Sensory Channel Threats to Cyber Physical Systems: A Wake-up Call

A. Selcuk Uluagac*, Venkatachalam Subramanian†, and Raheem Beyah†
*Electrical & Computer Engineering Department
Florida International University
Miami, FL 33173, USA
email: suluagac@fiu.edu

†GT CAP Group, The School of ECE
Georgia Institute of Technology
Atlanta, GA 30332, USA
emails:{venkat.subbu, rbeyah}@gatech.edu

***Abstract*—Cyber-Physical Systems (CPS) is a relatively novel computing paradigm where there is a tight integration of communications, computation, and the physical environment. An important component of the CPS devices is the sensors they use to interact with each other and the physical world around them. With CPS applications, engineers monitor the structural health of highways and bridges, farmers check the health of their crops, and ecologists observe wildlife in their natural habitat. Nonetheless, current security models consider protecting only networking components of the CPS devices utilizing traditional security mechanisms (e.g., an intrusion detection system for the data that traverse the network protocol stacks). The protection mechanisms are not sufficient to protect CPS devices from threats emanating from *sensory channels*. Using sensory channels (e.g., light, temperature, infrared), an adversary can successfully attack systems. Specifically, the adversary can (1) trigger existing malware, (2) transfer malware, or (3) combine malicious use of different sensory channels to increase the impact of the attack on CPS devices. In this work, we focus on these novel sensory channel threats to CPS devices and applications. We first note how sensory channel threats are an emerging area for the CPS world. Then, we analyze the performance various sensory channel threats. Moreover, using an iRobot Create as our CPS platform, we exploit simple vulnerable programs on iRobot through its infrared channel. Finally, we introduce the design of a novel sensory channel aware intrusion detection system as a protection mechanism against the sensory channel threats for CPS devices.**

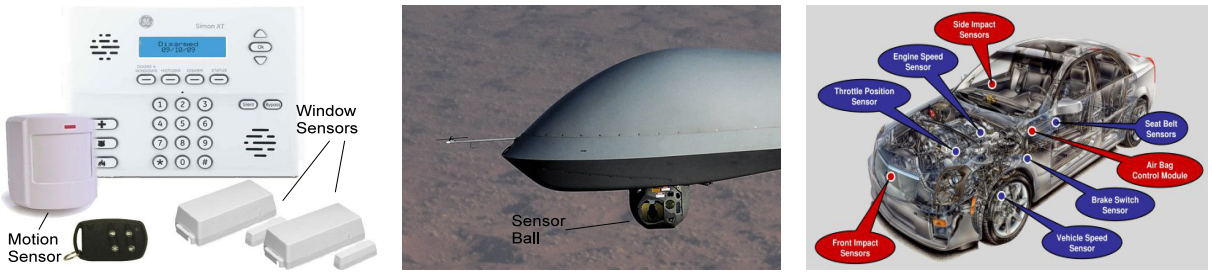
***Index Terms*—Sensory-channel threats, Cyber-Physical Systems, Sensory Attacks, CPS Security**

I. Introduction

Cyber-Physical Systems (CPS) consist of large-scale interconnected systems of heterogeneous components (e.g., sensors, actuators) interacting with their physical environments [1]. In CPS applications, humans and/or smart networked devices interact with and control the physical world around them through these sensors. Given their low cost and multiple functionalities, it is possible to see the utilization of sensors in different CPS settings. For instance, today sensors are integrated into smart phones [2], tablets, and cars and are used in many diverse application domains such as home security, health care, military, and environment monitoring. Figure 1(a) shows an example home security system (GE Simon XTi [3]) with different motion and window sensors. Unmanned aerial vehicles (UAVs) (Figure 1(b)) navigate via sensor balls and armored suits used by the military also depend on a number of different environment-monitoring sensors (e.g., optical, acoustic, seismic, and temperature) [4]. In a modern car (Figure 1(c)), there are more than 400 sensors that are accessible for programming [5], [6]. Similarly, robotic systems, which are used in homes (e.g., *Roomba* [7], [8]) and hospitals (e.g., *Da Vinci Surgery Robot* [9]), incorporate a significant number of sensors for their functionalities. In addition, with other similar recent initiatives such as the Internet of Things [10] and Planetary Skin [11], sensor-based CPS applications have gained new momentum in the research community and industry and are predicted to be one of the ten technologies that will change the world in the next 10 years [12]. Given the popularity of sensory devices in the CPS world, securing them against possible malicious activities is of utmost importance.

Recently, there have been a few attempts to exploit different host devices facilitating their sensing components. In [13], password keystrokes are extracted from ambient user activities via an accelerometer in smartphones. Similarly, graphical password patterns are decoded from the accelerometer recordings inside smartphones in [14]. In [15], authors show how a car tire pressure sensing system, which utilizes Radio Frequency (RF)-based wireless motes, could be exploited. Additionally, in [16] analog-based signal injection attacks are realized on cardiac medical devices and microphones via intentionally generated electromagnetic interference. In these works, sensors are utilized as auxiliary components for another malicious goal and abused in ways different from their originally intended uses because the host devices do not have any security mechanisms against such threats by default. *Unlike these works, we note that it is also possible to exploit sensor-based CPS applications and devices directly via their sensory components.* For instance, a light sensor normally activated by a certain illuminance value can easily be tricked by false input from a powerful flashlight. In fact, currently CPS security is limited to protecting the CPS components networked via traditional means (e.g., RF) or services on the host devices. In other words, securing a networked CPS device means utilizing the same tools and security mechanisms developed for the RF world. However, sensory components in CPS devices form *sensory channels* that serve as external interfaces to their host systems. Since a significant number of critical functionalities (Figure 2) in the CPS realm are realized interacting with the real world through these sensory channels, securing the sensory

978-1-4799-5890-0/14/$31.00 ©2014 IEEE



(a) (b) (c)

Fig. 1. (a) GE Home security system [3]; (b) Predator drone; (c) Sensors in a car [6].

channels is as vital as securing other components of CPS devices.

Most recently, certain sensory channels in the realm of wireless sensor motes (e.g., MicaZ, Telosb) were analyzed and their feasibility for supporting malicious activities were determined in our earlier work [17]. In this paper, we extend our earlier work that introduce the general idea of sensory channel attacks. We discuss threats for sensory channels of CPS devices and analyze the performance of the sensory threats. Moreover, we realize simple attack scenarios on the iRobot Create robotic platform [7] using its infrared light channel. Finally, we introduce the design of a novel sensory channel aware intrusion detection system as a protection mechanism against the sensory channel threats for CPS devices.

The remainder of the paper is organized as follows: we discuss the related work in Section II. In Section III, an overview of CPS sensory channels is presented. In Section IV, we evaluate the sensory channel threats. Simple exploitation of vulnerable programs through the infrared channel are demonstrated in Section V. In Section VI, we discuss why conventional security mechanisms would not be able to protect against sensory channel threats. The design of sensory channel aware IDS for CPS is introduced in Section VII. Finally, the conclusion and future work are described in Section VIII.

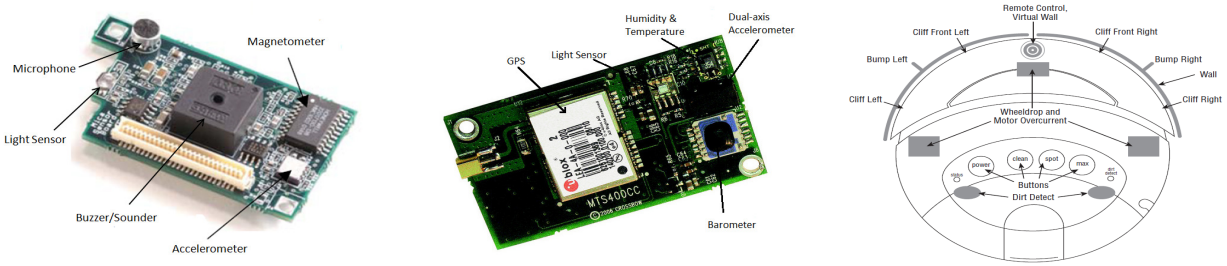
II. Related Work

Various attack scenarios and exploits have been designed and discussed for different sensor-dependent devices. In [13], the authors demonstrated that unfettered access to a smartphone's accelerometer data allows a malicious application to recover and decode the vibrations caused by keystrokes on a nearby keyboard. This work passively listens to the seismic channel to gather sensitive information about nearby external activities. Similarly, in [14], it was demonstrated that the accelerometer sensor could function as a side channel to learn user tap- and gesture-based input generated by the user when unlocking smartphones with a PIN/password or graphical password patterns. Furthermore, in [18], the authors developed *TouchLogger*, an Android application that extracts features from device orientation data to infer keystrokes. This work shows the usage of sensory side channels to infer keystrokes, but from within the same device. These aforementioned contributions utilized sensory components to either passively gather sensitive information from external user activities or decode secure credentials such as passwords on smartphones through the data collected via the sensors within the host device. In [15] authors analyze the security of a tire pressure monitoring system in automobiles, which utilizes battery-powered pressure sensors inside each tire. These sensors measures the tire pressure and communicates the measurement data via an RF transmitter to the pressure control unit. In another similar work [19], authors visit the same topic and discuss (more comprehensively) different threat models for cars. Although these studies are very useful to show the importance of the sensory devices, they primarily use the RF as the medium for the communication and the sensors are mainly utilized as an auxiliary component to achieve another task.

In [16], authors show that analog sensors are susceptible to signal injection attacks. They analyze the feasibility of these attacks with intentionally generated electromagnetic interference on cardiac medical devices and microphones. Although this work is remotely similar in its nature to our work, we are fundamentally different from this work in several ways. Unlike this work, we do not focus on analog signal based attacks simply jamming the operational frequencies of the devices. Rather, we focus on various *sensory channel* threats, analyze them in a more comprehensive manner considering multitude of sensory channels in the realm of CPS, and introduce a novel IDS-based solution for the sensory channel threats. In our earlier work [17], we analyzed certain sensory channels and discussed their feasibility for supporting malicious activities from outside the host device in the realm of wireless sensor networks. Different from this work, in this paper, we focus on exploiting simple vulnerable services on the iRobot Create robotic platform via its infrared light channel. Further, we treat the sensory channel threats in the realm of CPS and we introduce the design of a novel IDS for CPS as a remedy for sensory channel threats.

III. CPS Sensory Channels

In this section, we provide an overview of the sensory channels that are utilized by different CPS devices. We note that in realistic cases, a CPS device can have n different sensory channels where $n \geq 1$. In this work, we primarily focus on a subset of these channels, i, where $i \geq 1, 2, 3, .., n$. Specifically, we have $i = 4$ different representative sensory channels: *light, infrared, acoustic*, and the *seismic* channels. These sensors are widely utilized by popular systems such as iRobot Create, smartphones, and wireless sensor motes (e.g., MicaZ [21], Telosb [22]). These sensors depicted in different



(a) (b) (c)

Fig. 2. (a) MTS310CB sensing board [20]; (b) MTS420CC sensing board [20]; (c) iRobot Create/Roomba sensors [7].

platforms are shown in Figure 2 and their corresponding channels are briefly explained below.

A. Light Channel

The light channel is a widely used sensory channel in sensor systems and is utilized in various applications including automatic brightness control (i.e., control display brightness with respect to ambient lighting for power management) for LCDs [23]. Also, certain military applications are based on light detection and ranging (LIDAR) such as the *Airborne Laser Mine Detection Systems* [24]. Moreover, most wireless sensor boards (MTS310 [20] (Figure 2(a)), MTS400 [20] (Figure 2(b))) are equipped with a photosensor. Beyond the basic utilization of light sensors for ambient light measurements, they can be used as side channels to complement existing security solutions. For instance, in KeyLED [25] LEDs and photosensors and in Enlighten Me! [26] light sensors are established to transfer sensitive information and to distribute security keys.

B. Seismic Channel

The seismic channel is another sensory channel used in many sensory systems, predominantly in smartphones to enhance the user experience. Specifically, accelerometers are used to present landscape or portrait views of the device's screen in most of the smartphones and almost all mobile phones provide a *vibration mode*. Furthermore, popular game consoles such as *PlayStation 3* [27] and *Nintendo* [28] also provide an interface for the seismic channel through the three-axis accelerometers contained in their *DualShock 3* and *Wii* remote controllers, respectively. Different accelerometers are shown in Figure 2. Accelerometers can also be used in various devices to understand the user's physical interactions with the device in order to improve the overall user experience. For instance, in [29], new input mechanisms are presented in lieu of traditional keyboards, mice, and touch screens. Specifically, a single key with an accelerometer assisted positioning system is utilized to emulate the functions of keyboards and mice. Moreover, applications such as Bump [30] utilize the accelerometers in smartphones to authenticate two devices bumped against each other prior to file sharing. Similarly, a user identification mechanism is demonstrated in [31] where users are identified by their mobile devices based on their physical activities such as walking.

C. Acoustic Channel

The acoustic sensory channel is utilized by many robotic systems and military applications for obstacle avoidance, navigation, and map building. The primary method of communication in acoustic sensory channels is to emit a short burst of sound wave and measure the physical characteristics (e.g., speed, amplitude) of the sound wave reflected from obstacles via transducers. This is possible because when a sound wave is sent through a medium, it is affected by the physical properties of the medium through which it travels. For instance, using the speed of the medium that the sound wave travels through, the host devices (e.g., Sonar) determines the distance or existence of the obstacles. Because it is relatively easy to monitor the physical characteristics of acoustic waves, acoustic channels are widely utilized in military and industrial settings. For instance, it is possible to recognize different vehicles (e.g., automobiles, aircraft, trucks, missiles) based on their infrasonic signatures (0 to 20Hz) [32].

D. Infrared Channel

Similar to the light channel, the infrared channel (IR) is also a widely used sensory channel. A number of remote controller systems utilize the infrared channel. Many robotic systems (e.g., iRobot Create [7] (Figure 2(c)), Turtlebot [33]) use the infrared channel for navigation assistance and obstacle avoidance. Similarly, Microsoft Kinect [34] uses infrared light beams to create a 3D map of a room. An interesting application of IR sensory channel is that since it is not visible to the human eye unlike the visible light channel, it can support a covert side channel. Note that the IR sensory channel is further discussed in Section V to demonstrate how it can be utilized simply to trigger simple vulnerable services in iRobot Create.

IV. Evaluation of Sensory Channel Threats

We primarily envision three different ways to perpetrate malicious activities on sensory channels. Using the sensory channels, an adversary can (1) *trigger existing malware*, (2) *transfer malware*, or (3) improve the performance of the threats (1) and (2). In this section, we evaluate these.

A. Triggering existing malware

In this scenario, the adversary triggers a malicious program existing in the host CPS system where the sensor resides. The malicious program is assumed to be loaded into the system's hardware or software without the knowledge of its owner [35], [36]. The malicious program is activated by a specific value

TABLE I
OBSERVED DATA RATE ON SENSORY CHANNELS

Sensory Channel	Platform	Sensor Component	Observed Maximum Sampling Rate (bps)
Light	Telosb	Hamamatsu S1087	85-100
	MicaZ (MTS400CC)	TAOS 2115	2-3
	MicaZ (MTS310CB)	CdSe Photocell	50-65
Acoustic	MicaZ (MTS310CB)	LM567 CMOS Tone Detector	2-3
Seismic	MicaZ (MTS310CB)	ADXL202JE Accelerometer	50-65
Infrared	iRobot Create	LITEON SEN-00241	30-45

or sensory pattern received over the sensory channels. For instance, a malicious program can be triggered over an accelerometer to capture videos, pictures surreptitiously.

To better understand how practical this method is we performed simple experiments to evaluate the raw data rate (which is influenced by the sampling rate) supported by various sensory channels [17]. Specifically, we measured the maximum sampling rate supported by six sensors on their corresponding platforms and tabulated the results in Table I. For the light channel, we used three different platforms: MicaZ wireless mote [21] with MTS400CC sensor board [20], which uses the TAOS TSL2550D ambient light sensor; the MicaZ wireless mote with MTS310CB sensor board [20], which uses CdSe photocell; and a Telosb [22] wireless mote, which uses the Hamamatsu S1087 visible light sensor. For the acoustic channel, we used the MicaZ wireless mote with MTS310CB sensor board [20], which uses the LM567 CMOS Tone Detector. For the Infrared channel, we used the infrared sensor used in iRobot Create (Liteon Sen-00241) and for the seismic channel, we used the MicaZ wireless mote with MTS310CB sensor board. The results in Table I indicate that the maximum rate that can be supported with all the sensors tested is around 85-100 bps and it is observed over the light channel on the Telosb platform. For the acoustic channel, the rate is about 2-3 bps. For the seismic channel, the maximum rate is between 50-65 bps and it is observed in the MTS310CB sensor platform on MicaZ wireless motes. Finally, for the infrared channel, the iRobot Create can support rates up to 30-45 bps. A sensory channel with a lower rate can easily support the malware triggering activities while a higher rate sensory channel can support further advanced malicious activities like transferring a malware as discussed in the following sub-sections.

B. Transferring malware

An adversary can also utilize sensory channels to deliver a certain piece of malware.

In order to realistically estimate this method with the sensors analyzed in the previous sub-section, we evaluated the performance of the sensory channels while transmitting malware. Using the rates presented in Table I in the previous subsection, we measured the time, ψ, it takes to deliver a certain size malware sample with

$$\psi = \frac{\eta}{\rho} \tag{1}$$

where η is the malware's size and ρ denotes the rate of the sensory channel (Table I). The results are presented in Figure 3. As expected, the infrared sensory channel with the rate of 30 bps exhibits the lowest performance as it takes about 250 secs to transfer malware of size 1 KB. In the best case, the same malware requires a transmission time of 80 secs on the light sensory channel with the rate of 100 bps, which is a significant improvement. With future improvements in sensor technologies in CPS devices, we envision that the rates that would be supported on sensory channels would significantly improve over time.

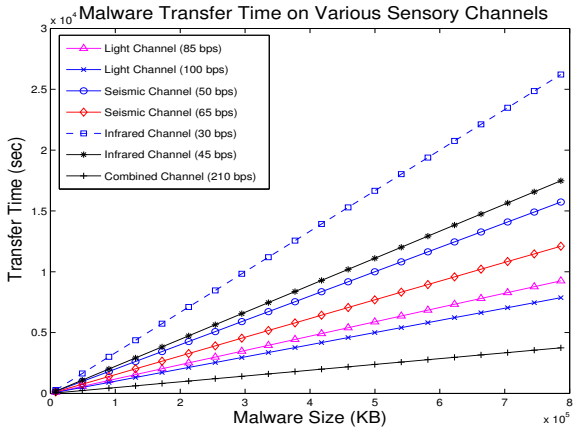


Fig. 3. Malware transfer time on various sensory channels

Another important factor that impacts the quality of the sensory channel and the data rate, thereof, is the physical properties of the channel. For instance, if the noise (e.g., ambient noise) is high or if there are other physical channel specific conditions (e.g., distance, path loss), then the quality of the sensory channel can further decrease. Physical properties of the sensory channels is outside the scope of this paper.

C. Combining sensory channels

Today most of the CPS devices are manufactured with more than one sensor. For instance, as shown in Figure 1(a), a home security system may include several sensors (e.g., window, motion sensors). Similarly, a wireless mote sensing board like MTS400CC shown in Figure 2(b) includes four sensors, including a dual-axis accelerometer, a temperature sensor, a barometer, and a light sensor.

Hence, a plausible and a more complicated possible scenario we envision is the combination of more than one sensory channel to increase the impact of one channel. In this case, an adversary can combine the sensory channels to increase the effective rate that can be achieved while delivering malware. If a CPS device has n sensory channels,

$$\Psi = \sum_{i=1}^{n} \psi_i \tag{2}$$

a combined rate from all the sensory channels would decrease the time that is required to secretly deliver the malware. For

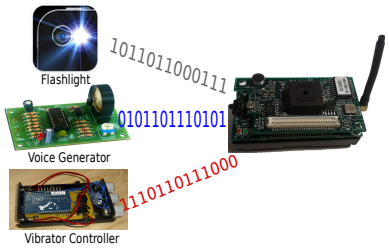


Fig. 4. Combining various sensory channels in CPS applications

instance, as illustrated in Figure 4, seismic, acoustic, and light sensory channels can be combined to increase the impact of these individual channels. As presented in Figure 3, a combined channel is able to transfer a malware of size 1KB in less than 40 secs.

V. Exploiting the Infrared Sensory Channel on iRobot Create

In this section, we present how we exploited simple vulnerable services in iRobot Create [7] through its infrared sensory channel. Specifically, we first created two simple exemplary vulnerable services that misuse timer registers on the iRobot Create platform and then triggered these through the infrared channel with two simple attack scenarios.

A. iRobot Create

The iRobot Create [7] is a robotic platform used in a number of robotic research projects while its variant Roomba is used widely in household applications. Our rationale for choosing iRobot Create as our platform to realize the sensory channel threats does not solely depend on its popularity in both academic and commercial circles. The iRobot Create provides 25 sensors (12 external, 13 internal) onboard and the flexibility to add more external ones as needed. Primarily, the iRobot Create utilizes the sensors to facilitate its navigation. For instance, it uses its mechanical *bump sensors*, *infrared wall sensors*, *infrared cliff detectors*, and *wheel-drop sensors* to detect obstacles. Some of the available sensors in the iRobot Create platform are shown in Figure 2(c). Within the scope of our work, we only focused on the infrared sensors onboard the iRobot Create to demonstrate the first sensory channel threat discussed in Section IV. These infrared sensors are located on the front bumper as cliff, bump, and remote control sensors (Figure 2(c)).

B. Design of simple vulnerable services

Like many other robotic systems, iRobot Create utilizes timer registers provided by the ATmega168 [37] microcontroller for a variety of functions including basic delay-based functions and other counter functionalities. Unauthorized manipulation of the ATmega168 timer registers forms the basis of our design for the simple vulnerable services in iRobot Create as explained below.

As listed in Table II, three different timers are supported by ATmega168. The timers TCNT0 and TCNT2 are 8-bit registers while TCNT1 is the 16-bit register. Note that these timers and their interrupt service routines (ISRs) can be manipulated

TABLE II
Timers and ISRs provided by ATmega168 in iRobot Create

Timer	Source	Interrupt Definition
TCNT0	Timer0 CompA	Timer/Counter0 compare match A
	Timer0 CompB	Timer/Counter0 compare match B
	Timer0 OVF	Timer/Counter0 overflow
TCNT1	Timer1 CompA	Timer/Counter1 compare match A
	Timer1 CompB	Timer/Counter1 compare match B
	Timer1 OVF	Timer/Counter1 overflow
	Timer1 CAPT	Timer/Counter1 capture event
TCNT2	Timer2 CompA	Timer/Counter2 compare match A
	Timer2 CompB	Timer/Counter2 compare match B
	Timer2 OVF	Timer/Counter2 overflow

easily through software. Additionally, on iRobot Create, two different program modules running as a single piece of software image or executable can access each others' resources. An adversary in the first program module can use (misuse) the timer registers and the corresponding ISR of the second program module. In this way, an adversary can indirectly access an access-restricted program module by modifying the value stored in the timer register of the program module.

Another interesting note is that the ATmega168 microcontroller uses a *Timer/Counter Overflow Flag (TOV0)* to signal an overflow. The counter overruns when it passes its maximum 8-bit value and then restarts from the bottom. In its normal operation, TOV0 will be set in the same clock cycle as the TCNT0 becomes zero. The TOV0 flag in this case behaves like an n^{th} bit, except that it is only set, not cleared. The associated ISR for timer overflow does not contain any specific functionality by default to handle the overflow (or to clear the TOV0 flag) and the required functionality needs to be explicitly programmed or the timer overflow interrupt needs to be enabled explicitly, which is not enabled by default. Therefore, an adversary could abuse a program that does not have the timer overflow interrupt enabled or programmed. In this way, by modifying the value stored in TOV0, an adversary would be able to disrupt the functioning of the timers.

C. Exploitation via the infrared channel

Utilizing the aforementioned programming features along with the timer registers and their ISRs, we implemented two simple vulnerable programs on iRobot Create that were exploited through its infrared sensory channel as explained in the following attack scenarios. Specifically, we used the infrared sensory channel to receive certain infrared messages (signal patterns) from its remote control and the vulnerable programs were designed to interpret these infrared messages.

1) Attack Scenario 1 - Unauthorized change in the mode of operation: In the first attack scenario, a vulnerable program, *A*, which is illustrated in Figure 5, was implemented. The program included a specific malware [35], [36] to receive a specific IR pattern and then modify the value of the timer, TCNT0. The value of this timer was manipulated by the malware to trigger the ISR corresponding to the timer overflow interrupt. As shown in Figure 5, the malware executes vulnerable program

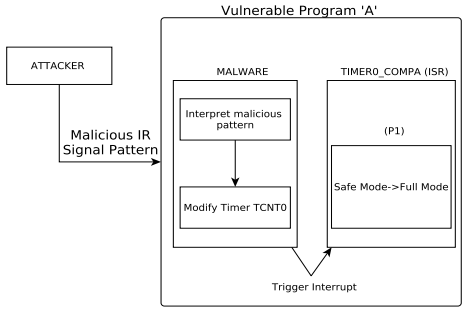


Fig. 5. Attack Scenario 1 - Unauthorized switch from safe mode operation to full mode

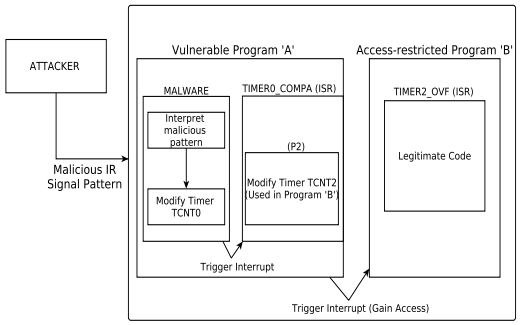


Fig. 6. Attack Scenario 2 - Malicious Access to Access-restricted Program Module

snippet, *P1*, which was placed in the corresponding ISR of the program *A*. The malicious simple IR pattern used in this attacking scenario was a combination of *PLAY* and *PAUSE* signals from the iRobot Create's remote control (*PLAY-PAUSE-PLAY-PAUSE*). An intentional generation of this combination by an attacker would execute P1. To create a malicious infrared bit pattern, we used the available macros in the remote control for *PLAY* and *PAUSE*. These macros would each constitute a specific infrared bit pattern. For instance, the bit pattern for sequence *PLAY-PAUSE* would look like *1011-1010*[1]. Therefore, the same malicious infrared bit pattern (corresponding to *PLAY-PAUSE-PLAY-PAUSE*) and any other arbitrary malicious bit pattern can be generated using any infrared emitter. More specifically, the vulnerable P1 was designed to change the mode of operation of the iRobot Create from *Safe Mode* to *Full Mode* which would lead to an unexpected behavior of the device as the outcome of the attack. For instance, in full mode an explicit *move* command would be given higher priority than the signals from the cliff sensors, thereby allowing the iRobot Create to fall down from a higher level. To realize this, we used the command *byteTX(CmdFull)* to change the mode of operation and it was pre-loaded inside the ISR of the timer for execution when the ISR was triggered via the specific IR pattern. This simple malicious operation that we implemented inside the ISR of P1 is given in the code snippet in Listing 1.

Listing 1. ISR or P1

```
SIGNAL(SIG_OUTPUT_COMPARE1A)
{\\ Malicious code
  byteTx(CmdFull); \\Safe Mode to Full Mode
  \\ Original interrupt routine
  if(timer_cnt)
    timer_cnt--;
  else timer_on = 0;
}
```

2) Attack Scenario 2 - Malicious Access to Access-restricted Program Module: In the second attack scenario, we implemented another simple vulnerable program snippet,*P2*, in the ISR of timer TCNT0 and created two programs called *A* and *B* as shown in Figure 6. In our scenario, program A was designed to modify the value of another timer, TCNT2, used in program B, which was not accessible and did not contain any vulnerability unlike program A. Both programs (A and B) were pre-installed on the iRobot Create as a single software image and the malicious IR signal pattern to trigger the vulnerable program used was the same combination (PLAY-PAUSE-PLAY-PAUSE) as in the previous attack scenario. Our goal in this scenario was to demonstrate a more complex vulnerability scenario where an external attacker shown in Figure 6 gains internal access through the infrared sensory channel to modify a program that is normally not accessible by any outside communication.

To achieve this goal, the program B was implemented to move the iRobot Create in a straight line for 100 seconds from Position A to Position B and then to turn right to continue on a straight line further for another 100 seconds to Position C as shown in Figure 7. The timer, TCNT2, of program B was utilized to handle the time delays requested by the same program. However, by modifying the value of TCNT2 (TCNT2 = 100) through P2 in program A, which was actually triggered by the malicious IR pattern, we were able to change the trajectory of the iRobot Create as illustrated in Figure 7 (from Position X to Position Y). This simple trajectory changing attack scenario could be extended to demonstrate more critical functionalities.

VI. Why conventional methods do not defend against sensory channel attacks

One effective traditional security mechanism to thwart attacks occurring in conventional communication channels (e.g., RF) is to deploy an intrusion detection system (IDS). Numerous security mechanisms involving IDSs have been proposed and are still under continuous development as a viable method for countering malicious traffic over the conventional RF communication channel such as *Snort* [38], *Suricata* [39]. These IDSs analyze network traffic looking at data traversing the protocol stack to identify malicious activity utilizing various techniques, including *signature-based detection* and *anomaly-based detection*. For instance, an example of a sample Snort alert configuration (*bind buffer overflow*) upon catching a text string (e.g., */bin/sh*) with analyzing traffic is shown below.

Listing 2. Sample Snort Alert

```
alert udp $EXTERNAL_NET any->$HOME_NET 53 \
(msg: "Exploit bind tsig Overflow attempt"; \
content:"|00 FA 00 FF|; content:"/bin/sh";)
```

[1]Each binary 1 value represents a pulse train whose duration would differ depending on the IR emitter. For the iRobot IR remote, we measured this value to be 28 ms using the Arduino shown in Figure 9(a).

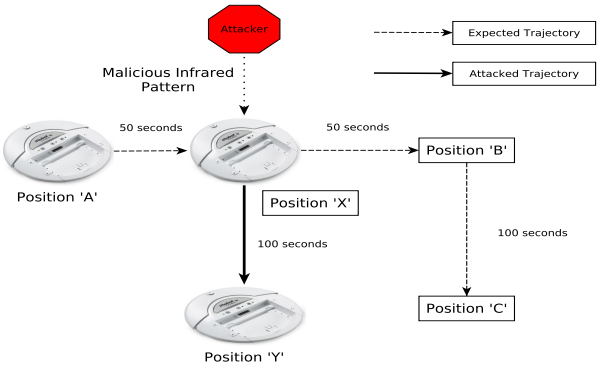


Fig. 7. Change of the trajectory as a result of Attack Scenario 2

However, such IDSs and similar security frameworks designed for analyzing network traffic cannot be directly used for traffic over the sensory channels (i.e., visible light patterns, infrared patterns and seismic vibrations). Given that the sensory channels are a new attack surface for CPS, currently there is no signature database with signatures corresponding to specific malware transmitted or attacks carried out over the sensory channels. In a similar way, anomalies that are well understood for the network traffic occurring over traditional communication lines are not defined for the sensory channel threats. For sensory channels, physical conditions of the medium, values, characteristics of the received sensory data need to be understood well. Also, each channel essentially creates an entirely new protocol stack of its own. The type of encoding used for the raw sensor values to generate traffic need to be considered by the IDS. For instance, consider a light sensor that samples ambient light intensity at specific time intervals, T_i. The raw light intensity, I_i, on the sensor's interface at the sampling instant, i, is converted to a value within the sensor's acceptable range I_r. Any program running on the sensor component would have to use these readings or values to record light measurements. Also, more advanced applications would compare these values with thresholds to look for patterns in the received or recorded readings. Usually, applications would normalize the readings or values (I_r^n) such that the general ambient light value is zero (i.e., all the readings would be scaled down such that a zero always signifies an *Off*). The sensor component would sample the environment light intensity prior to establishing a communication channel and the averaged sample value (I_{avg}) would be subtracted from all the readings recorded as:

$$I_r^n = |I_r - I_{avg}| \tag{3}$$

In this way, the normalized readings are encoded and the IDS would be required decode them first, prior to further processing. In addition, the *duty cycle* of sensory channels is important. For instance, consider an example scenario in which we have a light sensor that samples light at a specific sampling interval, T_s, and a light source generating light patterns with specific *On* duration (T_{On}) and *Off* duration (T_{Off}). A transmitted light pattern of *On-Off-On-Off* with T_s = *1s* and T_{On} = T_{Off} = *1s* is received as *On-Off-On-Off*. If the T_s and

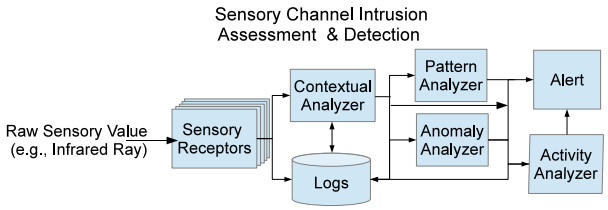


Fig. 8. Sensory IDS Architecture for CPS

T_{On}, T_{Off} are set to different values (500ms, 1s, respectively), the same transmitted pattern is received as *On-On-Off-Off-On-On-Off-Off'*. Hence, both the sensory transmitter and receiver would have agreed on an *On-Off* duration or duty cycle, ξ, which constitute the encoding technique for the transmissions:

$$\xi = \frac{T_{On}}{T_{On} + T_{Off}} \tag{4}$$

Therefore, an IDS covering the sensory channels needs to consider the duty cycle (ξ) associated with the sensory channels as well.

Another critical point to consider is the impact of environment noise. Specifically, environment noise would be more significant while operating on certain sensory channels such as infrared and seismic channels. For instance, consider a simple pattern of *On-Off-On-Off-On* that could represent a unique signature of a malicious light pattern. A sample range (R_{IR}) for a light sensor reading and its corresponding sample threshold value, R_γ (e.g., 600), for differentiating *On* and *Off* could have been defined as follows:

$$R_l \leq R_{IR} \leq R_u \tag{5}$$

$$R_{Off} < R_\gamma \leq R_{On} \tag{6}$$

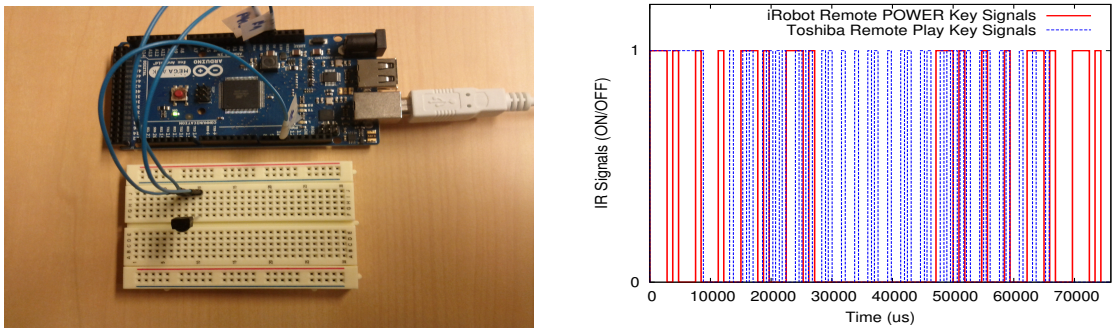
where, R_l (0) and R_u (e.g., 990 for MTS310CB [20], 1090 for MTS420CC [20]) are the lower limit and upper limit in the range of sensor readings supported by the sensor component. Noise arising from ambient light variations could generate the malicious light pattern and lead to false negatives if the fixed threshold value (R_γ) does not consider the ambient noise. Moreover, an advanced adversary can utilize values that are lower than the threshold to trigger or transfer malware. Finally, different sensors can have different thresholds as shown and analyzed in Figure 9 in Section VII. *Therefore, a direct implementation of a conventional IDS will not be efficient in detecting sensory channel threats for the CPS devices*. On the other hand, an IDS, which is similar to the conventional IDSs that combines the signature- and anomaly-based techniques, but also is cognizant of the factors influencing the sensory channels, is required and design of such an IDS for CPSs is discussed in the next section.

VII. Sensory Channel Aware IDS for CPS

In this section, we introduce the architecture of a novel IDS that is specifically designed to be aware of the sensory channels in the CPS. Given the nature of threats discussed in Section IV, the CPS IDS incorporates design features from both an anomaly-based and signature-based traditional IDSs. In addition, our design includes several sensory-channel-specific

TABLE III
SAMPLE CPS IDS RECORDS

#	Event Name	Event Timer	Event Counter	Sensory Channel	Sensory Value	Module	Action
1	Suspicious activity	05:00 AM	3	Light	800	Contextual Analyzer	Notify
2	Normal	10:00 AM	5	Light	450	Anomaly Analyzer	Log
3	Suspicious	11:00 AM	2	Infrared	40	Anomaly Analyzer	Notify
4	Suspicious	11:40 AM	3	Temperature	110	Pattern Analyzer	Notify



(a) (b)

Fig. 9. (a) Arduino setup to analyze the IR values from different IR emitters; (b) Received IR patterns from iRobot Create Remote and a TV Remote

components. Our design is modular and flexible that allows the adoption of the design for different host devices. A high level design of the IDS is illustrated in Figure 8 while the details for each module are explained below.

Sensory Receptors: The primary functionality of the Sensory Receptors module is to receive raw sensory values from the sensors on the host device. Depending on the host device type (e.g., a smartphone, an iRobot, a Zigbee device) (Figure 2), the Sensory Receptors include one sensor or a combination of several sensors. For instance, there would be accelerometers, IR sensors on an iRobot as explained in Section III.

Contextual Analyzer: One important component of the CPS IDS is the Contextual Analyzer module. In this module, a sensory value sent from the receptor module is simply checked if it is within normal ranges for a specific deployed region. For instance, a light intensity value sensed via a light sensor for a room in a home environment and for an office environment at work may be different from each other. Similarly, a temperature sensor deployed at different geographical regions in the world will have varying operational values.

Moreover, as discussed in Section VI, different sensory channels (e.g., light vs. infrared sensor) will have different operational parameters and characteristics (e.g., thresholds, duty cycle) while even different sensors for the same sensory channel can have different characteristics. To evaluate this, we performed a simple experiment shown in Figure 9(a). Specifically, using an Arduino, we measured the infrared values from two different IR emitters: iRobot Create's remote and a Toshiba TV remote. We observed that although different IR emitters operated in the same frequency, they have different pulse trains to represent on and off periods. For instance, for iRobot Create (red line in the figure), 28 ms represented the on period while for the TV remote (blue line in the figure) the on period was 6.9 ms. Each IR emitter also had different patterns as seen in the figure.

All these different situations necessitate a CPS IDS to be cognizant of the environment it operates in and have different operational parameters, thresholds for each sensory channel. Given the various usage scenarios of sensors in different working conditions, environment cognizant operation is considered as an important component for the CPS IDS. Similarly, characteristics of different sensory channels is also important. Although alternatively these different operational conditions can be configured as profiles under the Anomaly and Pattern Analyzer modules, our rationale for the Contextual Analyzer module is for simplicity of functions. In lieu of triggering all the functionalities provided by the Anomaly and Pattern Analyzer modules, a simple contextual value check supports efficient utilization of resources for mostly resource-constrained devices of the CPS realm.

Pattern Analyzer: This module serves a similar purpose as the traditional signature-based IDSs. As discussed and illustrated (Figure 9(b)) earlier, a specific sensory traffic pattern (signature) may represent a certain malicious activity or facilitate a malicious purpose on the host system. Hence, the primary function is to catch any traffic pattern in the received values from the sensory channels. This module provides a simple method to define malicious patterns and allows users an easy extension for different patterns/signatures. If a malicious pattern is identified then the Alert module is triggered to generate the necessary action.

Anomaly Analyzer: In this module, sensory events and values are analyzed to determine abnormalities in the observations. For each sensory channel, value thresholds, historical statistical value distributions (e.g., mean) are stored and utilized to determine abnormal behavior. Similar to the Pattern Analyzer module, if an abnormality is identified then the Alert module is triggered to generate the necessary action.

Activity Analyzer: It is possible that a malicious sensory

pattern does not match any of the patterns stored by the system in the Pattern Analyzer module or an abnormality in the sensory values is not detected by the Anomaly Analyzer module. In order to thwart and decrease the damage of such sensory attacks that evade the aforementioned detection modules, the Activity Analyzer comes into play. Given the fact that most CPS devices are resource-constrained, this module is an important component in our CPS IDS design. Specifically, this module is mainly responsible for tracking the energy consumption, CPU utilization, and the memory activities of the events associated with the sensory channel. For instance, if an abnormal increase in energy consumption or CPU utilization is detected, the Alert module is triggered to generate the necessary action.

Alert: This module is responsible for generating the alarms/alerts upon the detection of the intrusion events. The module can be triggered to generate an alarm by the Contextual, Pattern, or Anomaly Analyzer modules.

Logs: In order to facilitate the healthy operations of the aforementioned modules in the CPS IDS, sensory values received from the sensors and the patterns/signatures, records for abnormal activities are all stored in the Logs module for a certain configurable amount of time. A sample log of IDS entries are shown in Table III.

VIII. Conclusion and Future Work

An important component of Cyber-Physical Systems (CPS) is the sensors (e.g., light, temperature, infrared) they utilize to interact with each other and the physical world around them. These sensors form *sensory channels* serving as external interfaces to host CPS systems. *In this paper, we focused on threats to CPS applications and devices through their sensory channels. Specifically, we presented how an adversary could successfully attack systems using the sensory channels and analyzed the feasibility of these attacks on various sensory channels.* Using an iRobot Create as our CPS platform, we exploited simple vulnerable programs on iRobot through its infrared channel. Finally, we introduced the design of a novel sensory channel aware intrusion detection system as a protection mechanism against the sensory channel threats for CPS devices. Sensory channel threats is critical because current security solutions are limited to protecting the CPS components networked via traditional means (e.g., RF) or services on the host devices are not suitable for sensory channel threats. In the future, we will develop a comprehensive solution to detect sensory-channel based attacks and defend the host devices from them while evaluating other sensory channels available on other CPS platforms.

Acknowledgment

This work was partly funded by NSF Grant No. CNS-1052769.

References

[1] NSF, "Cyber-physical systems program solicitation (nsf 13-502)," http://www.nsf.gov/pubs/2013/nsf13502/nsf13502.htm.
[2] Asad-Uj-Jaman, "http://mobiledeviceinsight.com/2011/12/sensors-in-smartphones/," *Mobile Device Insight*, 2011.
[3] GE, "Ge monitored wireless home security system, http://www.gehomealarmsystems.com/ge-home-security-alarms-equipment/," 2013.
[4] D. Hambling, "http://www.wired.com/dangerroom/2009/04/army-tests-new/#more," *Wired*, 2009.
[5] J. Stokes, "http://www.wired.com/autopia/2011/02/the-future-of-cars-p2p-mesh-4g-and-the-cloud/," *Wired*, 2011.
[6] "Cars sensors system," http://www.roydrivingschool.com/2012/01/cars-sensors-system.html.
[7] *iRobot Create, http://www.irobot.com/*, iRobot.
[8] C. DesMarais, "http://www.pcworld.com/article/253882/domestic_robots_high_tech_house_helpers.html," *PCWorld*, 2012.
[9] *Da Vinci Surgery System, http://www.davincisurgery.com/da-vinci-surgery/*, da Vinci Surgery.
[10] "The internet of things," http://www.theinternetofthings.eu.
[11] "Planetary skin," http://www.planetaryskin.org/home.
[12] J. Bort, "10 technologies that will change the world in the next 10 years, http://www.networkworld.com/news/2011/071511-cisco-futurist.html," 2011.
[13] P. Marquardt and et al., "(sp)iphone: Decoding vibrations from nearby keyboards using mobile phone accelerometers," in *Proc. of ACM CCS*, October 2011.
[14] A. J. Aviv and et al., "Practicality of accelerometer side channels on smartphones," in *Proc. of 28th ACM ACSAC*, December 2012.
[15] I. Rouf and et al., "Security and privacy vulnerabilities of in-car wireless networks: A tire pressure monitoring system case study," in *Proc. of the USENIX Security Symposium*, 2010, pp. 323–338.
[16] D. Foo Kune and et al., "Ghost talk: Mitigating EMI signal injection attacks against analog sensors," in *Proc. of the IEEE Symposium on Security and Privacy*, May 2013.
[17] V. Subramanian, S. Uluagac, H. Cam, and R. Beyah, "Examining the characteristics and implications of sensor side channels," in *Proc. of IEEE ICC*, June 2013.
[18] L. Cai and H. Chen, "Touchlogger: Inferring keystrokes on touch screen from smartphone motion," in *Proc. of USENIX HotSec*, August 2011.
[19] S. Checkoway and et al., "Comprehensive experimental analyses of automotive attack surfaces," in *Proc. of the USENIX Security Symposium*, 2011.
[20] *MTS/MDA Datasheet, http://retis.sssup.it/sites/retis.sssup.it/files/Sensor*, Crossbow.
[21] *MicaZ Datasheet, http://bullseye.xbow.com:81/Products/productdetails.aspx?sid=164*, Crossbow.
[22] *Telosb Datasheet, http://bullseye.xbow.com:81/Products/productdetails.aspx?sid=252*, Crossbow.
[23] B. Ferguson, "Method and apparatus to control display brightness with ambient light correction," in *U.S.Patent US20090091560*, April 2009.
[24] C. J. Cassidy, "Airborne laser mine detection systems," Master's thesis, Naval Postgraduate School, Monetery, California, 1995.
[25] R. Roman and J. Lopez, "Keyled - transmitting sensitive data over out-of-band channels in wireless sensor networks," in *Proc. of the 5th IEEE MASS*, October 2008.
[26] M. Gauger and et al., "Enlighten me! secure key assignment in wireless sensor networks," in *Proc. of the 6th IEEE MASS*, October 2009.
[27] *PlayStation 3, http://us.playstation.com/ps3/*, Sony.
[28] *Nintendo Wii, http://www.nintendo.com/wii*, Nintendo Co.,Ltd.
[29] C. Gao, R. Pastel, and J. Tan, "Yet another user input method: Accelerometer assisted single key input," in *Proc. of world congress on Intelligent Control and Automation*, July 2010.
[30] *Bump, http://bu.mp.*, BUMP Technologies.
[31] H. Ketabdar, M. Roshandel, and D. Skripko, "Towards implicit enhancement of security and user authentication in mobile devices based on movement and audio analysis," in *Proc. of ACHI*, February 2011.
[32] B. Kaushik, D. Nance, and K. K. Ahuja, "A review of the role of acoustic sensors in the modern battlefield," in *Proc. of the 11th AIAA Aeroacoustics Conference*, May 2005.
[33] TurtleBot, http://ros.org/wiki/Robots/TurtleBot.
[34] *Kinect, http://www.xbox.com/en-US/kinect*, Microsoft.
[35] L. Lin and et al., "Trojan side-channels: Lightweight hardware trojans through side-channel engineering," in *Proc. of the International Workshop on CHES*, September 2009.
[36] A. Halderman and E. Felten, *Lessons from the sony CD DRM Episode, https://jhalderm.com/pub/papers/rootkit-sec06.pdf*.
[37] *ATmega168 Datasheet, http://www.atmel.com/images/doc2545.pdf*, ATMEL.
[38] M. Roesch, "Snort-lightweight intrusion detection for networks," in *Proc. of LISA Conference*, November 1999.
[39] *Suricata, http://suricata-ids.org/*, OISF.