On cyber-physical attacks in bilateral teleoperation systems: An experimental analysis

Andrei Munteanu, Riccardo Muradore, Massimo Merro and Paolo Fiorini
Department of Computer Science, University of Verona, Italy.

***Abstract*—In the last years there has been a significant increase in the number of attacks to the security of cyber-physical systems such as SCADA systems and PLC's. We investigate *security issues* in *bilateral teleoperation systems i.e.*, systems in which the tele-operated robots reflect back to the operator reaction forces from the task being performed. We classify the cyber-physical attacks in terms of target component, knowledge of the system, and goal. We then implement our attacks on a a single-input single-output bilateral teleoperation system equipped with a *two-layer control architecture* for force feedback. This setup allows us to study the impact of our attacks, and represents a first step towards a complete threat model with corresponding counter-measures in more complex bilateral teleoperation systems.**

I. Introduction

Robotic technologies supported by the Internet of Things paradigm are making their way into consumer and professional products, thus raising concerns about the impact of malicious activities targeting their physical and logical devices.

In the service robotics field, some of these devices are remotely controlled, *i.e.*, tele-operated by a remote operator who commands the robot motion through either dedicated networks or the Internet. Typical applications are deep sea exploration, emergency response, drone control and robotic surgery.

In the last years, several attacks have targeted cyber-physical systems (CPSs) and industrial critical systems (ICSs), *e.g.*, manipulating sensor readings and, in general, influencing physical processes to bring the system into a state desired by the attacker. A few attacks have been so effective that they made the international news: the Stuxnet worm, which reprogrammed PLCs of nuclear centrifuges in Iran [1]; the attack on a sewage treatment facility in Queensland, Australia, which manipulated the SCADA system to release raw sewage into local rivers and parks [2]; the attack on a German steel mill that prevented to shut down a blast furnace in 2014. Actually, 22 out 295 "security incidents" reported in the US ICS CERT 2015 [3] reached the core of critical control systems.

The primary approach followed by academia and industry to face cyber-physical attacks has been to secure the communication infrastructure and hardening of control systems. There is a large body of literature on how to adapt existing IT security methods to the characteristic features of the control domain [4]. However, as pointed out by Gollmann et al. [5], the concern for consequences at the physical level puts *CPS security* apart from standard *IT security*, and demands for *ad hoc* solutions to properly address a new kind of attacks: *cyber-physical attacks*.

Gollmann et al. [5] provide a clear picture of the possible goals of a cyber-physical attacker: (i) *equipment damage*, *i.e.*, attacks aiming for physical damage of equipment or infrastructure (*e.g.*, pipes, valves, etc.); (ii) *production damage*, when the attacker goes after the production process to spoil the product or make production more expensive; (iii) *compliance violation*, when the attacker tries to damage the safety and the environment impact of the industrial plant. Furthermore, they fix the *stages* that a cyber-physical attacker should go through before achieving her goals: *access*, *discovery*, *control*, *damage*, and *cleanup*. In the current paper, we focus on the fourth stage, damage, where the attacker may have a rough idea of the control plan of the target system. The systematic preparatory analysis to acquire knowledge about the system under attack is beyond the scope of our work.

In this paper, we investigate *security issues* in *bilateral teleoperation systems i.e.*, systems in which the tele-operated robots reflect back to the operator reaction forces from the task being performed. The security of teleoperation systems has been addressed in [6], [7], [8], [9] with a focus on *robotic surgery*, a safety critical application. The target system of these experiments is the Raven II laboratory surgical robot [10].

However, the aforementioned works analyze only *unilateral teleoperation systems*, *i.e.*, systems in which the remote robots do not reflect back to the operator forces and/or positions. Whereas in a bilateral contexts both master and slave provide sensing and actuating capabilities to leverage a perception to the operator as if she was in the remote environment. As a consequence, in bilateral systems both sensors and actuators on the master side provide an additional attack vector. Although bilateral teleoperation systems are used in specific applications, an exhaustive analysis of their security properties allows us to identify weaknesses occurring in unilateral systems as well. This is because bilateral systems are more sensitive to stability degradation and to the response capabilities of the human operator.

To the best of our knowledge, no previous work has studied cyber-physical attacks in *bilateral teleoperation systems*.

Contributions: We implement a number of different attacks on a simple bilateral teleoperation system consisting of two *haptic paddles* [11] equipped with the *two-layer control architecture* [12]. This setup avoids the complication of multi-input multi-output systems, such as real robots, allowing us to focus on the effects of the attacks. The proposed attacks tamper with both the computer network traffic and the mechatronics of the system, *i.e.*, sensors and actuators.

Require: $x_i(k), v_i(k), \tau_i(k), H_+(k)$, where $i \in \{s, m\}$
Ensure: $, \overline{\tau}_i(k), H_-(k)$

Update Tank energy value:
1: $H(k) = H(k-1) + H_+(k) - \Delta H_I(k) - H_-(k)$
where,
- $H_+(k)$ received energy from the other side
- $\Delta H_I(k) = \tau_i(k-1)\Delta x(k)$ energy needed by the robot
- $H_-(k) = \beta(H(k-1) + H_+(k) - \Delta H_I(k))$, $\beta \in \{0, 1\}$ sent energy to the other side

Compute torque:
2: $\tau_i^{PL}(k) = \text{sign}(\tau_i)\min(|\tau_i|, |\tau_{max1}|, |\tau_{max2}|)$
where
- $\tau_{max1} = \begin{cases} 0, & H(k) \leq 0 \\ \tau_i(k), & \text{otherwise} \end{cases}$
- $\tau_{max2} = H(k)/(v_\# T)$

3: **if** $isMaster$ **then**
4: $\overline{\tau}_m(k) = \tau_m^{PL}(k) + \tau_{TLC}(k)$
where,
- $\tau_{TLC} = -d(k)v_m(k)$
- $d(k) = \begin{cases} \alpha(H_d - H(k)), \text{if } H(k) < H_d, & \alpha \in \mathbb{R}^+ \\ 0 \end{cases}$

5: **else**
6: $\overline{\tau}_s(k) = \tau_s^{PL}(k)$
7: **end if**

Algorithm 1: Passivity layer

Our contribution can be summarized as follows:

- We classify attacks in terms of targeted components, system knowledge, and goals, in the context of bilateral teleoperation systems.
- We replicate known attacks on unilateral systems in our bilateral setting to analyze how the force feedback, transmitted to the master device from the slave, is affected during the attacks.
- We propose an attack that can only affect a bilateral architecture as it compromises the controller that provides stability to the system. This attack tampers with the network packets sent to the slave and degrades the performances of the system.
- Finally, we exhibit an attack that tampers the controller to damage the robot motors which does not show any immediate effects on neither the cyber component nor the physical one. This attack targets the slave's actuator to damage it in the long run and requires the knowledge of the mechatronical parts of the robot.

Outline: The paper is organized as follows. In Section II we briefly summarize the two-layer control architecture. In Section III we provide a taxonomy of cyber-physical attacks on bilateral teleoperation systems. Experimental results are discussed in Section IV. Finally, in Section V we draw conclusions and discuss related and future work.

II. Two-Layer Teleoperation Architecture

Our analysis focuses on bilateral teleoperation systems. These system may get unstable because of different combinations of the following aspects: communication delays, contact with the remote environment, relaxed grasp of the master device and force control settings of the slave manipulator. The *passivity theory* [13] has provided a solution for the stability of bilateral teleoperation systems through several implementations that can be found in the literature (see, for instance, [14], [15] and [12]). Our experimental setup implements the *two-layer approach* proposed in [12] to guarantee the stability in the presence of time-varying communication delay. In this approach, the control architecture has a *transparency layer*, where the commands for the master and the slave robots are computed according to the chosen architecture (*e.g.*, position-force as in [12], or position-position [16] as in the present setup), and a *passivity layer* that guarantees the passivity of the system by monitoring two energy tanks H_m and H_s at the master and slave side, respectively, (see Figure 1).

In our position-position architecture, the command torques are computed as follows:

$$\begin{aligned} \tau_s(k) &= k_P(x_{sd}(k) - x_s(k)) + k_D(v_{sd}(k) - v_s(k)) \\ \tau_m(k) &= k_P(x_{md}(k) - x_m(k)) + k_D(v_{md}(k) - v_m(k)) \end{aligned}$$

where $t = kT_s$ is the time, $k \in \mathbb{Z}$, T_s is the sample time of the discrete-time controllers, and the reference positions are

$$\begin{aligned} x_{sd}(k) &= x_m(k - d_{m2s}(k)) \\ x_{md}(k) &= x_s(k - d_{s2m}(k)) \end{aligned}$$

with d_{m2s} and d_{s2m} are the master-to-robot and robot-to-master time-varying delays, respectively. The force τ_m allows the operator to perceive a force feedback of the interaction of the tele-operated robot with the remote environment.

The passivity layer of [12] implements the Algorithm 1 reported above for reader's convenience. Please, note that if the energies H_m and H_s are positive in both master and robot tanks, then the system is passive and can be safely connected to passive (but unknown) operator and environment.

III. A taxonomy of Cyber-Physical Attacks on Teleoperation Systems

In this section, we provide a taxonomy of cyber-physical attacks in teleoperation systems according to: (i) the *target* of the attack, (ii) the attacker's *knowledge of the system*, and (iii) the *goal* of the attack. In the following we will refer to the letters $A1, \ldots, A6$ to address specific attacks that target different subcomponents of the system represented in Figure 1.

A. Targets of the attack

Most of the cyber-physical attacks target the *communication network*. A *man in the middle* attack (see $A1$ in Figure 1) can alter the content of packets traveling from master to the robot manipulator and vice versa, if no cryptography is implemented. In particular, the attacker may change the command or the measurement values. If the packets were encrypted then the attacker may still: (i) *drop* the packets; (ii) *resend* old packets (replay attack); (iii) *delay* the packets. The last attack is rather severe in a teleoperation system because delays affect not only system performance and operability, but also the *bilateral stability*. Moreover, it is very difficult for the user to understand if this performance degradation is due to a delay attack or to a network congestion.

Another target for a cyber-physical attack is the *mechatronics system*, *i.e.*, those sensors and actuators that compose

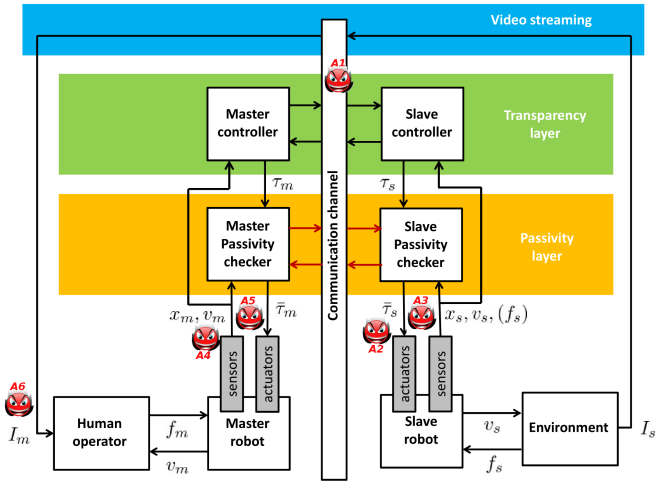


Fig. 1. Two-layers bilateral teleoperation system with potential vulnerabilities.

the teleoperation system. This type of attack is labeled in Figure 1 as $A2, A5$ if the target is an actuator and $A3, A4$ if the target is a sensor. Therefore, in a bilateral setting the attacker may target both the master and the slave, whereas in unilateral systems the adversary may only target actuators and sensors in the slave $A2, A3$. Through the actuation of the master, a bilateral architecture leverages an additional attack vector. Malicious activities targeting the master could alter the feedback reflected to the operator. In fact the haptic devices handled by the operator could have very powerful motors, and an attacker could induce sudden and abrupt movements that might harm the operator. To alter the data before they are sent to the network (measurements x_m and x_s) or after they are received (reference signals x_{md} and x_{sd}) the attacker must have a direct access to the system. This could be gained after a preparation phase during which the data flow to and from the system is analyzed via a proper malware [17].

Similar targets have been identified in papers [17], [6].

B. Attacker's knowledge

Another criterion to classify cyber-physical attacks is the attacker's knowledge of the system [5]. We consider two different scenarios: *blind attacks* and *model-aware attacks*.

In the blind attack scenario, the attacker has no insight about the system configuration and its control architecture. However, general informations on bilateral system may be enough to alter their functioning. For instance, *Denial of service (DoS) attacks* (see $A1$ in Figure 1) can be achieved by exhausting the bandwidth capacity of the communication network, as in [6]. In this case, the identification of the packets belonging to the system under attack is the only requirement.

In the model-aware attack scenario, the attacker has some knowledge of the system (plant and/or controller) and intends to exploit it for a malicious purpose. For example, knowing the actual position of the arms of the robot manipulator, the attacker could overwrite the actuator commands to damage the arms by inducing a collision. Moreover an accurate knowledge of the system may allow the attacker to operate in *stealthy* manner, *i.e.*, perpetrating the malicious actions without being detected. In the context of bilateral systems, the adversary might disable in a stealthy manner the controls that provide stability to the system which will make the system unstable when time-varying delays will occur.

C. Attacker's goal

Another way to classify the attacks is by considering the goal of the attacker:

1) *damage the robot*: the attacker may target sensors or actuators of the teleoperation system at the master or at the slave sides. By tampering with the content of the packets it is possible to induce the robot make unexpected movements and, if there is a communication delay, the operator has no time to send an alert and stop the system. The attacker can also adopt a *stealthy approach* to damage an electromechanical device. If a damaging signal is sent for a long period, superimposed to the correct commands, the system will apparently behave as usual but additional wearing effects will pass unnoticed. An example of this attack will be shown in the Experimental Section IV by adding a high frequency sinusoidal component to the actuator commands.
2) *performance degradation*: in this case the goal is to reduce the performance in terms of usability of the teleoperated system. The attack may be blind as it requires only the identification of the packets belonging traveling on the communication channel. Delays and packet dropouts are easy ways to reach the goal.
Alternatively the attacker may affect the system in such a way that the robots will move in an unsteady way instead of smoothly as expected by the operator. This will reduce the usability of the teleoperated system. For this purpose, the attacker should alter the commands that the actuator receives (attack $A2$ in Figure 1). Alternatively, the adversary could target the actuators of the master robot (attack $A4$). For instance, she could increase the stiffness of the haptic device controlled by the operator. The same goal may be achieved by attacking the *video streaming* (attack $A6$). If the attacker can delay the video stream by adding an offset, the operator will experience a disturbing time misalignment between the perceived force feedback and the video information.
3) *damage the environment*: in the context of industrial robots, the attacker might damage the system or cause injuries to operators, by changing the precision of the robot movements or by substantial alterations of safety devices. This is becoming a realistic scenario as vendors are introducing several models of collaborative robots (or cobots), able to work nearby humans (*e.g.*, ABBs YuMi, FANUCs CR-35iA) [18]. In the case of a tele-operated surgical robot even small uncontrollable or random motions of the robotic arms holding the surgical

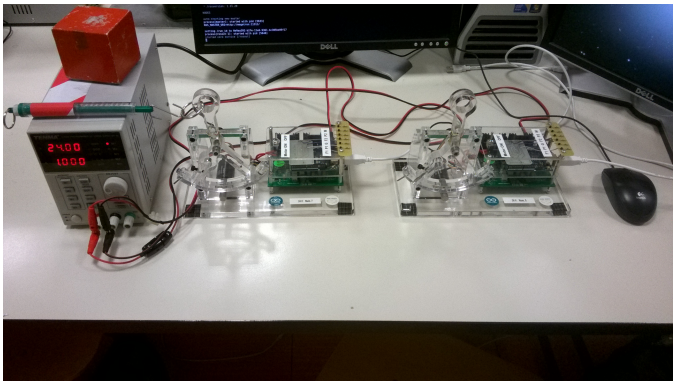


Fig. 2. Haptic Paddles at the master and robot side.

tools (*e.g.*, a scalpel) may cause damage to the internal organs of the patient.

In the context of industrial robots, production outcome altering, physical damage, production/plant halting and unauthorized accesses have been studied in [17].

IV. Experimental Results

The experimental setup is composed of two haptic paddles (see Figure 2) controlled by an *Arduino Uno rev3 board*, [19], and connected in a bilateral teleoperation fashion by ROS [20]. The haptic paddle is a partially revised version of the one developed at ETH Zürich and is described in details in [11]. It is worth recalling here that the sampling frequency is set to 100Hz and that the network is emulated by a ROS node running on a *Ubuntu 14.04 Linux machine* with low-latency kernel installed to guarantee a *soft* real-time architecture. The communication delays d_{m2s} and d_{s2m} can be constant or random (*e.g.*, Gaussian, uniform) and also packet loss can be simulated. The *two-layer algorithm* is implemented in two ROS nodes as well: one for the master side and one for the slave side, as they exchange data according to the block diagram in Figure 1.

In the following subsections we describe the implementation of some of the attacks described in Section III and their impacts on the bilateral teleoperation system. We also examine the consequences of some attacks that have been proposed in previous works on unilateral systems. In addition, we explore the effects on the operator at the master side when the slave robot is attacked. This is possible because our teleoperation system provides sensory perceptions from the remote environment to the operator via the haptic paddle.

For the sake of clarity, we divide our attacks in blind and model-aware attacks.

A. Blind Attacks

In this section, we provide a number of attacks that do not have any knowledge about the system under attack.

1) Denial of Service: The attacker needs to identify the packets of the teleoperation system and then send a large amount of packets to congest the channel. This kind of attack has been also discussed in [6] because no knowledge about the system is required to carry out the attack, and the effects on system usability can be dramatic.

Our attacker generates a time-varying delay according to:

$$d_i^A(k) = d_i^A(k-1) + r_i(k) \pm a_i, \qquad i = m2s, s2m \tag{1}$$

where $r_i(k)$ is a uniform random variable $\mathcal{U}(r_{min}, r_{max})$ and $a \in \mathbb{R}^+$. If $d_i^A(k) > N_d$ then the packet gets lost by buffer overflow in the router.

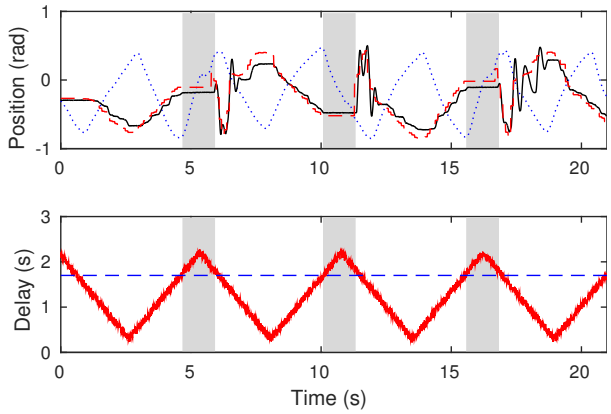


Fig. 3. DoS at the slave robot side. Legend: blue dotted line: position of the master $x_m(k)$, red dashed line: robot reference position $x_{sd}(k) = x_m(k - d_{m2s}(k))$, black solid line: robot position $x_s(k)$. Parameters: $N_d = \lceil T_d/Ts \rceil, T_d = 1.7$s, $r_{min} = 0.1$s, $r_{max} = 0.3$s, $a = 0.007$s.

Figure 3 shows an example of the impact of the DoS attack on the position error. During the DoS (gray region) packets get lost. This implies that the reference position x_{sd} is frozen to the last received value and the actual position x_s of the robot does not change. When the DoS is over, the new reference position x_{sd} is received: the tracking error $x_{sd} - x_s$ is large and therefore the motor torque τ_s is also large. This generates the oscillations that appear in the time series of x_s.

Due to the bilateral connection of master and slave, the attack at the slave side will occur at the master side as well. In addition, if the delay is not properly managed by the control architecture then unstable oscillations may occur.

2) Packet drop: Having access to the communication network, the attacker can drop packets traveling from the master to the slave robots to make the robots moving irregularly. Figure 4 shows the displacement between the master position and the robot position for two packet drop rates: the higher is the drop rate the closer is the position x_s to a step-wise signal.

Due to the derivative terms within the controllers, the packet drop can also affect the control theoretical stability of the system and force the passivity layer to intervene with a consequent change in the system behavior.

3) Delay of packets: This attack imposes an additional communication delay (D sample times) to the packets from the master to the slave robots over certain periods. The value of D can be selected according to a uniform or Gaussian distribution to mask the attack as a 'normal' congestion of the network. Figure 5 show two examples of this kind of attack for different Gaussian statistics $D \sim \mathcal{G}(\mu, \sigma^2)$.

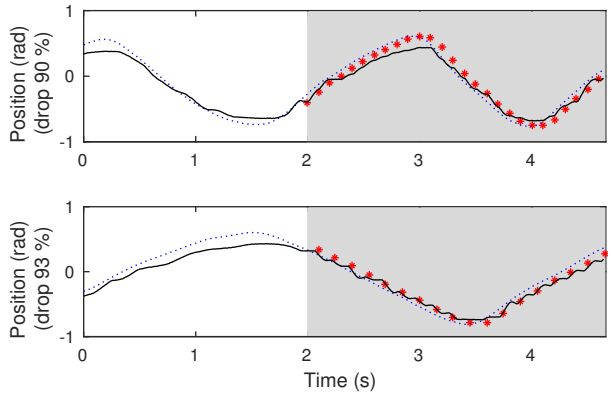


Fig. 4. Packet drop at the robotic side. Top: packet drop rate equal to 90% (1 out of 10 packets is received by the robot controller). Bottom: packet drop rate equal to 90% (1 out of 15 packets is received by the robot controller).

The attack works as follows:

- Gray regions: the attacker delays the packets by D sample times, *i.e.*, for DT_s seconds the slave robot does not receive any new packet from the master.
- Yellow regions: After DT_s seconds the robot receives the old packets for DT_s seconds whereas the proper packets are discharged.
- White regions: The correct packets are received by the robot. The discontinuity produces oscillations as can be seen in the time series of x_s in Figure 5.

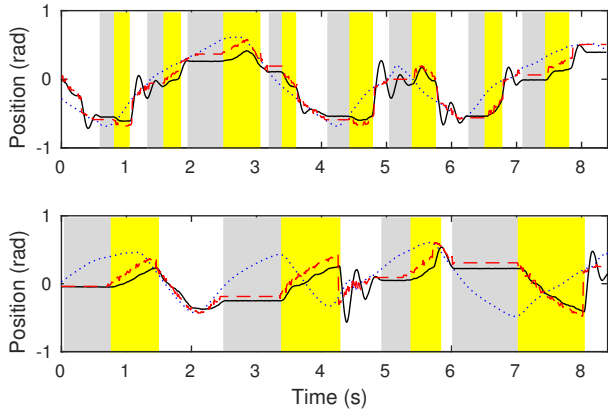


Fig. 5. Delay of packets. Legend: blue dotted line: position of the master $x_m(k)$, red dashed line: robot reference position $x_{sd}(k) = x_m(k - d_{m2s}(k) - D(k))$, black solid line: robot position $x_s(k)$. Top: $\mu = 0.3$s, $\sigma = 0.1$s; Bottom: $\mu = 0.6$s, $\sigma = 0.4$s

4) Force feedback analysis: All previous attacks affect the slave robot in a similar manner by inducing an oscillation in the robot. Moreover, if the robotic devices are connected on a shared network then the distinction between attacker's effects and normal congestions might become difficult.

Figure 6 shows the force feedback provided to the operator over the three attacks. Each attack induces sudden and abrupt movements on the slave robot. As we are dealing with a bilateral architecture and the attacks cause a misalignment between the position of the master and the slave, the tracking

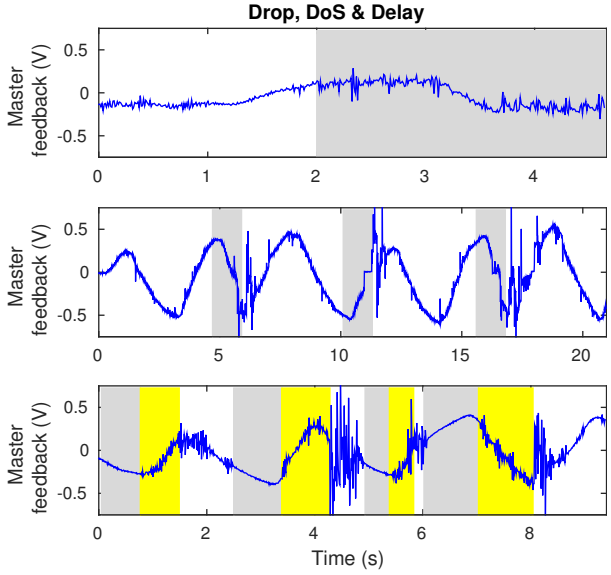


Fig. 6. Master feedback. Legend: blue line: master feedback. First plot: Drop attack (93 %). Second: DoS attack. Third: Delay attack ($\mu = 0.6$s, $\sigma = 0.4$).

error of the master controller will be affected as well. As a consequence, the operator will perceive sudden and abrupt feedbacks from the master robot that may not occur in unilateral teleoperation systems. In particular, on each time series of the force feedback, a large and abrupt peak indicates the occurrence of an attack. The higher is the peak the stronger is the impact of the attack; thus *DoS* and *Delay* attacks have a more severe impact than the *drop* attack. Note that the colored regions (gray and yellow) match the regions of the attacks given in the previous section.

Note that a bilateral architecture aims to leverage a perception to the operator as if it was in the remote environment. Hence, tampering with the force feedback might impede the operator to perform its task correctly. This is particularly relevant when the remote robot is interacting with a unknown remote environment. For instance, in a robot-surgical context the system might provide false tactile sensations to the surgeon, with obvious consequences.

B. Model-aware attacks

In this section, we assume that the attacker has some knowledge about the plant and its control architecture.

1) Tampering energy values for instability: Here, we provide a specific attack for bilateral teleoperation systems where the adversary compromises the correct functioning of the *two-layer algorithm*. We assume that the attacker has access to: (i) the position and reference data that both master and robot send through the network, and (ii) the energy quanta H_+ exchanged between master and robot tanks. Other teleoperation algorithms (*e.g.*, [16]) would require different knowledge.

We recall that the passivity layer detects and prevents unstable behavior of the master and the slave robots by comparing the available energy within the tanks, and the amount of energy needed to perform the actual actions. If the attacker would be

able to modify the amount of energy that arrives from the master then the slave robot can be induced to believe that there is plenty of energy available. As a consequence, the slave robot might move in an unstable way without the protection of the passivity layer. If the communication delay is large, the operator will see the wrong motion on the video streaming when is too late to react.

Figure 7(a) shows the normal working condition of the two-layer algorithm when the energy at the robot side is below a threshold (parameter selected by the designer according to the particular application); the passivity layer modulates the commands and asks for energy at the master side.

When the system is under attack (gray region), fake energy quanta H_+ are sent to the robot tank as shown in Figure 7(b). The level of the energy is always at the maximum (0.1J in this case). The position x_s exhibits some oscillatory behavior that the passivity layer does not counter-act because the checks on τ_{max1} and τ_{max2} are always satisfied (line 2 in Algorithm 1). This attack could be implemented at different points: (i) on the master controller which could send more energy than requested; (ii) on the communication channel, as the attacker could tamper with the packets and alter the energy quanta sent to the slave; (iii) on the slave controller.

Another possibility is to increase the amount of energy required by the slave robot. In order to extract energy from the operator, the master controller activates a software damper. In this case the operator is feeling a damping effect that is not real and that could affect its interpretation of the interaction of the slave robot with the environment.

2) Small vibration on the command: This attack superimposes a high-frequency sinusoidal term to the commands sent to the motors of the slave robots [21]. Since the mechanical subsystem behaves as a low pass filter, there is no observable effects on the motion of the robot. Nevertheless, the superimposition of this term will stress the electro-mechanical subsystem that eventually will be damaged. This additive term can be introduced directly on the command if the attacker has access to the input of the high-voltage amplifier that powers the motors:

$$\tau_s^A(k) = \tau_s(k) + A_\tau \sin(2\pi F T_s k) \tag{2}$$

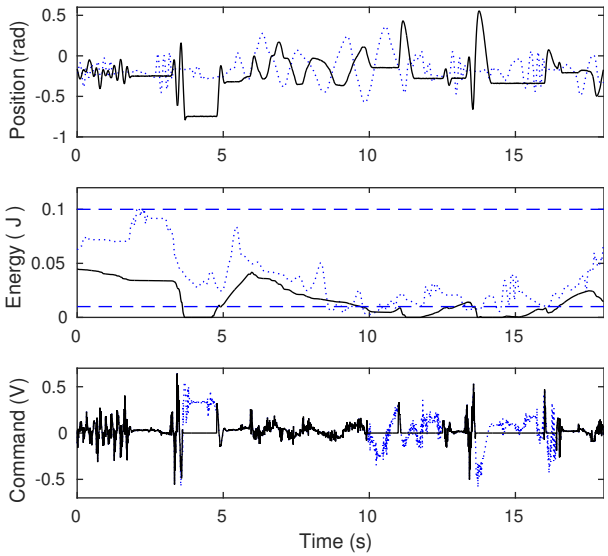
or by modifying the reference position accordingly:

$$\begin{aligned} \tau_s^A(k) &= k_P(x_{sd}^A(k) - x_s(k)) + k_D(v_{sd}(k) - v_s(k)) \\ &= k_P(x_{sd}(k) + A_x \sin(2\pi F T_s k) - x_s(k)) + \\ &\quad + k_D(v_{sd}(k) - v_s(k)) \\ &= \tau_s(k) + k_P A_x \sin(2\pi F T_s k) \end{aligned}$$

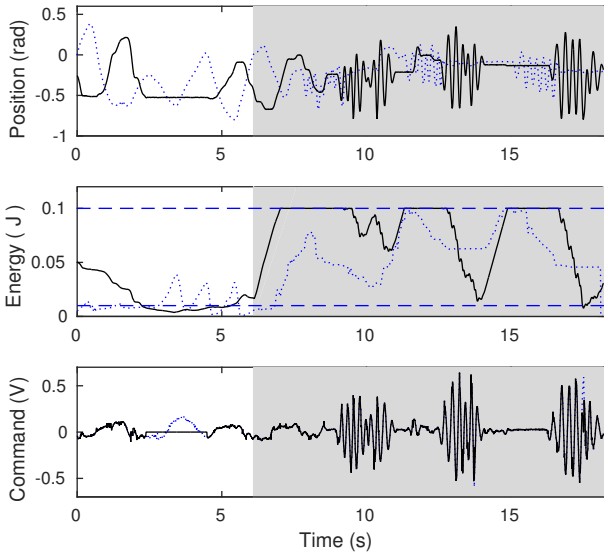
Note that an actuated master could also be affected by this attack. Figure 8 shows the total energy with and without the superposition of the sinusoidal term. The heating due to the extra energy will damage the motors in the long run.

V. Conclusions, Related and Future Work

In this paper, we have classified cyber-physical attacks in bilateral teleoperation systems in terms of target components,



(a) normal two-layer algorithm



(b) attack on the energy tank at the slave robot side

Fig. 7. Tampering energy values for instability. (a)/(b) Top: dotted blue line: x_d, solid black line: x_s. Middle: dotted blue line: energy level of the master tank, solid black line: energy level of the robot tank. Bottom: dotted blue line: command τ_s, solid black line: modulated command $\bar{\tau}_s$. Gray region in (b): attack period. Round trip time delay $d_{m2s} + d_{s2d} \sim \mathcal{G}(1, 0.1)$.

knowledege of the system, and goals. This setup allows us to study the impact of our attacks, and represents a first step towards a complete threat model with corresponding counter-measures in more complex bilateral teleoperation systems. We have analyzed the impact of different kinds of cyber-physical attacks carried out on a experimental setup of a single-input single-output bilateral teleoperation system with a two-layer control architecture for force feedback. Some of these attacks have already been studied in a telesurgery setting [6] and implemented in a unilateral teleoperation system. We have shown the effects of these malicious activities on a bilateral teleoperation system. In particular, we have shown how the

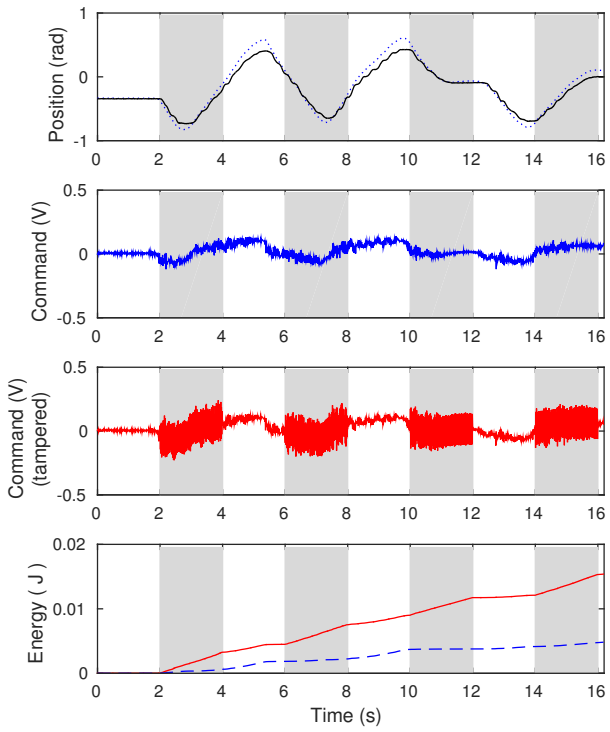


Fig. 8. Vibration on the robot commands. First plot: dotted blue line: x_d, solid black line: x_s. Second: command computed by the transparency layer τ_s. Third: Command modified by the attacker τ_s^A ($A_\tau = 0.12$V). Fourth: Dissipated energy on the robot motor: Dashed blue line: without attack, Solid red line: with attack. Round trip time delay $d_{m2s} + d_{s2d} \sim \mathcal{G}(0.05, 0.01)$.

force feedback transmitted to the master haptic device from the slave, is affected during the attacks. Then, we have provided a model-aware attack that may affect only bilateral architectures by compromising the controller that provides stability to the systems. Furthermore, we have shown a malicious tampering of the controller that does not trigger any immediate effect on neither the cyber nor the physical part.

Related work: A number of security issues of industrial robot controllers have been addressed in [17]. The authors have analyzed both system-specific attacks and the minimal set of security requirements that industrial robots should guarantee. Furthermore, they have proposed a domain-specific attacker model and conducted an experimental security assessment on a de facto-standard robot. Exploitation of software vulnerabilities are presented which can compromise completely and remotely the robot-controlling computers. Although this work does not address directly the security issues of teleoperation systems, it analyzes attacks on a complete and realistic deployment. Thus, the same security flaws might affect the controlled robot of a teleoperation system.

As said in the Introduction, the security of teleoperation systems has been addressed in papers [6], [7], [8], [9] in the context of the Raven II laboratory surgical robot [10]. In [6] the authors analyze *Denial-of-Service* (DoS) attacks, define *performance metrics* to evaluate the *attack impact* and also propose *defensive strategies* to mitigate the attack effects. In [7] the authors approach the attacks from a different perspective: they assume the attacker has some knowledge of robot inner workings, and is able to exploit this knowledge by injecting faults at the worst possible moments, either for the robots or the surgery as such. The attack is carried out by changing one valid control command to another valid (malicious) command, without violating any protocol syntax. The same authors have proposed [8] a general principle for detecting cyber-physical attacks that combines the knowledge of simulations from both cyber and physical domain process to estimate the adverse consequences of malicious activities in a timely manner. The attacks alter valid commands to impact the physical processes without affecting the cyber domain. In [9], the proposed attacks exploit the communication between the master and the surgical robot based on a network hub and the Interoperable Teleoperation Protocol (ITP).

Another important aspect of the security of cyber-physical systems regards their verification and validation using formal methodologies rather than simulation-test systems [22]. In this respect, we recently proposed formal languages to express and reason on CPSs and cyber-physical attacks [23]. The goal of that paper is to lay and streamline theoretical foundations to reason about, and statically detect, attacks to physical devices. In fact, the complexity of CPSs call for (semi-)formal verification tools to provide both an exhaustive search of all possibles behaviours of systems under attacks, and exact, rather than approximate, quantitative results. A rare example of formal security analysis of CPSs can ben found in [24].

Future work: Although ours and the aforementioned works have focused on experimental setups, we have learned from [17] that a robotic controller of a de-facto standard robot in a industrial context can be completely and remotely compromised. The final goal of our on going efforts is the development of countermeasures along the lines of [25], [26], [27]. For instance, Cómbita et al. [25] and Zhu and Basar [26] applied *game theory* to capture the conflict of goals between an attacker trying to maximize the damage inflicted to a CPS and a defender which aims to minimize it [27]. In the context of teleoperation, some preliminary results can be found in [28]. We aim to extend these results in the context of bilateral teleoperation systems.

VI. Acknowledgements

This work has been partially supported by the project of the Italian Ministry of Education, Universities and Research (MIUR) “Dipartimenti di Eccellenza 2018-2022”. We thank the anonymous reviewers for valuable comments.

References

[1] N. Falliere, L. Murchu, and E. Chien, “W32.Stuxnet Dossier,” 2011.
[2] J. Slay and M. Miller, “Lessons Learned from the Maroochy Water Breach,” in *Critical Infrastructure Protection*, ser. IFIP 253. Springer, 2007, pp. 73–82.
[3] Nccic/ics-cert year in review 2015, https://ics-cert.us-cert.gov/sites/default/files/Annual Reports/Year in Review FY2015 Final S508C.pdf.
[4] S. Sicari, A. Rizzardi, L. Grieco, and A. Coen-Porisini, “Security, privacy and trust in Internet of Things: The road ahead,” *Computer Networks*, vol. 76, pp. 146–164, 2015.

[5] D. Gollmann, P. Gurikov, A. Isakov, M. Krotofil, J. Larsen, and A. Winnicki, "Cyber-Physical Systems Security: Experimental Analysis of a Vinyl Acetate Monomer Plant," in *ACM CPSS*. ACM, 2015, pp. 1–12.

[6] T. Bonaci, J. Yan, J. Herron, T. Kohno, and H. J. Chizeck, "Experimental analysis of denial-of-service attacks on teleoperated robotic systems," in *ACM/IEEE ICCPS*. ACM, 2015, pp. 11–20.

[7] H. Alemzadeh, D. Chen, X. Li, T. Kesavadas, Z. T. Kalbarczyk, and R. K. Iyer, "Targeted attacks on teleoperated surgical robots: Dynamic model-based detection and mitigation," in *Proceedings of the 46th IEEE/IFIP International Conference on Dependable Systems and Networks*, 2015.

[8] H. Lin, H. Alemzadeh, D. Chen, Z. Kalbarczyk, and R. K. Iyer, "Safety-critical cyber-physical attacks: Analysis, detection, and mitigation," in *Proceedings of the Symposium and Bootcamp on the Science of Security*, ser. HotSos '16. ACM, 2016, pp. 82–89.

[9] T. Bonaci, J. Herron, T. Yusuf, J. Yan, T. Kohno, and H. J. Chizeck, "To make a robot secure: an experimental analysis of cyber security threats against teleoperated surgical robots," *arXiv preprint arXiv:1504.04339*, 2015.

[10] B. Hannaford, J. Rosen, D. W. Friedman, H. King, P. Roan, L. Cheng, D. Glozman, J. Ma, S. N. Kosari, and L. White, "Raven-ii: an open platform for surgical robotics research," *IEEE Transactions on Biomedical Engineering*, vol. 60, no. 4, pp. 954–959, 2013.

[11] C. Tadiello, G. De Rossi, M. Capiluppi, R. Muradore, and P. Fiorini, "Teaching physical human-robot interaction to computer science undergraduate students," in *Control Conference (ECC), 2016 European*. IEEE, 2016, pp. 376–381.

[12] M. Franken, S. Stramigioli, S. Misra, C. Secchi, and A. Macchelli, "Bilateral telemanipulation with time delays: A two-layer approach combining passivity and transparency," *IEEE Transactions on Robotics*, vol. 27, no. 4, pp. 741–756, 2011.

[13] M. Arcak, "L_2-gain and passivity techniques in nonlinear control: Arjan van der schaft: Springer, london, 2000, ISBN 1-85233-073-2," *Automatica*, vol. 39, no. 6, pp. 1118–1119, 2003. [Online]. Available: https://doi.org/10.1016/S0005-1098(03)00057-8

[14] P. F. Hokayem and M. W. Spong, "Bilateral teleoperation: An historical survey," *Automatica*, vol. 42, no. 12, pp. 2035–2057, 2006.

[15] E. Nuño, L. Basañez, and R. Ortega, "Passivity-based control for bilateral teleoperation: A tutorial," *Automatica*, vol. 47, no. 3, pp. 485–495, 2011.

[16] D. Lee and K. Huang, "Passive-set-position-modulation framework for interactive robotic systems." *IEEE Transactions on Robotics*, vol. 26, no. 2, pp. 354–369, 2010.

[17] D. Quarta, M. Pogliani, M. Polino, F. Maggi, A. M. Zanchettin, and S. Zanero, "An experimental security analysis of an industrial robot controller," in *2017 IEEE Symposium on Security and Privacy, SP 2017, San Jose, CA, USA, May 22-26, 2017*, 2017, pp. 268–286.

[18] J. R. Hagerty, "New robots designed to be more agile and work next to humans: Abb introduces the yumi robot at a trade fair in germany," *Wall Street Journal*, vol. 13, 2015.

[19] Arduino Mega, https://www.arduino.cc/en/Main/arduinoBoardMega.

[20] Robotic Operating System (ROS), http://www.ros.org/.

[21] R. Muradore and D. Quaglia, "Energy-efficient intrusion detection and mitigation for networked control systems security," *IEEE Transactions on Industrial Informatics*, vol. 11, no. 3, pp. 830–840, 2015.

[22] R. Lanotte and M. Merro, "A semantic theory of the Internet of Things," *Information and Compututation*, vol. 259, pp. 72–101, 2018.

[23] R. Lanotte, M. Merro, R. Muradore, and L. Viganò, "A formal approach to cyber-physical attacks," in *IEEE CSF*. IEEE Computer Society, 2017, pp. 436–450.

[24] R. Lanotte, M. Merro, and A. Munteanu, "A Modest Security Analysis of Cyber-Physical Systems: A Case Study," submitted for publication.

[25] L. F. Cómbita, J. Giraldo, A. A. Cárdenas, and N. Quijano, "Response and reconfiguration of cyber-physical control systems: A survey," in *Automatic Control 2015*. IEEE, 2015, pp. 1–6.

[26] Q. Zhu and T. Basar, "Game-theoretic methods for robustness, security, and resilience of cyberphysical control systems: games-in-games principle for optimal cross-layer resilient control systems," *IEEE Control Systems*, vol. 35, no. 1, pp. 46–65, 2015.

[27] M. Manshaei, Q. Zhu, T. Alpcan, T. Basar, and J.-P. Hubaux, "Game theory meets network security and privacy," *ACM Comput. Surv.*, vol. 45, no. 3, p. 25, 2013.

[28] H. Lin, H. Alemzadeh, D. Chen, Z. Kalbarczyk, and R. K. Iyer, "Safety-critical cyber-physical attacks: analysis, detection, and mitigation," in *Proceedings of the Symposium and Bootcamp on the Science of Security*. ACM, 2016, pp. 82–89.