The Security of Autonomous Driving: Threats, Defences, and Future Directions

Kui Ren, *Fellow, IEEE*, Qian Wang, *Senior Member, IEEE*, Cong Wang, *Senior Member, IEEE*, Zhan Qin, *Member, IEEE*, and Xiaodong Lin, *Fellow, IEEE*

***Abstract*—Autonomous vehicles (AVs) have promised to drastically improve the convenience of driving by releasing the burden of drivers and reducing traffic accidents with more precise control. With the fast development of artificial intelligence and significant advancements of IoT technologies, we have witnessed the steady progress of autonomous driving over the recent years. As promising as it is, the march of autonomous driving technologies also faces new challenges, among which security is the top concern. In this article, we give a systematic study on the security threats surrounding autonomous driving, from the angles of perception, navigation, and control. In addition to the in-depth overview of these threats, we also summarise the corresponding defence strategies. Furthermore, we discuss future research directions about the new security threats, especially those related to deep learning based self-driving vehicles. By providing the security guidelines at this early stage, we aim to promote new techniques and designs related to AVs from both academia and industry, and boost the development of secure autonomous driving.**

***Index Terms*—Autonomous Vehicles, Security, Sensors, In-Vehicle Systems, In-Vehicle Protocol.**

I. Introduction

Since the CMU Navlab group built the first computer-controlled vehicles for automated driving in 1984 [1], many researchers have promoted autonomous vehicle (AV) developments. One noteworthy breakthrough was in 1994, when the group of UniBw Munich and the group of Daimler-Benz have co-developed an AV that could reach the speed up to 130 km/h [2]. That very AV could automatically track different lane markings and decide when to change between lanes.

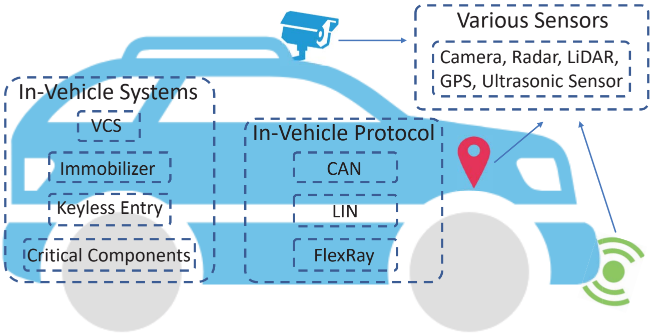


Fig. 1: The three types of attack surfaces of an AV.

Recently, with the prevalence of artificial intelligence (AI) and Internet of Things (IoT) technologies, autonomous driving has gained steady improvements, and is getting more and more intelligent to precisely sense environments in the real world, quickly analyze the sensor data, and autonomously make complex decisions. In the foreseeable future, AVs are widely believed to be one of the most popular AI applications in people's daily lives. For instance, IHS Markit predicts that the annual sales of AVs will exceed 33 million in 2040 [3].

As promising as it is, the fast development of autonomous driving technologies also faces new challenges, among which security is the top concern. Specifically, before the wide adoption of AVs on the road facing realistic traffic conditions, the security and trustworthiness of AVs must be guaranteed through all kinds of technical assurances. As we know, AVs are often equipped with varieties of functionally-rich sensors, such as cameras, Radars, GPS, etc., to perceive its surrounding environments. The data captured by sensors are fed into the AV's computing system for rounds of complicated processing and calculations in order to enable the autonomous control of the vehicle, including the braking mechanism as well as the engine. Hence, AVs heavily rely on the sensor data to make the right driving decisions, which inevitably enlarges the potential threat surface and incurs serious security risks from sensors [4]. In addition, the systems responsible for in-vehicle access and control (e.g., voice controllable systems and keyless entry systems), and the protocols indispensable for in-vehicle network operations (e.g., Controller Area Networks (CAN)), also require effective security countermeasures against various attacks whilst providing critical and decisive functionalities for AVs.

Kui Ren's research is supported in part by the NSFC under Grant 61772236 and Zhejiang Key R&D Plan under Grant 2019C03133. Qian Wang's research is supported in part by the NSFC under Grants 61822207 and U1636219, the Equipment Pre-Research Joint Fund of Ministry of Education of China (Youth Talent) under Grant 6141A02033327, the Outstanding Youth Foundation of Hubei Province under Grant 2017CFA047, and the Fundamental Research Funds for the Central Universities under Grant 2042019kf0210. Cong Wang's research is supported in part by the Research Grants Council of Hong Kong under Grants CityU C1008-16G, and by the NSFC under Grant 61572412. Zhan Qin's work is supported in part by Major Scientific Research Project of Zhejiang Lab under Grant 2018FD0ZX01. *(Corresponding author: Kui Ren)*

K. Ren and **Z. Qin** are with the Institute of Cyberspace Research, the College of Computer Science and Technology, and Alibaba-Zhejiang University Joint Institute of Frontier Technologies, Zhejiang University, P.R. China. Email: {kuiren, qinzhan}@zju.edu.cn.

Q. Wang is with the School of Cyber Science and Engineering, Wuhan University, P. R. China. Email: qianwang@whu.edu.cn.

C. Wang is with Department of Computer Science, City University of Hong Kong, Hong Kong SAR, P. R. China. Email: congwang@cityu.edu.hk.

X. Lin is with the School of Computer Science, University of Guelph, Canada. Email: xlin08@uoguelph.ca.

TABLE I: Various types of sensors.

Sensors	Signal	Working Area	Principle	Usage
GPS	Microwave	Global	Passive	Navigation
LiDAR	Infrared laser	Medium range	Active	Pedestrian detection Collision avoidance
MMW Radar	Microwave	Long range	Active	Collision avoidance Adptive cruise control
Ultrasonic Sensor	Ultrasound	Proximity	Active	Parking assistance
Camera	Visible light	Short range	Passive	Traffic sign recognition Lane detection Obstacle detection

In Fig. 1, we briefly categorize the broadly defined security threats surrounding an autonomous vehicle into three classes. The first type contains different kinds of sensors equipped in the vehicle, which perceive the surrounding road conditions. The sensor data are further used to guide driving without human intervention. Once they are jammed or spoofed by false signals, the autonomous driving car will lose the ability of precisely sensing the environments. The second one includes various in-vehicle access and control systems, e.g., the vehicle immobilizer, the keyless entry system, critical control components, and the voice controllable system (VCS). These in-vehicle systems guarantee the security of physical car access and human-vehicle interaction. If these in-vehicle access and control systems are broken, it would lead to critical security flaws and incidents to AV, as serious as a matter of life and death. The last is about the in-vehicle network protocols, such as the Local Interconnect Network (also known as LIN), the CAN mentioned above, FlexRay, and more. Any vulnerability of the protocols could be exploited through telematics modules and further magnified remotely by the attackers to illegally control the vehicles.

In this article, we first give a systematic study on the categories of security threats, particularly from the perspectives of perception, navigation, and control. We then respectively summarize the corresponding defense strategies. Last but not least, we highlight a few crucial open problems, especially those related to deep learning based self-driving vehicles, and discuss future research directions. We believe that our work can encourage new techniques and designs related to defenses against threats posed to AVs, and push forward the frontier and future development of secure autonomous driving.

II. Potential Threats of Sensors

When AVs cruise on the road, it is essential for AVs to sense the environmental circumstances precisely, due to the lack of drivers' control. Various sensors, like GPS, ultrasonic sensor, LiDAR (Light Detection and Ranging), and MMW (Millimeter Wave) Radar, are "eyes" indispensable for AVs. Fig. 2 illustrates sensors embedded in AVs, and Table I shows a generic description of these sensors as well as the corresponding usage scenarios for them. Armed with sensors, AVs can achieve environment perception, collision avoidance, obstacle/pedestrian recognition, navigation, etc. Considering such high reliance on sensors, once sensors are blinded, or even maliciously controlled, it may cause lethal catastrophes.

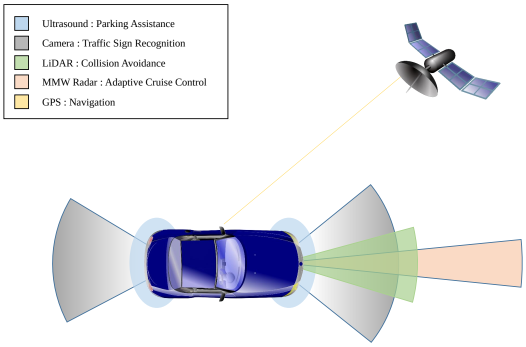

Ultrasound : Parking Assistance
Camera : Traffic Sign Recognition
LiDAR : Collision Avoidance
MMW Radar : Adaptive Cruise Control
GPS : Navigation

Fig. 2: Sensors embed in autonomous vehicle.

In this section, we introduce various types of attacks against most common sensors in AVs and provide some corresponding defense strategies.

A. Various Types of Attacks

1) Attacks against GPS: GPS is indispensable in the navigation of AVs. Relaying on position got from GPS and aided by a precise map, AVs can choose an optimized, shortest path from one location to another location, even without any previous knowledge. This is essential for AVs to work correctly without the assistance of drivers. Meanwhile, this also exposes vulnerabilities to malicious attackers.

The attacks toward GPS have been studied widely in the past decade [5]–[11]. Existing attacks, like [6], [9], [12], [13], demonstrate that the GPS attacks are practical. There are mainly two kinds of GPS attacks: *spoofing* and *jamming*. GPS signals from satellites are weak due to long-distance traveling [6]. Hence, the jamming attack is much easy to be launched by using stronger signals in the same frequencies . In the following part, we focus on introducing spoofing attacks since they are more threatening than jamming.

Spoofing aims to drag victims off to incorrect position (and time) by fabricating spurious signals which deviate the correct position of victims. A simple strategy that could be easily detected is to first jam the victim's GPS receiver and make it lose the lock of the signals. Then the attacker sends a much powerful spurious signal to take over the signals from satellites [6]. This attack is detectable since the victim's GPS loses signals or encounters an abrupt change [7]. A much more

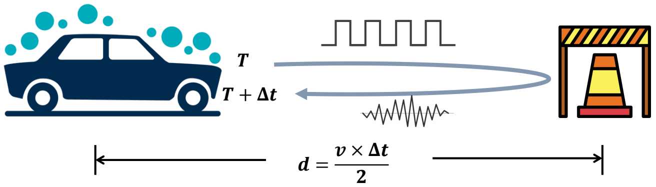


Fig. 3: Working principle of sensors using round-trip time of signal to calculate distance. Here, v is the speed of the signal in the air (like, sound: $340m/s$, electromagnetic wave and light: $3 \times 10^8 m/s$).

sophisticated strategy needs the attacker to be more patient [7], [10], [11]. To mount an attack toward the victim, the spurious signals of the attacker should synchronize on the signals from the satellite. After synchronization, the attacker increases the power of spurious signals, which makes the victim's GPS lock on spurious signals. Then, the attacker can manipulate the position of the victim by changing spurious signals. Other advanced strategies, like nulling, canceling GPS signals by emitting negative signals accordingly [8], could also be used to launch stealthy attacks.

In the aforementioned attack strategies, they mainly focus on how to take over the victim's GPS signals. A recent attack, proposed by Zeng *et al.* [9], utilizes elegantly selected fake position to guide the victim vehicle to drive into a pre-defined location when the victim is using the navigation system (e.g., Google Map). This attack may be caught when the driver involves, but is much more efficient with driverless AVs.

2) Attacks against LiDAR: LiDAR is an active sensing device, and compared with the camera, it can work during the whole day, neglecting the illumination condition. It also can be used to recognize signs, lanes, etc., since these infrastructures have retro-reflective surfaces [14]. With these strengths, almost all of the AVs, excluding Tesla, are armed with LiDAR for circumstance perception [15]. LiDAR senses obstacles around by rotating the transceiver, emitting infrared lasers, and calculating the distance of obstacles by measuring round-trip time of reflected lasers (like Fig. 3). Several existing works demonstrate that LiDAR is vulnerable to intentional attacks.

Petit *et al.* [16] first introduce an attack targeting at LiDAR embedded in AVs. In their attack, the attacker uses a transceiver to receive the laser pulse sent from LiDAR and relay the received signal to another transceiver, which sends a spurious signal back to the LiDAR after delaying it in a pre-defined time interval. By controlling the delayed time interval and frequency of sending the spurious signal back, the proposed attack could achieve injecting several obstacles in fixed positions. Later, Shin *et al.* [15] extend Petit's attack, which allows injecting closer fake obstacles. They leverage the fact that the LiDAR scans the environment through rotating laser transceiver and the light travels much faster than the rotating speed of LiDAR. Therefore, the attacker can receive laser pulse in advance, and then immediately relay the laser pulse to another transceiver in other angles of LiDAR. This allows the attacker to make fake obstacles closer to him. Besides, they also introduced a jamming attack by sending the same frequency laser to LiDAR.

3) Attacks against MMW Radar: The system structure of MMW Radar is very similar to LiDAR, as shown in Fig. 3, except the emitted signal. The MMW Radar emits the microwave whose wavelength is longer than laser emitted by LiDAR [17]. Comparing with LiDAR, the MMW Radar is robust to poor weather conditions, e.g., storms, fog, and dust [18]. However, due to the longer wavelength of MMW Radar, the MMW Radar has lower resolution and shorter detectable range. Currently, the MMW Radar is equipped in vehicles of Tesla. In DEF CON 2016, Yan *et al.* demonstrate practical attacks against Tesla Model S leveraging vulnerability of the MMW Radar [4]. They conduct experiments on the jamming attack by sending the same waveform signals to the MMW Radar to cause lower *signal-noise ratio* (SNR), and thus successfully launch the spoofing attack by carefully modulating signals similar to the MMW Radar. In their study, it is concluded that the experimental result is prominent, especially when Tesla works on the autopilot mode. As for AVs merely relaying on the MMW Radar to achieve obstacles recognition and collision avoidance, it is indeed an non-trivial threat.

4) Attacks against ultrasonic sensor: The ultrasonic sensor transmits and receives the ultrasound which is sound waves with high frequencies that human beings cannot hear. Normally, most people cannot sense the sound with a frequency higher than 18 kHz [19]–[21]. It leverages the propagation time of reflected ultrasonic pulses to calculate the distance to the nearest obstacles (see in Fig. 3). In AVs, this capability enables ultrasonic sensors to be used for automatic or semi-automatic parking. Similarly, spoofing and jamming are two kinds of attacks threatening the ultrasonic sensor.

The spoofing attack tries to utilize the carefully crafted ultrasound to create a forged obstacle. In [4], the spoofing attack can create pseudo-obstacles when there is no real one in the detection range. Conversely, if there are more obstacles, this attack can easily cause confusions during AV's decision-making procedures. Beyond this work, in [22], Xu *et al.* further demonstrate the effectiveness of the *Adaptive Spoofing* attacks by creating virtual obstacles against off-the-shelf sensors as well as those on-board ones equipped by AVs.

Simpler but still threatening, jamming attack aims to decrease the SNR of ultrasonic sensors by continuously emitting ultrasound. In [4], [22], Audi, Volkswagen, Tesla, and Ford are tested, and the result shows that jamming attack can mislead the cars when the driver does not receive any warning about obstacles. Another experiment in [22] shows that jamming attacks work effectively against Tesla cars in the self-parking mode, as well as those in the summon mode. In both cases, the jammed car may ignore and hit obstacles.

Moreover, the approaches of acoustic quieting such as cloaking and acoustic cancellation can be used for ultrasonic sensor attacks.

5) Attacks against camera: In AVs, cameras are used in many scenarios such as traffic sign recognition [33], lane detection [34], obstacle detection [35], etc. Fatal accidents may occur when the performance of the cameras is significantly degraded, which is caused by attacks against the cameras.

Petit *et al.* [16] get the efficiency of blinding MobilEye C2-270, a commercial camera system, with several light sources.

TABLE II: Summary of defense strategies of attacks against sensors.

Defense strategies		Principle	Modification	Extra hardwares	Reference
GPS	Signal check	Checking signal inherent characters (like strength)	No	Case dependent	[23]–[29]
	Cryptography	Encryption and authentication	Signal	No	[30]–[32]
LiDAR	Redundancy	Multiple LiDAR	No	Yes	[15], [16]
	Fusion	Multiple kinds sensors	No	Yes	[15], [16]
	Modification	Reducing receiving angle, pulsing laser multiple times, shortening pulsing time interval	Device	No	[15], [16]
	Randomization	Randomly rotating or pulsing signal	Device/Signal	No	[15], [16]
MMW Radar	Sanity check	Impossibility of high-power microwave in real world	No	No	[4]
	Redundancy	Multiple MMW Radars	No	Yes	[4]
	Fusion	Multiple kinds sensors combination	No	Yes	[4]
	Randomization	Randomly pulsing signal	Signal	Yes	[4]
Ultrasonic sensor	Sanity check	Impossibility of high-power ultrasound in real world	No	No	[4]
	Redundancy	Multiple ultrasonic sensors	No	Yes	[4], [22]
	Fusion	Multiple kinds sensors combination	No	Yes	[22]
	Randomization	Randomly pulsing	Signal	Yes	[22]
Camera	Redundancy	Multiple cameras cooperation	No	Yes	[16]
	Special optics	Filter and photochromic lenses	Device	Yes	[16]

It shows that leveraging a laser or LED matrix could blind the camera. Petit *et al.* also prove that in the laboratory environment, the attacker could continuously switch the light on and off to confuse the camera.

In [4], Yan *et al.* successfully blind the camera by aiming the LED and the laser light at the camera directly. In particular, aiming the LED light at the calibration board, which is a substitute of realistic scenes, would lead to the concealment of specific areas. According to the results, radiating a laser beam, even for just a short period of seconds in very close distance (less than half a meter) against an AV's camera, would cause irreversible damage and thus disrupt the corresponding autonomous procedures.

B. Defense Strategies

In this section, we list countermeasures proposed against attacks of sensors. Table II presents a summary of defense strategies against attacks aiming at different sensors. Detailed descriptions of these strategies are introduced as follows.

1) Defense strategies for GPS: Numerous countermeasures have been proposed to prevent GPS-targeted attacks.

For instance, the spurious signals appear different from signals transmitted from the satellites. It could be used to identify GPS attacks. Warner *et al.* detect attacks based on the signal strength, the time interval between signals, and the clock information of signals [23]. Wesson *et al.* utilize distortions of correlation function in the receiver to identify validity of GPS signal [24]. Other works [25]–[29] check the direction of arrival (DoA), which uses the antenna array to alleviate the attacks since DoA of GPS signals would show a distinct carry-phase compared with spoofing signals.

Other methods introduce cryptographic techniques into GPS signals for attack defense. O'Hanlon *et al.* [30] propose to encrypt GPS L1 P(Y) code to judge whether a spoofing attack is happening. Authentication strategies [31], [32] are also proposed to ensure the signals are authentic, e.g., the navigation message authentication (NMA), which embeds signature in the signal from the satellites.

Alternatively, works in different fields of study can be combined to achieve protection, like distance bounded protocol [36], [37]. They measure and ensure the distance between entities using cryptographic tools or computer vision techniques by comparing road signs and buildings of the current position.

2) Defense strategies for LiDAR: To resist attacks targeting at LiDAR, authors in [15], [16] list the following defense strategies.

Modifying how LiDAR emits and receives laser is a promising way. If the attacker wants to perform attack successfully, the spurious laser should be synchronized with the laser from LiDAR. Emitting laser pulse multiple times (like, three times) in one direction is efficient against an attacker who is not in sync with the laser of LiDAR. In addition, since LiDAR only accepts laser from a specific angle during rotating, reducing receiving angle can mitigate the effect of attacks, but it also is a trade-off of LiDAR's sensitivity [38]. Another countermeasure is to reduce the LiDAR receiving time, which reduces the probing range of LiDAR. To ensure certainty, LiDAR defines the receiving time, within which LiDAR receives incoming lasers. Specifically, reducing receiving time leaves fewer chances for an attacker to perform attacks, but also enables the lasers, which is reflected from a further object, to be taken invalid.

Another strategy is to introduce randomness while LiDAR is working. Since LiDAR rotates the transceiver for scanning circumstance around, LiDAR is designed to rotate in a random speed and emit laser to a random direction to prevent attacks proposed by [15]. In addition, making laser from LiDAR more unpredictable by emitting randomized signals or emitting signals in a random pulse interval is another efficient way against attackers.

Finally, redundancy of LiDARs or multi-sensor fusion allows AV to correct readings of LiDAR(s). It increases the cost and complexity of the attacker, and meanwhile introduces extra cost to customers due to installing new devices. In addition, in the non-overlapped area, the attack can still be launched [15].

3) Defense strategies for MMW Radar: The authors in [4] gave a discussion about how to confront attacks toward Radar.

TABLE III: Representative targets of vehicle immobilizer attacks.

Target name	Security scheme	Vulnerabilities	Type of attack	Reference
Digital Signature System	Challenge-response protocol	Short secret key	Spoofing	[39]
Passive Keyless Entry and Start system	LF RFID tag	Passive, fake proximity	Relay	[40]
Hitag2	48-bit LFSR & a non-linear filter	malleability, lack of PRG	Key-recovery	[41]
Megamos	96-bit secret key & PIN code	malleability, lack of PRG, invertibilty	Key-recovery	[42]
Security protocol stack	AES	Key storage method	Fault injection	[43]

TABLE IV: Characteristics of typical attacks against keyless entry systems.

Attack type	Vulnerable system	Implementation complexity	Countermeasure	Defense complexity	Reference
Jamming	All remote	Easy	Be careful	Easy	[44]–[47]
Replay	Fix-code remote	Medium	Cryptography	Easy	[44], [48], [49]
Relay	Passive remote	Hard	Electromagnetic shield	Medium	[40], [44]
Cryptographic analysis	Active remote	Hard	Improve cryptography	Hard	[50]–[52]

Firstly, they believe the jamming attack is easily detectable since the jamming-like signal is rare in the real-world. When Radar detects such signals, there is a high possibility that Radar is under attack. Then, for resisting spoofing attacks, they recommend introducing randomness into Radar's signal. Finally, they suggest sensor fusion strategy, namely using different sensor reading to correct each other.

4) Defense strategies for ultrasonic sensor: In [22], two approaches are proposed to defend against ultrasound sensor attacks. The first one leverages the idea of shifting the parameters of waveforms, and thus makes it possible to authenticate the physical signals. The second approach uses two or more sensors to detect attacks, recover the abilities of obstacle detection or localize attackers. According to the experiment, the two countermeasures can effectively defend against ultrasonic sensor attacks.

5) Defense strategies for camera: Because of the vulnerability of the camera caused by optical characteristics, it is difficult to build a completely secure camera system. Nevertheless, Petit *et al.* [16] give some possible countermeasures. Redundancy, removable near-infrared-cut filters, and photochromic lenses can provide proper protection from different aspects despite the fact that they may have limitations or introduce new problems.

III. Potential threats of In-Vehicle Systems

To facilitate the access control of the automated vehicles, it is possible to deploy certificates to support the authentication of the controllers. Next, we present the potential security threats related to the in-vehicle access control systems, including vehicle immobilizer, keyless entry systems, control components and voice controllable systems. After that, we investigate the countermeasures of the vulnerabilities.

A. Various types of attacks

Here, we introduce the attacks which target the in-vehicle authentication systems.

1) Vehicle Immobilizer Attack: As a common anti-theft device, the electronic vehicle immobilizer realizes electronic security to prevent the start of the vehicle engine, unless the corresponding key fob, also known as a transponder or physical security token, is used. In recent years, quite a few widely used transponders in-car immobilizer industry are discovered as insecure [39]–[42]. Table III shows the characteristics and vulnerabilities of these schemes. Among them, Hitag2 and Megamos are both broken due to the weaknesses in the designs of the cipher [41], [42], including the lack of pseudo-random number generators and the shortness of cipher's internal states, in comparison with the private key. In [41], vulnerabilities of the Hitag2 cryptographic scheme are revealed, three cryptanalytic attacks are proposed to retrieve the private keying materials. By exploiting the malleability of the cipher and the lack of a good-quality pseudo-random number generator (PRG), the first attack manages to read the identity of the transponder and recover keystream. The second attack is more generic, which can be utilized to break the generic cipher designs using linear feedback shift registers, which is also known as LFSR. It is used to bypass the read protection mechanism from the security token, and still successfully retrieves the private keying materials in just 60 seconds. The final attack attempt leverages the key observation that there are dependencies across different authentication sessions with the immobilizer of the car. Such dependencies can also be exploited to extract the private keying materials, although at a slightly lower rate, in the order of minutes, compared to the second attack.

Similar to the reference [41], three attacks aiming at Megamos are proposed in [42]. Apart from the vulnerabilities mentioned before, the first attack leverages two new observations for the retrieval of the private keying materials: 1) the cipher state successor can be invertible; 2) the multi-step authentication protocol reveals bits of plaintext in the final steps. The second proposed attack simply uses publicly known default PIN code to retrieve the private keying materials within a timeframe of half an hour. Moreover, the attacks in both [41] and [42] use Time-Memory Tradeoff (TMTO) to reduce the time cost of secret key retrieval from days to hours, minutes, and even seconds.

In 2010, an open protocol stack for the security of the car immobilizer system was proposed. It controls the authentication functionality and uses off-the-shelf AES encryption. Security vulnerabilities of the protocol stack are theoretically analyzed in [53], and there are several types of implementation attacks [43], [54], [55]. In reference [53], the authors discuss the possibility of several attacks, including relay attacks, tracking, denial-of-service attacks, replay attacks, spoofing attacks and hijack of the communication sessions.

Apart from the potential vulnerabilities discussed in [53], Takahashi et al. propose an invasive fault attack [43], which exploits the secret key storage specifics of the security protocol stack. Specifically, the secret key is replicated three times in the key fob storage space.

For key fob authentication, the secret key's all three copies are used one by one, thus enhancing the robustness and the availability of the immobilizer system. However, through fault injections, the adversary can alter a part of the data at the physical address of the very first secret key, while trying out the remaining part of the secret key. By repeating the process of fault injection and guessing with the other two secret keys, the adversary can retrieve most bits of the secret key for AES, and eventually retrieve the entire secret key with exhaustive search.

2) Keyless Entry Systems Attack: While the vehicle immobilizer system focuses more on starting the engine, the attacks towards keyless entry systems mainly aim to break into the car.

Entry system guarantees the safety of the properties inside the vehicle. With the technological development, there are three types of car keys: 1) traditional physical key; 2) remote active keyless entry; 3) remote passive keyless entry and start (also known as PKES). Features of the systems are displayed in Fig. 4. The earliest *physical key* only allows physically unlocking the door and starting the engine. The key should be inserted into the lock hole, and there is no electrical communication between the key and the vehicle. The *remote active keyless entry system* is embedded into a key fob. "Active" means that there are interactions between the user of the key fob and the entry system. When opening/closing the vehicle's door, the user needs to press a button to generate signals from a Radio Frequency (RF) transmitter. Then the car receives the signals and authenticates the data with cryptographic methods. However, some users may find searching for the key and pressing the button disturbing. The remote PKES system solves this problem by realizing keyless entry. In this system, the user only needs to approach the car and the door will open automatically. Moreover, the PKES also supports automatic engine start, which means that when the driver is seated, the engine is activated. The communication between the key and the car relies on an LF RFID tag for short-distance (≤2m) auto entry and start, and a fully-fledged UHF (Ultra High Frequency) for remote-distance (≤100m) door unlocking.

Here, we list several possible attacks on the keyless entry systems. The features of the attacks are presented in Table IV.

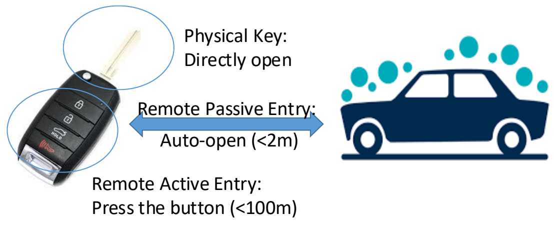


Fig. 4: Characteristics of typical entry systems.

Jamming attacks. Due to the wireless communication between the key and the car during the open or close process, there are chances for the adversaries to jam the signal when the user closes the door. When the user presses the "close" button, the attacker can generate an interference signal to jam the locking signal. The user is unaware of the fact that the door remains unlocked and leaves so that the attacker can break in. This method is reported in the news [45]. Technically, the jamming method could be regarded as Intentional Electromagnetic Interference (IEMI) [47]. Beek *et al.* [46] carry out a detailed robustness study against interference through a series of experiments about systems with such keyless entry. According to their experiment setting, the key fob is 2m away from the car and continuous wave interferences with the range from 420 MHz up to 460 MHz are generated to test the robustness of the original signals. Results show that the two keyless systems in their experiment are sensitive to interference with a bandwidth of 5 MHz and 4 MHz, respectively and the interference can be generated in a distance of 100m, which provides convenience for attackers. Furthermore, the jamming attack does not require any cryptographic or chip analysis, making it easy and cheap to launch.

Replay attacks. A typical scenario of replay attacks is that the thief eavesdrops and records the back and forth exchange signal between a common key transponder and a corresponding receiver on the car. For an unsupervised car, the attacker can replay the recorded signal and open the door. However, this kind of attack is not effective for most of the latest car models because of the adoption of rolling code for the key fob. In short, the rolling code keeps an incremental counter and the encrypted code will change whenever the button is pressed, which makes sure that an attacker cannot easily guess the code and replay it. Nevertheless, replay attacks could be integrated with other attacks. For example, the attacker can jam and record the valid "close" code, then replay the code after a break-in, after which the car is appropriately locked. Furthermore, the attacker can keep eavesdropping, jamming, recording the valid signals until she gets the expected signal (for example, the owner gets frustrated when keeping failing to open the door and leaves, as described in [48]) and then the attacker can record the latest valid "open" code and open the car [49].

Relay attacks. Relay attacks have been widely researched and are prevalent within communication systems [56]–[58]. A

TABLE V: Hidden voice attack techniques.

Attack name	Method	Attacker's knowledge	Range	Inaudible (Y/N)	Reference
Hidden voice commands	Machine learning	Black & white box	< 3.5m	N	[59]
Dolphin Attack	Hardware	White box	2cm$\sim$175cm	Y	[60]
Lipread	Hardware	White box	< 8m	Y	[61]
Audio adversarial examples	Machine learning	White box	N/A	N	[62]

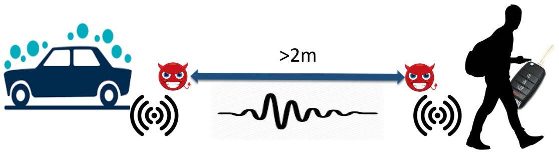


Fig. 5: The model of relay attack.

relay attack can break the distance restriction in the communication system by placing devices between the signal sender and receiver and relaying the signals between them. As for our topic, the remote passive keyless entry and start system enables the car owner, without looking for a key, to unlock the car, which is convenient, however, vulnerable to relay attacks. Recall that in a PKES system, when the key is near to the car, *e.g.*, 2 meters, the door will passively open. Moreover, when the receiver detects that the key is inside the car, the engine starts and then the driver can step on the gas and go. Nevertheless, the protocol only depends on the communication signals, not on the physical key. Alrabady *et al.* [44] first exploit this weakness and described a two-thief model conceptually. Later, Francillon *et al.* [40] follow the idea and launched the relay attack on the PKES system. The attacker places an antenna close to the car door and another antenna near the car owner and then the antennas can transfer the signals, as shown in Fig. 5. Although physically the distance between the key and the door is not short enough to complete the protocol, the signals from the key transferred by the antennas fool the receiver in the car and the challenge-response protocol can complete. In their experiments, the attack is effective when the key-side antenna is within 8m (in the best situation) from the key and the distance of the antennas can be up to 3000 km, which is practically effective. A typical scenario introduced by the authors is that in the parking lot, say when the car owner leaves the parked car, the car becomes unsupervised and often out of sight. After that, one attacker attempts to move the car-side antenna close to the door, and the other attacker with key-side antenna can tail after the owner. In this way, it is possible to establish the relayed communication between the key fob, which is with the owner, and the parked car, which is away from the key. Such communication could work as if the key and the car are spatially close. Note that this relay attack does not need to interpret or manipulate the signals, and thus the cryptographic authentication could not help in such scenarios.

Cryptographic analysis attacks. The aforementioned attacks mainly aim at physical-layer communication, and they do not consider the analysis of the signals. Another line of attacks can be described as cryptographic analysis attacks against the encryption and code algorithms in higher layers. The earlier generation of the remote keyless system does not provide an authentication mechanism. The code is fixed, and cryptography is not involved. To enable authentication, in [63], the authors propose to use rolling code techniques, which are effective in defending against the most straightforward replay attack. However, the Keeloq scheme is proven to be insecure against cryptographic analysis [50] and side-channel attacks [51]. Apart from the inherent vulnerability in cryptographic protocol, the PCBs (Printed Circuit Boards) in the entry systems can also be analyzed by attackers to steal the information in the firmware. A solid research [52] investigates the widely-used VW group remote control systems and succeeds in cloning a targeted remote control by analyzing the cryptography used in the schemes and eavesdropping the signals of the victim, after which the adversary can break into the car. The attack takes advantage of the vulnerability that most remote control systems share the same master key. If the attacker gets the PCBs and takes a deep insight into the firmware, there are chances that she can figure out the structures of the codes, the details of the cryptographic algorithms or even the encryption key. With the global used master key, the attacker can then get the counter by eavesdropping and decrypting the signal from the victim. The authors also propose an attack on the Hitag2. The correlation attack can recover the secret key in minutes.

3) Voice Controllable Systems Attack: Voice controllable systems (VCS) are widely applied in in-vehicle access control and the enhancement of the driving experience. As shown in Fig. 6, usually, a VCS is constructed of three basic modules: 1) the voice capture module that records the ambient voices and digitalizes it before the next stage, 2) the speech recognition module that operates on the digitalized signals and uses machine learning techniques to further understand the instructions, and 3) the command execution to perform the designated command. Recently, researchers focus on the inaudible voice attacks, which are incomprehensible to humans but recognizable to VCS as commands, thus control the systems without being detected [59]–[62]. The attack schemes are summarized in Table V.

In [60], the authors propose DolphinAttack, which exploits the hardware properties of the audio circuits to insert hidden voice commands that are inaudible by the human. The key idea of DolphinAttack is to modulate the regular voice signal, which is often at low-frequency band, on an ultra-high frequency carrier, also known as an ultrasonic carrier. Doing so ensures the inaudibility of the voice commands. Therefore, amplitude modulation is utilized to exploit the nonlinear property of MEMS (Micro Electro Mechanical Systems) microphones which can down-convert high-frequency

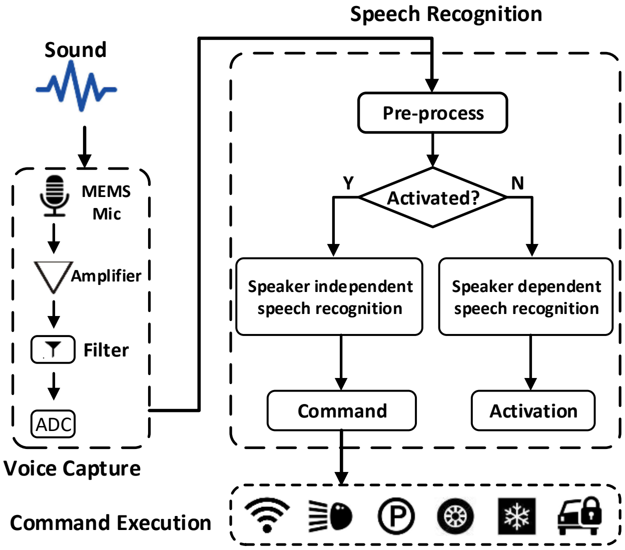


Fig. 6: The architecture of a typical VCS that takes voice commands as inputs and executes corresponding commands.

signals to lower frequencies. Thus, with a carefully designed input signal, the microphone with nonlinearity can recover the wanted voice control signal. Though effective on major speech recognition systems, DolphinAttack [60] requires vicinity to the target devices, e.g., the attack can be launched from a distance of 5ft to Amazon Echo. This is because the speaker with the same nonlinearity can also produce audible lower frequencies while playing the higher frequencies. Thus DolphinAttack must be operated at low power, which constrains the range of the attacks. To enlarge the range of a successful breach, the inaudible attack system LipRead is devised in [61]. To tackle the contradiction between the long range and inaudibility, the authors use multiple speakers. The "signal leakage" from an individual speaker is limited to a narrow low-frequency band. By solving a min-max optimization problem, the aggregated leakage can be kept under the human auditory response curve. Following this methodology, the maximal attack distance is improved to 8 meters in [61]. Furthermore, the researchers also propose effective defense mechanisms against such attacks, through identifying nonlinearity traces, which is a feature often preserved in signals with commands of hidden voice. In spite of their efficiency and innovation, both DolphinAttack [60] and LipRead [61] require the attack devices to emit ultrasound signals [64], which means the adversary must carry a customized device. Since the range of the inaudible voice command attack is still restrained, the transmitter that produces special signals could still be noticed by the targeted victim. This limitation hinders the feasibility of the hidden voice command attacks.

B. Defense Strategies

There are several possible strategies to defend the aforementioned attacks. The physical-layer attacks, especially the attacks based on signal interference and signal transmission, can be easily prevented by intentionally paying more attention, where the countermeasures can be carried out by individuals. More complicated defenses include cryptographic update, extra authentication and scheme modification, etc. Here, we introduce these defenses and encourage readers to explore more countermeasures.

1) Leave with caution: The simplest way to prevent the jamming attack is to make the car owner assure that the door is locked before she leaves, as advised in [65]. For remote confirmation, light or sound could be used to indicate that a car is locked properly. However, the countermeasure is only effective for the jamming-only attack. If the attacker can replay the "unlock" signal, the door will lock appropriately and the remote confirmation method becomes insufficient. Hence, the basic countermeasure is to make sure that the doors are locked before moving away from the vehicle.

2) Block the source signal: An instant way to avoid relay attack is to shield the key when it is not being used [40]. If the key is shielded by a box, the antenna on the key-side cannot receive and transmit the signal from the key fob. However, this method brings inconvenience to the user because when she wants to get into the car, she needs to take out the key, which disables the most attracting advantage of the passive remote keyless system. A similar countermeasure is to remove the battery from the key so that the key will not send and receive signals. Also, this method impacts the functionality of PKES.

3) Distance bounding: Distance bounding is a helpful method to defend against the relay attack [66]–[68]. In a distance bounding algorithm, rapid exchanges of messages are conducted in order to verify the distance between the parties. Only if the distance between the key fob and the car is proven valid will the door open automatically. Francillon *et al.* [40] give the sketch of the distance bounding solution to deal with relay attack on PKES and discussed the implementation details. The reason that the relay attack can work is that the antennas can transfer the signals even if the parties are distant. However, this may bring latency and a long latency is not allowed in a distance bounding protocol.

4) Authentication improvement: Quite a few attacks on the vehicle immobilizer and the keyless entry systems are aiming at cracking the cryptographic protocols. One solution is to improve the authentication mechanism. In other words, a more secure cryptographic algorithm and key distribution method should be used in nowadays remote keyless entry systems. Fortunately, Amir *et al.* [69] have already presented a more secure RKE (Remote Keyless Entry) architecture, which can resist side-channel attacks. To improve the security level for the control systems in practice, vehicle manufacturers may make efforts to implement the state-of-the-art secure mechanism on the new-designed cars.

5) Hidden Voice Detection: To prevent VCS against hidden voice commands, various strategies have been introduced, including device enhancement, signal analysis [60], audio turbulence [70], [71] and liveness detection [72], [73], etc.

As introduced in Section III-A3, in its principle, the attacks on VCS use the electronic devices to produce inaudible voice commands, while the normal commands of controllers come from live speakers. With this observation, the general defense is to analyze the signals via standard signal processing techniques, thus differentiating the attack signals from the normal

TABLE VI: The comparison of in-vehicle protocols.

Bus	CAN	LIN	FlexRay
Applications	Engine control, airbags, antilock break system, body system	Body control(dorr locking, lights, seat belts)	Multimedia and X-by-wire (drive-by-wire, brake-bywire, steering-by-wire)
Data Rate	1 MBit/s	20 kBit/s	10 MBit/s
Exposure	Big	Little	medium
Architecture	Multi-Master	Single-Master	Multi-Master
Access Control	CSMA/CA	Polling	TDMA
Kind	Event-triggered	Subbus	Time-triggered
Redundancy	None	None	2 Channels
Transfer Mode	Asynchronous	Synchronous	Asynchronous/Synchronous
Physical Layer	Dual-Wire	Single-Wire	Optical Fiber Dual-Wire

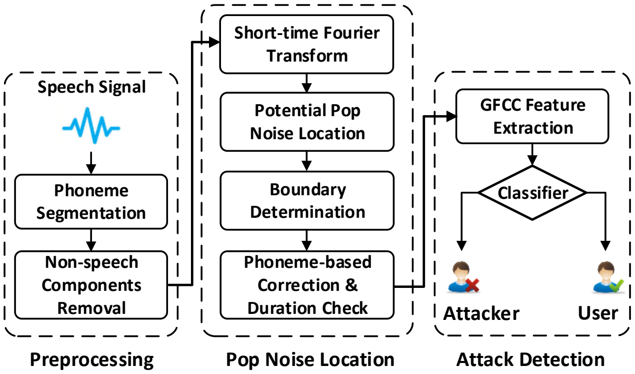


Fig. 7: The detection scheme based on pop noise.

ones. As introduced in [60], concrete methods in this line of work can be classified as hardware-based ones and software-assisted ones. A typical hardware-based approach would be producing command cancellation. While on the software level, it is also possible to extract unique features from signals. In [70], similar ideas are discussed in the name of audio turbulence and audio squeezing.

In [72], a specific method with targeted scenarios is elaborated. Their basic idea is to identify the sound source. On that account, the authors devise a liveness detection approach by leveraging the pop noise. For instance, the pop noise could be an explosive burst caused by the breath of a live speaker. When replayed by a speaker, the adversarial audio cannot reproduce the burst of airflow without real human breath. Hence, the pop noise can be used to distinguish the adversarial audio from live commands. As shown in Fig. 7, the defense scheme consists of three different phases, which are preprocessing of signal, the location of pop noise, and the detection of the attack, respectively.

IV. Potential threats of In-Vehicle Protocol

With more and more requirements on automobiles to pursue a comfortable and smart driving environment, the number of on-board electronic devices increases dramatically, in which Electronic Control Units (ECUs) are most significant. Hence, the communications of ECUs is becoming more and more complex. It is vital to take into account the security of in-vehicle network communications [74], for example, in the Local Interconnect Network (LIN), the Controller Area Network (CAN) or the FlexRay. We present a brief introduction and comparison of CAN, LIN, and FlexRay in Table VI. It exposes vehicles to various attacks. Attackers can take arbitrary control of multiple vehicles or even kill them with a remote connection. Moreover, autonomous vehicles exacerbate these threats because of the lack of human driving and monitoring. This section presents the most recent in-vehicle network attack and defense methods.

A. Various Types of Attacks

The in-vehicle protocols, including CAN, LIN, and FlexRay, have drawn much attention from the attackers. In particular, the research on the security of CAN bus has received extensive attention. Recent studies have demonstrated that many attacks have been launched against in-vehicle protocols, like spoofing and DoS. We comprehensively analyze and introduce these attacks on CAN, LIN, and FlexRay protocol.

With the increase of on-board electronic devices, CAN protocol began the dominant communication method of motor vehicles. CAN bus is famous for its advantages of multiple masters, low cost, and high transmit rate [75]. However, CAN protocol was designed without security consideration at the beginning, and thus is vulnerable to some attacks, such as injecting false messages into CAN bus. Here are five major security threats inherent in CAN protocol.

- Broadcast nature. CAN protocol broadcasts the packet into all nodes. Hence, all packets can be snooped by the malicious node, which paves the way for malicious attacks on CAN, such as replay attacks.
- No authenticator fields. Without authenticator fields, a node cannot tell whether a packet is from a malicious node. Thus, malicious nodes can easily impersonate other nodes and tamper with data.
- No authenticator fields. Without authenticator fields, a node cannot tell whether a packet is from a malicious node. Thus, malicious nodes can easily impersonate other nodes and tamper with data.
- Defective arbitration scheme. The CAN protocol utilizes the CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance) methods and the priority-based arbitration scheme. Hence, malicious nodes can achieve the DoS attack by repeatedly playing the high-priority message.
- Dangerous interface. The most dangerous and significant interface is the on-board diagnostic (OBD)-II port, and it is also a federally mandated port in the United States.

TABLE VII: The different studies related to attacking on CAN.

Access Method	Attack Method	Attack Vehicles	Attack Results	Year	Reference
Direct Access	Spoofing	-	braking, stopping the engine, disable the brakes	2010	[76]
		Ford Escape and Toyota Prius	steering, acceleration, braking and display	2013	[77]
	DoS	Honda Accord and Hyundai Sonata	shutting down healthy ECU	2016	[78]
Remote access	Spoofing	-	controling brakes, engine, locks, uploading firmware, and exfiltering data	2011	[79]
		Jeep Cherokee	steering, braking	2015	[80]
		Tesla Model S and Model X	compromising CAN bus, achiving arbitrary code execution	2017	[81]

Then the CAN bus is accessed directly via the OBD-II port, which is used for reprogramming and diagnostic. These properties have abstracted much attention from many attackers.

Moreover, researchers show that CAN bus is more vulnerable than we expected. Karl *et al.* [76] demonstrate that the attacker could be in control of many automotive functions, e.g., it can stop the engine, disables the brakes, brakes the wheels, changes the display and so on, by directly injecting false CAN message through OBD-II. As the automotive driving system gets smarter, it has more and more I/O interfaces and is therefore potentially vulnerable to attack. Stephen *et al.* [79] comprehensively analyze the vehicle attack surface. It empirically demonstrates that the post-compromise I/O interface can be remotely triggered to control any vehicle function and filtering data. Similar to [76], [77] can control the steering, acceleration, braking and display on Ford Escape and Toyota Prius.

However, in the aforementioned attack methods in order to control the vehicle at will, reverse-engineering is required to understand the meaning of packets. However, reverse-engineering is hard and relies on different vehicles. Cho *et al.* [78] present a new kind of DoS attack, called the bus-off attack, exploiting the error-handling method that automatically isolates misbehaving or defective ECUs in CAN protocol. Specifically, bus-off attack iteratively injects false messages to deceive a healthy ECU into believing itself defective. Finally, it can trigger the CAN fault confinement, forcing the attacked ECU or more severely the entire network to close down. This attack does not require reverse-engineering packets that make it easy to mount.

The most famous attack is [80], which leads to a recall of about 1.4 million Jeeps. This is the first time a vehicle has been attacked through a remote connection without direct access to the bus. Nie *et al.* [81] remotely attack the Tesla Model by utilizing a complex chain of vulnerabilities, including previous vulnerabilities in the IT field. We summarize the above attacks in Table VII.

1) LIN: The LIN bus is an inexpensive serial communications protocol, which is intended to complement the CAN bus. The LIN bus is commonly used for vehicle body control, such as seats and doors. Although the threat of the LIN bus attack is not as significant as that of CAN bus attack, it also poses considerable security risks to high-speed cars. A brief introduction about security on the LIN bus is provided [82].

The LIN bus is broadcast and comprises master-slave nodes (one master and typically up to 15 slaves). The master node initiates a header containing the identifier (ID), and at most one slave node replies to the given identifier. Because the master initiates all communication, it is not necessary to implement a collision detection algorithm. Hence, the defective error-handling mechanism is used to attack LIN bus [83]. In the LIN error-handling mechanism, the normal sender node stops the packet transfer when the collision is detected. It creates an opportunity for the malicious nodes to send a false message to replace the valid one.

2) FlexRay: FlexRay has a reputation for next-generation automotive communications protocols, but it is not used as a replacement for CAN bus and LIN bus. It meets the future communication demands of high data rates, low-cost, high stability, and flexible data communication. FlexRay is a time-triggered protocol. It employs Time Division Multiple Access (TDMA) in order to prevent bus contention and achieves real-time redundant communication. Like the CAN bus, FlexRay bus also lacks data confidentiality and authentication mechanism. Hence, it is easy to perform the read and spoof attacker action [84].

B. Defense Strategies

As mentioned above, CAN has some major limitations. Therefore, most current defense methods are to circumvent these limitations.

1) Gateway Installation: Gateway is a common and effective defense method. Wolf *et al.* [85] introduce the gateway in the automotive bus system. Within the in-vehicle network, the central gateway-based architecture has been transformed into a backbone-based architecture [86]. The gateway transfers the message from various ECUs, which also provides the functions of protocol conversion, message verification, and error protection. It acts as the interface for vehicle communication. In addition, the gateway also includes the firewall mechanism, which increases the difficulty of access to the bus through vehicle attack surfaces. For example, the attack message cannot be directly injected into the in-vehicle bus by the OBD-II port. The gateway can manage the data transmission between the low-speed bus and high-speed bus.

2) Encryption Scheme: One of the necessary steps to enhance bus communication security is to encrypt data transmission. Wolf *et al.* [85] utilize cryptographic tools and present a secure communication scheme for automobiles which combined symmetric and asymmetric encryptions to achieve high performance and adequate security. Lu *et al.* [87] present encryption and obfuscation techniques to prevent code tampering and data sniffing. The Obfuscation is a cost-effective method against reverse engineering. Moreover, to effectively encrypt the data transmission between external memory and the ECU internal memory, the on-the-fly decryption is introduced. Woo *et al.* [88] utilize AES-128 and keyed-hash MAC together for encryption and authentication, finally reducing the bus load.

3) Authentication Mechanism: Nilsson *et al.* [89] present delayed data authentication using MACs. However, problems associated with the communication cost also arise. A vote-based technique [90] integrated with time-triggered authentication is proposed to reduce authentication latency and improve bandwidth efficiency. This technique uses unanimous voting on the message validity and value among a set of nodes to decrease the probability that a per-packet forgery succeeds. Instead of independently authenticating each node, Groza *et al.* [91] present the lightweight broadcast authentication protocol (LiBrA-CAN), which splits the authentication key between any two groups of nodes. The assumption that the compromised nodes are only a minority is practical. MAC is considered on the AUTomotive Open System ARchitecture (AUTOSAR) in 2017. MAC can effectively prevent unauthorized CAN messages because the attackers do not have the authentication key. However, the attack on MAC is implemented [92].

Moreover, the error frame transmission [93]–[95] is proposed to prevent unauthorized CAN message. The basic idea is that when a node detects an unauthorized message, the node sends an error frame immediately to override it before the receiving node receives it.

4) Anomaly Detection: Anomaly detection [96]–[98] on CAN is developed from anomaly packet detection on the Internet. Larson *et al.* [99] introduce security specifications for ECU behavior and communications and presented some example specifications. Müter *et al.* [100] introduce a batch of sensors of different types for anomaly detection to detect the characteristics of in-vehicle networks, such as frequencies and load. As most normal CAN packets arriving at a fixed frequency, Taylor *et al.* [101] propose an inter-packet timing measurement algorithm over a sliding window. The following SVM can detect anomalies with satisfying results. Moreover, some works utilize the inimitable physical characteristics of the message, including voltage and signal, to achieve authentication and detect malicious ECU [102], [103], [103].

V. Future Directions

Fully automated driving, which can operate on any road at any time with no human interaction [104], has been viewed as the holy grail of autonomous vehicles that would vastly revolutionize the industry of automotive and bring engaging transporting experience in our daily life. Recently, the success of self-driving systems based on deep learning algorithms has, for the first time, shed light on a practical and very promising direction for achieving such an ultimate goal. In general, such a system consists of a trained machine learning model and many advanced sensors. The trained model serves as the brain for the vehicle to "see, hear, and make reasonable driving decisions" all on its own. Though it is intriguing and convenient to delegate all of the control rights to the vehicle itself, fatal incidents could also occur if the self-driving system goes wrong. In addition, updating the self-driving system requires new incoming training data from the vehicle, which potentially leaks information of the daily routine as well as other private information. This section first summarizes and discusses the new severe threats in future generations of autonomous vehicles, i.e., fully automated self-driving vehicles. Then, it provides possible defense strategies to make fully automated self-driving vehicles safer.

A. New Security Threats

We now introduce the new security threats in fully automated self-driving vehicles.

1) Trained Model Errors: Self-driving vehicles rely on a deep learning model based perception system to identify objects and drive autonomously on their own. However, due to algorithm bugs or model errors, the perception system in a self-driving vehicle may misclassify objects and lead to fatal car incidents [105]. One recent example is the Uber self-driving vehicle incident [105], [106] a self-driving vehicle misclassified a pedestrian as other objects and failed to break in time to prevent the collision. Therefore, the first new threat in fully automated self-driving vehicles would be the errors in the implemented in the trained deep learning model. In such a safety-critical system, it is crucial to make sure that the trained model for object identification and classification is robust and bug-free.

2) Adversarial Examples: Besides the errors in the trained deep learning models, misclassification can also be triggered by specially crafted adversarial inputs. This new threat is much more serious than inherent model bugs/errors, because an outside attacker can trick the self-driving vehicle to actively deviate from the correct actions by inputting adversarial (image) examples [71], [107], [108]. For example, as demonstrated in a recent work [109], an attacker can deceive a self-driving vehicle by deliberately generating toxic signs alongside the road, causing the trained deep learning model to misclassify signs and drive recklessly. Consequentially, such a severe threat from adversarial examples, if not carefully addressed, would lead to potentially life-threatening consequences. Moreover, adversarial examples under black-box attack models [110] where no parameter information of the target deep learning model is required, pose even severer threats to self-driving vehicles. On that account, an attacker could train an adversarial network [111] to generate more advanced adversarial examples for attacking, which makes the defense for adversarial examples more challenging.

3) Model Training Privacy: Training an accurate deep learning model for self-driving vehicles requires a very large dataset of road images or real driving videos as learning inputs. Thus, continuously contributing learning inputs collected from self-driving vehicles is essential to make the deep learning model robust and accurate in real deployments. However, most current model training infrastructures are centrally structured, which means that the input data from self-driving vehicles are transferred to a centralized server transparently. Since the contributed learning dataset is closely related to daily lives, it might reveal sensitive information of people, e.g., routine, locations, etc [112]. Besides, according to a recent study [113], the trained deep learning models can also leak sensitive information of the data contributors. More specifically, an attacker can leverage the models memorization of unique or rare sequences in the learning inputs, and extract useful information from the trained models. Therefore, how to collect data for model training while preventing privacy leakage needs a thorough study before deploying self-driving vehicles in the real world.

4) Model Execution Threats: In the implementation of fully automated self-driving systems, many new designs of hardware like TPU (Tensor Processing Unit), GPU, ASIC (Application-specific Integrated Circuit) and FPGA (Field Programmable Gate Array), are incorporated inside the autonomous vehicles [114] for achieving lightweight and efficient deep learning. Existing systems construct a new central operating component inside the self-driving vehicle for controlling the hardware to work seamlessly without mutual intervention. However, similar to in-vehicle systems, such a central operating component might subject to malicious attacks, e.g., malware injections, and thus can hardly guarantee the correctness of model executions. Therefore, such an operating system should be modeled as an untrusted environment whose attack surface may be easily leveraged by the attackers, and advanced defense mechanisms should be deployed for ensuring execution integrity in self-driving vehicles.

B. Defense Strategies

We briefly discuss the strategies for defending against the aforementioned security threats in fully automated self-driving vehicles.

1) DNN Robustness Improvement: In order to improve the robustness and reduce errors of deep learning models, we could conduct comprehensive testing on those trained models. Existing testing of trained models for self-driving vehicles is mostly based on either 1) measuring and analyzing the recognition error over a newly-inputted learning dataset, or 2) running real driving tests on the road and giving attention to disengagements, i.e., the incidents where the self-driving vehicle cannot decide [105]. In the future, the testing procedures of deep learning models could be more automatic. When an error is detected, the system can automatically retrain the model for improving accuracy. Also, the testing can be extended to real-time, so that errors can be continuously monitored during model execution, and automatically patched to further enhance driving safety.

2) Adversarial Example Defense: To address adversarial examples that can trick the deep learning model into behaving what the attacker wants, the first possible defense strategy is to pre-process or filter of input data so as to detect and eliminate the adversarial examples before executing. For example, we can use standard blurring techniques, e.g., Gaussian blur [115], to let our trained model "escape" from adversarial examples. Another useful defense strategy is to generate adversarial examples or detect potential adversarial examples using data mining methods, and then re-train the model with these generated adversarial examples to make the deep learning model more robust. Lastly, we can also try to enhance the interpretability of underlying deep learning models. Using the poison traffic sign adversarial example as an example, we can let the self-driving vehicle give the reasoning of the made decisions, by explaining what it "sees" in the input image [116]. In this way, we can closely monitor the model execution procedures and detect incorrect driving decisions in time to prevent fatal accidents.

3) Data Privacy Preservation: To provide privacy of the contributed training data, one possible strategy is to leverage the emerging federated learning architecture [117] to train and update the deep learning model. With this privacy-enhanced architecture, the sensitive inputs for model training never leave the self-driving vehicles, and only model parameter updates are sent to the server for model converging and updating. As a result, the private training data from all self-driving vehicles can be protected during the model training and updating process.

Specific configurations could be set to minimize memorization during training to prevent data leakage. In particular, one potential strategy for defending against memorization is by adding the chosen noise carefully to each gradient update during learning, so as to make the trained models differentially private [118]–[120]. In this way, we can effectively hide the occurrence of some private information in the trained models, and can thus prevent an attacker from extracting them by abusing model memorization.

4) Execution Integrity Enhancement: To enhance the execution integrity inside the self-driving vehicle, we can leverage Trusted Execution Enclave (TEE) [121] to construct a secure and isolated environment for executing integrity-critical driving decisions and learning. Currently, available TEE constructions are implemented in CPUs manufactured by Intel [122] and AMD [123]. In the near future, we can further design enclaves for new hardware, from GPUs to ASIC circuits, so that both performance and execution integrity are guaranteed at the same time in a self-driving vehicle.

VI. Conclusion

In this article, we have conducted a comprehensive and systematic survey on the security threats, defenses, and future directions of autonomous vehicles. First, we have targeted three types of potential attacks against the existing autonomous vehicles, focusing on security threats of sensors, in-vehicle systems, and in-vehicle protocols, respectively, and gave corresponding defense strategies. Second, we have further dived into the future of autonomous vehicles, i.e., self-driving vehicles based on deep learning algorithms, and elaborated the new security threats therein. Specifically, we have focused on the security threats of the deep learning model, including system errors, adversarial examples, model privacy, and hardware security. We have also presented potential practical defense strategies for all the mentioned new threats, aiming to provide a useful security guideline to boost the development of fully automated self-driving vehicles.

References

[1] C. Thorpe, M. Herbert, T. Kanade, and S. Shafter, "Toward autonomous driving: the CMU Navlab. ii. architecture and systems," *IEEE expert*, vol. 6, no. 4, pp. 44–52, 1991.

[2] T. Luettel, M. Himmelsbach, and H.-J. Wuensche, "Autonomous ground vehicles concepts and a path to the future," *Proceedings of the IEEE*, vol. 100, no. Special Centennial Issue, pp. 1831–1839, 2012.

[3] "Autonomous vehicle sales to surpass 33 million annually in 2040, ihs markit says," https://technology.ihs.com/599099/autonomous-vehicle-sales-to-surpass-33-million-annually-in-2040-enabling-new-autonomous-mobility-in-more-than-26-percent-of-new-car-sales-ihs-markit-says, 2018.

[4] C. Yan, W. Xu, and J. Liu, "Can you trust autonomous vehicles: Contactless attacks against sensors of self-driving vehicle," *DEF CON*, vol. 24, 2016.

[5] J. V. Carroll, "Vulnerability assessment of the US transportation infrastructure that relies on the global positioning system," *The Journal of Navigation*, vol. 56, no. 2, pp. 185–193, 2003.

[6] J. S. Warner and R. G. Johnston, "A simple demonstration that the global positioning system (GPS) is vulnerable to spoofing," *Journal of Security Administration*, vol. 25, no. 2, pp. 19–27, 2002.

[7] T. E. Humphreys, B. M. Ledvina, M. L. Psiaki, B. W. O'Hanlon, and P. M. Kintner, "Assessing the spoofing threat: Development of a portable GPS civilian spoofer," in *Proc. of Radionavigation laboratory conference proceedings*, 2008.

[8] M. L. Psiaki and T. E. Humphreys, "GNSS spoofing and detection," *Proceedings of the IEEE*, vol. 104, no. 6, pp. 1258–1270, 2016.

[9] K. C. Zeng, S. Liu, Y. Shu, D. Wang, H. Li, Y. Dou, G. Wang, and Y. Yang, "All your GPS are belong to us: Towards stealthy manipulation of road navigation systems," in *Proc. of 27th USENIX Security Symposium (USENIX Security)*, 2018, pp. 1527–1544.

[10] N. O. Tippenhauer, C. Pöpper, K. B. Rasmussen, and S. Capkun, "On the requirements for successful GPS spoofing attacks," in *Proc. of the 18th ACM conference on Computer and communications security (CCS)*, 2011, pp. 75–86.

[11] T. Nighswander, B. Ledvina, J. Diamond, R. Brumley, and D. Brumley, "GPS software attacks," in *Proc. of the ACM Conference on Computer and Communications Security (CCS)*, 2012, pp. 450–461.

[12] A. J. Kerns, D. P. Shepard, J. A. Bhatti, and T. E. Humphreys, "Unmanned aircraft capture and control via GPS spoofing," *Journal of Field Robotics*, vol. 31, no. 4, pp. 617–636, 2014.

[13] J. Bhatti and T. E. Humphreys, "Hostile control of ships via false GPS signals: Demonstration and detection," *NAVIGATION: Journal of the Institute of Navigation*, vol. 64, no. 1, pp. 51–66, 2017.

[14] "Lidar and autonomous technology," http://velodynelidar.com/newsroom/lidar-autonomous-technology/, 2016.

[15] H. Shin, D. Kim, Y. Kwon, and Y. Kim, "Illusion and dazzle: Adversarial optical channel exploits against lidars for automotive applications," in *Proc. of International Conference on Cryptographic Hardware and Embedded Systems (CHES)*. Springer, 2017, pp. 445–467.

[16] J. Petit, B. Stottelaar, M. Feiri, and F. Kargl, "Remote attacks on automated vehicles sensors: Experiments on camera and lidar," *Black Hat Europe*, vol. 11, p. 2015, 2015.

[17] M. I. Skolnik, "Radar handbook," 1970.

[18] "Lidar vs. radar," https://www.sensorsmag.com/components/lidar-vs-radar, 2018.

[19] M. Zhou, Q. Wang, K. Ren, D. Koutsonikolas, L. Su, and Y. Chen, "Dolphin: Real-time hidden acoustic signal capture with smartphones," *IEEE Transactions on Mobile Computing*, vol. 18, no. 3, pp. 560–573, 2019.

[20] M. Zhou, Q. Wang, T. Lei, Z. Wang, and K. Ren, "Enabling online robust barcode-based visible light communication with realtime feedback," *IEEE Transactions on Wireless Communications*, vol. 17, no. 12, pp. 8063–8076, 2018.

[21] M. Zhou, Q. Wang, J. Yang, Q. Li, F. Xiao, Z. Wang, and X. Chen, "Patternlistener: Cracking android pattern lock using acoustic signals," in *Proc. of the 25th ACM conference on Computer and communications security (CCS)*, 2018, pp. 1775–1787.

[22] W. Xu, C. Yan, W. Jia, X. Ji, and J. Liu, "Analyzing and enhancing the security of ultrasonic sensors for autonomous vehicles," *IEEE Internet of Things Journal*, vol. 5, no. 6, pp. 5015–5029, 2018.

[23] J. S. Warner and R. G. Johnston, "GPS spoofing countermeasures," *Homeland Security Journal*, vol. 25, no. 2, pp. 19–27, 2003.

[24] K. D. Wesson, D. P. Shepard, J. A. Bhatti, and T. E. Humphreys, "An evaluation of the vestigial signal defense for civil GPS anti-spoofing," in *Proc. of Radionavigation Laboratory Conference Proceedings*, 2011.

[25] A. Konovaltsev, M. Cuntz, C. Haettich, and M. Meurer, "Autonomous spoofing detection and mitigation in a GNSS receiver with an adaptive antenna array," in *Proc. ION GNSS*, 2013.

[26] P. Y. Montgomery, T. E. Humphreys, and B. M. Ledvina, "A multi-antenna defense: Receiver-autonomous GPS spoofing detection," *Inside GNSS*, vol. 4, no. 2, pp. 40–46, 2009.

[27] M. L. Psiaki, B. W. O'hanlon, S. P. Powell, J. A. Bhatti, K. D. Wesson, and T. E. Humphreys, "Gnss spoofing detection using two-antenna differential carrier phase," in *Proc. of Radionavigation Laboratory Conference Proceedings*, 2014.

[28] S. Daneshmand, A. Jafarnia-Jahromi, A. Broumandan, and G. Lachapelle, "A low-complexity GPS anti-spoofing method using a multi-antenna array," *aa*, vol. 2, p. 2, 2012.

[29] M. L. Psiaki, S. P. Powell, and B. W. Ohanlon, "GNSS spoofing detection using high-frequency antenna motion and carrier-phase data," in *Proc. of the ION GNSS+ Meeting*, 2013, pp. 2949–2991.

[30] B. W. O'Hanlon, M. L. Psiaki, J. A. Bhatti, D. P. Shepard, and T. E. Humphreys, "Real-time GPS spoofing detection via correlation of encrypted signals," *Navigation*, vol. 60, no. 4, pp. 267–278, 2013.

[31] A. J. Kerns, K. D. Wesson, and T. E. Humphreys, "A blueprint for civil GPS navigation message authentication," in *Proc. of IEEE/ION Position, Location and Navigation Symposium (PLANS)*, 2014, pp. 262–269.

[32] L. Scott, "Anti-spoofing & authenticated signal architectures for civil navigation systems," in *Proc. of the 16th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GPS/GNSS)*, 2001, pp. 1543–1552.

[33] C. Bahlmann, Y. Zhu, V. Ramesh, M. Pellkofer, and T. Koehler, "A system for traffic sign detection, tracking, and recognition using color, shape, and motion information," in *Proc. of IEEE Intelligent Vehicles Symposium (IV)*, 2005, pp. 255–260.

[34] H.-Y. Cheng, B.-S. Jeng, P.-T. Tseng, and K.-C. Fan, "Lane detection with moving vehicles in the traffic scenes," *IEEE Transactions on intelligent transportation systems*, vol. 7, no. 4, pp. 571–582, 2006.

[35] C. Häne, T. Sattler, and M. Pollefeys, "Obstacle detection for self-driving cars using only monocular cameras and wheel odometry," in *Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2015, pp. 5101–5108.

[36] S. Brands and D. Chaum, "Distance-bounding protocols," in *Proc. of Workshop on the Theory and Application of of Cryptographic Techniques*, 1993, pp. 344–359.

[37] K. B. Rasmussen and S. Capkun, "Realization of rf distance bounding," in *Proc. of the 19th USENIX Security Symposium (USENIX Security)*, 2010, pp. 389–402.

[38] "The basics of microscopy," http://www.vanosta.be/microscopy.htm, 1995.

[39] S. Bono, M. Green, A. Stubblefield, A. Juels, A. D. Rubin, and M. Szydlo, "Security analysis of a cryptographically-enabled RFID device," in *Proc. of the 14th USENIX Security Symposium (USENIX Security)*, 2005.

[40] A. Francillon, B. Danev, and S. Capkun, "Relay attacks on passive keyless entry and start systems in modern cars," in *Proc. of the Network and Distributed System Security Symposium (NDSS)*, 2011.

[41] R. Verdult, F. D. Garcia, and J. Balasch, "Gone in 360 seconds: Hijacking with Hitag2," in *Proc. of the 21th USENIX Security Symposium (USENIX Security)*, 2012, pp. 237–252.

[42] R. Verdult, F. D. Garcia, and B. Ege, "Dismantling megamos crypto: Wirelessly lockpicking a vehicle immobilizer," in *Proc. of the 22th USENIX Security Symposium (USENIX Security)*, 2013, pp. 703–718.

[43] J. Takahashi and T. Fukunaga, "Implementation attacks on an immobilizer protocol stack," in *Proc. of 11th Embedded Security in Cars Conference Europe (escar Europe)*, 2013.

[44] A. I. Alrabady and S. M. Mahmud, "Analysis of attacks against the security of keyless-entry systems for vehicles and suggestions for improved designs," *IEEE Trans. Vehicular Technology*, vol. 54, no. 1, pp. 41–50, 2005.

[45] "Lock 'jammers' steal cars in a click," https://www.telegraph.co.uk/news/uknews/crime/9623150/Lock-jammers-steal-cars-in-a-click.html, 2012.

[46] S. van de Beek, R. Vogt-Ardatjew, and F. Leferink, "Robustness of remote keyless entry systems to intentional electromagnetic interference," in *Proc. of International Symposium on Electromagnetic Compatibility (EMC)*, 2014, pp. 1242–1245.

[47] W. A. Radasky, C. E. Baum, and M. W. Wik, "Introduction to the special issue on high-power electromagnetics (hpem) and intentional electromagnetic interference (iemi)," *IEEE Transactions on Electromagnetic Compatibility*, vol. 46, no. 3, pp. 314–321, 2004.

[48] "Jam intercept and replay attack against rolling code key fob entry systems using rtl-sdr," http://spencerwhyte.blogspot.ca/2014/03/delay-attack-jam-intercept-and-replay.html, 2014.

[49] S. Kamkar, "Drive it like you hacked it: new attacks and tools to wirelessly steal cars," *Presentation at DEFCON*, vol. 23, 2015.

[50] A. Bogdanov, “Attacks on the keeloq block cipher and authentication systems,” in *Proc. of 3rd Citeseer Conference on RFID Security*, 2007.

[51] T. Eisenbarth, T. Kasper, A. Moradi, C. Paar, M. Salmasizadeh, and M. T. M. Shalmani, “On the power of power analysis in the real world: A complete break of the keeloq code hopping scheme,” in *Proc. of Annual International Cryptology Conference (Crypto)*, 2008, pp. 203–220.

[52] F. D. Garcia, D. Oswald, T. Kasper, and P. Pavlidès, “Lock it and still lose it - on the (in)security of automotive remote keyless entry systems,” in *Proc. of the 25th USENIX Security Symposium (USENIX Security)*, 2016.

[53] S. Tillich and M. Wójcik, “Security analysis of an open car immobilizer protocol stack,” in *Proc. of Trusted Systems, 4th International Conference (INTRUST)*, 2012, pp. 83–94.

[54] S. Mangard, E. Oswald, and T. Popp, *Power analysis attacks: Revealing the secrets of smart cards*. Springer Publishing Company, Incorporated, 2010.

[55] M. Joye and M. Tunstall, Eds., *Fault analysis in cryptography*, ser. Information Security and Cryptography. Springer, 2012.

[56] G. P. Hancke, K. Mayes, and K. Markantonakis, “Confidence in smart token proximity: Relay attacks revisited,” *Computers & Security*, vol. 28, no. 7, pp. 615–627, 2009.

[57] S. Drimer and S. J. Murdoch, “Keep your enemies close: Distance bounding against smartcard relay attacks,” in *Proc. of the 16th USENIX Security Symposium (USENIX Security)*, 2007.

[58] Y. Hu, A. Perrig, and D. B. Johnson, “Wormhole attacks in wireless networks,” *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 2, pp. 370–380, 2006.

[59] N. Carlini, P. Mishra, T. Vaidya, Y. Zhang, M. Sherr, C. Shields, D. A. Wagner, and W. Zhou, “Hidden voice commands,” in *Proc. of 25th USENIX Security Symposium (USENIX Security)*, 2016, pp. 513–530.

[60] G. Zhang, C. Yan, X. Ji, T. Zhang, T. Zhang, and W. Xu, “Dolphinattack: Inaudible voice commands,” in *Proc. of the ACM SIGSAC Conference on Computer and Communications Security (CCS)*, 2017, pp. 103–117.

[61] N. Roy, S. Shen, H. Hassanieh, and R. R. Choudhury, “Inaudible voice commands: The long-range attack and defense,” in *Proc. of 15th USENIX Symposium on Networked Systems Design and Implementation (NSDI)*, 2018, pp. 547–560.

[62] N. Carlini and D. A. Wagner, “Audio adversarial examples: Targeted attacks on speech-to-text,” in *Proc. of IEEE Security and Privacy Workshops*, 2018, pp. 1–7.

[63] “An introduction to keeloq® code hopping.”

[64] M. Zhou, Z. Qin, X. Lin, S. Hu, Q. Wang, and K. Ren, “Hidden voice commands: Attacks and defenses on the VCS of autonomous driving cars,” *IEEE Wireless Communications*, vol. PP, pp. 1–1, DOI: 10.1109/MWC.2019.1800477, 2019.

[65] “Lock it or lose it,” https://www.youtube.com/watch?v=Mmi2LRF7al8, 2011.

[66] G. P. Hancke and M. G. Kuhn, “An rfid distance bounding protocol,” in *Proc. of First IEEE International Conference on Security and Privacy for Emerging Areas in Communications Networks (SECURECOMM)*, 2005, pp. 67–73.

[67] S. Drimer, S. J. Murdoch *et al.*, “Keep your enemies close: Distance bounding against smartcard relay attacks,” in *Proc. of the 16th USENIX Security Symposium (USENIX Security)*, vol. 312, 2007.

[68] C. H. Kim and G. Avoine, “Rfid distance bounding protocol with mixed challenges to prevent relay attacks,” in *Proc. of International Conference on Cryptology And Network Security (CANS)*, 2009, pp. 119–133.

[69] A. Moradi and T. Kasper, “A new remote keyless entry system resistant to power analysis attacks,” in *Proc. of 7th IEEE International Conference on Information, Communications and Signal Processing (ICICS)*, 2009, pp. 1–6.

[70] X. Yuan, Y. Chen, Y. Zhao, Y. Long, X. Liu, K. Chen, S. Zhang, H. Huang, X. Wang, and C. A. Gunter, “Commandersong: A systematic approach for practical adversarial voice recognition,” in *Proc. of 27th USENIX Security Symposium (USENIX Security)*, 2018, pp. 49–64.

[71] S. Hu, X. Shang, Z. Qin, M. Li, Q. Wang, and C. Wang, “Adversarial examples for automatic speech recognition: Attacks and countermeasures,” *IEEE Communications Magazine*, vol. PP, pp. 1–1, DOI: 10.1109/MCOM.2019.1900006, 2019.

[72] Q. Wang, X. Lin, M. Zhou, Y. Chen, C. Wang, Q. Li, and X. Luo, “Voicepop: A pop noise based anti-spoofing system for voice authentication on smartphones,” in *Proc. of IEEE Conference on Computer Communications (INFOCOM)*, 2019, pp. 2062–2070.

[73] L. Wu, J. Yang, M. Zhou, Y. Chen, and Q. Wang, “LVID: A multimodal biometrics authentication system on smartphones,” *IEEE Transactions on Information Forensics and Security*, vol. PP, pp. 1–1, DOI: 10.1109/TIFS.2019.2944058, 2019.

[74] X. Lin, X. Sun, P.-H. Ho, and X. Shen, “GSIS: A secure and privacy-preserving protocol for vehicular communications,” *IEEE Transactions on vehicular technology*, vol. 56, no. 6, pp. 3442–3456, 2007.

[75] S. C. HPL, “Introduction to the controller area network (CAN),” *Application Report SLOA101*, pp. 1–17, 2002.

[76] K. Koscher, A. Czeskis, F. Roesner, S. Patel, T. Kohno, S. Checkoway, D. McCoy, B. Kantor, D. Anderson, H. Shacham *et al.*, “Experimental security analysis of a modern automobile,” in *Proc. of 2010 IEEE Symposium on Security and Privacy (S&P)*, 2010, pp. 447–462.

[77] C. Miller and C. Valasek, “Adventures in automotive networks and control units,” *Def Con*, vol. 21, pp. 260–264, 2013.

[78] K.-T. Cho and K. G. Shin, “Error handling of in-vehicle networks makes them vulnerable,” in *Proc. of the 2016 ACM SIGSAC Conference on Computer and Communications Security (CCS)*, 2016, pp. 1044–1055.

[79] S. Checkoway, D. McCoy, B. Kantor, D. Anderson, H. Shacham, S. Savage, K. Koscher, A. Czeskis, F. Roesner, T. Kohno *et al.*, “Comprehensive experimental analyses of automotive attack surfaces,” in *Proc. of 20th USENIX Security Symposium (USENIX Security)*, vol. 4, 2011, pp. 447–462.

[80] C. Miller and C. Valasek, “Remote exploitation of an unaltered passenger vehicle,” *Black Hat USA*, vol. 2015, p. 91, 2015.

[81] S. Nie, L. Liu, and Y. Du, “Free-fall: Hacking Tesla from wireless to CAN bus,” *Briefing, Black Hat USA*, pp. 1–16, 2017.

[82] J. M. Ernst and A. J. Michaels, “LIN bus security analysis,” in *Proc. of 44th Annual Conference of the IEEE Industrial Electronics Society (IECON)*, 2018, pp. 2085–2090.

[83] J. Takahashi, Y. Aragane, T. Miyazawa, H. Fuji, H. Yamashita, K. Hayakawa, S. Ukai, and H. Hayakawa, “Automotive attacks and countermeasures on LIN-bus,” *Journal of Information Processing*, vol. 25, pp. 220–228, 2017.

[84] D. K. Nilsson, U. E. Larson, F. Picasso, and E. Jonsson, “A first simulation of attacks in the automotive network communications protocol flexray,” in *Proc. of the International Workshop on Computational Intelligence in Security for Information Systems (CISIS)*, 2009, pp. 84–91.

[85] M. Wolf, A. Weimerskirch, and C. Paar, “Security in automotive bus systems,” in *Proc. of 2th Int. Conf. on Embedded Security in Cars (ESCAR)*, 2004.

[86] J. H. Kim, S.-H. Seo, N.-T. Hai, B. M. Cheon, Y. S. Lee, and J. W. Jeon, “Gateway framework for in-vehicle networks based on CAN, FlexRay, and Ethernet,” *IEEE Transactions on Vehicular Technology*, vol. 64, no. 10, pp. 4472–4486, 2014.

[87] L. Yu, J. Deng, R. R. Brooks, and S. B. Yun, “Automobile ecu design to avoid data tampering,” in *Proc. of the 10th ACM Annual Cyber and Information Security Research Conference (CISRC)*, 2015, p. 10.

[88] S. Woo, H. J. Jo, and D. H. Lee, “A practical wireless attack on the connected car and security protocol for in-vehicle CAN,” *IEEE Transactions on intelligent transportation systems*, vol. 16, no. 2, pp. 993–1006, 2014.

[89] D. K. Nilsson, U. E. Larson, and E. Jonsson, “Efficient in-vehicle delayed data authentication based on compound message authentication codes,” in *IEEE 68th Vehicular Technology Conference (VTC)*, 2008, pp. 1–5.

[90] C. Szilagyi and P. Koopman, “Low cost multicast authentication via validity voting in time-triggered embedded control networks,” in *Proc. of the 5th ACM Workshop on Embedded Systems Security (WESS)*, 2010, p. 10.

[91] B. Groza, S. Murvay, A. Van Herrewege, and I. Verbauwhede, “LiBrA-CAN: A lightweight broadcast authentication protocol for controller area networks,” in *Proc. of International Conference on Cryptology and Network Security (CANS)*, 2012, pp. 185–200.

[92] Y. Weisglass, “Practical attacks on CAN message authentication,” *Proc. The 4th Int. Conf. on Embedded Security in Cars (escar Asia)*, 2017.

[93] T. Matsumoto, M. Hata, M. Tanabe, K. Yoshioka, and K. Oishi, “A method of preventing unauthorized data transmission in controller area network,” in *Proc. of IEEE 75th Vehicular Technology Conference (VTC)*, 2012, pp. 1–5.

[94] R. Kurachi, Y. Matsubara, H. Takada, N. Adachi, Y. Miyashita, and S. Horihata, “Cacan-centralized authentication system in CAN (controller area network),” in *Proc. of 12th Int. Conf. on Embedded Security in Cars (ESCAR)*, 2014.

[95] Y. Ujiie, T. Kishikawa, T. Haga, H. Matsushima, T. Wakabayashi, M. Tanabe, Y. Kitamura, and J. Anzai, "A method for disabling malicious CAN messages by using a centralized monitoring and interceptor ecu," in *Proc. of 13th Int. Conf. on Embedded Security in Cars (ESCAR)*, 2015.

[96] D. D. Yao, X. Shu, L. Cheng, and S. J. Stolfo, *Anomaly detection as a service: Challenges, advances, and opportunities*, ser. Synthesis Lectures on Information Security, Privacy, and Trust. Morgan & Claypool Publishers, 2017.

[97] L. Cheng, K. Tian, and D. D. Yao, "Orpheus: Enforcing cyber-physical execution semantics to defend against data-oriented attacks," in *Proceedings of the 33rd Annual Computer Security Applications Conference, 2017*, 2017, pp. 315–326.

[98] L. Cheng, K. Tian, D. Yao, L. Sha, and R. A. Beyah, "Checking is believing: event-aware program anomaly detection in cyber-physical systems," *IEEE Transactions on Dependable and Secure Computing*, pp. 1–1, 2019.

[99] U. E. Larson, D. K. Nilsson, and E. Jonsson, "An approach to specification-based attack detection for in-vehicle networks," in *Proc. of IEEE Intelligent Vehicles Symposium (IV)*, 2008, pp. 220–225.

[100] M. Müter, A. Groll, and F. C. Freiling, "A structured approach to anomaly detection for in-vehicle networks," in *Proc. of Sixth International Conference on Information Assurance and Security (IAS)*, 2010, pp. 92–98.

[101] A. Taylor, N. Japkowicz, and S. Leblanc, "Frequency-based anomaly detection for the automotive CAN bus," in *Proc. of IEEE Intelligent Vehicles Symposium World Congress on Industrial Control Systems Security (WCICSS)*, 2015, pp. 45–49.

[102] P.-S. Murvay and B. Groza, "Source identification using signal characteristics in controller area networks," *IEEE Signal Processing Letters*, vol. 21, no. 4, pp. 395–399, 2014.

[103] W. Choi, H. J. Jo, S. Woo, J. Y. Chun, J. Park, and D. H. Lee, "Identifying ECUs using inimitable characteristics of signals in controller area networks," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 6, pp. 4757–4770, 2018.

[104] "Taxonomy and definitions for terms related to on-road motor vehicle automated driving systems," https://www.sae.org/standards/content/j3016_201401/, 2016.

[105] A. Balakrishnan, A. G. Puranic, X. Qin, A. Dokhanchi, J. V. Deshmukh, H. B. Amor, and G. Fainekos, "Specifying and evaluating quality metrics for vision-based perception systems," in *Proc. of Design, Automation & Test in Europe Conference & Exhibition, (DATE)*, 2019, pp. 1433–1438.

[106] "Uber cleared over Arizona pedestrian's self-driving car death," http://fortune.com/2019/03/06/uber-cleared-arizona-self-driving-death/, 2019.

[107] K. Eykholt, I. Evtimov, E. Fernandes, B. Li, A. Rahmati, C. Xiao, A. Prakash, T. Kohno, and D. Song, "Robust physical-world attacks on deep learning models," *arXiv preprint arXiv:1707.08945*, 2017.

[108] A. Chernikova, A. Oprea, C. Nita-Rotaru, and B. Kim, "Are self-driving cars secure? Evasion attacks against deep neural networks for steering angle prediction," *arXiv preprint arXiv:1904.07370*, 2019.

[109] C. Sitawarin, A. N. Bhagoji, A. Mosenia, M. Chiang, and P. Mittal, "Darts: Deceiving autonomous cars with toxic signs," *arXiv preprint arXiv:1802.06430*, 2018.

[110] A. N. Bhagoji, W. He, B. Li, and D. Song, "Practical black-box attacks on deep neural networks using efficient query mechanisms," in *European Conference on Computer Vision*. Springer, 2018, pp. 158–174.

[111] C. Xiao, B. Li, J.-Y. Zhu, W. He, M. Liu, and D. Song, "Generating adversarial examples with adversarial networks," *arXiv preprint arXiv:1801.02610*, 2018.

[112] Q. Wang, Y. Zhang, X. Lu, Z. Wang, Z. Qin, and K. Ren, "Real-time and spatio-temporal crowd-sourced social network data publishing with differential privacy," *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 4, pp. 591–606, 2018.

[113] N. Carlini, C. Liu, J. Kos, lfar Erlingsson, and D. Song, "The secret sharer: Measuring unintended neural network memorization & extracting secrets," *IACR Cryptology ePrint Archive*, 2018.

[114] S.-C. Lin, Y. Zhang, C.-H. Hsu, M. Skach, M. E. Haque, L. Tang, and J. Mars, "The architectural implications of autonomous driving: Constraints and acceleration," in *ACM SIGPLAN Notices*, vol. 53, no. 2. ACM, 2018, pp. 751–766.

[115] A. C. Berg and J. Malik, "Geometric blur for template matching," in *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, vol. 1. IEEE, 2001, pp. I–I.

[116] X. Chen, C. Liu, and D. Song, "Tree-to-tree neural networks for program translation," in *Advances in Neural Information Processing Systems*, 2018, pp. 2547–2557.

[117] "Federated learning: Collaborative machine learning without centralized training data," https://ai.googleblog.com/2017/04/federated-learning-collaborative, 2017.

[118] I. Dinur and K. Nissim, "Revealing information while preserving privacy," in *Proc. of the Twenty-Second ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems (PODS)*, 2003, pp. 202–210.

[119] C. Dwork, "Differential privacy: A survey of results," in *Proc. of Theory and Applications of Models of Computation (TAMC)*, 2008, pp. 1–19.

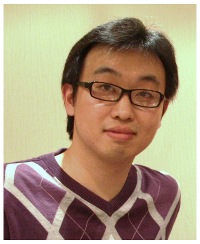
[120] L. Zhao, Q. Wang, Q. Zou, Y. Zhang, and Y. Chen, "Privacy-preserving collaborative deep learning with unreliable participants," *IEEE Transactions on Information Forensics and Security*, vol. PP, pp. 1–1, DOI: 10.1109/TIFS.2019.2 939 713, 2019.

[121] J. S. Jang, S. Kong, M. Kim, D. Kim, and B. B. Kang, "Secret: Secure channel between rich execution environment and trusted execution environment," in *NDSS*, 2015.

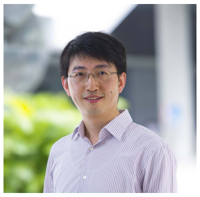
[122] V. Costan and S. Devadas, "Intel sgx explained." *IACR Cryptology ePrint Archive*, vol. 2016, no. 086, pp. 1–118, 2016.

[123] S. Mofrad, F. Zhang, S. Lu, and W. Shi, "A comparison study of intel SGX and AMD memory encryption technology," in *Proceedings of the 7th International Workshop on Hardware and Architectural Support for Security and Privacy*. ACM, 2018, p. 9.

Kui Ren received the Ph.D. degree from the Worcester Polytechnic Institute, Worcester, MA, USA. He is currently a Professor of Computer Science and Technology and the Director of the Institute of Cyberspace Research at Zhejiang University Hangzhou, Zhejiang, China. His current research interest spans cloud and outsourcing security, wireless and wearable system security, and artificial intelligence security. He is a fellow of IEEE, and a distinguished scientist of the ACM. He was a recipient of the IEEE CISTC Technical Recognition Award 2017 and the NSF CAREER Award in 2011.

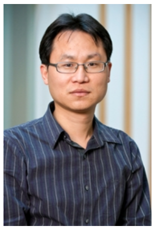


Qian Wang is a Professor with the School of Cyber Science and Engineering, Wuhan University. He received the Ph.D. degree from Illinois Institute of Technology, USA. His research interests include AI security, data storage, search and computation outsourcing security and privacy, wireless systems security, big data security and privacy, and applied cryptography etc. Qian received National Science Fund for Excellent Young Scholars of China in 2018. He is also an expert under National "1000 Young Talents Program" of China. He is a recipient of the 2018 IEEE TCSC Award for Excellence in Scalable Computing (Early Career Researcher), and the 2016 IEEE Asia-Pacific Outstanding Young Researcher Award. He is also a co-recipient of several Best Paper and Best Student Paper Awards from IEEE ICDCS'17, IEEE TrustCom'16, WAIM'14, and IEEE ICNP'11 etc. He serves as Associate Editors for IEEE Transactions on Dependable and Secure Computing (TDSC) and IEEE Transactions on Information Forensics and Security (TIFS). He is a Member of the IEEE and a Member of the ACM.



Cong Wang (SM'17) is currently an Associate Professor in the Department of Computer Science, City University of Hong Kong. He received his Ph.D. degree from Illinois Institute of Technology, USA. His current research interests include data security, network security, privacy-enhancing technologies, blockchain and decentralized applications. He is one of the Founding Members of the Young Academy of Sciences of Hong Kong. He received the Outstanding Research Award in 2019, the Outstanding Supervisor Award in 2017, and the Presidents Award in 2016 from City University of Hong Kong. He is a co-recipient of the Best Student Paper Award of IEEE ICDCS 2017, the Best Paper Award of IEEE ICPADS 2018, MSN 2015 and CHINACOM 2009. His research has been supported by multiple government research fund agencies, including National Natural Science Foundation of China, Hong Kong Research Grants Council, and Hong Kong Innovation and Technology Commission. He serves/has served as associate editor for IEEE Transactions on Dependable and Secure Computing (TDSC), IEEE Internet of Things Journal (IoT-J) and IEEE Networking Letters, and TPC co-chairs for a number of IEEE conferences/workshops. He is a senior member of the IEEE, and member of the ACM.

Qin Zhan is currently a ZJU100 Young Professor, with both the College of Computer Science and Technology and the Institute of Cyberspace Research (ICSR) at Zhejiang University, China. He was an assistant professor at the Department of Electrical and Computer Engineering in the University of Texas at San Antonio after receiving the Ph.D. degree from the Computer Science and Engineering department at State University of New York at Buffalo in 2017. His current research interests include data security and privacy, secure computation outsourcing, artificial intelligence security, and cyber-physical security in the context of the Internet of Things. His works explore and develop novel security sensitive algorithms and protocols for computation and communication on the general context of Cloud and Internet devices.



Xiaodong Lin received his Ph.D. degree in information engineering from Beijing University of Posts and Telecommunications, China, and his Ph.D. degree (with the Outstanding Achievement in Graduate Studies Award) in electrical and computer engineering from the University of Waterloo. He is currently an associate professor in the School of Computer Science at the University of Guelph, Canada. His research interests include computer and network security, privacy protection, applied cryptography, computer forensics, and software security.