**Noname manuscript No.**
(will be inserted by the editor)

# On Sensor Security in the Era of IoT and CPS

**Max Panoff · Raj Gautam Dutta · Yaodan Hu · Kaichen Yang · Yier Jin***

Received: August 31, 2020 / Accepted: November 29, 2020

**Abstract** Sensors play an integral role in numerous devices across a diverse range of domains. While Cyber-Physical Systems (CPSs) and the Internet of Things (IoT) use them extensively, sensors can also be commonly found in many standalone electronic devices. Concerns over the susceptibility of sensors to malicious attacks have led academia to focus on the security of these sensors. To help unite these efforts, we propose a lexicon to easily differentiate between types and methods of attacks on sensors. By using these definitions, one can quickly and clearly understand the method and the target of an attack. We examine the most recent and influential attacks on sensors, especially when they are acting as edge nodes of systems, as well as defenses against said attacks. We then seek to categorize these methods according to our lexicon, demonstrating its usefulness and solidifying the meaning of proposed terms.

## 1 Introduction

Sensors play an integral role in numerous devices across a diverse range of domains. Cyber-Physical Systems (CPSs) [18] and the future Internet of Things (IoT) will heavily rely on sensors [26]. CPSs and the IoT seek to integrate sensing, computation, and control of actuators by networking distinct devices together [16, 56]. Specifically, they tie sensors, processors, and actuators in the real (physical) world together through network (cyber) communications. CPSs are now involved in many areas: everything from certain brands of modern refrigerators [45], to controllable loads in power grids [25] [71] [2], and potentially to fleets of autonomous vehicles [9]. Because of the IoT, CPSs, and other fast growing techniques that rely

Max Panoff, Raj Gautam Dutta, Yaodan Hu, Kaichen Yang, and Yier Jin
Security in Silicon Lab in the Department of Electrical and Computer Engineering
University of Florida
Gainesville, Florida 32611 USA
ORCIDs (in order of authorship): 0000-0003-2849-7197, 0000-0002-5686-5666, 0000-0003-3075-982X, 0000-0003-1027-6708, and 0000-0002-8791-0597
E-mail: yier.jin@ece.ufl.edu, Phone: +1-352-294-0401
*Corresponding Author

on accurate sensor readings, attacks on sensors are gaining increasing attention in academia. This is especially true for sensors often used in high-stakes situations, e.g. cameras and lidar in autonomous vehicles or phasor measurement units in power grids.

In these systems, sensors commonly play the role of *edge nodes* [4]. Edge nodes are the components of a system that interact with environments, components, or other miscellanea that are not a part of that system [4]. Thus edge nodes form the border or *edge* between what belongs to the system and what does not, hence their name. As such, edge nodes are often more susceptible to attacks than other points in a system [6], due to their external exposure. That is not to say that internal connections are perfectly secure, there are many issues facing such nodes [21], but a system's sensors are often exposed both in the physical and cyber domains. As such, there have been a number of prior literature reviews on sensor attacks [3, 17, 25, 26, 55], with the most related to this particular topic being from Alladi *et al.* [3], Thapliyal *et al.* [55], and Giechaskiel and Rasmussen [17]. Alladi *et al.* examined IoT security in general and proposed their own taxonomy to describe attacks in that realm. Thapliyal *et al.* similarly propose a taxonomy specific to vehicular security issues. Meanwhile, Giechaskiel and Rasmussen covered materials more similar to our paper, where they proposed a taxonomy for what they termed "Out-of-Band Signal Injection Attacks." These attacks fall under Perception Stage Sensor Exploit Attacks in our work, which will be further expanded upon in Section 2. Our main contribution expands upon these papers by providing a unified classification for attacks on sensors as edge nodes.

There have been many demonstrations impairing CPSs through the sensors they depend on. Lidar sensors and point cloud object detection in autonomous vehicles are vulnerable to external influences [44] [7, 48, 64]. Additionally, attacks on DNN image analysis are a well studied topic [31, 36, 65]. Global Positioning System (GPS) receivers in smart grids [47, 71], and wheel speed detection in cars [49] are all vulnerable to sensor attacks. Even attacks on Automatic Speech Recognition (ASR) like Apple's *Siri* or Amazon's *Alexa* have been deeply explored [1,29,54,68,69]. Attacks on sensors as edge nodes have ramifications outside of CPSs as well. GPS spoofing and jamming attacks are believed to be responsible for several ship collisions [63], in addition to less disastrous maritime events [19]. They even have been shown capable of disrupting aircraft landing systems [46]. Medical systems that are traditionally offline, such as baby incubators [59] and medical dosage regulators [43] have also been attacked. Even commonly used gyroscopes and accelerometers have been exploited [52, 58], and these can and do belong to many traditional and cyber-physical systems. There likely are other sensors with exploits unknown to academia but which are known to black hat agents. Even for sensors with known exploits and tested defenses, implementing said defenses may be non-trivial, or uneconomical after considering the risk. To rectify said issues, simpler and cheaper methods of defense are required.

Sensors' susceptibility to external actors, especially malicious ones and the defense of said sensors, is an expanding topic of research. Currently the terms used across the many disciplines involved in sensor defense can be highly diverse or overlapping [17]. This can lead to confusion when discussing methodology. Take "False Data Injection" and "Spoofing" as an example. The differences between these can often be unclear for newcomers to this field. We present a specific terminology that allows readers to determine the goal of the attack and gain some measure of

its operating principles. Even in recent works that propose their own taxonomies, categories can be difficult to distinguish. While there may be distinctions in these that hold for IoT at large, the categories are very similar for edge nodes. For example in [3] when describing a "Device Software Attack" the authors present a case where a line of code is changed through the debug port, but then latter describe "Malicious Code Injection" as when a "device is compromised by injecting malicious code into the device via exposed and insecure software/hardware interfaces." In our method, all such attacks would be called Sensor Commandeering Attacks, as rather than altering the transduction of a device, a third-party takes direct control of it. Specific language allows for complex ideas to be discussed quickly and clearly [38,39]. Although this can create a slight learning curve for the current community, it has been found to be overall beneficial to new minds entering the field [62]. To help unite the efforts to protect sensors against malicious attacks, we propose a lexicon to help differentiate and compare between types and methods of attacks. We go further to provide definitions and examples to our lexicon, and in doing so seek to cover the current state of the art on sensor attacks and defenses through our contributions.

Our paper is organized as follows, in Section 2 we introduce stages divided according to the proposed lexicon and present examples of various sensor attacks belonging to each stage. Following that, in Section 3, we list defenses to the types of attacks, and how they relate to the general defense strategies at each stage. We then move onto Section 4, where we discuss new directions of research. Finally, in Section 5, we present our conclusions.

## 2 Attacks

Table 1: Details of each attack type, the stage to which it belongs, and its defining feature.

| Attack Stage | Attack Type | Defining Feature |
|---|---|---|
| Reception | Classical | In-Band Transduction [17] |
| Reception | Data | Inter-System Communication |
| Perception | Sensor Exploit | Out-of-Band [17] |
| Perception | Algorithmic Attacks | Perturbed Input [31] |
| Projection | Sensor Commandeering | External Control |
| Projection | Signal | Intra-System Communication |

In this survey, we divide attacks on sensors into three stages: *The Reception stage* contains attacks that alter information the sensor receives from its environment through an expected vector (*i.e.* in-band transduction attacks [17]), *Perception stage* attacks alter how the system interprets the information it receives, and *Communication stage* attacks target communication of information. Attacks on each stage are broken down into two sub-categories. These are listed in Table 1, and examples of each occurring in an autonomous vehicle system are given in Figure 1. For clarity's sake, we here on use the term "attack" to refer to malicious


Fig. 1: Examples of different attacks on an autonomous vehicles as classified by our lexicon. 1 refers to the GPS Data Spoofing attack as described by Oligeri *et al.* 2 is the Classical Spoofing/Jamming attack on lidar as conducted by Petit *et al.* against lidar. 3 represents Algorithmic attacks generated by Kurakin *et al.* for deep learning image classification. For 4, we select the Sensor Commandeering Sleuthing attack by Kumar *et al.* against in-car Automatic Speech Recognition (ASR). In 5, we show the Sensor Exploit Spoofing attack against gyroscopes by Son *et al.* Finally, 6 refers to the Signal Sleuthing/Spoofing attack completed by Roufa *et al.* against tire pressure networks.

Table 2: An outline of each type of attack and their potential Targets.

| Attack Type | Spoofing | Jamming | Sleuthing |
|---|---|---|---|
| Classical | ✓ | ✓ | ✗ |
| Data | ✓ | ✓ | ✗ |
| Sensor Exploit | ✓ | ✓ | ✗ |
| Algorithmic Attacks | ✓ | ✓ | ✗* |
| Sensor Commandeering | ✗ | ✓ | ✓ |
| Signal | ✓ | ✓ | ✓ |

exploitation of a single vulnerability. In spite of this, we note that it is possible for a single exploit to require multiple vulnerabilities from different sub-categories concurrently. Further, we propose that attacks can also be further specified by *Target*. We propose the following Targets: *Spoofing, Jamming*, and *Sleuthing*, which are distinct, though it is possible for a single attack have the ability to have multiple Targets, just not at the same time. Spoofing attacks aim to introduce false data into a system. Jamming attacks are Denial-of-Service (DoS) attacks. Sleuthing attacks provide confidential information about a target system to an adversary. Some stages have no current examples of attacks with certain Targets, as shown in Table 2.

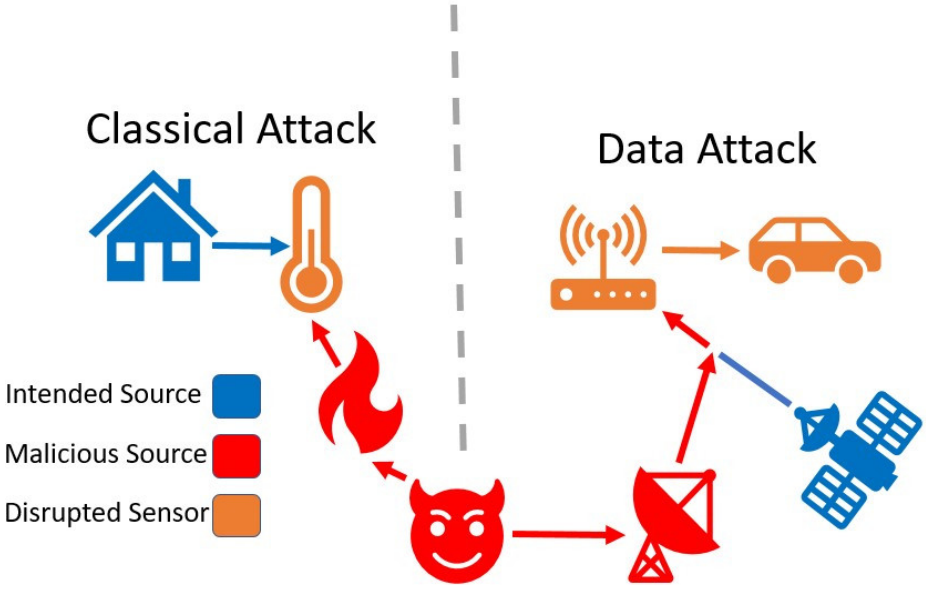


Fig. 2: A visualization of Reception Stage Attacks: Classical Attack (left); Data Attack (right).

## 2.1 Reception Stage Attacks

Reception Stage attacks entail an adversary who has the ability to influence the sensing apparatus along its intended vector. When we say that an attacker's ability to influence sensing apparatus along its intended vector, we mean they can influence the environmental conditions the sensor is monitoring. Examples include: directing lasers (light) at camera lenses, increasing heat near a thermometer, and creating Electromagnetic signals with the same method as a victim receiver (*i.e.* FM broadcasting for FM radio). These are referred to as an In-Band attacks [17]. The Reception stage has two sub-categories, *Classical* and *Data attacks*. Classical attacks encompass simple transduction attacks, while Data attacks focus on Electromagnetic (EM) signals external to the victim system, which are more complex. We can see a simple example of Reception Stage Attacks in Figure 2. In a rudimentary Classical attack, an adversary heats the area near a thermometer monitoring a smart home, resulting in incorrect readings for the home as a whole. And in the Data Attack, the adversary overwhelms actual GPS info with their own signals.

### *2.1.1 Classical Attacks*

Classical attacks on the Reception stage of systems are often the most straightforward and simplest attacks to understand, as they are often In-Band [17] attacks. Giechaskiel and Rasmussen describe Out-of-Band attacks in their work, which are attacks taking place on vectors that the sensor is not designed to operate on [17]. Classical attacks are the opposite of these, where attackers manipulate signals that the sensor is expecting. Petit *et al.* provide a clear example of a Classical Spoof-

ing attack in their lidar attack. They direct lasers of the same wavelength as a victim lidar at it, causing the victim lidar to incorrectly record the time its own projections have traveled before returning. By altering the recorded flight time of a victim lidar's beams, its understanding of the distance to nearby objects is changed [44]. By creating false points of data, Petit *et al.* are specifically conducting a *Spoofing attack*. Petit *et al.* can Spoof fake walls from 40 to 70 meters from the victim lidar. Spoofing attacks can be achieved through various methods across all three stages, but Classical Spoofing Attacks are done by directly altering the sensing apparatus or environment. Petit *et al.* achieve this by directing a laser at the target lidar's photodiodes. Petit *et al.* also direct Infrared (IR) beams at cameras, resulting in the camera's autoexposure blinding the camera. This is a *Jamming attack*, where sensors are barred from fulfilling their function, which is often achieved by overwhelming or otherwise saturating the sensor in Reception attacks. Shin *et al.* expand upon Petit's work on Classical attacks against the Velodyne lidar [48], creating up to 10 new points at any position (*i.e.* a Spoofing Attack) and saturating (*i.e.* a Jamming attack) the lidar to prevent the detection of up to $1m^2$ of area. This can potentially create situations where Autonomous Car CPSs can be tricked into detecting nonexistent objects or fail to detect existing ones [7]. These failures could result in non-optimal behaviors or even accidents [7]. Shoukry *et al.* also perform a Classical attack against anti-lock braking systems. Shoukry finds that by activating a magnetic actuator near a magnetic encoder, malicious entities can effectively control the input to the encoder. As the actuator produces much stronger emissions than the magnet sensor paired with the target sensor, it can overwhelm the true readings, resulting in both successful Spoofing or Jamming attacks [49]. This could especially cause issues for CPSs such as autonomous vehicles, which depend on accurate tire speed and traction readings.

#### *2.1.2 Data Attacks*

*Data attacks* imitate (*i.e.* Spoof) or deny (*i.e.* Jam) external data carried by structured Electromagnetic (EM) waves (e.g. AM and FM radio communication, GPS signals). Reception stage Data attacks also "cross over" the edge of a system. While very similar to Classical attacks in that intended transduction is the core mechanic, we distinguish between Data and Classical Attacks due to the complexity of Data attacks. An example of this is in GPS Spoofing. GPS Spoofing is not simply creating EM waves, or even recording real EM waves and replaying them. In order to Spoof, the attacker must have a desired location to translocate the victim to. In order to achieve that, complex calculations must be done to identify the timing of multiple signals. In many cases Data attacks are even more complex, as the EM wave contains encryption that protects the system from false signals. As the satellites transmitting the GPS data do not belong to the system (*i.e.* are external), attacks that target these transmissions belong to the Reception Stage. This is in contrast to Communication stage Signal attacks, which will be discussed latter on. An example of a Data Spoofing attack is given in Oligeri *et al.*, where the authors overwhelm GPS receivers in a car with transmissions that mimic true GPS [42]. GPS Spoofing is a danger to maritime ships as well [19, 63]. GPS Jamming against ships has also been successfully conducted, and is thought to be behind several maritime accidents [19, 63]. Smart grids and other CPSs [47] are also potential victims of GPS Data attacks. Shepard *et al.* examine the susceptibil-

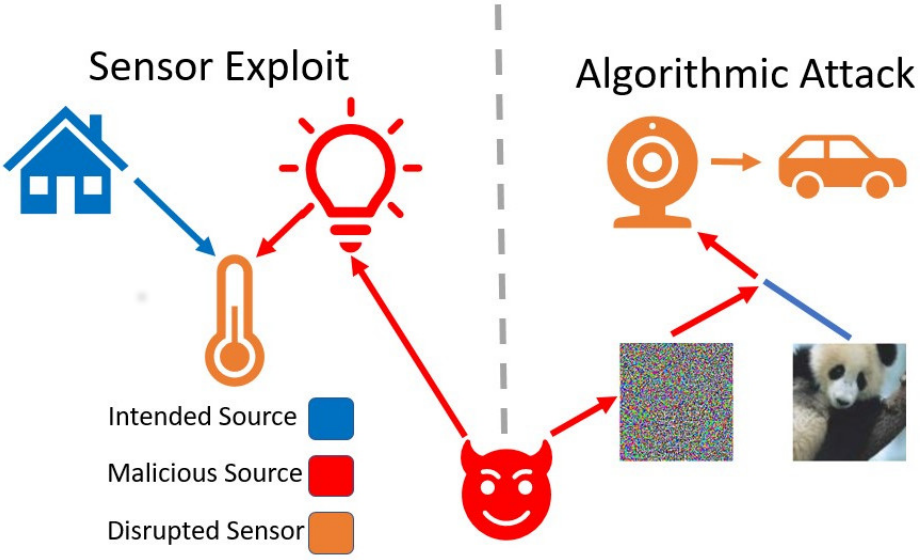


Fig. 3: Sensor Exploit Attacks (left) and Data Attacks (right), both belong to the Perception Stage.

ity of smart grid sensors called Phasor Measurement Units (PMUs) to GPS Data Spoofing attacks. PMUs are used in smart grids to obtain the voltage, current, and phase at different points of the network, and then send that data to a central location. Even small changes to the reported time of PMU data could potentially lead to an incorrect response from the system, causing blackouts or generator shutdowns [47]. PMUs use GPS information to obtain accurate timing and Shepard uses Data Spoofing Attacks on individual PMUs to destabilize the Smart Grid as a whole [47]. Sathaye *et al.* target Aircraft Instrument landing system sensors with a Data Spoofing attack and lead them to large offsets of ground truth (50 m) due to the presence of external signals [46].

### 2.2 Perception Stage Attacks

Perception Stage attacks are attacks utilizing design oversights to control a system. We divide Perception stage Attacks into two categories. Sensor Exploit attacks, described as Out-of-Band attacks by Giechaskiel and Rasmussen [17], use unintended transduction to alter readings. Further analysis of these attacks can be found in [17]. Algorithmic attacks meanwhile target control algorithms, especially machine learning algorithms, to disrupt how a victim system perceives stimuli. Examples of Perception Stage Attacks are in Figure 3. The left side shows that through a fictitious Sensor Exploit, the adversary can alter a smart home's reported temperature by directing light towards an in-home thermometer. On the right, an Algorithmic Attack has adversary creating an adversarial example to the image as seen from an on-car camera.

*2.2.1 Sensor Exploit Attacks*

*DolphinAttack* by Zhang *et al.* uses high-frequency sounds, inaudible to humans, but audible to commonly used microphones due to non-linearities [69] in the victim sensor's circuit. This non-linearity allows malicious entities to imperceptibly command Automatic Speech Recognition (ASR) devices such as Amazon's *Alexa*. ASRs are a type of CPS where audio data is captured at the user's home and analyzed by a remote server in order to determine the command given. By modulating a voice command with an ultrasonic carrier, the authors are able to both activate (command: "Hey Siri") and recognize commands("turn on airplane mode") over 80% of the time even with 75 dB of background noise. DolphinAttack is chiefly a Sensor Exploit Spoofing Attack, as if an attack is conducted to create a false order, or change some settings, it is a Spoofing attack. An induced voice command to gain information from the system is also a Spoofing attack, as the the attack is actually creating false data points, rather than obtaining data directly. Son *et al.* and Trippel *et al.* show the susceptibility of Micro-electromechanical systems (MEMS) gyroscopes and Inertial Measurement Units (IMUs), respectively, to external vibrations [52,58]. They demonstrate the ability to conduct Sensor Exploit Spoofing by changing the readings to a certain desired input. Trippel *et al.* has shown their capability to control the victim IMU to spell "WALNUT" when its readings are plotted. Similar to DolphinAttack, in *LightCommands*, Sugawara *et al.* find that they can induce an ASR to perceive spoken words by directing a laser at a victim microphone by modulating the power of the laser with recorded audio. The authors were able do this 110m from the victim device, and while in a separate building by directing a laser through adjacent windows. Attacks such as these, called Out-of-Band attacks [17], are Sensor Exploit attacks in our lexicon. [54]. They are grouped by their shared use of unintended transduction vectors. That is, attacks that make use of transactions capabilities are unintended. One does not expect an accelerometer to record vibration in addition to acceleration. Sensor Exploit attacks make use of these oversights to alter sensor readings through unprotected vectors.

*2.2.2 Algorithmic Attacks*

Algorithmic attacks focus on the algorithm through sensor data collected by the system. Many CPSs, as well as more traditional systems, use various algorithms to control a system. These "control algorithms" respond and adapt to their environment. However, it is often possible to "trick" this control algorithm into incorrect responses through certain stimuli, which is the focus of Algorithmic attacks. Algorithmic attacks depend on the creation of *adversarial examples*. Adversarial examples are most often deployed against Deep Neural Networks (DNNs). While other methods of machine learning, such as k Nearest Neighbor (KNN), have been shown to be vulnerable to adversarial examples as well, there is far less academic interests in those areas. In [51], Sitawarin *et al.* focus on creating examples that are adversarial to both DNN and a KNN. This is because KNNs can be used to screen for adversarial examples and identify them before passing them to a DNN for analysis. However, they find creating adversarial examples against KNNs to be intractable [51]. Adversarial examples add *perturbations* to a given instance, causing DNNs to incorrectly evaluate the instance. Perturbations are artificially

induced changes to inputs that greatly alter DNN responses. They are often limited in number or "distance" (*i.e.* how much they change the input). A traditional example of abiding by this limitation is given by Kurakin *et al.* where small changes to pixel values in an image results in large changes to classification by a popular DNN [31]. These perturbations are often created to Target Spoofing or Jamming.

In *CommanderSong*, Yuan *et al.* place malicious voice commands in a song, hidden in plain sight, rather than being undetectable [68]. They do this by adding perturbations to an existing song, encoding voice commands that sound different to ASR than to humans. This is an Algorithmic Spoofing attack very similar in goal to DolphinAttack. The created song can trick ASR above 70% of the time, even when being played back from a speaker. The authors also asked subjects to identify if they heard anything abnormal from within the altered song. Less than 30% of respondents could identify an abnormal element, and none could recognize the command. More recently, in *Devil's Whisper*, Chen *et al.* are able to further improve on this strategy, with even lower detection rates [10]. Abdulla *et al.* [1] describe an alternative Algorithmic Spoofing attack on ASR systems using a fundamentally unique methodology. Rather than utilizing an existing sound source (*e.g.* a song) to cover their commands, the authors use psychoacoustics to restrict their inputs to phonemes that fall outside of a given language but which Automatic Speech Recognition (ASR) for that language recognizes. They use deep learning to create a sequence of these phonemes that elicits a response form the ASR.

Adversarial point cloud creation is also a highly active topic, with particular focus on networks used by autonomous vehicles [7]. Creating adversarial point clouds is often done by either shifting or creating points as perturbations. However, in many cases the amount of perturbation is limited to amounts similar to what can be achieved by current lidar Classical Spoofing attack methods. As an example, Shin *et al.* find the ability to spoof only 10 points in a point cloud, without much control over their placement [48]. That said, by restricting the attack, it can feasibly be implemented as a Classical Spoofing attack on the Reception stage through exiting lidar methods immediately [44] [48]. Alternatively, some researchers ignore the current restrictions and instead change the actual surface of an object to reflect a generated point cloud [7]. In [7], Cao *et al.* create a neural network that generates adversarial point clouds against the lidar perception module of Baidu Apollo. The authors then use a 3d printer to construct these designed objects in the real world. The created objects are able to avoid detection by Baidu Apollo around 90% of the time, even when placed within 1m of a lidar.

In sticker attacks against video deep learning models, carefully constructed stickers are attached to either objects or placed over the camera lens [36] [65]. Li *et al.* create stickers with a Deep Neural Network (DNN) as static masks that subtly change how pictures are interpreted by other target DNNs, preventing accurate object detection, (*i.e.* a Jamming attack). These stickers are overlaid on a target camera, appearing as dots or smudges to humans, but completely distort a DNN's perception of the scene. These adversarial actions include: changing all cars in an image to be detected as toasters or not at all (*i.e.* Jamming), or having toasters be detected where they are not (*i.e.* Spoofing). Sticker attacks were found to be effective on image classification and segmentation, as well as object detection [36]. That said, Li's attack succeeded between only 27.9% (for 1 dot) and 49% (for 10 dots) of the time. Attacks such as these could lead to accidents if autonomous vehicles or other transportation CPS are not adequately protected [36].

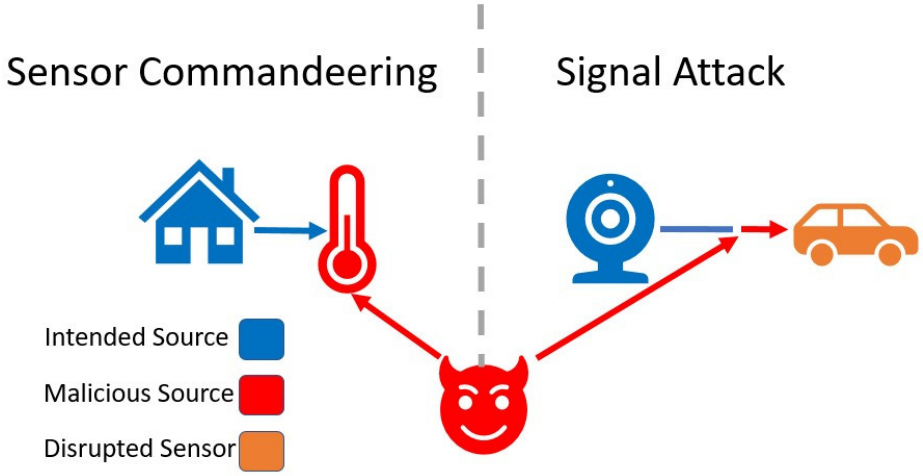


Fig. 4: Attacks on the Communication Stage, specifically, a Sensor Commandeering Attack (left) and a Signal Attack (right)

## 2.3 Communication Stage Attacks

Communication Stage attacks target the sensor as an edge node, that is, as a sensor relays data either into, or out of, a system. Attacks on this Stage are often more "cyber" than "physical" in nature, the opposite of Reception stage attacks. That is, attacks on this stage targets communication, rather than transduction. This stage can be divided into two subcategories, Sensor Commandeering attacks, and Signal attacks. Sensor Commandeering attacks change some functionality of the system to benefit the attacker, while Signal attacks target the communication between the sensor and the rest of the system. We demonstrate Communication Stage attacks in Figure 4. In this Figure, a Sensor Commandeering attack on the left has an adversary take direct control of a thermometer in a smart home, while in the Signal Attack on the right, the adversary disrupts the communication between an intact on-board camera and a victim vehicle.

### *2.3.1 Sensor Commandeering Attacks*

The hard drive attack as described by Kwong *et al.* in *Hard Drive of Hearing* [32] is an example of a Sleuthing attack through *Sensor Commandeering*. Kwong *et al.* use information gathered by target sensors to extrapolate an additional facet of the environment. By repurposing information that hard disks collect to maintain functionality, namely the offset of a needle from a target position, to record nearby noises allowing it to act as a microphone. This data is then sent to the attacker. Data centers or other locations with many hard drives and high amounts of background noise, where people need to speak loudly to be heard, would be particularly susceptible to the *Hard Drive of Hearing* attack [32] Kwong *et al.* are able to recover words spoken 10cm from the hard drive with a 11.8dB signal to noise ratio for male speakers and a 12.8 signal to noise ratio for female speak-

ers. As this attack changes how a sensor's data is used, namely expanding the sensor's detection capabilities in a new direction, it is a Sensor Commandeering attack. Sensor Commandeering attacks can be difficult to detect, as the sensor likely never stops performing as intended, instead it gains new responsibilities, likely to the detriment of the operators of its original system. In Kumar *et al.*'s *Skill Squatting* [29], the attack consists of changing the outputs the sensor recognizes as valid. Specifically, they examine if Alexia *Skills* (*i.e.* commands that need to be downloaded from external sources) can be created with similarly pronounced names to true skills, or with names similar to other skills that are commonly mispronounced or misidentified. Examples of the "squatted skills" include company names that have different regional pronunciation, or certain sounds that humans commonly misspeak. This can be done to Jam users who wish to access a certain service or skill. This functions is a Sleuthing attack as well, as the attack is used to access confidential information. Sensor Commandeering attacks may be particularly effective against CPSs as a CPSs component may be in infrequently checked or improperly secured areas. This may allow for adversaries to gain access to the device and alter it for their own purposes.

#### *2.3.2 Signal Attacks*

Signal attacks focus on disrupting the EM signals of a system, after transduction. Instead of interfering with transduction directly, Signal attacks can mimic a sensor and provide false information to (Spoofing or Jamming attacks), and receive confidential information back (Sleuthing attacks), from the system. They are most often deployed against networks of sensors that utilize wireless communication, such as CPSs. Roufa *et al.* present an example of this, where the authors Spoof and conduct a Sleuthing attack on a tire pressure monitoring system [24]. In this work, tire pressure sensors communicate wirelessly with a central node in a victim car. Roufa *et al.* implement *Signal attacks*, where attackers inject (*i.e. Signal Spoofing*), deny (*i.e. Signal Jamming*), or listen in on internal system signals *i.e. Signal Sleuthing* even while the car is in motion and 9m away from the attacker. Similar to its counterparts in the Reception stage, Data attacks, Signal attacks use external EM waves, but have a different focus. Communication stage Signal attacks focus on communication between the sensor and system, rather than between the environment and sensor. Instead of interfering with transduction directly, Signal attacks can mimic a sensor and provide false information to the system (Spoofing or Jamming attacks) or receive confidential information back from the system (Sleuthing attacks).Signal attacks often have greater control than Data attacks however, as the reported sensor readings themselves are modified rather than the input to the sensor.

Signal attacks can be implemented on wired systems as well. In *Ghost Talk* [30], Kune *et al.* explore the idea of Electro-Magnetic Interference (EMI) injection into analog sensors, corrupting readings from Electrocardiograms and Cardiac implantable electrical devices (CIEDs). The authors are able to cause interruptions in a CIED, as well as dominate true signals in an ECG while using only 50mW to transmit. A more recent paper using EMI to attack sensors, *Trick or Heat* [59], focuses on the effect of EMI on temperature sensors. In this work, Tu *et al.* can alter the temperature of baby incubators from a distance, even through walls, with a small hand-held device. This is done not by increasing the actual temperature near

the sensor, as in a Classical attack, nor is EM a designed vector as in Data attacks. Instead, after the victim sensor obtains a signal, EMI is introduced, and both are amplified, resulting in an incorrect reading. Tu *et al.* achieve a 60°C change with a 3.08-Watts device at a distance of 3m with this method. Both methods can either Spoof (*e.g.* create false singular set points) or Jam (*e.g.* prevent the sensor from reading correct values).

## 3 Defences

### 3.1 Reception Stage Defenses

There are two main methods to defend against Reception Stage Attacks. 1) lower an adversary's control over the environment and 2) implement checks on the data received from a sensor. These checks differ depending on the particulars of the system. For some Data Attacks encryption on the desired signals is sufficient defense, while defending against Classical Attacks may entail sensing multiple times from multiple perspectives, a technique known as sensor fusion.

#### *3.1.1 Classical Attack Defenses*

Classical Attacks on the Reception Stage are the simplest to create, but often difficult to defend against. Sensors are transducers, and as such must react to environmental changes [4]. If a malicious actor has a measure of control over the sensing environment, or can impact it sufficiently, then they can influence the data of a target sensor. Classical Attacks take advantage of this to compromise the system as a whole, often by generating information that requires an immediate, incorrect response, or to slowly build up errors over time, called a *meaningful response* [12]. Petit *et al.* discuss common countermeasures to their attacks, which could reasonably be expanded to Classical Attacks as a whole, such as sensor redundancy and random sampling [44].

*Sensor Redundancy*, or *Sensor Fusion* is a defense technique where multiple sensors, typically of the same type (and perhaps even make and model), are used to sample an environment. This redundancy in sensors can improve performance in many ways. It can increase the accuracy of a system's understanding of the environment by reducing the effects of noise. Should the sensors give vastly different readings, the system can determine that at least one of the sensors is either malfunctioning or under attack [60]. Sensor Fusion could keep a system to function while under either Spoofing or Jamming attacks. By identifying the affected sensors and ignoring their data, the attack is mitigated. There are downsides to this strategy however. One is cost: not only does the system need multiples of each sensor (typically odd multiples to break ties) but it needs to have the enhanced capabilities to receive data from all the sensors and process said data in a timely manner [60]. Another potential issue of this defense concerns Sensor Exploit Attacks. If the exact same type of device is used as part of a Sensor Fusion defense scheme, that system becomes more susceptible to Sensor Exploit Attacks. Also, while Sensor Fusion can be effective against targeted attacks, like the lidar and camera attacks described by Petit *et al.* [44], is not effective against attacks that truly change the environment. For example, take a system that monitors the

temperature in a room. Sensor redundancy cannot protect against an adversary actually raising the temperature near all sensors in the room.

Random sampling is a defense where the time at which a sensor samples its environment is randomly determined [44]. This would only be effective in situations similar to the lidar spoofing attack in which attacks heavily rely on predetermining the time at which a sensor is expecting a response, and attacking during that window. Random sampling is also ineffective against continuous attacks, as when the sensor samples matters little.

While none of the proposed methods are without downside, or applicable to every case, they do have their merits. Sensor Fusion in particular is often used in systems that are vulnerable to noise or malfunction [60], and while the high price of lidar currently prohibits the use of multiple units on a single system, autonomous vehicles often have multiple cameras or radar units to achieve similar functionality. These could foreseeably be used to determine if the system is under attack, if not outright correct for the attacks.

#### *3.1.2 Data Attack Defenses*

Data Spoofing Attacks are more difficult to defend against. Rather than attacking a singular target sensor directly, as can happen in Classical Spoofing Attacks, Data Spoofing Attacks always flood the environment of the sensor with false EM. This means that simple implementations of Sensor Fusion are ineffective, as they all receive the same information. One possible defense to this is encryption, which is standard for military use but often non-existent or weak for civilian applications [19]. As Data Spoofing Attacks affect an area continuously, random sampling and other probing techniques are likewise ineffective. GPS Systems historically use the signal's strength as a measure inversely proportional to its veracity (*i.e.* strong signals are likely closer to the receiver, and are less likely to be from satellites, hence they are more likely to be fake). There are issues with this assumption: attackers could simply use weaker signals, or be more distant to their target, and if no true GPS signal is received, then the attacker's signal would be used regardless.

A more complex implementation of Sensor Fusion can be an extremely potent defense to this however. By combining different types of receivers, it is possible to identify malicious signals, through comparing the time and errors present [42]. This requires more efforts than the method mentioned in the prior paragraph. For example, in *Drive-Me-Not* [42], Oligeri *et al.* suggest two new defenses against GPS Data Spoofing attacks. Firstly, by gaining a rough location of the vehicle through the cellular network, it is possible to determine if GPS location shifts suddenly, signifying an attack. They also find that it is possible to use multiple receivers (sensors) to give additional, redundant information about the metric being sensed. This information is in turn used to create a more accurate estimation of the state of the environment (*e.g.* the location of the source of a given GPS signal) [42]. As the actual signal origin is a satellite in orbit, all the sensors would receive it at about the same time. The input from a malicious source, on the other hand, would likely be earthbound and be received at very different times.

## 3.2 Perception Stage Defenses

Perception Stage defenses are difficult to discuss as whole, due to the wide disparity between the two subgroups of attacks within the Stage. Sensor Exploits are often a physical attack on the sensor, while Algorithmic Attacks seek to disrupt the system's underlying algorithms. What both share however, is that defenses revolve around identifying manipulated data.

### *3.2.1 Sensor Exploit Attack Defenses*

*DolphinAttack* [69] describes several methods of defense against their proposed attack, closing the Sensor Exploit they use. Methods for defending against Sensor Exploit Attacks can be broken down into two types: *Exploit Closure*, and *Exploit Detection*. Exploit Closure entails a solution that renders that particular exploit obsolete or ineffective, most commonly through a change in hardware. *DolphinAttack* for example suggests placing filters earlier in the microphone circuit, which would stop their attack from functioning. Exploit Detection on the other hand, consists of software analysis of inputs. This analysis is either conventional or through machine learning, and determines whether a system is under attack. *DolphinAttack* trains a machine learning Support Vector Machine (SVM) to classify input signals into attack signals and benign signals with 100% true positive and negative rates [69].

Further examples of Exploit Closure can by found in both Son *et al.*'s work *Rocking Drones* and Trippel *et al.*'s *WALNUT*. *Rocking Drones* [52] suggests physical countermeasures (*e.g.* foam) to prevent vibrations from reaching vulnerable sensors, an idea shared by *WALNUT* [58]. This particular solution to MEMS sensors susceptibility to high-frequency noise has existed since at least 2013. Soobramaney [53] finds that the effects of high-frequency noise are reduced almost 90% when MEMS sensors are place in nickle microfibers enclosures. One explanation for the lack of present implementation of these solutions is their lack of efficiency. Foam would increase costs and provide incomplete protection (*i.e* it would not block the entire attack, but reduce its potency). Foam in particular has its effectiveness tied to volume, meaning an increase in product size may be necessary. It also blocks airflow, potentially causing issues with heat dissipation.

### *3.2.2 Algorithmic Attack Defenses*

The machine learning community has identified a few methods to protect their innovations from adversarial examples, and therefore Algorithmic Attacks. There are a few general theories on training networks resilient to these attacks. The first, and most straightforward method, is to include adversarial examples in training [57]. Adversarial examples are often changed as minimally as possible to straddle decision lines. Training on these examples slightly adjusts the decision lines so that the examples are properly classified. Fine-tuning this way can be difficult however, as the adversarial examples need to be generated/collected. Every known method of generating adversarial attacks should be used to create examples to train on. This easily results in a tremendous amount of work to be done and data to be added [57]. Even then, new methods are likely to be discovered in the future, which would entail further fine-tuning.

This type of defense would also still be vulnerable to black box attacks as discussed by Tramèr *et al.* [57]. Black box attacks in this context are adversarial attacks where the attacker treats the target network as a "black box", with the internal methods of the target model obfuscated in such a way that the attacker only has access to responses to inputs [57]. This is as opposed to "white box" attacks, where the attacker has access to the internals of the software, which simplifies the creation of attacks. Black box attacks are more resistant to traditional Algorithmic defense methods, so Tramèr *et al.* introduce *Ensemble Training*. Examples generated by other static pretrained networks are supplied as training data in addition to examples against the given network. Tramèr *et al.* find it to be more effective at defending against black box attacks than training only on non-adversarial data and known Algorithmic attacks against a given model.

Another method of defending against Algorithmic attacks is detecting if the target data is artificially perturbed [40]. In their work, Metzen *et al.* create a sub network that determines if a given example has perturbations. Given that information, it would be possible for a system evaluating sensor data to ignore inputs that are classified as artificial [40]. This sub network could be included in any number of other networks, meaning that it could impact many fields that use machine learning and could be vulnerable to adversarial examples.

## 3.3 Communication Stage Defenses

Projection stage attacks target intra-system communication. Defenses at this Stage are often not focused on the sensor itself, but rather on guaranteeing the communication security of the system. Defenses include encryption, and traditional anti-jamming techniques and there are more unique methods being explored as well.

### *3.3.1 Sensor Commandeering Attack Defenses*

Sensor Commandeering Attacks are almost impossible to completely defend against at a sensor level as they take advantage of the sensor as an edge node. Sensor Commandeering Attacks can utilize sensor data in an unintended way, raising privacy or security concerns. To defeat such an attack, securing the system against unwanted access is required. Following this philosophy, in *Hard Disk of Hearing* the authors suggest methods to better secure hard drive firmware [32]. As the attacker needs to upload malicious firmware in order to commandeer the drives, this would prevent the attack. Preventing internally generated messages exiting the environment is also suggested, in addition to preventing external access. Zhang *et al.* additionally suggest using foam to reduce the impact of external noise on the senor. Foam is again a feasible defense, but has the same issues as when discussed under Sensor Exploit defenses. Kumar *et al.* propose countermeasures against their skill squatting attack. As skills go through a certification process, analysis may be done to detect skill names that are very similar. This could help defend against malicious attacks, as well as prevent benign errors. It is important to note here that none of the suggested defenses affect the sensor directly. Defenses against Sensor Commandeering attacks are best implemented not in a single device, but by securing the system's communications. This is because Sensor Commandeering attacks rely

on a method to send commands to and export data from the system, but many potential ways to gather said data. We believe this to be out of the scope of this paper, so we instead focused on other defense methods.

#### *3.3.2 Signal Attack Defenses*

Once again, Signal Attacks are similar to Data Attacks, with one pronounced difference. Notably, unlike Data Attacks from the Reception Stage, Signal Attacks do not have the intrinsic requirement to be open to the environment. Disabling outside access, especially for wireless communication, would greatly increase the difficulty of Spoofing, Jamming, or Sluething intra-system signals. However, the most straightforward way of doing this, by implementing wired communication system, is not always feasible. Also, even if a CPS communicated purely through wired connections, it would likely still be accessible through the internet. To protect these systems, additional strategies are required. For example, in their work, Roufa *et al.* describe using "Reliable Software Design" to prevent impossible or highly unlikely values input via a Signal Spoofing Attack from being processed as valid sensor readings [24]. Reliable Software Design also would prevent over responses to singular readings (*e.g.* there is only one message indicating an emergency state, but preceding and following packets report non emergency conditions). Reliable Software Design involves constraining input signals to reasonable levels.

This restriction on inputs can be achieved either through testing with *Fuzzing* techniques, or through state estimation. Fuzzing is a security analysis technique in which all valid input signals are sent as input to a device, and their reactions are recorded. Fuzzing tests if certain combinations of commands or sensor readings can result in exploits or unintended behaviors in a system. This could be used to determine how to restrict inputs in such a way that those unintended behaviours are avoided [27]. State Estimation on the other hand has the system predict its current state (*i.e.* environment), position in the said environment, and the positions of all other objects or actors in said environment. Commonly used in robotics to determine robot position, Kalman filters are the most common state estimators. Once a system knows its state, it can determine how likely a change in said state is [14,15]. For example, if a robot knows it is at the origin of its map and is moving along the x-axis at a speed of 1 unit, it can safely disregard any readings saying that it is now at a position of 100 units along the y-axis. Chang *et al.* further build on this and develop a filter explicitly around CPSs [8]. This allows for the system to intelligently adapt to different nodes being attacked at different times. Chang *et al.* demonstrate that they can correct for up to a certain number of incorrect readings, given by Equation (1):

$$p/2 - 1 \tag{1}$$

where $p$ is the number of sensor readings in a certain time window.

The third option is to encrypt intra-system messages. Encryption is a viable defense in many systems as it is simple to implement while greatly increases security of communications. This strategy is effective against both Sleuthing and Spoofing attacks. Roufa *et al.* [24] however cannot implement encryption in their work, as its non-negligible overhead affects the timeliness of the measurements. In many CPSs, Encryption could be effective and efficient and as many wireless communication

Table 3: A categorization of current works according to our proposed methodology.

| Category | Attacks | Defences |
|---|---|---|
| Classical | [7, 44, 48, 49] | [44, 60] |
| Data | [42, 46, 47] | [42] |
| Sensor Exploit | [52, 54, 54, 58, 69] | [52, 52, 53, 69] |
| Algorithmic Attacks | [1, 10, 31, 36, 65, 68] | [10, 31, 36, 40, 51, 57, 65, 68] |
| Sensor Commandeering | [29, 32] | [29, 32] |
| Signal | [24, 30, 59] | [8, 14, 15, 24, 27, 70] |

protocols now support it, easy to integrate. Two of the most popular methods for CPS communication, Zigbee and MQTT both support encryption [5, 6, 50, 66].

Lastly, in [70], Zhang *et al.* develop a simple method to determine if a sensor is under a Signal Attack. By randomly changing the voltage provided to a device, external sources of power can be detected. As an example, if the voltage to a sensor is set to 0v, but the sensor still reports a reading other than 0, it is likely under attack. They report that this defense can be easily and cheaply applied to almost any system and has a true positive detection rate of 100% [70]. This would be highly effective when deployed against the exploit in *Trick or Heat* [59] as it would filter malicious signals injected.

## 4 Future Research Directions

While there is a variety of research being conducted on sensor attacks, there are a number of new directions that have recently emerged. In this section a few new directions of research will be discussed, focusing on what problems they are attempting to solve, and the methods they employ. More research are expected to be performed in these emerging areas for CPS/IoT security.

### 4.1 Lexicon

While it is often difficult to fully account for all potential developments in a field, we feel that, barring the introduction of a completely new attack method, future attacks will fall under one of our proposed subcategories. Even in that case, it will still be able to be described as under one of our proposed Stages. Our support for this stance comes from Tables 3 and 4, as it demonstrates the flexibility of the current lexicon with the breadth of current work. As such, this taxonomy can adapt and grow with the field. Also of note are the Target-Attack pairs that have no current examples, as seen in Table 2. If examples of these are found, they could open new fields of research.

### 4.2 Adversarial Sleuthing

Adversarial Sleuthing Attacks are a relatively new idea. While there have been a few prior works that explore them, this could be a very large field in the future.

Table 4: An analysis of current works' available Targets.

| Category | Jamming | Spoofing | Sleuthing |
|---|---|---|---|
| Classical | [7, 44, 48, 49] | [44, 48, 49] | |
| Data | [42, 47] | [46] | |
| Sensor Exploit | [52, 58, 69] | [52, 54, 58, 69] | |
| Algorithmic Attacks | [10, 31, 36, 65, 68] | [1, 10, 31, 36, 65, 68] | |
| Sensor Commandeering | [29] | [29] | [32] |
| Signal | [24, 30, 59] | [24, 30, 59] | [24] |

Adversarial Sleuthing attacks attempt to use adversarial attacks to gain information (*i.e.* a deep learning model) from the system. An attack is conducted by providing a model with inputs that lie on decision boundaries, as it is possible to reconstruct those boundaries from responses to those queries [23]. Companies expend a large amount of effort in training proprietary models in order to sell access to them to end users. However, by selling access to these models, the models can potentially be stolen. A large amount of work remains on conducting, identifying, and defending against these attacks.

### 4.3 Lidar

Improving the resilience of lidar to Classical Attacks is another growing research area. There are two main methods being explored to achieve this, chaotic lidar and MEMS lidar. Chaotic lidar has excellent anti-jamming and anti-interference properties [11]. While not the newest idea, as it dates back at least to 2004 [37], feasible implementations remain an open question. MEMS lidar, on the other hand, has the potential to greatly lower the costs associated with lidar. Lidar units are currently prohibitively expensive, heavy, and power consuming [61]. Velodyne, possible the largest producers of commercial lidar, have recently announced much cheaper MEMS based lidar units. These "Velobit" lidar units could sell for as little as $100 [28], as opposed to the current price of their flagship product, the Puck which retails for four thousand dollars [22]. This could allow multiple lidar units per vehicle, and in turn, implementation of Sensor Fusion techniques. MEMS lidars in research are not nearly as effective as traditional ones currently are, as seen in some recent work by Yoo *et al.* [67], so the development of better performing MEMS lidars will be a pressing topic. This also carries with it new risks, as MEMS lidars have not been tested from a security standpoint.

### 4.4 Physically Uncloneable Functions

Physically Uncloneable Functions (PUFs) are a relatively recent solution to authentication problems. They leverage randomness (typically from process variation [20], though other sources exist as well [13, 33]) to create a unique function that only that device can reliably match. The challenge-response pairs will be recorded first. Later, when authenticating that device, a challenge will be sent to the device and the response will be collected. If the output matches the recorded

value, then it is the same device. As the need for secure IoT communications grows, an increasing percentage of devices will likely start implementing PUFs to meet this need [41]. Importantly, several new types of PUFs obtain their randomness through sensor readings [34, 35] rather than purely through internal process variation. While there are several strong advantages to these methods, PUFs may be now vulnerable to the full range of Sensor Attacks. Jamming Attacks of various stages on these implementations are the obvious focus of future research, as it may be possible to deny service to authentic devices.

## 5 Conclusion

In this paper, we present most recent and impact attack and defense strategies on sensors, as well as a new classification scheme to describe them. We divide these attacks into three categories, based on the method of attack. Reception Stage Attacks Target In-Band Signals, typically changing the environment or the sensing medium directly. Perception Stage Attacks target the information a sensor provides as well, but not directly. Communication Stage attacks target the system a sensor belongs to, seeking to use it for malicious purposes. We also present avenues for new research on sensor attacks and defenses, including new Targets for proposed attacks. More research efforts are expected in the area of sensor security given its wide usage in CPS and IoT domains. We hope that our survey paper will help researchers to perform systematic investigations in this area.

## Conflict of Interest Statement

On behalf of all authors, the corresponding author states that there is no conflict of interest.

**Acknowledgements** This work is partially supported by the National Science Foundation (NSF-1818500, NSF-1916175).

## References

1. Abdullah, H., Garcia, W., Peeters, C., Traynor, P., Butler, K.R., Wilson, J.: Practical hidden voice attacks against speech and speaker recognition systems. arXiv preprint arXiv:1904.05734 (2019)
2. Aggarwal, A., Kunta, S., Verma, P.K.: A proposed communications infrastructure for the smart grid. In: 2010 Innovative Smart Grid Technologies (ISGT), pp. 1–5. IEEE (2010)
3. Alladi, T., Chamola, V., Sikdar, B., Choo, K.R.: Consumer iot: Security vulnerability case studies and solutions. IEEE Consumer Electronics Magazine **9**(2), 17–25 (2020). DOI 10.1109/MCE.2019.2953740
4. Beavers, I.: Intelligence at the edge part 1: The edge node (2017). `https://www.analog.com/en/technical-articles/intelligence-at-the-edge-part-1-the-edge-node.html#` (Date last accessed: 28/11/2020)
5. Beavers, I.: Intelligence at the edge part 2: Reduced time to insight (2017). `https://www.analog.com/en/technical-articles/intelligence-at-the-edge-part-2-reduced-time-to-insight.html` (Date last accessed: 28/11/2020)

6. Beavers, I., MacLean, E.: Intelligence at the edge part 4: Edge node security (2018). https://www.analog.com/en/technical-articles/intelligence-at-the-edge-part-4-edge-node-security.html (Date last accessed: 28/11/2020)
7. Cao, Y., Xiao, C., Cyr, B., Zhou, Y., Park, W., Rampazzi, S., Chen, Q.A., Fu, K., Mao, Z.M.: Adversarial sensor attack on lidar-based perception in autonomous driving. arXiv preprint arXiv:1907.06826 (2019)
8. Chang, Y.H., Hu, Q., Tomlin, C.J.: Secure estimation based kalman filter for cyber–physical systems against sensor attacks. Automatica **95**, 399–412 (2018)
9. Chen, B., Yang, Z., Huang, S., Du, X., Cui, Z., Bhimani, J., Xie, X., Mi, N.: Cyber-physical system enabled nearby traffic flow modelling for autonomous vehicles. In: 2017 IEEE 36th International Performance Computing and Communications Conference (IPCCC), pp. 1–6 (2017). DOI 10.1109/PCCC.2017.8280498
10. Chen, Y.: Devil's whisper: A general approach for physical adversarial attacks against commercial black-box speech recognition devices. In: 29th USENIX Security Symposium (USENIX Security 20). USENIX Association, Boston, MA (2020). https://www.usenix.org/conference/usenixsecurity20/presentation/chen-yuxuan
11. Cheng, C.H., Chen, C.Y., Chen, J.D., Pan, D.K., Ting, K.T., Lin, F.Y.: 3d pulsed chaos lidar system. Opt. Express **26**(9), 12,230–12,241 (2018). DOI 10.1364/OE.26.012230. http://www.opticsexpress.org/abstract.cfm?URI=oe-26-9-12230
12. Davidson, D., Wu, H., Jellinek, R., Singh, V., Ristenpart, T.: Controlling uavs with sensor input spoofing attacks. In: 10th USENIX Workshop on Offensive Technologies (WOOT 16). USENIX Association, Austin, TX (2016). https://www.usenix.org/conference/woot16/workshop-program/presentation/davidson
13. Degada, A., Thapliyal, H.: An integrated trng-puf architecture based on photovoltaic solar cells. IEEE Consumer Electronics Magazine pp. 1–1 (2020). DOI 10.1109/MCE.2020.3019762
14. Dutta, R.G., Yu, F., Zhang, T., Hu, Y., Jin, Y.: Security for safety: A path toward building trusted autonomous vehicles. In: Proceedings of the International Conference on Computer-Aided Design, ICCAD '18, pp. 92:1–92:6. ACM, New York, NY, USA (2018). DOI 10.1145/3240765.3243496. http://doi.acm.org/10.1145/3240765.3243496
15. Dutta, R.G., Zhang, T., Jin, Y.: Resilient distributed filter for state estimation of cyber-physical systems under attack. In: 2019 American Control Conference (ACC), pp. 5141–5147 (2019). DOI 10.23919/ACC.2019.8815298
16. Foundation, N.S.: National science foundation - where discoveries begin (2019). https://www.nsf.gov/news/special_reports/cyber-physical/
17. Giechaskiel, I., Rasmussen, K.B.: Taxonomy and challenges of out-of-band signal injection attacks and defenses. IEEE Communications Surveys Tutorials pp. 1–1 (2019). DOI 10.1109/COMST.2019.2952858
18. Giraldo, J., Sarkar, E., Cardenas, A.A., Maniatakos, M., Kantarcioglu, M.: Security and privacy in cyber-physical systems: A survey of surveys. IEEE Design & Test **34**(4), 7–17 (2017)
19. Goward, D.: Mass gps spoofing attack in black sea? (2017). https://www.maritime-executive.com/editorials/mass-gps-spoofing-attack-in-black-sea (Date last accessed: 28/11/2020)
20. Guajardo, J., Kumar, S.S., Schrijen, G.J., Tuyls, P.: Fpga intrinsic pufs and their use for ip protection. In: International workshop on cryptographic hardware and embedded systems, pp. 63–80. Springer (2007)
21. Hassija, V., Chamola, V., Saxena, V., Jain, D., Goyal, P., Sikdar, B.: A survey on iot security: Application areas, security threats, and solution architectures. IEEE Access **7**, 82,721–82,743 (2019). DOI 10.1109/ACCESS.2019.2924045
22. Higgins, S.: Velodyne cuts vlp-16 lidar price to $4k (2019). https://www.spar3d.com/news/lidar/velodyne-cuts-vlp-16-lidar-price-4k/
23. Hitaj, D., Mancini, L.V.: Have you stolen my model? evasion attacks against deep neural network watermarking techniques. CoRR **abs/1809.00615** (2018). http://arxiv.org/abs/1809.00615
24. Ishtiaq Roufa, R.M., Mustafaa, H., Travis Taylora, S.O., Xua, W., Gruteserb, M., Trappeb, W., Seskarb, I.: Security and privacy vulnerabilities of in-car wireless networks: A tire pressure monitoring system case study. In: 19th USENIX Security Symposium, Washington DC, pp. 11–13 (2010)

25. Karnouskos, S.: Cyber-physical systems in the smartgrid. In: 2011 9th IEEE International Conference on Industrial Informatics, pp. 20–23. IEEE (2011)
26. Kim, N.Y., Rathore, S., Ryu, J.H., Park, J.H., Park, J.H.: A survey on cyber physical system security for iot: Issues, challenges, threats, solutions. Journal of Information Processing Systems **14**(6) (2018)
27. Kim, T., Kim, C.H., Rhee, J., Fei, F., Tu, Z., Walkup, G., Zhang, X., Deng, X., Xu, D.: Rvfuzzer: Finding input validation bugs in robotic vehicles through control-guided testing. In: 28th USENIX Security Symposium (USENIX Security 19), pp. 425–442. USENIX Association, Santa Clara, CA (2019). https://www.usenix.org/conference/usenixsecurity19/presentation/kim
28. Krok, A.: Velodyne's tiny velabit packs a big lidar punch for just $100 (2020). https://www.cnet.com/roadshow/news/velodyne-velabit-small-inexpensive-lidar-ces/
29. Kumar, D., Paccagnella, R., Murley, P., Hennenfent, E., Mason, J., Bates, A., Bailey, M.: Skill squatting attacks on amazon alexa. In: 27th {USENIX} Security Symposium ({USENIX} Security 18), pp. 33–47 (2018)
30. Kune, D.F., Backes, J., Clark, S.S., Kramer, D., Reynolds, M., Fu, K., Kim, Y., Xu, W.: Ghost talk: Mitigating emi signal injection attacks against analog sensors. In: 2013 IEEE Symposium on Security and Privacy, pp. 145–159. IEEE (2013)
31. Kurakin, Alexey, Goodfellow, Ian, Bengio, Samy: Adversarial examples in the physical world. arXiv preprint arXiv:1607.02533 (2016)
32. Kwong, A., Xu, W., Fu, K.: Hard drive of hearing: Disks that eavesdrop with a synthesized microphone. In: 2019 IEEE Symposium on Security and Privacy (SP), pp. 125–139. IEEE (2019)
33. Labrado, C., Kumar, S.D., Badhan, R., Thapliyal, H., Singh, V.: Exploration of solar cell materials for developing novel pufs in cyber-physical systems. SN Computer Science **1**(6), 1–13 (2020)
34. Labrado, C., Thapliyal, H.: Design of a piezoelectric-based physically unclonable function for iot security. IEEE Internet of Things Journal **6**(2), 2770–2777 (2019). DOI 10.1109/JIOT.2018.2874626
35. Labrado, C., Thapliyal, H., Prowell, S., Kuruganti, T.: Use of thermistor temperature sensors for cyber-physical system security. Sensors **19**(18), 3905 (2019). DOI 10.3390/s19183905. http://dx.doi.org/10.3390/s19183905
36. Li, J., Schmidt, F.R., Kolter, J.Z.: Adversarial camera stickers: A physical camera attack on deep learning classifier. In: Proceedings of the 36th International Conference on Machine Learning (2019)
37. Lin, F., Liu, J.: Chaotic lidar. IEEE Journal of Selected Topics in Quantum Electronics **10**, 991–997 (2004). DOI 10.1109/JSTQE.2004.83596
38. Martin, J.R.: Literacy in science: Learning to handle text as technology. Writing science: Literacy and discursive power pp. 166–202 (1993)
39. Martin, J.R.: Technicality and abstraction: Language for the creation of specialized texts. Writing science: Literacy and discursive power pp. 203–220 (1993)
40. Metzen, J.H., Genewein, T., Fischer, V., Bischoff, B.: On detecting adversarial perturbations. In: 5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings. OpenReview.net (2017). https://openreview.net/forum?id=SJzCSf9xg
41. Mukhopadhyay, D.: Pufs as promising tools for security in internet of things. IEEE Design & Test **33**(3), 103–115 (2016)
42. Oligeri, G., Sciancalepore, S., Ibrahim, O.A., Di Pietro, R.: Drive me not: Gps spoofing detection via cellular network. In: WiSec '19 Proceedings of the 12th Conference on Security and Privacy in Wireless and Mobile Networks (2019)
43. Park, Y., Son, Y., Shin, H., Kim, D., Kim, Y.: This ain't your dose: Sensor spoofing attack on medical infusion pump. In: 10th USENIX Workshop on Offensive Technologies (WOOT 16). USENIX Association, Austin, TX (2016). https://www.usenix.org/conference/woot16/workshop-program/presentation/park
44. Petit, J., Stottelaar, B., Feiri, M., Kargl, F.: Remote attacks on automated vehicles sensors: Experiments on camera and lidar. Black Hat Europe **11**, 2015 (2015)
45. Samsung: Family hub refrigerator (2019). https://www.samsung.com/us/explore/family-hub-refrigerator/overview/
46. Sathaye, H., Schepers, D., Ranganathan, A., Noubir, G.: Wireless attacks on aircraft instrument landing systems. In: 28th USENIX Security Symposium. USENIX Association (2019)

47. Shepard, D.P., Humphreys, T.E., Fansler, A.A.: Evaluation of the vulnerability of phasor measurement units to gps spoofing attacks. International Journal of Critical Infrastructure Protection **5**(3), 146 – 153 (2012). DOI https://doi.org/10.1016/j.ijcip.2012.09.003. http://www.sciencedirect.com/science/article/pii/S1874548212000480
48. Shin, H., Kim, D., Kwon, Y., Kim, Y.: Illusion and dazzle: Adversarial optical channel exploits against lidars for automotive applications. In: International Conference on Cryptographic Hardware and Embedded Systems, pp. 445–467. Springer (2017)
49. Shoukry, Y., Martin, P., Tabuada, P., Srivastava, M.: Non-invasive spoofing attacks for anti-lock braking systems. In: International Workshop on Cryptographic Hardware and Embedded Systems, pp. 55–72. Springer (2013)
50. Singh, M., Rajan, M.A., Shivraj, V.L., Balamuralidhar, P.: Secure mqtt for internet of things (iot). In: 2015 Fifth International Conference on Communication Systems and Network Technologies, pp. 746–751 (2015). DOI 10.1109/CSNT.2015.16
51. Sitawarin, C., Wagner, D.: Minimum-norm adversarial examples on knn and knn-based models. arXiv preprint arXiv:2003.06559 (2020)
52. Son, Y., Shin, H., Kim, D., Park, Y., Noh, J., Choi, K., Choi, J., Kim, Y.: Rocking drones with intentional sound noise on gyroscopic sensors. In: 24th {USENIX} Security Symposium ({USENIX} Security 15), pp. 881–896 (2015)
53. Soobramaney, P.: Mitigation of the effects of high levels of high-frequency noise on mems gyroscopes. Ph.D. thesis, Auburn University (2013)
54. Sugawara, T., Cyr, B., Rampazzi, S., Genkin, D., Fu, K.: Light commands: Laser-based audio injection attacks on voice-controllable systems. In: 29th USENIX Security Symposium (USENIX Security 20), pp. 2631–2648. USENIX Association (2020). https://www.usenix.org/conference/usenixsecurity20/presentation/sugawara
55. Thapliyal, H., P. Mohanty, S., Prowell, S.: Emerging paradigms in vehicular cybersecurity. IEEE Consumer Electronics Magazine **8**(6), 81–83 (2019). DOI 10.1109/MCE.2019.2928066
56. Thompson, K.D.: Cyber-physical systems (2019). https://www.nist.gov/el/cyber-physical-systems
57. Tramèr, F., Kurakin, A., Papernot, N., Goodfellow, I., Boneh, D., McDaniel, P.: Ensemble adversarial training: Attacks and defenses. arXiv preprint arXiv:1705.07204 (2017)
58. Trippel, T., Weisse, O., Xu, W., Honeyman, P., Fu, K.: Walnut: Waging doubt on the integrity of mems accelerometers with acoustic injection attacks. In: 2017 IEEE European Symposium on Security and Privacy (EuroS&P), pp. 3–18. IEEE (2017)
59. Tu, Y., Rampazzi, S., Hao, B., Rodriguez, A., Fu, K., Hei, X.: Trick or heat? manipulating critical temperature-based control systems using rectification attacks. In: CCS '19 Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security (2019). DOI 10.1145/3319535.3354195
60. Um, D.: Massive sensor array fault tolerance: Tolerance mechanism and fault injection for validation. Journal of Robotics (2010). http://dx.doi.org/10.1155/2010/745834
61. Wang, Y., Chao, W., Garg, D., Hariharan, B., Campbell, M., Weinberger, K.Q.: Pseudo-lidar from visual depth estimation: Bridging the gap in 3d object detection for autonomous driving. CoRR **abs/1812.07179** (2018). http://arxiv.org/abs/1812.07179
62. Woodward-Kron, R.: More than just jargon – the nature and role of specialist language in learning disciplinary knowledge. Journal of English for Academic Purposes **7**(4), 234 – 249 (2008). DOI https://doi.org/10.1016/j.jeap.2008.10.004. http://www.sciencedirect.com/science/article/pii/S1475158508000799
63. Woody, C.: The navy's 4th accident this year is stirring concerns about hackers targeting us warships (2017). https://www.businessinsider.com/hacking-and-gps-spoofing-involved-in-navy-accidents-2017-8 (Date last accessed: 28/11/2020)
64. Xiang, C., Qi, C.R., Li, B.: Generating 3d adversarial point clouds. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
65. Xiao, Q., Chen, Y., Shen, C., Chen, Y., Li, K.: Seeing is not believing: Camouflage attacks on image scaling algorithms. In: 28th USENIX Security Symposium (USENIX Security 19), pp. 443–460. USENIX Association, Santa Clara, CA (2019). https://www.usenix.org/conference/usenixsecurity19/presentation/xiao
66. Yang, B.: Study on security of wireless sensor network based on zigbee standard. In: 2009 International Conference on Computational Intelligence and Security, vol. 2, pp. 426–430. IEEE (2009)

67. Yoo, H.W., Druml, N., Brunner, D., Schwarzl, C., Thurner, T., Hennecke, M., Schitter, G.: Mems-based lidar for autonomous driving. e & i Elektrotechnik und Informationstechnik **135**(6), 408–415 (2018). DOI 10.1007/s00502-018-0635-2. https://doi.org/10.1007/s00502-018-0635-2
68. Yuan, X., Chen, Y., Zhao, Y., Long, Y., Liu, X., Chen, K., Zhang, S., Huang, H., Wang, X., Gunter, C.A.: Commandersong: A systematic approach for practical adversarial voice recognition. In: 27th {USENIX} Security Symposium ({USENIX} Security 18), pp. 49–64 (2018)
69. Zhang, G., Yan, C., Ji, X., Zhang, T., Zhang, T., Xu, W.: Dolphinattack: Inaudible voice commands. In: Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, pp. 103–117. ACM (2017)
70. Zhang, Y., Rasmussen, K.: Detection of electromagnetic interference attacks on sensor systems. In: IEEE Symposium on Security and Privacy (1997)
71. Zhang, Z., Gong, S., Dimitrovski, A.D., Li, H.: Time synchronization attack in smart grid: Impact and analysis. IEEE Transactions on Smart Grid **4**(1), 87–98 (2013). DOI 10.1109/TSG.2012.2227342