal from the sensors.

The system is considered not under attack if $g_t < \gamma$, otherwise if $g_t \geq \gamma$ the system is considered to be under attack and the detector generates an alarm.

3.3 Cyber-physical Adversary

Let us assume the system employs the detector described in Section 3.2, so that the controller superposes its output with an authentication watermark $\Delta U(z)$. At steady-state, i.e. after the transient has been exhausted, the output of the system can be considered as the sum of its steady-state value and a component that is due to watermark signal that shall be only known by the controller.

Let us now introduce an enhanced adversary that is aware of the fact that the system employs the χ^2 detector presented above. Since the detector is based on a stationary watermark signal $\Delta U(z)$, we will show that an adversary that is able to extract the model of the system from the control law $U(z)$ and the sensor measurement $Y(z)$, is able to conduct an attack while remaining undetected.

Definition 3.3 An attacker that, in addition to the capabilities of the cyber adversary, is also able to eavesdrop the messages containing the output of the controller $U(z)$ with the intention of improving its knowledge about the system model using a *parametric or non-parametric identification model*, is defined as a *cyber-physical adversary*.

Based on the way to model the system's behaviour, two different cyber-physical adversaries can be defined.

Definition 3.4 An attacker that only uses the previous input and output of the system to identify the system model is defined a *non-parametric cyber-physical adversary*.

Remark 3.5 A non-parametric cyber-physical adversary can use, e.g., a Finite Impulse Response (FIR) model identification tool to identify the system model [21]. In Figure 2, the signals $u'_t = Z^{-1}\{U'\}$ and $y'_t = Z^{-1}\{Y'\}$ are assumed to be respectively the output of the controller and the output of the measurement when an attack is taking place. We denote with $\Delta u' = Z^{-1}\{\Delta U'\}$ the watermark guessed by the non-parametric cyber-physical adversary.

Definition 3.6 An attacker able to estimate the parameters of the system using input and output data to mislead the controller detector is defined as a parametric cyber-physical adversary.

Remark 3.7 A parametric cyber-physical adversary is able to estimate the parameters of the system using input and output data to mislead the controller detector. This adversary can use an ARX (autoregressive with exogenous input) model or an ARMAX (autoregressive-moving average with exogenous input) to estimate the model [22].

We assume that the main constraint of this adversary is the energy spent to eavesdrop and analyze the communication data, i.e., the number of samples eavesdropped to obtain the system model parameters.

Proposition 3.8 A cyber-physical adversary that is able to exactly estimate the system controlled by the controller cannot be detected by the χ^2 detector (cf. Equation 6).

Proof Without loss of generality, we assume an attack is started at time T_0 and we compute the residues r_t for $t \in [T_0, T_0 + T - 1]$:

$$Z\{r_t\} = Y'(z) - C\hat{X}^{(-)}z^{-(t-T_0-1)} \qquad (7)$$

Moreover, it is easy to show that the following holds:

$$\begin{aligned}
\hat{X}^{(-)}z^{-(t-T_0-1)} &= Z\{\hat{x}'_{t|t-T} + \mathcal{A}^{t-T_0}(\hat{x}_{T_0|T_0-1} - \hat{x}'_{T_0|T_0-1}) \\
&\quad + \sum_{i=0}^{t-T_0-1} (\mathcal{A}^i B(\Delta u_{t-1-i} - \Delta u'_{t-1-i}))\} \\
&= \hat{X}'^{(-)}z^{-(t-T-1)} + (\mathcal{A}^{t-T_0})(\hat{X}^{(-)} - \hat{X}'^{(-)})z^{-(t-T_0-1)} \\
&\quad + \sum_{i=0}^{t-T_0-1} (\mathcal{A}^i B(\Delta U(z) - \Delta U'(z)))z^{-(t-T_0-1-i)} \qquad (8)
\end{aligned}$$

where $\hat{X}'$ is the estimated state when the system is under attack and $\mathcal{A} = (A + BL)(I - KC)$ is a stable matrix [4, 5]. Substitution of (8) in (7) yields:

$$\begin{aligned}
Z\{r_t\} &= \underbrace{Y'(z) - C\hat{X}'^{(-)}}_{\text{First term}} \\
&- \underbrace{C(\mathcal{A}^{t-T_0})(\hat{X}^{(-)} - \hat{X}'^{(-)})z^{-(t-T_0-1)}}_{\text{Second term}} \\
&- \underbrace{C\sum_{i=0}^{t-T_0-1} (\mathcal{A}^i B(\Delta U(z) - \Delta U'(z))z^{-(t-T_0-1-i)})}_{\text{Third term}}
\end{aligned}$$

Let us consider separately the three terms in the equation written above: the first term follows the same distribution of $Y'(z) - C\hat{X}'^{(-)}$; since $\mathcal{A}$ is asymptotically stable – i.e. all its eigenvalues are inside the open unit disk of the complex plane – the second term converges exponentially fast to zero. In fact, the entries of $\mathcal{A}^{t-T_0}$ converge exponentially fast to zero. Now, if the third term is equal to zero, the dynamics of $Z\{r_t\}$ would recover the dynamics of the residues when no attack is undergoing and thus, the attack would not be detected. Under the hypothesis of this proposition, the adversary knows exactly the watermark signal and thus $\Delta U(z) = \Delta U'(z)$ which makes the third term equal to zero and concludes the proof. □

3.4 Acquiring the Watermark Signal Model

Motivated by Definition 3.4, we show now a practical method that can be used to acquire the watermark signal Δu_t. In particular, we propose an adversary that employs a well know LMS (Least Mean Square) adaptive FIR filter, a non-parametric identification model, with the purpose of running an online identification of the system model. With the identified model, it is possible to obtain the watermark and, finally, using it to authenticate messages with the aim of driving the system to an undesired state.

We denote with p the LMS filter order and with μ its step size. The step size μ is upper bounded by $2/\lambda_{max}$, where λ_{max} is the maximum eigenvalue of the auto-correlation matrix $R = E[XX^H]$, where X^H is the Hermitian transpose, or conjugate transpose, of X. Observe that if μ is chosen too small, the time to converge to optimal weights tends to be large [23]. The adversary initializes the weight matrix $\mathcal{W}$ to be equal to the zero matrix. Then, the adversary's algorithm shown in Algorithm 1, is run online. It is worth noting that in this algorithm $X[x(t-p+1), ..., x(t)]$ is the input signal vector, $e(t)$ is the error vector, $\bar{e}(t)$ is its complex conjugate, and $d(t)$ is the desired output signal.

Algorithm 1 Non-parametric Cyber-Physical Adversary Algorithm

```
procedure ADVERSARY ALGORITHM
    k ← length of eavesdropped data
    p ← filter order
    $j \leftarrow p$
top:
    if $j < k$ then $i \leftarrow 1$.
loop:
    if $i \leqslant p$ then
        $ini \leftarrow j - p + 1$.
        $e(ini) \leftarrow d(ini) - \mathcal{W}^T X[x(ini), \dots x(j)]$.
        $W \leftarrow \mathcal{W} + \mu \bar{e}(ini) X[x(ini), \dots x(j)]$.
        $j \leftarrow j + i$.
        $i \leftarrow i + 1$.
        goto loop.
        close;
    goto top.
```

Once the system model has been identified, the adversary is able to extract the watermark and to carry out the replay attack. In particular, the adversary follows the steps described below:

1 *Eavesdropping of $U(z)$ and $Y(z)$ and decomposition*: The adversary captures both the control law $U(z)$ and the sensors output $Y(z)$ to make the decomposition between the information data and the watermark using the LMS filter as a noise cancellation adaptive filter. With this first step, we are able to separate $U^*(z)$ and the watermark $\Delta U(z)$ starting from $U(z)$. Notice that, since the system is linear, it follows from the superposition principle that $Y(z) = Y^*(z) + Y^{\Delta U}$, being $Y^*(z)$ the output due to $U^*(z)$ and $Y^{\Delta U}(z)$ the output due to the watermark $\Delta U(z)$.

2 *Acquiring the weight matrix, $\mathcal{W}$*: To acquire the weight matrix $\mathcal{W}$, the adversary uses the LMS adaptive filter described before, as a system identification method.

3 *Computing the attack sensor measurement $Y'(z)$*: The adversary attacks the system by sending fake sensor measurements $Y'(z)$, where $Y^{\Delta U}(z)$ is computed using the watermark $\Delta U(z)$ as follows:

$$Y^{\Delta U}(z) = \mathcal{W}^T \Delta U(z)$$

and $Y'(z) = Y^*(z)z^{-1} + Y^{\Delta U}(z)$.

In the remainder of this section, we show via numerical simulations that the detection mechanism proposed in [4, 5] is not sufficiently robust and is not able to detect cyber-physical adversaries (cf. Section 3.3) that are able to identify the system model by eavesdropping the data channel.

In order to simulate the scenario proposed in this section, we use a simplified version of the Tennesse Eastman control challenge problem [24] (also used in [25]). This system simulates a MIMO system of order $n = 7$ with $p = 4$ inputs and $m = 4$ outputs. In particular, the model of the discrete LTI system described by Equations (1)-(2) is defined by the following matrices:

$$\text{A=}\begin{bmatrix} 0.987 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.895 & -0.025 & 0 & 0 & 0 & 0 \\ 0 & 0.036 & 0.999 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -0.008 & 0 & 0 \\ 0 & 0 & 0 & 0.005 & 0.960 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.999 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.990 \end{bmatrix}, \quad \text{B=}\begin{bmatrix} 0.149 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.071 \\ 0 & 0 & 0 & 0.001 \\ 0.380 & 0 & -0.096 & 0 \\ 1.000 & 0 & -0.096 & 0 \\ 0 & 0.038 & 0 & 0 \\ 0 & 0 & 0 & 0.075 \end{bmatrix},$$

$$\text{C=}\begin{bmatrix} 0.151 & -0.076 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.040 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.133 \end{bmatrix}.$$

Moreover, the co-variance matrices are equal to $Q = 0.01I$ and $R = I$, whereas the cost matrices are $\Gamma = 1.5I$ and $\Omega = 10I$.

To validate our approach, we compare the system dynamics considering the two adversaries described above. Figures 3(a) and 3(c) show the plant dynamics and the state estimated by the controller in the case of a cyber adversary. Figure 3(a) shows that the adversary is able to drive the state components to an undesired value. Nevertheless, the controller, misled by the adversary, does not perceive such situation (cf. Figure 3(c)). Figure 3(b) and Figure 3(d) show the dynamics of the plant and the ones of the controller under a non-parametric cyber-physical adversary model which exhibits the same behavior described above for the case of the cyber adversary.

Let us now contrast the performance of the detector described in Section 3.2 when detecting either the cyber or the non-parametric cyber-physical adversary. Towards this end, Figure 3(e) and Figure 3(f) show the value of the alarm signal g_t produced by the same χ^2 detector in the case of the cyber adversary (cf. Figure 3(e)) and the non-parametric cyber-physical adversary (cf. Figure 3(f)). Figure 3(e) shows that the detector is able to detect the cyber adversary thanks to the added watermark signal as soon as the attack starts at $t = 700$s. However, Figure 3(f) shows that the same detector is not able to detect the non-parametric cyber-physical adversary since g_t does not exceed the threshold γ during the attack. In order to quantify the detector performance, we define a Detection Ratio (DR) metric as follows:

$$DR = \frac{\sum_{t=T_0}^{T_0+T_a} \mathbb{1}_{g_t \geq \gamma}}{T_a} \tag{9}$$

where T_a is the attack duration, and $\mathbb{1}$ is the indicator function whose output is equal to 1 if the Boolean condition given as its argument ($g_t \geq \gamma$) is true; or 0 otherwise. In a nutshell, $DR \in [0, 1]$ can be considered as an efficiency index for the detector: DR is equal to one when the attack is always detected; and it is equal to zero when the attack is always undetected.

Figure 4 shows the CDF (Cumulative Distribution Function) of the detection ratio obtained by measuring DR for 200 simulations both in the case of the non-parametric cyber-physical and the cyber adversary. The figure shows that the detection scheme proposed in [4, 5] is able to provide a median detection ratio that is larger than 0.9 when a cyber adversary attacks the system. However, using a non-parametric cyber-physical adversary that acquires the watermark, the median detection ratio drops to around 0.2. This quantitatively shows that the detection strategy proposed in [4, 5] is not sufficiently robust for security.

4 Multi-Watermark based Attack Detection

In the previous section, we have defined three different kinds of adversaries who use different vulnerabilities of a control system to carry out attacks; cyber-adversary, non-parametric cyber-physical adversary, and parametric cyber-physical adversary. In this section, we propose a detection scheme that extends the one presented in [4, 5], in order to detect non-parametric cyber-physical adversaries. We also study the performance loss of the new detection scheme with regard to the one presented in [4, 5].

4.1 On the use of Multi-watermark Signals

The goal of our new detection scheme is to increase the difficulty in retrieving the authentication watermark $\Delta U(z)$ from the control signal $U(z)$, so that the probability of detecting an attack from a non-parametric cyber-physical adversary can be increased. We assume that the control system under attack employs exactly the same type of controller and the same detection strategy presented in Section 3.2. The only difference in the proposed detection scheme is the way that the watermark signal $\Delta U(z)$ is generated. The control input $U(z)$, as in the case of the detection scheme presented in Section 3.2, is computed as the superposition of the optimal control signal $U^*(z)$ produced by the LQR controller and a given multi-watermark signal $\Delta U(z)$. The idea is to construct the authentication watermark signal by switching between N different and independent processes with different co-variance and average (offsets). More precisely, the non-stationary watermark ΔU is obtained by periodically switching, with a period T, between N signals $\Delta U^{(i)}$, with $i \in \mathcal{I} = \{0, 1, \ldots, N-1\}$, extracted by different stochastic processes. Hence, the watermark signal $\Delta U(z)$ can be formalized as follows:

$$\Delta U(z) = Z\{\Delta u_t^{(s(t,T))}\} \tag{10}$$

where $s : \mathbb{N} \times \mathbb{R} \rightarrow \mathcal{I}$ is a static function that maps the time sample t and the switching period T to an element of the index set $\mathcal{I}$, defined as follows:

$$s(t,T) = \left\lfloor \frac{1}{T} \mod (t, NT) \right\rfloor \tag{11}$$

where $\mod(x, y)$ is the modulo operator and $\lfloor \cdot \rfloor$ is the floor function.

By using the proposed watermark (cf. Equation 10), we now have an adaptive protection mechanism with two main configurable parameters: the number of distributions N and the switching frequency $f = 1/T$. It is worth to notice that the original watermark signal described in Section 3.2 is recovered when $f \rightarrow 0$ and when $\Delta U^{(0)}$ being a stationary zero mean Gaussian process.

4.2 Single-watermark LQG Structure Performance Loss

In this section, we compute the increment of cost in the LQG structure due to the single-watermark added to the control input. This supplementary cost is the degradation in the performance of the system, as shown in [5], and can be defined as follows:

$$J = J^* + \Delta J_s \tag{12}$$

where J^* is the optimal cost of the system described in Section 3.1; and ΔJ_s is the increment of cost due to the use of the single-watermark-based detector. In the following, we develop the cost of the system in the time domain.

$$\begin{aligned} J &= \lim_{n\to\infty} \frac{1}{n} \sum_{j=0}^{n-1} E\left[(x_j^T \Gamma x_j + u_j^T \Omega u_j)\right] \\ &= \lim_{n\to\infty} \frac{1}{n} \sum_{j=0}^{n-1} \left[tr(\Gamma Cov(x_j)) + tr(\Omega Cov(u_j))\right] \end{aligned} \quad (13)$$

where x_t is defined as:

$$x_t = \mathcal{L}_x(w_t, v_t) + \sum_{j=t-1}^{\infty} (A+BL)^j BVar(\Delta u_{-j}) \quad (14)$$

and u_t as follows:

$$u_t = \mathcal{L}_u(w_t, v_t) + \sum_{j=t-1}^{\infty} (A+BL)^j BVar(\Delta u_{-j}) + Var(\Delta u_t) \quad (15)$$

where $\mathcal{L}_x$ and $\mathcal{L}_u$ are linear functions. Their definition is not relevant for the target of this paper, but the reader can find the details in [5]. Assuming that the noise of the system and the watermark are independents, we can define the increment of cost due to the single-watermark as [5]:

$$\begin{aligned} \Delta J_s &= \Gamma tr\left[Cov\left(\sum_{j=t-1}^{\infty} (A+BL)^j BVar(\Delta u_{-j})\right)\right] \\ &\quad -L\Omega tr\left[Cov\left(\sum_{j=t-1}^{\infty} (A+BL)^j BVar(\Delta u_{-j}) + Var(\Delta u_t)\right)\right] \end{aligned} \quad (16)$$

4.3 Multi-watermark LQG Structure Performance Loss

Let us now evaluate the increment of the cost generated by the multi-watermark based detector, ΔJ_m, and next compare the cost generated by the single-watermark and the multi-watermark. The equation of ΔJ_m, is given by:

$$\begin{aligned} \Delta J_m(t) &= tr[\Gamma Cov(\sum_{j=t-1}^{\infty} (A+BL)^j B(Var(\Delta u_{-j}^{(i)}) + E[\Delta u_{-j}^{(i)}]))] \\ &\quad +tr[L\Omega Cov(\sum_{j=t-1}^{\infty} (A+BL)^j B(Var(\Delta u_{-j}^{(i)}) + E[\Delta u_{-j}^{(i)}]) \\ &\quad +(Var(\Delta u_t) + E[\Delta u_t])] \end{aligned} \quad (17)$$

where $Var(\Delta u_t^{(i)})$ and $E[\Delta u_t^{(i)}]$ are respectively the variance and the mean of the watermark sent at moment t. The performance loss of the LQG structure depends linearly on the variance and the mean of the multi-watermark $\Delta u_t^{(i)}$ for each T samples.

The following theorem shows the difference between the performance loss due to the single-watermark, Δu, and the performance loss due to the multi-watermark, $\Delta u^{(i)}$.

Theorem 4.1 *Let us assume that a watermark is a Gaussian signal with a couple of parameters to be characterized, the mean and the variance. The multi-watermark distribution is defined as* $M_w = N(E[\Delta u^{(i)}], Var(\Delta u^{(i)}))$*, and the single-watermark distribution is defined as* $S_w = N(E[\Delta u], Var(\Delta u))$*. If we define for the multi-watermark* β *as:*

$$\beta = E[\Delta u^{(i)}] + Var(\Delta u^{(i)}) \qquad \forall i \in \mathcal{I} \tag{18}$$

and for the single-watermark ϵ *as:*

$$\epsilon = E[\Delta u] + Var(\Delta u) \tag{19}$$

where ϵ *and* β *are constant for single and multi-watermark respectively. Then, we can conclude that the performance loss of both approaches is equal if* $\epsilon = \beta$*.*

Proof If we assume that $E[\Delta u] = 0$ for the single-watermark, we can prove the theorem as follows:

$$\begin{aligned}
&\mathit{Diff}(\Delta J_m(t), \Delta J_s(t)) = \Delta J_m(t) - \Delta J_s(t)\\
&\quad = \Gamma tr\left[Cov\left(\sum_{j=t-1}^{\infty}(A+BL)^j B(\beta_{-j} - \epsilon_{-j})\right)\right]\\
&\qquad + L\Omega tr\left[Cov\left(\sum_{j=t-1}^{\infty}(A+BL)^j B(\beta_{-j} - \epsilon_{-j}) + (\beta_t - \epsilon_t)\right)\right] = 0
\end{aligned}$$

□

Remark 4.2 Note that using Theorem 4.1, the performance loss due to the multi and the single-watermark is equal. The assumption of the equal performance loss allows compare both approaches under the same conditions. This can be formally stated as follows:

$$E[\Delta u^{(i)}] + Var(\Delta u^{(i)}) = \epsilon = \beta. \tag{20}$$

4.4 Numerical Validation of the Multi-Watermark Detector against non-Parametric Cyber-physical Adversaries

This section validates through numerical simulations the detection scheme proposed in Section 4.1. In particular, we aim at showing that the proposed watermark signal is able to detect non-parametric cyber-physical adversaries (cf. Section 3.3) with a higher detection ratio with respect to the one obtained with the watermark proposed in [4, 5]. Towards this end, we employ a MIMO system with four inputs and four outputs described by the following matrices:

$$\text{A=}\begin{bmatrix} 0.3991 & 0.07113 & 0.1573 & -0.1274 & 0.0226 & -0.0225 & 0.001 \\ 0.003 & -0.07588 & -0.005092 & -0.03893 & 0.09917 & -0.0168 & 0 \\ -0.1974 & -0.01849 & 0.0453 & 0.1579 & -0.1597 & 0.1405 & -0.002 \\ -0.1246 & -0.0726 & 0.1515 & -0.1148 & 0.5156 & -0.0665 & 0 \\ 0.4309 & -0.1204 & 0.09715 & 0.055 & 0.2406 & 0.2812 & 0.0001 \\ -0.0827 & -0.01092 & 0.1234 & -0.1318 & 0.0348 & 0.469 & 0 \\ 0.08312 & -0.0829 & 0.081 & 0.0358 & 0.1124 & 0.02475 & 0.4469 \end{bmatrix},$$

$$\text{B=}\begin{bmatrix} 0.947 & 0 & 0.002 & -0.021 \\ -0.0086 & 0 & -0.406 & 0.02829 \\ -0.8708 & 0.0011 & 0.0011 & -0.106 \\ 0.4872 & 0.002 & 0.188 & -0.041 \\ 0.1233 & 0 & 0.01 & -0.9344 \\ 0 & 0 & 0 & 0.521 \\ 0 & 0.7658 & 0 & 0 \end{bmatrix},$$

$$\text{C=}\begin{bmatrix} -1.102 & 0.302 & -0.1004 & 0.0386 & 0.053 & 0.0891 & 0 \\ 0 & 0.114 & -0.0132 & -1.087 & 0.116 & 0.051 & 0.905 \\ 0.0003 & 1.593 & -0.002 & 0 & 0.093 & 0 & 0.0428 \\ -0.163 & -0.0712 & -0.1074 & 0 & 0 & -0.7443 & 0.089 \end{bmatrix}.$$

and co-variance matrices equal to $Q = 0.2I$ and $R = I$. The positive definite cost matrices Γ and Ω are both equal to the identity matrix. The simulation is based on Matlab and Simulink models of the plant, as well as the models of the non-parametric cyber-physical adversaries. The attacks start at $t = 700$s. We use three different distributions (i.e., $N = 3$) switched at random: a Gaussian, a Rician and a Rayleigh distribution. Table 2 shows the co-variance and offset configured in the simulations for each distribution.

To validate the proposed attack detection scheme, we compare the system dynamics considering two different switching frequencies. We have simulated a high frequency switching watermark configured to switch each seven time samples, and a low frequency switching configured to switch each 20 time samples. Figures 5(a) and (c) show the plant dynamics and the dynamics of the states estimated by the controller in the case of a switching frequency watermark configured to seven time samples and a cyber-physical adversary attack. Figure 5(a) shows that the adversary is able to drive the state to an undesired value. Nevertheless, the controller misled by the adversary, does not perceive such situation (cf. Figure 5(c)). Figures 5(b) and (d) show the plant dynamics and the dynamics of the state estimated by the controller when the watermark is switched each 20 time samples. The dynamics show exactly the same behavior described above.

Figures 5(e) and 5(f) show the dynamics of the alarm signal g_t produced by the detector, respectively in the case of high and low switching frequency. Notice that switching the watermark distributions at a high frequency provides better detection performances compared to the case of a low switching frequency.

To quantify the effectiveness of the proposed detection scheme, we compute the detection ratio DR as a function of the switching frequency. In particular, for each considered frequency f we run 200 Monte Carlo simulations (with randomly generated system parameters) both in the case of the cyber-physical and the cyber adversary, and we compute the CDF (Cumulative Distribution Function) of the detection ratio.

We start by confronting the performance obtained with the detection strategy based on multiple watermark signals proposed in this paper with that proposed in [4, 5] in both the case of a cyber-physical and a cyber adversary. In the case of the proposed multi-watermark strategy we consider two switching frequencies $f_L = 0.05$Hz (switching watermark each 20 time steps) and $f_H = 0.14$Hz (switching watermark each seven time steps). The results of this comparison are shown in Figure 6. Let us focus on the detection strategy proposed in [4, 5]: as shown before, the detector is able to consistently detect a cyber attack but it performs poorly when a cyber-physical adversary attacks the system. Nevertheless, the proposed detection strategy based on multiple watermarks is able to provide a higher detection ratio. In particular, we notice that the detector employing a higher switching frequency

f_H provides better performances with respect to the case of using the lower switching frequency f_L.

In the following we are interested in analyzing in more details the performance of the proposed detection strategy when the switching frequency f is varied, to give a more in depth explanation of the anecdotal evidence shown above. Towards this end, Figure 7(a) shows the CDF of the detection ratio obtained when the switching frequency varies in the range $[0.10, 0.33]$Hz. In this case, we only consider cyber-physical adversaries. The CDFs shown in Figure 7(a) confirm that when the switching frequency increases, the detection ratio is also increased.

To finish this section, we provide in Figure 7(b) the median detection ratio function of f. The figure also contains the case $f = 0$ that corresponds to the detection strategy proposed in [4, 5] – used as a baseline. The shaded area in the figure corresponds to the values of the detection ratio between the 25-th and the 75-th percentile for each f. As expected, the figure shows that by increasing the frequency we are able to obtain a detection ratio that goes from around 0.2 in the case of the baseline approach of [4, 5] to around 0.6 in correspondence of $f = 0.33$Hz.

Observe that the probability of false alarms without attack (often referred in the related literature se false positives) is fixed, $\alpha = 1\%$. Notice as well that false negatives (i.e., undetected real attacks), are inversely proportional to the detection ratio for each switching frequency.

4.5 Efficiency Validation

We have validated above the multi-watermark detector using a static function, $\mathcal{I}$, to define the multi-watermark and different performance loss between single and multi-watermark. We next present the results and validations obtained for a system with the same performance loss between single and multi-watermark detector and where the multi-watermark is generated from a non-static function, $\mathcal{I}_d$. Figure 8 shows the result obtained after running 200 Monte Carlo simulations of a system with single and multi-watermark detector against a non-parametric cyber-physical adversary. In this simulation, single and multi-watermark detector have 30% performance loss, ΔJ, with respect to the optimal cost. Moreover the watermark uses a dynamic function to define the multi-watermark. In Figure 8(a), we show the result of using both single and multi-watermark for a system of order four. We can confirm that the multi-watermark detector, with the same performance loss as the single-watermark detector, has a higher detection ratio. Figure 8(b) shows the result of single and multi-watermark for a higher system of order, 25. On one hand, these results confirm that the multi-watermark detector is able to detect properly non-parametric cyber-physical adversaries. Additionally, we can conclude that the detection ratio increases with the complexity of the system. On the other hand, Figure 9 depicts that using the multi-watermark approach, with same performance loss as in the case of using only the single-watermark, the detection ratio increases when the switching frequency varies in the range $[0, 0.14]$Hz, where $f = 0$ is the single-watermark detector. In the following section, we extend the analysis to the case of parametric cyber-physical adversaries.

4.6 Numerical Validation of the Multi-Watermark Detector against Parametric Cyber-physical Adversaries

In the previous sections, we have seen how the multi-watermark detector is able to detect both cyber and non-parametric cyber-physical adversaries. In this section, we extend the study to the case of parametric cyber-physical adversaries (cf. Definition 3.6). We recall that parametric cyber-physical adversaries are able to identify the system model parameters from the input and the output plant signals. In fact, a parametric cyber-physical adversary can obtain the system model with great accuracy, if control commands and sensor measurements are accessible.

Let us first show how a parametric cyber-physical adversary acquires the watermark signal presented in [4, 5]. Remember that such a watermark is modeled as a Gaussian signal with zero-mean. Its variance is represented by $\mathcal{U}$, *i.e.* $\Delta u_t \sim N(0, \mathcal{U})$. The variance modifies the control inputs and propagates the modification to the system outputs. However, it does not modify the system dynamics. Control inputs are represented by:

$$U(z) = U^*(z) + \Delta U(z) \tag{21}$$

and the system outputs are represented by:

$$\begin{aligned} Y(z) &= H(z)(U(z) + W(z)) + V(z) \\ &= H(z)(U^*(z) + \Delta U(z) + W(z)) + V(z) \end{aligned} \tag{22}$$

Using the aforementioned characteristic of the watermark, a parametric cyber-physical adversary can use an ARX (autoregressive with exogenous input) model to define the system as follows [22]:

$$Y(z) = H(z)U(z) + V(z) \tag{23}$$

where $U(z)$ and $Y(z)$ represent the inputs and the outputs of the plant, respectively; $V(z)$ represents the external noise which affects the outputs of the plant; and $H(z)$ is another way to describe the model of the system presented in Section 3, such that:

$$H(z) = \frac{\mathcal{N}(z)}{\mathcal{D}(z)} = \left(\frac{n_0 z^m + n_1 z^{m-1} + ... + n_m}{d_0 z^n + d_1 z^{n-1} + ... + d_n} \right) \tag{24}$$

where $\mathcal{N}(z)$ and $\mathcal{D}(z)$ are the polynomial functions which build the model of the system.

Following the same simulation setup introduced in Section 4.4, Figures 10 and 11 show the detection ratio of the watermark detector against a parametric cyber-physical adversary. Figure 10 shows the results of 200 Monte Carlo simulations using systems of order 10, against this adversary. The results present the detection ratio if the adversary uses a window size equal to 200 and different system orders for the model. If the adversary order varies in the range $[8, 12]$, the detection ratio is not higher than 10%. Out of this range, the detection ratio increases drastically. Figure 11 shows the detection ratio using systems of order 25, against seven different parametric cyber-physical adversaries. The assumed window size is settled to $\hat{T} = 300$. The range of orders where the detection ratio does not increase drastically is $[18, 28]$. If an adversary uses an order in this range, the detection ratio is not higher than 10%. Otherwise, the likelihood to detect the adversary is high.

Figure 12 shows the detection ratio of the same system, against a parametric cyber-physical adversary with different window sizes (125, 150, 200, 250, and 300), and the correct system order. It is worth noting that the adversary needs a bigger window size in order to attack a system using a higher order, with a detection ratio less than 10%. Following the previous results, we can conclude that a parametric cyber-physical adversary, who is capable to eavesdrop and analyze a large number of samples from the communication channel, and using an equivalent order system, is capable of evading detection.

Remark 4.3 A parametric cyber-physical adversary is able to obtain the system model, $H(z)$, and mislead the controller, by eavesdropping the control inputs and the sensor measurements. The probability of being detected is equivalent to the probability of obtaining an erroneous model. This probability is directly proportional to the order of the system; and inversely proportional to the window size to eavesdrop the data channel.

Following the Remark 4.3, and under the hypothesis of considering the real system like a black box, erroneous system identification depends on the order selected by the adversary to recreate the system model, as well as the number of eavesdropped samples and the window size used by the adversary to recompute the parameters of the target system. This situation can be quantified using Mean Square Error (MSE) [26, 27]. In a nutshell, the likelihood to obtain the correct model of the target system is directly proportional to the order chosen by the adversary to generate the model, and inversely proportional to the number of samples eavesdropped (cf. Figures 10, 11, and 12). The computational cost for the adversary is directly proportional to the system order, since the adversary needs to increase the order of the model, as well as the window size in order to minimize the MSE. Therefore, the number of samples eavesdropped before conducting the attack, together with the order system chosen by the adversary, are the two main parameters to evade detection.

5 Experimental Results with a Laboratory SCADA Testbed

In this section, we present some experimental results obtained with a real-world implementation of the detection mechanisms and adversary models presented in this paper. The implementation is conducted using a Laboratory testbed based on SCADA protocols such as Modbus and DNP3 (cf. Section 2.1).

5.1 Testbed Design

The architecture proposed for our SCADA testbed works as follows. All the elements (controller, sensors, actuators, PLCs, RTUs) can be distributed across several nodes in a shared network combining DNP3 and Modbus protocols (cf. Figure 13). Likewise, one or various elements can be embedded into a single device. From a software standpoint, the controller never connects directly to the sensors. Instead, it is integrated in the architecture as a SCADA PLC node, with eventual connections to some other intermediary nodes. Such nodes are able to translate the controller commands into SCADA (e.g., either Modbus or DNP3) commands. As depicted in Figure 13, the architecture is able to handle several industrial protocols and connect to complementary SCADA elements, such as additional PLCs and RTUs. To evolve the architecture into a complete testbed, new elements can be included in the system, such as additional proxy-like RTU nodes.

From a data transmission standpoint, we include in our SCADA testbed the possibility of using different sampling frequencies, in order to cover a larger number of experimental scenarios. The architecture is able to handle several PLCs. To avoid overloading one channel with all the possible registers of the PLCs, separate ports are designated in order to isolate the communication between separated PLCs. DNP3 commands perform an Integrity Scan which gathers all the data from the PLCs in case several PLCs were being handled in the same channel, all variables of the a PLC would be fetched causing overhead in the communication.

5.2 Experimentation

We present in this section the results of applying the watermark authentication techniques proposed in this paper. Several repetitions of the experiments are orchestrated using automated scripts handling the elements of some representative scenarios. A set of attacks and detectors are used and posteriorly analyzed. The combinations, attack–detector, are the following:

- *Replay Attack–Watermark Disabled:* In this scenario, the attacker is likely to evade the detector, since no watermark is injected into the system.
- *Replay Attack–Watermark Enabled:* In this scenario, the attacker is likely identified by the detector, since the attack is not able to adapt to the current watermark.
- *Non-parametric Attack–Stationary Watermark:* Using this scenario, the attacker and the detector have equal chances of success.
- *Non-parametric Attack–Non-stationary Watermark:* Using this scenario, the non-stationary watermark changes the distribution systematically, hence preventing the attack to adapt to such changes. The expected results are an increase of the detection ratio.
- *Parametric Attack–Stationary Watermark:* In this scenario, the attacker is likely to evade the detector when the attack properly infers the system parameters.
- *Parametric Attack–Non-stationary Watermark:* The attacker is also likely to evade the detector when the system parameters are properly identified.

The cyber-physical implications of the testbed hinder the experimentation process especially when several repetitions are required in order to obtain statistical results, contrary to simulations where only the code is executed. The creation of the orchestration script, which automates the test, is necessary to simplify the experimentation tasks. The next section presents the results using the testbed for the aforementioned attacker-detector combinations. Some sample executions of the *Replay Attack – Watermark Disabled* scenario, as well as information about the implemented techniques, is available at http://j.mp/TSPScada.

5.3 Experimention and Results

After collecting data from different devices across the SCADA testbed, the data is analyzed accordingly to interpret the performance of the detector with regard to the attack scenario. Since the stationary watermark detector was correctly refined for each test scenario, we are able to analyze in depth the results through a statistical evaluation of the data. Experimental results with the non-stationary watermark mechanism are also conducted. Figure 14 shows the detector values, g_t, for all the attack-detector combinations defined in Section 5.2.

For all the plots, the solid horizontal line represents the threshold; and the vertical dotted line represents the moment when the attacker starts injecting malicious data. The short peaks on the left side of the plots, those bypassing the threshold line before the start of the attacks, are counted as false positives or system faults.

Figures 14(a) and 14(b) show the experimental results of the replay attack. When the watermark is disabled (cf. Figure 14(a)), the attacker properly evades the detector. Since the controller is not inserting the protection watermark, it does not detect the attack. On the contrary, the results in Figure 14(b) show that the activation of the watermark under the same scenario allows the controller to alert about the attack almost immediately. Based on these results, we can conclude that the stationary watermark based detector properly works out to detect the replay attack.

Figure 14(c) represents the non-parametric attacker against the previously tested stationary watermark. The detector is now unable to detect the attacker. Figure 14(d) shows the case where the non-stationary watermark is enabled. Under this situation, the detector has lightly more chances of detecting the attack. This shows how the non-stationary watermark mechanism does improve the detection abilities compared to the stationary watermark approach.

Figures 14(e) and 14(f) evaluate the scenario associated to the parametric attacks. Theoretically, the attacker is expected to evade the detector when the attack succeeds at properly identifying the parameters of the system dynamics. Figure 14(e) represents the experiments where the parametric attack is executed under the stationary watermark scenario. The figure shows that the detector value, g_t, remains most of the time below the detection threshold. Figure 14(f) shows the behavior of the detector under the non-stationary watermark scenario. This time, the detector has slightly more chances of detecting the attack.

5.4 Statistical Data Evaluation

Using the watermark-based detection mechanism, we run for each attack scenario 75 automated rounds (about 4 hours of data collection processing). In order to evaluate the results, we use the following metrics:

1 *Detection Ratio*, associated to the success percentage of the detector, calculated with regard to the time range after each attack starts.

2 *Average Detection Time*, determining the amount of time needed by the detector to trigger the attack alert.

3 *False Negative (FN)* ratio, determining the number of samples where the detector fails at successfully alerting about the attacks. The ratio is calculated as follows,

$$FN = \frac{SA - AD}{SA} \tag{25}$$

where *SA* represents the values of the samples under attack, and *AD* the samples detected as an attack.

4 *False Positives (FP)* ratio, calculated as the number of samples where the detector signals benign events as attacks. The ratio is calculated as follows,

$$FP = \frac{AD}{SN} \tag{26}$$

where *SN* represents the number of samples under normal operation, and *AD* the number of samples detected as attack by mistake.

Table 3 shows the performance results of the detector, based on the *Detection Ratio* and the *Average detection Time* metrics.

Regarding the results shown in Table 3, we can emphasize that the replay attack is the most detectable scenario, with a detection ratio of about 40%. This detection ratio is still far from being perfect, which we ascribe to imperfect sensors accuracy and resolution. The non-parametric attacker has a lower detection ratio, of about 18%. This result is expected, as suggested by the theoretical and simulation-based conclusions presented in Sections 3 and 4. The parametric attack has the most robust system identification approach. The attacks can evade the detection process if they succeed at properly identifying the system attributes. In terms of results, they lead to the lowest detection rate of about 12%.

During the replay attack, the *Average Detection Time* is the slowest of all the adversarial scenarios. This behavior is due to the watermark distribution properties, since the watermark variation makes the replay attack highly detectable. At the same time, the injection attacks (either the parametric or the non-parametric version) are detected much faster than the replay attack. This is due to the transition period needed by the attackers to estimate the correct data prior misleading the detector. For this reason, if the attacker does not choose the precise moment to start the attack, the detector implemented at the controller side is able to detect the injected data, right at the beginning of the attack. Furthermore, the attackers shall also synchronize their estimations to the measurements sent by the sensors. In the case the synchronization process fails, the detector identifies the uncorrelated data and reports the attack.

Table 4 shows that the detection of the replay attack has the lowest false negative ratio, 64.06%, hence confirming that this adversarial scenario is the most detectable situation with regard to the detection techniques reported in [4, 5]. The detection of the non-parametric attacks has a higher false negative ratio, 85.20%, confirming the theoretical and simulation-based results reported in Section 4. The detection of the parametric attacks also confirms the results obtained via numeric simulations, leading to the highest false negative ratio (about 88.63%). In terms of false positive ratio, the three adversarial scenarios show a very low impact of our detection approach (on average, about 1.33% false positive ratio).

6 Discussion

In Section 3, we have reviewed the watermark-based detector proposed in [4, 5]. We have shown that the detector fails at properly handling attacks carried out by *cyber-physical* adversaries. In particular, we have shown that an adversary that learns about the system model is able to separate the watermark from the control signal and succeeds at attacking the system without being detected. Then, we have presented an enhanced detection scheme. The main idea of the new scheme, is to use multiple watermark distributions and non-stationary identification signals. The resulting approach is able to detect both cyber- and cyber-physical adversaries.

To summarize, the detector in [4, 5] fails at detecting cyber-physical adversaries. The multi-watermark proposal succeeds at properly detecting such adversaries under the assumption that the watermark distributions change quite frequently. The rationale is that, the non-parametric adversary has little chances of acquiring the necessary information to acquire the watermark and bypass the detector. Moreover, the detector performance loss in the multi-watermark approach is equivalent to the performance loss in the case of the single-watermark approach. We have also shown that a smarter parametric cyber-physical adversary is able to attack the system without being detected in case of detecting the correct parameters. We have detailed the strategy of this adversary in Section 4. It is worth noting that the detection ratio increases with respect to the lack of accuracy of the adversary.

7 Related Work

Security of cyber-physical systems (CPSs) is drawing a great deal of attention recently [1] after the infamous StuxNet malware [2] uncovered the potential of successful security attacks carried out against such systems.

In the following, we separately overview some related work in the literature of CPS security by focusing on three main research areas: (1) security requirements and threat analysis; (2) control-theoretic solutions for the detection of cyber-physical attacks; and (3) implementation of experimental testbeds.

Security requirements and threat analysis on CPSs – Several authors have studied the requirements to take into account the new security issues when designing security mechanisms for cyber-physical systems. In [28], Cardenas *et al.* define the issue of secure control by analyzing separately the problem first from a information security point of view and then by looking at specific control issues. In [29], the same authors outline for the first time the difference between corporate ICT security and cyber-physical system security, and the importance of explicitly considering the interactions of the control system with the plant. In [30], Wang *et al.* provide a classification of threats and possible attacks against CPS. They also introduce a taxonomy of adversaries by classifying them into several categories depending on skills and motivation. A different perspective is taken in [15] where Texeira *et al.* propose to classify attacks according to a three-dimensional *cyber-physical attack space*: disruption of assets, disclosure of resources, and adversary knowledge. Cyber attacks specific to SCADA systems are addressed in [31, 32]. In [31], Ten *et al.* carry out a quantitative assessment of vulnerability of SCADA systems. Zhu *et al.* analyze in [32] some attacks towards SCADA systems by making a distinction between hardware, software, and communication stack threats. In a complementary research direction, the issue of designing testbeds to experimentally evaluate both threats (and their countermeasures) has been carried out by Hahn *et al.* and Mallouhi *et al.* in [33, 34].

Control-theoretic solutions for the detection of cyber-physical attacks – This research line explicitly considers the interconnection between cyber and physical control domains in networked control systems. Recently, the control system community started to study security of cyber-physical systems both under the methodological point of view and from a more technological standpoint by looking at particular problems arising in, e.g., smart grids, power grids, water distribution systems. Concerning the methodological aspects, several studies have proposed to adapt classical frameworks (developed to cope with model uncertainties and fault diagnosis) to handle security issues in networked control systems. Table 5 gathers the cyber-physical attacks handled in the literature. Among these cyber-physical attacks, the replay attack is the only one in which the adversary is able to carry out actions without knowledge about system model dynamics. To carry out the rest of the attacks, it is necessary some system knowledge. For example, to execute the dynamic false-data injection attack handled by Mo *et al.* in [35], the adversary has to have a perfect knowledge of the plant's behaviour. To execute a covert attack, handled by Smith *et al.* [16], it is necessary knowledge of the plant's and controller's behaviour. Concerning detection mechanisms, several lines of research consider the adaptation of fault detection systems to detect intentional attacks [5, 15, 36]. Mo *et al.* show in [5] that it is possible to detect replay attacks by properly watermarking control inputs. Pasqualetti *et al.* propose in

[36] to employ geometric control theory to model undetectable and unidentifiable attacks. Those attacks are equivalent to the aforementioned *covert attack* by Smith *et al.*, where the adversary knows perfectly the system model dynamics. The parametric cyber-physical adversary defined in our work deals with equivalent scenarios.

Implementation of experimental testbeds – Research on cyber-physical systems has progressed substantially resulting in a large number of experimental testbeds developed and established in the literature. Myat-Aung present in [37] a Secure Water Treatment (SWaT) simulation and testbed to test defense mechanisms against a variety of attacks. Siaterlis *et al.* [38] define a cyber-physical Experimentation Platform for Internet Contingencies (EPIC) that is able to study multiple independent infrastructures and to provide information about the propagation of faults and disruptions. Green *et al.* [39] focus their work on an adaptive cyber-physical testbed where they include different equipments, diverse networks, and also business processes. Yardley reports in [40] a cyber-physical testbed based on commercial tools in order to experimentally validate emerging research and technologies. The testbed combines emulation, simulation, and real hardware to experiment with smart grid technologies. Sanchez Arago *et al.* present in [41] a framework able to assess remotely the security of these systems. Krotofil and Larsen show in [42] several testbeds and simulations concluding that a successful attack against their envisioned systems has to manage cyber and physical knowledge.

From a more control-theoretic standpoint, Candell *et al.* report in [43] a testbed to analyze the performance of security mechanisms for cyber-physical systems. The work reports as well discussions from control and security practitioners. McLaughlin *et al.* analyze in [44] different testbeds and conclude that it is necessary to use pathways between cyber and physical components of the system in order to detect attacks. Also, Koutsandria *et al.* [45] implement a testbed where the data are cross-checked, using cyber and physical elements. Holm *et al.* survey, classify and analyze in [46] several cyber-physical testbeds proposed for scientific research. Inline with the aforementioned contributions, we have presented in this paper a testbed that aims at evaluating mitigation techniques targeting attacks at the physical layer of cyber-physical systems operated via SCADA protocols. The initial focus of our testbed has been the evaluation of the control-theoretic security mechanisms reported analyzed previously in the paper.

8 Conclusion

We have addressed security issues in cyber-physical systems. We have focused on the adaptation of failure detection mechanisms. The goal is to handle, in addition to faults and errors, the detection of attacks against industrial environments that close their loops through networked control systems, more specifically cyber-physical attacks, i.e., attacks against cyber and physical layer of these systems.

We have reviewed the watermark-based detector proposed in [4, 5]. We have shown that the detector fails at properly handling attacks carried out by cyber-physical adversaries. In particular, we have shown that an adversary that is able to acquire knowledge about the system model is able to succeed at attacking the system without being detected. Then, we have presented an enhanced detection scheme. The main idea of the new scheme is to use multiple watermark distributions and non-stationary identification signals. The resulting approach is able to detect both cyber and non-parametric cyber-physical adversaries. We have also shown that the new watermark-based detector works against a parametric cyber-physical adversary who knows only a set of control inputs. Nevertheless, if the adversary knows all the control inputs and sensor measurements of the system, and uses the correct orders range with a window size sufficiently long, the watermark-based detector fails.

Acknowledgements

The authors declare that they have no competing interests. The authors acknowledge support from the Cyber CNI Chair of Institut Mines-Télécom (cf. http://chaire-cyber-cni.fr/). The chair is supported by Airbus Defence and Space, Amossys, EDF, Orange, La Poste, Nokia, Société Générale and the Regional Council of Brittany. It has been acknowledged by the Center of excellence in Cybersecurity.

Authors' Contribution

All authors contributed equally to the manuscript. In addition, JRH carried out the numeric simulations and laboratory experiments, and drafted the initial version of the manuscript.

Author details

[1] SAMOVAR, Telecom SudParis, CNRS, Université Paris-Saclay, 9 Rue Charles Fourier, 91000 Evry, France. [2] Dipartimento di Ingegneria Elettrica e dell'Informazione, Politecnico di Bari, Via Orabona n.4, 70125 Bari, Italy.

References

1. Corman, D., Pillitteri, V., Tousley, S., Tehranipoor, M., Lindqvist, U.: NITRD Cyber-Physical Security Panel. 35th IEEE Symposium on Security and Privacy, IEEE S&P 2014, San Jose, CA, USA, May 18-21
2. Falliere, N., Murchu, L.O., Chien, E.: W32. stuxnet dossier. White paper, Symantec Corp., Security Response **5**, 6 (2011)
3. Rubio-Hernan, J., De Cicco, L., Garcia-Alfaro, J.: Revisiting a Watermark-based Detection Scheme to Handle Cyber-Physical Attacks. In: 11th International Conference on Availability, Reliability and Security. IEEE, Salzburg, Austria (2016)
4. Mo, Y., Sinopoli, B.: Secure control against replay attacks. In: Communication, Control, and Computing. 47th Annual Allerton Conference On, pp. 911–918 (2009). doi:10.1109/ALLERTON.2009.5394956. IEEE
5. Mo, Y., Weerakkody, S., Sinopoli, B.: Physical Authentication of Control Systems: Designing Watermarked Control Inputs to Detect Counterfeit Sensor Outputs. IEEE Control Systems **35**(1), 93–109 (2015). doi:10.1109/MCS.2014.2364724
6. Rubio-Hernan, J., De Cicco, L., Garcia-Alfaro, J.: Event-Triggered Watermarking Control to Handle Cyber-Physical Integrity Attacks. In: Secure IT Systems: 21st Nordic Conference, NordSec 2016, Oulu, Finland, November 2-4, 2016. Proceedings, pp. 3–19. Springer, Cham (2016). doi:10.1007/978-3-319-47560-8_1
7. Brown, S.: Functional safety of electrical/electronic/programmable electronic safety related systems. Computing & Control Engineering Journal **11**(11), 14 (2000)
8. Åkerberg, J., Björkman, M.: Exploring Network Security in PROFIsafe. In: Computer Safety, Reliability, and Security: 28th International Conference, SAFECOMP 2009, Hamburg, Germany, September 15-18, 2009. Proceedings, pp. 67–80. Springer, Berlin, Heidelberg (2009). doi:10.1007/978-3-642-04468-7_7
9. PROFIBUS and PROFINET International: International Standard, PROFINET Security Guideline (2013, http://www.profibus.com/download/specifications-standards/, Last access: October 2016)
10. International Electrotechnical Commission: Industrial communication networks - Fieldbus specifications - Part 6-2: Application layer protocol specification - Type 2 elements (2014, https://webstore.iec.ch/publication/4695, Last access: Octobre 2016)
11. Rinaldi, J.: ETHERNET/IP OVERVIEW (2014, http://www.rtaautomation.com/technologies/ethernetip/, Last access: Octobre 2016)
12. Knapp, E.D.: Industrial Network Security: Securing Critical Infrastructure Networks for Smart Grid, SCADA, and Other Industrial Control Systems, 1st edn. Syngress Publishing, Boston, MA, USA (2011)
13. Pasqualetti, F.: Secure control systems: A control-theoretic approach to cyber-physical security. PhD thesis, Department of Mechanical Engineering, University of California, Santa Barbara (September 2012)
14. Zhu, Q., Başar, T.: A hierarchical security architecture for smart grid, pp. 413–438. Cambridge University Press, Cambridge (2012). doi:10.1017/CBO9781139013468.019. Cambridge Books Online
15. Teixeira, A., Shames, I., Sandberg, H., Johansson, K.H.: A secure control framework for resource-limited adversaries. Automatica **51**, 135–148 (2015). doi:10.1016/j.automatica.2014.10.067
16. Smith, R.S.: Covert Misappropriation of Networked Control Systems: Presenting a Feedback Structure. IEEE Control Systems **35**(1), 82–92 (2015). doi:10.1109/MCS.2014.2364723
17. Arvani, A., Rao, V.S.: Detection and protection against intrusions on smart grid systems. International Journal of Cyber-Security and Digital Forensics (IJCSDF) **3**(1), 38–48 (2014)
18. Do, V.L., Fillatre, L., Nikiforov, I.: A statistical method for detecting cyber/physical attacks on SCADA systems. In: 2014 IEEE Conference on Control Applications (CCA), pp. 364–369 (2014). doi:10.1109/CCA.2014.6981373
19. Franklin, G.F., Powell, J.D., Workman, M.L.: Digital Control of Dynamic Systems, 3rd edn. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA (1998)
20. Brumback, B., Srinath, M.: A chi-square test for fault-detection in Kalman filters. IEEE Transactions on Automatic Control **32**(6), 552–554 (1987). doi:10.1109/TAC.1987.1104658
21. Tripathi, S., Ikbal, M.A.: Step Size Optimization of LMS Algorithm Using Aunt Colony Optimization & Its comparison with Particle Swarm optimization Algorithm in System Identification. International Research Journal of Engineering and Technology (IRJET) **2**, 599–605 (2015)

22. Natke, H.: System identification: Torsten Söderström and Petre Stoica. Automatica **28**(5), 1069–1071 (1992)
23. Widrow, B., McCool, J.M., Larimore, M.G., Johnson Jr, C.R.: Stationary and nonstationary learning characteristics of the LMS adaptive filter. Proceedings of the IEEE **64**(8), 1151–1162 (1976). doi:10.1109/PROC.1976.10286
24. Ricker, N.L.: Model predictive control of a continuous, nonlinear, two-phase reactor. Journal of Process Control **3**(2), 109–123 (1993). doi:10.1016/0959-1524(93)80006-W
25. Chabukswar, R., Mo, Y., Sinopoli, B.: Detecting Integrity Attacks on SCADA Systems. {IFAC} Proceedings Volumes **44**(1), 11239–11244 (2011). doi:10.3182/20110828-6-IT-1002.03712
26. Ljung, L.: Perspectives on system identification. Annual Reviews in Control **34**(1), 1–12 (2010). doi:10.1016/j.arcontrol.2009.12.001
27. Barenthin Syberg, M.: Complexity Issues, Validation and Input Design for Control in System Identification. PhD thesis, KTH School of Electrical Engineering, Stockholm, Sweden (2008)
28. Cardenas, A.A., Amin, S., Sastry, S.: Secure control: Towards survivable cyber-physical systems. In: The 28th International Conference on Distributed Computing Systems Workshops, pp. 495–500 (2008). doi:10.1109/ICDCS.Workshops.2008.40. IEEE
29. Cardenas, A.A., Amin, S., Sinopoli, B., Giani, A., Perrig, A., Sastry, S.: Challenges for securing cyber physical systems. In: Workshop on Future Directions in Cyber-Physical Systems Security, p. 7 (2009). DHS. http://chess.eecs.berkeley.edu/pubs/601.html
30. Wang, E.K., Ye, Y., Xu, X., Yiu, S., Hui, L., Chow, K.: Security issues and challenges for cyber physical system. In: Proceedings of the 2010 IEEE/ACM Int'L Conference on Green Computing and Communications & Int'L Conference on Cyber, Physical and Social Computing. GREENCOM-CPSCOM '10, pp. 733–738. IEEE Computer Society, Washington, DC, USA (2010). doi:10.1109/GreenCom-CPSCom.2010.36
31. Ten, C.-W., Liu, C.-C., Manimaran, G.: Vulnerability assessment of cybersecurity for scada systems. IEEE Transactions on Power Systems **23**(4), 1836–1846 (2008). doi:10.1109/TPWRS.2008.2002298
32. Zhu, B., Joseph, A., Sastry, S.: A Taxonomy of Cyber Attacks on SCADA Systems. In: Internet of Things (iThings/CPSCom), 2011 International Conference on and 4th International Conference on Cyber, Physical and Social Computing, pp. 380–388 (2011). doi:10.1109/iThings/CPSCom.2011.34. IEEE
33. Hahn, A., Ashok, A., Sridhar, S., Govindarasu, M.: Cyber-Physical Security Testbeds: Architecture, Application, and Evaluation for Smart Grid. IEEE Transactions on Smart Grid **4**(2), 847–855 (2013). doi:10.1109/TSG.2012.2226919
34. Mallouhi, M., Al-Nashif, Y., Cox, D., Chadaga, T., Hariri, S.: A testbed for analyzing security of SCADA control systems (TASSCS). In: Innovative Smart Grid Technologies (ISGT), 2011 IEEE PES, pp. 1–7 (2011). doi:10.1109/ISGT.2011.5759169
35. Mo, Y., Garone, E., Casavola, A., Sinopoli, B.: False data injection attacks against state estimation in wireless sensor networks. In: 49th IEEE Conference on Decision and Control (CDC), pp. 5967–5972 (2010). doi:10.1109/CDC.2010.5718158
36. Pasqualetti, F., Dorfler, F., Bullo, F.: Cyber-physical security via geometric control: Distributed monitoring and malicious attacks. In: 2012 IEEE 51st IEEE Conference on Decision and Control (CDC), pp. 3418–3425 (2012). doi:10.1109/CDC.2012.6426257
37. Kaung Myat, A.: Secure Water Treatment Testbed (SWaT): An Overview (2015, `https://itrust.sutd.edu.sg/wp-content/uploads/sites/3/2015/11/Brief-Introduction-to-SWaT_181115.pdf`, Last access: October 2016)
38. Siaterlis, C., Genge, B., Hohenadel, M.: EPIC: a testbed for scientifically rigorous cyber-physical security experimentation. IEEE Transactions on Emerging Topics in Computing **1**(2), 319–330 (2013). doi:10.1109/TETC.2013.2287188
39. Green, B., Hutchison, D., Frey, S.A.F., Rashid, A.: Testbed diversity as a fundamental principle for effective ICS security research. In: Proceedings of the First International Workshop on Security and Resilience of Cyber-Physical Infrastructures (SERECIN) (2016). Lancaster University, Technical Report SCC-2016-01, pp. 12–15
40. Yardley, T.: Testbed cross-cutting research (2014, `https://tcipg.org/research/testbed-cross-cutting-research`, Last access: October 2016)
41. Aragó, A.S., Martínez, E.R., Clares, S.S.: SCADA Laboratory and Test-bed As a Service for Critical Infrastructure Protection. In: Proceedings of the 2Nd International Symposium on ICS & SCADA Cyber Security Research 2014. ICS-CSR 2014, pp. 25–29. BCS, UK (2014). doi:10.14236/ewic/ics-csr2014.4. http://dx.doi.org/10.14236/ewic/ics-csr2014.4
42. Krotofil, M., Larsen, J.: Rocking the pocket book: Hacking chemical plants for competition and extortion. In: DefCon 23, Las Vegas, USA (2015)
43. Candell, R., Stouffer, K., Anand, D.: A cybersecurity testbed for industrial control systems. In: Process Control and Safety Symposium, International Society of Automation, Houston, TX (2014)
44. McLaughlin, S., Konstantinou, C., Wang, X., Davi, L., Sadeghi, A.-R., Maniatakos, M., Karri, R.: The Cybersecurity Landscape in Industrial Control Systems, vol. 104, pp. 1039–1057 (2016). doi:10.1109/JPROC.2015.2512235
45. Koutsandria, G., Gentz, R., Jamei, M., Scaglione, A., Peisert, S., McParland, C.: A real-time testbed environment for cyber-physical security on the power grid. In: 1st ACM Workshop on Cyber-Physical Systems-Security And/or Privacy, pp. 67–78 (2015). ACM
46. Holm, H., Karresand, M., Vidström, A., Westring, E.: A Survey of Industrial Control System Testbeds. In: Buchegger, S., Dam, M. (eds.) Secure IT Systems: 20th Nordic Conference, NordSec 2015, Stockholm, Sweden, October 19–21, 2015, Proceedings, pp. 11–26. Springer, Cham (2015). doi:10.1007/978-3-319-26502-5_2
47. Mo, Y., Chabukswar, R., Sinopoli, B.: Detecting integrity attacks on SCADA systems. IEEE Transactions on Control Systems Technology **22**(4), 1396–1407 (2014). doi:10.1109/TCST.2013.2280899
48. Teixeira, A., Pérez, D., Sandberg, H., Johansson, K.H.: Attack Models and Scenarios for Networked Control

Systems. In: Proceedings of the 1st International Conference on High Confidence Networked Systems. HiCoNS '12, pp. 55–64. ACM, New York, NY, USA (2012). doi:10.1145/2185505.2185515

49. Pasqualetti, F., Dorfler, F., Bullo, F.: Control-Theoretic Methods for Cyberphysical Security: Geometric Principles for Optimal Cross-Layer Resilient Control Systems. IEEE Control Systems **35**(1), 110–127 (2015). doi:10.1109/MCS.2014.2364725
50. Dán, G., Sandberg, H.: Stealth Attacks and Protection Schemes for State Estimators in Power Systems. In: Smart Grid Communications (SmartGridComm), 2010 First IEEE International Conference On, pp. 214–219 (2010). doi:10.1109/SMARTGRID.2010.5622046
51. Liu, Y., Ning, P., Reiter, M.K.: False data injection attacks against state estimation in electric power grids. ACM Transactions on Information and System Security (TISSEC) **14**(1), 13 (2011)
52. Bobba, R.B., Wang, K.M.R.Q., Khurana, H., Nahtstedt, K., Overbye, T.J.: Detecting False Data Injection Attacks on DC State Estimation. In: Proceeding of the 1st Workshop on Secure Control Systems (CPSWEEK), pp. 1–9. Citeseer, Stockholm, Switzerland (2010)
53. Cárdenas, A.A., Amin, S., Lin, Z.-S., Huang, Y.-L., Huang, C.-Y., Sastry, S.: Attacks against process control systems: risk assessment, detection, and response. In: Proceedings of the 6th ACM Symposium on Information, Computer and Communications Security. ASIACCS '11, pp. 355–366. ACM, New York, NY, USA (2011). doi:10.1145/1966913.1966959
54. A decoupled feedback structure for covertly appropriating networked control systems. {IFAC} Proceedings Volumes **44**(1), 90–95 (2011). doi:10.3182/20110828-6-IT-1002.01721. 18th {IFAC} World Congress

Figures and Tables

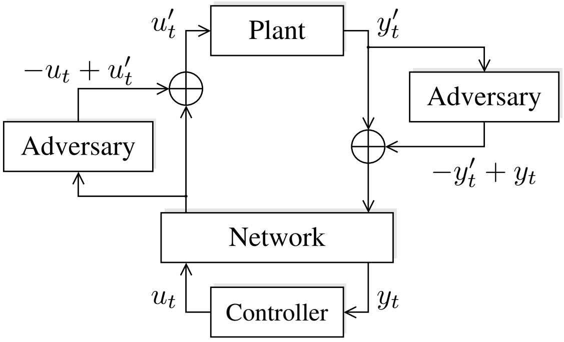


Figure 1 Sample representation of a cyber-physical attack against a networked control system.

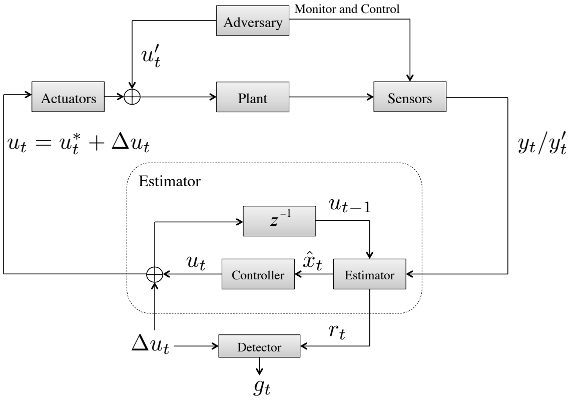
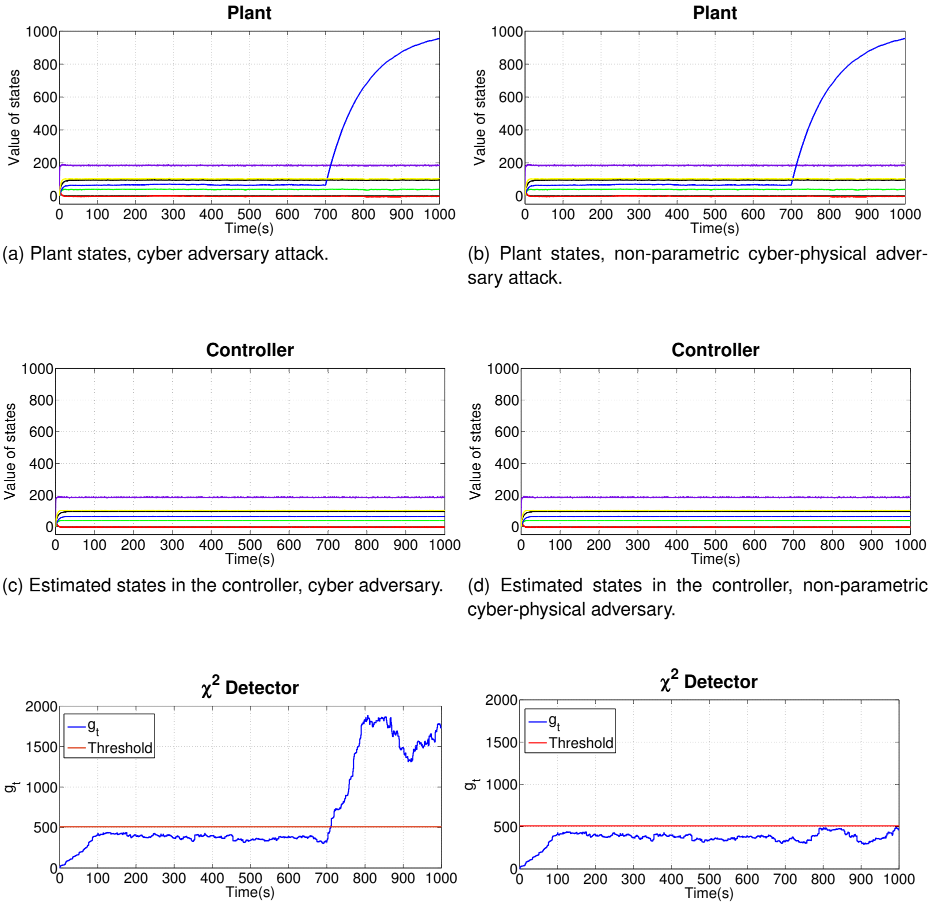


Figure 2 Overall control system equipped with the watermark-based detection scheme in [4, 5].



(a) Plant states, cyber adversary attack.

(b) Plant states, non-parametric cyber-physical adversary attack.

(c) Estimated states in the controller, cyber adversary.

(d) Estimated states in the controller, non-parametric cyber-physical adversary.

(e) Detector results, cyber adversary.

(f) Detector results, non-parametric cyber-physical adversary.

Figure 3 Numeric simulation results. Attacks start at $t = 700s$. (a)(c) The dynamics of the state vector in the plant and in the controller under the cyber adversary attack. (b),(d) The dynamics of the state vector in the plant and in the controller under the non-parametric cyber-physical adversary attack. (e),(f) χ^2 detector results under the two aforementioned attack scenarios.

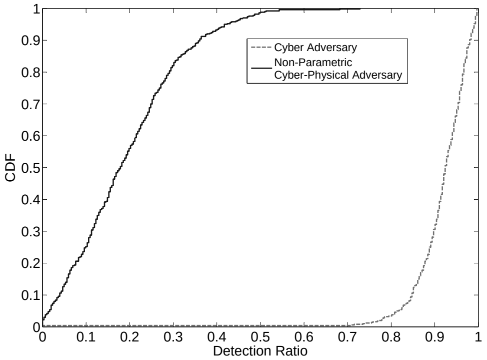
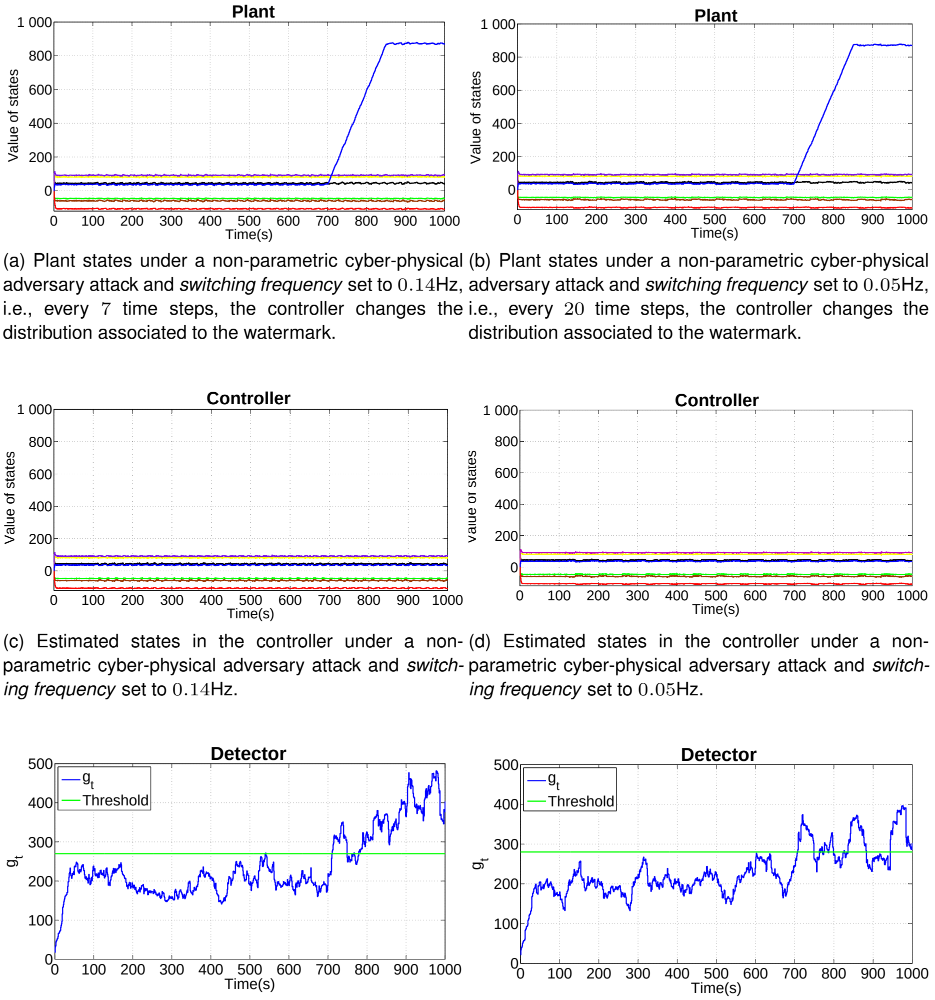


Figure 4 Cumulative distribution function (CDF) of the detection ratio associated to the χ^2 detector (see Section 3.2), obtained by measuring the DR metric (see Equation 9) for 200 simulations (both cyber and non-parametric cyber-physical adversary cases).



(a) Plant states under a non-parametric cyber-physical adversary attack and *switching frequency* set to 0.14Hz, i.e., every 7 time steps, the controller changes the distribution associated to the watermark.

(b) Plant states under a non-parametric cyber-physical adversary attack and *switching frequency* set to 0.05Hz, i.e., every 20 time steps, the controller changes the distribution associated to the watermark.

(c) Estimated states in the controller under a non-parametric cyber-physical adversary attack and *switching frequency* set to 0.14Hz.

(d) Estimated states in the controller under a non-parametric cyber-physical adversary attack and *switching frequency* set to 0.05Hz.

(e) Detector results, *switching frequency* set to 0.14Hz.

(f) Detector results, *switching frequency* set to 0.05Hz.

Figure 5 Numeric simulation results. Attacks start at $t = 700s$. (a),(b) The dynamics of the states vector in the plant under a non-parametric cyber-physical adversary attack and switching frequency configured with two different configurations (0.14Hz and 0.05Hz). (c),(d) The dynamics of the states vector estimated in the controller, under the same scenarios. (e),(f) The dynamics of the alarm signal g_t produced by the multi-watermark based detector, under the same scenarios.

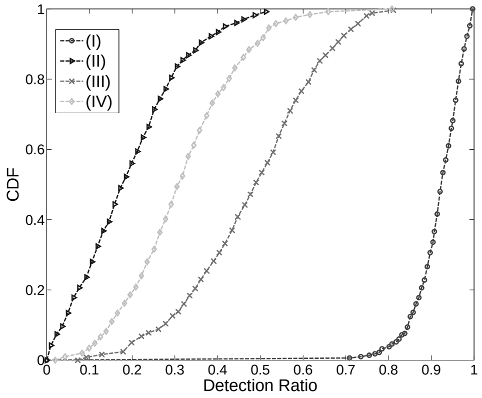
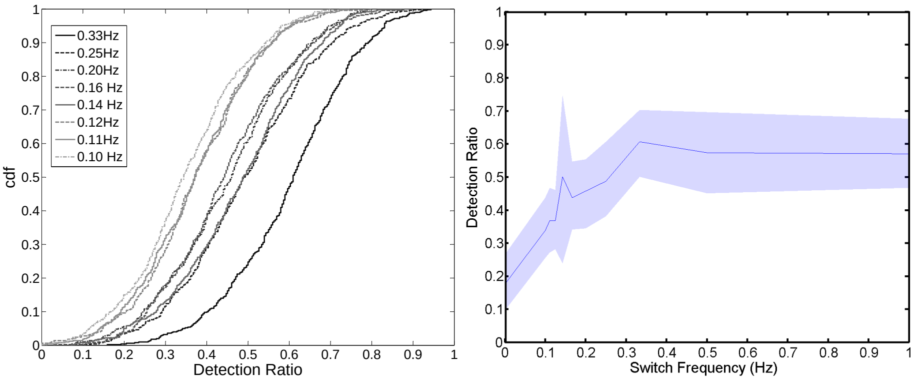


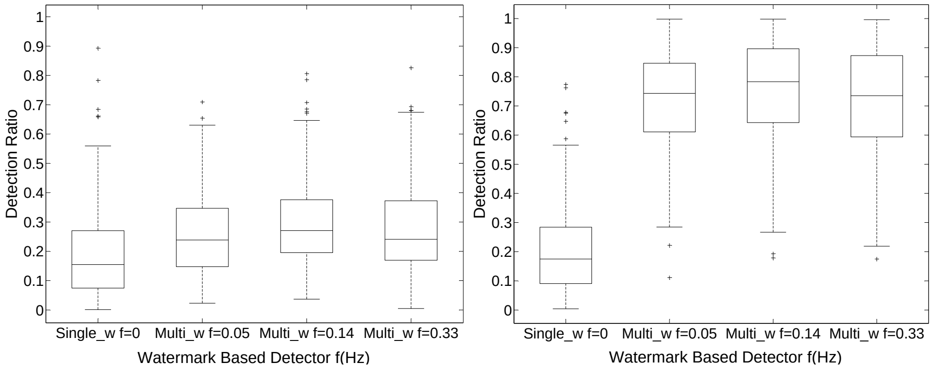
Figure 6 Numeric simulation results using the single and multi-watermark detection scheme. Confronting the performance of the two detectors. (I) χ^2 detector in [4, 5], and cyber adversary. (II) χ^2 detector and non-parametric cyber-physical adversary. (III) Multi-watermark detector with switching frequency set to 0.14Hz, and non-parametric cyber-physical adversary. (IV) Multi-watermark detector with switching frequency set to 0.05Hz, and non-parametric cyber-physical adversary.



(a) CDF and detection ratio per switching frequencies.

(b) Detection ratio function with respect to the switching frequency.

Figure 7 Numeric simulation results using the multi-watermark detection scheme. (a) Cumulative distribution function (CDF) of the detector under different switching frequencies. (b) Median detection ratio function per switching frequency, in which the shaded area corresponds to the 25-th and the 75-th percentiles (i.e., confidence intervals).



(a) Detection ratio function with respect to the single- and multi-watermark with different switching frequencies for fourth order systems.

(b) Detection ratio function with respect to the single- and multi-watermark with different switching frequencies for systems of order 25.

Figure 8 Numeric simulation results using the single and multi-watermark detection scheme with the same performance loss. (a) Detection ratio for a system of order four, using single- and multi-watermark. (b) Detection ratio for a system of order 25, using single- and multi-watermark.

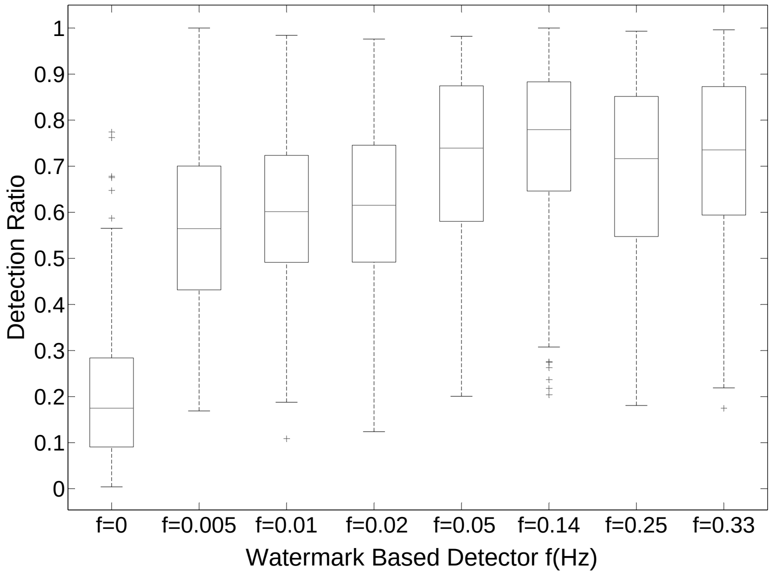
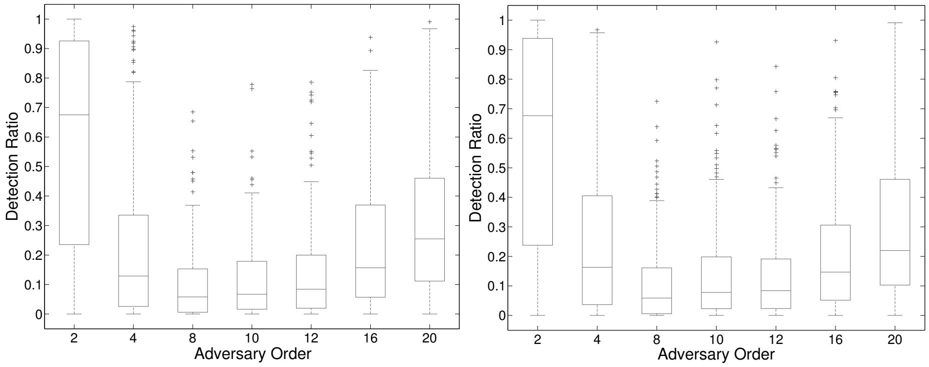


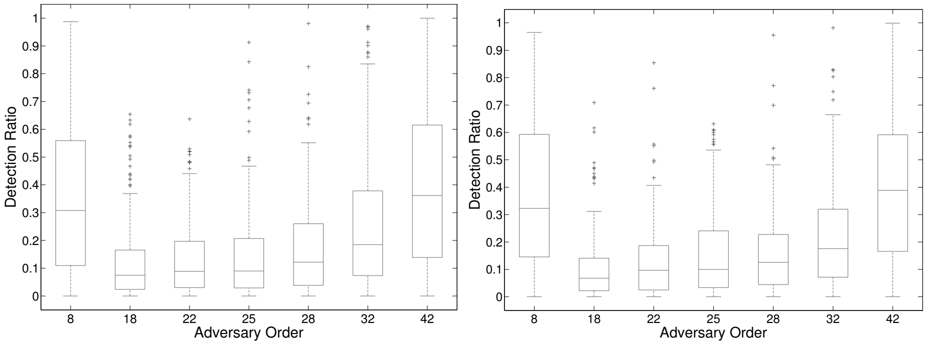
Figure 9 Single $(f = 0)$ and multi $(f = [0.005, 0.01, 0.02, 0.05, 0.14, 0.25, 0.33])$ watermark detector against a non-parametric cyber-physical adversary (the order of the system is 25).



(a) Detection ratio function with respect to the single-watermark against a parametric cyber-physical adversary using different adversary system order.

(b) Detection ratio function with respect to the multi-watermark against a parametric cyber-physical adversary using different adversary system order.

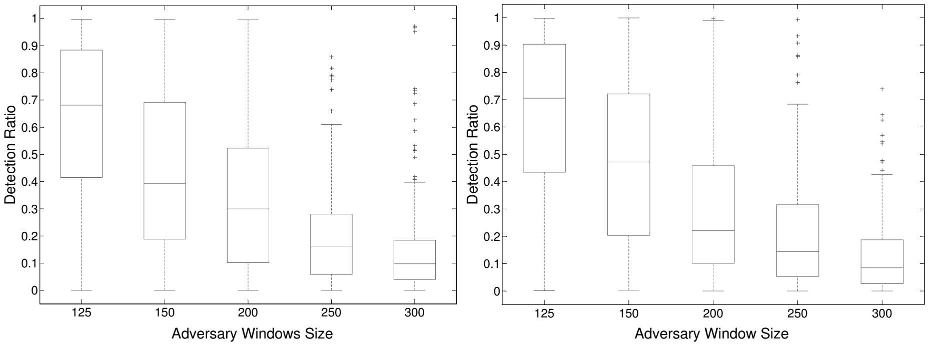
Figure 10 Numeric simulation results using the single and multi-watermark detection scheme with the same performance loss, different adversary system order, and a window size equal to 200. (a) Detection ratio regarding single-watermark for systems of order ten. (b) Detection ratio regarding multi-watermark for systems of order ten.



(a) Detection ratio function with respect to the single-watermark for systems of order 25, against a parametric cyber-physical adversary using different adversary system order.

(b) Detection ratio function with respect to the multi-watermark for systems of order 25, against a parametric cyber-physical adversary using different adversary system order.

Figure 11 Numeric simulation results using the single and multi-watermark detection scheme with the same performance loss, different adversary system order, and a window size equal to 300. (a) Detection ratio regarding single-watermark for systems of order 25. (b) Detection ratio regarding multi-watermark for systems of order 25.



(a) Detection ratio function with respect to the single-watermark against a parametric cyber-physical adversary using different adversary window size for eavesdropping the data of the channel before the attack.

(b) Detection ratio function with respect to the multi-watermark against a parametric cyber-physical adversary using different adversary window size for eavesdropping the data of the channel before the attack.

Figure 12 Numeric simulation results using the single and multi-watermark detection scheme with the same performance loss, and different adversary window size for eavesdropping the data channel before the attack. The order used by the adversary is the correct system order, $p = 25$ (a) Detection ratio for the single-watermark approach. (b) Detection ratio for the multi-watermark approach.

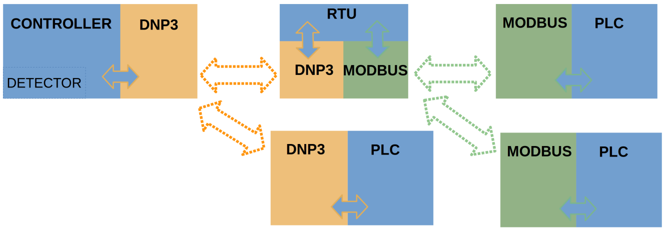
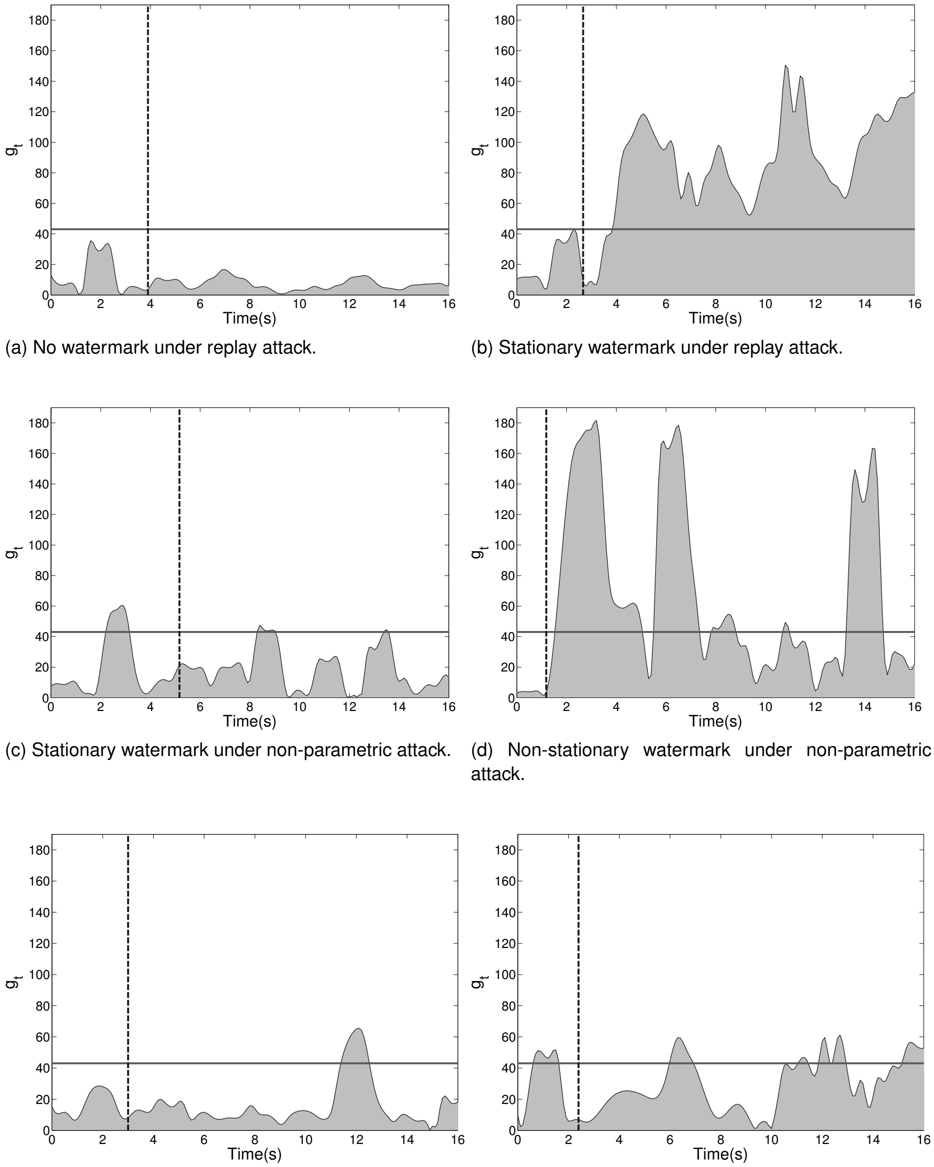


Figure 13 Abstract architecture overview.



(a) No watermark under replay attack.

(b) Stationary watermark under replay attack.

(c) Stationary watermark under non-parametric attack.

(d) Non-stationary watermark under non-parametric attack.

(e) Stationary watermark under parametric attack.

(f) Non-stationary watermark under parametric attack.

Figure 14 Experimental testbed results. The horizontal solid line represents the threshold. The vertical dotted line represents the moment when the attack starts. Peaks on the left side of the vertical dotted line represent false positives. (a),(b) detection values of g_t, without and with stationary watermark under replay attack. (c),(d) detection values with stationary and non-stationary watermark under non-parametric attack. And (e),(f) detection values with stationary and non-stationary watermark under parametric attack.

Table 1 Notations used in this paper.

A	:	State matrix.
B	:	Input matrix.
C	:	Output matrix.
$W(z)$	:	Process noise.
Q	:	Process noise variance.
$V(z)$	:	Output noise.
R	:	Output noise variance.
X or $X(z)$	:	Vector of state variables.
U or $U(z)$	:	Control input vector.
Y or $Y(z)$	:	Vector of the sensors measurements.
U^* or $U^*(z)$	:	Optimal control input vector.
ΔU or $\Delta U(z)$	:	Watermark.
$\Delta U(z)^{(i)}$	:	Multi-watermark.
$Y^{\Delta U}(z)$	:	Output due to the watermark.
Y' or $Y'(z)$	:	Measurements injected by the adversary.
U' or $U'(z)$	:	Control inputs injected by the adversary.
$\hat{X}$ or $\hat{X}(z)$	:	Vector of estimated state variables.
$\hat{X}^{(-)}$ or $\hat{X}^{(-)}(z)$	:	Vector of estimated state variables before applying the rectification.
$\hat{X}^{(+)}$ or $\hat{X}^{(+)}(z)$	:	Vector of estimated state variables after applying the rectification.
K_f	:	Kalman gain.
$P^{(-)}$	:	A priori error covariance.
$P^{(+)}$	:	A posteriori error covariance.
L	:	Feedback grain.
S	:	Riccati equation solution.
J	:	Quadratic cost.
ΔJ_s	:	Increment of quadratic cost due to the single-watermark.
ΔJ_m	:	Increment of quadratic cost due to the multi-watermark.
$E[\Delta u]$	:	Offset of Δu.
$Var[\Delta u]$	:	Variance of Δu.
$\mathcal{W}$	:	LMS weight matrix.
DR	:	Detection ratio.
g_t	:	Alarm signal.
$\hat{T}$	:	Samples eavesdropped by the adversary.
$\mathcal{P}$	:	Co-variance of the i.i.d. Gaussian signal.
$r(z)$	:	Residue.
γ	:	Detection threshold.
Γ and Ω	:	Ponderation matrices.
$(n_0$ ... $n_m)$	:	Weight of the polinomial N(z).
$(d_0$ ... $d_n)$	:	Weight of the polinomial D(z).
FN	:	False negatives.
FP	:	False positives.
AD	:	Samples detected.
SA	:	Samples under attack.

Table 2 Sample parameters used in the multi-watermark Matlab/Simulink implementation.

Distribution	**Variance (σ^2)**	**Offset**
Gaussian	5.9536	0.0
Rician	3.8870	3.7106
Rayleigh	3.0581	2.5553

Table 3 Detector performance results.

	Replay Attack	**Non-parametric Attack**	**Parametric Attack**
Detection Ratio	40.00%	18.00%	12.00%
Average Detection Time	18.81s	10.17s	6.08s

Table 4 Long run experimental results.

	Replay Attack	**Non-parametric Attack**	**Parametric Attack**
False Negatives	64.06%	85.20%	88.63%
False Positives	0.98%	1.66%	1.35%

Table 5 Representative control-theoretic results on cyber-physical attacks.

Attack Name	**References**
Replay attack	[5], [47], [48]
Dynamic false-data injection attack	[35], [49]
Stealth / False-data injection attack (Bias, Surge, Geometric)	[50] [48], [51], [52], [53]
Covert attack	[16],[54]