Vehicular Communications 23 (2020) 100214

Contents lists available at ScienceDirect

Vehicular Communications

www.elsevier.com/locate/vehcom

Cybersecurity challenges in vehicular communications

Zeinab El-Rewini [a], Karthikeyan Sadatsharan [a], Daisy Flora Selvaraj [a], Siby Jose Plathottam [b], Prakash Ranganathan [a,*]

[a] School of Electrical Engineering and Computer Science (SEECS), University of North Dakota, Grand Forks, ND, USA
[b] Energy Systems Division, Argonne National Laboratory, IL, USA

ARTICLE INFO

Article history:
Received 9 August 2019
Received in revised form 18 October 2019
Accepted 21 November 2019
Available online 3 December 2019

Keywords:
Cyber-physical systems
Security
Vehicle safety
Cyber-attacks
Vehicle-to-vehicle

ABSTRACT

As modern vehicles are capable to connect to an external infrastructure and Vehicle-to-Everything (V2X) communication technologies mature, the necessity to secure communications becomes apparent. There is a very real risk that today's vehicles are subjected to cyber-attacks that target vehicular communications. This paper proposes a three-layer framework (sensing, communication and control) through which automotive security threats can be better understood. The sensing layer is made up of vehicle dynamics and environmental sensors, which are vulnerable to eavesdropping, jamming, and spoofing attacks. The communication layer is comprised of both in-vehicle and V2X communications and is susceptible to eavesdropping, spoofing, man-in-the-middle, and sybil attacks. At the top of the hierarchy is the control layer, which enables autonomous vehicular functionality, including the automation of a vehicle's speed, braking, and steering. Attacks targeting the sensing and communication layers can propagate upward and affect the functionality and can compromise the security of the control layer. This paper provides the state-of-the-art review on attacks and threats relevant to the communication layer and presents countermeasures.

1. Introduction

Modern-day vehicles can no longer be perceived as just mechanical systems, with over 100 million lines of code in the overall architecture, higher than a modern operating system or a Boeing 757, as shown in Fig. 1 [1]. Vehicles are growing increasingly connected and computer-like, with capabilities to sync with mobile phones, provide vehicle occupants with the latest weather and navigation updates, and communicate safety information to other vehicles and surrounding infrastructure. Though vehicle connectedness and computerization bring evident advantages to the passenger experience and to road safety, they have also created more opportunities for hackers to hijack vehicles and place both passenger and pedestrian lives at risk.

Many well-publicized car hacks were successful because the hackers were able to exploit vehicular communications. In 2015, security researchers Charlie Miller and Chris Valasek took control of a Chrysler Jeep Cherokee traveling at high speeds on the interstate and forced it to come to a stop in the middle of traffic [3]. They were able to remotely gain control of the vehicle by exploiting vulnerabilities within Chrysler's UConnect system, which provides Chrysler vehicles with entertainment, navigation, and Wi-Fi capabilities. As a result of the hack, Chrysler recalled over 1.4 million automobiles. Chrysler Jeep Cherokees were not the only vehicles found to be susceptible to malicious interference. In 2016,

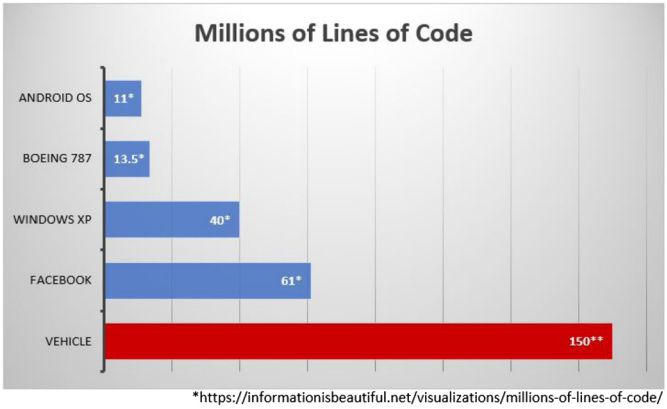

Fig. 1. Overall lines of code in various systems [2].

* Corresponding author.
E-mail addresses: zeinabrewini@gmail.com (Z. El-Rewini), karthikeyans068@gmail.com (K. Sadatsharan), daisy.selvaraj@und.edu (D.F. Selvaraj), sibyjackgrove@gmail.com (S.J. Plathottam), prakash.ranganathan@und.edu (P. Ranganathan).

https://doi.org/10.1016/j.vehcom.2019.100214

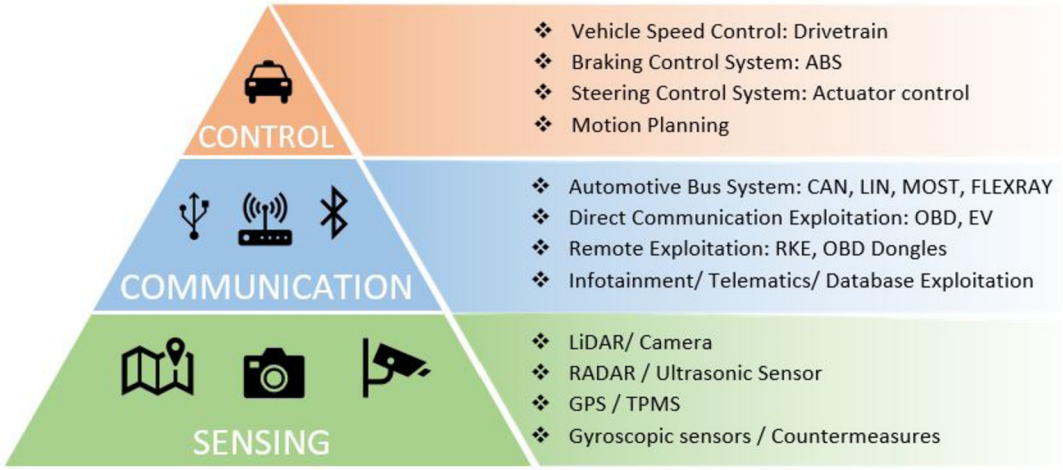

Fig. 2. The Autonomous Vehicular Sensing-Communication-Control (AutoVSCC) framework.

Volkswagen (VW) vehicles were discovered to have flaws with their Remote Keyless Entry (RKE) systems that would enable attackers to unlock VW vehicles [4]. That same year, the members of the Keen Security Lab, a division of the Chinese firm Tencent, demonstrated their ability to remotely hijack the brakes of a Tesla Model S vehicle [5]. Two years later, the same lab released a report identifying vulnerabilities within BMW vehicles that would allow attackers to access the Controller Area Network (CAN) bus and send illegitimate CAN messages [6].

Vulnerabilities within vehicular communications lead to four vehicular cybersecurity challenges, which are described in [7] and [8] by Onishi:

- Limited connectivity: Though the external connectivity of vehicles is increasing, most vehicles do not yet have the capability to update their software through Over-the-Air (OTA) updates, which would enable vehicles to always be protected against the latest cyber-attacks. Even as OTA updates become more standard, vehicles will also be at risk of malfunctions due to incomplete updates.
- Limited computational performance: Vehicular computational performance is generally limited, as compared to the computational performance of a computer. This limitation exists because vehicles have a longer lifetime and must endure higher temperatures and vibrations than the average PC or laptop. As a result of their computational disadvantage, vehicles are more likely to be hacked than computers. The limited computational performance of vehicles will also mean that some vehicular cybersecurity solutions will have too high an overhead to be implemented.
- Unpredictable attack scenarios and threats: A vehicular architecture can be infiltrated through many different entry-points, including vehicular databases, remote communication technologies, and vehicular parts. New attacks are continually being developed, which means that automakers will find it difficult to predict where hackers will strike next. An unsecured product manufactured by Original Equipment Manufacturers (OEMs) can provide hackers with additional entry-points into a vehicle.
- Critical risk for drivers or passengers lives: Even if just a few sensors are misinformed or only a small number of illegitimate messages are sent, a vehicle could experience malfunctions that place the lives of drivers, passengers, and pedestrians at risk.

Threats targeting vehicular communications can be understood through the three layer Autonomous Vehicular Sensing Communication Control (AutoVSCC) framework illustrated in Fig. 2. At the bottom of the hierarchy is the sensing layer, which is vulnerable to spoofing and eavesdropping attacks on vehicle sensors, such as the inertial or radar sensors. Above the sensing layer is the communication layer, which encompasses both inter-vehicular and intra-vehicular communications and is vulnerable to eavesdropping attacks and the manipulation of messages between vehicles and roadside infrastructure. The communication layer is also susceptible to threats that propagate upward from the sensing layer, which is made of vehicular sensors. Threats to both the sensing and communication layers can affect the top most tier, the control layer, which describes automated vehicular control techniques, such as vehicle speed and steering control.

This paper aims to bridge the knowledge gap in understanding various vulnerabilities and cyber threats in communication networks of autonomous and connected vehicles by providing a timely and systematic review. More specifically, the unique contributions of this work include:

- First, an overview of the three layer framework (sensing, communication and control) of autonomous and connected vehicles discussing different layers that are vulnerable to cybersecurity threats is presented.
- Second, the paper presents a comprehensive review of the threats to vehicular communication using knowledge derived from literature. Intravehicular security is discussed with a focus on automotive bus systems, infotainment and telematics systems and vehicular ports. In addition, V2X security related to remote communication technologies, clustering, databases, vehicle-to-vehicle communication and vehicle-to-infrastructure communication is also discussed.
- Finally, potential countermeasures for various threats in V2X communications are detailed. The paper provides potential security solutions (blockchain, machine learning) to fill the research gaps needed to secure vehicular communication.

The remainder of the paper is organized as follows: Section 2 describes the communication layer; Section 3 describes intravehicular security threats; Section 3 include three subsections: automotive bus systems (Section 3.1), infotainment and telematics systems (Section 3.2), and vehicular ports (Section 3.3); Section 4 examines the security of V2X communication under multiple headings: the security of remote communication technologies (Section 4.1), databases (Section 4.2), and clustering (Section 4.3),

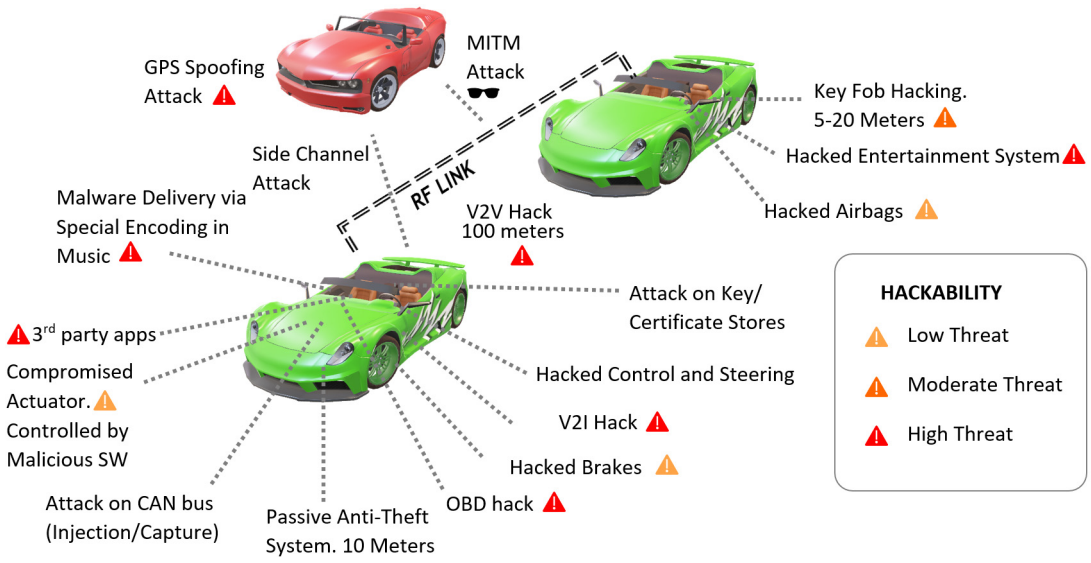

Fig. 3. Cyber vulnerabilities in a vehicular ecosystem. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

specific to vehicle-to-vehicle communication (Section 4.4), and Section 4.5. discusses threats specific to vehicle-to-infrastructure communication. Section 5 discusses the research gaps in vehicular communications and potential future solutions that can be applied to vehicular communications. Finally, the paper concludes with Section 6.

2. The communication layer

The communication layer is made up of vehicular communications that can occur both internally and externally to a vehicle. Internal vehicular communications can take place within the in-vehicle Network, which is also known as the automotive network or intra-vehicle network. The in-vehicle network is based upon the intercommunication of the many Electronic Control Units (ECUs) within a vehicle's electronic subsystems [9], [10].

The external vehicular communication occurs when vehicles connect directly to USBs and maintenance tools, connect remotely to Remote Keyless Entry systems, or engage in V2X communication, which enables the transmission of messages between vehicles and infrastructure. To facilitate V2X communication, connected and autonomous vehicles can operate as nodes within vehicular ad hoc networks (VANETs), which are self-organized [11]. VANETs are primarily composed of two types of wireless nodes: On-Board Units (OBUs) and Road-Side Units (RSUs). OBUs are wireless transmitters installed within V2X-capable vehicles. OBU-equipped vehicles can communicate with one another and with Road-Side Units (RSUs), which are stationary devices located along roads and infrastructure that can provide internet connectivity for OBUs and report on the state of traffic.

OBU and RSU nodes can transmit and receive messages over the wireless network. They are able to enter into communication with surrounding nodes and exit from these communications when the nodes are no longer in range [11], [12]. To protect against malicious transmissions, Trust Authorities (TAs) perform authenticity checks and remove malicious nodes within VANETs [12]. In this way, real-time information about vehicles and infrastructure can be transmitted to increase road safety and efficiency and ultimately work to support fully automated and driverless vehicles.

V2X communication commonly refers to: Vehicle-to-Vehicle (V2V) and Vehicle-to- Infrastructure (V2I). V2V involves the exchanging of information such as speed, heading, and brake status among vehicles with OBUs. Through V2V, vehicles receive transmissions regarding the movements of surrounding vehicles. Drivers can then see beyond their immediate surroundings and perceive threats that may not even be in their line of sight [13]. V2I communication occurs between vehicles and RSUs [12]. Through V2I, RSUs transmit warnings relating to red light and stop sign violations or even upcoming changes in speed limits [13]. In [14], Axelrod bypasses this traditional categorization in favor of a more precise classification by defining interactions between vehicles, their surroundings, infrastructure, transportation networks and the larger ecosystem. In addition to V2V and V2I, Axelrod's system includes additional categories such as Vehicle-to-Surroundings (V2S), which allows local sources to communicate with vehicles; Vehicle-to-Ecosystem (V2E), which allows distant sources such as GPS to communicate with vehicles; and Vehicle-to-Transportation Networks(V2TN), which enables communications between taxi services and vehicles. There is also another category known as V2P (Vehicle-to-Pedestrian): the National Highway Traffic Safety Administration (NHTSA) has suggested that pedestrians use mobile apps that transmit safety signals, which can then be received by vehicles and lessen the chance of pedestrians being hit [13].

Fig. 3 illustrates the many ways that attackers can maliciously interfere with vehicular communication. In order to combat these threats, all forms of communications within the communication layer must meet underlying common security requirements, viz: confidentiality, integrity and availability. Other vital properties include non- repudiation, privacy, real-time constraints and flexibility [15].

- Confidentiality: According to National Institute of Standards and Technology (NIST) the confidentiality is defined as "preserving authorized restrictions on information access and disclosure, including means for protecting personal privacy and proprietary information [19]."
- Integrity: A message delivered to the receiver node should not be altered by any intruder. Integrity intertwines with authentication in VANETs to guarantee secure transmission. The receiver node should be able to verify whether the received data is corrupted or legitimate. Hence, data verification is another related property that is enclosed under integrity for VANETs.
- Availability: According to NIST the definition of availability is "ensuring timely and reliable access to and use of information [19]."

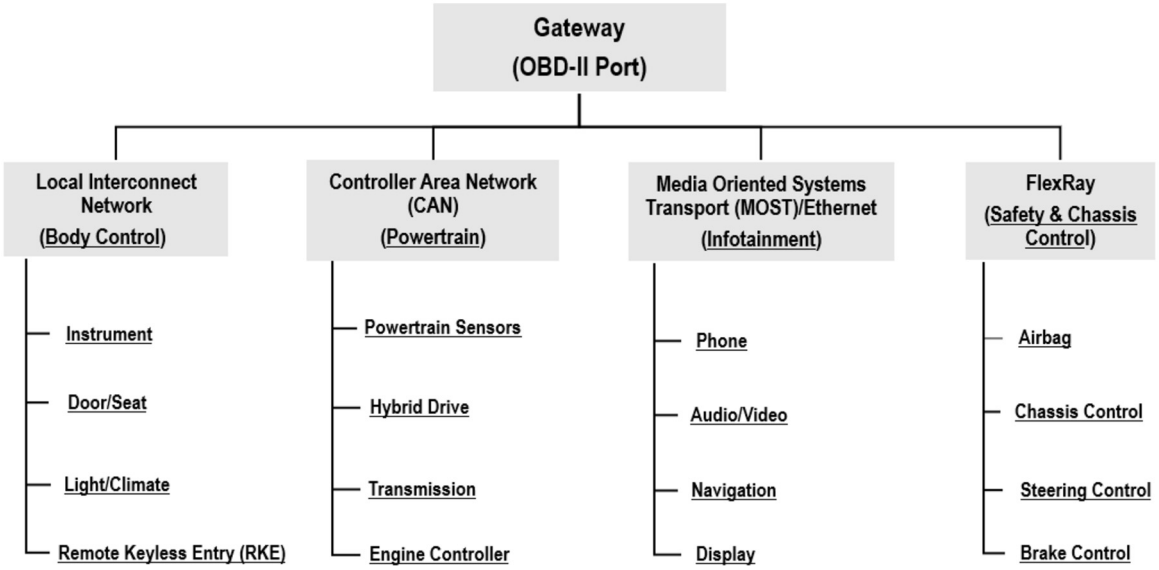

Fig. 4. A common in-vehicle network architecture, based on [16], [17], [18].

- Non-Repudiation: The right to authenticate should not be denied by any broadcasting node. This property is especially important in the case of an accident. After the accident, the driver must be rightly identified during investigation, and before the accident, every message should be transmitted reliably.
- Privacy: Unauthorized users should not have access to a vehicle or a driver's personal information. The concept of anonymity also coexists with privacy in the case of VANETs [20].
- Real-Time constraints: Outdated information is of no use in the high mobility environment of a VANET. Outdated weather or traffic information is not useful and therefore must be prevented from delayed transmission.
- Flexibility: The need for a flexible means of communication within a security architecture is significant in a dynamic environment, irrespective of the fact that VANETs exist predominantly for a short duration. The deployment of a security architecture for a vehicular ad-hoc network is significantly challenging due to the highly dynamic nature of the various attacks that are performed and the high number of alien electronic systems that are in proximity.

3. In-vehicle security threats

In-vehicle communication occurs within automotive bus systems, which enable message transmission between vehicle ECUs; infotainment and telematics systems, which provide entertainment capabilities and vehicle system information to passengers; and vehicular ports, which allow vehicles to connect to diagnostic devices, import media, and charge. This section discusses the security of these in-vehicle communication environments. Automotive bus systems are discussed in Section 3.1, whereas threats related to infotainment and telematics systems and vehicular ports are discussed in Section 3.2 and 3.3 respectively.

3.1. Automotive bus system exploitation

Within the in-vehicle network, automotive bus systems enable ECUs in one electronic subsystem to communicate with one another and with other subsystems. Fig. 4 demonstrates a very common architecture for the in-vehicle network, in which different subsystems are connected to one another and to external networks through the use of a gateway. Kim et al. [21] refer to this type of architecture as a central-gateway architecture. They note that backbone-based architectures, in which a domain control unit (DCU) takes on the role of a gateway between various communication protocols, are beginning to replace central-gateway architectures.

Whether central gateway-based architectures or backbone-based architectures are used, communication protocols are employed to transmit messages within each subsystem and between different subsystems. These protocols include the Controller Area Network (CAN), CAN with Flexible Data Rate (CAN-FD), Local Interconnect Network (LIN), FlexRay, Media Oriented Systems Transport (MOST), and Ethernet. Different protocols experience their own unique security concerns. For instance, CAN messages are broadcast to all nodes without discretion which means they are vulnerable to eavesdropping attacks. On the other hand, LIN and MOST transmissions are vulnerable to attacks targeting synchronization. In this section, the attacks targeting CAN and CAN-FD, LIN, FlexRay, MOST, and Ethernet are discussed along with proposed countermeasures.The literature on automotive bus system exploitation is presented in Table 1.

3.1.1. CAN and CAN-FD

CAN transmissions are broadcast to all nodes in the network. Each node must decide whether or not a transmission is relevant and then discard or act upon packets as necessary [10], [22]. Ueda et al. [24] identify CAN attacks as following two use cases: replacing an authorized ECU program with an illegitimate, malicious program, and connecting to the CAN bus using an unauthorized device. The CAN frame is generally unable to support Message Authentication Code (MAC)[46] and other methods of securing communication. However, incorporating these security methods may be possible for CAN with Flexible Data Rate (CAN-FD) which provides additional flexibility and higher bandwidth to the traditional CAN. The CAN payloads only allow for up to 8 bytes of data, while CAN-FD allows for up to 64 bytes of data. In addition, CAN-FD allows the changing of the baud rate, which can enable speeds up to 8 MBit/s, as compared to CAN's standard 1 Mbit/s, by setting the Bit Rate Switch (BIT) bit [47].

3.1.1.1. Threats Malicious actors can perform masquerading, eavesdropping, injection, replay, Denial of Service (DoS), and bus-off attacks to tamper with CAN communication.

Table 1
Automotive bus system exploitation.

Protocol	Attack & countermeasures	References
Controller Area Network (CAN)	Masquerading attack	[22], [23]
	Eavesdropping attack	[22]
	Injection attack	[22]
	Replay attack	[22]
	Denial of service attack	[10], [22]
	Bus-off attack	[23]
	Countermeasures	[22], [24], [23], [25], [26], [27], [28], [29], [30], [31], [32]
Local Interconnect Network (LIN)	Message spoofing attacks	[16]
	Response collision attacks	[33]
	Header collision attacks	[33]
	Countermeasures	[33]
FlexRay	Static segment attacks	[34]
	Eavesdropping attacks	[35]
	Countermeasures	[35], [34], [36], [37]
Media Oriented Systems Transport (MOST)	Synchronization Disruption Attack	[16]
	Jamming attack	[1], [16]
	Countermeasures	[1]
Ethernet	Network access attacks	[38], [39], [40]
	Traffic confidentiality attacks	[41], [39]
	Traffic integrity attacks	[42], [39], [43], [44], [45]
	Denial of service attack	[39], [43]
	Countermeasures	[38], [39]

- **Masquerading attack:** In a masquerading attack, an attacker masquerades as a legitimate node. Liu et al. [22] and Choi et al. [23] identify two CAN vulnerabilities that facilitate masquerading attacks. First, CAN frames are not encrypted and thus can be studied by attackers to locate system entry points. Second, CAN does not support message authentication, i.e., recipients of messages are not given information about the validity of the source, meaning that illegitimate frames can be sent without being detected [22], [23].
- **Eavesdropping attack:** Eavesdropping attacks occur when unauthorized individuals are able to gain access to vehicular messages. CAN's broadcast transmissions allow attackers who gain access to the in-vehicle network to then eavesdrop on CAN transmissions and identify patterns in legitimate CAN frames [22].
- **Injection attack:** In an injection attack, attackers inject fake messages into an automotive bus system. Attackers can gain entry to the in-vehicle network through OBD-II ports, compromised ECUs, or infotainment & telematics systems [22]. Since traditional CAN does not authenticate sending or receiving nodes, illegitimate frames will not be recognized.
- **Replay attack:** In a replay attack, attackers continually resend valid frames to impede the vehicle's real-time functioning [22].
- **Bus-off attack:** Bus-off attacks occur when attackers continually send bits both in the identifier field and in other fields, which causes the ECU's transmit error counter (TEC) to then be incremented. When the TEC has a value greater than 255, the corresponding ECU has to shut down [23].
- **Denial of Service (DoS) attack:** Denial of Service (DoS) attacks occur when attackers continually send high priority messages that block legitimate low priority messages [22]. In a standard CAN packet, the identifier segment determines the message priority. To give their messages high priority status, attackers can simply assign the identifier segment a low value. DoS attacks could be used as an avenue to conduct control override attacks, which allow attackers to take control of the vehicle [10].

3.1.1.2. Countermeasures Masquerading, eavesdropping, injection and replay attacks can occur when messages between ECUs are not encrypted and authenticated. To guard against these types of attacks, MAC can be utilized. However, MAC often does not fit into standard CAN data fields, which allow for up to 8 bytes. Therefore some authors have attempted to either create new protocols or spread MAC across multiple transmissions [23]. Nowdehi et al. [25] examine many of these altered protocols in light of five criteria for potential CAN message authentication solutions from an industry perspective. The criteria are cost effectiveness, backward compatibility, vehicle repair and maintenance, sufficient implementation details and acceptable overhead. The authors found that no solutions met all the criteria, but the most promising seemed to be WooAuth, which is not backward compatible and VatiCAN, which is backward compatible but might not be performant. VatiCAN takes the approach of utilizing maintenance support, "sufficient implementation details" and no excessive overhead, while WooAuth alters the extended CAN protocol to allow more room for authentication codes. The authors ultimately suggest that the CAN bus might "be fundamentally unsuited for secure communication" [25]. Mundhenk et al. [26] propose a lightweight authentication framework that consists of two phases: ECU authentication and stream authorization. During ECU authentication, asymmetric cryptography is used as ECUs authenticate "against a central security module." During the stream authorization phase, symmetric cryptography is used as ECUs request keys in order to initiate message streams. Kang et al. [27] present the Source Authentication Protocol (SAP), an authentication protocol using a one-way hash chain and a sender-based group key that guards against masquerading and replay attacks. Similarly, in [28], Tashiro et al. propose a protocol that provides protection from replay, masquerading and injection attacks by sending a partial MAC in each frame, so that tampering detection can be conducted for both individual frames and entire sections.

An Intrusion Detection System (IDS), which can incorporate machine learning in order to train itself to recognize abnormal behavior, can be an alternative or a supplement to MAC. An IDS can guard against masquerading, injection, bus-off and denial of service attacks. Choi et. al [29] suggest incorporating an external monitoring unit that is capable of identifying analog characteristics of the electrical CAN signal for authentication and intrusion detection purposes. The monitoring unit is first trained to recognize characteristics of signals from valid ECUs and malicious ECUs. Though the external monitoring unit would mean that vehicle owners would not have to alter their vehicle's hardware, the solution that Choi et al. present utilizes the extended CAN protocol, meaning that the firmware would have to be updated. In [23], Choi et al. go on to introduce an IDS known as VoltageIDS, which takes advantage of ECU signal inconsistency to, first, undergo training and testing phases in which signal characteristics would be identified and, second, use the training data in order to verify whether or not ECUs have been compromised. VoltageIDS is able to detect masquerading attacks by using a multi-class classifier, where one class corresponds to one ECU. It predicts the most likely sender

and compares that information to the actual CAN ID of the message. If they differ, then a masquerading attack has been detected. Choi et al. state that VoltageIDS is the first to successfully defend against bus-off attacks. In [31], Lee et al. present the Offset Ratio and Time Interval based Intrusion Detection System (OTIDS), which is able to detect denial of service, masquerading and injection attacks by periodically requesting a data frame from nodes within the CAN bus and monitoring the response time and offset ratio to determine whether an attack is taking place. Liu et al. [22] advocate for an anomaly-based IDS, which detects general anomalous behavior, rather than a misuse-based IDS, which detects patterns of known attacks. Since the effectiveness of misuse-based IDS depend on knowledge of existing attacks, gaps in security might form as the patterns of newer attacks are not known. Tomlinson et al. [30] suggest an IDS that identifies attacks by noting timing changes in CAN traffic. For instance, their system interprets the increased broadcast frequency of certain CAN IDs to mean that an injection attack might be taking place. Ueda et al. [24] propose a security monitoring system that authenticates ECUs through the use of MAC and identifies and overwrites spoofed messages.

Both MAC and intrusion detection systems are discussed in [32] by Groza and Murvay, who examine potential security solutions for CAN's vulnerabilities in light of application and physical layers, as well as cryptography and physical controller characteristics.

3.1.2. LIN

While most research done on in-vehicle networks centers around CAN, Local Interconnect Network (LIN) is another in-vehicle network subject to exploitation by malicious actors. In cases where CAN's higher bandwidth, and increased adaptability are not needed, LIN is an alternative free of CAN's higher overhead [33]. For this reason, LIN is often used to facilitate ECU intercommunication that does not require high transmission speeds and is often used to control lights, engines, air conditioning, steering wheels, seats, and doors [33], [16]. In [33], Takahashi et al. note that there is a significant gap in the research being done on threats to LIN communication, and that, though LIN is often used within CAN networks and is thus vulnerable to unauthorized entry via the CAN bus, LIN's master-slave architecture means that CAN-specific vulnerabilities will not always be applicable to LIN. In LIN networks, one master node communicates with multiple slave nodes [33]. LIN communication depends upon a time schedule that the master node uses to determine when to transmit a message frame, which consists of a header, sent by the master, and a response, sent by a slave. Master nodes transmit headers instructing slave ECUs to either subscribe or publish to the bus and slave ECUs are polled by the master ECU while the master awaits a response [16]. Data rates within LIN communication generally do not exceed 20 kbps [33].

3.1.2.1. Threats Threats to LIN communication include Message Spoofing, Response Collision, and Header Collision attacks.

- **Message spoofing attacks:** In message spoofing attacks, attackers send illegitimate messages with inaccurate information in order to disrupt vehicular communications. Two vulnerabilities within LIN master-slave communications can lead to Message Spoofing attacks [16]. First, a master within a LIN network can transmit a message that will cause one of the slaves to sleep. Second, the master can also set the SYNC field within a LIN message to synchronize slaves. Attackers can take advantage of the master's capabilities in order to spoof messages ordering slaves to sleep, thereby shutting off the LIN network. They could also spoof messages and alter the SYNC field to tamper with synchronization.
- **Response collision attacks:** During a response collision attack, an attacker sends an illegitimate message at the same time that a legitimate message is being sent. Within the LIN protocol, response collision attacks exploit LIN's error handling mechanism, which takes effect when a slave node sending a response notices that the value in the bus differs and stops transmission. Attackers can exploit this mechanism by either sending a false header or waiting until the master node sends a header [33]. They can then send a false response that will collide with a legitimate response sent by a slave node. This will alter the value in the bus, which means that the legitimate slave node will halt transmission. If the attacker can manage to calculate the correct checksum used in responses, then other nodes will consider the false message to come from legitimate sources.
- **Header collision attacks:** Takahashi et al. [33] describe the header collision attack, which occurs when an attacker sends a false header to collide with a legitimate header sent by the master node. The legitimate header specifies that a certain slave node must publish a response, but the attacker's collision means that the publisher node has changed. When the new publisher node sends a response, attackers can conduct a response collision attack to inject their own false message. In this way, attackers can tamper with the sequence of responses sent within the LIN bus and can keep automated vehicle sliding doors open and lock steering wheels while vehicles are traveling along the road.

3.1.2.2. Countermeasures Established countermeasures focus on response and head-er collision attacks. In [33], Takahashi et al. suggest that a slave node send out an abnormal signal, which would overwrite a false message sent by an attacker, if it detects that the bus value does not match its response. Additional suggestions include incorporating MAC and assigning important data to the first byte of a transmission, as the first byte is more difficult to corrupt.

3.1.3. FlexRay

FlexRay, like LIN, is neglected in terms of automotive network security [35]. The FlexRay network is often thought of as a potential successor to CAN, with a communication rate of 10 Mbps as compared to CAN's 1 Mbps rate [48], [34]. FlexRay is a more expensive and complex communication protocol than CAN or LIN, so it is often used in safety-critical applications where message transmissions must follow a precise timing schedule [47]. FlexRay communication takes place in the context of communication cycles, which can contain both static and dynamic segments. Static segments use Time-division Multiple Access (TDMA) for messages that need to be delivered in real-time, while dynamic segments use an event-driven communication protocol for maintenance-related messages [34].

3.1.3.1. Threats FlexRay communication is vulnerable to Eavesdropping and Static Segment attacks.

- **Eavesdropping attacks:** Eavesdropping on the FlexRay protocol occurs when attackers can gain access to and understand FlexRay messages. Mousa et al. [35] argue that FlexRay faces the same security concerns as CAN and that those concerns are the potential leakage of security primitives, network privacy, and data confidentiality.
- **Static segment attacks:** Gu et al. [34] offer the opinion that FlexRay security efforts should focus on the static segment, which could have the most danger if compromised. A static segment attack is a general term used to denote an attack that targets the static segment of the FlexRay communication

cycle. Static segment attacks can include Masquerading, Injection, and Replay attacks.

3.1.3.2. Countermeasures Both eavesdropping attacks and static segment attacks can be combated by incorporating authentication. Gu et al. [34] propose the authentication of messages within the static segment through hardware co-processors. They also suggest an optimization method that would mean not every ECU would need a hardware co-processor, thereby reducing performance costs. Han et al. [36] propose the Security-Aware FlexRay Scheduling Engine (SAFE), which is a FlexRay scheduling framework that incorporates the Timed Efficient Stream Loss-tolerant Authentication (TESLA) authentication protocol [49]. Mousa et al. [35] suggest incorporating [26]'s Lightweight CAN Authentication Protocol (LCAP) within FlexRay, since in-vehicle networks often rely upon interconnected CAN and FlexRay networks and also LCAP is lightweight and backward-compatible. Püllen et al. [37] provide several suggestions on FlexRay authentication and cryptographic key transmission, including utilizing FlexRay's dual-channel mode to divide message authentication codes.

3.1.4. MOST

Media Oriented Systems Transport (MOST) bus systems are primarily used for entertainment purposes and allow for the transmission of "audio, video, voice, and control data via fiber optic cables" [1]. A MOST network is synchronized by a master node, which sends out timing frames that synchronize nodes within the network [16]. MOST transmissions contain FlexRay-like static and dynamic sections, a clear recipient and sender, and a control channel through which data channels can be claimed [1]. MOST communication is vulnerable to attacks that disrupt synchronization and to jamming or denial of service attacks [16].

3.1.4.1. Threats MOST communication is vulnerable to synchronization disruption attacks and jamming attacks.

- **Synchronization disruption attacks:** Synchronization disruption attacks tamper with MOST synchronization, which is controlled by the timing frames sent by MOST masters [16]. A malicious node can send out false timing frames to disrupt this synchronization.
- **Jamming attacks:** Jamming attacks prevent legitimate messages from being sent through the MOST protocol. MOST devices base the availability of the fixed-length dynamic segment on message priority [16]. Therefore, a malicious node could conduct a Jamming attack by continually sending false messages that block legitimate lower-priority messages. Jamming attacks could also be conducted by continually requesting a data channel through the control channel within a MOST transmission [1].

3.1.4.2. Countermeasures Limited work has been carried out on the security of the MOST bus. However, Wolf et al.'s [1] suggestions for the security of general automotive bus systems can be applied to MOST. These suggestions include authenticating senders, encrypting transmissions, and including gateway firewalls. Future research should examine protections against synchronization disruption attacks and jamming attacks.

3.1.5. Ethernet

Automotive networks based on Ethernet provide high bandwidths and timing guarantees. For this reason, Ethernet is, like FlexRay, also considered a next-generation in-vehicle network [38]. Ethernet networks consist of hosts, which are connected by switches. A switch is a multiport bridge. Hosts transmit and receive Ethernet frames, which can be sent via unicast, mulitcast, and broadcast modes of communication [39]. Two Ethernet-based protocols, Time-Triggered Ethernet (TTEthernet) and Time-Sensitive Networking (TSN) seem to be the most applicable to automotive settings. TSN requires clock synchronization, allows frames to differ in priority level, and is not limited to supporting only audio and video data types. In additional to traditional network traffic, TTEthernet supports two additional types of network traffic: Time-Triggered (TT), which requires clock synchronization and allows traffic transmission to occur at specific times, and Rate-Constrained (RC), which has lower efficiency than TT traffic but allows a predetermined bandwidth.

3.1.5.1. Threats Threats to Ethernet communication include Network Access attacks, Traffic Confidentiality attacks, Traffic Integrity attacks, and Denial of Service attacks [39].

- **Network access attacks:** Network access attacks enable attackers to gain access to the Ethernet network. These attacks can be standalone, or they can facilitate other categories of attacks; for instance, taking control of other hosts or switches. Attackers can physically join the Ethernet network by connecting to an unconnected port on a switch [40] or even remotely access the network through the use of social engineering [39].
- **Traffic confidentiality attacks:** Once attackers have gained access to the network, they can conduct traffic confidentiality attacks, which allow them to eavesdrop on network traffic. Attackers can transmit messages and analyze the replies to determine the network topology and structure [39]. They can attach listening devices either on a cable connecting a host and switch, or between two switches to listen in on network traffic. Switches that are unsure where to forward a frame will flood the frame out to all ports. Attackers can take advantage of this feature to perform MAC flooding attacks [41]. During these attacks, they are able to eavesdrop on all frames by overwriting a MAC table so that all the data frames will be flooded.
- **Traffic integrity attacks:** Traffic integrity attacks alter network traffic [39]. The Address Resolution Protocol (ARP) and the Dynamic Host Configuration Protocol (DHCP) are two protocols used within Ethernet communication. During ARP and DHCP poisoning attacks, attackers can send ARP replies to capture network traffic and respond to DHCP server requests to control network traffic [42]. These attacks can be precursors to man-in-the-middle attacks, which redirect network traffic to an attacker's node so that information can be manipulated [43]. Other types of traffic integrity attacks include Session hijacking attacks and Replay attacks. During a Session hijacking attack, attackers can snoop to discover session information created by protocols layered over Ethernet, and then act as one endpoint of the session or tamper with the session [44], [45]. During a replay attack, eavesdropped messages are continually resent [39].
- **Denial of Service (DoS) attacks:** Denial of service (DoS) attacks disable access to the ethernet service by damaging physical equipment or by overwhelming the system. Kiravuo et al. [39] describe two types of DoS attacks. Layer 1 attacks physically damage links or circuitry, causing the ethernet service to be completely inoperable. Layer 2 attacks can be either resource exhaustion attacks, which exhaust resources by continually sending frames to be processed, or protocol based denial of service attacks, which exploit the self-configurable Spanning Tree Protocol (STP) by continually sending STP messages [43].

3.1.5.2. Countermeasures In [38], Lin and Yu examine the balance of safety and security within Ethernet-based communication and propose several countermeasures for network access, traffic confidentiality, and traffic integrity attacks. Lin and Yu propose au-

thentication methods and a frame replication approach that would guard against traffic confidentiality and traffic integrity attacks. They also discuss virtual local area network segmentation, which could prevent network access attacks. To combat DoS attacks, Meyer et al. [50] propose metering Ethernet frames and Pesé et al. [51] discuss firewall DoS attack detection. Kiravuo et al. [39] provide an extensive examination of different security measures for plain ethernet under the categories of router-based security, access control, secure protocols and security monitoring.

3.2. Infotainment & telematics exploitation

Infotainment systems provide information and entertainment to vehicle occupants [48]. The information that is provided by infotainment systems can include "voice calls, text messages, emails, social networking, personal contacts," and other forms of data that can be received by connecting to a mobile phone [52]. Infotainment systems' entertainment capabilities allow people to pair their mobile devices, stream music, and watch videos. Some advanced infotainment systems enable mirroring which allows a mobile device's screen to be shared with a vehicle's screen [48].

Telematics systems complement infotainment systems by providing information on internal vehicular systems, which includes "fuel efficiency, engine failures, brake pad wear, transmission issues, oil life, climate control, biometric sensors, vehicle speed, acceleration, direction, braking, cornering, ignition, steering, seat belts, door locking, tire pressure and recently visited destinations including routes traveled" [52]. This information could be used for vehicle maintenance purposes, emergency situations where vehicle roadside assistance services require information about the state and location of the vehicle or even programs that enable insurance companies to track driving data and reward drivers who are safe and do not drive excessive miles. Jaisingh et al. [52] note that the distinction between telematics and infotainment systems is becoming less clear, as some applications utilize both non-vehicular and vehicular information to, for instance, draw upon vehicle location information to send text messages alerting recipients to the time remaining until the vehicle's arrival. Both infotainment and telematics systems depend upon a Telematics Control Unit (TCU), and telematics systems generally rely on "long-range mobile networks or Global Navigation Satellite System (GNSS)" [52]. Infotainment system vulnerabilities were demonstrated when the BMW ConnectedDrive infotainment system was hacked, in part because its corresponding in-vehicle Network gateway, the Combox, lacked thorough security mechanisms [53], [54].

3.2.1. Threats

Infotainment and telematics systems are vulnerable to control override attacks and injection attacks.

- **Control override attack:** During a control override attack, a malicious actor's commands override the vehicle operator's attempts to take corrective action. Jo et al. [55] identifies security risks in Android OS-based telematics systems that enabled drivers to remotely unlock and lock car doors, start and stop the car engine using low-speed CAN, and access diagnostic information using high-speed CAN. The authors present an attack scenario in which attackers download OTA firmware, add functionality to enable remote door opening and GPS tracing, and then distribute the altered firmware. Victims then install the modified firmware, which exposes them to remote attacks.
- **Injection attack:** Injection attacks occur when attackers inject illegitimate and malicious messages within the In-Vehicle network. In [56], Mazloom et al. show that the MirrorLink protocol, which is used to link smartphones to vehicular infotainment systems, has security vulnerabilities that could lead to an attacker gaining access to the in-vehicle Network and injecting malicious messages. Mazloom et al. were able to generate a heap overflow within MirrorLink and take control of a process within the infotainment system. They suggest that attackers can use similar methods to gain access to the in-vehicle network and send illegitimate messages.

Table 2
Direct communication exploitation.

Vehicular unit	Attack & countermeasures	References
Infotainment and telematics	Control override attack	[48], [52], [53], [54]
	Injection attack	[56]
	Countermeasures	[56], [57], [58], [55], [59]
OBD-II ports	In-vehicle network access attack	[10], [61]
	Dongle exploitation	[9], [62]
	Countermeasures	[22], [63], [62]
USB ports	Threats	[65]
	Countermeasures	[64], [65]
Electric vehicle charging	Threats	[68], [67], [73], [69], [70], [71], [72]
	Countermeasures	[72], [76]

3.2.2. Countermeasures

Countermeasures to telematics system vulnerabilities that could enable control override attacks are discussed by Jo et al. [55]. First, they suggest that source code and byte code obfuscation be implemented. Second, they suggest that an entity other than the device manufacturer should issue a private key during code signing. Third, they advocate for remote attestation, which involves an external entity ensuring that no exploitation has occurred.To guard against injection attacks, Mazloom et al. [56] provide recommendations for infotainment-to-smartphone protocols such as MirrorLink. They advise changing the privileges of certain protocol processes and watching for memory safety when using low-level languages. General recommendations for the security of infotainment and telematics systems security can be found in the IEEE Center for Secure Design's (CSD) [57] guide for those who are involved in the development of infotainment and telematics systems. In addition, Lee and Lee [58] propose an implementation of a session key establishment protocol for infotainment systems using Elliptic Curve Cryptography. Mandal et al. [59] analyze the security of Android apps that are able to connect to infotainment systems.

3.3. Vehicular ports

Within modern-day vehicles are a variety of ports, which, when connected to external devices, enable drivers to gain access to maintenance information and on-vehicle entertainment, synchronize their mobile phones, and charge their electric vehicles. However, if attackers were to gain access to these vehicle ports, then they could gain entry to the in-vehicle network, perform eavesdropping attacks, and even install malware and viruses. In this section, we discuss the vulnerabilities of three vehicular ports: the On-Board Diagnostics II port, the USB port and the charging port found within electric vehicle charging infrastructure. Proposed solutions and protections against these vulnerabilities are also discussed. Table 2 provides the literature on direct communication exploitation.

3.3.1. OBD-II ports

On-Board Diagnostics (OBD) systems monitor "various emission control and engine components/subsystems" and can light the

Malfunction Indicator Lamp (MIL) [60]. The first-generation of OBD regulations primarily detected electrical failures and did not have standardized Diagnostic Trouble Codes (DTC), communication formats, or connector locations. The second-generation OBD, known as OBD-II, was federally mandated for vehicles in the United States in 1996. In addition to monitoring electrical failures, the second-generation OBD also monitored emission-related systems and provided standardization across different manufacturers. OBD systems have OBD-II ports, which can generally be found underneath a vehicle's steering column [10].

3.3.1.1. Threats OBD-II ports are vulnerable to in-vehicle network access attacks and dongle exploitation attacks.

- **In-vehicle network access attack:** During an in-vehicle network access attack, attackers insert an external device into the OBD-II port and gain access to the in-vehicle network. OBD-II ports are points of vulnerability for vehicular security, since connecting to the OBD-II port enables the gathering of diagnostic information, access to the in-vehicle network, and the installation of malware [10]. Valasek and Miller [61] were able to transmit and receive messages over CAN using an ECOM cable and homemade connectors to connect to the OBD-II port.
- **Dongle exploitation attack:** Dongles plugged into the OBD-II port can be controlled remotely and are not difficult to decrypt [9]. One such dongle was the Bosch Drive-log connector dongle, which tracked vehicle maintenance and was able to guide the vehicle operator to appropriate locations for servicing. This dongle, which connects to a vehicle's OBD-II port, was hacked when the Argus Cybersecurity firm conducted a brute-force attack that enabled them to connect to the dongle via bluetooth and send malicious transmissions over the Controller Area Network. These transmissions ultimately led to the engine failure of a traveling vehicle [62].

3.3.1.2. Countermeasures In-vehicle network Access attacks could be thwarted by monitoring the frame injection coming from the OBD-II port, as Liu et al. [22] suggest. Klinedinst and King [63] recommend disallowing message transmissions from the OBD-II port to the in-vehicle network and cryptographically signing and encrypting firmware updates. To prevent dongle exploitation attacks, Kovelman [62] recommends that vehicle manufacturers incorporate attack detection capabilities in addition to cryptographic solutions so that automakers can implement remote security updates when an attack is detected.

3.3.2. USB port

USB ports have become prevalent in modern-day vehicles, since they can connect phones, navigation systems, and USB devices to the vehicle [64].

3.3.2.1. Threats USB ports within vehicles pose additional security risks. In 2014, Security Research Labs showed that it was possible to reprogram USB controller chips in order to install malware, spoof network cards, and boot small viruses targeting the operating system [65]. Cai et al. [66] found that attackers could use the USB port to create a backdoor within the BMW Next Best Thing (NBT) vehicle entertainment system.

3.3.2.2. Countermeasures Onishi [64] argues that it will be difficult to protect against USB vulnerabilities due to the high number of USB devices and the time that is required to develop standards for USB security. Similarly, in [65], it is stated that there are currently no truly effective protections against USB cyber-attacks. Onishi proposes two countermeasures: first, having a USB device initially connect to a website in order to receive a security certificate that will then allow it to connect to the vehicle; and second, not allowing malware or viruses from a USB device to access safety-critical areas.

3.3.3. Electric vehicle charging infrastructure

For electric vehicles, (EVs), another entry point to the in-vehicle network is the charging infrastructure. Bernardini et al. [48] note that EVs will need to be charged more often than gasoline cars will need to be fueled. While charging, EVs will be vulnerable to attacks through their charging infrastructure. Eventually, the charging infrastructure could be used to conduct attacks on a smart grid.

3.3.3.1. Threats Many attacks on electric vehicle charging within a smart grid environment have been identified. Mustafa et al. [67] find that EV charging is susceptible to masquerading, tampering, eavesdropping, and denial of service attacks, in addition to privacy concerns and charging thievery. Fries and Falk [68] discuss EV charging susceptibility to eavesdropping, man-in-the-middle and tampering attacks on the payment price and the amount of energy that the meter believes the EV has received. They also discuss the potential for malicious software within the vehicle to affect a charging station, or a compromised charging station to affect an EV.

Sun et al. [69] discuss threats to EV location privacy. An EV's location can be tracked when it is in close proximity to a charging station, and its charging station entry and exit information can be used to identify the vehicle.

Alcaraz et al. [70] identify security threats within the Open Charge Point Protocol (OCPP), which is used in communications between charging stations and a smart grid's central energy management system. In [71], Vaidya and Mouftah discuss OCPP security in light of their OCPP-based secure charging system, the Secure Electric Vehicle Charging Ecosystem in Smart Grid, or SecCharge. Lee et al. [72] argue that both OCPP and other protocols used within EV charging, such as ISO/IEC 15188, are not secure. Attackers can take advantage of vulnerabilities within the ISO/IEC 15188 protocol to assume another vehicle's identity by manipulating an identification number stored in the EV's internal storage. They can also manipulate message properties to illegally charge more than the EV requires. Other identified threats to ISO/IEC 15188 include manipulating meter statuses, payment types, and tariff table type messages. Attackers can then reduce or eliminate the charging price or shut off a charging station's service.

3.3.3.2. Countermeasures Chan and Zhou [73] present a security-conscious architecture for EV charging that thwarts masquerading, eavesdropping, and tampering attacks within a smart grid. Roberts et al. [74] propose an authentication scheme that could guard against man-in-the-middle attacks during EV to EV charging. Morrison [75] suggests implementing OCPP 2.0, which provides additional security features, and incorporating cryptographic signatures and firmware updates in EV charging stations.

To combat attacks on location privacy, Huang et al.'s [76] Lightning Network and Smart Contracts (LNSC) model could be utilized. The LNSC is a blockchain-based approach to securing the EV charging process. The model consists of three phases: registration, scheduling, authentication and changing. During the registration phase, EVs, charging piles, and operators register within a blockchain system. During the scheduling phase, shortest-path, time cost, comprehensive cost, or wait time-based scheduling algorithms are implemented to determine the optimal charging pile for the EV. During the authentication and changing phases, both the EV and the charging pile undergo an authentication process using information stored on the blockchain and a commitment generated by the EV is recorded in the blockchain. Sun et al. [69] presents another means of protecting privacy. They propose associating each

charging station with a power hub, which will be associated with multiple power routers. Vehicles can then charge by connecting to the power hub. Since charging vehicles are no longer directly associated with a single charging station, the electric vehicle's location privacy is protected.

In [77], Morosan and Pop suggest the use of a neural network to detect malicious behavior in OCPP traffic. For greater OCPP and ISO 15118 security, Buschlinger et al. [78] recommend the incorporation of a Hardware Security Module (HSM) within both standards. In [72], Lee et al. propose incorporating an authentication process, error-checking, and verifying information within transmissions in order guard against attacks on the ISO/IEC 15188 protocol.

4. V2X security threats

In Vehicle-to-Everything (V2X), Vehicle-to-Vehicle (V2V) communication between OBUs and Vehicle-to-Infrastructure communication (V2I) between OBUs and Road-Side Units (RSUs) is facilitated by remote communication technologies, such as Dedicated Short-Range Communications (DSRC) and Bluetooth. Vehicles can transmit their sensor data to in-vehicle and cloud databases, and they can be clustered into smaller dynamic groups within the vehicular ad hoc network. This section discusses the security vulnerabilities of these different modes of V2X communication. In Section 4.1, remote communication technologies are discussed. Sections 4.2 and 4.3 discuss V2V and V2I, respectively. Section 4.4 examines vehicle to database communication, and Section 4.5 discusses vehicular clustering methods.

4.1. Remote communication technologies

Remote communication technologies are used within V2X communication to pass messages between OBUs and RSUs and to provide localization and positioning for automated vehicles. Attackers can rely on weaknesses within these technologies to remotely tamper with a vehicle's functioning. This section discusses the following remote communication technologies: Remote Keyless Entry (RKE) systems, Dedicated Short Range Communications (DSRC)/Wireless Access in Vehicular Environments (WAVE), cellular networks, Zigbee, Bluetooth, Wi-Fi and WiMAX, Ultra WideBand (UWB) and Radio Frequency Identification (RFID). Both the threats facing remote communication technologies and suggested protective measures are covered in this section. The literature on these remote communication threats is presented in Table 3.

4.1.1. Remote Keyless Entry system

Remote Keyless Entry (RKE) systems are the successors to the traditional method of opening car doors by inserting physical keys. Keys with RKE-capabilities allow key-holders to remotely lock and unlock car doors, start or stop engines, or turn on and off anti-theft alarms. Accepted button clicks within RKE systems trigger a counter, which is then used to create a rolling code signal that helps to prevent against replay attacks [79].

4.1.1.1. Threats Some manufacturers, such as the VW Group, have used only a small number of cryptographic RKE keys for all their vehicles [80]. Rolling code schemes used by manufacturers such as the VW Group are subject to eavesdropping attacks and thus make vehicles vulnerable to RKE cloning. Both Wetzels [81] and Glocker [82] identify Brute force, replay, man-in-the-middle, and jamming attacks as threats to RKE. Wetzels notes that three main vulnerabilities of RKE systems are: 1) frequent use of outdated devices and techniques, 2) weak cryptographic schemes, and 3) implementation faults. Liu et al. [83] examine potential attacks on the Hitag2 cipher, which is used in many RKE systems.

Table 3
Remote communication exploitation.

Communication technology	Attack & countermeasures	References
Remote Keyless Entry Systems (RKES)	Threats	[80], [81], [82], [79], [83]
	Countermeasures	[79], [80], [82], [84], [85], [86], [83]
Wireless access technologies (Threats)	DSRC/WAVE	[87], [88], [89], [90], [91], [92], [93], [93], [94], [95], [96]
	Cellular	[97], [98], [99], [100], [101], [102], [103]
	ZigBee	[104], [105], [106], [107]
	Bluetooth	[64], [108]
	Wi-Fi/WiMAX	[109], [110], [111], [112], [113]
	UWB	[114]
	RFID	[115], [116]
Wireless access technologies (Countermeasures)	DSRC/WAVE	[92], [117], [118], [119]
	Cellular	[98], [120], [103]
	ZigBee	[105], [107][121], [122], [123], [104], [124]
	Bluetooth	[125]
	Wifi/WiMAX	[109], [111], [112], [113]
	UWB	[114], [116]
	RFID	[115], [126], [127], [128], [129], [130], [131]

4.1.1.2. Countermeasures Lee et al.'s [86] Rhythm Key based approach to encrypting RKE communications could be a method of prevention against eavesdropping attacks. In [84], Hamadaqa et al. present a Secret Unknown Cipher scheme that would allow RKE controllers to be clone-resistant. Van de Beek and Leferink [79] design a more robust receiver, which they argue is less susceptible to jamming attacks.

Glocker et al. [82] propose a security-conscious RKE protocol and an accompanying encryption algorithm that provides a defense against Brute force, man-in-the-middle, and replay attacks. Countermeasures to these attacks are also listed in [81]. In [80], Garcia and Oswald advocate for both "secure cryptographic algorithms and secure key distribution." Zhang et al. [85] present Effective k-Means Authentication 2 (EKA2), a scheme that validates the revocation status of digital certificates using a clustering approach in order to authenticate RKE devices that wish to communicate with the vehicle. Liu et al. [83] suggest an eventual transition to more secure algorithms, such as the Advanced Encryption Standard (AES).

4.1.2. DSRC/WAVE

Dedicated Short-Range Communications (DSRC) are a suite of standards that enable the communication of safety messages with "low latency, fast network connectivity, highly secure and high-speed communication" within V2V and V2I communication [92]. The Federal Communications Commission set a 75 MHz bandwidth at 5.850-5.925 GHz band as the spectrum for Intelligent Transport Systems. This spectrum is known as the DSRC spectrum [93]. Communications within the DSRC spectrum follow the Wireless Access in Vehicular Environments (WAVE) standard [93]. WAVE communications are built on top of the IEEE 802.11p standard, which ensures that the maximum delay for high-priority messages will not exceed tens of milliseconds [94]. WAVE is defined in

the IEEE 1609.x family of standards, which specify multi-channel device communication architecture and services, resource manager services and interfaces, security services, networking services, multi-channel operations, and the services and formats required for secure electronic payment [95]. WAVE systems include OBUs and RSUs [96]. Li [96] provides an overview of DSRC and WAVE.

4.1.2.1. Threats Laurendeau and Barbeau [87] rank security threats within the WAVE architecture as either critical, major, or minor based on a ranking methodology by the European Telecommunications Standards Institute (ETSI). Critical and major threats included malware, black hole attacks, GPS spoofing, location tracking, and denial of service attacks.

Biswas et al. [90] examine the potential for a synchroniza-tion-based Distributed Denial of Service attack on IEEE 802.-11p. An attacker can gain access to the transmission schedule used to send service announcements and broadcast messages simultaneously, causing a message collision that would go unnoticed by vehicle occupants. Whyte et al. [91] identify the threats faced by WAVE service advertisement (WSA). They find that WSA is vulnerable to attacks on availability and privacy.

4.1.2.2. Countermeasures Laurendeau and Barbeau [87] provide countermeasures for the five threats classified as critical and major. In [117], Lyamin et al. propose an algorithm that detects DoS attacks within IEEE 802.11p in the context of platooning. Since IEEE 802.11p communications can suffer collisions even in normal operation, Nguyen-Minh et al. [118] suggest a means of detecting jamming attacks and differentiating them from regular collisions. Ucar et al. [88] propose the use of a hybrid IEEE 802.11p and Visible Light Communication (VLC) platoon to address IEEE 802.11p's vulnerabilities. However, the hybrid communication method is still susceptible to packet falsification and replay attacks. In [119], Ghambir and Sharma propose a hybrid-authentication technique for RSUs and vehicles with OBUs communicating over IEEE 802.11p. Biswas et al. [90] provide defenses against synchronized distributed denial of service attacks on IEE 802.11p, and Whyte et al. [91] analyze the potential countermeasures for attacks on WSA availability and privacy.

Though DSRC is often the most discussed means of wireless communication in a V2V or V2I context, Dey et al. [92] argue that, as Intelligent Transportation Systems evolve to accommodate high-speed, secure communication between moving vehicles and roadside infrastructure, DSRC will not be a sufficient implementation. Instead, they advocate for a heterogeneous network (Het-Net), that utilizes Wi-Fi and LTE in addition to DSRC.

4.1.3. Cellular

DSRC suffers from a limited bandwidth, which will not support future V2V communication [97]. For this reason, cellular networks are thought to be good candidates for DSRC communication. The Long Term Evolution (LTE) 4G cellular network, and its low-cost variant, Long Term Evolution Vehicular (LTE-V), provide another form of wireless communication in V2X contexts [98], [94]. LTE is a packet-based system that is characterized by a low access latency and high data rates. It supports integration with other wireless communication technologies [99].

Cellular Vehicle-To-Everything (C-V2X) is another mode of wireless communication that is based upon LTE [132]. C-V2X can operate in two different modes: a direct communication mode, which does not operate within cellular networks, and a network communication mode, which uses cellular networks to facilitate communication. The Third Generation Partnership Project (3GPP), which has standardized LTE for a V2X context, has also been working on incorporating V2X communications into 5G-New Radio (5g-NR) technology [97].

4.1.3.1. Threats Cichonski et al. [98] provide an extensive report on LTE security threats, which include jamming attacks, eavesdropping attacks, and attacks conducted from unlicensed base stations or from compromised femtocells. In [101], Jover discusses security concerns related to the fact that many LTE signaling messages are transmitted without cryptographic protections. Rupprecht et al. [100] examine LTE security in light of user data encryption and network authentication. Cao et al. [99] present security functionalities and the architectural security vulnerabilities within LTE and another variant, LTE-Advanced (LTE-A). Muhammad and Safdar [97] examine security in terms of both LTE and 5G-based vehicular networks. In [102], Marojevic provides an analysis of threats faced by C-V2X communication.

4.1.3.2. Countermeasures Cichonski et al. [98] mention potential mitigations to the attacks they noted in LTE communication. Kaur et al. [120] propose a lightweight key management scheme that would help authenticate nodes within LTE communications. Liyange et al. [103] suggest additions to existing LTE security features using Software Defined Networking and Network Function Virtualization. Rupprecht et al. [100], Cao et al. [99], Muhammad and Safdar [97], and Marojevic, [102] all discuss countermeasures to cellular communication vulnerabilities.

4.1.4. ZigBee

Zigbee is a short-range communication standard that is intended for low data rate communications, [104]. Nandhakumar et al. [121] propose the use of Zigbee within V2V and V2I communications. Lei and Wu [122] suggest its use within Forward Collision Warning Systems, and Pawade et al. [123] suggest the use of Zigbee for Advanced Driver Assistance Systems (ADAS). ZigBee networks, which are intended for low data rate communications, are comprised of a coordinator, router, and end devices and are based upon the IEEE 802.15.4 standard [104], [124]. The coordinator is generally the initial node in the network, has a unique ID, and oversees other nodes' connections to the network. ZigBee routers forward network messages, and act as coordinators that cannot start a new network. End devices are generally in a sleeping state and awaken when they need to respond to a message or at certain periods of time [124].

4.1.4.1. Threats Automotive Zigbee security analyses are mostly non-existent. Zigbee security is mostly discussed in the context of the general Internet-of-Things (IOT). Olawumi et al. [105] describe and carry out three attacks on ZigBee networks. These attacks exploit the following vulnerabilities: first, that attackers can discover configuration details of devices within an in-range ZigBee network; second, that some Zigbee networks are unencrypted; and third, that attackers can re-transmit ZigBee network traffic to conduct Replay attacks. In [107], Fan et al. analyze security vulnerabilities within Zigbee IoT devices. Both Zillner [106] and Sun and Qian [104] provide an assessment of ZigBee security features.

4.1.4.2. Countermeasures Olawumi et al. [105] suggest the use of intrusion detection, the installation of network keys before deployment, and the incorporation of time-stamping within ZigBee's encryption methods to combat the three attacks they examined. Fan et al. [107] recommend not using pre-loaded and factory generated keys. Instead, they suggest providing each ZigBee device with a network security key and foregoing static device IDs in favor of a rotating system of device IDs.

4.1.5. Bluetooth

Within the realm of general short-range communications, Bluetooth is the most widespread protocol [108]. In a connected vehicle context, Bluetooth is often used to pair smartphones with

the vehicle's infotainment and telematics system to provide calling functionality, music streaming, calendars, and car diagnostics [64]. However, Bluetooth is overly complex, suffers from fragmentation, and has not been heavily scrutinized on a security basis [108].

4.1.5.1. Threats Attackers can take advantage of vulnerabilities within Bluetooth to conduct automotive attacks. Onishi [64] provides a listing of Bluetooth vulnerabilities. First, Bluetooth's security mechanisms are limited at best, and users may not even have those mechanisms enabled. Second, if a carry-in device, such as a smartphone, has been compromised, then Bluetooth's few security mechanisms will be rendered ineffective. Third, once an attacker gains access to a device via Bluetooth, the victim's privacy will have been compromised. Fourth, attackers can take advantage of vulnerabilities within the Bluetooth stack that enable buffer overflow attacks; for instance, one vulnerability allowed buffers of fixed size to be overwritten by buffers of variable size [108], [64]. In 2017, [108] identified a new attack vector known as BlueBorne, which spreads wirelessly and attacks devices by exploiting their Bluetooth vulnerabilities. Once a device has been compromised, attackers can take control of the device, access the device's data, spread malware, and infiltrate networks. Blueborne could have serious effects on automotive security.

4.1.5.2. Countermeasures Research that focuses specifically on automotive Bluetooth vulnerabilities and their countermeasures is lacking. However, in [125], a methodology for testing automotive Bluetooth interfaces and an automotive Bluetooth attack classification are provided.

4.1.6. Wi-Fi and WiMAX

Wi-Fi is also a candidate for V2V and V2I communication, or at least a potential complement to DSRC/WAVE. Similar to but faster than Wi-Fi is the Worldwide Interoperability for Microwave Access (WiMAX), which refers to IEEE 802.16, a standard with low-latency, Quality of Service (QoS), security features and all-IP core network support [109], [133].

4.1.6.1. Threats Little has been published on automotive Wi-Fi and WiMAX security threats. However, in [110], Nie et al. were able to remotely hack a Tesla Model S vehicle, in part by exploiting the fact that the password to an embedded Wi-Fi Service Set Identifier (SSID) was saved in plain-text. They were then able to fake a hotspot and redirect traffic to their own domain. Vo-Huu et al. [111] examined the IEEE 802.11 standard, which Wi-Fi is based upon, and found that its interleaver/ convolutional scheme leaves it vulnerable to jamming attacks. In [134], Nakhila and Zou examine how Evil Twin attacks on Wi-Fi can be detected. During an Evil Twin attack, individuals unsuspectingly connect to an illegitimate Wi-Fi access point, which can then eavesdrop on their activity. Vanhoef et al. [112] discuss the potential of Denial of Service attacks on Wi-Fi Protected Access. In [113], Scarfone et al. outline threats to WiMAX security, which include Radio Frequency Jamming, Radio Frequency Interference, DoS, Replay, Man-in-the-Middle, and Eavesdropping attacks. Kolias et al. [109] provide an extensive listing of attacks targeting the WiMAX architecture.

4.1.6.2. Countermeasures Sensitive information, such as SSID passwords, should be encrypted. In [111], Vo-Huu et al. suggest incorporating random, encrypted interleaving to counter against Wi-Fi jamming attacks. Nakhila et al. [134] present a technique of detecting Evil-Twin attacks that rely on a different gateway than the true Wi-Fi access point. Vanhoef et al. [112] discuss countermeasures for Denial of Service attacks. Scarfone et al. [113] describe technical countermeasures for WiMAX's threats and vulnerabilities within the categories of Confidentiality and Integrity Protection, Authentication and Authorization, Client Device Security, and Patches, Upgrades, and Updates. Kolias et al. [109] also offer some solutions to WiMAX's vulnerabilities, which include focusing security efforts on the first entry to the network, private key management, multicast/broadcast communications, and a special type of traffic rerouting known as mesh mode.

4.1.7. UWB

Ultra WideBand (UWB) is another means of wireless communication that is known for its ability to transmit high data rates and for its "low transmission power" [135]. UWB has been proposed for use within VANETs' collision avoidance and vehicle positioning systems [136]. UWB has also been proposed for the off-road localization of autonomous vehicles [137], outdoor localization [135], electric vehicle localization [138], and protection against relay attacks within Passive Entry Passive Start (PEPS) systems [139].

4.1.7.1. Threats The literature on UWB cyber-vulnerabiliies is lacking. However, Hennessy [114] discusses the susceptibility of low-power UWB systems to eavesdropping attacks, due to the fact that low-power systems are not able to support the overhead that accompanies many cryptographic methods.

4.1.7.2. Countermeasures Hennessy [114] presents a method for protecting UWB trans-ceivers from eavesdropping attacks. Instead of relying on cryptographic methods, UWB signals are protected through the use of time hopping. In [140], Zhang et al. propose a mobility-assisted localization algorithm that secures UWB wireless sensor networks against attacks.

4.1.8. RFID

Radio Frequency Identification (RFID) allows identification to occur using radio signals [126]. Some RFID applications within VANETs include passes for public transportation and traffic systems [126]. RFID systems are composed of tags, readers, and backend servers [127]. Tags contain unique identifying information about the tagged object and readers are able to communicate with tags to send and receive identification information. Readers can send tag data to backend servers, where the information can be processed and stored.

4.1.8.1. Threats Cho et al. [115] identify the two main threats to RFID communication as privacy infringement and forgery attacks. These categories include eavesdropping, brute force, replay, and man-in-the-middle attacks. Malicious individuals can also physically attack RFID tags to alter them. Zhang et al. [116] note that RFID systems are vulnerable to de-Synchronization attacks, which occur when an attacker's interception of messages will not allow RFID tag time stamps to be updated. Though one method of guarding against these concerns would be the incorporation of cryptographic protections within the communication between RFID tags and RFID readers, Cho et al. [115] note that the accompanying overhead is not practical for RFID systems, which generally rely on RFID tags with low overhead.

4.1.8.2. Countermeasures Both Khedr [127] and Cho et al. [115] present hash-based authentication methods that could thwart privacy infringement and forgery attacks. Qian et al. [129], Gope and Hwang [128], and Moradi et al. [126] all propose lightweight authentication protocols for RFID systems. Moradi et al. focus on RFID systems within VANETs, while Qian et al. [129] and Gope and Hwang [128] take a more general approach. Both Qian et al. [129] and Liao and Hsiao [130] present authentication schemes based on elliptic curve cryptography, and both Zhang et al. [116] and Vijaykumar and Elango [131] present a mutual authentication protocol for RFID. Zhang et al.'s [116] work defends against De-Synchronization attacks.

OSI Stack	Vehicular Protocols	Function	Threats	Countermeasures
Application	MOST	Supply Services to Applications	Synchronization disruption attacks	Vehicular gateway firewalls
Presentation	MOST	Ensure data produced is in a usable format	Viruses, worms, malware	Intrusion detection system
Session	MOST	Maintain and manage sessions between ECUs	Session hijacking	Authentication
Transport	MOST	Segment and reassemble data for delivery	Injection attacks, DoS	Intrusion detection system
Network	MOST	Establish routing and addressing	Jamming attacks	Intrusion detection system
Datalink	MOST, CAN, LIN, FlexRay	Maintain data flow control and packet delivery	Replay attacks, Bus-off attacks	Message Authentication Code (MAC), Source Authentication Protocol (SAP)
Physical	MOST, CAN, LIN, FlexRay	Convert data packets into electrical signals	Onboard malware, physical tampering	Restricted access to hardware

Fig. 5. Threats and countermeasures in different OSI layers for V2X communications.

4.2. V2V

Within the remote communications section, various remote communication technologies facilitating V2X communication were discussed with respect to their inherent vulnerabilities. Using these technologies, messages can be passed between from one vehicle's OBU to another vehicle's OBU. However, malicious actors can manipulate these messages to cause chaos on roadways. In this section, attacks on message passing, along with potential defenses, are discussed. Fig. 5 shows various threats and countermeasure in different OSI layers for V2X communications.

4.2.1. Threats

The privacy and integrity of V2V communication can be compromised by illusion, bogus information, sybil, timing, impersonation, and alteration/replay attacks.

- **Illusion attack:** During an illusion attack, attackers create false traffic events by altering vehicle sensor readings to trigger the sending of false traffic information messages [141]. Since these messages are sent from a legitimate source, other nodes on the network may receive this data and make erroneous decisions. The illusion attack is one of the tougher attacks to detect because forms of authentication, such as node registration or signature verification, will not work, as the data is sent from an authorized user.
- **Bogus information attack:** During a bogus information attack, attackers generate bogus traffic information and make other vehicles choose different paths, freeing up the road for themselves [142]. The bogus information attack can be performed on various wireless networks at the same time, thus routing the whole path from source to destination for the attacker. An example of a bogus information attack is shown in Fig. 6. The attacker's vehicle sends bogus information to Vehicle A and Vehicle B. The vehicles change their lanes or even their routes assuming that there is heavy traffic ahead of them, thereby freeing up the road for the attacker.
- **Sybil attack:** In a sybil attack, a single intruder node can declare itself as multiple nodes, eventually leading to extensive damage to network topologies and consuming large amounts of bandwidth [143]. The sybil attack is one of the most hurtful and dangerous attacks possible for vehicular ad-hoc networks. Since many vehicular networks are implemented with no certificate authorities or digital signatures, the feasibility of a sybil attack is quite high. A sybil attack is illustrated in Fig. 7.
- **Timing attack:** In timing attacks, a malicious vehicle receives a message, adds some time delay, and then forwards the message to other vehicles, thus leading to improper timing information [144]. This attack can be devastating to vehicular networks, which depend upon real-time applications. An example of a Timing attack is illustrated in Fig. 8. The attacker had the obligation to communicate Vehicle A's positional information when Vehicle B changed the lane. But the attacker adds a time delay to the information and delivers the information only when the Vehicle B changes its position to B', leading to an accident.
- **Impersonation attack:** Impersonation attacks are carried out by providing a vehicle with a false identity [145]. Impersonation is detrimental to the legitimacy of the overall vehicular network architecture and is specifically hurtful in the case of

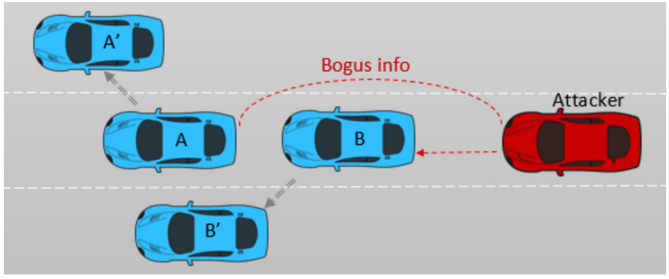

Fig. 6. Illustration of a Bogus information attack.

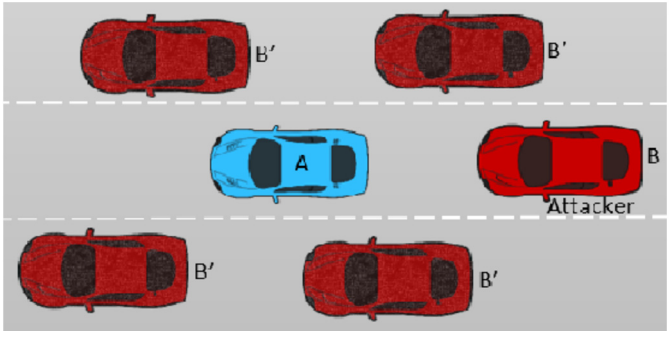

Fig. 7. Illustration of a sybil attack.

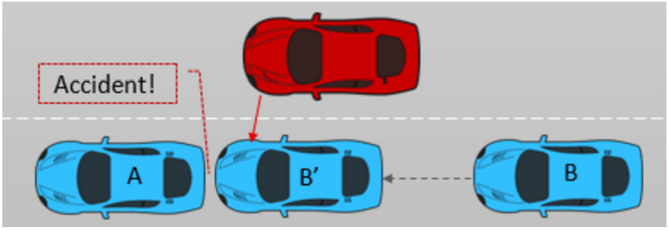

Fig. 8. Illustration of a timing attack.

an accident, since the vehicle under investigation becomes untrackable.

- **Alteration/Replay attack:** As the name suggests, an alteration/replay attack occurs when an attacker employs any previously generated frames to send and communicate with other nodes, with or without alteration [146].

4.2.2. Countermeasures

Lo and Tsai [141] propose a Plausibility Validation Network (PVN) model as a countermeasure against illusion attacks. The PVN is a system that uses a Plausibility Network (PN) containing a set of rules pertaining to each type of data. These rules can be used to determine whether a message is plausible. This database needs to be constantly updated, and needs to be secure. In some situations, the real time constraints will not be solved by this PVN, as the received messages require time to be validated. Hashing techniques are generally used to prevent bogus information attacks or to provide more authentication to the data. One such hashing technique is the Elliptic Curve Digital Signature Algorithm (ECDSA). [147], which is a variation of the Digital Signature Algorithm (DSA) and through which all the nodes agree upon the elliptic curve metrics. Since any change in the message will also create a change in the hash, the bogus information attack can be easily detected.

A few methods of preventing sybil attacks are registration, position verification and radio resource testing. However, there are many disadvantages to these three conventional methods. Registration might introduce the possibility of stealing registrant information by other malicious nodes. Radio resource testing banks on the idea of overwhelming a single malicious node with high complexity computations to test if the traffic information is legitimate. However, this method also fails, as the malicious node might have a very high computational power thereby leading to false sense of authentication for the user [143], [148]. One method used is the concept of a radio having just a single channel to transmit and receive simultaneously to detect sybil attacks [148]. Another method of detecting a sybil attack is by measuring the signal strength of the messages from different nodes and verifying whether each strength is consistent with a claimed position [149]. One of the most successful methods of detecting a sybil attack is the timestamp series approach [150]. In this approach, the nodes claiming to be in a different position must attach a timestamp that is obtained from a previously crossed RSU. This timestamp cannot be altered and, based on the timestamps on messages from the nodes, we can determine whether or not the nodes are employed in a sybil attack. But this method is hugely dependent upon the RSU infrastructure and incurs a huge cost of implementation. RobSAD (Robust method of Sybil Attack Detection) is another approach to dealing with sybil attacks which analyzes the trajectories of the adjacent nodes [151]. Under usual operating condition, all the nodes have their own trajectories, which cannot be exactly the same. But if an attacker performs a sybil attack, then all the nodes have the same trajectory. By analyzing the trajectory over a period of time, the sybil attack can be detected. This is a fairly simple and effective approach, with lower requirements and a higher success rate.

Timing attacks can be prevented by using the Timing Attack Prevention (TAP) protocol, which enables a vehicle to check a packet for delays before transmitting the packet to other vehicles [152]. Impersonation attacks can be mitigated by including digital signatures, hash-based message authentication code (HMAC), checksums, and active bundles, and replay attacks can be detected by time-stamping messages and using unique sequence numbers to identify messages [153].

4.3. V2I

Within V2I environments, vehicle On-Board Units (OBUs) communicate with Roadside-Units (RSU) to relay information about road conditions. RSUs can authenticate OBUs and grant them Internet access [167]. Within a VANET, both OBUs and RSUs are vulnerable to malicious activity. This section discusses known threats to V2I communication and proposed countermeasures.

4.3.1. Threats

Islam et al. [155] identify potential attacks on V2I. One attack is the distributed denial of service attack, which is described as the unnecessary transmission of information by an attacker that renders road-side unit software unable to function. Other attacks include impersonation attacks, which enable attackers to pose as RSUs or OBUs; malware attacks, which can infect the roadside unit software; and eavesdropping attacks, which allow attackers to gain access to confidential information. Kim et al. [156] discuss the Road-Side Unit (RSU) replication attack, which moves an RSU or replicates it at another location to provide incorrect traffic information and perform erroneous services. There is also a discussion on trust authorities, which evaluate the authenticity of nodes within the VANETs, to examine eavesdropping attacks and monitor vehicle locations. See Table 4 for list of potential attacks in V2X communication.

4.3.2. Countermeasures

In [155], Islam et al. present their developed security architecture “CVGuard”, which they argue can both identify and guard against V2I cyber-attacks, including distributed denial of service attacks. Singh et al. [168] propose a detection method for distributed denial of service attacks in V2I communication using software-defined networking (SDN). Authentication schemes can prevent impersonation and eavesdropping attacks. Guo et al. [169] introduce Fast Pre-distribution Authentication Protocol (FPAP), which authenticates communications between a vehicle and a roadside-unit. Using Liu et al.’s [167] Lightweight V2I Authentication Protocol (LVAP), trusted authorities can provide the group key to incoming OBUs and nearby RSUs. Zhou et al. [170] describe an authentication scheme that relies on a changing private key, which is divided in half, with one half managed by a helper device and the other half by a vehicle. Kim et al. [156] propose a mutual authentication scheme that guards against RSU replication attacks and protects vehicle users from Trust authority attacks. Vassallo and Manaugh [171] discuss defenses against malware in an environment with autonomous vehicles and RSUs. Van der Heijden [172] examines three security architectures that can guard against V2I attacks. Network isolation and slicing provides added flexibility for secured V2X communication. Without network slicing process, overall end-to-end delay, and packet loss can increase significantly [173]. For example, operations such as network convergence, and flow interaction analysis will require isolating and slicing the optical network units or terminals to improve overall bandwidth utilization [174]. Also, Software Defined Networks (SDN) are capable of decoupling the control and data planes of vehicular networks [175] to provide such isolation and slicing.

Table 4
Potential cyber attacks in V2X communications [17].

Attack	Property	Ease of attack	Detection probability	References
Eavesdropping	Confidentiality	High	Low	[154], [22], [35], [113], [155], [155], [156]
GPS Spoofing	Authentication, Privacy	High	Low	[157], [158], [142]
Alteration/Replay	Integrity, Authentication	High	Low	[159], [22], [39], [115], [145]
Magnetic	Privacy, Integrity, Availability, Real-time Constraint	High	Low-Driver, High-System	[17]
Identity tracking	Location, Privacy	High	Low-at High Traffic Density	[160]
Sybil	Authentication, Availability	High	Moderate	[143], [148], [151]
Denial of service	Authentication, Availability	High	High	[22], [39], [67], [112], [155], [161]
Timing	Availability, Real-time Constraint	High	High	[142]
Bogus information	Integrity, Authentication	Moderate	Low-Driver, Moderate-System	[142]
Black hole	Availability, Confidentiality, Integrity	Moderate	Moderate	[162]
Man-in-the-middle	Confidentiality, Integrity, Authentication	Moderate	Moderate	[113], [115], [163], [68]
Injection	Integrity	Moderate	Moderate-Driver, High-System	[22], [56]
Blinding	Privacy, Integrity, Real-Time constraint	Moderate	High	[164], [159], [165]
Illusion	Authentication, Integrity	Low	Low-Driver/System	[141]
Impersonation	Integrity, Authentication	Low	High	[22], [23], [166], [145], [155]

The bibliography sources cited in this table only: [154,157–162,164–166].

4.3.3. V2X handover techniques

The management of handover methods in vehicle to vehicle communication is a daunting task due to the dynamic topological changes in the road infrastructure and conditions. More often, connectivity is the problem, as there are effects of channel fading, and shadowing. Current literature focuses on classifying the handover methods into two types: 1) vertical handover method; and 2) horizontal handover method. In horizontal handover method, a single point of attachment is used (PoA), when transferring the handover process to another PoA, and they are on the same network infrastructure. In vertical handover method, multiple networks are involved during the handover process. Authors in [176], discusses handover methods in OSI layered architectures (Table 5). For example, the link-layer is responsible for seamless connectivity, when vehicles move from one access point to another access point. These techniques are controlled by WLAN link layer handoff mechanisms to provide real time downlink service [177].

A FAST DL-MAP-IE (Downlink-MAP) based MAC messaging system was developed in [177] to provide downlink traffic during handoff process. The network layer handover strategies, such as MIPV6 [178], are applied when the vehicle move from one network (or subnet) to another [178]. These handover approaches suffer from limitations such as packet loss, high latency, and they are not scalable. In order to overcome these issues, a Fast handover for mobile IPv6 (FMIPv6) and hierarchical mobile IPv6 (HMIPv6) were developed by [179]. In transport layer, a mobile Stream Control Transmission Protocol (mSCTP) [180] was proposed, and a session initiating protocol (SIP) was proposed for application layer to handle the handover between different sessions. To improve the performance of cross layer handover mechanisms, message passing approaches between layers were also studied in [179]. A summary of literature on these handover techniques is presented in the Table 5.

4.4. Databases

Connected vehicles face security vulnerabilities related to the transmission of data to both on-board and cloud-based databases. Though vehicles can send data gathered from vehicular sensors and systems to their ECUs, they can also send this data to the cloud as they become more connected and autonomous [163]. Cheng et al. [188] explains the process: initially, sensor data is stored on-board and is processed to remove redundancy. Then the data is sent to cloud or edge-based data storage systems for further processing. Drivers can then receive service-related information relating to weather and parking, customized routes, multimedia content, and an analysis of the vehicle's current safety status [189].

Table 5
V2X handover techniques.

Ref.	Change in network	Cross layer information	Route optimization of exchanged information	Latency	High speed	Layers	Overhead	Handover method	V2V or V2I
[181]	NR	NR	NC	Low	C	Link layer	Low	V	V2I
[178]	R	NR	C	High	C	Network layer	High	V	V2X
[182]	R	R	NC	Low	C	Network layer	High	–	V2I
[179]	R	R	C	Low	C	Network layer	High	–	V2I
[183]	R	R	NC	Low	C	Network layer	Low	V	V2I
[184]	R	R	NC	Low	C	Network layer	Low	V	V2X
[185]	NR	NR	C	High	NA	Application layer	High	–	V2I
[180]	NR	NR	C	High	C	Session layer	Low	–	V2X
[186]	R	NR	NC	High	C	Network layer	High	V	V2X
[187]	NR	NR	C	High	C	Application layer	High	V	V2X

NR: Not Required; R: Required; NC: Not Considered; C: Considered; V: Vertical; V2X: Vehicle to anything; NA: Not Applicable.

Along with the traditional cloud, there is the vehicular cloud, a network that pools together individual vehicle resources. Data collected by the vehicular cloud can be more extensive than the data collected by vehicle sensors and diagnostics. In [189], Xu and Zhou describe data collected within VANETs as one of three categories: location-centric, which consists of traffic-related information; user-centric, which consists of user-specific data relating to "gender, age, education background, and historical data of user access"; and vehicle-centric, which consists of vehicular data obtained by "vehicle sensors and driving recorders." Cheng et al. [188] partition big data in the context of VANETs into the following categories: sensing data, GPS data, autonomous driving data, and vehicular mobile service data.

4.4.1. Threats

Thum et al. [163] discuss the potential for attackers to connect to the a vehicular bus system and conduct injection or man-in-the-middle attacks on in-vehicle databases. Limbasiya and Das [190] list attacks on cloud-based automotive databases, which include denial of service impersonation, tampering, masquerading, replay, and session key disclosure attacks.

4.4.2. Countermeasures

Thum et al. [163] argue that ensuring that ECUs encrypt data before transmission is necessary to prevent injection and man-in-the-middle attacks. If data sent by an ECU is not encrypted, then the database should consider the data illegitimate and the ECU compromised. In the future, ECUs should support public key infrastructure for greater security measures. Thum et al. also introduce AutoDaMa, a customizable vehicular database management system that could provide additional security measures such as the verification of data.

As for databases in the cloud, Sharma and Kaur [191] suggest the incorporation of vehicle authentication and the use of public key infrastructure to encrypt transmissions. To prevent tampering and eavesdropping attacks, Xu and Zhou [189] suggest the incorporation of a hierarchical database encryption system, and Chen et al. [151] propose an authentication scheme that would allow data to be deleted if a vehicle is compromised. Limbasiya and Das [190] describe a scheme to authenticate vehicles before the sender and recipient of a data transmission can communicate. This scheme protects against impersonation, tampering, masquerading, and replay attacks. The scheme's effectiveness on denial of service and session key disclosure attacks was not addressed.

4.5. Clustering

Within VANETs, communication also occurs during the process of clustering, which separates vehicles in VANETs into smaller, hierarchical, and dynamic subgroups. Each vehicle is a node, and each cluster has a cluster head, which serves as a point of communication between its specific cluster and the overall vehicular network [192]. Yang et al. [193] categorize nodes that are not cluster heads as either cluster members, which are nodes within the cluster; isolation nodes, which are not part of a cluster; and temporary members, which are cluster members that cannot reach communication with the cluster head. A significant amount of research on clustering algorithms has been done, mostly focusing on optimization and stability. Cooper et al. [192] and Almheiri and Alqamzi [194] both survey the major VANET clustering algorithms. Cooper et al. provide a taxonomy of clustering approaches, and they also argue that a chain-like topology, where cluster members acting as cluster heads can be beneficial in some situations, such as "platoon formation on a narrow road". Such a structure could mean that boundary nodes are further apart reducing data throughput. Yang et al. [193] propose a scheme that would create a clustering structure that limits the hops between a cluster head and cluster members. Clustering schemes can protect VANETs from being subject to broadcasting storms, which occur when vehicles repeatedly forward transmissions to surrounding nodes, thus causing repetitive, redundant messages to be sent and competition for resources on the network to increase [194].

4.5.1. Threats

Oubabas et al. [195] note that most clustering algorithms assume that all nodes will be trustworthy. As a result, most algorithms focus on stability rather than security. The authors identify several security concerns relating to clustering schemes. Many are tied to the general VANET security concerns, including the falsification of information and the presence of malicious nodes. One specific vulnerability related to clustering is that vehicles might falsify information. Cheng and Huang [196] discuss the potential for sybil and denial of service attacks. During a sybil attack, an attacker can falsely take on several vehicle identities. Since all the vehicles associated with the attacker are traveling in the same direction and with the same speed, they will likely be assigned to the same cluster, and one of the identities will become a cluster head. During a denial of service attack, an attacker can overwhelm the cluster head with transmissions so that other vehicles cannot communicate with the cluster head.

4.5.2. Countermeasures

Oubabas et al. propose a clustering scheme that considers both cluster stability and the trustworthiness of potential cluster heads. This scheme examines cluster nodes' cooperation, meaning their reliable forwarding of messages to other vehicles, and cluster nodes' information legitimacy, which refers to the quality of the information that they broadcast. Cheng and Huang [196] propose two metrics to combat denial of service and sybil attacks. A trajectory similarity metric can detect sybil attacks by identifying cluster heads that are following the exact same path as one of their cluster members. An activity similarity metric can identify denial of

Table 6
Security assessment and virtualization platforms for V2X communications.

Tool	OS compatibility	Utility	Use-case	Type	Reference
Wireshark	Windows/Mac OS X	Packet capture and sniffing	Extract useful information from raw.pcap data and packet analysis	V2V/V2I	[207], [208], [209]
Metasploit	Windows/Linux/Mac OS X	Exploit framework	Gather valuable information and assess vulnerabilities	V2V	[210], [211]
Snort	Windows/Linux	IDS/IPS	Real-time scanning in vehicular environments and rule-based packet capture/analysis	V2V/V2I	[208], [212]
Ettercap	Windows/Linux/Mac OS X	Packet sniffing	Intercept unencrypted data	V2V/V2I	[213]
Burp Suite	Windows/Linux/Mac OS X	Request interception	Intercept unencrypted data	V2V/V2I	[213]
OpenVAS	Windows/Linux/Mac OS X	Vulnerability scanning	Test for vulnerabilities in vehicular communication	V2V	[214], [215]
Kali Linux	Linux (Debian)	Security testing framework	Test for vulnerabilities in vehicular communication/attack simulation	V2V/V2I	[209], [213], [215]
Docker	Windows/Linux	Virtualization containers for real environment simulation	Containers were configured for each communication unit (ECUs, Mother ECUs)	V2V/V2I	[216], [217]
VirtualBox	Windows/Mac OS X/Linux	Hypervisor to run virtual environments	Simulate internal and external networks on guest operating systems using VirtualBox	V2V/V2I	[218]
VMWare	Windows/Mac OS X/ Linux	Hypervisor to run virtual environments	Boot a Fedora environment to run "NCTUns" for network simulation/ emulation	V2V/V2I	[219]

service attacks by comparing the number of messages sent to the cluster head by each cluster member. If one cluster member has an abnormally large proportion of messages within the cluster, then a DoS attack could be taking place. In [197], Mahmood et al. present a hybrid trust management model, which chooses cluster heads based on a metric that examines both trust and resource availability. This model examines both trust and the reliability of information coming from a node. If a node's metrics increase, its history is examined to see if at any point it had metrics below a certain threshold and if so, it is removed from the cluster and prevented from becoming a cluster head.

In [198], Daeinabi et al. introduce the Vehicular Clustering based on Weighted Clustering (VWCA) algorithm which incorporates a distrust value when selecting cluster heads. The distrust value is calculated using the Monitoring of Malicious Vehicles (MMV) algorithm, which sets an initial distrust value to vehicles that join a VANET. All vehicles are then monitored by their neighbors with lower distrust values. If a neighbor notices an abnormal behavior, then a node that is even more trustworthy than the neighbor verifies that the neighbor has lower distrust than the abnormal vehicle, and the abnormal vehicle's distrust value is raised. Gazdar et al. [199] propose a Public Key Infrastructure (PKI) architecture, which includes a trust model and a clustering algorithm that tracks trust and mobility metrics. They also introduce the VANET Dynamic Demilitarized Zone (VDDZ), which prevents vehicles outside of a cluster from communicating directly with the certificate authority. If an unknown vehicle requests to join a cluster, a vehicle within the VDDZ intercepts the transmission and does not forward the request to the certificate authority until the unknown vehicle is authenticated and its trust metric is analyzed.

4.6. V2X simulation platforms

Simulators for automotive platforms serve a dual purpose: 1) First, they are integral to testing cybersecurity solutions that are developed by researchers, as real cars are often not accessible, and 2) Second, they serve as a source of data that can be used for training machine learning algorithms. Hence, the need for cost-effective, easily reconfigurable, and high-performance simulator platforms.

Table 6 presents a collection of simulation used in vehicular environments for modeling real-world scenarios. The "Utility" column provides the main purpose (e.g., packet sniffing, vulnerability scanning, IDS) area of these tools. The "Use-case" column provides a description of utility in vehicular environments. The "Type" columns categorizes whether the tools used were for V2V, V2X or both environments. In addition, Tables 7 and 8 show various traffic and network simulators used for V2X communications. The "GUI" column shows whether graphical interface is supported by the corresponding tool. The "obstacle modelling" column conveys whether obstacles can be modeled by the simulation platform. The "speed control and multi-lane" column show if these vehicular parameters can be set in the simulated environments. The "scalability" column lists if large scale ($>$ 500 nodes) simulation can be supported by the platform or not. See also Table 9.

There are various ways one can model how data messages can be exchanged in simulation platforms. For example, the authors in MobEyes [200–202] discuss a mobility assisted data transport model, where each sensor in the vehicles performs sensing, processing or classification related to an event. During such event occurrence, the model generates meta-data containing critical data features. Here, only meta-data features are exchanged among various vehicles to autonomously execute any task. In Fleanet [203], sensors in a vehicle periodically transmit its data measurements to 1-hop neighbors. Thus, each neighbor only stores local output information, relaying only local information to its neighbors during vehicle is in motion. This strategy limits message exchanges to 1-hop neighbors. Similarly authors in VITP [204] focuses on vehicles pulling only telemetry information by sending requests to other vehicles, and thus allowing vehicles to aggregate (or summarize) and reports only sensitive and limited information exchange. Peer-to-peer overlay networks are used to store distributed information in Senster[205]. Cartel[206] uses opportunistic routing

Table 7
Traffic simulators for V2X communication.

Tool	Type	GUI	Interface	Obstacle modeling	Speed control	Multi-lane	References
SHIFT	V2V	S	C++	NS	S	S	[220]
STRAW	V2V	S	Jist/Swans	NS	NS	NS	[221]
Groovesim	V2V	S	C++	NS	S	NS	[222–224]
Voronoi [9]	V2V	NS	C++	NS	NS	NS	[225]
CanuMobisim [10–12]	V2I	S	Java	S	NS	NS	[226,227]
City	–	S	C++	NS	NS	NS	[228]
UdelModels [14–16]	V2I	S	C++	S	S	S	[229,230]
VanetMobisim	V2X	S	Java	S	S	S	[226,227]
MoVeS	–	S	C++	NS	S	NS	[231]
NCTUns5.0	V2V	S	C++	NS	S	S	[232–235]
SUMO [25,26]	V2X	S	C++	NS	S	S	[236]
MOVE	V2V	S	C++	NS	S	S	[237]
TraNS [28]	V2X	S	C++	NS	S	S	[238]
MobiReal [29]	V2V	S	C++	S	NS	NS	[239]
CARISMA	V2V	S	C++	S	S	S	[240]
FreeSim [33,34]	V2V	S	Java	S	S	NS	[241,242]
VISSIM	V2V	S	C++	S	S	S	[243,244]
CarSims	V2V	NS	C++	NS	S	S	[245,246]

S: Supported; NS: Not Supported.

Table 8
Network simulators for V2X communication.

Tool	Interface	Type	GUI	License	Scalability	References
NS2/NS3	C++	V2X	NS	Open	Small	[247–250]
OMNET++	C++	V2V	S	Free for Academia	Large	[251–254]
GlomoSIM	C	V2V	S	Open	Large	[255,256]
J-Sim	Java	V2V	S	Open	Small	[257]
Qualnet	C	V2X	S	Commercial	Large	[258–260]

S: Supported; NS: Not Supported.

where the closest node is selected for forwarding the data to distribute messages.

5. Research gaps and future scope

Within the field of vehicular communications security, research on CAN exploitation and V2V communication is plentiful. However, other areas are under-represented in the literature. In this section, we call for additional work on the following research gaps:

1. Defense against message spoofing attacks that exploit LIN's master-slave architecture and synchronization capabilities; a low-overhead incorporation of MAC into LIN; intrusion detection in LIN.
2. Protections against man-in-the-middle and replay attacks on FlexRay; intrusion detection in FlexRay; gateway firewalls that could prevent less safety-critical networks, such as LIN, from being able to send messages to the FlexRay bus.
3. MOST communication threats, especially those relating to synchronization disruption and jamming; mitigating MOST attacks; intrusion detection in MOST.
4. Vulnerabilities of vehicular telematics systems.
5. Threats to Zigbee, Bluetooth, Wi-Fi, WiMAX, and UWB in vehicular contexts; defenses against automotive BlueBorne attacks.
6. Attacks on vehicle to database communication; defenses against denial of service and session key disclosure attacks on cloud-based databases.

The past few years have brought about significant changes in the design philosophy of vehicle platforms. OEMs are designing the next generation of vehicle platforms to not only have excellent performance and features but also be an integral part of the intelligent transportation system (ITS) infrastructure. The key technology components in such intelligent vehicles are:

Table 9
Cyber attacks in V2X communications.

Type	Attack & countermeasures	References
Message manipulation	Illusion attack	[145], [141]
	Bogus information attack	[142]
	Sybil attack	[143], [148], [149], [150], [151]
	Timing attack	[142]
	Impersonation attack	[145]
	Alteration attack	[145]
V2I	Threats	[155], [156]
	Countermeasures	[155], [156], [172]
Database	Threats	[163]
	Countermeasures	[151], [189], [261], [191]
Infotainment and telematics	Control override attack	[262], [55]
	Injection attack	[56]
	Countermeasures	[55], [56], [58], [57]
Clustering	Threats & Countermeasures	[195]

1. Over the air updates
2. Partial/fully autonomous (self-driving) vehicles
3. Vehicle platooning

Needless to say, these components have brought about new cybersecurity challenges as well as opportunities since they add additional software components on top of existing control components. For e.g. vehicle platooning adds a V2V component where each vehicle in the platoon shares detailed diagnostic information like velocity, and acceleration with its partner vehicles, which in

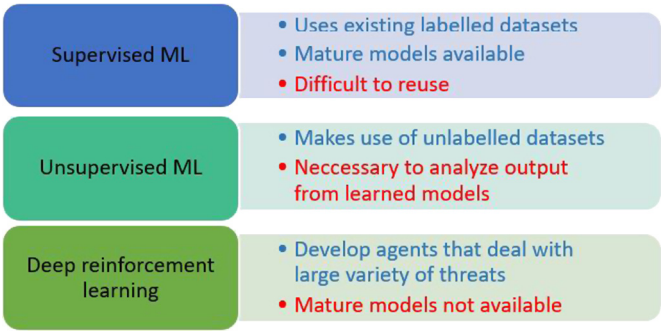

Fig. 9. Salient features of the three different machine learning approaches.

turn use this data for making decisions. A security attack on one vehicle could lead to unsafe behavior by the whole platoon. An essential requirement for intelligent vehicles is that they remain connected to the OEM's automotive cloud during whole or part of their operation. Hence the number of connected vehicles on the road is expected to grow exponentially over the next decade (at a rate of over 23 percent by 2023 according to Global Market Insights [263]).

The same study has found that the automotive industry is focusing on three areas in cybersecurity research to address challenges introduced by the next generation of vehicles. These are a) software/hardware in the loop simulators (HIL/SIL) b) security software for automotive cloud services and c) security software for the automotive platform. In this section we want to focus on how machine learning and blockchain can aid in developing cybersecurity software solutions in both the cloud and the platform. Next generation vehicle platforms are edge computing devices i.e. all the computation required for real-time control takes place onboard the platform and not on the cloud. The cloud plays the role of a high-speed conduit for exchanging information and performing big data analytics. The blockchain, being a distributed ledger, can ensure the security of vehicular communications through the exchange of cryptographic data.

5.1. Cybersecurity software solutions using Machine Learning

Advances in Machine Learning (ML) and Deep Learning (DL) have created a paradigm shift in how software is developed. In software solutions based on ML and DL, the most critical components are not the algorithm/model but rather the availability of data which can be used to train the model to perform useful functions. Fortunately, the connected nature of next-generation vehicles means that hundreds of terabytes of automotive operational and diagnostic data are being generated and stored every day from a wide geographic area, which can be a great resource for developing next-generation cybersecurity solutions both for the automotive platform and cloud. Moreover solutions based on classical statistical models and rule-based logic cannot will not be able to fully utilize data at this scale. Recent surveys like [264], and [265] have captured this trend of increasing research interest in the application of ML and DL for developing cybersecurity solutions.

As shown in Fig. 9, an overview of learning types and their applicability in automotive cybersecurity is listed below:

- Supervised ML model: The simplest anomaly detecting solution is to leverage an existing automotive database with labeled datasets (e.g. a dataset of clean and anomalous CAN messages) to train an ML classification model as shown in recent works like [266]. The same models may be retrained with different automotive platforms using a new dataset.
- Unsupervised/self-supervised ML model: Supervised learning depends on human-labeled datasets, which may not be available and often expensive to create. Using unsupervised learning, we are able to create clusters from various data streams within the vehicle (for e.g. telemetry data from an ECU) which can then be further analyzed to detect anomalous behavior. Clustering algorithms like k-means may be sufficient for data streams having a few features. In order to capture the non-linear relationships in data streams with hundreds of features, deep generative models like autoencoders (AE's) and variational autoencoders (VAE's) are required. The work in [267], and [268] are recent examples showing cybersecurity applications.
- Reinforcement learning: Compared to supervised or unsupervised learning, reinforcement learning (RL) algorithms are less mature. But some of the most noteworthy achievements in machine learning were due to reinforcement learning [269]. RL provides a means to develop autonomous cybersecurity solutions that can take human-defined meta-goals (for e.g. reducing occurrences of manual takeover by a human driver in a self-driving vehicle) as input and take decisions to achieve that goal. An RL based solution would interact with the environment, learn about it, and make decisions using the knowledge it has learned. The authors in [270] discuss an approach to developing such cybersecurity agents using deep reinforcement learning. Recent work in [271] applied deep reinforcement for estimating optimal sub-band and power level of a V2V link.

The key design tasks when developing ML based cybersecurity solutions are determining the a) problem type (e.g. classification, regression), b) learning type (e.g. supervised, unsupervised), and c) model architecture (e.g. trees, dense, and recurrent etc.) that are suitable for vehicular data sets. The type of the ML model used depends primarily on the type of data stream and on the features most relevant to the problem. For e.g., if these features are discrete CAN messages then a densely connected neural network might be sufficient. In case of time series data (for e.g. telemetry data like speed, and engine/motor RPM), models which can take time steps in their input layer like a recurrent neural network or a 1-D convolutional neural network is necessary [272]. Note that an ML/DL based cybersecurity solution need not be limited to using a single architecture or single model. We can use multiple architectures within the same model and multiple models to develop a solution. In the following subsections we list a few cybersecurity solutions and potential applications of machine learning models as illustrated in Fig. 10.

5.1.1. Machine learning for Network security

Detecting and isolating anomalies (intrusion detection) during information exchange between different subsystems or components of the automotive system is a critical task (e.g. telemetry information transmitted between ECU's and ADAS). Anomalies in the data stream can be detected by a) rule-based learning by validating each measurement derived from prior knowledge, b) cross-checking parameters (e.g. velocity, GPS) across data streams from multiple sensors or c) monitoring the data stream over a temporal sliding window to detect suspicious trends. The scale and diversity of the data may often make it impractical to write hand-written rules to detect anomalous signals in every instance of a data stream. One direct application of supervised ML is to replace rule-based intrusion detection algorithm(s) written by humans experts. It will leverage the power of ML and deep learning to find patterns in large datasets. Recent works like [274] and [275] describe the use of feed-forward and recurrent deep learning models for intrusion detection. Solutions based on ML/DL models will have following advantages over rule-based algorithms:

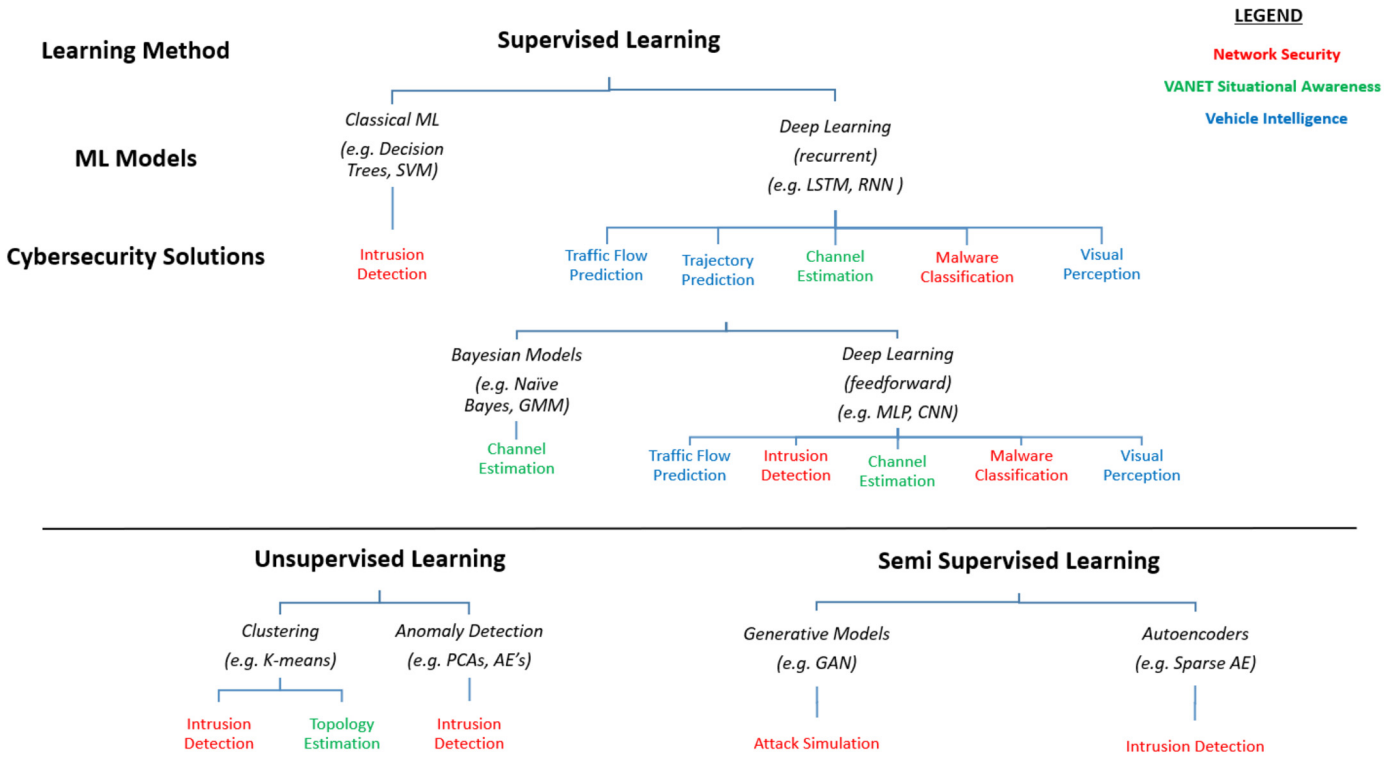

Fig. 10. ML techniques applied to automotive cybersecurity solutions.

Table 10
Machine learning solutions.

Solution class	Solution type	ML models	References
Network security	Intrusion detection	Bayesian, LSTM, MLP	[273], [274], [275]
	Malware classification	RNN, CNN	[276], [277]
	Attack simulation	GAN	[278], [279]
VANET situational awareness	Channel estimation	Bayesian, MLP	[280], [281]
	Topology estimation	Bayesian, Clustering	[282], [283]
Vehicle intelligence	Visual perception	CNN, LSTM	[284], [285]
	Trajectory estimation	Bayesian, LSTM	[286], [287]
Others	Resource allocation	Deep reinforcement learning	[271]

1. Similar detection models for multiple vehicular platforms: An ML detection model is data-driven i.e. it forms rules from the data. Hence a model architecture developed for a particular detection problem in a particular subsystem can be reused across multiple vehicle platforms. For e.g. an ML model developed for intrusion detection in an ADAS system in a sedan may be used for the same application in an SUV, after training it on data relevant to an SUV. Hence using ML models may simplify the development of platform specific cybersecurity software for each automotive sub-system.
2. Robust detection: Deep learning models using LSTM and CNN layers can be used to complex architectures that can use information from multiple temporally and spatially separated data streams to perform robust detection.

Testing detection algorithms by simulating attacks is as important as creating new algorithms. A computationally efficient means of generating high-quality synthetic intrusion signals is necessary for this. Deep generative models like Generative Adversarial Networks (GANs) can be used to generate statistically similar intrusion signals. See Table 10.

5.1.2. VANET situational awareness and vehicular intelligence

Situational awareness refers to perception, comprehension, and projection of the dynamics of an environment. In the automotive cybersecurity context, this may refer to a security solution in the automotive cloud having knowledge of the VANET topology, communication channels and can be of vital importance both in threat assessment and response. This is more challenging than intrusion detection since the solution has to make predictions within a rapidly evolving environment using imperfect information. Fortunately there are several machine learning models suitable for these applications.For e.g. the work in [280] used Gaussian mixture models while [281] used a densely connected DL model for robust estimation of channel state information.

5.1.3. Vehicle intelligence

Network security solutions and situational awareness tools are essentially defensive strategies, and even the best defenses can fail. Hence it is necessary to have a fallback solution that would prevent the automotive platform from taking unreasonable decisions that would endanger the occupants and occupants in other platforms. For e.g. assume that a GPS spoofing attack succeeds and the vehicle is given coordinates that would make it drive into the di-

Table 11
Features of a blockchain network.

Features	Significance
Immutability	-Data can never be tampered by any means after validation and storage.
Distributed environment	-Operates on a peer-to-peer basis. -No single point of failure as there is no central control.
Security	-Data is secure and tamper-proof. -Uses the asymmetric cryptographic algorithms and consensus mechanism. -Defends against the cyber-attacks and prevents fraudulent transactions.
Transparency	-Stores information about every transaction (or event) in a network. -Transparency in accessing information for all members.
Privacy and anonymity	-Identity of the parties involved in the transaction are not revealed. -Information is private and secure.

vider. A fallback solution would prevent such an attack because the vehicle platform would know how to differentiate between safe and unsafe drivable areas of the road. Hence the vehicle would need to have some form of perception of its surroundings.

For vision-based perception, computer vision models based on Convolutional Neural Networks (CNN) [288] are already quite mature and deployed in several production vehicle models. Recent works like [284], [285] describe CNN's specifically designed for segmenting driving surfaces in real-time with high accuracy. Another perception task is to estimate the dynamics of neighboring vehicles by predicting their velocities and trajectories. Earlier work like [286] and [287] have used probabilistic machine learning while recent works like [289] use recurrent deep neural network architectures to estimate vehicle trajectories. In case of the automotive cloud, perception can be in the form of predictions for traffic flow in an area for which recent works like [290] have used deep learning.

5.1.4. Challenges in Machine Learning models

Despite their success in numerous applications, ML/DL based solutions have some fundamental challenges when applied to automotive cybersecurity:

- Protecting ML/DL models against adversarial attacks: In recent years, it has been discovered that convolutional neural networks are susceptible to adversarial inputs that cause them to misclassify images (e.g. classify a car as a tree or a person as a traffic sign) [291]. Since computer vision in almost all ADAS systems uses CNN's, this is a legitimate cybersecurity threat. During the adversarial attack, the attacker adds specific amounts of noise to the image output from the camera (imperceptible to the human eye) before it is processed by the computer vision system. Hence it is necessary to find CNN architectures that are robust enough to adversarial attacks and firewalls to protect the image data pipeline before developing cybersecurity solutions that are dependent on them.
- Optimizing model architectures: The majority of computing operations within ML/DL models are addition and multiplication. However,hundreds of thousands of such operations may be required to complete a single inference pass. Since computing hardware on automotive platforms has hard limits on computing, memory, and power usage, ML/DL based solutions have to be constrained. However techniques like reduced floating-point based precision operations such as FP16 [292], model pruning [293], and specialized computing hardware offer solutions.

5.2. Secure communication in VANETS using Blockchain

Traditional cyber security mitigation approaches are not adequate and robust enough in offering reliable solutions in vehicular networks. Increased connectedness, larger discrepancy in random arrival and departure times of vehicles, and mobility in wireless networks, make VANETs more vulnerable to cyber-attacks. In order to realize secure and efficient communication in vehicular networks, the following challenges need to be addressed:

1. Centralized communication models: Here, all vehicles are identified and connected through central cloud servers. Thus, any center point of failure can throw the entire network into disarray.
2. Lack of privacy: Most of the existing communication networks reveal the user interface data to the requester and thus elevate privacy concerns.
3. Safety: The security breach in terms of data or processes in any of the vehicular functionalities can result in fatal accidents.

5.2.1. Blockchain

Blockchain is an emerging technology and it has the potential to overcome the security challenges of existing VA-NETs and help combat cyber-attacks. A blockchain is a distributed data structure containing blocks that are chained together cryptographically in chronological order [294], [295]. Each block has time-stamped transactions with associated data that are encrypted for secure data transport. More importantly, blockchains execute smart contracts in peer-to-peer (P2P) networks and any data across the members (nodes) are updated using a consensus mechanism. Here, a smart contract is defined as a "collection of code and data that is deployed using cryptographically signed transaction on the blockchain network" and are executed by nodes within the network [296]. All nodes in a blockchain rely on consensus (e.g., rule-based learning) to ensure the consistency of data storage. The commonly used consensus algorithms are Proof of Work (PoW), Proof of Stake (PoS), Byzantine Fault Tolerance (BFT) etc. Thus, the blockchain relies on four major components: distributed ledger platform, an encryption algorithm, a consensus mechanism and smart contract. The important features of a block-chain network are presented in Table 11, [295], [297].

5.2.1.1. Types of blockchain The blockchain can be broadly classified into two categories based on its construction, access, and verification methods, namely: Permissionless (Public) and Permissioned blockchain (Consortium/Private).

Permissionless blockchains can have anyone add a new block in the network while permissioned blockchains are deployed for a particular group of users typically referred to as a consortium or an organization [298]. Permissioned blockchains can be decentralized or centralized and have an authority that authorizes the publishing of blocks while permissionless blockchains are decentralized [296]. A comparison of these blockchain technologies is presented in Table 12, [295], [298].

Table 12
Types of blockchain.

Operational characteristics	Permissionless blockchain	Permissioned blockchain	
	Public	Consortium	Private
Read transaction	Any member	Any member	Any member
Write transaction	Any member	Only pre-selected members	Only one member (or organization)
Number of untrusted writers	High	Low	Low
Throughput	Low	High	High
Latency	Slow	Medium	Medium
Consensus mechanism	PoW and PoS	BFT	BFT
Scenarios	Global decentralized scenarios	Among selected organizations	Information sharing within an organization
Example	Bitcoin and ethereum	Quorum	Hyperledger fabric

5.2.2. Implementation of blockchain in intra-vehicular networks

In [299], the authors propose blockchain for secure data communication between intra-vehicular ECUs. In this scheme, blockchain is implemented in identity based access controllers called MECUs (Mother ECUs) such that all other ECUs have to relay their data to MECUs prior to broadcasting that data on the blockchain network. The authors suggest that a vehicle has multiple MECUs and they all relay their data to the leader MECU which can add blocks to the network given that a consensus mechanism and some security checks are passed. Each MECU verifies (integrity and authenticity checks) data from each ECU before relaying this data to the leader. However, the authors note that some properties of the blockchain are resource intensive and thus render on-board hardware incapable of handling such loads. For this reason, they enable the blockchain to run on MECUs and not on ECUs as they are significantly more powerful in terms of processing power, storage and data speed. The limitations of this scheme are limited storage, and susceptibility to replay attacks as the ECUs/MECUs do not check the timestamp of a transaction prior to sending it to the leader and the possibility of corrupted data being sent to an immutable blockchain.

5.2.3. Implementation of blockchain in V2V/V2X communications

Blockchain technology can be realized in V2V and V2X communication systems to facilitate the secure distribution of basic safety messages or co-operative awareness messages between vehicles and RSUs and/or the cloud platform [297]. In [300], the authors proposed a blockchain framework focusing on an Intelligent Transport Systems (ITS) infrastructure that contains a wireless module following a Wireless Access in Vehicular Environments (WAVE) or IEEE 802.11p standard. On the hardware side, the OBUs (Onboard Units) are equipped to support two-way communication which is enabled between infrastructure to vehicle and/or vehicle to vehicle. The connected vehicles periodically transfer safety messages such as speed (s), position (p) and direction (d), to the network. The ITS infrastructure contains Security Managers (SMs), which aid in message broadcast between vehicles and associated units in blockchain network. These SMs are typically available on the upper layer of the system, and responsible for the timely transfer of data to the neighboring SMs, when a vehicle passes the cross-domain border. The significance of blockchain in VANETs is possibly most evident at this step, as the nodes (e.g., vehicles, RSUs) can share information securely without the need for a central party. On the other hand, in a traditional communication structure, a trusted third-party authority manages all the cryptographic data sent by the participating nodes. This necessitates the need for a complex, and series of exchangeable handshakes via handover methods. This creates significant delay causing latency issues, and thus considered inefficient for real-time applications. Such a delay is easily mitigated in block chain via “transport keys”, as every SM is connected to other SMs in the network.

In [297], the authors describe a blockchain scheme for secure V2V communication as shown in Figs. 11 and 12. Here, all vehicles broadcast their position through beacon messages (e.g.driving status and position of vehicles), where a location certificate (LC) is generated as digital proof. Such a blockchain scheme has two modules: 1) broadcasting module and 2) mining module. In the broadcasting module, a vehicle broadcasts event messages to its neighboring vehicles during an event. The event message has data attributes such as a type of event, pseudo ID, Proof of location and level of trust. The peer vehicles evaluate the trust level of sender vehicle by validating the event message in the mining module. These vehicles use message verification policies to validate the message trustworthiness. Therefore, it is evident that the use of blockchain in VANET ensures the authenticity of the broadcasted messages and at the same time, it satisfies the conditional anonymity. Its distributed data structure offers faster access to information than the central cloud servers.

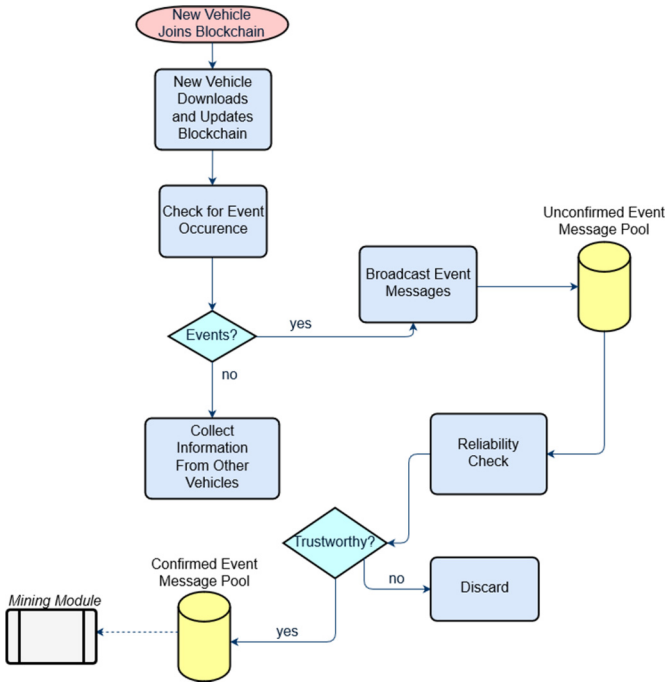

Fig. 11. Broadcast module for blockchain in VANET.

6. Conclusion

The paper provides a hierarchical layer framework to isolate threats and attacks in three layers: sensing, communication, and control layers. As vehicle manufacturers compete with each other in integrating Artificial Intelligence (AI) approaches to modernize the communications for semi or full autonomy, it also opens doors for vulnerabilities. As threat types vary across these layers, understanding its origin, scope and potential impact that it can cause to passenger safety is critical. The paper calls for a need of new architectures that will utilize Systems of Systems (SoS) approach to detect and mitigate threats. Such architectures should focus on se-

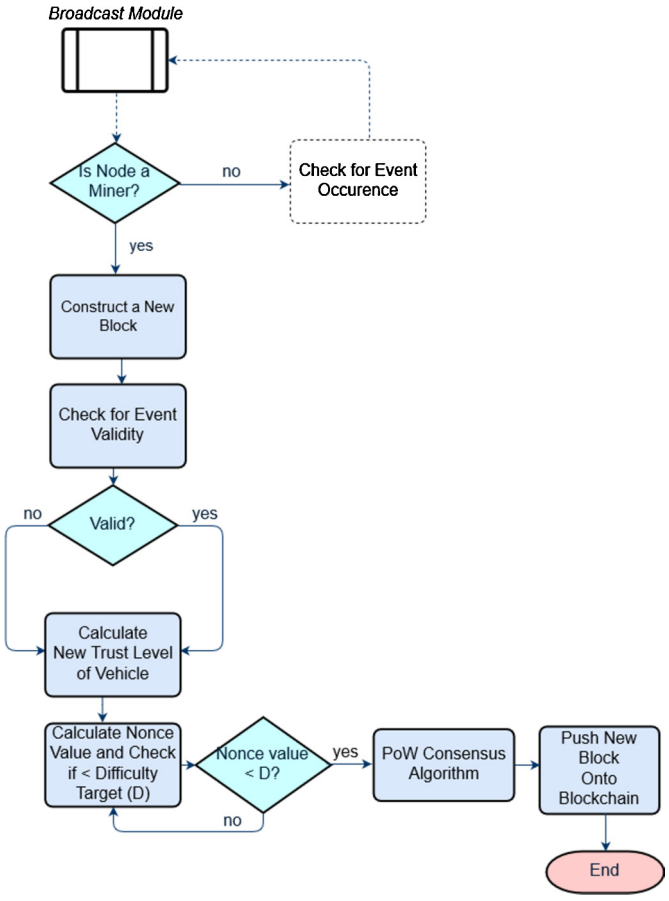

Fig. 12. Mining module for blockchain in VANET.

curing critical units such as power train ECUs with cryptographic and non-crypto based algorithms, registration, authentication procedures, and much more.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

The authors acknowledge the support provided by Prakash Ranganathan (advisor) for his mentoring, Debanjan Sadhukhan (Post Doctoral Fellow), and Niroop Sugunaraj (Undergraduate Research Scholar) for accommodating reviewer's final corrections in the revision of the Figures and Tables.

References

[1] M. Wolf, A. Weimerskirch, P. Christof, Security in automotive bus systems, in: Proceedings of the Workshop on Embedded Security in Cars (ESCAR)'04, 2004, pp. 1–13, http://link.springer.com/10.1007/s10916-017-0883-4.
[2] D. McCandless, Million lines of code — information is beautiful, https://informationisbeautiful.net/visualizations/million-lines-of-code/, 2015.
[3] A. Greenberg, Hackers remotely kill a jeep on the highway—with me in it, https://www.wired.com/2015/07/hackers-remotely-kill-jeep-highway/, 2015.
[4] Andy Greenberg, A new wireless hack can unlock millions of Volkswagens, WIRED, 2016.
[5] A. Greenberg, Tesla responds to Chinese hack with a major security upgrade, https://www.wired.com/2016/09/tesla-responds-chinese-hack-major-security-upgrade/, 2016.
[6] Experimental Security Assessment of BMW Cars: A Summary Report, Technical Report, Keen Security Lab., 2018, https://keenlab.tencent.com/en/Experimental_Security_Assessment_of_BMW_Cars_by_KeenLab.pdf.
[7] H. Onishi, Paradigm change of vehicle cyber security, in: 4th International Conference on Cyber Conflict, 2012, pp. 381–391.
[8] H. Onishi, Guidelines for vehicle cybersecurity, https://docplayer.net/7458872-For-vehicle-cyber-security.html, 2013.
[9] M.H. Eiza, Q. Ni, Driving with sharks: rethinking connected vehicles with vehicle cybersecurity, IEEE Veh. Technol. Mag. (2017).
[10] P. Carsten, M. Yampolskiy, T.R. Andel, J.T. Mcdonald, In-vehicle networks: attacks, vulnerabilities, and proposed solutions, in: Proceedings of the 10th Annual Cyber and Information Security Research Conference, 2015.
[11] A.K. Jadoon, L. Wang, T. Li, M.A. Zia, Lightweight cryptographic techniques for automotive cybersecurity, Wirel. Commun. Mob. Comput. (2018) 1–15.
[12] Z. Lu, G. Qu, Z. Liu, A survey on recent advances in vehicular network security, trust, and privacy, IEEE Trans. Intell. Transp. Syst. (2018).
[13] NHTSA, Cybersecurity protection methods, https://www.nhtsa.gov/technology-innovation/vehicle-cybersecurity, 2018.
[14] C.W. Axelrod, Cybersecurity in the age of autonomous vehicles, intelligent traffic controls and pervasive transportation networks, in: 2017 IEEE Long Island Systems, Applications and Technology Conference, LISAT 2017, 2017.
[15] A.Y. Dak, S. Yahya, M. Kassim, A literature survey on security challenges in VANETs, Int. J. Comput. Theory Eng. 4 (2012).
[16] J. Deng, L. Yu, L. Fu, H. Oluwakemi, R.R. Brooks, Security and data privacy of modern automobiles, in: Data Analytics for Intelligent Transport Systems, Elsevier Inc., 2017, pp. 131–163.
[17] J. Petit, S.E. Shladover, Potential cyberattacks on automated vehicles, IEEE Trans. Intell. Transp. Syst. 16 (2015) 546–556.
[18] O. Henniger, L. Apvrille, A. Fuchs, Y. Roudier, A. Ruddle, B. Weyl, Security requirements for automotive on-board networks, in: 9th International Conference on Intelligent Transport Systems Telecommunications, 2009.
[19] B. Guttman, E.A. Roback, An Introduction to Computer Security: The NIST Handbook, DIANE Publishing, 1995.
[20] L. Zhang, Research on Security and Privacy in Vehicular Ad Hoc Networks, Ph.D. thesis, Universitat Rovira I Virgili, 2010, www.tesisenxarxa.net.
[21] J.H. Kim, S.H. Seo, N.T. Hai, B.M. Cheon, Y.S. Lee, J.W. Jeon, Gateway framework for in-vehicle networks based on CAN, flexray, and ethernet, IEEE Trans. Veh. Technol. 64 (2015) 4472–4486.
[22] J. Liu, W. Sun, Yongpeng Shi, In-vehicle network attacks and countermeasures: challenges and future directions, IEEE Netw. 31 (2017) 50–58.
[23] W. Choi, K. Joo, H.J. Jo, M.C. Park, D.H. Lee, VoltageIDS: low-level communication characteristics for automotive intrusion detection system, IEEE Trans. Inf. Forensics Secur. 13 (2018) 2114–2129.
[24] H. Ueda, R. Kurachi, H. Takada, T. Mizutani, M. Inoue, S. Horihata, Security authentication system for in-vehicle network, SEI Tech. Rev. (2015).
[25] N. Nowdehi, A. Lautenbach, T. Olovsson, In-vehicle CAN message authentication: an evaluation based on industrial criteria, in: IEEE 86th Vehicular Technology Conference, 2017.
[26] P. Mundhenk, S. Steinhorst, M. Lukasiewycz, S.A. Fahmy, S. Chakraborty, Lightweight authentication for secure automotive networks, in: Proceedings of the 2015 Design, Automation & Test in Europe Conference & Exhibition, 2015, pp. 285–288.
[27] K.D. Kang, Y. Baek, S. Lee, S.H. Son, An attack-resilient source authentication protocol in controller area network, in: ACM/IEEE Symposium on Architectures for Networking and Communications Systems, 2017, pp. 109–118.
[28] A. Tashiro, H. Muraoka, S. Araki, K. Kakizaki, S. Uehara, A secure protocol consisting of two different security-level message authentications over CAN, in: 3rd IEEE International Conference on Computer and Communications, 2017, pp. 1520–1524.
[29] W. Choi, H.J. Jo, S. Woo, J.Y. Chun, J. Park, D.H. Lee, Identifying ECUs using inimitable characteristics of signals in controller area networks, IEEE Trans. Veh. Technol. 67 (2018) 4757–4770.
[30] A. Tomlinson, J. Bryans, S.A. Shaikh, H.K. Kalutarage, Detection of automotive CAN cyber-attacks by identifying packet timing anomalies in time windows, in: 48th Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshops, 2018.
[31] H. Lee, S.H. Ergen, Jeong, H.K. Kim, OTIDS: a novel intrusion detection system for in-vehicle network by using remote frame, in: 15th Annual Conference on Privacy, Security and Trust (PST), 2017.
[32] B. Groza, P.S. Murvay, Security solutions for the controller area network: bringing authentication to in-vehicle networks, IEEE Veh. Technol. Mag. (2018) 40–47.
[33] J. Takahashi, Y. Aragane, T. Miyazawa, H. Fuji, H. Yamashita, K. Hayakawa, S. Ukai, H. Hayakawa, Automotive attacks and countermeasures on LIN-Bus, J. Inf. Process. 25 (2017) 220–228.
[34] Z. Gu, G. Han, H. Zeng, Q. Zhao, Security-aware mapping and scheduling with hardware co-processors for flexray-based distributed embedded systems, IEEE Trans. Parallel Distrib. Syst. 27 (2016) 3044–3057.
[35] A.R. Mousa, P. NourElDeen, M. Azer, M. Allam, Lightweight authentication protocol deployment over flexray, in: Proceedings of the 10th International Conference on Informatics and Systems, 2016, pp. 233–239.
[36] G. Han, H. Zeng, Y. Li, W. Dou, SAFE: security-aware flexray scheduling engine, in: Proceedings of the Conference on Design, Automation & Test in Europe, 2014.

[37] D. Püllen, N.A. Anagnostopoulos, T. Arul, S. Katzenbeisser, Security and safety co-engineering of the flexray bus in vehicular networks, in: Proceedings of the International Conference on Omni-Layer Intelligent Systems, COINS '19, ACM, New York, NY, USA, 2019, pp. 31–37, http://doi.acm.org/10.1145/3312614.3312626.

[38] C.-W. Lin, H. Yu, INVITED: cooperation or competition? Coexistence of safety and security in next-generation ethernet-based automotive networks, in: 53rd ACM/EDAC/IEEE Design Automation Conference, 2016.

[39] T. Kiravuo, M. Sarela, J. Manner, A survey of ethernet LAN security, IEEE Commun. Surv. Tutor. 15 (2013) 1477–1491.

[40] K.D. Mitnick, W. Simon, The Art of Intrusion: The Real Stories Behind the Exploits of Hackers, Intruders & Deceivers, Wiley Publishing, Inc, 2005.

[41] S. Convery, Hacking layer 2: fun with ethernet switches, https://www.blackhat.com/presentations/bh-usa-02/bh-us-02-convery-switches.pdf, 2002.

[42] C.L. Abad, R.I. Bonilla, An analysis on the schemes for detecting and preventing arp cache poisoning attacks, in: 27th International Conference on Distributed Computing Systems Workshops, ICDCSW'07, 2007, p. 60.

[43] G.M. Marro, Attacks at the Data Link Layer, Ph.D. thesis, 2003, http://seclab.cs.ucdavis.edu/papers/Marro_masters_thesis.pdf.

[44] E. Norris, Analysis of a telnet session hijack via spoofed mac addresses and session resynchronization analysis of a telnet session hijack via spoofed mac addresses and session resynchronization, https://www.giac.org/paper/gsec/552/analysis-telnet-session-hijack-spoofed-mac-addresses-sessionresynchronization/101288, 2001.

[45] O. Zheng, J. Poon, K. Beznosov, Application-based tcp hijacking, in: Proceedings of the Second European Workshop on System Security, EUROSEC '09, ACM, New York, NY, USA, 2009, pp. 9–15.

[46] O. Avatefipour, H. Malik, State-of-the-art survey on in-vehicle network communication (can-bus) security and vulnerabilities, preprint, arXiv:1802.01725, 2018.

[47] P. Vasile, B. Groza, S. Murvay, Performance analysis of broadcast authentication protocols on CAN-FD and FlexRay, in: Proceedings of the WESS'15: Workshop on Embedded Systems Security, 2015.

[48] C. Bernardini, M.R. Asghar, B. Crispo, Security and privacy in vehicular communications: challenges and opportunities, Veh. Commun. 10 (2017) 13–28.

[49] A. Perrig, R. Canetti, J.D. Tygar, D. Song, The Tesla broadcast authentication protocol, CryptoBytes 5 (2002) 2–13.

[50] P. Meyer, T. Häckel, F. Korf, T.C. Schmidt, Dos protection through credit based metering – simulation based evaluation for time-sensitive networking in cars, arXiv:1908.09646, 2019.

[51] M.D. Pesé, K. Schmidt, H. Zweck, Hardware/software Co-design of an Automotive Embedded Firewall, SAE Technical Paper, SAE International, 2017.

[52] K. Jaisingh, K. El-Khatib, R. Akalu, Paving the way for intelligent transport systems (ITS): privacy implications of vehicle infotainment and telematics systems, in: Proceedings of the 6th ACM Symposium on Development and Analysis of Intelligent Vehicular Networks and Applications, 2016, pp. 25–31.

[53] D. Spaar, Beemer, open thyself! – security vulnerabilities in BMW's connecteddrive, heise, 2015.

[54] T. Bécsi, S. Aradi, P. Gáspár, Security issues and vulnerabilities in connected car systems, in: Models and Technologies for Intelligent Transportation Systems, 2015, pp. 477–482.

[55] H.J. Jo, W. Choi, S.Y. Na, S. Woo, D.H. Lee, Vulnerabilities of Android OS-based telematics system, Wirel. Pers. Commun. 92 (2017) 1512–1530.

[56] S. Mazloom, M. Rezaeirad, A. Hunter, D. Mccoy, A security analysis of an in vehicle infotainment and app platform, in: Proceedings of the 10th USENIX Conference on Offensive Technologies, 2016, pp. 232–243.

[57] IEEE Cyber Security, Design flaws and security considerations for telematics and infotainment systems, https://cybersecurity.ieee.org/blog/2017/05/30/design, 2017.

[58] S. Lee, J.-H. Lee, TEE based session key establishment protocol for secure infotainment systems, Des. Autom. Embed. Syst. 22 (2018) 215–224.

[59] A.K. Mandal, A. Cortesi, P. Ferrara, F. Panarotto, F. Spoto, Vulnerability analysis of Android auto infotainment apps, in: Proceedings of the 15th ACM International Conference on Computing Frontiers, CF '18, ACM, New York, NY, USA, 2018, pp. 183–190.

[60] EPA, On-Board Diagnostic (OBD) Regulations and Requirements: Questions and Answers, Technical Report, 2003.

[61] C. Valasek, C. Miller, Adventures in Automotive Networks and Control Units, Technical Report, 2014.

[62] A. Kovelman, A remote attack on the bosch drivelog connector dongle - argus cyber security, https://argus-sec.com/remote-attack-bosch-drivelog-connector-dongle/, 2017.

[63] D.J. Klinedinst, C. King, On Board Diagnostics: Risks and Vulnerabilities of the Connected Vehicle, Technical Report, Carnegie Mellon University Software Engineering Institute, 2016.

[64] H. Onishi, K. Wu, K. Yoshida, T. Kato, Approaches for vehicle cyber-security in the US: vulnerabilities of carry-in devices, GNSS, & vehicle-to-vehicle communication, Int. J. Automot. Eng. (2017).

[65] Security Research Lab, USB peripherals can turn against their users, https://srlabs.de/bites/badusb/, 2014.

[66] Z. Cai, A. Wang, W. Zhang, 0-days & mitigations: roadways to exploit and secure connected BMW cars, in: Black Hat USA, 2019.

[67] M.A. Mustafa, N. Zhang, G. Kalogridis, Z. Fan, Smart electric vehicle charging: security analysis, in: IEEE PES Innovative Smart Grid Technologies Conference, 2013.

[68] S. Fries, R. Falk, Electric vehicle charging infrastructure-security considerations and approaches, in: INTERNET 2012: the Fourth International Conference on Evolving Internet, 2012, pp. 58–64, https://www.researchgate.net/publication/279852975.

[69] X. Sun, L. Xia, S. Jia, Enhancing location privacy for electric vehicles by obfuscating the linkages of charging events, in: Proceedings of 2015 IEEE 5th International Conference on Electronics Information and Emergency Communication, 2015, pp. 155–158.

[70] C. Alcaraz, J. Lopez, S. Wolthusen, OCPP protocol: security threats and challenges, IEEE Trans. Smart Grid 8 (2017) 2452–2459.

[71] B. Vaidya, H.T. Mouftah, Deployment of secure EV charging system using open charge point protocol, in: 14th International Wireless Communications & Mobile Computing Conference, 2018, pp. 922–927.

[72] S. Lee, Y. Park, H. Lim, T. Shon, Study on analysis of security vulnerabilities and countermeasures in ISO/IEC 15118 based electric vehicle charging technology, in: International Conference on IT Convergence and Security, 2014.

[73] A.C. Chan, J. Zhou, A secure, intelligent electric vehicle ecosystem for safe integration with the smart grid, IEEE Trans. Intell. Transp. Syst. 16 (2015) 3367–3376.

[74] B. Roberts, K. Akkaya, E. Bulut, M. Kisacikoglu, An authentication framework for electric vehicle-to-electric vehicle charging applications, in: 2017 IEEE 14th International Conference on Mobile Ad Hoc and Sensor Systems (MASS), 2017, pp. 565–569.

[75] G.S. Morrison, Threats and mitigation of ddos cyberattacks against the US, 2018.

[76] X. Huang, C. Xu, P. Wang, H. Liu, LNSC: a security model for electric vehicle and charging pile management based on blockchain ecosystem, IEEE Access 6 (2018) 13565–13574.

[77] A.G. Morosan, F. Pop, Ocpp security - neural network for detecting malicious traffic, in: Proceedings of the International Conference on Research in Adaptive and Convergent Systems, RACS '17, ACM, New York, NY, USA, 2017, pp. 190–195.

[78] L. Buschlinger, M. Springer, M. Zhdanova, Plug-and-patch: secure value added services for electric vehicle charging, in: Proceedings of the 14th International Conference on Availability, Reliability and Security, ARES '19, ACM, New York, NY, USA, 2019, pp. 2:1–2:10.

[79] S. Van De Beek, F. Leferink, Vulnerability of remote keyless-entry systems against pulsed electromagnetic interference and possible improvements, IEEE Trans. Electromagn. Compat. 58 (2016) 1259–1265.

[80] F.D. Garcia, D. Oswald, Lock it and still lose it-on the (in)security of automotive remote keyless entry systems, in: Proceedings of the 25th USENIX Security Symposium, 2016, https://www.usenix.org/conference/usenixsecurity16/technical-sessions/presentation/garcia.

[81] J. Wetzels, Broken keys to the kingdom: security and privacy aspects of RFID-based car keys, Computing Research Repository (2014).

[82] T. Glocker, T. Mantere, M. Elmusrati, A protocol for a secure remote keyless entry system applicable in vehicles using symmetric-key cryptography, in: 8th International Conference on Information and Communication Systems (ICICS), IEEE, 2017, pp. 310–315.

[83] H.-L. Liu, J.-S. Ma, S.-Y. Zhu, Z.-J. Lu, Z.-L. Liu, Practical contactless attacks on hitag2-based immobilizer and RKE systems, in: International Conference on Computer, Communication and Network Technology, 2018, pp. 505–512.

[84] E. Hamadaqa, A. Mars, W. Adi, S. Mulhem, Clone-resistant vehicular RKE by deploying SUC, in: 7th International Conference on Emerging Security Technologies, 2017, pp. 221–225.

[85] Q. Zhang, M. Almulla, A. Boukerche, An improved scheme for key management of RFID in vehicular adhoc networks, IEEE Latin Am. Trans. 11 (2013) 1286–1294.

[86] J.D. Lee, H.J. Im, W.M. Kang, J.H. Park, Ubi-RKE: a rhythm key based encryption scheme for ubiquitous devices, Math. Probl. Eng. (2014).

[87] C. Laurendeau, M. Barbeau, Threats to security in DSRC/WAVE, in: Ad-Hoc, Mobile, and Wireless Networks, 2006, pp. 266–279.

[88] S. Ucar, S.C. Ergen, O. Ozkasap, Security vulnerabilities of IEEE 802.11p and visible light communication based platoon, in: 2016 IEEE Vehicular Networking Conference, 2016.

[89] S. Ucar, S.C. Ergen, O. Ozkasap, IEEE 802.11p and visible light hybrid communication based secure autonomous platoon, IEEE Trans. Veh. Technol. 67 (2018) 8667–8681.

[90] S. Biswas, J. Mišić, V. Mišić, DDoS attack on WAVE-enabled VANET through synchronization, in: Globecom 2012: Communication and Information System Security Symposium, 2012, pp. 1079–1084.

[91] W. Whyte, J. Petit, V. Kumar, J. Moring, R. Roy, Threat and countermeasures analysis for WAVE service advertisement, in: IEEE 18th International Conference on Intelligent Transportation Systems, 2015, pp. 1061–1068.

[92] K. Chandra Dey, A. Rayamajhi, M. Chowdhury, P. Bhavsar, J. Martin, Vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communication in a heterogeneous wireless network - performance evaluation, Transp. Res., Part C 68 (2016) 168–184.

[93] Z. Liu, T. Liang, J. Guo, L. Zhang, Priority-based access for dsrc and 802.11p vehicular safety communication, in: International Conference on Connected Vehicles and Expo, 2012, pp. 103–107.

[94] I. Ivanov, C. Maple, T. Watson, S. Lee, Cyber security standards and issues in V2X communications for Internet of vehicles, in: Living in the Internet of Things: Cybersecurity of the IoT, 2018.

[95] Department of Transportation, IEEE 1609-Family of Standards for Wireless Access in Vehicular Environments (WAVE), Technical Report Department of Transportation, 2009, https://www.standards.its.dot.gov/Factsheets/Factsheet/80.

[96] Y. Li, An overview of the DSRC/WAVE technology, in: Quality, Reliability, Security and Robustness in Heterogeneous Networks, Springer, Berlin, Heidelberg, 2010, pp. 544–558, https://link.springer.com/chapter/10.1007/978-3-642-29222-4_38.

[97] M. Muhammad, G.A. Safdar, Survey on existing authentication issues for cellular-assisted V2X communication, Veh. Commun. 12 (2018) 50–65.

[98] J. Cichonski, J.M. Franklin, M. Bartock, Guide to LTE Security, Technical Report, NIST, 2017.

[99] J. Cao, M. Ma, H. Li, Y. Zhang, Z. Luo, A survey on security aspects for LTE and LTE-A networks, IEEE Commun. Surv. Tutor. 16 (2014) 283–302.

[100] D. Rupprecht, K. Jansen, C. Pöpper, Putting LTE security functions to the test: a framework to evaluate implementation correctness, in: Proceedings of the 10th USENIX Conference on Offensive Technologies, 2016, pp. 40–51.

[101] R.P. Jover, LTE security, protocol exploits and location tracking experimentation with low-cost software radio, Computing Research Repository (2016).

[102] V. Marojevic, C-V2X Security Requirements and Procedures: Survey and Research Directions, 2018.

[103] M. Liyanage, I. Ahmad, M. Ylianttila, A. Gurtov, A.B. Abro, E.M. De Oca, Leveraging LTE security with SDN and NFV, in: IEEE 10th International Conference on Industrial and Information Systems, 2016, pp. 220–225.

[104] M. Sun, Y. Qian, Study and application of security based on ZigBee standard, in: Third International Conference on Multimedia Information Networking and Security, 2011, pp. 508–511.

[105] O. Olawumi, K. Haataja, M. Asikainen, N. Vidgren, P. Toivanen, Three practical attacks against ZigBee security: attack scenario definitions, practical experiments, countermeasures, and lessons learned, in: 14th International Conference on Hybrid Intelligent Systems, 2014, pp. 199–206.

[106] T. Zillner, ZIGBEE Exploited: the Good, the Bad and the Ugly, Technical Report, Cognosec, 2015.

[107] X. Fan, F. Susan, W. Long, S. Li, Security Analysis of Zigbee, Technical Report, 2017.

[108] B. Seri, G. Vishnepolsky, BlueBorne, Technical Report, Armis, 2017.

[109] C. Kolias, G. Kambourakis, S. Gritzalis, Attacks and countermeasures on 802.16: analysis and assessment, IEEE Commun. Surv. Tutor. 15 (2013) 487–514.

[110] S. Nie, L. Liu, Y. Du, Free-fall: hacking Tesla from wireless to can bus, in: DEFCON, 2017.

[111] T.D. Vo-Huu, T.D. Vo-Huu, G. Noubir, Interleaving jamming in wi-fi networks, in: Proceedings of the 9th ACM Conference on Security & Privacy in Wireless and Mobile Networks, 2016, pp. 31–42.

[112] M. Vanhoef, F. Piessens, Denial-of-service attacks against the 4-way wi-fi handshake, in: 9th International Conference on Networks & Communications, 2017.

[113] K. Scarfone, C. Tibbs, M. Sexton, Guide to Securing Wimax Wireless Communications: Recommendations of the National Institute of Standards and Technology, Technical Report, NIST, 2010.

[114] A.P. Hennessy, Implementation of Physical Layer Security of an Ultra-Wideband Transceiver, Ph.D. thesis, 2016.

[115] J.S. Cho, Y.S. Jeong, S.O. Park, Consideration on the brute-force attack cost and retrieval cost: a hash-based radio-frequency identification (RFID) tag mutual authentication protocol, Comput. Math. Appl. 69 (2015) 58–65.

[116] C. Zhang, W. Zhang, H. Mu, A mutual authentication security RFID protocol based on time stamp, in: 2015 1st International Conference on Computational Intelligence Theory, Systems and Applications, 2015, pp. 166–170.

[117] N. Lyamin, A. Vinel, M. Jonsson, J. Loo, Real-time detection of denial-of-service attacks in IEEE 802.11p vehicular networks, IEEE Commun. Lett. 18 (2014) 110–113.

[118] H. Nguyen-Minh, A. Benslimane, A. Rachedi, Jamming detection on 802.11p under multi-channel operation in vehicular networks, in: 2015 IEEE 11th International Conference on Wireless and Mobile Computing, Networking and Communications, 2015, pp. 764–770.

[119] N. Ghambir, P. Sharma, A hybrid approach for intelligent communication and performance analysis over DSRC VANET, in: IEEE International Conference on Information, Communication, Instrumentation and Control, 2017.

[120] K. Kaur, A.S. Sharma, H.S. Sohal, A. Kaur, Adaptive random key scheme for authentication and key agreement (ARKS-AKA) for efficient LTE security, in: Proceedings of 2015 RAECS UIET Panjab University Chandigarh, IEEE, 2015.

[121] T. Nandhakumar, R. Guruprasath, R. Madhumita, S. Janani, V2I technology and energy efficient solution using Zigbee, Int. J. Sci. Res. Dev. 4 (2017) 139–143.

[122] Y. Lei, J. Wu, Study of applying ZigBee technology into forward collision warning system (FCWS) under low-speed circumstance, in: 25th Wireless and Optical Communication Conference, 2016.

[123] S. Pawade, S. Shah, D. Tijare, Zigbee based intelligent driver assistance system, Int. J. Eng. Res. Appl. 3 (2013) 1463–1468.

[124] R.A. Gheorghiu, A.C. Cormos, V.A. Stan, V. Iordache, Overview of network topologies for V2X communications, in: Proceedings of the 9th International Conference on Electronics, Computers and Artificial Intelligence, 2017.

[125] M. Cheah, S.A. Shaikh, O. Haas, A. Ruddle, Towards a systematic security evaluation of the automotive bluetooth interface, Veh. Commun. 9 (2017) 8–18.

[126] F. Moradi, H. Mala, B.T. Ladani, Security analysis and strengthening of an RFID lightweight authentication protocol suitable for VANETs, Wirel. Pers. Commun. 83 (2015) 2607–2621.

[127] W.I. Khedr, SRFID: a hash-based security scheme for low cost RFID systems, Egypt. Inform. J. 14 (2013) 89–98.

[128] P. Gope, T. Hwang, A realistic lightweight authentication protocol preserving strong anonymity for securing RFID system, Comput. Secur. 55 (2015) 271–280.

[129] Q. Qian, Y.-L. Jia, R. Zhang, A lightweight RFID security protocol based on elliptic curve cryptography, Int. J. Netw. Secur. 18 (2016) 354–361.

[130] Y.P. Liao, C.M. Hsiao, A secure ECC-based RFID authentication scheme integrated with ID-verifier transfer protocol, Ad Hoc Netw. 18 (2014) 133–146.

[131] V.R. Vijaykumar, S. Elango, Hardware implementation of tag-reader mutual authentication protocol for RFID systems, Integration 47 (2014) 123–129.

[132] GSMA, Cellular Vehicle-to-Everything (C-V2X) Enabling Intelligent Transport, Technical Report, 2017.

[133] K. Tanuja, T.M. Sushma, M. Bharathi, K.H. Arun, A survey on VANET technologies, Int. J. Comput. Appl. 121 (2015).

[134] O. Nakhila, E. Dondyk, M.F. Amjad, C. Zou, User-side wi-fi evil twin attack detection using random wireless channel monitoring, in: 12th Annual IEEE Consumer Communications and Networking Conference, 2015.

[135] S. Hu, M. Kang, C. She, Vehicle positioning based on UWB technology, J. Phys., Conf. Ser. (2017), https://doi.org/10.1088/1742-6596/887/1/012069.

[136] M.S. Anwer, C. Guy, A survey of VANET technologies, J. Emerg. Trends Comp. Inf. Sci. 5 (2014) 661–671.

[137] H. Stoll, P. Zimmer, F. Hartmann, E. Sax, GPS-independent localization for off-road vehicles using ultra-wideband (UWB), in: IEEE 20th International Conference on Intelligent Transportation Systems, Proceedings, ITSC, 2017.

[138] J. Tiemann, J. Pillmann, S. Böcker, C. Wietfeld, Ultra-wideband aided precision parking for wireless power transfer to electric vehicles in real life scenarios, in: IEEE 84th Vehicular Technology Conference, 2016.

[139] T.P. Oman, K.J. Hawes, Relay Attack Prevention for Passive Entry Passive Start (PEPS) Vehicle Security Systems, 2015.

[140] Y. Zhang, W. Liu, Y. Fang, D. Wu, Secure localization and authentication in ultra-wideband sensor networks, IEEE J. Sel. Areas Commun. 24 (2006).

[141] N.W. Lo, H.C. Tsai, Illusion attack on VANET applications - a message plausibility problem, in: IEEE Global Telecommunications Conference, 2007.

[142] V.H. La, A. Cavalli, Security attacks and solutions in vehicular ad hoc networks: a survey, Int. J. AdHoc Netw. Syst. 4 (2014).

[143] J.R. Douceur, The sybil attack, in: International Workshop on Peer-to-Peer Systems, 2002, pp. 251–260.

[144] I.A. Sumra, J.-L.A. Manan, H. Hasbullah, Timing attack in vehicular network, in: Proceedings of the 15th WSEAS International Conference on Computers, World Scientific and Engineering Academy and Society (WSEAS), Stevens Point, Wisconsin, USA, 2011, pp. 151–155, http://dl.acm.org/citation.cfm?id=2028299.2028330.

[145] Kasra Amirtahmasebi, Seyed Reza Jalalinia, Vehicular Networks-Security, Vulnerabilities and Countermeasures, Ph.D. thesis, University of Gothenburg, Goteborg, Sweden, 2010.

[146] B. Parno, A. Perrig, Challenges in securing vehicular networks, in: Proceedings of the Workshop on Hot Topics in Networks (HotNets-IV), 2005.

[147] S.S. Manvi, M.S. Kakkasageri, D.G. Adiga, Message authentication in vehicular ad hoc networks: ECDSA based approach, in: International Conference on Future Computer and Communication, 2009, pp. 16–20.

[148] J. Newsome, E. Shi, D. Song, A. Perrig, The sybil attack in sensor networks, in: Third International Symposium on Information Processing in Sensor Networks, 2004.

[149] B. Xiao, B. Yu, C. Gao, Detection and Localization of Sybil Nodes in VANETs*, Technical Report, 2006, http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.111.4836&rep=rep1&type=pdf.

[150] S. Park, B. Aslam, D. Turgut, C.C. Zou, Defense against sybil attack in the initial deployment stage of vehicular ad hoc network based on roadside unit support, Secur. Commun. Netw. 6 (2013) 523–538.

[151] C. Chen, F. Shi, H. Yu, N. Fei, Anonymous authentication based on cloud storage for cross-regional vehicles in VANET, in: 2016 IEEE International Conference on Ubiquitous Wireless Broadband, 2016.

[152] A. Arsalan, R. Rehman, Prevention of timing attack in software defined named data network with vanets, in: 2018 International Conference on Frontiers of

Information Technology (FIT), IEEE Computer Society, Los Alamitos, CA, USA, 2018, pp. 247–252, https://doi-ieeecomputersociety-org.proxymu.wrlc.org/10.1109/FIT.2018.00050.
[153] B. Bhargava, A.M. Johnson, G.I. Munyengabe, P. Angin, A systematic approach for attack analysis and mitigation in V2V networks, Integration 7 (2016) 79–96.
[154] I. Rouf, R. Miller, H. Mustafa, T. Taylor, S. Oh, W. Xu, M. Gruteser, W. Trappe, I. Seskar, Security and privacy vulnerabilities of in-car wireless networks: a tire pressure monitoring system case study, in: USENIX Security'10 Proceedings of the 19th USENIX Conference on Security, 2010.
[155] M. Islam, M. Chowdhury, H. Li, H. Hu, Cybersecurity attacks in vehicle-to-infrastructure (V2I) applications and their prevention, Computing Research Repository (2017).
[156] J.Y. Kim, H.K. Choi, J.A. Copeland, An efficient authentication scheme for security and privacy preservation in V2I communications, in: IEEE 72nd Vehicular Technology Conference, 2010.
[157] S. Narain, A. Ranganathan, G. Noubir, Security of GPS/INS based on-road location tracking systems, Computing Research Repository (2018).
[158] S. Bittl, A.A. Gonzalez, M. Myrtus, H. Beckmann, S. Sailer, B. Eissfeller, Emerging attacks on VANET security based on GPS time spoofing, in: 2015 IEEE Conference on Communications and Network Security, 2015, pp. 344–352.
[159] J. Petit, S. Bas, M. Feiri, F. Kargl, Remote attacks on automated vehicles sensors: experiments on camera and lidarn, in: Black Hat Europe, 2015.
[160] R.S. Raw, M. Kumar, N. Singh, Security challenges, issues and their solutions for VANET, Int. J. Netw. Secur. Appl. 5 (2013) 95–105.
[161] L. He, W.T. Zhu, Mitigating DoS attacks against signature-based authentication in VANETs, in: IEEE International Conference on Computer Science and Automation Engineering, 2012, pp. 261–265.
[162] R.A.R. Mahmood, A.I. Khan, A survey on detecting black hole attack in AODV-based mobile ad hoc networks, in: International Symposium on High Capacity Optical Networks and Enabling Technologies, 2007, https://www.researchgate.net/publication/4362772.
[163] T. Thüm, S. Schulze, M. Pukall, G. Saake, S. Günther, Secure and customizable data management for automotive systems: a feasibility study, ISRN Softw. Eng. (2012) 2012.
[164] H. Shin, D. Kim, Y. Kwon, Y. Kim, Illusion and dazzle: adversarial optical channel exploits against lidars for automotive applications, in: Cryptographic Hardware and Embedded Systems – CHES 2017, 2017, pp. 445–467.
[165] C. Yan, W. Xu, J. Liu, Can you trust autonomous vehicles: contactless attacks against sensors of self-driving vehicle, in: DEFCON, 2016.
[166] D. Singelee, B. Preneel, Location verification using secure distance bounding protocols, in: IEEE International Conference on Mobile Adhoc and Sensor Systems Conference, 2005.
[167] Y. Liu, W. Guo, Q. Zhong, G. Yao, Lvap: lightweight v2i authentication protocol using group communication in vanets, Int. J. Commun. Syst. 30 (2017) e3317, E3317 IJCS-16-0721.R1.
[168] P.K. Singh, S. Kumar Jha, S.K. Nandi, S. Nandi, Ml-based approach to detect ddos attack in v2i communication under sdn architecture, in: TENCON 2018 - 2018 IEEE Region 10 Conference, 2018, pp. 0144–0149.
[169] W. Guo, Y. Liu, J. Wang, Fpap: fast pre-distribution authentication protocol for v2i, in: X. Sun, A. Liu, H.-C. Chao, E. Bertino (Eds.), Cloud Computing and Security, Springer International Publishing, Cham, 2016, pp. 25–36.
[170] Y. Zhou, S. Liu, M. Xiao, S. Deng, X. Wang, Spatially clustered autonomous vehicle malware: producing new urban geographies of inequity, Mob. Inf. Syst. (2018) 2018.
[171] E.W. Vassallo, K. Manaugh, Spatially clustered autonomous vehicle malware: producing new urban geographies of inequity, Transp. Res. Rec. 2672 (2018) 66–75.
[172] R. van der Heijden, Security Architectures in V2V and V2I Communication, Ph.D. thesis, 2010.
[173] X. Ge, Ultra-reliable low-latency communications in autonomous vehicular networks, IEEE Trans. Veh. Technol. 68 (2019) 5005–5016.
[174] R. Gu, S. Zhang, Y. Ji, Z. Yan, Network slicing and efficient onu migration for reliable communications in converged vehicular and fixed access network, Veh. Commun. 11 (2018) 57–67.
[175] M. Chahal, S. Harit, K.K. Mishra, A.K. Sangaiah, Z. Zheng, A survey on software-defined networking in vehicular ad hoc networks: challenges, applications and use cases, Sustain. Cities Soc. 35 (2017) 830–840.
[176] W.M. Eddy, At what layer does mobility belong?, IEEE Commun. Mag. 42 (2004) 155–159.
[177] S. Choi, G.-H. Hwang, T. Kwon, A.-R. Lim, D.-H. Cho, Fast handover scheme for real-time downlink services in ieee 802.16 e bwa system, in: 2005 IEEE 61st Vehicular Technology Conference, vol. 3, IEEE, 2005, pp. 2028–2032.
[178] L. Dimopoulou, G. Leoleis, I. Venieris, Fast handover support in a wlan environment: challenges and perspectives, IEEE Netw. 19 (2005) 14–20.
[179] R. Koodli, Fast Handovers for Mobile ipv6, 2005.
[180] S.J. Koh, M.J. Chang, M. Lee, MSCTP for soft handover in transport layer, IEEE Commun. Lett. 8 (2004) 189–191.
[181] H. Cheng, J. Cao, H.-H. Chen, H. Zhang, Grls: group-based location service in mobile ad hoc networks, IEEE Trans. Veh. Technol. 57 (2008) 3693–3707.
[182] H. Kim, Y. Kim, An early binding fast handover for high-speed mobile nodes on mipv6 over connectionless packet radio link, in: Seventh ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing, SNPD'06, IEEE, 2006, pp. 237–242.
[183] I. Vivaldi, M.H. Habaebi, B.M. Ali, V. Prakesh, Fast handover algorithm for hierarchical mobile ipv6 macro-mobility management, in: 9th Asia-Pacific Conference on Communications (IEEE Cat. No. 03EX732), vol. 2, IEEE, 2003, pp. 630–634.
[184] G. Koo, K. Yu, M. Noh, Y. Mun, Improved fast handover protocol using hmipv6 based on ieee 802.16 e network, in: International Conference on Computational Science and Its Applications, Springer, 2007, pp. 415–423.
[185] B. Campbell, J. Rosenberg, H. Schulzrinne, C. Huitema, D. Gurle, Session Initiation Protocol (sip) Extension for Instant Messaging, 2002.
[186] V. Devarapalli, R. Wakikawa, A. Petrescu, P. Thubert, Network Mobility (nemo) Basic Support Protocol, 2005.
[187] C.-M. Huang, C.-H. Lee, J.-R. Zheng, A novel sip-based route optimization for network mobility, IEEE J. Sel. Areas Commun. 24 (2006) 1682–1691.
[188] N. Cheng, F. Lyu, J. Chen, W. Xu, H. Zhou, S. Zhang, X.S. Shen, Big data driven vehicular networks, IEEE Netw. (2018).
[189] C. Xu, Z. Zhou, Vehicular content delivery: a big data perspective, IEEE Wirel. Commun. 25 (2018) 90–97.
[190] T. Limbasiya, D. Das, Secure Smart Vehicle Cloud Computing System for Smart Cities, Springer International Publishing, Cham, 2018, pp. 395–415.
[191] M.K. Sharma, A. Kaur, A survey on vehicular cloud computing and its security, in: 1st International Conference on Next Generation Computing Technologies, 2015, pp. 67–71.
[192] C. Cooper, D. Franklin, M. Ros, F. Safaei, M. Abolhasan, A comparative survey of VANET clustering techniques, IEEE Commun. Surv. Tutor. 19 (2017) 657–681.
[193] F. Yang, S. Zou, Y. Tang, X. Du, A multi-channel cooperative clustering-based MAC protocol for V2V communications, Wirel. Commun. Mob. Comput. 16 (2016) 3295–3306.
[194] S.M. Almheiri, H.S. Alqamzi, MANETs and VANETs clustering algorithms: a survey, in: IEEE 8th GCC Conference and Exhibition, 2015.
[195] S. Oubabas, R. Aoudjit, J.J. Rodrigues, S. Talbi, Secure and stable vehicular ad hoc network clustering algorithm based on hybrid mobility similarities and trust management scheme, Veh. Commun. 13 (2018) 128–138.
[196] X. Cheng, B. Huang, A center-based secure and stable clustering algorithm for vanets on highways, Wirel. Commun. Mob. Comput. (2019) 2019.
[197] A. Mahmood, B. Butler, W.E. Zhang, Q.Z. Sheng, S.A. Siddiqui, A hybrid trust management heuristic for vanets, in: 2019 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops), 2019, pp. 748–752.
[198] A. Daeinabi, A.G.P. Rahbar, A. Khademzadeh, VWCA: an efficient clustering algorithm in vehicular ad hoc networks, J. Netw. Comput. Appl. 34 (2011) 207–222.
[199] T. Gazdar, A. Benslimane, A. Belghith, Secure clustering scheme based keys management in vanets, in: 2011 IEEE 73rd Vehicular Technology Conference, VTC Spring, 2011, pp. 1–5.
[200] U. Lee, B. Zhou, M. Gerla, E. Magistretti, P. Bellavista, A. Corradi, Mobeyes: smart mobs for urban monitoring with a vehicular sensor network, IEEE Wirel. Commun. 13 (2006) 52–57.
[201] U. Lee, E. Magistretti, M. Gerla, P. Bellavista, A. Corradi, Dissemination and harvesting of urban data using vehicular sensing platforms, IEEE Trans. Veh. Technol. 58 (2008) 882–901.
[202] U. Lee, E. Magistretti, M. Gerla, P. Bellavista, P. Lió, K.-W. Lee, Bio-inspired multi-agent data harvesting in a proactive urban monitoring environment, Ad Hoc Netw. 7 (2009) 725–741.
[203] U. Lee, J. Lee, J.-S. Park, M. Gerla, Fleanet: a virtual market place on vehicular networks, IEEE Trans. Veh. Technol. 59 (2009) 344–355.
[204] M.D. Dikaiakos, S. Iqbal, T. Nadeem, L. Iftode, VITP: an information transfer protocol for vehicular computing, in: Proceedings of the 2nd ACM International Workshop on Vehicular Ad Hoc Networks, ACM, 2005, pp. 30–39.
[205] J.H. Ahnn, U. Lee, H.J. Moon, M. Gerlag, Senster: scalable smartphone based vehicular sensor networking systems, 2009.
[206] D. Jiang, V. Taliwal, A. Meier, W. Holfelder, R. Herrtwich, Design of 5.9 ghz dsrc-based vehicular safety communication, IEEE Wirel. Commun. 13 (2006) 36–43.
[207] A.S. Mihăiță, P. Tyler, A. Menon, T. Wen, Y. Ou, C. Cai, F. Chen, An Investigation of Positioning Accuracy Transmitted by Connected Heavy Vehicles Using DSRC, Technical Report, 2017.
[208] T.P. Vuong, Cyber-physical intrusion detection for robotic vehicles, Ph.D. thesis, University of Greenwich, 2017.
[209] S. Lamichhane, Penetration Testing in Wireless Networks, 2017.
[210] M. Singh, D. Singh, A. Jara, Secure cloud networks for connected & automated vehicles, in: 2015 International Conference on Connected Vehicles and Expo (ICCVE), IEEE, 2015, pp. 330–335.
[211] L. Yu, J. Deng, R.R. Brooks, S.B. Yun, Automobile ecu design to avoid data tampering, in: Proceedings of the 10th Annual Cyber and Information Security Research Conference, ACM, 2015, p. 10.
[212] W. Fu, X. Xin, P. Guo, Z. Zhou, A practical intrusion detection system for Internet of vehicles, China Commun. 13 (2016) 263–275.

[213] Y. Li, Q. Luo, J. Liu, H. Guo, N. Kato, Tsp security in intelligent and connected vehicles: challenges and solutions, IEEE Wirel. Commun. (2019).
[214] K. Strandberg, T. Olovsson, E. Jonsson, Securing the connected car: a security-enhancement methodology, IEEE Veh. Technol. Mag. 13 (2018) 56–65.
[215] M. Ficco, M. Choraś, R. Kozik, Simulation platform for cyber-security and vulnerability analysis of critical infrastructures, J. Comput. Sci. 22 (2017) 179–186.
[216] M.S.U. Alam, S. Iqbal, M. Zulkernine, C. Liem, Securing vehicle ecu communications and stored data, in: ICC 2019-2019 IEEE International Conference on Communications (ICC), IEEE, 2019, pp. 1–6.
[217] J. Pereira, L. Ricardo, M. Luís, C. Senna, S. Sargento, Assessing the reliability of fog computing for smart mobility applications in vanets, Future Gener. Comput. Syst. 94 (2019) 317–332.
[218] S.-M. Chung, H.-W. Jin, Isolating system faults on vehicular network gateways using virtualization, in: 2010 IEEE/IFIP International Conference on Embedded and Ubiquitous Computing, IEEE, 2010, pp. 791–796.
[219] A. Rajabi Tari, Performance Evaluation of Vehicular Ad Hoc Networks Using Simulation Tools, 2010.
[220] Y.P. Fallah, C. Huang, R. Sengupta, H. Krishnan, Congestion control based on channel occupancy in vehicular broadcast networks, in: 2010 IEEE 72nd Vehicular Technology Conference-Fall, IEEE, 2010, pp. 1–5.
[221] R. Fernandes, P.M. d'Orey, M. Ferreira, Divert for realistic simulation of heterogeneous vehicular networks, in: The 7th IEEE International Conference on Mobile Ad-Hoc and Sensor Systems (IEEE MASS 2010), IEEE, 2010, pp. 721–726.
[222] R. Mangharam, D.S. Weller, D.D. Stancil, R. Rajkumar, J.S. Parikh, Groovesim: a topography-accurate simulator for geographic routing in vehicular networks, in: Proceedings of the 2nd ACM International Workshop on Vehicular Ad Hoc Networks, ACM, 2005, pp. 59–68.
[223] N. Wisitpongphan, O.K. Tonguz, J.S. Parikh, P. Mudalige, F. Bai, V. Sadekar, Broadcast storm mitigation techniques in vehicular ad hoc networks, IEEE Wirel. Commun. 14 (2007) 84–94.
[224] V.D. Khairnar, S. Pradhan, Comparative study of simulation for vehicular ad-hoc network, preprint, arXiv:1304.5181, 2013.
[225] H.-M. Zimmermann, I. Gruber, C. Roman, A Voronoi-based mobility model for urban environments, in: 11th European Wireless Conference 2005-Next Generation Wireless and Mobile Communications and Services, VDE, 2005, pp. 1–5.
[226] J. Harri, M. Fiore, Vanetmobisim–vehicular ad hoc network mobility extension to the canumobisim framework, Institut Eurécom Department of Mobile Commu 6904 (2006) 1–19.
[227] J. Härri, F. Filali, C. Bonnet, M. Fiore, Vanetmobisim: generating realistic mobility patterns for vanets, in: Proceedings of the 3rd International Workshop on Vehicular Ad Hoc Networks, ACM, 2006, pp. 96–97.
[228] S. Jaap, M. Bechler, L. Wolf, Evaluation of routing protocols for vehicular ad hoc networks in typical road traffic scenarios, 2005, pp. 584–602.
[229] S. Bohacek, V. Sridhara, J. Kim, Udel Models for Simulating Urban Wireless Networks, 2000.
[230] U. Models, Udel Models for Simulation of Urban Mobile Wireless Networks, 2007.
[231] L. Bononi, M. Di Felice, M. Bertini, E. Croci, Parallel and distributed simulation of wireless vehicular ad hoc networks, in: Proceedings of the 9th ACM International Symposium on Modeling Analysis and Simulation of Wireless and Mobile Systems, ACM, 2006, pp. 28–35.
[232] S.-Y. Wang, C.-L. Chou, Nctuns Simulator for Wireless Vehicular Ad Hoc Network Research, Ad Hoc Networks: New Research, Nova Science Publishers, 2009.
[233] B.K. Chaurasia, R.S. Tomar, S. Verma, G.S. Tomar, Suitability of manet routing protocols for vehicular ad hoc networks, in: 2012 International Conference on Communication Systems and Network Technologies, IEEE, 2012, pp. 334–338.
[234] T. Jeyaprakash, R. Mukesh, A survey of mobility models of vehicular adhoc networks and simulators, Int. J. Electron. Inf. Eng. 2 (2015) 94–101.
[235] T. Mahmood, M.E.M. Ali, A. Durdu, A two stage fuzzy logic adaptive traffic signal control for an isolated intersection based on real data using sumo simulator, 2019.
[236] J. Mena-Oreja, J. Gozalvez, Permit-a sumo simulator for platooning maneuvers in mixed traffic scenarios, in: 2018 21st International Conference on Intelligent Transportation Systems (ITSC), IEEE, 2018, pp. 3445–3450.
[237] F.K. Karnadi, Z.H. Mo, K.-c. Lan, Rapid generation of realistic mobility models for vanet, in: 2007 IEEE Wireless Communications and Networking Conference, IEEE, 2007, pp. 2506–2511.
[238] M. Piorkowski, M. Raya, A.L. Lugo, P. Papadimitratos, M. Grossglauser, J.-P. Hubaux, TRANS: realistic joint traffic and network simulator for vanets, Mob. Comput. Commun. Rev. 12 (2008) 31–33.
[239] T. Fujiki, M. Kirimura, T. Umedu, T. Higashino, Efficient acquisition of local traffic information using inter-vehicle communication with queries, in: 2007 IEEE Intelligent Transportation Systems Conference, IEEE, 2007, pp. 241–246.
[240] C. Schroth, F. Dötzer, T. Kosch, B. Ostermaier, M. Strassberger, Simulating the traffic effects of vehicle-to-vehicle messaging systems, in: Proceedings of the 5th International Conference on ITS Telecommunications, 2005, p. 4, Citeseer.
[241] J. Miller, E. Horowitz, Freesim-a free real-time freeway traffic simulator, in: 2007 IEEE Intelligent Transportation Systems Conference, IEEE, 2007, pp. 18–23.
[242] J. Magtoto, A. Roque, Real-time traffic data collection and dissemination from an Android smartphone using proportional computation and freesim as a practical transportation system in metro Manila, in: TENCON 2012 IEEE Region 10 Conference, IEEE, 2012, pp. 1–5.
[243] M. Fellendorf, P. Vortisch, Microscopic traffic flow simulator VISSIM, in: Fundamentals of Traffic Simulation, Springer, 2010, pp. 63–93.
[244] X. Xu, H. Liu, J.M. Anderson, Y. Xu, M.P. Hunter, M.O. Rodgers, R.L. Guensler, Estimating project-level vehicle emissions with VISSIM and moves-matrix, Transp. Res. Rec. 2570 (2016) 107–117.
[245] R.F. Benekohal, J. Treiterer, Carsim: car-following model for simulation of traffic in normal and stop-and-go conditions, Transp. Res. Rec. 1194 (1988) 99–111.
[246] R. Johansson, D. Williams, A. Berglund, P. Nugues, Carsim: a system to visualize written road accident reports as animated 3d scenes, in: Proceedings of the 2nd Workshop on Text Meaning and Interpretation, 2004, pp. 57–64.
[247] T. Osafune, L. Lin, M. Lenardi, Multi-hop vehicular broadcast (MHVB), in: 2006 6th International Conference on ITS Telecommunications, IEEE, 2006, pp. 757–760.
[248] K. Pandey, S.K. Raina, R.S. Rao, Performance analysis of routing protocols for vehicular adhoc networks using ns2/sumo, in: 2015 IEEE International Advance Computing Conference (IACC), IEEE, 2015, pp. 844–848.
[249] W. Liu, X. Wang, W. Zhang, L. Yang, C. Peng, Coordinative simulation with sumo and ns3 for vehicular ad hoc networks, in: 2016 22nd Asia-Pacific Conference on Communications (APCC), IEEE, 2016, pp. 337–341.
[250] G. Sallam, A. Mahmoud, Performance evaluation of olsr and aodv in vanet cloud computing using fading model with sumo and ns3, in: 2015 International Conference on Cloud Computing (ICCC), IEEE, 2015, pp. 1–5.
[251] M. Báguena, S.M. Tornell, Á. Torres, C.T. Calafate, J.-C. Cano, P. Manzoni, Vacamobil: vanet car mobility manager for omnet++, in: 2013 IEEE International Conference on Communications Workshops (ICC), IEEE, 2013, pp. 1057–1061.
[252] M. Pasha, M.U. Farooq, et al., A proof-of-concept model for vehicular cloud computing using omnet++ and sumo, in: Innovations in Computer Science and Engineering, Springer, 2016, pp. 193–198.
[253] B. Sliwa, J. Pillmann, F. Eckermann, C. Wietfeld, Limosim: a lightweight and integrated approach for simulating vehicular mobility with omnet++, preprint, arXiv:1709.02020, 2017.
[254] R. Nagel, S. Eichler, Efficient and realistic mobility and channel modeling for vanet scenarios using omnet++ and inet-framework, in: Proceedings of the 1st International Conference on Simulation Tools and Techniques for Communications, Networks and Systems & Workshops, ICST (Institute for Computer Sciences, Social-Informatics and …), 2008, p. 89.
[255] S.A. Hussain, A. Saeed, An analysis of simulators for vehicular ad hoc networks, World Appl. Sci. J. 23 (2013) 1044–1048.
[256] S. Al-Sultan, M.M. Al-Doori, A.H. Al-Bayatti, H. Zedan, A comprehensive survey on vehicular ad hoc network, J. Netw. Comput. Appl. 37 (2014) 380–392.
[257] P. Costa, D. Frey, M. Migliavacca, L. Mottola, Towards lightweight information dissemination in inter-vehicular networks, in: Proceedings of the 3rd International Workshop on Vehicular Ad Hoc Networks, ACM, 2006, pp. 20–29.
[258] J. Yin, T. ElBatt, G. Yeung, B. Ryu, S. Habermas, H. Krishnan, T. Talty, Performance evaluation of safety applications over dsrc vehicular ad hoc networks, in: Proceedings of the 1st ACM International Workshop on Vehicular Ad Hoc Networks, ACM, 2004, pp. 1–9.
[259] S. Gräfling, P. Mähönen, J. Riihijärvi, Performance evaluation of ieee 1609 wave and ieee 802.11 p for vehicular communications, in: 2010 Second International Conference on Ubiquitous and Future Networks (ICUFN), IEEE, 2010, pp. 344–348.
[260] D. Popescu, P. Jacquet, B. Mans, R. Dumitru, A. Pastrav, E. Puschita, Information dissemination speed in delay tolerant urban vehicular networks in a hyperfractal setting, IEEE/ACM Trans. Netw. (2019).
[261] S. Cai, B. Gallina, D. Nystrom, C. Seceleanu, Customized real-time data management for automotive systems: a case study, in: 43rd Annual Conference of the IEEE, Industrial Electronics Society, 2017, pp. 8397–8404.
[262] C. Miller, C. Valasek, Remote Exploitation of an Unaltered Passenger Vehicle, Technical Report, 2015.
[263] C. Bordonali, S. Ferraresi, W. Richter, Shifting Gears in Cyber Security for Connected Cars, 2017.
[264] L. Liang, H. Ye, G.Y. Li, Toward intelligent vehicular networks: a machine learning framework, IEEE Int. Things J. 6 (2019) 124–135.
[265] H. Ye, L. Liang, G.Y. Li, J. Kim, L. Lu, M. Wu, Machine learning for vehicular networks: recent advances and application examples, IEEE Veh. Technol. Mag. 13 (2018) 94–101.
[266] L. Zhang, N. Kaja, L. Shi, A two-stage deep learning approach for can intrusion detection system, in: Vehicle Electronics and Architecture (VEA) & Ground Systems Cyber Engineering (GSCE) Technical Session, Novi, Michigan, 2018.
[267] M. Yousefi-Azar, V. Varadharajan, L. Hamey, U. Tupakula, Autoencoder-based feature learning for cyber security applications, in: 2017 International Joint Conference on Neural Networks (IJCNN), IEEE, 2017, pp. 3854–3861, http://ieeexplore.ieee.org/document/7966342/.

[268] S.E. Chandy, A. Rasekh, Z.A. Barker, M.E. Shafiee, Cyberattack detection using deep generative models with variational inference, Computing Research Repository (2018).
[269] O. Vinyals, I. Babuschkin, J. Chung, M. Mathieu, M. Jaderberg, W.M. Czarnecki, A. Dudzik, A. Huang, P. Georgiev, R. Powell, T. Ewalds, D. Horgan, M. Kroiss, I. Danihelka, J. Agapiou, J. Oh, V. Dalibard, D. Choi, L. Sifre, Y. Sulsky, S. Vezhnevets, J. Molloy, T. Cai, D. Budden, T. Paine, C. Gulcehre, Z. Wang, T. Pfaff, T. Pohlen, Y. Wu, D. Yogatama, J. Cohen, K. McKinney, O. Smith, T. Schaul, T. Lillicrap, C. Apps, K. Kavukcuoglu, D. Hassabis, D. Silver, AlphaStar: mastering the real-time strategy game StarCraft II, https://deepmind.com/blog/alphastar-mastering-real-time-strategy-game-starcraft-ii/, 2019.
[270] R. Elderman, L.J.J. Pater, A.S. Thie, M.M. Drugan, M.M. Wiering, Adversarial reinforcement learning in a cyber security simulation, in: 9th International Conference on Agents and Artificial Intelligence, 2017, pp. 559–566.
[271] H. Ye, G.Y. Li, Deep reinforcement learning for resource allocation in V2V communications, in: 2018 IEEE International Conference on Communications (ICC), IEEE, 2018, pp. 1–6, https://ieeexplore.ieee.org/document/8422586/.
[272] C. Yin, Y. Zhu, J. Fei, X. He, A deep learning approach for intrusion detection using recurrent neural networks, IEEE Access 5 (2017) 21954–21961.
[273] A. Bezemskij, G. Loukas, D. Gan, R.J. Anthony, Detecting cyber-physical threats in an autonomous robotic vehicle using Bayesian networks, in: 2017 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), IEEE, 2017, pp. 98–103, http://ieeexplore.ieee.org/document/8276737/.
[274] M.-J. Kang, J.-W. Kang, A novel intrusion detection method using deep neural network for in-vehicle network security, in: 2016 IEEE 83rd Vehicular Technology Conference (VTC Spring), IEEE, 2016, pp. 1–5, http://ieeexplore.ieee.org/document/7504089/.
[275] A. Taylor, S. Leblanc, N. Japkowicz, Anomaly detection in automobile control network data with long short-term memory networks, in: 2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA), IEEE, 2016, pp. 130–139, http://ieeexplore.ieee.org/document/7796898/.
[276] M. Kalash, M. Rochan, N. Mohammed, N.D.B. Bruce, Y. Wang, F. Iqbal, Malware classification with deep convolutional neural networks, in: 2018 9th IFIP International Conference on New Technologies, Mobility and Security (NTMS), IEEE, 2018, pp. 1–5, http://ieeexplore.ieee.org/document/8328749/.
[277] R. Pascanu, J.W. Stokes, H. Sanossian, M. Marinescu, A. Thomas, Malware classification with recurrent networks, in: 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2015, pp. 1916–1920, http://ieeexplore.ieee.org/document/7178304/.
[278] Z. Lin, Y. Shi, Z. Xue, IDSGAN: generative adversarial networks for attack generation against intrusion detection, arXiv, 2018.
[279] C.R. Sweet, Synthesizing Cyber Intrusion Alerts using Generative Adversarial Networks, Ph.D. thesis, Rochester Institute of Technology, 2019.
[280] C.-K. Wen, S. Jin, K.-K. Wong, J.-C. Chen, P. Ting, Channel estimation for massive MIMO using Gaussian-mixture bayesian learning, IEEE Trans. Wirel. Commun. 14 (2015) 1356–1368.
[281] H. Ye, G.Y. Li, B.-H. Juang, Power of deep learning for channel estimation and signal detection in OFDM systems, IEEE Wirel. Commun. Lett. 7 (2018) 114–117.
[282] E. Eziama, K. Tepe, A. Balador, K.S. Nwizege, L.M.S. Jaimes, Malicious node detection in vehicular ad-hoc network using machine learning and deep learning, in: 2018 IEEE Globecom Workshops (GC Wkshps), IEEE, 2018, pp. 1–6, https://ieeexplore.ieee.org/document/8644127/.
[283] N. Taherkhani, S. Pierre, Centralized and localized data congestion control strategy for vehicular ad hoc networks using a machine learning clustering algorithm, IEEE Trans. Intell. Transp. Syst. 17 (2016) 3275–3285.
[284] P.-R. Chen, H.-M. Hang, S.-W. Chan, J.-J. Lin, DSNet: an Efficient CNN for Road Scene Segmentation, 2019, pp. 1–9.
[285] Y. Lyu, L. Bai, X. Huang, Road segmentation using CNN and distributed LSTM, in: 2019 IEEE International Symposium on Circuits and Systems (ISCAS), IEEE, 2019, pp. 1–5, https://ieeexplore.ieee.org/document/8702174/.
[286] J. Wiest, M. Hoffken, U. Kresel, K. Dietmayer, Probabilistic trajectory prediction with Gaussian mixture models, in: 2012 IEEE Intelligent Vehicles Symposium, IEEE, 2012, pp. 141–146, http://ieeexplore.ieee.org/document/6232277/.
[287] C. Ide, F. Hadiji, L. Habel, A. Molina, T. Zaksek, M. Schreckenberg, K. Kersting, C. Wietfeld, LTE connectivity and vehicular traffic prediction based on machine learning approaches, in: 2015 IEEE 82nd Vehicular Technology Conference (VTC2015-Fall), IEEE, 2015, pp. 1–5, http://ieeexplore.ieee.org/document/7391019/.
[288] A. Khan, A. Sohail, U. Zahoora, A.S. Qureshi, A Survey of the Recent Architectures of Deep Convolutional Neural Networks, 2019, pp. 1–67.
[289] F. Altche, A. de La Fortelle, An LSTM network for highway trajectory prediction, in: 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC), IEEE, 2017, pp. 353–359, http://ieeexplore.ieee.org/document/8317913/.
[290] Y. Lv, Y. Duan, W. Kang, Z. Li, F.-Y. Wang, Traffic flow prediction with big data: a deep learning approach, IEEE Trans. Intell. Transp. Syst. (2014) 1–9.
[291] N. Akhtar, A. Mian, Threat of adversarial attacks on deep learning in computer vision: a survey, IEEE Access 6 (2018) 14410–14430.
[292] A.R., E.S., E. Meiri, E. Fomenko, Y.J.K., H.S., B.Z., Lower numerical precision deep learning inference and training, 2018.
[293] M. Zhu, S. Gupta, To Prune, or Not to Prune: Exploring the Efficacy of Pruning for Model Compression, 2017.
[294] D.C.P.G. Saranti, S. Karatzas, Autonomous vehicles and blockchain technology are shaping the future of transportation, in: Conference on Sustainable Urban Mobility, 2018.
[295] R. Zhang, R. Xue, L. Liu, Security and privacy on blockchain, preprint, arXiv: 1903.07602, 2019.
[296] D. Yaga, P. Mell, N. Roby, K. Scarfone, Blockchain Technology Overview, NIST Interagency/Internal Report, 2018, p. 57.
[297] R. Shrestha, R. Bajracharya, A.P. Shrestha, S.Y. Nam, A new-type of blockchain for secure message exchange in vanet, Digital Commun. Netw. (2019).
[298] K. Wüst, A. Gervais, Do you need a blockchain?, in: 2018 Crypto Valley Conference on Blockchain Technology (CVCBT), IEEE, 2018, pp. 45–54.
[299] M.S.U. Alam, Securing Vehicle Electronic Control Unit (ECU) Communications and Stored Data, Ph.D. thesis, Queen's University, Canada, 2018.
[300] A. Lei, H. Cruickshank, Y. Cao, P. Asuquo, C.P.A. Ogah, Z. Sun, Blockchain-based dynamic key management for heterogeneous intelligent transportation systems, IEEE Int. Things J. 4 (2017) 1832–1843.