ANNUAL REVIEWS

Annual Review of Control, Robotics, and Autonomous Systems

Secure Networked Control Systems

Henrik Sandberg,[1] Vijay Gupta,[2] and Karl H. Johansson[1]

[1]School of Electrical Engineering and Computer Science and Digital Futures, KTH Royal Institute of Technology, Stockholm, Sweden; email: hsan@kth.se, kallej@kth.se

[2]Department of Electrical Engineering, University of Notre Dame, Notre Dame, Indiana, USA; email: vgupta2@nd.edu

Annu. Rev. Control Robot. Auton. Syst. 2022. 5:445–64

First published as a Review in Advance on November 10, 2021

The *Annual Review of Control, Robotics, and Autonomous Systems* is online at control.annualreviews.org

https://doi.org/10.1146/annurev-control-072921-075953

Keywords

attack space, false data injection attack, replay attack, denial-of-service attack, networked control systems, cyber-physical security

Abstract

Cyber-vulnerabilities are being exploited in a growing number of control systems. As many of these systems form the backbone of critical infrastructure and are becoming more automated and interconnected, it is of the utmost importance to develop methods that allow system designers and operators to do risk analysis and develop mitigation strategies. Over the last decade, great advances have been made in the control systems community to better understand cyber-threats and their potential impact. This article provides an overview of recent literature on secure networked control systems. Motivated by recent cyberattacks on the power grid, connected road vehicles, and process industries, a system model is introduced that covers many of the existing research studies on control system vulnerabilities. An attack space is introduced that illustrates how adversarial resources are allocated in some common attacks. The main part of the article describes three types of attacks: false data injection, replay, and denial-of-service attacks. Representative models and mathematical formulations of these attacks are given along with some proposed mitigation strategies. The focus is on linear discrete-time plant models, but various extensions are presented in the final section, which also mentions some interesting research problems for future work.

1. INTRODUCTION

Security in networked control systems refers to the study of attack algorithms by which an adversary can degrade the performance of a control system by utilizing a cyber-channel over which sensor measurements or control signals are being transmitted in a control loop, as well as the corresponding detection and mitigation mechanisms that can be employed to prevent performance degradation in the presence of such attacks. Attacks on control systems are notably different from attacks in which the goal of the adversary is degradation of the cyber-network itself, e.g., through distributed denial-of-service (DoS) attacks on internet routers (1, 2). They are also different from faults, which may also degrade the performance of a control system, since in the traditional fault diagnosis literature, a basic assumption is that the fault is happening without active participation by an adversary (3). Finally, cyberattacks are different from physical attacks, as the latter are carried out by an adversary who does not utilize a cyber-channel to carry out the attack.

Although cyberattacks on industrial control systems have been reported regularly for quite some time, the area was catapulted into public awareness through a few high-profile attacks, such as the Stuxnet computer worm that attacked industrial sites in Iran in 2010 (4) and the cyberattack on the Ukrainian power grid in 2015 (5). The latter consisted of interfering with the power grid's supervisory control and data acquisition (SCADA) system to remotely switch substations off and disable infrastructure components in order to interrupt the proper functioning of the power network. More generally, such cyberattacks have been noted or demonstrated against a variety of systems, including infrastructure systems, automobiles, manufacturing industries, oil refineries, and smart homes. As more control systems become networked—meaning that sensor and controller signals are transmitted over wireless or wired networks—the possibilities available to the attacker, and hence the importance of security in control systems, only increase.

The security problem in networked control systems is essentially a game between the attacker and defender. This can be interpreted formally in the context of game theory or in an informal sense as referring to a competition between these two entities: The attacker seeks to stay one step ahead of the defender in terms of designing an attack that is sophisticated enough to beat the detection and mitigation mechanisms employed by the defender. Similarly, the defender seeks to estimate the capabilities or sophistication of the attacker and deploy mechanisms to detect or mitigate the attack. This point of view has three immediate consequences. The first is that technically it makes sense to define and identify a spectrum of capabilities or resources available to the attacker and the defender in order to calculate the outcome for the control loop in a variety of possible scenarios. The second is that algorithms that can identify the capabilities of an attacker or a defender can be very useful in practice. The final consequence is that the interaction between the attacker and the defender needs to be ideally captured by a dynamic or iterative model, in which the attacker and the defender have the ability to learn from previous interactions and to improve their strategies or resource allocations.

Several important technologies are available and commonly used to defend industrial control systems against cyberattacks. How to engineer secure systems in general is a large and broad cross-disciplinary topic with specific solutions for certain application domains (6). Cryptography is a critical component and is about securing information systems in general against adversarial attacks by encrypting data for secret communication, deploying protocols for secure key exchange, allowing proper user authentication, and so on (7). In contrast to cryptography, there are also many specialized defense mechanisms, such as so-called honeypots aimed at luring an adversary into the system and then monitoring and analyzing that adversary's behavior in order to create counteractions (8). Obfuscation has often played a key role in traditional control system security in that knowledge of dedicated computer systems and communication protocols for industrial

control systems has not been widely available. The trend of building modern control systems based on off-the-shelf components and standard technologies has reduced the role of obfuscation and made these systems more vulnerable to adversarial organizations and individuals. It is a mistake for a defender to rely only on the lack of knowledge by an attacker about the system or the strategy employed by the defender. Kerckhoffs's principle (9) on military ciphers from 1883 and Shannon's maxim "the enemy knows the system" from 1948 can be interpreted as saying that the reliance on secrets in a system design should be kept to a minimum and that the defender should implement security algorithms under the assumption that the algorithms are public and thus known to the adversary. This principle leads to stronger systems in the long run and is also adopted for the algorithms discussed in this article.

Our focus is on security for networked control systems and particularly on attack and mitigation techniques targeting the requirements and nature of such systems. A key requirement is the need for real-time response as the systems are monitoring and controlling physical processes, which thus imposes a timescale that needs to be followed for the detection and mitigation of any attacks. This also means that solutions that rely on collecting and analyzing large amounts of data need to be evaluated for the latency that they impose. A related requirement is the need for the closed-loop control system to be operated safely and continuously, since breaking the loop might deteriorate the performance of the controlled physical process or even render it unstable. While one solution for mitigating an attack on an IT system may simply be to restart a device or a server, interrupting a physical process such as an infrastructure system or an autonomous car is in most practical situations impossible. In some applications, such as the power grid or the transport infrastructure, legacy devices may also be present that need to be considered while designing secure systems.

Confidentiality, integrity, and availability are fundamental requirements for computer security (6). This paradigm can also be applied to the security of networked control systems while keeping in mind the differences above. In the present context, confidentiality refers to authorized access to information. In other words, sensor or control information should be encrypted and otherwise authenticated. Authentication attacks seek to impersonate authorized users or otherwise defeat the measures in place. If networked control systems have legacy components, the problem is even more difficult. Similarly, integrity refers to the property that the data being transmitted are not replaced with malicious data by the adversary. Such data can lead to incorrect control updates, and thus the state of the process may evolve in a manner that does not satisfy constraints such as safety or stability. Most of the attacks that we consider are data integrity attacks. The physics of the process imposes some constraints on what the attacker can transmit; however, there is still considerable freedom available that can be exploited. Finally, availability refers to the property that the control system components are always available. DoS or jamming attacks can lead to sensor or state data not being available to the controller, thus challenging the decision-making process and possibly degrading the system performance to unacceptable levels.

The outline of this article is as follows. Section 2 presents three motivating examples and describes how they relate to the methodologies presented later. The system and attacker models used throughout the article are introduced in Section 3. Three extensively studied types of cyber-attacks are then discussed—false data injection (FDI) attacks (Section 4), replay attacks (Section 5), and DoS attacks (Section 6)—together with some proposed mitigation mechanisms. The article concludes with an outlook on extensions and future work in Section 7.

2. MOTIVATING EXAMPLES

Cyber-physical security vulnerabilities are present in a wide range of control applications. Some of them have already been exploited over the last few years in various attacks. In this section,

we describe three examples to illustrate this trend and to motivate models and defense strategies introduced in later sections.

2.1. Power Grid

Toward the end of 2015, the operation of part of the power grid in a region of western Ukraine was interrupted through a sophisticated cyberattack that impacted approximately 225,000 customers. The attack was probably the first one of such a scale on a power system, and it was widely reported in the media (5) and described in detail by security corporations and government organizations (10). Through spear-phishing efforts prior to the actual attack, adversaries were able to obtain virtual private network credentials to the SCADA network of the grid. On the day of the attack, these credentials allowed them to gain remote access to several substations (similar to the one shown in **Figure 1**), giving them control over circuit breakers. Opening such breakers took down power delivery for several hours for many customers. As part of the attack, they remotely installed malicious firmware on field devices at the substations, making it impossible for the remote operators to restore their operation automatically and requiring extensive manual work.

Many security vulnerabilities of the power grid have been discussed over the last decade. The grid is a large-scale networked control system that regulates frequency and voltage by ensuring that the supply equals the demand. Sensors, actuators, and control algorithms are distributed over the network and over large geographic areas. All of the cyberattacks and mitigation strategies discussed in this article are relevant for the grid. As an example, consider the state estimator for the transmission grid generating estimates for the voltage phasor magnitudes and angles, which are extensively used by control and optimization algorithms in the grid control center. Studies showed more than 10 years ago how FDI attacks could fool the state estimator (11) and that this vulnerability can be systematically evaluated (12). FDI attacks for networked control systems are discussed in detail in Section 4. DoS attacks, which are presented in Section 6, have also been studied for load frequency control (13).

Figure 1

An example of a substation. Substations transform electric voltage levels as part of the power grid, often between the transmission and distribution networks. They typically have several sensing and actuation devices and have been shown to be vulnerable to cyberattacks. Photo by ETA+ via Unsplash.

Figure 2

A vehicle platoon of three trucks. The control of the distances between the trucks is based on vehicle states being wirelessly communicated among the vehicles. Photo courtesy of Scania.

2.2. Connected and Automated Vehicles

In 2015, Miller & Valasek (14) demonstrated that it was possible to remotely take over the essential functionalities of a 2014 Jeep Cherokee while a journalist from *Wired* was driving the vehicle on a highway (15). At a conference and in an extended report, they detailed the vulnerabilities in the system and the procedure they went through to take control of the car, basically providing a recipe for how to perform cyberattacks on a large number of car models. The attack surface was the radio interface in the car, which supported both Wi-Fi and cellular communication. This interface gave indirect access to the controller area network that connected several electronic control units handling steering, transmission, braking, and many other critical functionalities. Reprogramming an embedded microcontroller with new firmware made it possible to send commands through the cellular network to an electronic control unit in order to shut down the engine remotely. The demonstrated remote attack could be done from any location, did not require any modification to the vehicle, and showed that hundreds of thousands of vehicles were vulnerable.

Future intelligent transport systems with connected and automated vehicles will have many more radio interfaces than today's vehicles and thus many more potential attack surfaces for remote intruders than the ones described above. An example of an emerging technology is heavy-duty vehicle platooning, as illustrated in **Figure 2**. Such road trains provide more energy-efficient freight transport under almost driverless operations but require wireless communication among the vehicles to regulate the intervehicle distances tightly (16). Vehicle platoons and automated vehicles in general will be supported by an advanced wireless sensing and communication infrastructure to enable the vehicles to have beyond-human situational awareness for smooth interactions between all kinds of road users and traffic controllers. Such infrastructure needs to be resilient to a large range of potential cyberattacks, such as DoS attacks overloading the communication networks and servers, which are further discussed in Section 6.

2.3. Industrial Process Automation

The Stuxnet cyberattack on industrial sites in Iran 2010 was the first large-scale attack on a SCADA system to directly exploit the nature of the targeted control system (4). It consisted of multiple phases and used specific vulnerabilities, including four zero-day exploits, which are software flaws unknown to the target software vendor. The Stuxnet worm entered the system via a USB stick and then spread to multiple computers while checking to see whether each machine was part of a targeted industrial control system. For the machines that were part of this control system, such as the control of centrifuges in a uranium-enrichment plant, the worm compromised

Figure 3

A paper mill. Such a plant consists of thousands of control loops, which in the future might be connected to the cloud to enable basically unlimited computing power for data analytics, which could be used in, for instance, advanced predictive maintenance. Photo courtesy of Iggesund Paperboard.

their programmable logic controllers in a sophisticated way. The compromised controllers first silently recorded plant information such as sensing and actuation data for a while, then used these data to generate control commands that destabilized and destroyed the controlled process; meanwhile, they also provided supervisory controllers and operators with false data, giving them the impression that the process status was normal. Such a cyberattack is nowadays called a replay attack and is further discussed in Section 5.

Wireless sensors and networks are not commonly used in today's process industry control loops, as in the industrial plant in **Figure 3**. With the current trend in embedded and cloud computing, wireless and cellular communication, and Internet of Things technology, however, drastic changes can be expected in the architecture and operation of industrial automation systems (17). Existing and new embedded devices will collect data online and feed them into the cloud for data analytics, predictive maintenance, operation optimization, and so on. Such systems will provide huge advantages in performance, resource efficiency, and flexibility, but several new attack surfaces could be established if this technological development is not done properly. Industrial processes could be open to all of the cyberattacks discussed in this article, including the replay attack described above.

3. SYSTEM AND ATTACK MODELS

The networked control systems we consider are schematically illustrated by the block diagram in **Figure 4**. The physical layer consists of the plant $\mathcal{P}$ and also includes devices such as sensors and actuators. Examples of relevant plants were given in Section 2. The cyber-layer consists of a communication network, such as a SCADA system or a field network, together with the controller $\mathcal{C}$ and a possible detector $\mathcal{D}$. The controller and detector could be centralized and located at a control center or could be distributed in, for example, programmable logic controllers. In the following, we introduce an abstract modeling framework for system-theoretic security analysis. Note that for practical security considerations, implementation details such as the choice of hardware and protocols are also essential (for a comprehensive overview, see 6).

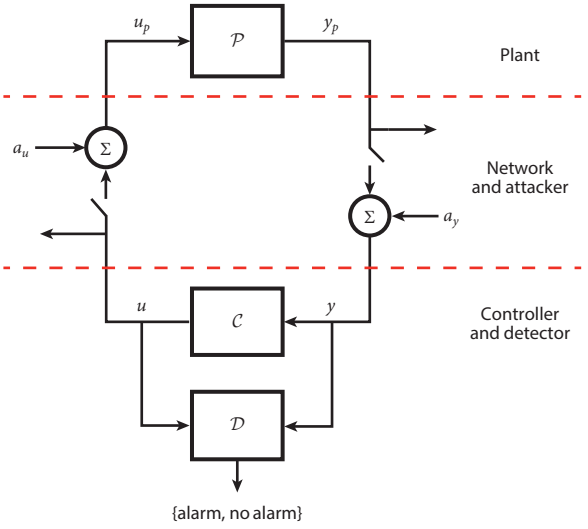

Figure 4

Block diagram of a networked control system. The network-and-attacker layer illustrates three modes of attack: denial of service (*switches*), eavesdropping (*outgoing arrows*), and false data injection (a_u, a_y). Figure adapted from Reference 59 with permission from Springer International Publishing; permission conveyed through Copyright Clearance Center, Inc.

In the following sections, we suppose that the plant $\mathcal{P}$ can be modeled as a linear discrete-time system,

$$\mathcal{P}: \begin{cases} x(k+1) = Ax(k) + Bu(k) + B_a a(k) + w(k), \quad x(0) = x_i, \\ y(k) = Cx(k) + D_a a(k) + v(k), \end{cases} \quad 1.$$

for times $k \geq 0$, with state $x(k) \in \mathbb{R}^n$, communicated control input $u(k) \in \mathbb{R}^m$, and received measurement $y(k) \in \mathbb{R}^p$. We assume that the plant is subject to process disturbance $w(k) \in \mathbb{R}^n$ and measurement noise $v(k) \in \mathbb{R}^p$. The signal $a(k) \in \mathbb{R}^q$ models attacks and is introduced in Section 4. Depending on the attack scenario considered, we also specify models of the controller $\mathcal{C}$ and detector $\mathcal{D}$.

We consider malicious attacks carried out through the network communication system as indicated in **Figure 4**. As explained in Section 1, the process operator desires confidentiality, integrity, and availability; conversely, the attacker seeks to violate one or more of these properties. For example, the attacker may inject, or manipulate, the data content in the packets between the controller, sensors, and actuators. This violates integrity and is modeled by an additive signal $a = (a_u, a_y)$ injected into the control loop in **Figure 4**. Such FDI attacks are the focus of Section 4. An attacker may also eavesdrop on communication and thus violate confidentiality. This is modeled by outgoing arrows in the network layer of **Figure 4**. For example, the attacker may record sensor measurements over a period of time to later inject and fool the controller about the plant state. This could possibly be coordinated with harmful FDI in the actuator channel; such attacks are called replay attacks and are the focus of Section 5. The actuator or sensor channels may also be taken out (or rendered unavailable) by the attacker by overwhelming the network with communication packets. This is modeled in **Figure 4** by switches, which show that such attacks effectively render the system open loop. DoS attacks are the focus of Section 6.

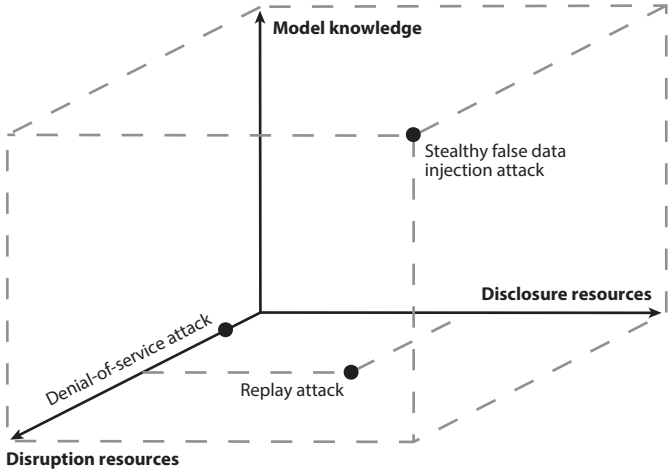

Figure 5

Attack space illustrating the adversarial resources required to conduct attacks. Disclosure resources quantify the number of channels the attacker can read, whereas disruption resources quantify channel writing ability. Model knowledge refers to how much the adversary knows about the models of components $\mathcal{P}$, $\mathcal{C}$, and $\mathcal{D}$ in use by the designer and the operator. Figure adapted from Reference 59 with permission from Springer International Publishing; permission conveyed through Copyright Clearance Center, Inc.

Figure 5 shows a three-dimensional attack space that serves to illustrate the relative resources required to implement attacks against networked control systems. Disclosure resources measure the possibilities the attacker has to read the channels u and y in **Figure 4**. These can vary in a network depending on the attacker's access to links and physical devices and on whether the operator has encrypted some channels. Disruption resources measure the possibilities to overwrite or inject new u and y packets. These depend on factors similar to those affecting disclosure resources but also on whether the operator authenticates some links. Finally, model knowledge measures how much the attacker knows about the models of $\mathcal{P}$, $\mathcal{C}$, and $\mathcal{D}$ in use by the designer and the operator. An attacker can use such knowledge to mask data attacks. For example, FDI attacks can be made undetectable or stealthy. Replay attacks do not require such knowledge, but instead exploit the ability of the adversary to read, record, and write data packets. DoS attacks only exploit the ability of the adversary to send large numbers of data packets to specific target devices and thereby overload the device or the communication channel. More advanced multistage attacks could very well move in the attack space after the attack has commenced. For example, one could envisage an attacker who first passively eavesdrops on channels to learn the models in use and later stages a stealthy FDI attack by exploiting the learned models.

4. FALSE DATA INJECTION ATTACKS

In this section, the physical plant $\mathcal{P}$ is targeted by FDI attacks, modeled by $a \neq 0$ in Equation 1. FDI attacks generally originate from interception and corruption of communication channels to and from the plant (see **Figure 4**). In such scenarios, we split the attack into the components directly affecting the control signal u and measurement y as $a = (a_u, a_y)$. In this case, $B_a = \begin{bmatrix} B & 0 \end{bmatrix}$ and $D_a = \begin{bmatrix} 0 & I \end{bmatrix}$. We define three types of FDI attacks: sensor attacks ($a_y \neq 0$, $a_u = 0$), actuator attacks ($a_u = 0$, $a_y \neq 0$), and coordinated attacks ($a_y \neq 0$, $a_u \neq 0$). FDI attacks generally affect the state, control, and sensor signals. The reason for the difference in plant

state under actuator attack is obvious. For sensor attacks, the plant state may be affected when operating in a closed loop in which a feedback controller reacts to the manipulated signal y and applies the affected control u, and thus perturbs the state. We assume, however, that the disturbances and noise, w and v, respectively, are unaffected by FDI attacks.

4.1. Attack Detection

A fundamental problem for the system operator is to determine whether the plant is subject to an FDI attack—that is, whether $a \neq 0$ or $a \equiv 0$ in Equation 1. In fault diagnosis (18) terminology, this is a detection problem. If an attack is detected, one may proceed to the isolation problem to determine the attack type and affected channels. Finally, the identification problem, which is the most complex problem, concerns the reconstruction of the exact attack sequence $a_0^k := (a(0), a(1), \ldots, a(k))$.

Assuming a statistical setup where probability distributions of $w_0^k := (w(0), w(1), \ldots, w(k))$, $v_0^k := (v(0), v(1), \ldots, v(k))$, and x_i are known, and with given data (u_0^k, y_0^k), the detection problem can be understood as a multiple hypothesis test:

$$\mathcal{H}_0 : a \equiv 0 \qquad \text{(no FDI attack)},$$

$$\mathcal{H}_{k_0} : a_0^{k_0-1} \equiv 0,\ a_{k_0}^k \neq 0 \qquad \text{(FDI attack starts at time } k_0 > 0).$$

This test should determine whether there is sufficient evidence to reject the null hypothesis $\mathcal{H}_0$ in favor of one of the alternative hypotheses $\mathcal{H}_{>0}$. A difficulty here is for the operator to characterize and model the alternative hypothesis $\mathcal{H}_{>0}$ since the capabilities and goals of the attacker are generally unknown, although risk assessment may give valuable insights. Assuming deterministic but unknown attack signals a, we may use recursive generalized likelihood-ratio tests on online data (19, 20) to generate a detection alarm as soon as possible after the actual attack has started. References 19 and 21 discuss optimality properties and relations to the so-called cumulative sum (CUSUM) test, and References 18 and 19 explain how and when it is possible to proceed to isolate and identify attacks following a detection alarm. In general, this follow-up analysis is best done in an offline mode using all available data.

If the alternative hypotheses $\mathcal{H}_{>0}$ are not well characterized, so-called nonparametric CUSUM tests (22) can be run online in order to quickly detect certain deviations from the null hypothesis $\mathcal{H}_0$. How to tune such detectors to minimize the physical damage in the presence of malicious FDI attacks is discussed in Reference 22. An example of a nonparametric CUSUM test to detect attacks is the following. First, compute residual sequence r_0^k using a state estimator:

$$\begin{aligned} \hat{x}(k+1) &= A\hat{x}(k) + Bu(k) + Kr(k), \quad \hat{x}(0) = \mathbb{E}x_\mathrm{i}, \\ r(k) &= y(k) - C\hat{x}(k). \end{aligned} \tag{2.}$$

A common choice is to use a Kalman gain K tuned under the null hypothesis $\mathcal{H}_0$, in which case the residual is the innovation sequence with desirable independent and identically distributed (i.i.d.) statistical properties. Then, compute the CUSUM statistic as

$$S(k) = \max\{S(k-1) + |r(k)|^2 - \delta, 0\}, \quad S(0) = 0,$$

where δ is a tunable forgetting parameter chosen such that $\mathbb{E}|r(k)|^2 - \delta < 0$ under the null hypothesis $\mathcal{H}_0$. The test is now

$$S(k) \underset{\mathcal{H}_0}{\overset{\mathcal{H}_{>0}}{\gtrless}} \tau,$$

where τ is a tunable alarm threshold chosen to trade off the false-alarm and correct detection rates. An alarm at time k_a supports that an abrupt change has occurred in the system at time $k_0 \leq k_a$, and we can estimate the change time as $k_0 \approx k_a - N_{k_a} + 1$, where $N_k = N_{k-1}\mathbf{1}_{\{S(k-1)>0\}} + 1$ (see 19). Here, N_k counts the number of time steps since $S(k)$ was last set to zero, and $\mathbf{1}$ is the indicator function.

The CUSUM statistic is an example of a so-called stateful test, which takes the history of the residual into account. Popular alternatives are stateless tests, such as the χ^2 test (18, 22), where only the current residual is monitored:

$$|r(k)|^2 \underset{\mathcal{H}_0}{\overset{\mathcal{H}_{>0}}{\gtrless}} \tau. \qquad 3.$$

Generally, such tests are easier to theoretically analyze but may perform worse in practice (higher false-alarm rate, easier to bypass, etc.) (22). Windowed χ^2 tests (23) are a possible stateful extension of the regular χ^2 test, with better performance.

The tests and schemes mentioned above were originally designed for the detection of faults with no malicious intent. More recent anomaly detection schemes and mitigation tools are discussed in Section 4.3. However, as discussed in the following section, a resourceful attacker can often bypass particular detection tests.

4.2. Undetectable and Stealthy Attacks

The design of attack detection algorithms is clearly very important and challenging even under strong assumptions on the attacker. A related line of work has focused on characterizing types of attacks that are particularly difficult to detect for any detection test. Such attacks are called undetectable or stealthy. The possibility of these attacks highlights inherent weaknesses in the system itself. As part of a risk management process, knowledge of such attacks and their impact can be used to guide the allocation of new security resources (24). Examples of security resources are redundant sensors, redundant actuators, and the deployment of encryption or authentication mechanisms on selected communication channels.

Next, we review some classes of undetectable and stealthy attacks.

4.2.1. Deterministic undetectable attacks. A deterministic approach to characterizing attacks that are hard to detect leads to the concept of undetectable attacks (25). Informally, an attack a is undetectable if the signals available to the controller and detector coincide with signals due to some nominal operating condition. More formally, $a \neq 0$ is undetectable if there exist initial states x_i and x_i^a such that

$$y(k; x_i, u, 0) = y(k; x_i^a, u, a), \quad k \geq 0, \qquad 4.$$

for at least some input u.[1] Hence, there exist (at least) two different conditions for the same input and output sequences. One possibility is that the system is not under attack, and another is that there is an FDI attack (albeit with a different initial condition). Typically the operator does not exactly know what the initial state is and hence cannot determine whether there is an attack. The attack is thus undetectable.

Under certain situations (generically, when there are more attack signals than outputs), the condition given in Equation 4 may hold with $x_i = x_i^a$. In this case, even if the operator knows the

[1] $y(k; x_i, u, a)$ denotes the output of Equation 1 at time k when the input is u and attack is a. For simplicity, we leave out w and v in the deterministic scenario here.

initial state, undetectable attacks exist. Such attacks are perfectly undetectable attacks (or shorter, perfect attacks; see 26).

For linear systems, the condition in Equation 4 translates into the existence of zero dynamics (transmission zeros) (25), and these attacks are also sometimes referred to as zero-dynamics attacks (27).

4.2.2. Deterministic stealthy attacks. Another deterministic approach leads to the concept of stealthy attacks (27). Whereas undetectable attacks are designed to be independent of possible fault or anomaly detection mechanisms (Section 4.1), stealthy attacks take them explicitly into account. For simplicity, consider a constraint derived from Equation 3:

$$|r(k; y, x_i, u, a)|^2 \leq \tau + \epsilon, \quad k \geq 0, \tag{5.}$$

where $r(\cdot\,; y, x_i, u, a)$ emphasizes the dependence of the residual on the system signals and attack under $\mathcal{H}_{>0}$. A stealthy attack $a \neq 0$ against the detector given in Equation 3 satisfies the constraint from Equation 5 for some threshold $\epsilon \geq 0$ chosen by the attacker. By choosing $\epsilon = 0$, the attack will not generate an alarm from this particular detector. Using $\epsilon > 0$ increases the risk for an alarm but may also increase the impact of the attack. In a similar manner, stealthy attacks may be defined for other detectors.

Intuitively, the class of deterministic stealthy attacks is larger than the class of undetectable attacks. Indeed, undetectable attacks are often stealthy attacks. However, it is possible to design detectors that trigger alarms for dangerous, or unlikely, actual outputs y. In general, it is not possible to conclude that undetectable attacks are always stealthy or vice versa.

4.2.3. Stochastic stealthy attacks. The previous classes of attacks have taken a deterministic approach. There are various methods and assumptions in the literature to handle uncertainty in noise, disturbance, and initial state (see 25, 27). If probabilistic models of the uncertain variables are available, they can be used to define stochastic stealthy attacks (28–30). Let $p_0(y_0^k)$ and $p_a(y_0^k)$ be the probability densities of the output sequence in closed loop under no attack ($\mathcal{H}_0$) and under attack ($\mathcal{H}_{>0}$), respectively. A (possibly stochastic) attack a is then ϵ-stealthy if

$$\limsup_{k\to\infty} \frac{1}{k+1} D_{\mathrm{KL}}(p_a(y_0^k) \,\|\, p_0(y_0^k)) \leq \epsilon, \tag{6.}$$

where $D_{\mathrm{KL}}(\cdot\|\cdot)$ is the Kullback–Leibler divergence between two probability densities. Intuitively, if $\epsilon > 0$ is small, correctly distinguishing between the two hypotheses with high probability requires a very long output sequence. In fact, as is shown in Reference 29, the Chernoff–Stein lemma yields that for any $\delta \in (0, 1)$ there exists no detector with detection probability $p^{\mathrm{D}}(k)$ (deciding $\mathcal{H}_{>0}$ when $\mathcal{H}_{>0}$ is true) satisfying $0 < 1 - p^{\mathrm{D}}(k) < \delta$ and simultaneously a false-alarm probability $p^{\mathrm{F}}(k)$ (deciding $\mathcal{H}_{>0}$ when $\mathcal{H}_0$ is true) decaying faster than rate ϵ,

$$\limsup_{k\to\infty} -\frac{1}{k+1} \ln p^{\mathrm{F}}(k) > \epsilon,$$

illustrating the difficulty of detecting such an attack.

4.2.4. Impact of stealthy attacks. Malicious adversaries often have specific attack objectives. In practice, a resourceful system operator can stop many attacks soon after they are detected. It then becomes important to evaluate the possible impact of stealthy attacks. Indeed, if a stealthy attack cannot cause much damage, it may not be of the highest priority in a risk management process (24). A system operator may tune the detection system to balance the impact of possible

attacks and the cost for false alarms, which is a central idea in Reference 22. We next illustrate such trade-offs in a simple example.

For detectors in stochastic systems, the worst mean delay for detection, T_D, is asymptotically bounded by the mean time between false alarms, T_F, as

$$T_D \geq \frac{\ln T_F}{\epsilon}, \quad T_F \to \infty,$$

where ϵ is a bound on the Kullback–Leibler divergence between the attacked and unattacked scenario as expressed in Equation 6. We point the reader to Reference 21 for a precise statement and definitions of the involved quantities. Since operators typically require very long times between false alarms, this bound is of practical relevance. Assume next that the damage an attack can incur at every time step it is undetected is approximately quadratic in that attack, $D(a) \approx \frac{1}{2} a^T D_{aa} a$, where D_{aa} is a positive semidefinite matrix and a is a constant (bias) attack vector. If the attack magnitude is small, we can furthermore approximate the Kullback–Leibler divergence as $D_{KL}(p_a(y) \| p_0(y)) \approx \frac{1}{2} a^T I_{aa} a$, where I_{aa} is the Fisher information matrix. Hence, the total impact of an optimal attack $a^\star$, before it is detectable and stoppable on average, can be bounded by a generalized Rayleigh quotient,

$$\ln T_F \frac{a^T D_{aa} a}{a^T I_{aa} a} \leq \ln T_F \frac{(a^\star)^T D_{aa} a^\star}{(a^\star)^T I_{aa} a^\star} = \ln T_F \cdot \lambda_{\max}(I_{aa}^{-1} D_{aa}) \leq T_D D(a^\star). \quad 7.$$

Here, $\lambda_{\max}(\cdot)$ denotes the largest eigenvalue, and we assume that the Fisher information matrix is positive definite (which corresponds to the condition that the attack cannot be undetectable).[2] Note that under the quadratic approximation mentioned above, the total impact is independent of the stealthiness parameter ϵ.

The system operator can use expressions such as those provided in Equation 7 to assess whether a stealthy attack is serious. This depends on the visibility of the attack (I_{aa}), how the attack affects the plant (D_{aa}), and the threshold τ of the detector [$\tau \mapsto T_F(\tau)$]. Assuming that there is also a cost proportional to the frequency of false alarms, C/T_F (cf. 31), the overall cost

$$\ln T_F(\tau) \cdot \lambda_{\max}(I_{aa}^{-1} D_{aa}) + \frac{C}{T_F(\tau)}$$

can be minimized by choosing an optimal detection threshold $\tau^\star$ such that

$$T_F(\tau^\star) = \frac{C}{\lambda_{\max}(I_{aa}^{-1} D_{aa})}.$$

This expression optimally balances false alarm and attack impact costs.

4.3. Mitigation

Several mitigation strategies against FDI attacks have been proposed in the literature. Fault diagnosis and change detection methods in combination with threshold tuning (18, 19, 22) have been discussed in Sections 4.1 and 4.2. In Reference 25, fundamental limitations and distributed implementations of FDI attack detectors are derived. Such fundamental limitations can be exploited in preventive security resource allocation (24). Secure estimators (32, 33) exploit sensor redundancy to always reconstruct a correct state estimate that can be used for resilient control. Data-based anomaly detection schemes based on machine learning methodologies are currently also an active

[2] The first upper bound in Equation 7 is attained by choosing the optimal attack $a^\star$ as the eigenvector of $I_{aa}^{-1} D_{aa}$ corresponding to $\lambda_{\max}$. Normalizing such that $(a^\star)^T I_{aa} a^\star = 2\epsilon$ achieves the desired level of stealthiness.

research area (see, e.g., 34, 35). Approaches for increasing the opportunities for detecting stealthy FDI attacks include moving-target defense (36) and multiplicative watermarking (37).

5. REPLAY ATTACKS

In this section, we consider replay attacks, which proceed by replaying or transmitting a delayed version of the true sensor measurements while changing the control input to degrade the control performance. Since the received measurements correspond to a measurement sequence that is possible, it can be very hard to detect via statistical means that an attacker has altered the measurements in this way. It is, thus, considered to be a different class of attacks than the FDI attacks discussed in the previous section, even though the control signal is still corrupted by an adversary. As discussed in Section 2, the Stuxnet attack is believed to have been a replay attack.

A formal definition of a replay attack was considered in References 23 and 38. In this version, the attacker is able to collect real-time sensor measurements being transmitted from the sensor to the controller and has the capability to change the true sensor measurements to values collected previously. This could be done, for instance, by knowing the cryptographic keys, or, in simpler systems that do not employ cryptography or time-stamping, by simply retransmitting the sensor measurements collected earlier. After the data collection phase, the attacker implements a different control action than the one transmitted by the controller. Thus, a replay attack can be implemented starting at time $k = 0$ for T steps by the attacker making the substitution

$$y_a(k) = y(k - T), \qquad 0 \le k \le T - 1, \tag{8.}$$

and simultaneously injecting an external input signal $u_a(k)$ for $0 \le k \le T - 1$. This attack is best applied when the system is at a steady state since the measurements will then correspond to a system state that is well regulated. In this case, the attack is difficult to detect with any statistical test on the sequence of received measurements. Depending on the actuation capability available to the attacker, the true system state can degrade rapidly.

5.1. Attack Detection

The detection methodology for a replay attack relies on the basic idea of the controller purposefully and unpredictably moving the system away from a steady-state value. This variation would not show up in the measurements that the attacker substitutes, and hence the attack can be detected. The price to pay for such variations is that even in the absence of an attack, the plant state moves away from the desired value. Furthermore, there is a trade-off between the amount of control energy expended in such variations and the ease with which an attack can be detected.

To consider a simple example, consider a system of the form in Equation 1 with $a(k) \equiv 0$ and zero-mean white Gaussian noises w and v with covariances Σ_w and Σ_v, respectively. Assume that the nominal controller is a linear–quadratic–Gaussian (LQG) controller that minimizes the standard quadratic cost

$$J = \lim_{T\to\infty} \mathbb{E}\frac{1}{T}\left(\sum_{k=0}^{T-1}\left(x^{\mathrm{T}}(k)Qx(k) + u^{\mathrm{T}}(k)Ru(k)\right)\right). \tag{9.}$$

In steady state, the controller first computes the minimum mean squared error estimate $\hat{x}(k|k)$ of the state $x(k)$ given measurements $y(0), \ldots, y(k)$ and then calculates the control input

$$u(k) = u^{\star}(k) = -\left(B^{\mathrm{T}}SB + R\right)^{-1} B^{\mathrm{T}}SA\hat{x}(k|k), \tag{10.}$$

where S satisfies the Riccati equation

$$S = A^{\mathrm{T}}SA + Q - A^{\mathrm{T}}SB - \left(B^{\mathrm{T}}SB + R\right)^{-1} B^{\mathrm{T}}SA. \tag{11.}$$

If the covariance of the steady-state estimation error for the estimate $\hat{x}(k|k-1)$ is given by P, then the cost achieved by the LQG controller is given by

$$J = \operatorname{trace}\left(S\Sigma_w\right) + \operatorname{trace}\left(\left(A^{\mathrm{T}}SA + Q - S\right)\left(P - PC^{\mathrm{T}}\left(CPC^{\mathrm{T}} + R\right)^{-1}CP\right)\right). \tag{12.}$$

One could imagine that if an attacker changes the measurements, it might show up in the statistics of the innovation sequence of the minimum mean squared error estimates being calculated. We could propose checking the mean and variance of the sequence $y(k) - C\hat{x}(k|k-1)$ to see whether an attack is present. It is easy to see that, in the absence of an attack, this sequence is an i.i.d. sequence of Gaussian variables with mean zero and variance $CPC^{\mathrm{T}} + \Sigma_v$. Thus, for instance, for a window size of detection N, a χ^2 detector could be used to check for the mean and variance of this sequence being calculated. However, for a replay attack, if the measurements being replayed were collected when the plant was at steady state, these statistics do not change, and such a detector would not be able to identify an attack in progress.

5.2. Mitigation

To detect a replay attack, we redesign the controller as

$$u(k) = u^{\star}(k) + \Delta u(k), \tag{13.}$$

where $\delta u(k)$ is drawn from an i.i.d. Gaussian sequence with mean 0 and covariance Σ_u in a manner that is independent of the inputs $u^{\star}(k)$, which are calculated as in Equation 10. These variables can be viewed as an authentication signal and have also been viewed in the literature as a physical watermarking signal, since the detector expects to see a signature of this signal in the future time steps. Clearly, the controller is no longer LQG optimal, and the choice of the authentication signal above is rather ad hoc. However, for this sequence, the trade-off between the ease of detection of the attack (say, by a χ^2 detector) and the degradation in control performance can be easily characterized. Specifically, under some technical conditions, we can calculate the LQG performance now as

$$\bar{J} = J + \operatorname{trace}\left(\left(R + B^{\mathrm{T}}SB\right)\Sigma_u\right). \tag{14.}$$

We can characterize the expectation of a χ^2 detector as follows. First, we note that the following holds:

$$\lim_{k\to\infty} \mathbb{E}\left[(y(k) - C\hat{x}(k|k-1))^{\mathrm{T}}(CPC^{\mathrm{T}} + R)^{-1}(y(k) - C\hat{x}(k|k-1))\right]$$
$$= \begin{cases} p & \text{if no attack} \\ p + 2\operatorname{trace}(C^{\mathrm{T}}(CPC^{\mathrm{T}} + R)^{-1}C)\mathcal{U} & \text{if attack,} \end{cases} \tag{15.}$$

where p is the dimension of $y(k)$, P is the solution of the Riccati equation corresponding to the estimation problem, and $\mathcal{U}$ is the solution of a Lyapunov equation (for more details, see 38). This result can then be used to show that for a χ^2 detector that considers a window of T measurements, its steady-state expectation is increased from pT without attack to $pT + 2\operatorname{trace}(C^{\mathrm{T}}(CPC^{\mathrm{T}} + R)^{-1}C\mathcal{U})T$.

Since $\mathcal{U}$ is an increasing function of Σ_u, there is a trade-off between the ease of detection of the attack and the loss in LQG performance without an attack. In a single-input, single-output

system, there is only one way to insert the random signal δu and only one way to observe it. Thus, to achieve a certain detection rate, a certain performance loss has to be accepted. In multiple-input, multiple-output systems, however, we have more degrees of freedom. The random signal can be optimized such that the detection requirements are met while minimizing the effect on controller performance. Reference 23 provides two ways in which the optimization problem can be posed and solved.

The formulation discussed above can be extended in various ways. For instance, Reference 39 removed the assumption of the authentication (or watermarking) signal $\delta u(k)$ being chosen to be an i.i.d. sequence, as well as the assumption of the χ^2 detector being employed to detect whether an attack is in progress. That work considered instead the case when the authentication signal is designed as the output of a linear dynamical system expressed in a state-space form. Furthermore, instead of the χ^2 detector, an optimal Neyman–Pearson detector was identified to decide between the two hypotheses of an attack being present or not. Together, these assumptions allow consideration of an adversary employing more intelligent attack strategies. The performance of this detector can be characterized by an appropriate Kullback–Leibler divergence between the probability density functions characterizing the output sequence under the two rival hypotheses. An optimization problem for designing the authentication signal can once again be posed.

Satchidanandan & Kumar (40), in addition to presenting a nice overview of the area of replay attacks, extended these results in many directions, such as considering systems with arbitrary delay, partially observed systems, and non-Gaussian systems. The latter, in particular, remains a significant open problem since the assumption of linear systems driven by Gaussian noises provides a significant advantage in designing and analyzing the statistical tests for detecting an attack. We would also like to mention a paper by Ferrari & Teixeira (41), which considered a multiplicative watermarking algorithm that utilized a watermark-removing function to prevent any sacrifice of control performance.

6. DENIAL-OF-SERVICE ATTACKS

In general, a DoS attack is a cyberattack in which a malicious actor aims to render a computer, machine, or system unavailable for its user. This is often done by flooding a computer network or server with data traffic, preventing it from serving its users or even causing it to crash. In this section, we discuss DoS attacks for networked control systems. Such attacks temporarily or indefinitely interrupt the feedback control loop and can thereby have drastic consequences for the system operation. Obviously, if the loop of an unstable plant is opened, the attack leads to the overall system becoming unstable. It is natural to model DoS attacks as switches introduced into the networked control system, where an open switch indicates a control loop under attack. We present such a model next, followed by some mitigation strategies proposed in the literature. Reference 42 provides a more extensive overview of DoS attacks in control systems.

Consider the networked control system in **Figure 4** with the plant dynamics given in Equation 1 but with no additive injected signal: $a(k) \equiv 0$. Suppose that the closed-loop system is exposed to DoS attacks in the communication of the control commands and sensor measurements. Such attacks are represented by the switches in **Figure 4** and in the model by introducing the multiplicative relations

$$u_{\mathrm{p}}(k) = \gamma(k)u(k), \qquad y(k) = \delta(k)y_{\mathrm{p}}(k),$$

where u_{p} is the control applied to the plant and y_{p} is the plant output. The binary variables $\gamma(k)$ and $\delta(k)$ represent DoS attacks, such that $\gamma(k) = 0$ if the control command communication is interrupted by an attack and $\gamma(k) = 1$ otherwise, and $\delta(k)$ is defined in a similar way. In this model,

γ and δ are decision variables of the adversary. The attack space in **Figure 5** shows that a DoS attack can be performed using only disruption resources, while in a more sophisticated scenario the adversary could also use other available information and resources.

The interplay between the decision by the controller and the one by the adversary leads to different analysis and design problems. One relevant case is when the controller is given, for instance, as a static feedback law $u(k) = -Ky(k)$. The closed-loop system then becomes a switched system, which, without noise [$w(k) = v(k) \equiv 0$] and with scalar controls and measurements, can be written as

$$x(k+1) = (A - BK\gamma(k)\delta(k))x(k). \quad 16.$$

Such systems can be analyzed using Lyapunov methods based on input-to-state stability (43), as was done in Reference 44, which derived conditions on how long and how often DoS attacks can happen without rendering the closed-loop system unstable. In this work, the adversary is represented by deterministic sequences $\gamma(k)$ and $\delta(k)$, $k = 0, 1, \ldots$. These results make it possible to reason about worst-case situations and can give an indication about what combinations of open- and closed-loop dynamics together with DoS attack patterns are especially undesirable. Uncertainty in the model and communication can also be conveniently included in the analysis. Nonlinear and continuous-time plant models can be considered as well (45).

Networked control systems under DoS attacks have also been considered when the plant is exposed to stochastic uncertainties and the attack signal is randomly generated. Amin et al. (46) considered an LQG setting, where w and v in Equation 1 are assumed to be i.i.d. Gaussian noise. The adversary is supposed to let the sequences γ and δ be Bernoulli distributed with fixed success probabilities $\bar{\gamma}$ and $\bar{\delta}$. Under such a model, it is shown that the optimal controller subject to power constraints can be derived by solving a semidefinite program. A slightly more realistic attack model is also considered, where the adversary has to decide how a finite budget of dropouts should be utilized, denoted as block attacks. For an LQG-controlled system, Zhang et al. (47) derived the worst possible (optimal for the adversary) DoS attack and showed under general assumptions that it has such a block structure. If the dropouts are due to a jamming attack of a wireless communication channel, it is possible to model the influence of the jamming signal on the signal-to-interference-plus-noise ratio. Li et al. (48) studied such an attack for a remote state estimation problem, showing that under a power-constrained setup, a game can be formulated between the transmitter and the attacker, and this game admits a pure-strategy Nash equilibrium. Another modification of the memoryless Bernoulli process attack model above is to extend it to a hidden Markov model. Befekadu et al. (49) considered this problem and came up with a risk-sensitive control formulation, which they showed has a nice separation principle that allows the optimal control to be recursively computed.

6.1. Mitigation

Several mitigation strategies for networked control systems under DoS attacks have been proposed in the literature. Some strategies are based on the analysis provided by the attacks described above, such as making sure that a sufficient number of sensor or control packets are transmitted (44, 46) or that these packets are transmitted with sufficiently high power (48) to weaken the influence of the malicious intent. For a networked control system, it is natural to utilize the inherent resilience provided by the network, such as multipath routing and data authentication. For instance, allowing certain sensor data packets to take multiple paths or to randomly change paths can considerably enhance overall security (50). Sarıtaş et al. (51) considered a more sophisticated defense setup that uses a two-level defense mechanism with an intrusion detection system and continuous

authentication. The intrusion detection system is responsible for detecting the attacker and network anomalies. Since traditional authentication is insufficient to prevent identity theft attacks, continuous authentication based on the characterization of user behavior has emerged. The authors showed how the operator can combine these defense mechanisms to make the system more secure even under advanced learning-based attacks.

7. OUTLOOK

To conclude, we present a brief outlook and discuss some aspects of secure networked control systems not covered in the previous sections. To simplify the discussion, we focused in this article on a discrete-time linear system setup. For some of the results discussed on FDI, replay, and DoS attacks, there exist extensions to continuous-time as well as nonlinear systems. However, many questions remain open in that setting, particularly in the stochastic setting.

Another important aspect covered only to a limited extent in the literature is the robustness to system model uncertainties and to limited knowledge about the adversary (see Section 3 on system and attack models). For example, the problems in which the system parameters are unknown are becoming increasingly important. In practice, both attack and defense strategies based on probing and learning the system and user responses from available data seem to be more and more common (52). Some recent works exist in which the attacks above are combined with learning algorithms. For example, the watermark design problem mentioned as a mitigation strategy for the replay attack in Section 5 was extended in this direction by Liu et al. (53), who presented an online learning algorithm to simultaneously infer the parameters of the system and generate the watermark signal as well as the optimal detector. Interestingly, the watermark signal asymptotically converges to a signal that does not satisfy the persistent excitation condition. The authors showed that by controlling the convergence rate, one can still guarantee that the system parameters will almost surely converge to the true parameters. This work could expand watermarking in several directions, including consideration of nonlinear systems, through appropriate learning techniques. It is also interesting to consider other attack scenarios, where a mitigation strategy could be based on joint online system identification and defense algorithm design problems.

Machine learning and data analytics have been widely applied to cyber-security problems (54, 55). Some of these studies are also relevant for networked control systems. Methods not relying on a known system model can be advantageous in many situations. One aspect where they are particularly relevant even if the system model is known is when the model is nonlinear or the noises are not Gaussian, since the resulting statistics of the signals make tools depending on Kalman filters or hypothesis testing difficult to apply. Signal-based methods include some mitigation strategies based on classical detection schemes, as discussed in Sections 4 and 5. Causality measures can be useful in detecting anomalies in a networked control system without accurate knowledge of the model or statistics. Transfer entropy is one such measure from physics, indicating how much a signal can improve the prediction capability of another signal. Shi et al. (56) showed that, by detecting changes in the transfer entropy, one can mitigate FDI, replay, and DoS attacks in a data-driven fashion without relying on a model of the underlying system.

The discussion in this article also assumed a stand-alone process. There is a rich literature on secure multiagent control systems that has examined a variety of problems, from the simple consensus protocol under cyberattacks to more general multiagent dynamics (see, e.g., 57). As a further illustration of these results, consider the consensus network under DoS attacks described by Senejohnny et al. (58). The authors showed that consensus can be preserved despite the attack by suitable time-varying control and communication policies. However, for more general distributed control systems with multiple attackers, the problem becomes quite complicated

(especially if coordination, detection, and mitigation need to be done in a distributed manner), as issues of signaling become relevant.

Finally, the development of courses and curricula in which the security of networked control systems plays an essential role is important for the wider deployment of the protection methods developed over the last couple of decades. Today, security is seldom taught as part of control courses in chemical, civil, electrical, and mechanical engineering, even though systems from these domains are continuously exposed to cyber-vulnerabilities. The curricula at various levels—undergraduate, graduate, and continuing education—need to be developed in a holistic manner that includes both theory and practice.

DISCLOSURE STATEMENT

The authors are not aware of any affiliations, memberships, funding, or financial holdings that might be perceived as affecting the objectivity of this review.

ACKNOWLEDGMENTS

This work was partially supported by the Swedish Research Council, Swedish Strategic Research Foundation, Swedish Civil Contingencies Agency, Knut and Alice Wallenberg Foundation, US Army Research Office, and US Air Force Office of Scientific Research.

LITERATURE CITED

1. Garber L. 2000. Denial-of-service attacks rip the Internet. *Computer* 33(4):12–17
2. Pelechrinis K, Iliofotou M, Krishnamurthy SV. 2010. Denial of service attacks in wireless networks: the case of jammers. *IEEE Commun. Surv. Tutor.* 13:245–57
3. Gao Z, Cecati C, Ding SX. 2015. A survey of fault diagnosis and fault-tolerant techniques—part I: fault diagnosis with model-based and signal-based approaches. *IEEE Trans. Ind. Electron.* 62:3757–67
4. Kushner D. 2013. The real story of Stuxnet. *IEEE Spectr.* 50(3):48–53
5. Zetter K. 2016. Inside the cunning, unprecedented hack of Ukraine's power grid. *Wired*, Mar. 3. **https://www.wired.com/2016/03/inside-cunning-unprecedented-hack-ukraines-power-grid**
6. Anderson R. 2020. *Security Engineering: A Guide to Building Dependable Distributed Systems*. Indianapolis, IN: Wiley. 3rd ed.
7. Katz J, Lindell Y. 2007. *Introduction to Modern Cryptography*. Boca Raton, FL: CRC
8. Mokube I, Adams M. 2007. Honeypots: concepts, approaches, and challenges. In *Proceedings of the 45th Annual Southeast Regional Conference*, pp. 321–26. New York: ACM
9. Petitcolas FAP, ed. 2011. *Encyclopedia of Cryptography and Security*. Boston: Springer
10. Cybersecur. Infrastruct. Secur. Agency. 2016. ICS alert (IR-ALERT-H-16-056-01): cyber-attack against Ukrainian critical infrastructure. *Cybersecurity and Infrastructure Security Agency*, Feb. 25. **https://us-cert.cisa.gov/ics/alerts/IR-ALERT-H-16-056-01**
11. Liu Y, Reiter MK, Ning P. 2009. False data injection attacks against state estimation in electric power grids. In *Proceedings of the 16th ACM Conference on Computer and Communications Security*, pp. 21–32. New York: ACM
12. Teixeira A, Amin S, Sandberg H, Johansson KH, Sastry SS. 2010. Cyber-security analysis of state estimators in electric power systems. In *49th IEEE Conference on Decision and Control*, pp. 5991–98. Piscataway, NJ: IEEE
13. Liu S, Liu XP, El Saddik A. 2013. Denial-of-Service (DoS) attacks on load frequency control in smart grids. In *2013 IEEE PES Innovative Smart Grid Technologies Conference*. Piscataway, NJ: IEEE. **https://doi.org/10.1109/ISGT.2013.6497846**
14. Miller C, Valasek C. 2015. *Remote exploitation of an unaltered passenger vehicle*. Paper presented at Black Hat USA, Las Vegas, NV, Aug. 1–6. Extended report available at **http://illmatics.com/Remote%20Car%20Hacking.pdf**

15. Greenberg A. 2015. Hackers remotely kill a Jeep on the highway—with me in it. *Wired*, July 21. **https://www.wired.com/2015/07/hackers-remotely-kill-jeep-highway**
16. Besselink B, Turri V, van de Hoef S, Liang KY, Alam A, et al. 2016. Cyber-physical control of road freight transport. *Proc. IEEE* 104:1128–41
17. Ahlen A, Akerberg J, Eriksson M, Isaksson AJ, Iwaki T, et al. 2019. Toward wireless control in industrial process automation: a case study at a paper mill. *IEEE Control Syst. Mag.* 39(5):36–57
18. Ding SX. 2008. *Model-Based Fault Diagnosis Techniques: Design Schemes, Algorithms, and Tools*. London: Springer. 1st ed.
19. Basseville M, Nikiforov IV. 1993. *Detection of Abrupt Changes: Theory and Application*. Englewood Cliffs, NJ: Prentice Hall
20. Willsky A, Jones H. 1976. A generalized likelihood ratio approach to the detection and estimation of jumps in linear systems. *IEEE Trans. Autom. Control* 21:108–12
21. Lai TL. 1998. Information bounds and quick detection of parameter changes in stochastic systems. *IEEE Trans. Inform. Theory* 44:2917–29
22. Giraldo J, Urbina D, Cardenas A, Valente J, Faisal M, et al. 2018. A survey of physics-based attack detection in cyber-physical systems. *ACM Comput. Surv.* 51:76
23. Mo Y, Chabukswar R, Sinopoli B. 2014. Detecting integrity attacks on SCADA systems. *IEEE Trans. Control Syst. Technol.* 22:1396–407
24. Miloević J, Teixeira A, Tanaka T, Johansson KH, Sandberg H. 2020. Security measure allocation for industrial control systems: exploiting systematic search techniques and submodularity. *Int. J. Robust Nonlinear Control* 30:4278–302
25. Pasqualetti F, Dörfler F, Bullo F. 2013. Attack detection and identification in cyber-physical systems. *IEEE Trans. Autom. Control* 58:2715–29
26. Weerakkody S, Liu X, Son SH, Sinopoli B. 2017. A graph-theoretic characterization of perfect attackability for secure design of distributed control systems. *IEEE Trans. Control Netw. Syst.* 4:60–70
27. Teixeira A, Shames I, Sandberg H, Johansson KH. 2015. A secure control framework for resource-limited adversaries. *Automatica* 51:135–48
28. Bai CZ, Gupta V, Pasqualetti F. 2017. On Kalman filtering with compromised sensors: attack stealthiness and performance bounds. *IEEE Trans. Autom. Control* 62:6641–48
29. Bai CZ, Pasqualetti F, Gupta V. 2017. Data-injection attacks in stochastic control systems: detectability and performance tradeoffs. *Automatica* 82:251–60
30. Kung E, Dey S, Shi L. 2017. The performance and limitations of ϵ-stealthy attacks on higher order systems. *IEEE Trans. Autom. Control* 62:941–47
31. Umsonst D, Sandberg H. 2018. A game-theoretic approach for choosing a detector tuning under stealthy sensor data attacks. In *2018 IEEE Conference on Decision and Control*, pp. 5975–81. Piscataway, NJ: IEEE
32. Chong MS, Wakaiki M, Hespanha JP. 2015. Observability of linear systems under adversarial attacks. In *2015 American Control Conference*, pp. 2439–44. Piscataway, NJ: IEEE
33. Fawzi H, Tabuada P, Diggavi S. 2014. Secure estimation and control for cyber-physical systems under adversarial attacks. *IEEE Trans. Autom. Control* 59:1454–67
34. Al Makdah AA, Katewa V, Pasqualetti F. 2020. A fundamental performance limitation for adversarial classification. *IEEE Control Syst. Lett.* 4:169–74
35. Li D, Martínez S. 2021. High-confidence attack detection via Wasserstein-Metric computations. *IEEE Control Syst. Lett.* 5:379–84
36. Weerakkody S, Sinopoli B. 2015. Detecting integrity attacks on control systems using a moving target approach. In *2015 54th IEEE Conference on Decision and Control*, pp. 5820–26. Piscataway, NJ: IEEE
37. Teixeira A, Ferrari RM. 2018. Detection of sensor data injection attacks with multiplicative watermarking. In *2018 European Control Conference*, pp. 338–43. Piscataway, NJ: IEEE
38. Mo Y, Sinopoli B. 2009. Secure control against replay attacks. In *Proceedings of the 47th Annual Allerton Conference on Communication, Control, and Computing*, pp. 911–18. Piscataway, NJ: IEEE
39. Mo Y, Weerakkody S, Sinopoli B. 2015. Physical authentication of control systems: designing watermarked control inputs to detect counterfeit sensor outputs. *IEEE Control Syst. Mag.* 35(1):93–109
40. Satchidanandan B, Kumar PR. 2017. Dynamic watermarking: active defense of networked cyber–physical systems. *Proc. IEEE* 105:219–40

41. Ferrari R, Teixeira A. 2017. Detection and isolation of replay attacks through sensor watermarking. *IFAC-PapersOnLine* 50(1):7363–68
42. Cetinkaya A, Ishii H, Hayakawa T. 2019. An overview on denial-of-service attacks in control systems: attack models and security analyses. *Entropy* 21:210
43. Sontag ED. 2008. Input to state stability: basic concepts and results. In *Nonlinear and Optimal Control Theory*, ed. P Nistri, G Stefani, pp. 163–220. Berlin: Springer
44. De Persis C, Tesi P. 2015. Input-to-state stabilizing control under denial-of-service. *IEEE Trans. Autom. Control* 60:2930–44
45. De Persis C, Tesi P. 2016. Networked control of nonlinear systems under denial-of-service. *Syst. Control Lett.* 96:124–31
46. Amin S, Cárdenas AA, Sastry SS. 2009. Safe and secure networked control systems under denial-of-service attacks. In *Hybrid Systems: Computation and Control*, ed. R Majumdar, P Tabuada, pp. 31–45. Berlin: Springer
47. Zhang H, Cheng P, Shi L, Chen J. 2016. Optimal DoS attack scheduling in wireless networked control system. *IEEE Trans. Control Syst. Technol.* 24:843–52
48. Li Y, Quevedo DE, Dey S, Shi L. 2017. SINR-based DoS attack on remote state estimation: a game-theoretic approach. *IEEE Trans. Control Netw. Syst.* 4:632–42
49. Befekadu GK, Gupta V, Antsaklis PJ. 2015. Risk-sensitive control under Markov modulated Denial-of-Service (DoS) attack strategies. *IEEE Trans. Autom. Control* 60:3299–304
50. Vukovic O, Sou KC, Dan G, Sandberg H. 2012. Network-aware mitigation of data integrity attacks on power system state estimation. *IEEE J. Sel. Areas Commun.* 30:1108–18
51. Sarıtaş S, Shereen E, Sandberg H, Dán G. 2019. Adversarial attacks on continuous authentication security: a dynamic game approach. In *Decision and Game Theory for Security*, ed. T Alpcan, Y Vorobeychik, JS Baras, G Dán, pp. 439–58. Cham, Switz.: Springer
52. Trejo KK, Clempner JB, Poznyak AS. 2016. Adapting strategies to dynamic environments in controllable stackelberg security games. In *2016 IEEE 55th Conference on Decision and Control*, pp. 5484–89. Piscataway, NJ: IEEE
53. Liu H, Mo Y, Yan J, Xie L, Johansson KH. 2020. An online approach to physical watermark design. *IEEE Trans. Autom. Control* 65:3895–902
54. Dua S, Du X. 2016. *Data Mining and Machine Learning in Cybersecurity*. Boca Raton, FL: CRC
55. Xin Y, Kong L, Liu Z, Chen Y, Li Y, et al. 2018. Machine learning and deep learning methods for cybersecurity. *IEEE Access* 6:35365–81
56. Shi D, Guo Z, Johansson KH, Shi L. 2018. Causality countermeasures for anomaly detection in cyber-physical systems. *IEEE Trans. Autom. Control* 63:386–401
57. Sundaram S, Hadjicostis CN. 2011. Distributed function calculation via linear iterative strategies in the presence of malicious agents. *IEEE Trans. Autom. Control* 56:1495–508
58. Senejohnny D, Tesi P, De Persis C. 2018. A jamming-resilient algorithm for self-triggered network coordination. *IEEE Trans. Control Netw. Syst.* 5:981–90
59. Sandberg H. 2021. Cyber-physical security. In *Encyclopedia of Systems and Control*, ed. J Baillieul, T Samad, pp. 480–87. Cham, Switz.: Springer. 2nd ed.