electronics

MDPI

Review

Demystifying In-Vehicle Intrusion Detection Systems: A Survey of Surveys and a Meta-Taxonomy

Georgios Karopoulos [1,†], Georgios Kambourakis [1,*,†], Efstratios Chatzoglou [2,†], José L. Hernández-Ramos [1,†] and Vasileios Kouliaridis [2,†]

[1] European Union, Joint Research Centre, 21027 Ispra, Italy; georgios.karopoulos@ec.europa.eu (G.K.); jose-luis.hernandez-ramos@ec.europa.eu (J.L.H.-R.)
[2] Department of Information & Communication Systems Engineering, University of the Aegean, 83200 Karlovasi, Samos, Greece; efchatzoglou@gmail.com (E.C.); bkouliaridis@aegean.gr (V.K.)
* Correspondence: georgios.kampourakis@ec.europa.eu
† These authors contributed equally to this work.

Abstract: Breaches in the cyberspace due to cyber-physical attacks can harm the physical space, and any type of vehicle is an alluring target for wrongdoers for an assortment of reasons. Especially, as the automobiles are becoming increasingly interconnected within the Cooperative Intelligent Transport System (C-ITS) realm and their level of automation elevates, the risk for cyberattacks augments along with the attack surface, thus inexorably rendering the risk of complacency and inaction sizable. Next to other defensive measures, intrusion detection systems (IDS) already comprise an inextricable component of modern automobiles in charge of detecting intrusions in the system while in operation. This work concentrates on in-vehicle IDS with the goal to deliver a fourfold comprehensive survey of surveys on this topic. First, we collect and analyze all existing in-vehicle IDS classifications and fuse them into a simpler, overarching one that can be used as a base for classifying any work in this area. Second, we gather and elaborate on the so-far available datasets which can be possibly used to train and evaluate an in-vehicle IDS. Third, we survey non-commercial simulators which may be utilized for creating a dataset or evaluating an IDS. The last contribution pertains to a thorough exposition of the future trends and challenges in this area. To our knowledge, this work provides the first wholemeal survey on in-vehicle IDS, and it is therefore anticipated to serve as a groundwork and point of reference for multiple stakeholders at varying levels.

Keywords: vehicle intrusion detection system; intra-vehicle network; CAN bus; taxonomy

Citation: Karopoulos, G.; Kambourakis, G.; Chatzoglou, E.; Hernández-Ramos, J.L.; Kouliaridis, V. Demystifying In-Vehicle Intrusion Detection Systems: A Survey of Surveys and a Meta-Taxonomy. *Electronics* **2022**, *11*, 1072. https://doi.org/10.3390/electronics11071072

Academic Editor: Vijayakumar Varadarajan

Received: 8 March 2022
Accepted: 26 March 2022
Published: 29 March 2022

1. Introduction

In recent years, the technological development of the automotive industry is promoting the manufacturing of increasingly connected vehicles, allowing interaction with other vehicles and components on the road, the so-called vehicle-to-everything (V2X) communications [1]. A key aspect in this trend is the incorporation and integration of a large number of electronic components, including sensors, actuators, and electronic control units (ECUs), to provide specific functions within the vehicle, such as power train, chassis, and body systems. These components are grouped forming subnets, which communicate through gateways using different protocols composing an in-vehicle dense network. While an increasing number of electronic components in vehicles—modern vehicles are composed of 70 to 100 ECU connecting to the in-vehicle network (IVN)—is essential for the development of future autonomous vehicles, this trend also brings along a much larger attack surface that could ultimately affect passengers' safety.

Among the different intra-vehicular network protocols, including FlexRay [2], Local Interconnect Network (LIN) [3], or Media Oriented Systems Transport (MOST) [4]), currently the Controller Area Network (CAN) protocol represents the prevailing standard due to its low cost and fault tolerance properties [5]. However, as often pinpointed in the

literature [6–8], CAN suffers from the lack of basic security services, including authentication and data encryption, and presents a vulnerable arbitration mechanism. These security shortages have motivated the need to develop security techniques to identify potential attacks as well as to mitigate their impact.

Particularly, the development and deployment of intrusion detection systems (IDS) in the vehicular context have aroused a significant interest in recent years [9]. IDS approaches are widely used in information communication technologies (ICT) to monitor and analyze network traffic and/or local activity, so that attacks or misuse can be detected. This analysis identifies anomalous patterns when potentially suspicious activity occurs, revealing violations of the established security policy (such as the transmission of unusually large amounts of data). However, the use of IDS for IVN must consider any requirement applicable to the particular context, including the real-time reaction times and resource constraints. On the other hand, a vehicular IDS (VIDS) can detect assaults by just monitoring the IVN traffic, and this is a significant plus, vis-à-vis other defense approaches such as ECU authentication [10] and hardware-enforced isolation [11]. Simply stated, opposite to other types of defenses, a VIDS does not alter the existing IVN architecture, does not produce extra IVN traffic, and does not mandate any changes to the underlying bus protocol. Thus far, several works have analyzed the use of VIDS, proposing diverse categories to classify these approaches [6,12,13]. Nevertheless, the lack of a unified, plain taxonomy hinders the analysis of existing VIDS proposals, as well as the identification of new research opportunities addressing cybersecurity issues in IVN. Moreover, none of the existing surveys on the topic cover the large volume of related wok in the last couple of years, whereas information on datasets and simulators to support IVN IDS research is rather scattered.

Our contribution: To address this literature gap, the work at hand examines this topic using a multi-fold, holistic approach. Overall, differently or supplementary to the related work, the following key contributions are identified.

- We compile a new meta-taxonomy that groups the main VIDS classification features proposed in existing surveys. The main purpose here is to offer a unified picture of the main aspects related to the development of VIDS for aiding researchers in obtaining a clear overview of the existing landscape of solutions and to easily classify the development of new approaches for IVN. Naturally, this axis of contribution provides a solid and positive answer to the following question: Is it possible to fuse the numerous current (and possibly future) INV IDS approaches to a simpler one that consolidates every aspect of all the proposed schemes?
- Based on the created meta-taxonomy, we provide an updated analysis of VIDS proposals by classifying the ones proposed in the period 2020–2022.
- VIDS goes hand in hand with (a) datasets used to train and test a VIDS and (b) simulators that may be used for either creating datasets or evaluating the performance of a VIDS. In this respect, as a side contribution, this work offers an all-encompassing survey of both the previous aspects, which, as previously mentioned, are regarded as highly complementary to VIDS.
- Last, but not least, based on the analysis of the previous work in the field, the current work contributes a holistic, contemporary view of the main challenges and future trends in this rapidly evolving and interesting research branch.

The rest of this work is structured as follows. The next section details the related work focusing on the major VIDS surveys contributed in the literature so far. Building on the results of the preceding analysis, Section 3 conflates the various VIDS individual categorizations into an overarching simpler one, which is then used to classify recent VIDS approaches. Section 4 elaborates on the available datasets for vehicular networks, while Section 5 provides a discussion on the publicly available vehicular network simulation tools. The identified challenges and future trends are given in Sections 6 and 7, respectively. The last section concludes.

2. Related Work

Recently, the research of in-vehicle security has seen increased attention. This section reviews recent surveys on VIDS. Specifically, we provide a summary of twelve recent surveys on intra-vehicle IDS and filter out key points per work, such as categorization of VIDS approaches, feature extraction, employed datasets, attack types, performance and evaluation, and research gaps. To ease comparison, Table 1 offers an overview of the surveyed works and their contribution.

Table 1. Summary of previous survey works on the field of IVN IDS sorted by year in descending order. ✓: provided, ✗: not provided.

Survey	Year	No of Works	Intra-Vehicle Protocol	Categorization of Works	Performance Comparison	Research Gaps
[14]	2021	30	CAN	✓	✗	✓
[8]	2021	23	CAN	✓	✗	✓
[15]	2020	5	CAN	✓	✓	✗
[13]	2020	20	CAN	✓	✓	✓
[5]	2019	42	LIN, CAN, FlexRay, Ethernet, MOST	✓	✓	✓
[6]	2019	15	CAN	✓	✗	✗
[7]	2019	25	CAN	✓	✗	✓
[16]	2019	24	CAN	✓	✗	✗
[17]	2019	6	CAN	✓	✗	✗
[18]	2018	19	CAN	✓	✓	✓
[9]	2018	9	CAN	✓	✓	✓
[19]	2018	17	CAN	✓	✗	✓

The authors in [14] review cyberattacks and the relevant countermeasures for CAN-based IVN. The paper gives a brief analysis of the main communication protocols (CAN, LIN, and FlexRay), but basically focuses on CAN only. To this end, the authors provide an overview of vulnerabilities, potential entry points for data injection, and attacks against the CAN bus. The countermeasures are categorized into cryptographic and IDS solutions. Regarding the latter, 30 VIDS approaches were surveyed from 2008 to 2020. Furthermore, the research challenges associated with IDS-based approaches are identified and summarized.

The work in [8] provides a survey of cybersecurity of in-vehicle networks. It analyzes vulnerabilities and security requirements in CAN-based IVNs, as well as protection mechanisms. Vulnerabilities pertaining to confidentiality, authenticity, availability, integrity, and non-repudiation are analyzed; IDS systems are referenced as an availability protection measure. The authors review 23 state-of-the-art works on CAN-based VIDS systems and classify them into four categories: physical characteristics-based, timing interval-based, entropy-based, and artificial learning-based.

Hafeez et al. [15] classify in-vehicle IDS in four categories: message parameter-based, information theory-based, machine learning-based, and fingerprinting-based. The first detection method works on the MAC layer and was identified in 11 of the surveyed works. An information theory-based VIDS, discussed in three of the included works, exploits entropy. On the other hand, machine learning and fingerprinting-based approaches were identified in seven and five works, respectively. The authors focused on the latter approach, which operates on the physical layer, and provided a survey of such methods. All the considered fingerprinting-based VIDS approaches were attached to CAN and ECU units and followed physical layer detection techniques, such as variations in clock and energy. It was noted that four out of the five fingerprinting-based IDS approaches achieved high accuracy (>96%). While different advantages and disadvantages of each approach are presented, it has to be noted that three of them require the presence of an additional ECU.

The work in [13] surveys intrusion detection solutions in CAN-based IVN. The authors classify generic IVN countermeasures as follows: (a) encryption- and authentication-based, (b) firewall implementations, and (c) IVN IDSs. Out of these, encryption- and authentication-based solutions are not appropriate due to the resource constraints of IVNs, that is, cost, computational power, bandwidth, and storage capacity. The implementation of firewalls is not a realistic solution as vehicles tend to have a long lifecycle and IVNs a wide attack surface. The authors argue that an IDS applied to IVN is a viable countermeasure that can be applied to such a resource-constrained environment while being backward-compatible. Furthermore, the paper provides a classification of attacks to IVNs based on the network layer model into (a) physical layer, (b) data-link layer, and (c) application layer attacks. They survey 8 works and identify 15 different attacks in total for all categories. Regarding intrusion detection, a taxonomy is proposed from the technology implementation perspective. The authors survey 20 papers and categorize them as (a) fingerprint-based, (b) parameter monitoring-based, (c) information theory-based, and (d) machine learning-based. The comparison of the aforementioned IDS systems led to the following observations. First, there is no single IVN IDS that can detect attacks from different layers; thus, different IDS solutions should be used to cover all layers. Second, while machine learning methods have the advantage of detecting unknown attacks, they require more resources and do not fit well with the automotive environment; to this end, a cloud-based solution has been proposed. Third, most existing VIDS methods show high accuracy, but this accuracy is measured against attacks on a single layer only. This means that these VIDS use features associated with a single layer only; a single VIDS solution covering all layers would benefit from features associated with more than one layers. The paper also provides a discussion of future trends and challenges in the IVN IDS domain.

Al-Jarrah et al. [5] contributed a review of the state-of-the-art intra-vehicle IDS. First, the paper presented an overview of each intra-vehicle network, that is, LIN, CAN, FlexRay, Ethernet, and MOST. The authors compared the aforementioned networks in terms of system cost, bandwidth, protocol efficiency, fault tolerance, MAC mechanism, topology, and security threats. They also categorized the reviewed intra-vehicle IDSs into flow-based, payload-based, and hybrid IDSs, with 19, 17, and 6 works in each category, respectively. Regarding datasets used to evaluate VIDSs, 21 works out of 42 works used real data, 11 out of 42 works used simulated data, and 10 out of 42 did not provide information on the data used. Furthermore, features used by intra-vehicle IDSs were categorized into two types, i.e., physical and cyber features. As such, 2 out of 42 works used physical features to detect attacks, 4 out of 42 works used a combination of cyber and physical features, and 2 out of 42 works did not provide any description of the features used. Regarding attack types, the authors considered the following cyberattacks against intra-vehicle networks: denial of service (DoS), message injection and replay, message manipulation, masquerade, and malware attack. The authors compared each work using various metrics, that is, confusion matrix, detection accuracy, detection rate, false positive, false negative, F-measure, and ROC curve. In addition, the authors took under consideration the following benchmark models for each work: decision trees, ANNs and deep learning, SVM and OCSVM, and random forest. The research challenges presented can be summarized in the following topics: importance of intra-vehicle IDS placement, missing standard benchmark detection model for performance comparison, defining and selecting important features, lack of benchmark datasets, conclusive evaluation metrics, and developing a context-aware IDS.

Young et al. [6] provided an overview of the vulnerabilities and threats in the automotive ecosystem, identified known attacks in CAN, compared VIDS approaches, and discussed advantages and disadvantages of each surveyed work. The authors first explained three major vulnerabilities in CAN, namely, lack of message authentication, unsegmented network, and unencrypted messages. Regarding threats and attacks, the authors detailed known attacks by using either the onboard diagnostics (OBD) port to scan the CAN bus network or remote exploitation techniques. These attacks may allow the attacker to acquire complete control of several functions of the vehicle, such as disabling the brakes or stop-

ping the engine. Additionally, they provided a categorization of IDSs based on detection features, namely, message frequency, message interval, signatures, cyber-physical, entropy, CAN fields, sensor data, and deep neural network. Finally, they compared 15 works in terms of features used, types of detected attacks, and dataset used.

Also focused on IDS approaches for the CAN bus, [7] offered a detailed description of vulnerabilities and potential attacks that can be launched against CAN-based IVN. They propose a taxonomy to categorize VIDS approaches for a CAN bus network considering deployment strategies, detection approaches, attack techniques, and technical challenges. In particular, they analyze 25 approaches and describe a set of challenges derived from the proposed analysis for the definition of VIDS approaches for CAN bus networks. Moreover, 24 IDS approaches for the CAN bus are analyzed by [16] based on the information they extract from the network and the way they build their model. Additionally in the same direction, [17] proposes five criteria to classify CAN-based VIDS approaches: data source, detection method, data analysis location, analysis frequency, and behavior after detection. Besides describing some of the main CAN vulnerabilities, the authors analyze six papers considering such classification.

Loukas et al. [18] presented a classification and survey of in-vehicle IDS. Specifically, they classified the surveyed works based on the target vehicle category, i.e., aircraft, land vehicle, and watercraft, and compared works in the literature for each of these categories. Regarding CAN-based VIDS approaches, the authors compared 19 works in the literature, dated from 2008 to 2017. They used various characteristics to collate the surveyed works, such as the employed architecture, deployment, features, technologies, and evaluation. A similar comparison is also presented for 23 works for VANET. The authors also summarized IVN threats and attacks used for the evaluation of each VIDS approach. Finally, they discussed open issues and presented their conclusions.

The survey in [9] examined 24 relevant works, 9 out of which are directly identified as in-vehicle IDSs; in Table 1 we consider only those intended for IVN networks. The authors approached the topic through three major axes: attacks, VIDS taxonomy, and challenges in IDS deployment. Attacks are classified into insider or outsider, active or passive, and attacks on confidentiality, integrity, authentication, or availability. Regarding a possible taxonomy, the authors classify VIDSs based on reaction type, detection methodology, validation strategy, and deployment location. One of the main challenges in the deployment of IDS is the absence of real-world deployment and testing, which may affect the actual performance and applicability of these VIDSs. Additionally, most of the proposed VIDSs in the literature were utilized in few attacks, not covering a large portion of the attack surface, and these works did not elaborate on the pros and cons of their proposed scheme. Other key factors are related to (a) the absence of publicly available datasets to run experiments, and (b) the deployment location of the VIDS, because it can greatly affect its energy consumption and overall detection effectiveness. The authors concluded that the so-far proposed IDS schemes are unable to identify zero-days and mitigate threats beforehand.

The work in [19] surveyed proposals in the CAN intrusion detection area and considered their adoption implications. The authors gave an overview of CAN protocol and presented the challenges associated with intrusion detection in CAN-based vehicles. They reviewed 17 VIDS solutions from 2012 to 2018 and classified them into signature- and anomaly-based, further dividing the latter into statistical, knowledge-based, and machine learning.

The following are additional review works on IVN security that, although not exclusively IDS-oriented, partially cover the IVN IDS domain and are cited here for the sake of completeness; note that these are not included in Table 1. In [12], security in intelligent connected vehicles is reviewed, covering attacks and defenses on vehicles and vehicular communication networks (both in-vehicle and inter-vehicle). The paper provides a classification of attacks; the categories pertaining to in-vehicle networks are replay, Sybil, and impersonation assaults. There is also a classification of defenses; the categories of defenses related to the in-vehicle attacks listed above are cryptography and network security

(IDS) solutions. In [20], the authors provide an overview of IVN security by summarizing IVN vulnerabilities and attacking methodologies; furthermore, they present a generic attack procedure that outlines the different phases of attacking IVNs. The countermeasures that have been proposed to tackle existing attacks are reviewed and classified into three distinct categories: (a) encryption- and authentication-based, (b) anomaly-detection-based IDSs, and (c) separating the IVN from input interfaces, such as the OBD port. Finally, challenges and future directions are discussed.

A summary of the main characteristics of the related work in IVN IDS is presented in Table 1. Overall, the identified surveys are recent, that is, between 2018 and 2021, with their majority published in 2019 (5 out of 12 works); regarding the surveyed works that each paper includes, the oldest ones are dated back to 2008. However, each survey covers only part of the topic and there is no complete, up-to-date comparison of the related work in IVN IDSs. The vast majority of surveys focuses on CAN as intra-vehicle protocol, whereas all of them offer some kind of taxonomy, although very different to each other. Interestingly, all the surveys recapitulated in Table 1 refer to possible attacks and some additionally provide a taxonomy for attacks, but none rely on some generally accepted threat model such as STRIDE, even though the idea of such an analysis already exists [21]. Moreover, no work elaborates on the currently publicly available datasets that can be used for evaluating the proposed solution, whereas recent standardization efforts are not included and discussed in detail.

3. IDS Taxonomy

3.1. Taxonomies in Related Work

There are several taxonomies that can be used to classify VIDSs; others apply more to IVNs and others to VIDSs in general. This section presents the taxonomies used in the related work analyzed in Section 2. For easy reference, a summary of the different taxonomies used in the literature is depicted in Figure 1.

In [14], the VIDS approaches are classified into signature-based and anomaly-based. A signature-based IDS monitors traffic and compares it with pre-existing databases of attack signatures. While this is an effective mechanism with high accuracy and low error rates, it cannot detect new, unknown, or known but modified attacks, leaving a window of vulnerability open until the signature database is updated. In the anomaly-based approach, the IDS is trained with a model of what is considered normal activity and the detection engine tries to identify deviations from this activity. This type of IDS can detect unknown attacks, but on the other hand it has higher levels of false positives, which are analogous to the completeness and freshness of the training model used. The IDSs that are based on anomaly detection are further subdivided into statistical, machine learning, and physical characteristics-based. A statistical IDS creates a profile of normal system behavior based on statistical relationship analysis of CAN features, such as throughput, response time, number of packets exchanged within a time period, and transmission frequency of a particular CAN ID. A machine-learning IDS detects anomalous behavior using machine learning (ML) algorithms, whereas physical characteristics-based systems work at the physical layer of CAN and use the signals and voltage signatures of ECUs for detection.

The taxonomy utilized in [8] spans four categories. In the first, physical characteristics-based detection is used with fingerprinting methods based on ECU characteristics such as clock offset, voltage distribution, and signal characteristics. The second category is timing interval-based and comprises statistical methods considering that most CAN messages have predictable periodicity. The third category includes entropy-based IDSs that use statistical methods, considering that the format of CAN messages is defined in the design phase. The last category comprises artificial learning-based IDSs.

Figure 1. A map of the IVN IDS taxonomies used in the literature.

The work in [13] proposes the following taxonomy based on the technology implementation perspective: (a) fingerprint-based, (b) parameter-monitoring-based, (c) information-theory-based, and (d) ML-based. A fingerprint-based IDS operates on the bus/physical layer and takes advantage of the unique hardware characteristics of each ECU, such as voltage. By creating a fingerprint profile for known ECUs it is then quite easy to detect illegal nodes. In a similar fashion, parameter-monitoring-based IDSs monitor unique network parameters, operating on the message/network layer; they are further divided into frequency-based, which measure the transmission intervals between messages, and remote frame, which measures the response time of the receiver node. Information theoretic IDSs operate on the data flow level and calculate the information entropy of the exchanged messages in an attempt to detect anomalies. An automotive IDS can detect anomalous behavior by applying ML algorithms on network traffic; according to the algorithm, these methods can be further divided into classification-based, deep learning, and sequential techniques. The same taxonomy, although without the subcategories, is used also in Hafeez et al. [15].

Al-Jarrah et al. [5] categorized IVN IDSs into flow-based, payload-based, and hybrid IDSs. Flow-based IDSs monitor the messages exchanged in the internal network of a vehicle and perform feature extraction (such as message frequency and interval) with the purpose of identifying suspicious behavior. Flow-based IDSs are further subdivided into rule-based, time- and frequency-analysis-based, computational intelligence and information theory, and others/hybrid. A payload-based IDS monitors the payload of the exchanged messages, whereas hybrid combines the two concepts. Payload-based are further subdivided into rule-based, computational intelligence and information theory, and others/hybrid.

Young et al. [6] follows a more traditional approach into IDS taxonomy; they first classify IDSs into host-based (HIDS) and network-based (NIDS). An HIDS is located inside the vehicle and monitors individual ECUs, checking packets entering and leaving, as well as the ECU itself to identify suspicious traffic or behavior. Nevertheless, implementing an HIDS in the ECU is challenging, given that a typical ECU possesses low processing power. An automotive NIDS monitors the network traffic of the IVN and analyzes the header and content of each packet to detect suspicious messages. The second taxonomy examines the detection method and divides IDSs into signature-based and anomaly-based.

The authors in [7] first categorize IDS approaches for CAN bus networks according to where the IDS system is deployed into: gateway-, ECU-, or CAN-based. They also classify them based on the detection approach as signature-based, anomaly-based, or specification-based. The first two approaches have the same meaning as in other taxonomies described in this section. In a specification-based approach, the normal behavior of the system is described manually using a set of thresholds and rules, whereas in an anomaly-based one, an automated training phase precedes detection. IDSs can further be categorized, based on the attacking techniques that are most commonly used to evaluate their sensitivity and effectiveness, into attacks on CAN packet frequency and attacks on CAN packet payload. Finally, IDS systems can be classified according to the technical challenges that are taken into consideration when they are designed as follows: (a) limited resources, when they consider memory, processing, and bandwidth limitations, (b) timing requirement, when considering prioritization of traffic with real-time requirements, (c) traffic patterns behavior, taking into account the broadcast nature of CAN messages, (d) unstable connections, when considering that a vehicle may move to an area with no Internet connection, and (e) size, weight, and cost of the IDS, taking also into account any modifications required before deployment.

The survey in [16] categorizes CAN-based IDSs based on three aspects. The first one is the number of frames required by the IDS to detect the attack. The second aspect concerns the data used for the detection; these data are certain features of CAN frames, such as arbitration ID and time interval between messages. Lastly, the third aspect for classifying IDSs is model-building and pertains to how a model of normal behavior is built when an anomaly-based detection approach is followed; models can be learning- or specification-based.

In [17], a taxonomy with five criteria is described. The data source criterion examines where the data come from, dividing IDs into host- and network-based. Another criterion is the detection method comprising a scenario category, which is equivalent to the signature-based category seen in other taxonomies, and a behavioral category of IDSs. The data analysis location criterion considers whether the data are analyzed locally or centrally. Depending on the analysis frequency, an IDS can be classified as periodic or continuous. Finally, according to its behavior after detection, an IDS can be passive, when triggering an alarm, or active, when stopping the attack.

The taxonomy described in [18] for CAN bus-based IDSs is based on the different aspects that they have. The first aspect concerns deployment and an IDS is considered onboard, when deployed onboard the vehicle, and external, when deployed externally; clearly, this division concerns other types of IDSs, such as those intended for VANETs, and not the CAN-based ones which are all onboard. Regarding the architecture, an IDS can be considered self-detective, collaborative, or offloaded. A self-detective IDS operates individually, whereas an offloaded one offloads the detection processing operations to the cloud; a collaborative architecture does not apply to CAN bus but rather to VANETs. Another IDS aspect is type, which is divided into knowledge and behavior, depending on whether the IDS operates based on signatures or models of normal behavior, respectively. An IDS can also be characterized by the type of features used, which can be cyber, such as network data, or physical, such as speed. Depending on the detection technology used, IDSs can be learning, when using statistical or ML techniques, or rule-based, when specified rules are used. IDSs can also be classified according to the attacks targeting a vehicle in the following categories: confidentiality, integrity, or availability. Finally, the evaluation category shows the approach followed in each proposed IDS and it can be analytical, simulation, or experimental.

The authors of [9] propose four main classifications: reaction type (active, passive, real-time detection), detection methodology (signature, watchdog, anomaly, cross layer, hybrid, honeypot), validation strategy (simulation, empirical, hypothetical, theoretical), and deployment location (centralized RSU, distributed individual node, cluster head, hybrid). An active IDS, also known as an intrusion detection and prevention system (IDPS), automatically blocks detected suspicious traffic. A passive IDS only monitors and analyzes traffic, alerting an operator when something suspicious has been detected. Real-time IDSs undertake the challenging task to detect intrusions in ultra-high-speed environments in real or near real time. Regarding the detection methodologies, the well-known signature-based methodology tries to match the signature of an incident against those stored in a database. In case of a match, the IDS flags that attack based on the signature of the database. According to the authors, a watchdog is a special security feature installed in different nodes of a VANET network. These nodes have the responsibility to monitor, capture, and report every potential malicious action. That is, a node can operate in promiscuous mode, listening to the packets of its neighborhood. Then, based on the collected packets, the watchdog can decide if a particular node behaves as a selfish, black, or gray hole router [22]. As already pointed out, opposite to the signature-based IDS, an anomaly-based one can alert for suspicious behavior that is unknown; this is carried out by monitoring system activity and classifying it as either normal or anomalous. A cross-layer-based IDS monitors multiple layers of the communication that a node may have. As a result, it can detect an assault irrespective of the layer the latter operates. A hybrid IDS combines signature and anomaly ones. Lastly, an onboard unit (OBU) honeypot should simulate the in-vehicle network, and therefore should be equipped with an ECU simulator software. In the same mindset, an RSU honeypot can be built. The main issue with honeypots is that they should operate as both normal and intermediate nodes, and therefore are reachable both by attackers and legitimate nodes. Therefore, if the honeypot exposes a real service, then it is made prone to exploitation, while if it simulates one, it may delude legitimate nodes. According to the authors, an empirical validation approach, say, setting up a CAN network and conducting experiments, was utilized in the 16% of the 24 surveyed works

proposing an IDS. A simulation approach was the most popular one, with 76% of the surveyed papers to validate their experiments by utilizing an emulator, such as SUMO or VANET Mobisim. Only 4% of these works utilized a theoretical approach, and an equal percent did not perform any validation at all. The deployment location classification pertains to three IDS architectures, namely, decentralized (an IDS exists in multiple or all the individual nodes of the network), centralized (the IDS relies on cluster head or road side unit (RSU) components), and hybrid, which comprises a combination of decentralized and centralized.

The authors in [19] classify CAN-based IDSs mainly into signature- and anomaly-based. Then, the anomaly-based ones are divided into statistical, knowledge-based, and machine learning. Statistical IDSs are further subdivided into univariate, when modeling each variable independently, and multivariate, when multiple variables are considered at the same time. A knowledge-based IDS uses training data to create a set of rules and, after the training phase, events can be labeled into normal or anomalous, based on these rules. ML IDSs can also be further categorized according to the method used into clustering techniques, hidden Markov models, support vector machines, and neural networks.

3.2. A Unified Taxonomy

The main observation from the analysis in the previous section and the summary presented in Figure 1 is that the taxonomies used in the literature are very diverse. Moreover, it is very rare that the same taxonomy is used across different surveys. However, a single, comprehensive way of categorizing IVN IDS proposals would allow better comparison among them and easier identification of their advantages and disadvantages. Additionally, there would be no need to reclassify all previous work every time a new survey with a new taxonomy is published.

In an effort to synthesize an inclusive but more abstract taxonomy, considering all the existing categorizations as described above, we present a unified taxonomy illustrated in Figure 2. In this new unified taxonomy, an IDS does not belong to a single category but has four characteristics: location, type, layering, and reaction type. Regarding its location, it can be host- or network-based, its type can be signature-, anomaly-based, or hybrid, it can cover a single or multiple OSI layers and, finally, it can be active or passive. In the proposed taxonomy it is not enough to state that an IDS is just signature- or anomaly-based, but all four characteristics must be defined. For example, one category would comprise host-based/hybrid/single-layer/active IDSs, another category would be host-based/hybrid/cross-layer/active IDSs, and so on.

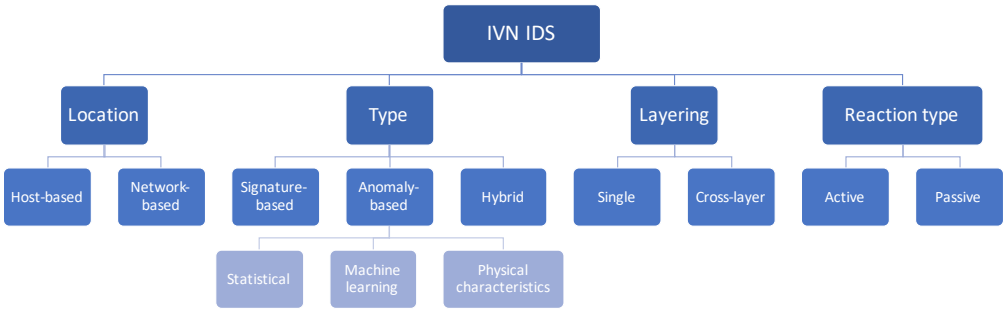

Figure 2. Proposed unified IVN IDS taxonomy.

In more detail, the location characteristic shows the deployment point of the in-vehicle IDS; a host-based IDS is installed in a single ECU and monitors the ECU itself and packets entering and leaving, whereas a network-based IDS checks the packets inside the entire IVN network for suspicious behavior. The type differentiates IDSs according to the detection method. A signature-based IDS, usually referenced as knowledge- or fingerprint-based, creates a profile of what is considered normal behavior using patterns or signatures and detects actions that deviate from this "normal behavior"; naturally, such an

IDS can only detect known attacks. An anomaly-based IDS, also known as behavior-based, monitors the system's activity and classifies it as either normal or anomalous based on heuristics or rules; the advantage of such IDSs is that they can detect novel attacks as well. Anomaly-based IDSs can be further classified into statistical, when using statistical approaches such as information entropy, machine learning, when using ML techniques such as deep learning, and IDSs, based on physical characteristics when measuring unique hardware characteristics of the ECUs. It is possible that an IDS uses both a signature- and an anomaly-based approach; in this case, it can be classified as the hybrid type. The layering characteristic of IVN IDSs shows the extent of data that the IDS analyzes during detection. A single-layer system monitors information from a single layer of a networking model, for example, the headers of CAN bus messages; a cross-layer approach monitors more than one layer, for example, network and application layer data. Regarding the reaction type, an active IDS is designed to automatically block suspected malicious behavior, whereas a passive one only raises alerts.

In the rest of this section, we demonstrate the capacity of the proposed unified taxonomy by employing it to classify related work in IVN IDS. In this context, and considering that the surveys presented in Section 2 already provide such analysis based on their own taxonomy or specific aspects ([6,13,18]), we narrow down our analysis to works published in the period 2020–2022, so that we cover the most recent proposals in this field. Table 2 summarizes this related work and classifies it according to the proposed unified taxonomy for IVN IDSs. From the table it is evident that most IDSs are NIDS, anomaly-based (and more specifically ML-based), single-layer, and passive. The NIDS/anomaly (machine learning)/single/passive type is followed in 68% (or in 30 out of 44) of the proposed IDSs. Regarding the in-vehicle bus protocol that they are intended for, the majority is designed for the CAN bus, whereas only three works focus on automotive Ethernet. Taking each characteristic individually, host-based deployment is preferred in only 2 out of 44 IDSs, and the cross-layer approach in only 3 proposals, while all IDSs are passive. Regarding the detection method, there are three signature-based and one hybrid IDS; the rest are anomaly-based with the majority being ML (83% or 33 out of 40 of the anomaly-based ones), six statistical, and one based on physical characteristics.

Table 2. IDS approaches for IVN between 2020 and 2022. N: NIDS, H: HIDS, A: anomaly-based, ML: machine learning, PC: physical characteristics, S: single, C: cross, P: passive.

Reference	Year	Bus Protocol	Location	Type	Layering	Reaction Type
[23]	2022	CAN	N	A (ML)	S	P
[24]	2022	CAN	N	A (ML)	S	P
[25]	2021	CAN	N	A (ML)	S	P
[26]	2021	CAN	N	A (ML)	S	P
[27]	2021	CAN	N	A (ML)	S	P
[28]	2021	CAN	N	A (ML)	S	P
[29]	2021	CAN	N	A (ML)	S	P
[30]	2021	CAN	N	A (ML)	S	P
[31]	2021	CAN	N	A (ML)	S	P
[32]	2021	CAN	N	A (ML)	S	P
[33]	2021	CAN	N	A (ML)	S	P
[34]	2021	CAN	N	A (ML)	S	P
[35]	2021	CAN	N	A (ML)	S	P
[36]	2021	CAN	N	A (ML)	S	P
[37]	2021	CAN	N	A (ML)	S	P
[38]	2021	CAN	N	A (ML)	S	P
[39]	2021	CAN	N	A (ML)	S	P
[40]	2021	CAN	N	A (ML)	S	P
[41]	2021	CAN	N	A (ML)	S	P
[42]	2021	CAN	N	A (ML)	S	P

Table 2. *Cont.*

Reference	Year	Bus Protocol	Location	Type	Layering	Reaction Type
[43]	2020	CAN	N	A (ML)	S	P
[44]	2020	CAN	N	A (ML)	S	P
[45]	2020	CAN	N	A (ML)	S	P
[46]	2020	CAN	N	A (ML)	S	P
[47]	2020	CAN	N	A (ML)	S	P
[48]	2020	CAN	N	A (ML)	S	P
[49]	2020	CAN	N	A (ML)	S	P
[50]	2020	CAN	N	A (ML)	S	P
[51]	2021	CAN	N	A (ML)	Cross	P
[52]	2021	CAN	N	A (ML)	Cross	P
[53]	2020	CAN	N	A (ML)	Cross	P
[54]	2022	Ethernet	N	A (ML)	S	P
[55]	2021	Ethernet	N	A (ML)	S	P
[56]	2021	CAN	N	A (statistical)	S	P
[57]	2021	CAN	N	A (statistical)	S	P
[58]	2021	CAN	N	A (statistical)	S	P
[59]	2021	CAN	N	A (statistical)	S	P
[60]	2021	CAN	N	A (statistical)	S	P
[61]	2020	CAN	N	A (statistical)	S	P
[62]	2021	CAN	H	A (PC)	S	P
[63]	2021	Ethernet	N	Signature	S	P
[64]	2020	CAN	N	Signature	S	P
[65]	2021	CAN	H	Signature	S	P
[66]	2022	CAN	N	Hybrid	S	P

4. Datasets

The current section elaborates on publicly released datasets destined primarily to IVN and, for the sake of completeness, to VANET, and automotive in general. With reference to [67], the great majority of the so-far proposed VIDS utilized older datasets or employed a simulation tool, depending on the particular case. This section aspires to provide a deeper look at each security-oriented dataset based on three key axes: the technology covered by the dataset, the attack types it contains, and its characteristics. Table 3 provides a condensed view of the various datasets based on the latter categorization and specifically on nine distinct criteria. Note that for offering a complete picture, we also succinctly refer to non-security-focused datasets.

Table 3. Publicly available VIDS datasets sorted by technology and year in ascending order. The "*" denotes that the mentioned statement is partially applied. T/T: training/testing, R/S: realistic or simulated testbed, N/A: not applicable or unknown, SD: depends on the simulation scenario.

Dataset	Year	Technology	T/T Sets	# Features	# Total Rows	# Attacks	Labeled	R/S	# Nodes
Kang and Kang [68]	2016	CAN	✓	7	200K	1	✗	S	3
OTIDS [69]	2017	CAN	✗	✓*	≈4.6M	3	✓*	R	1
Car Hacking Dataset v1 [70]	2018	CAN	✗	12	SD	4	✓	R	N/A
Survival [71]	2018	CAN	✗	12	SD	4	✓	R	4
IDS in CAN [72]	2018	CAN	✗	N/A	SD	2	N/A	R	2
CAN Intrusion Dataset v2 [73]	2019	CAN	✓	N/A	SD	5	✗	Both	2
SynCAN [74]	2020	CAN	✓	7	≈42M	5	✗	Both	1
Car Hacking Dataset v2 [75]	2021	CAN	✗	6	SD	4	N/A	R	1
KITTI [76]	2020	Ethernet	✓	41	500K	14	✓	R	N/A
Automotive Ethernet ID [55]	2021	Ethernet/AVTP	✗	6	SD	1	N/A	Both	N/A
AWID2 [77]	2015	Wi-Fi	✓	155	SD	22	✓	R	12
AWID3 [78]	2021	Wi-Fi	✗	254	≈30M	21	✓	R	17
UAV [79]	2021	SDR	✗	N/A	N/A	2	N/A	N/A	N/A
BlueTack [80]	2021	Bluetooth	✓	22	N/A	3	N/A	R	N/A
VeReMi v1 [81]	2018	GPS/DSRC	✗	N/A	SD	5	N/A	S	SD
VeReMi v2 [82]	2020	GPS/DSRC/Sensors	✗	N/A	SD	6	N/A	S	SD

4.1. Wired Datasets

In Kang and Kang [68], a simulated in-vehicular network comprising three ECUs was built, and during the simulation, the packet generator Open Car Test-bed and Network Experiments (OCTANE) [83] was used to insert packets in the CAN bus. This 200K CAN bus messages dataset was created to be used for benchmarking a deep-neural-network (DNN)-based IDS for CAN and is publicly available in text format (https://doi.org/10.1371/journal.pone.0155781.s001 (accessed on 7 March 2022)). Out of these data, 70% were used as training data and the remaining 30% as testing data. The authors consider a general injection type of attack where malicious CAN frames are injected into the bus.

The OTIDS IDS dataset [69] released back in 2017 targets IVN, and specifically CAN. Along with CAN normal traffic, it contains three attack categories, namely, DoS, fuzzy, and impersonation [84]. The two datasets of normal traffic have a total size of around 354 MB (≈4.6 million messages), whereas the sizes of the three attack datasets are 60 MB (DoS), 50 MB (Fuzzy), and 84 MB (impersonation). The dataset was extracted from a KIA SOUL car by logging the CAN traffic via the OBD port. In addition, the dataset is partially labeled, that is, only for the DoS attack, and given in text format.

The Car Hacking Dataset v1 [70] was introduced in 2018. It focuses on a quartet of CAN bus attacks, namely, DoS, fuzzy, and spoofing of the drive gear or the revolutions per minute (RPM) gauge. The dataset is offered in CSV format and has a size of 185 MB (DoS), 193 MB (fuzzy), 233 MB (gear), and 224 MB (RPM). The dataset also contains an 88 MB file of normal data in text format. The authors gathered the relevant data via the OBD port of a real vehicle. The total duration of each attack was ≈35 min, containing around 300 malicious frames. The relevant dataset includes 12 features, namely, timestamp, CAN ID, the data length of the requested message (DLC), DATA[0] to DATA[7], and Flag.

The Survival dataset [71] presented in 2018 is also CAN-oriented and extends the Car Hacking Dataset v1 [70]. The authors collected data from three different vehicles, namely, Hyundai Sonata, KIA Soul, and Chevrolet Spark. Regarding the assaults, they replaced the spoofing attack of the Car Hacking Dataset with a "malfunction" one, where malicious CAN ID data frames are sent with the purpose to induce a malfunction. Additionally, an attack-free sample was collected from each vehicle. The whole unzipped size of the dataset is 13 MB. The dataset has the same dozen features as the Car Hacking Dataset v1, but it is given in text format.

Another dataset destined to CAN bus was given in [72]. This dataset, titled "IDS in CAN" in Table 3, comprises a fusion of three other datasets. The first one is the Car Hacking Dataset v1 [70], from which the authors picked the "fuzzy" attack. The other two datasets were new collections, utilizing the ML350 and CL2000 Mercedes models. Overall, this collective dataset contains two attacks, namely, DoS and "fuzzy", and has an unzipped size of 43 MB.

The authors in [73] contributed a dataset called "Automotive Controller Area Network (CAN) Bus Intrusion Dataset v2". Its purpose is to be used in CAN bus VIDS evaluations. The dataset contains CAN bus data from two real-life cars and a CAN bus prototype built by the authors. For each of these setups, the dataset comprises a set of log files, either attack-free or others used for training and testing the VIDS. Five different types of attacks are considered: diagnostic, fuzzing, replay, suspension, and DoS.

The work in Hanselmann et al. [74], proposed an unsupervised learning approach to detect known and zero day intrusions in CAN traffic. The contributed deep-learning-based VIDS concentrates on the data structure of the high-dimensional CAN bus, where diverse message types are transmitted at varying times. They evaluated their proposal using a dataset of both real (13 message IDs with a total number of 20 signals) and synthetic (10 different message IDs with 20 signals) CAN data, corresponding to about 13 and 24 h of recorded data, respectively. The dataset called "SynCAN" contains five attacks, namely plateau, continuous change, playback, flooding, and suppress.

An extension of the previously mentioned datasets [70–72] was presented in 2021 [75]. This variation includes four attacks, namely, replay, flooding, spoofing, and fuzzing,

with only the first of them being new. The dataset was created following the same procedure, that is, via the use of an OBD tool in a Hyundai Avante CN7. A key difference, as compared to the previous datasets by the same authors, is that in this one the data gathering process is different, that is, it happened in two rounds, namely, preliminary and final. In the first round, two subsets were formed, "training" and "submission". In the second round, only a "submission" subset was compiled. Offered in CSV format, the dataset has an unzipped size of 77 MB and contains six features per CAN frame, namely, timestamp (logging time), arbitration_ID (CAN identifier), DLC (data length code), data (CAN data field), class (normal or attack), and subclass (attack type).

The KITTI dataset introduced in [76] is based on the widely tested, but rather obsolete, benchmark dataset KDD99. Precisely, the authors kept four categories of relevance to CAV attacks, namely, probe, DoS, U2R, and R2L from the original KDD99, whereas the entire KITTI dataset contains 14 attacks split among the aforementioned categories. The training and test datasets were built based on the 10% of the KDD99 dataset, that is, around 500k data records.

The work given in [55] presents an intrusion detection method for detecting audio–video transport protocol (AVTP) stream injection attacks in automotive Ethernet-based networks. The authors also generated a dataset [85] based on a BroadR-Reach-based testbed, which allowed them to capture real AVTP packets. Precisely, for realizing a replay attack, they injected previously generated AVTP data units (AVTPDUs) during a certain period into the IVN. The dataset has an unzipped size of 1.44 GB, and is offered in pcap format.

4.2. Wireless Datasets

An in-vehicle IDS can protect from both insiders and outsiders. Regarding Wi-Fi connections, two wireless datasets [77,78] are publicly available. The first one, namely AWID2, focuses on WPA and WPA2-Personal and contains 24 attacks that are exercised on the MAC layer. This dataset is given in CSV format. On the other hand, AWID3 focuses on WPA2-Enterprise, including protected management frames (PMF). Twenty-one different attacks are considered, ranging from legacy deauthentication to more advanced and higher-layer ones, such as amplification, malware, and botnets. This dataset is offered in both CSV and pcap format. Both of these datasets are not destined to ad hoc scenarios.

The ad hoc network topology is addressed by the rather tiny simulated dataset given in [86]. Specifically, by utilizing five different simulation platforms, the authors managed to create a dataset that contains two attack scenarios, namely, GPS spoofing and jamming. Although this dataset seems to be the only one available for unmanned aerial vehicle (UAV) security, it has a small size of <19 MB, and is available in [79].

For Bluetooth, there is only one security-focused dataset [80]. It contains three type of attacks, namely, DDoS, DoS on L2CAP, and the BlueSmack attack. On the negative side, this dataset has a tiny size of 3 MB and comprises 22 classification features. The interested reader can also refer to [87] for a non-security-oriented Bluetooth dataset. Regarding cellular networks, and specifically 4G or 5G, to our knowledge, the literature misses a security-oriented dataset; a general purpose dataset containing normal traces of 4G communication traffic can be found in [88].

The Vehicular Reference Misbehavior dataset (VeReMi) v1 [81] comprises a simulated dataset for the evaluation of misbehavior detection mechanisms for VANETs. According to the authors, VeReMi is extensible, "allowing anyone to reproduce the generation process, as well as contribute attacks and use the data to compare new detection mechanisms against existing ones". The dataset is labeled and comprises message logs of OBUs per vehicle generated from a simulation environment, namely, Luxembourg SUMO Traffic (LuST) scenario [89] and VEINS. The data are encoded in JavaScript Object Notation (JSON) format. Specifically, it contains GPS data about the local vehicle and basic safety messages (BSM) received from other vehicles over dedicated short-range communications (DSRC). The malicious messages included in the dataset are assumed to initiate incorrect application behavior, which should then be detected by an IDS. VeReMi consists of diverse

density levels from 35 to 519 vehicles, five different VEINS-coded attacks, and three different attacker densities, that is, out of the total number of vehicles, a subset is malicious. The attacks implemented are associated with one out of five attacker models: constant attacker, constant offset attacker, random attacker, random offset attacker, and eventual stop attacker. The first type transmits a static position, the second a static offset added to their real position, the third a random position, the fourth a random position in a preset rectangle around the vehicle, and the fifth acts normally for a specified window of time and then assaults by transmitting the current position continually. The number of messages transmitted depends on the simulation scenario and the densities; generally, this number spans between 908 to 1144 (low densities), 3996 to 4489 (medium densities), and 20,482 to 21,878 (high densities).

VeReMi has been extended (v2) in [82] by adding four vehicle sensor error models (position, velocity, acceleration, and heading), an updated repertoire of attacks (DoS, DoS random, data replay, disruptive, eventual stop, and traffic congestion Sybil), and greater number of data points. Precisely, for creating the dataset, the authors exploited F^2MD, which caters for the generation and detection of various misbehavior detection use cases. Depending on the attack scenario, the dataset files have a compressed size spanning from around 2.2 to 8.4 GB. It is to be noted that the authors differentiate between malfunctions and attacks; a malfunction is a non-malicious behavior stemming from a malfunctioning OBU or sensors, while an attack has a malicious intention, purposely sending erroneous data.

4.3. Discussion

In summary, the creation of full-fledged automotive datasets is a challenging task due to the diversity of network technologies utilized in this discipline, e.g., CAN, Flexray, Ethernet, Bluetooth, cellular, and Wi-Fi. On the other hand, the majority of the currently available datasets present a limited number of features and number of rows, which typically is a major restrictive factor towards constructing an IDS model, especially if the latter exploits neural networks. Moreover, there is a distinct lack of public datasets addressing cellular and Bluetooth communications, while those available for some other technologies, say, Wi-Fi, cover only infrastructure-based networks and not ad hoc or mesh ones. On the other hand, while there is an abundance of CAN bus datasets, this is logical, since this technology has been a workhorse for the automobile industry for around three decades, Ethernet is emerging as the clear choice for new bus architecture in automotive electronics, and on top of the fact that CAN bus speed lags behind Ethernet, a principal advantage of the latter is that it enables cybersecurity by design. In light of the foregoing, the construction of modern datasets in terms of both network technology and diversity, as well as sophistication of the attacks they contain, is an insistent and continuous demand in this sector. A last, but important, observation is that a significant number of the examined datasets do not provide enough technical details regarding their characteristics, say, number and type of features, whether they are labeled or not, amount of network nodes used, etc. This makes its utilization harder for researchers and other interested parties.

5. Simulation Tools

In the case of VIDS, simulation tools are not only handy to imitate the operation of a real-world IDS process over time, but also to generate datasets that can be later on used to train or evaluate a VIDS. This section elaborates on the insofar publicly available simulators, and especially those which provide security modules and have been utilized in VANET security literature. In this respect, commercial simulators, including NetSim [90] (only the pro and standard licenses provide support for VANET simulations), EstiNet [91], CANoe [92], and ezCar2X [93] are intentionally left out. For a more detailed analysis of some of the simulators mentioned in this section, the interested reader can refer to [94].

Vehicles in-network simulation (VEINS) [95] comprises an open-source inter-vehicular communication (IVC) simulation framework available for Linux, Windows, and Mac OS.

It is based on two well-respected simulators, namely, the event-based network simulator OMNeT++ and the Simulation of Urban MObility (SUMO), a road traffic (V2X) simulator. VEINS does not provide a dedicated advanced driver-assistance systems (ADAS) module and, thus, can only partially simulate sensor measurements. VEINS includes several extensions that allow modeling of diverse protocol stacks, including IEEE 802.11p and ETSI ITS-G5, as well as specific applications such as security [96], misbehavior detection through the simulation framework F^{2}MD proposed in [97], and location privacy via the PREXT module [98]. For instance, the work in [99] capitalizes on the PREXT module to evaluate a context-based location privacy scheme in VANET. No less important, VEINS offers APIs for building custom applications that run locally in a vehicle. Lastly, Artery [100] is an extension to VEINS that offers an implementation of the ETSI ITS-G5 protocol. This extension supports the collection and provision of state and perception data required for ADAS algorithms for each vehicle.

VENTOS [101] is an open-source integrated VANET C++ simulator for analyzing vehicular network applications, including collaborative driving, automated cruise control, and platooning. Similar to VEINS, it uses SUMO and OMNET++ for mobility and network modeling, respectively. VENTOS does not support ADAS applications due to lack of a sensor model. VENTOS has also been utilized to study security attacks in collaborative driving [102] and specifically for edge-assisted misbehavior detection for platoons [103,104].

The Integrated Wireless and Traffic Platform for Real-Time Road Traffic Management Solutions (iTETRIS) [105] is an open-source simulation framework. Its V2X communication module is based on NS-3. Similar to VEINS, the traffic mobility module of iTETRIS relies on SUMO. Currently, this framework does not provide an ADAS or a security-focused module.

Car Learning to Act (CARLA) [106] is an open-source urban driving simulator for supporting the "development, training, and validation of autonomous driving systems". The simulator does not include a V2X communication module, except the extension proposed in [107]. CARLA incorporates a traffic simulator for both vehicles and pedestrians and supports an autonomous driving sensor suite allowing the configuration of different sensors such as LIDARs, cameras, depth sensors, and GPS. CARLA does not offer any security or privacy module.

Another simulator that does not currently provide any security module is AirSim [108]. This is an open-source project, which is supported by Microsoft Research Team. AirSim is designed with extendability in mind, so it can accommodate "new types of vehicles, hardware platforms, and software protocols". Moreover, sensor models have been implemented as a C++ header-only library, thus being easily portable to other environments. Given that the simulator is built around the Unreal Engine, i.e., a real-time 3D creation tool, it can be used towards collecting a large amount of annotated training data under a plethora of conditions and environments.

VANETsim [109] is an event-driven simulator designed to "investigate application-level privacy and security implications in vehicular communications". Specifically, VANETsim allows for analyzing attacks and countermeasures from an application viewpoint, namely creating an attack and assessing its impact on a vehicle. Unfortunately, the VANETsim project in GitHub was closed in April of 2017.

Vanetza [110] is an open-source implementation of the ETSI ITS-G5 protocol suite. Among others, Vanetza offers geoNetworking, basic transport protocol, decentralized congestion control, security, and support for cooperative awareness and decentralized environmental notification messages. The built-in security component can be used to sign and verify packets through the use of third-party libraries such as Crypto++ or OpenSSL.

The Veneris open-source framework introduced in [111] comprises a traffic simulator, implemented on top of the Unity game engine. Veneris also allows bidirectional coupling with OMNET++. Currently, this simulator neither offers a C-ITS communication stack nor implements a data flow scheme from vehicular sensors.

NCTUns [112] is a open-source discrete event-based network simulator/emulator that runs on Linux. While it supports IEEE 802.11p/1609 WAVE vehicular networks, it does not include any security focused modules.

With reference to security functionalities, Veins, VENTOS, and Eclipse MOSAIC do offer some potential for conducting testing in a VANET environment. For instance, the work in [113] proposes a scheme for detecting selfish nodes in MANETs; the authors rely on OMNET++ simulations. Moreover, the authors in [114] propose a redundancy-based protocol for safety message dissemination in VANET; for simulating the proposed scheme, they exploited SUMO. OMNET++ has been also used to test digital signature and authentication mechanisms in cognitive radio networks [115]. Researchers have also utilized VEINS, VENTOS, and VANETsim for implementing security in V2X. For instance, the authors in [116] propose an authentication and key agreement scheme for V2V and evaluate it through VEINS. Additionally, the work in [117] uses VENTOS to model diverse attacks against cooperative adaptive cruise control (platooning). The authors in [118] capitalized on VANETsim to assess a priority-based routing protocol for inter-vehicular communication.

6. Challenges

The continuous developments in automotive electronics technologies bring along important challenges that span across several levels, including latency, intelligence, security, and mobility support. This section concentrates on key open issues that need to be worked on or resolved in current and future VIDS.

6.1. External Interfaces and Attack Surface

The increasing vehicle connectivity will be boosted with the advent of connected and autonomous vehicles (CAVs), which will require communication with other vehicles and objects (such as traffic lights) in the surrounding environment to provide efficient and safe autonomous driving. However, as described by [12], the integration of these communication interfaces with the IVN could also increase the impact and likelihood of security threats. As also pinpointed in [14,20], separating potential attacking network interfaces from IVN, that is, minimizing the attack surface, makes much harder for adversaries to connect to the IVN and mount assaults. This, however, is not practical for certain interfaces. For instance, the OBD port is used to communicate with the vehicle's systems to help diagnose problems, and thus isolating it from the in-vehicle network is rather unrealistic. Therefore, current solutions incorporate detectors to OBD port to discern between normal and aberrantly injected frames. Moreover, the communication related to systems such as telematics or GPS goes usually through a central gateway or multiple distributed gateways [119], which are also used to interconnect different buses. Therefore, the implementation of restrictions or rules in such components could be considered to isolate such external communications from the IVN, so that the attack surface could be reduced.

6.2. Interoperability

As already mentioned, CAVs require remaining in constant communication with their surroundings for being able to assess the current situation and make decisions in real time, ensuring safe transportation. In this respect, the authenticity and integrity of the sensed or received data is a sine qua non for achieving autonomous interconnected safe driving in a dynamic environment. This, however, is not limited to the vehicles themselves as an ad hoc network, but also to the rest of the components and especially to the road and network infrastructure, including Wi-Fi and cellular base stations and both the access and backhaul network links. While some wireless standards, such as IEEE 802.11p (ETSI ITS-G5), LTE-V2X, and New Radio (NR) V2X [120], have been developed especially for the vehicular terrain, several interoperability issues related to security aspects could be derived from the coexistence of some of these technologies in a real-life deployment [121] that could have a direct impact on the IVN security.

Naturally, as already pointed out, to support interoperability, diminish the attack surface, and support future applications, the connectivity security issues should focus not only on the vehicle itself but on the whole infrastructure and service value chain as well. In this respect, key questions, e.g., who controls and provides the communication infrastructure in the CAV ecosystem, are still to be defined. Simply stated, it is important to correctly and clearly stipulate roles and responsibilities for the involved parties and attain a wise balance between private and public control. Furthermore, the use of private protocols (or the use of own tailored implementations of standard protocols) used by different OEMs may be an impediment for VIDS. That is, the implementation of ECU systems and transport protocols responsible to convey, say, CAN messages may be quite dissimilar across various OEMs. This may result in lowering the detection accuracy of an IDS if it is used in a dissimilar setting. Moreover, the accuracy of detection can be affected by inherent properties of the bus protocol. For instance, CAN messages transmission may occur abruptly due to, say, re-transmissions, bit errors, etc. Therefore, if the IDS is designed based on the normal message transmission profile, it may be prone to a high rate of false positives [122].

6.3. Heterogeneity of Network Technologies

Heterogeneous wireless networks, and especially VANETs, are considered to be susceptible to an assortment of threats in comparison to their wired counterparts. This stems from the complex network topology (in-vehicle, road-side, cloud infrastructure, etc.) and the high mobility conditions in the VANET realm. The opponent is presented with a large attack surface, that is, multiple points of entry. These range from accompanying, to vehicle mobile apps [123], to a plethora of wireless or other type of interfaces, including Wi-Fi [124], Bluetooth [125], cellular [126], FM [127], keyless entry systems, and even voice commands [128]. All these access points can offer an initial foothold that can possibly lead to compromising the in-vehicle security. Upon compromise, the vehicle and the VANET itself may become prone to a range of perilous attacks, including botnets [129], and, continuing from the previous point, most of the proposed IVN IDS systems have high accuracy. However, each system monitors a single network layer, and, consequently, they do not provide comprehensive approaches considering the potential impact of diverse attacks at several vehicle network layers.

6.4. Data Privacy

Another major issue is who regulates access to the data collected by the vehicle, the associated applications, and the backend. In fact, such privacy concerns have been already touched upon by current standards such as the ISO 20077 [130], but, in general, the relevant issues are not well-tackled or defined even in the newest standards and regulations, including the UN R155 [131] and R156 [132]. In particular, ISO 20077 defines the concept of *extended vehicle* to represent the increased functionality of vehicles based on the development of services by using their data. Even if a general process is defined for the access to such data by considering the vehicle's manufacturer, it is still not clear which responsibilities are associated with manufacturers and service providers, which will use such data to develop new services. Furthermore, a 2017 survey performed by the German consumer organization "Stiftung Warentest" revealed that the great majority of connectivity schemes offered by automotive OEMs are prone to certain privacy leaks [133]. Threats against privacy have also been recently exposed for the official (OEM's) mobile applications that accompany modern vehicles [123]. Namely, among others, personal information may be communicated unencrypted, and certain pieces of private information, including those collected by a VIDS, may be gathered and transmitted without prior user consent. Indeed, in the case of ML-enabled VIDS, the use of modern techniques could lead to increasing privacy risks because of the access to all the data derived from the IVN traffic, as well as the possibility of inferring new sensitive information as a result of applying such techniques.

Such solutions could negatively impact data protection, therefore calling for compliance to, say, the General Data Protection Regulation (GDPR) [134].

6.5. Safety Engineering and Quantification of Risk

As the automotive industry relies heavily on ICT technology, cyberattacks can have direct and adverse effects on transport safety. However, as described by recent works, there is a gap between a vehicle's functional security and the security aspects of IVNs that requires an adaptation of the existing functional safety methods and processes [13]. The functional safety requiremenFts of the complete lifecycle of every safety-related automotive electronic/electrical system is defined in the ISO 26262 standard [135]. In particular, it addresses possible hazards caused by malfunctioning behavior of safety-related electrical and electronic (E/E) systems, including interaction of these systems. Furthermore, IEC 61,508 [136] covers those aspects to be considered when electrical/electronic/programmable electronic (E/E/PE) systems are used to carry out safety functions. However, such standards cover neither the cybersecurity aspects throughout the vehicle's lifecycle nor the relationship between safety and cybersecurity concerns. In this context, traditional security engineering falls short and should be combined with safety engineering [137,138]. Indeed, apart from detecting purely ICT-related threats, IVN IDS should also take into account the safety dimension identifying the risk associated with each attack and implementing the appropriate countermeasures accordingly.

In this respect, cyber risk standardization and regulation is deemed as a decisive factor towards decreasing cyberattacks against automotive. Indeed, cyber risk in this sector may span across diverse levels and be cumbersome to assess and quantify. Moreover, the residual cyber risk, that is, the remaining risk after every cybersecurity recommendation has been taken into account, can be quite high. A prominent example of this situation is the risk associated with the supply chain threat as discussed in Section 6.10. Under this prism, the standardization of cyber risks and risk assessment, also through the lens of recent regulations UN 155 [131] and UN 156 [132], can serve as a lodestar for better understanding and quantifying cyber risk posture of IVN and automotive in general. The interested reader is also referred to the interesting work by Radanliev et al. [139] for analyzing uncontrollable states in complex systems. Additionally, a potential starting point could be based on the consideration of the SAE J3061 guidebook [140], which provides a cybersecurity process framework and guidance to help organizations identify and assess cybersecurity threats in vehicle systems. More specifically for IVN IDS approaches, the recent ITU-T X.1375 Recommendation [141] establishes a set of guidelines for IVN IDS and identifies threats to existing IVNs, such as CAN, that could potentially imply safety concerns.

6.6. Hardware Limitations

Hardware limitations of ECU may be a serious hindering factor for the application of some resource-intensive VIDS. Namely, legacy ECUs typically comprise microcontrollers with a maximum clock speed of several hundred MHz and a limited RAM. In this respect, computational complex schemes, such as the one in [142] or others which require extra equipment [143], may be not be practical for current vehicles. Indeed, storage, computation, battery, and bandwidth limitations can prevent an IDS approach from satisfying the real-time requirements of vehicular environments with the consequent safety implications, which typically involve security risks [7]. As also detailed in Section 6.7, these aspects may be exacerbated in the case of sophisticated ML/DL-enabled IDS approaches that could make their deployment in existing IVNs infeasible. To overcome these limitations, a potential approach is the deployment of the IVN IDS in the different gateways, which are typically used to interconnect ECUs. However, it is not clear if existing gateways of commercial vehicles have enough resources to execute complex machine learning algorithms to identity potential security attacks. Another potential approach is the use of intermediate nodes (see Section 7.6) to offload learning tasks for internal vehicle components. However, as already pointed out, this approach could have privacy implications if vehicles need to share their

data with external entities. Furthermore, as described by [7], the deployment of IVN IDS must take into account the instability of vehicle connections due to mobility.

6.7. Use of ML Approaches

ML algorithms can improve the detection capabilities of IVN IDS [52]. However, the following points should be considered before their deployment. First, a centralized approach sending vehicle data (such as MAC, VIN, and device ID) to a server or the cloud could potentially be associated with privacy concerns. Second, in the same centralized approach, the network and processing delays could hinder the efficacy of the IDS. Third, IVN IDS are based on resource-constrained devices, with limited throughput and intermittent communications, and a centralized machine learning solution could pose high overhead. To this end, federated learning (FL) (see Section 7.4), which refers to a collaborative learning approach based on decentralized data storage, could provide a viable solution respecting privacy and resource limitations. Furthermore, while many ML-enabled IVN IDS approaches have been proposed, the performance evaluation of some of these works have serious limitations, since they only consider performance metrics associated with the accuracy of the ML model being evaluated. As described by [5], an IDS has real-time requirements in the automotive context, so that appropriate countermeasures can be applied immediately after detecting such an attack. Therefore, the development of IDS approaches for in-vehicle networks must consider the complexity to demonstrate its feasibility in a real environment. Furthermore, such evaluations are based on non-exhaustive datasets, which do not cover a variety of attacks spanning across multiple network interfaces and protocol layers. For example, according to Section 4, so far only one dataset contains application layer attacks targeting data exfiltration [78], which nevertheless is quite common to advanced persistent threat (APT) groups [144]. Moreover, no dataset incorporates attacks relevant to FM [127] or voice-commands [128] exploitation. These aspects have also been mentioned by [5]. An additional point is that, similar to other contexts, many of the so-far proposed schemes do not provide a detailed overview of the tests performed to assess the accuracy of IDS approaches, or how the datasets were actually used. The main consequence is the difficulty in comparing existing ML-enabled IVN IDS approaches.

6.8. Adversarial ML

Related to the previous point, adversarial machine learning attacks [145–147] should be considered as a serious threat to VANET and CAV in general. This is because, among others, this type of assaults may aim at manipulating the results that an IDS can provide [148,149]. That is, the adversaries may exploit multiple ways of feeding the IDS's machine learning model with deceptive inputs in an attempt to trick it, and ultimately taint the results. Precisely, as with every other category of attacks, the adversary's goal can greatly vary because it depends on their position in the network, knowledge, capacity, and motivation. For instance, in a so-called evasion attack, a rogue vehicle may contribute malicious test samples to the network, or the adversary may be able to alter the training data, thus leading the classifier to produce faulty results. In another instance, the aggressor may be able to inject noise to a machine learning model, that is, by manipulating sensor readings or by changing the physical environment in the vehicle's vicinity. For more information on this topic, the interested reader is referred to [148].

6.9. Type Approval

In the automotive ecosystem, the *type approval* is usually referred to as the process to certify a vehicle, or verify that a certain vehicle's component meets a set of standard requirements. Ideally, such a process should be rooted in a commonly accepted security certification scheme and applied across all the vehicle digital components, either internal, e.g., in-vehicle firewalls, IDSs, and anti-tampering mechanisms, or external, e.g., the associated applications, the backend systems, and the roadside components. In this direction, the standardized common criteria (CC) framework as defined in ISO/IEC 15408 [150]

seems a straightforward choice. Indeed, CC [151] represents the most widely deployed and adopted certification scheme; however, it also presents some limitations related to the time and effort required for the execution of the certification process, the analysis of the evaluation-related documentation, and the management of changes in the certified product [152]. Irrespective of whether the CC or a similar methodology will be adopted in the future or not in the automotive realm, certain CC methods can be exploited for evaluations within the framework of ISO/SAE 21434 [153], even without formal certification. By doing so, the security of virtually any vehicle component, including IDS ones, can be systematically scrutinized and assessed at least against any known threat. Moreover, the identification of a certain attack or threat by an IDS could also have an impact on the type approval process of a certain vehicle by requiring a re-certification process. Indeed, vehicles could undergo changes, e.g., due to a software update [132] or a new vulnerability discovered by the IDS, that would require the execution of a new type approval process during their lifecycle.

6.10. Supply Chain

Supply chain attacks should be considered a serious threat [154] to VANET security in general and to VIDS in particular. In actuality, as described by [13], the use of components from different manufacturers in a certain vehicle poses significant security challenges for the whole vehicle. For instance, if an official or aftermarket vendor of a certain electronic component is being compromised somewhere along its supply chain, the perpetrator may be able to gain access to the firmware updates of the vendor. Next, the attacker may send malware along a legitimate software update request. Such a compromise may go unnoticed by the VIDS. Therefore, the use of standard approaches and the definition of standard security requirements and guidelines is crucial to ensure that the supply chain of vehicles' components is based on widely recognized procedures to avoid potential security breaches. In particular, the recent UN 156 Regulation [132] focuses on the requirements of software updates to be considered for the type approval process of the vehicle. Moreover, the use of blockchain approaches could also be considered in the future, so that different manufacturers could share information about their components in a trusted and decentralized way. Indeed, as described by [155], blockchain technology could also aid in maintaining the security information of each component updated throughout its lifecycle, including information about the certification scheme that was used to certify its security level, as well as the vulnerabilities or threats discovered.

6.11. Components beyond the Vehicle Bus

Both real-time operating system (RTOS) and middleware security is scarcely addressed in the context of VANET [156,157]. However, RTOSs such as QNX Neutrino and VxWorks, and middlewares such as Autosar and ZF, may be susceptible to a range of threats and behave differently if being attacked. That is, every RTOS or middleware vendor may implement and assess otherwise the security features of their product, and on top of security by design concerns, in this domain, security through obscurity still remains a thorny issue. This calls for the establishment of minimum security requirements, say, also in the context of and across certification schemes, regulations, and standards. For instance, the UN R155 [131] and UN R156 [132] regulations, adopted in June 2020 by the UNECE World Forum for Harmonization of Vehicle Regulations (UNECE WP.29), are expected to globally shape the future framework around vehicle cybersecurity. Both these regulations applying to passenger cars, vans, trucks, and buses came into force in January 2021. Jointly, they require that cybersecurity measures be implemented in the CAV ecosystem across four distinct axes: (a) managing vehicle cyber risks, as discussed in Section 6.5, (b) securing vehicles in a by-design fashion to mitigate risks along the whole value chain, (c) detecting and responding to security incidents across the vehicle fleet, and (d) providing safe and secure software updates and ensuring vehicle safety is not compromised, thus introducing a legal basis for so-called over-the-air (OTA) updates to onboard vehicle software. Pre-

cisely, UN R155 mandates the existence of a certified cybersecurity management system, while UN R156 demands a software update management system as a future condition of type approval.

7. Future Trends

Moving one step further from the previous section, the current section identifies trends and forthcoming issues in regard to VIDS technology for the advancement of next-generation automotive electronic systems.

7.1. The Transition to Automotive Ethernet

The increasing use of Ethernet will probably replace existing bus technologies, including FlexRay, MOST, or CAN. Typically, CAN is in charge of controlling the core part of the IVN, while LIN, FlexRay, and MOST serve mainly as auxiliary to the former. It is well known that CAN presents major security issues and other sorts of limitations, including the protocol's broadcast nature, lack of network segmentation, lack of authentication, and lack of data encryption [17], and therefore the great majority of VIDS are intended for CAN (see Section 3). In this respect, ECU consolidation, say, through the use of dedicated domain controllers (i.e., gateways) can be seen as a mechanism to lower the complexity of CAVs and diminish the attack surface. In actuality, the heterogeneous automotive networks of proprietary protocols, such as CAN, are anticipated to be quite soon replaced by hierarchical homogeneous Ethernet networks; due to advances in Ethernet time-sensitive networking (TSN) in terms of bandwidth and cost, it is expected that automotive Ethernet will interconnect all the components in the car [158,159]. Stated simply, while currently CAN has the largest experience and support base from any other bus technology, and despite any extensions such as CAN FD [160] and FlexCAN [161], Ethernet offers improved network speed, bandwidth, built-in security, and native support for TCP/IP. Namely, the IEEE 802.3ch-2020 [162] amendment to the IEEE 802.3-2018 standard [163] has been developed for serving as the network backbone in the vehicle. This standard defines physical layer specifications and management parameters for a single balanced pair of conductors for links of 2.5 Gb/s, 5 Gb/s, and 10 Gb/s for automotive applications. This transition goes hand-in-hand with diagnostics over IP as standardized in ISO 13400 [164], which, in its latest edition, adds support for transport layer security (TLS). Such diagnostics are not limited to, say, emission-related diagnostics or reading-out of relevant data from the computers in the car, but also apply to vehicle manufacturer-specific applications, such as calibration or electronic component software updates. Using an Ethernet backbone for in-vehicle communications renders external communications, say, between a vehicle and the cloud, transparently compatible. This means that vendors rely on the same networking technology across their whole vehicle infrastructure, thus diminishing complexity and enabling both trouble-free OTA software updates and diagnostics-over-IP. On the downside, the shift to automotive Ethernet instantly makes available to the opponent the whole repertoire of legacy Internet attacks in the automotive ecosystem. However, this also means that legacy IDS methodologies and architectures may be more or less applicable to the automotive sector.

7.2. Use of Blockchain Technology

The application of distributed ledger technologies (DLTs) in the vehicular ecosystem could serve to establish a decentralized mediator among different stakeholders to promote the development of trusted and innovate services [165]. Indeed, as already mentioned in the previous section, blockchain could help to keep track of the potential attacks performed over IVN components. In particular, several works have been proposed integrating blockchain in the development of IDS approaches for the vehicular ecosystem. For example, based on the fact that V2X brings along dynamic intrusions where the attacks vary by location and time, while the current vehicle IDSs typically deploy preset static rules, the authors in [166] proposed a micro-blockchain-based dynamic IDS. Precisely, micro-blockchains

are nested into a macro-blockchain, and jointly provide strategies for detecting intrusions. The scheme has each micro-blockchain deployed in a small geographic region with the purpose of generating, in a tamper-resistant manner, local intrusion detection strategies for vehicles. Moreover, macro-blockchains store all the micro-blockchain models and provide dynamic intrusion detection regional strategies for roaming vehicles. For deploying micro-blockchains in the same region, the scheme relies on network slicing. Proof of work (PoW) is used as the consensus algorithm for the macro-blockchain, while the authors evaluated their scheme through simulations. While this is currently the only work that attempts to harness blockchain technology for V2X IDS, it provides a solid background for the design of advanced IDS schemes in the future. Furthermore, a recent work proposes the integration of blockchain and federated learning (see Section 7.4), so that RSUs train cooperatively in a certain area for IVN IDS scenarios. In spite of these efforts, it has been widely recognized that the deployment of blockchain poses important challenges that should be considered in such a context. Indeed, according to [167], it generally presents three main challenges: (a) secure and synchronized software update and validation rules are quite difficult to achieve in blockchain networks, which, for the automotive sector may require the participation of multiple parties/actors; this can be leveraged by an attacker towards exploiting an outdated network or a network that suffers from obsolete validation rules, (b) scalability and high mobility of the blockchain network can possibly affect its overall performance, and (c) blockchain protection against malware is currently not addressed specifically for automotive. Furthermore, most of the works considering blockchain in this context lack a comprehensive evaluation to demonstrate its application in large-scale scenarios.

7.3. Use of Unsupervised ML Techniques

According to Section 3.1, the use of ML techniques represents a clear future trend in the development of VIDS. However, it should be noted that, in most of the cases, the proposed approaches are based on supervised learning techniques. This is aligned with a recent work [148] that provides an exhaustive survey of ML approaches to enhance security aspects in vehicular networks. Indeed, the authors analyze 67 papers; while 35 of the analyzed works are based on supervised learning, only 8 use unsupervised techniques. The main limitation of supervised learning techniques is that they require fully labeled datasets, which may be unfeasible in real scenarios where IVNs could generate a large volume of data on a continuous basis. This aspect is also discussed by recent works [5,7,168,169], which consider the need to foster the use of unsupervised and semisupervised approaches in ML-enabled VIDS. Indeed, based on our analysis in Section 3.1, the works proposed by [27,29] lack an exhaustive evaluation of the unsupervised techniques (based on autoencoders and clustering) for detecting attacks in the CAN bus. Furthermore, [45] evaluates the use of a Kohonen self-organizing map (SOM) network with promising results on a public dataset with several CAN bus attacks. In addition, the work in [74] creates a dataset with several CAN bus attacks that is evaluated by using a long short-term memory (LSTM) and autoencoders. While both approaches present high accuracy scope, still there is the need to evaluate the delay required for the identification of the different attacks, as well as the comparison with other unsupervised techniques. Moreover, in addition to unsupervised approaches, the use of reinforcement learning techniques [170] could also be considered in the vehicular ecosystem, as demonstrated by recent works [171] for detecting misbehaving vehicles as an alternative to the aforementioned works. However, these techniques still have to meet the performance and accuracy requirements of VIDS to be deployed in the vehicular ecosystem.

7.4. Federated Learning Enabled VIDS

As an alternative to the use of traditional centralized ML approaches, federated learning (FL) [172,173] has aroused a significant interest recently from academia and industry [174]. FL provides a key advantage around privacy since the training nodes are

able to create a global model without sharing their data. Specifically, the learning process is carried out through a certain number of training rounds, in which each node updates the parameters of a global model by training on its local data. Then, these parameters are aggregated by a central entity to compute an updated version of the global model, which is shared again with the nodes in each training round. The advantages of FL in the vehicular ecosystem have been highlighted in recent works [175,176], especially in terms of efficiency and privacy. In the case of VIDS, the use of FL allows to build an intrusion detection model while the vehicle's IVN data are not shared. Despite these advantages, the use of FL for VIDS is still in its infancy, and only a few works have been proposed [148]. In particular, [33] proposes a system integrating a federated DL approach with blockchain using the Car-Hacking Dataset (see Section 4). The authors also evaluate the proposed system considering different configurations of malicious nodes. Furthermore, [28] proposes a VIDS for the CAN bus using random forests in a federated scenario where models are shared through the blockchain. However, as described in recent works [177], the use of FL for intrusion detection still has to face different challenges around communication overhead, delay, and scalability, as well as security and privacy aspects, even if training data are not disclosed [178]. These challenges are exacerbated in the vehicular context where the communication channel and network topology are highly dynamic due to nodes' mobility [179], and, consequently, vehicles may join and leave the training process continuously. Therefore, more research efforts are required evaluating the application of FL techniques in vehicular scenarios under real traffic conditions.

7.5. Honeypots and Watchdogs

Honeypots and watchdogs can cooperate with, or be an integral part of, in-vehicle IDS to improve security, and increase the overall vehicle's defense capacity against known or unknown attacks. Recall from Section 6 that adversarial ML assaults against an in-vehicle IDS system may be able to fool the IDS and, generally, any ML-driven component. In this mindset, honeypots and watchdogs can be used for minimizing the available opportunities for the attacker as explained in Section 3.1. However, so far, both these security components are not explored much in the VANET literature. In particular, ref. [180] proposed a cooperative monitoring process in which several watchdogs were intended to obtain and share evidences about vehicles' behavior. Then, the resulting dataset was used as an input for a classification approach based on SVM to detect malicious vehicles. Authors also reduced the overhead of the proposed approach by restricting the data analysis to specific nodes and migrating a subset of tuples between detection iterations. Furthermore, [181] introduced an intelligent watchdog to monitor the behavior of vehicles' ECUs in order to detect potential faults in such components. It is connected to the ECU through a calibration protocol and, in case of detecting abnormal behavior, it can also be used to perform the ECU's operation. Moreover, a recent work called *HoneyCar* [182] integrates game theory and vulnerabilities from the common vulnerability and exposure (CVE) database to compute optimal honeypot configuration strategies in the vehicular ecosystem. While these works demonstrate the potential of using watchdogs/honeypots in such scenarios, it has not received much attention from the research community so far.

7.6. Mobile Edge Computing for VIDS

To address the performance and real-time requirements of the vehicular ecosystem, the deployment of edge-computing-based solutions has been widely considered in recent years by using the concept of vehicular edge computing (VEC) [183]. The main purpose is to increase storage and computing capabilities at the network to allow end nodes (i.e., vehicles) to offload certain tasks into intermediate devices without the need to use cloud nodes, which can incur an increasing latency [184]. VEC is also intended to facilitate a more efficient approach to manage resource allocation in the vehicular environment, which is considered to be extremely challenging due to frequent network topology changes and communication [175]. In the context of VIDS, the use of edge nodes can facilitate the de-

ployment of more efficient approaches by allowing vehicles to offload the training process to RSUs acting as edge nodes [13,18]. Indeed, as described in recent works [179], VEC is considered a key component for the deployment of FL-enabled VIDS and FL in general (see Section 7.4). A potential approach could also be based on vehicles offloading the local training to RSUs, but it could have similar privacy implications to traditional centralized ML approaches. An alternative approach may be based on RSUs acting as the central entity of the FL process by aggregating the model updates calculated by the vehicles themselves using their own local data. In this direction, [185] integrates an edge infrastructure composed of RSUs to build a collaborative intrusion detection model. However, the evaluation does not consider a real vehicle scenario and is based on the obsolete KDDCup99 dataset [186]. Furthermore, [33] uses VEC devices acting as blockchain nodes to enable a federated VIDS approach. As already mentioned for the development of FL-enabled VIDS, the deployment of VEC-based solutions still needs additional research considering traffic scenarios with real conditions to demonstrate their feasibility.

7.7. IVN Security for Future CAVs

As already discussed in Section 6.6, IVNs are currently deployed in environments with limitations in cost, computing capacity, bandwidth, and storage. The evolution of CAVs will eventually lead to new IVN standards. Indeed, such evolution will be realized through an increasing interconnection with vehicles and devices deployed on the roadside composing the so-called Internet of Vehicles (IoV). Therefore, the security concerns of future CAVs will take a broader dimension that needs to address the potential attacks affecting external components, which can be used to launch other attacks over IVNs. As already mentioned by [13], contrary to how CAN was developed, security should be considered in the design phase of these new standards. In fact, standardization activities in the scope of IVNs will be key for the successful deployment of CAVs to come up with a harmonized set of requirements and countermeasures to ensure a more secure vehicular ecosystem. These aspects could also be used to enhance the cybersecurity certification process (see Section 6.9) under a common set of techniques to foster the interoperability of security solutions in such a context.

8. Conclusions

Transportation is one of the critical infrastructure sectors which are necessary to maintain normalcy in everyday life. Under the prism of the IoV, and, more generally, the Internet of Everything (IoE), security and privacy issues become imperative due to increased machine-to-machine connectivity, interoperability, and communication requirements. As stated in [187], in the 1950s, automotive electronics cost only 1% of the total vehicle expenditure, while this percentage is expected to reach 50% in 2030. However, this steep augmentation in the electronic components goes hand in hand with an increased attack surface, and new threats and vulnerabilities.

Specifically, modern vehicles rely on a diverse collection of digital components and technologies to fulfill their mission. Excluding legacy sensors and actuators, which lie at the physical layer, such components include artificial intelligence, ML, backend (cloud-based) systems, mobile apps, and wireless technologies. All these ICT components and technologies bring along their own attack surface, which is ultimately added to that of the vehicle. It becomes therefore clear that the complexity of modern CAVs creates a large, complex, and continuously expanded attack surface, which can potentially be exploited by different kinds of malicious actors in a plethora of ways. From a 10,000-foot view, and from a defender's perspective, one can classify CAV security in two broad axes; the first refers to in-vehicle, that is, security measures implemented within the vehicle, while the second concentrates to inter-vehicle, namely, the wireless communications with external entities of the cooperative intelligent transport system (C-ITS) through internal interfaces. Both these perspectives apply to CAV security from an offensive viewpoint, with attacks exercised

against or via the exploitation of in-vehicle or inter-vehicle components or interfaces, respectively.

This work focuses on the first abovementioned axis, and more particularly on IVN IDS. After studying the related literature, we realized that, so far, no work addresses this matter in a full-fledged way. Namely, while there exists a critical mass of surveys in this ecosystem, none of them tackle the four key angles of this subject in a holistic way: (a) the provision of a unified, overarching taxonomy that can be used to classify IVN IDS, (b) the available datasets that can be used to train and evaluate a given IVN IDS, (c) the non-commercial simulators that may be exploited for either creating datasets or testing an IDS prior to its deployment, and (d) the gathering and analysis of both the future trends and challenges in this area in an exhaustive manner. With this goal in mind, the current paper is an attempt to not just collect and quote in a sterile manner the results of the relevant work in the literature, but to serve as a comprehensive survey of surveys that can be used by a diverse audience, including researchers, practitioners, and policymakers. In this respect, to our knowledge, the work at hand is the first to offer a unified, but at the same time quad-dimensional, simple taxonomy of IVN IDS to be used as a basis and point of reference in future work in this topic. Overall, we hope this work will shed more light on this fast-paced, vivid, and interesting research branch and serve as a solid starting point for the interested readers.

Author Contributions: Conceptualization, G.K. (Georgios Kambourakis); methodology, G.K. (Georgios Kambourakis), G.K. (Georgios Karopoulos), and J.L.H.-R.; investigation, G.K. (Georgios Karopoulos), G.K. (Georgios Kambourakis), E.C., J.L.H.-R., and V.K.; resources, G.K. (Georgios Karopoulos), G.K. (Georgios Kambourakis), E.C., J.L.H.-R., and V.K.; writing—original draft preparation, G.K. (Georgios Karopoulos), G.K. (Georgios Kambourakis), E.C., J.L.H.-R., and V.K.; writing—review and editing, G.K. (Georgios Karopoulos), G.K. (Georgios Kambourakis), E.C., J.L.H.-R., and V.K.; supervision, G.K. (Georgios Kambourakis). All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: All data are available in the manuscript.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

ADAS	Advanced Driver-Assistance System
AVTP	Audio-Video Transport Protocol
AVTPDU	AVTP Data Units
BSM	Basic Safety Message
CAN	Controller Area Network
CAV	Connected and Autonomous Vehicle
CC	Common Criteria
C-ITS	Cooperative Intelligent Transport System
CSV	Comma-separated values
DNN	Deep Neural Network
DoS	Denial of Service
DSRC	Dedicated Short-Range Communications
ECU	Electronic Control Unit
FL	Federated Learning
GDPR	General Data Protection Regulation
GPS	Global Positioning System
HIDS	Host-based IDS
ICT	Information Communication Technologies
IDPS	Intrusion Detection and Prevention System
IDS	Intrusion Detection System
IoV	Internet of Vehicles

ISO	International Organization for Standardization
iTETRIS	Integrated Wireless and Traffic Platform for Real-Time Road Traffic Management Solutions
IVC	Inter-Vehicular Communication
IVN	In-Vehicle Network
JSON	JavaScript Object Notation
LIDAR	LIght Detection And Ranging
LIN	Local Interconnect Network
ML	Machine Learning
MOST	Media Oriented Systems Transport
NIDS	Network-based
NR	New Radio
OBD	Onboard Diagnostics
OBU	Onboard Unit
OCTANE	Open Car Test-bed and Network Experiments
OEM	Original Equipment Manufacturer
OTA	Over-the-Air
PMF	Protected Management Frames
RPM	Revolutions Per Minute
RSU	Roadside Unit
RTOS	Real-time operating system
TSN	Time-Sensitive Networking
UAV	Unmanned Aerial Vehicle
UN	United Nations
UNECE	United Nations Economic Commission for Europe
V2X	Vehicle-to-Everything
VANET	Vehicular ad hoc Network
VEINS	Vehicles In-Network Simulation
VeReMi	Vehicular Reference Misbehavior Dataset
VIDS	Vehicular IDS

References

1. Ghosal, A.; Conti, M. Security issues and challenges in V2X: A survey. *Comput. Netw.* **2020**, *169*, 107093. [CrossRef]
2. Makowitz, R.; Temple, C. Flexray-a communication network for automotive control systems. In Proceedings of the 2006 IEEE International Workshop on Factory Communication Systems, Turin, Italy, 28–30 June 2006; pp. 207–212.
3. Ruff, M. Evolution of local interconnect network (LIN) solutions. In Proceedings of the 2003 IEEE 58th Vehicular Technology Conference. VTC 2003-Fall (IEEE Cat. No. 03CH37484), Orlando, FL, USA, 6–9 October 2003; Volume 5, pp. 3382–3389.
4. Cooperation, M. MOST-Media Oriented Systems Transport. *MOST Specif. Rev.* **2005**, *3*, E2.
5. Al-Jarrah Y., O.; Maple, C.; Dianati, M.; Oxtoby, D.; Mouzakitis, A. Intrusion Detection Systems for Intra-Vehicle Networks: A Review. *IEEE Access* **2019**, *7*, 21266–21289. [CrossRef]
6. Young, C.; Zambreno, J.; Olufowobi, H.; Bloom, G. Survey of Automotive Controller Area Network Intrusion Detection Systems. *IEEE Des. Test* **2019**, *36*, 48–55. [CrossRef]
7. Lokman, S.F.; Othman, A.; Husaini, M. Intrusion detection system for automotive Controller Area Network (CAN) bus system: A review. *EURASIP J. Wirel. Commun. Netw.* **2019**, *2019*, 184. [CrossRef]
8. Xie, Y.; Zhou, Y.; Xu, J.; Zhou, J.; Chen, X.; Xiao, F. Cybersecurity protection on in-vehicle networks for distributed automotive cyber-physical systems: State-of-the-art and future challenges. *Softw. Pract. Exp.* **2021**, *51*, 2108–2127. [CrossRef]
9. Sharma, S.; Kaul, A. A survey on Intrusion Detection Systems and Honeypot based proactive security mechanisms in VANETs and VANET Cloud. *Veh. Commun.* **2018**, *12*, 138–164. [CrossRef]
10. Palaniswamy, B.; Camtepe, S.; Foo, E.; Pieprzyk, J. An Efficient Authentication Scheme for Intra-Vehicular Controller Area Network. *IEEE Trans. Inf. Forensics Secur.* **2020**, *15*, 3107–3122. [CrossRef]
11. Hu, S.; Chen, Q.A.; Joung, J.; Carlak, C.; Feng, Y.; Mao, Z.M.; Liu, H.X. CVShield: Guarding Sensor Data in Connected Vehicle with Trusted Execution Environment. In Proceedings of the AutoSec@CODASPY '20: Proceedings of the Second ACM Workshop on Automotive and Aerial Vehicle Security, New Orleans, LA, USA, 18 March 2020; Chen, Q.A., Zhao, Z., Ahn, G., Eds.; ACM: New York, NY, USA, 2020; pp. 1–4. [CrossRef]
12. Dibaei, M.; Zheng, X.; Jiang, K.; Abbas, R.; Liu, S.; Zhang, Y.; Xiang, Y.; Yu, S. Attacks and defences on intelligent connected vehicles: A survey. *Digit. Commun. Netw.* **2020**, *6*, 399–421. [CrossRef]
13. Wu, W.; Li, R.; Xie, G.; An, J.; Bai, Y.; Zhou, J.; Li, K. A Survey of Intrusion Detection for In-Vehicle Networks. *IEEE Trans. Intell. Transp. Syst.* **2020**, *21*, 919–933. [CrossRef]

14. Aliwa, E.; Rana, O.; Perera, C.; Burnap, P. Cyberattacks and countermeasures for in-vehicle networks. *ACM Comput. Surv.* **2021**, *54*, 1–37. [CrossRef]
15. Hafeez, A.; Rehman, K.; Malik, H. State of the Art Survey on Comparison of Physical Fingerprinting-Based Intrusion Detection Techniques for In-Vehicle Security. *SAE Int.* **2020**, 7. [CrossRef]
16. Dupont, G.; den Hartog, J.; Etalle, S.; Lekidis, A. A survey of network intrusion detection systems for controller area network. In Proceedings of the 2019 IEEE International Conference on Vehicular Electronics and Safety (ICVES), Cairo, Egypt, 4–6 September 2019; pp. 1–6. [CrossRef]
17. Gmiden, M.; Gmiden, M.H.; Trabelsi, H. Cryptographic and Intrusion Detection System for automotive CAN bus: Survey and contributions. In Proceedings of the 2019 16th International Multi-Conference on Systems, Signals Devices (SSD), Istanbul, Turkey, 21–24 March 2019; pp. 158–163. [CrossRef]
18. Loukas, G.; Karapistoli, E.D.; Panaousis, E.A.; Sarigiannidis, P.G.; Bezemskij, A.; Vuong, T. A taxonomy and survey of cyber-physical intrusion detection approaches for vehicles. *Ad Hoc Netw.* **2019**, *84*, 124–147. [CrossRef]
19. Tomlinson, A.; Bryans, J.; Shaikh, S.A. Towards viable intrusion detection methods for the automotive controller area network. In Proceedings of the 2nd ACM Computer Science in Cars Symposium, Munich, Germany, 13–14 September 2018; pp. 1–9.
20. Liu, J.; Zhang, S.; Sun, W.; Shi, Y. In-Vehicle Network Attacks and Countermeasures: Challenges and Future Directions. *IEEE Netw.* **2017**, *31*, 50–58. [CrossRef]
21. Islam, M.; Sandberg, C.; Bokesand, A.; Olovsson, T.; Kleberger, P.; Lautenbach, A.; Söderberg-Rivkin, A.; Kadhirvelan, S.P.; Hansson, A.; Broberg, H. Deliverable D2: Security Models (Version 2.0), Vinnova/FFI (Fordonsutveckling/Vehicle Development). 2016. Available online: https://autosec.se/wp-content/uploads/2018/03/HEAVENS_D2_v2.0.pdf (accessed on 3 March 2022).
22. Hortelano, J.; Ruiz, J.C.; Manzoni, P. Evaluating the Usefulness of Watchdogs for Intrusion Detection in VANETs. In Proceedings of the 2010 IEEE International Conference on Communications Workshops, Cape Town, South Africa, 23–27 May 2010; pp. 1–5. [CrossRef]
23. Basavaraj, D.; Tayeb, S. Towards a Lightweight Intrusion Detection Framework for In-Vehicle Networks. *J. Sens. Actuator Netw.* **2022**, *11*, 6. [CrossRef]
24. Islam, R.; Devnath, M.K.; Samad, M.D.; Al Kadry, S.M.J. GGNB: Graph-based Gaussian naive Bayes intrusion detection system for CAN bus. *Veh. Commun.* **2022**, *33*, 100442. [CrossRef]
25. Han, M.; Cheng, P.; Ma, S. PPM-InVIDS: Privacy protection model for in-vehicle intrusion detection system based complex-valued neural network. *Veh. Commun.* **2021**, *31*, 100374. [CrossRef]
26. He, Y.; Jia, Z.; Hu, M.; Cui, C.; Cheng, Y.; Yang, Y. The Hybrid Similar Neighborhood Robust Factorization Machine Model for Can Bus Intrusion Detection in the In-Vehicle Network. Available online: https://ieeexplore.ieee.org/document/9557759 (accessed on 3 March 2022).
27. Narasimhan, H.; Vinayakumar, R.; Mohammad, N. Unsupervised Deep Learning Approach for In-Vehicle Intrusion Detection System. Available online: https://ieeexplore.ieee.org/document/9555398 (accessed on 3 March 2022). [CrossRef]
28. Aliyu, I.; Feliciano, M.C.; Van Engelenburg, S.; Kim, D.O.; Lim, C.G. A blockchain-based federated forest for SDN-enabled in-vehicle network intrusion detection system. *IEEE Access* **2021**, *9*, 102593–102608. [CrossRef]
29. Leslie, N. An Unsupervised Learning Approach for In-Vehicle Network Intrusion Detection. In Proceedings of the 2021 55th Annual Conference on Information Sciences and Systems (CISS), Baltimore, MA, USA, 24–26 March 2021; pp. 1–4.
30. Sun, H.; Chen, M.; Weng, J.; Liu, Z.; Geng, G. Anomaly Detection for In-Vehicle Network Using CNN-LSTM With Attention Mechanism. *IEEE Trans. Veh. Technol.* **2021**, *70*, 10880–10893. [CrossRef]
31. Rehman, A.; Rehman, S.U.; Khan, M.; Alazab, M.; Reddy, T. CANintelliIDS: Detecting in-vehicle intrusion attacks on a controller area network using CNN and attention-based GRU. *IEEE Trans. Netw. Sci. Eng.* **2021**, *8*, 1456–1466.
32. Ahmed, I.; Ahmad, A.; Jeon, G. Deep Learning-based Intrusion Detection System for Internet of Vehicles. *IEEE Consum. Electron. Mag.* Available online: https://ieeexplore.ieee.org/document/9665273 (accessed on 3 March 2022).
33. Abdel-Basset, M.; Moustafa, N.; Hawash, H.; Razzak, I.; Sallam, K.M.; Elkomy, O.M. Federated Intrusion Detection in Blockchain-Based Smart Transportation Systems. *IEEE Trans. Intell. Transp. Syst.* **2021**, *23*, 2523–2537. [CrossRef]
34. Derhab, A.; Belaoued, M.; Mohiuddin, I.; Kurniawan, F.; Khan, M.K. Histogram-Based Intrusion Detection and Filtering Framework for Secure and Safe In-Vehicle Networks. *IEEE Trans. Intell. Transp. Syst.* **2021**, *23*, 2366–2379. [CrossRef]
35. Shi, D.; Xu, M.; Wu, T.; Kou, L. Intrusion Detecting System Based on Temporal Convolutional Network for In-Vehicle CAN Networks. *Mob. Inf. Syst.* **2021**, *2021*, 1440259. [CrossRef]
36. Nam, M.; Park, S.; Kim, D.S. Intrusion detection method using bi-directional GPT for in-vehicle controller area networks. *IEEE Access* **2021**, *9*, 124931–124944. [CrossRef]
37. Chen, M.; Zhao, Q.; Jiang, Z.; Xu, R. Intrusion Detection for in-vehicle CAN Networks Based on Auxiliary Classifier GANs. In Proceedings of the 2021 International Conference on High Performance Big Data and Intelligent Systems (HPBD&IS), Macau, China, 5–7 December 2021; pp. 186–191.
38. Sharmin, S.; Mansor, H. Intrusion Detection on the In-Vehicle Network Using Machine Learning. In Proceedings of the 2021 3rd International Cyber Resilience Conference (CRC), Virtual Conference, 29–31 January 2021; pp. 1–6.
39. Alfardus, A.; Rawat, D.B. Intrusion Detection System for CAN Bus In-Vehicle Network based on Machine Learning Algorithms. In Proceedings of the 2021 IEEE 12th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON), New York, NY, USA, 1 December 2021; pp. 0944–0949.

40. Baldini, G. Intrusion detection systems in in-vehicle networks based on bag-of-words. In Proceedings of the 2021 5th Cyber Security in Networking Conference (CSNet), Abu Dhabi, United Arab Emirates, 12–14 October 2021; pp. 41–48.
41. Fenzl, F.; Rieke, R.; Dominik, A. In-vehicle detection of targeted CAN bus attacks. In Proceedings of the 16th International Conference on Availability, Reliability and Security, Vienna, Austria, 17–20 August 2021; pp. 1–7.
42. Duan, X.; Yan, H.; Tian, D.; Zhou, J.; Su, J.; Hao, W. In-Vehicle CAN Bus Tampering Attacks Detection for Connected and Autonomous Vehicles Using an Improved Isolation Forest Method. *IEEE Trans. Intell. Transp. Syst.* **2021**. Available online: https://ieeexplore.ieee.org/abstract/document/9652038 (accessed on 7 March 2022).
43. Song, H.M.; Woo, J.; Kim, H.K. In-vehicle network intrusion detection using deep convolutional neural network. *Veh. Commun.* **2020**, *21*, 100198. [CrossRef]
44. Kalkan, S.C.; Sahingoz, O.K. In-Vehicle Intrusion Detection System on Controller Area Network with Machine Learning Models. In Proceedings of the 2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT), Kharagpur, India, 1–3 July 2020; pp. 1–6.
45. Barletta, V.S.; Caivano, D.; Nannavecchia, A.; Scalera, M. Intrusion detection for in-vehicle communication networks: An unsupervised kohonen som approach. *Future Internet* **2020**, *12*, 119. [CrossRef]
46. Park, S.; Choi, J.Y. Hierarchical anomaly detection model for in-vehicle networks using machine learning algorithms. *Sensors* **2020**, *20*, 3934. [CrossRef] [PubMed]
47. Hossain, M.D.; Inoue, H.; Ochiai, H.; Fall, D.; Kadobayashi, Y. LSTM-based intrusion detection system for in-vehicle can bus communications. *IEEE Access* **2020**, *8*, 185489–185502. [CrossRef]
48. Desta, A.K.; Ohira, S.; Arai, I.; Fujikawa, K. MLIDS: Handling Raw High-Dimensional CAN Bus Data Using Long Short-Term Memory Networks for Intrusion Detection in In-Vehicle Networks. In Proceedings of the 2020 30th International Telecommunication Networks and Applications Conference (ITNAC), Melbourne, Australia, 25–27 November, 2020; pp. 1–7.
49. Hossain, M.D.; Inoue, H.; Ochiai, H.; Fall, D.; Kadobayashi, Y. An Effective In-Vehicle CAN Bus Intrusion Detection System Using CNN Deep Learning Approach. In Proceedings of the GLOBECOM 2020-2020 IEEE Global Communications Conference, Taipei, Taiwan, 7–11 December 2020; pp. 1–6.
50. Lin, Y.; Chen, C.; Xiao, F.; Avatefipour, O.; Alsubhi, K.; Yunianta, A. An Evolutionary Deep Learning Anomaly Detection Framework for In-Vehicle Networks-CAN Bus. *IEEE Trans. Ind. Appl.* **2020**. Available online: https://ieeexplore.ieee.org/document/9143513 (accessed on 7 March 2022). [CrossRef]
51. Levy, E.; Shabtai, A.; Groza, B.; Murvay, P.S.; Elovici, Y. CAN-LOC: Spoofing Detection and Physical Intrusion Localization on an In-Vehicle CAN Bus Based on Deep Features of Voltage Signals. *arXiv* **2021**, arXiv:2106.07895.
52. Moulahi, T.; Zidi, S.; Alabdulatif, A.; Atiquzzaman, M. Comparative Performance Evaluation of Intrusion Detection Based on Machine Learning in In-Vehicle Controller Area Network Bus. *IEEE Access* **2021**, *9*, 99595–99605. [CrossRef]
53. Zhang, X.; Cui, X.; Cheng, K.; Zhang, L. A Convolutional Encoder Network for Intrusion Detection in Controller Area Networks. In Proceedings of the 2020 16th International Conference on Computational Intelligence and Security (CIS), Guangxi, China, 27–30 November 2020; pp. 366–369.
54. Alkhatib, N.; Mushtaq, M.; Ghauch, H.; Danger, J.L. AVTPnet: Convolutional Autoencoder for AVTP Anomaly Detection in Automotive Ethernet Networks. 2022. Available online: http://xxx.lanl.gov/abs/2202.00045 (accessed on 20 March 2022).
55. Jeong, S.; Jeon, B.; Chung, B.; Kim, H.K. Convolutional neural network-based intrusion detection system for AVTP streams in automotive Ethernet-based networks. *Veh. Commun.* **2021**, *29*, 100338. [CrossRef]
56. Bozdal, M.; Samie, M.; Jennions, I.K. WINDS: A Wavelet-Based Intrusion Detection System for Controller Area Network (CAN). *IEEE Access* **2021**, *9*, 58621–58633. [CrossRef]
57. Linghu, Y.; Li, X. WSG-InV: Weighted State Graph Model for Intrusion Detection on In-Vehicle Network. In Proceedings of the 2021 IEEE Wireless Communications and Networking Conference (WCNC), Nanjing, China, 29 March–1 April 2021; pp. 1–7.
58. Cheng, A.; Peng, Y.; Yan, H.; Shen, X. An intrusion detection method for the in-vehicle network. In Proceedings of the 2021 33rd Chinese Control and Decision Conference (CCDC), Kunming, China, 22–24 May 2021; pp. 4893–4899.
59. Pascale, F.; Adinolfi, E.A.; Coppola, S.; Santonicola, E. Cybersecurity in Automotive: An Intrusion Detection System in Connected Vehicles. *Electronics* **2021**, *10*, 1765. [CrossRef]
60. Jiang, W.; Li, Z.; Tan, K.; Guan, Y.; Tong, W. *An Adaptive Intrusion Detection Algorithm for In-vehicle CAN Bus Based on Periodicity of Message*; Journal of Physics: Conference Series; IOP Publishing: Bristol, UK, 2021; Volume 1748, p. 032023.
61. Baldini, G. On the application of entropy measures with sliding window for intrusion detection in automotive in-vehicle networks. *Entropy* **2020**, *22*, 1044. [CrossRef]
62. Halder, S.; Conti, M.; Das, S.K. A holistic approach to power efficiency in a clock offset based Intrusion Detection Systems for Controller Area Networks. *Pervasive Mob. Comput.* **2021**, *73*, 101385. [CrossRef]
63. Zihan, Z.; Lirong, C.; Haitao, Z.; Fan, Z. Research on Intrusion Detection Technology Based on Embedded Ethernet. In Proceedings of the 2021 18th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP), Chengdu, China, 17–19 December 2021; pp. 587–600. [CrossRef]
64. Tian, M.; Jiang, R.; Qu, H.; Lu, Q.; Zhou, X. Advanced Temperature-Varied ECU Fingerprints for Source Identification and Intrusion Detection in Controller Area Networks. *Secur. Commun. Netw.* **2020**, *2020*, 8834845. [CrossRef]
65. Jin, S.; Chung, J.G.; Xu, Y. Signature-Based Intrusion Detection System (IDS) for In-Vehicle CAN Bus Network. In Proceedings of the 2021 IEEE International Symposium on Circuits and Systems (ISCAS), Daegu, Korea, 22–28 May 2021; pp. 1–5.

66. Yang, L.; Moubayed, A.; Shami, A. MTH-IDS: A Multitiered Hybrid Intrusion Detection System for Internet of Vehicles. *IEEE Internet Things J.* **2022**, *9*, 616–632. [CrossRef]
67. Bangui, H.; Buhnova, B. Recent Advances in Machine-Learning Driven Intrusion Detection in Transportation: Survey. In Proceedings of the 11th International Symposium on Frontiers in Ambient and Mobile Systems (FAMS), Procedia Computer Science, Warsaw, Poland, 23–26 March 2021; Volume 184, pp. 877–886. doi: doi: 10.1016/j.procs.2021.04.014 [CrossRef]
68. Kang, M.J.; Kang, J.W. Intrusion Detection System Using Deep Neural Network for In-Vehicle Network Security. *PLoS ONE* **2016**, *11*, 1–17. [CrossRef] [PubMed]
69. Lee, H.; Jeong, S.H.; Kim, H.K. OTIDS: A Novel Intrusion Detection System for In-vehicle Network by Using Remote Frame. In Proceedings of the 15th Annual Conference on Privacy, Security and Trust, PST 2017, Calgary, AB, Canada, 28–30 August 2017; IEEE Computer Society: Calgary, AB, Canada; pp. 57–66. [CrossRef]
70. Seo, E.; Song, H.M.; Kim, H.K. GIDS: GAN based Intrusion Detection System for In-Vehicle Network. In Proceedings of the 2018 16th Annual Conference on Privacy, Security and Trust (PST), Belfast, Ireland, 28–30 August 2018; pp. 1–6. [CrossRef]
71. Han, M.L.; Kwak, B.I.; Kim, H.K. Anomaly intrusion detection method for vehicular networks based on survival analysis. *Veh. Commun.* **2018**, *14*, 52–63. [CrossRef]
72. Sami, M. Intrusion Detection in CAN Bus 2019. Available online: https://ieee-dataport.org/documents/intrusion-detection-can-bus (accessed on 3 March 2022).
73. Dupont, G.; Lekidis, A.; den Hartog, J.J.; Etalle, S.S. Automotive Controller Area Network (CAN) Bus Intrusion Dataset v2 2019. Available online: https://doi.org/10.4121/uuid:b74b4928-c377-4585-9432-2004dfa20a5d (accessed on 3 March 2022).
74. Hanselmann, M.; Strauss, T.; Dormann, K.; Ulmer, H. CANet: An Unsupervised Intrusion Detection System for High Dimensional CAN Bus Data. *IEEE Access* **2020**, *8*, 58194–58205. [CrossRef]
75. Kang, H.; Kwak, B.I.; Lee, Y.H.; Lee, H.; Lee, H.; Kim, H.K. Car Hacking: Attack & Defense Challenge 2020 Dataset. Available online: https://ieee-dataport.org/open-access/car-hacking-attack-defense-challenge-2020-dataset (accessed on 3 March 2022).
76. He, Q.; Meng, X.; Qu, R.; Xi, R. Machine Learning-Based Detection for Cyber Security Attacks on Connected and Autonomous Vehicles. *Mathematics* **2020**, *8*, 1311. [CrossRef]
77. Kolias, C.; Kambourakis, G.; Stavrou, A.; Gritzalis, S. Intrusion Detection in 802.11 Networks: Empirical Evaluation of Threats and a Public Dataset. *IEEE Commun. Surv. Tutor.* **2016**, *18*, 184–208. [CrossRef]
78. Chatzoglou, E.; Kambourakis, G.; Kolias, C. Empirical Evaluation of Attacks Against IEEE 802.11 Enterprise Networks: The AWID3 Dataset. *IEEE Access* **2021**, *9*, 34188–34205. [CrossRef]
79. Whelan, J.; Sangarapillai, T.; Minawi, O.; Almehmadi, A.; El-Khatib, K. UAV Attack Dataset 2020. Available online: https://ieee-dataport.org/open-access/uav-attack-dataset (accessed on 3 March 2022).
80. Unal, D. BlueTack 2021. Available online: https://ieee-dataport.org/documents/bluetack (accessed on 3 March 2022).
81. van der Heijden, R.W.; Lukaseder, T.; Kargl, F. VeReMi: A Dataset for Comparable Evaluation of Misbehavior Detection in VANETs. In *Security and Privacy in Communication Networks*; Beyah, R., Chang, B., Li, Y., Zhu, S., Eds.; Springer International Publishing: Cham, Switzerland, 2018; pp. 318–337.
82. Kamel, J.; Wolf, M.; van der Hei, R.W.; Kaiser, A.; Urien, P.; Kargl, F. VeReMi Extension: A Dataset for Comparable Evaluation of Misbehavior Detection in VANETs. In Proceedings of the ICC 2020–2020 IEEE International Conference on Communications (ICC), Dublin, Ireland, 7–11 June 2020; pp. 1–6. [CrossRef]
83. Borazjani, P.; Everett, C.; McCoy, D. Octane: An extensible open source car security testbed. In Proceedings of the Embedded Security in Cars Conference 2014, Hamburg, Germany, 18–19 November 2014; Volume 40.
84. Lee, H.; Jeong, S.H.; Kim, H.K. CAN Dataset for Intrusion Detection (OTIDS). Available online: https://ocslab.hksecurity.net/Dataset/CAN-intrusion-dataset (accessed on 3 March 2022).
85. Jeong, S.; Jeon, B.; Chung, B.; Kim, H.K. Automotive Ethernet Intrusion Dataset 2021. Available online: https://ieee-dataport.org/open-access/automotive-ethernet-intrusion-dataset (accessed on 3 March 2022).
86. Whelan, J.; Sangarapillai, T.; Minawi, O.; Almehmadi, A.; El-Khatib, K. Novelty-Based Intrusion Detection of Sensor Attacks on Unmanned Aerial Vehicles. In Proceedings of the Q2SWinet '20: 16th ACM Symposium on QoS and Security for Wireless and Mobile Networks, Alicante, Spain, 16–20 November 2020; Association for Computing Machinery: New York, NY, USA, 2020; pp. 23–28. [CrossRef]
87. Alhomayani, F.; Mahoor, M.H. OutFin, a multi-device and multi-modal dataset for outdoor localization based on the fingerprinting approach. *Nat. Sci. Data* **2021**, *8*, 66. [CrossRef]
88. Li, Y.; Peng, C.; Yuan, Z.; Li, J.; Deng, H.; Wang, T. *Mobileinsight: Extracting and Analyzing Cellular Network Information on Smartphones*; MobiCom '16; Association for Computing Machinery: New York, NY, USA, 2016; pp. 202–215. [CrossRef]
89. Codeca, L. GitHub Project—The Luxembourg SUMO Traffic (LuST) Scenario. Available online: https://github.com/lcodeca/LuSTScenario (accessed on 3 March 2022).
90. TETCOS. NetSim Network Simulator. Available online: https://www.boson.com/netsim-cisco-network-simulator (accessed on 3 March 2022).
91. Wang, S.Y.; Chou, C.L.; Yang, C.M. EstiNet openflow network simulator and emulator. *IEEE Commun. Mag.* **2013**, *51*, 110–117. [CrossRef]
92. Vector Informatik GmbH. Testing ECUs and Networks With CANoe. 2022. Available online: https://www.vector.com/int/en/products/products-a-z/software/canoe (accessed on 3 March 2022).

93. Roscher, K.; Bittl, S.; Gonzalez, A.; Myrtus, M.; Jiru, J. ezCar2X. Rapid-Prototyping of Communication Technologies and Cooperative ITS Applications on Real Targets and Inside Simulation Environments. In Proceedings of the 8th International Workshop on Communication Technologies for Vehicles, Nets4Cars/Nets4Trains/Nets4Aircraft, Sousse, Tunisia, 6–8 May 2015; Volume 9066; pp. 1–6.
94. Weber, J.S.; Neves, M.; Ferreto, T. VANET simulators: An updated review. *J. Braz. Comput. Soc.* **2021**, *27*, 1–31. [CrossRef]
95. Sommer, C.; German, R.; Dressler, F. Bidirectionally Coupled Network and Road Traffic Simulation for Improved IVC Analysis. *IEEE Trans. Mob. Comput.* **2011**, *10*, 3–15. [CrossRef]
96. Riebl, R.; Monz, M.; Varga, S.; Maglaras, L.; Janicke, H.; Al-Bayatti, A.H.; Facchi, C. Improved Security Performance for VANET Simulations. *IFAC-PapersOnLine* **2016**, *49*, 233–238. [CrossRef]
97. Kamel, J.; Ansari, M.R.; Petit, J.; Kaiser, A.; Jemaa, I.B.; Urien, P. Simulation Framework for Misbehavior Detection in Vehicular Networks. *IEEE Trans. Veh. Technol.* **2020**, *69*, 6631–6643. [CrossRef]
98. Emara, K. Poster: PREXT: Privacy extension for Veins VANET simulator. In Proceedings of the 2016 IEEE Vehicular Networking Conference, VNC 2016, Columbus, OH, USA, 8–10 December 2016; pp. 1–2. [CrossRef]
99. CLPS: Context-Based Location Privacy Scheme for VANETs. *Int. J. AD HOC Ubiquitous Comput.* **2018**, *29*, 141–159. [CrossRef]
100. Riebl, R.; Günther, H.J.; Facchi, C.; Wolf, L. Artery: Extending Veins for VANET applications. In Proceedings of the 2015 International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS), Budapest, Hungary, 3–5 June 2015; pp. 450–456. [CrossRef]
101. Amoozadeh, M.; Ching, B.; Chuah, C.N.; Ghosal, D.; Zhang, H.M. VENTOS: Vehicular Network Open Simulator with Hardware-in-the-Loop Support. *Procedia Comput. Sci.* **2019**, *151*, 61–68. [CrossRef]
102. Amoozadeh, M.; Raghuramu, A.; Chuah, C.N.; Ghosal, D.; Zhang, H.M.; Rowe, J.; Levitt, K. Security vulnerabilities of connected vehicle streams and their impact on cooperative driving. *IEEE Commun. Mag.* **2015**, *53*, 126–132. [CrossRef]
103. Kan, X.; Ganlath, A.; Ucar, S.; Han, K.; Tiwari, P.; Karydis, K. Edge Assisted Misbehavior Detection for Platoons. In Proceedings of the 2019 IEEE Vehicular Networking Conference (VNC), Los Angeles, CA, USA, 4–6 December 2019; pp. 1–4. [CrossRef]
104. Ucar, S.; Ergen, S.C.; Ozkasap, O. Data-driven abnormal behavior detection for autonomous platoon. In Proceedings of the 2017 IEEE Vehicular Networking Conference (VNC), Turin, Italy, 27–29 November 2017; pp. 69–72. [CrossRef]
105. Rondinone, M.; Maneros, J.; Krajzewicz, D.; Bauza, R.; Cataldi, P.; Hrizi, F.; Gozalvez, J.; Kumar, V.; Röckl, M.; Lin, L.; et al. iTETRIS: A modular simulation platform for the large scale evaluation of cooperative ITS applications. *Simul. Model. Pract. Theory* **2013**, *34*, 99–125. [CrossRef]
106. Dosovitskiy, A.; Ros, G.; Codevilla, F.; Lopez, A.; Koltun, V. CARLA: An Open Urban Driving Simulator. In Proceedings of the 1st Annual Conference on Robot Learning, Mountain View, CA, USA, 13–15 November 2017; pp. 1–16.
107. Lee, T.K.; Wang, T.W.; Wu, W.X.; Kuo, Y.C.; Huang, S.H.; Wang, G.S.; Lin, C.Y.; Chen, J.J.; Tseng, Y.C. Building a V2X Simulation Framework for Future Autonomous Driving. In Proceedings of the 2019 20th Asia-Pacific Network Operations and Management Symposium (APNOMS), Matsue, Japan, 18–20 September 2019; pp. 1–6. [CrossRef]
108. Shah, S.; Dey, D.; Lovett, C.; Kapoor, A. AirSim: High-Fidelity Visual and Physical Simulation for Autonomous Vehicles. In Proceedings of the Field and Service Robotics, Zurich, Switzerland, 12–15 September 2017. Available online: http://xxx.lanl.gov/abs/arXiv:1705.05065 (accessed on 3 March 2022).
109. Tomandl, A.; Herrmann, D.; Fuchs, K.P.; Federrath, H.; Scheuer, F. VANETsim: An open source simulator for security and privacy concepts in VANETs. In Proceedings of the 2014 International Conference on High Performance Computing Simulation (HPCS), Bologna, Italy, 21–25 July 2014; pp. 543–550. [CrossRef]
110. Raphael, R.; Christina, O.; Stefan, N.; Christian, F. Vanetza: Boosting Research on Inter-Vehicle Communication. In Proceedings of the 5th GI/ITG KuVS Fachgespraach Inter-Vehicle Communication (FG-IVC 2017), Erlangen, Germany, 6–7 April 2017; Anatoli, D., Kai-Steffen, H., Reinhard, G., Eds.; Erlangen-Nürnberg: Erlangen, Germany, 2017; pp. 37–40.
111. Egea-Lopez, E.; Losilla, F.; Pascual-Garcia, J.; Molina-Garcia-Pardo, J.M. Vehicular Networks Simulation with Realistic Physics. *IEEE Access* **2019**, *7*, 44021–44036. [CrossRef]
112. Wang, S.Y.; Lin, C.C. NCTUns 6.0: A Simulator for Advanced Wireless Vehicular Network Research. In Proceedings of the 2010 IEEE 71st Vehicular Technology Conference, Taipei, Taiwan, 16–19 May 2010; pp. 1–2. [CrossRef]
113. Bakar, K.A.A.; Irvine, J. A Scheme for Detecting Selfish Nodes in MANETs Using OMNET++. In Proceedings of the 2010 6th International Conference on Wireless and Mobile Communications, Valencia, Spain, 20–25 September 2010; pp. 410–414. [CrossRef]
114. Achour, I.; Bejaoui, T.; Busson, A.; Tabbane, S. A Redundancy-Based Protocol for Safety Message Dissemination in Vehicular Ad Hoc Networks. In Proceedings of the 2015 IEEE 82nd Vehicular Technology Conference (VTC2015-Fall), Boston, MA, USA, 6–9 September 2015; pp. 1–6. [CrossRef]
115. Soliman, J.N.; Mageed, T.A.; El-Hennawy, H.M. Digital signature and authentication mechanisms using new customized hash function for cognitive radio networks. In Proceedings of the 2017 12th International Conference on Computer Engineering and Systems (ICCES), Cairo, Egypt, 19–20 December 2017; pp. 175–181. [CrossRef]
116. Liu, Y.; Wang, Y.; Chang, G. Efficient Privacy-Preserving Dual Authentication and Key Agreement Scheme for Secure V2V Communications in an IoV Paradigm. *IEEE Trans. Intell. Transp. Syst.* **2017**, *18*, 2740–2749. [CrossRef]

117. Singh, P.K.; Saikamal Tabjul, G.; Imran, M.; Nandi, S.K.; Nandi, S. Impact of Security Attacks on Cooperative Driving Use Case: CACC Platooning. In Proceedings of the TENCON 2018—2018 IEEE Region 10 Conference, Jeju, Korea (South), 28–31 October 2018; pp. 0138–0143. [CrossRef]
118. Mohan, A.P.; Elshakankiri, M. Enhanced Priority-Based Routing Protocol (EPRP) for Inter-vehicular Communication. In *Advances in Data Science, Cyber Security and IT Applications*; Alfaries, A., Mengash, H., Yasar, A., Shakshuki, E., Eds.; Springer International Publishing: Cham, Switzerland, 2019; pp. 325–337.
119. den Hartog, J.; Zannone, N.; den Hartog, J. Security and privacy for innovative automotive applications: A survey. *Comput. Commun.* **2018**, *132*, 17–41.
120. Anwar, W.; Franchi, N.; Fettweis, G. Physical layer evaluation of V2X communications technologies: 5G NR-V2X, LTE-V2X, IEEE 802.11 bd, and IEEE 802.11 p. In Proceedings of the 2019 IEEE 90th Vehicular Technology Conference (VTC2019-Fall), Honolulu, HI, USA, 22–25 September 2019; pp. 1–7.
121. Yoshizawa, T.; Preneel, B. Survey of security aspect of v2x standards and related issues. In Proceedings of the 2019 IEEE Conference on Standards for Communications and Networking (CSCN), Granada, Spain, 28–30 October 2019; pp. 1–5.
122. Koyama, T.; Shibahara, T.; Hasegawa, K.; Okano, Y.; Tanaka, M.; Oshima, Y. Anomaly Detection for Mixed Transmission CAN Messages Using Quantized Intervals and Absolute Difference of Payloads. In Proceedings of the ACM Workshop on Automotive Cybersecurity, AutoSec@CODASPY 2019, Richardson, TX, USA, 27 March 2019; Zhao, Z., Chen, Q.A., Ahn, G., Eds.; ACM: New York, NY, USA, 2019; pp. 19–24. [CrossRef]
123. Chatzoglou, E.; Kambourakis, G.; Kouliaridis, V. A Multi-Tier Security Analysis of Official Car Management Apps for Android. *Future Internet* **2021**, *13*, 58. [CrossRef]
124. Chatzoglou, E.; Kambourakis, G.; Kolias, C. How is your Wi-Fi connection today? DoS attacks on WPA3-SAE. *J. Inf. Secur. Appl.* **2022**, *64*, 103058. [CrossRef]
125. Classen, J.; Gringoli, F.; Hermann, M.; Hollick, M. Attacks on Wireless Coexistence: Exploiting Cross-Technology Performance Features for Inter-Chip Privilege Escalation. 2021. Available online: http://xxx.lanl.gov/abs/2112.05719 (accessed on 3 March 2022).
126. Bitsikas, E.; Pöpper, C. Don't Hand It Over: Vulnerabilities in the Handover Procedure of Cellular Telecommunications. In *Annual Computer Security Applications Conference*; Association for Computing Machinery: New York, NY, USA, 2021; pp. 900–915. [CrossRef]
127. Fernandes, E.; Crispo, B.; Conti, M. FM 99.9, Radio Virus: Exploiting FM Radio Broadcasts for Malware Deployment. *IEEE Trans. Inf. Forensics Secur.* **2013**, *8*, 1027–1037. [CrossRef]
128. Abdullah, H.; Garcia, W.; Peeters, C.; Traynor, P.; Butler, K.R.B.; Wilson, J. Practical Hidden Voice Attacks against Speech and Speaker Recognition Systems. 2019. Available online: http://xxx.lanl.gov/abs/1904.05734 (accessed on 3 March 2022).
129. Kolias, C.; Kambourakis, G.; Stavrou, A.; Voas, J.M. DDoS in the IoT: Mirai and Other Botnets. *Computer* **2017**, *50*, 80–84. [CrossRef]
130. *ISO 20077:2017(en)*; Road Vehicles—Extended Vehicle (ExVe) Methodology—Part 1: General Information. International Organization for Standardization: Geneva, Switzerland, 2017.
131. United Nations. UN Regulation No. 155, Uniform Provisions Concerning the Approval of Vehicles with Regards to Cyber Security and Cyber Security Management System. 2020. Available online: https://unece.org/transport/documents/2021/03/standards/un-regulation-no-155-cyber-security-and-cyber-security (accessed on 3 March 2022).
132. United Nations. UN Regulation No. 156, Uniform Provisions Concerning the Approval of Vehicles with Regards to Software Update and Software Updates Management System. 2020. Available online: https://unece.org/transport/documents/2021/03/standards/un-regulation-no-156-software-update-and-software-update (accessed on 3 March 2022).
133. Stiftung Warentest. Connected Cars: Apps of the Automobile Manufacturer Are Data Sniffers (Connected Cars: Die Apps der Autohersteller sind Datenschnüffler). 2017. Available online: https://www.test.de/Connected-Cars-Die-Apps-der-Autohersteller-sind-Datenschnueffler-5231839-0/ (accessed on 3 March 2022).
134. Commission, E. Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the Protection of Natural Persons with Regard to the Processing of Personal Data and on the Free Movement of Such Data, and Repealing Directive 95/46/EC (General Data Protection Regulation) 2016. Available online: https://eur-lex.europa.eu/eli/reg/2016/679/oj (accessed on 3 March 2022).
135. *ISO 26262:2018(en)*; Road Vehicles—Functional Safety. International Organization for Standardization: Geneva, Switzerland, 2018.
136. Bell, R. Introduction and Revision of IEC 61508. In *Advances in Systems Safety*; Springer: Berlin/Heidelberg, Germany, 2011; pp. 273–291.
137. Dürrwang, J.; Beckers, K.; Kriesten, R. A Lightweight Threat Analysis Approach Intertwining Safety and Security for the Automotive Domain. In *Computer Safety, Reliability, and Security*; Tonetta, S., Schoitsch, E., Bitsch, F., Eds.; Springer International Publishing: Cham, Switzerland, 2017; pp. 305–319.
138. Martin, H.; Ma, Z.; Schmittner, C.; Winkler, B.; Krammer, M.; Schneider, D.; Amorim, T.; Macher, G.; Kreiner, C. Combined automotive safety and security pattern engineering approach. *Reliab. Eng. Syst. Saf.* **2020**, *198*, 106773. [CrossRef]
139. Radanliev, P.; Roure, D.D.; Burnap, P.; Santos, O. Epistemological Equation for Analysing Uncontrollable States in Complex Systems: Quantifying Cyber Risks from the Internet of Things. *Rev. Socionetwork Strateg.* **2021**, *15*, 381–411. [CrossRef]
140. Schmittner, C.; Ma, Z.; Reyes, C.; Dillinger, O.; Puschner, P. Using SAE J3061 for automotive security requirement engineering. In *International Conference on Computer Safety, Reliability, and Security*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 157–170.

141. ITU, I.T.T.S.S.O. Guidelines for an Intrusion Detection System for In-Vehicle Networks 2020. https://www.itu.int/rec/T-REC-X.1375-202010-I (accessed on 3 March 2022).
142. Kang, M.J.; Kang, J. A Novel Intrusion Detection Method Using Deep Neural Network for In-Vehicle Network Security. In Proceedings of the IEEE 83rd Vehicular Technology Conference, VTC Spring 2016, Nanjing, China, 15–18 May 2016; pp. 1–5. [CrossRef]
143. Kneib, M.; Huth, C. Scission: Signal Characteristic-Based Sender Identification and Intrusion Detection in Automotive Networks. In Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security, CCS 2018, Toronto, ON, Canada, 15–19 October 2018; Lie, D., Mannan, M., Backes, M., Wang, X., Eds.; ACM: New York, NY, USA, 2018; pp. 787–800. [CrossRef]
144. Alshamrani, A.; Myneni, S.; Chowdhary, A.; Huang, D. A Survey on Advanced Persistent Threats: Techniques, Solutions, Challenges, and Research Opportunities. *IEEE Commun. Surv. Tutor.* **2019**, *21*, 1851–1877. [CrossRef]
145. Papernot, N.; McDaniel, P.; Goodfellow, I.; Jha, S.; Celik, Z.B.; Swami, A. Practical Black-Box Attacks against Machine Learning. In Proceedings of the ASIA CCS '17: 2017 ACM on Asia Conference on Computer and Communications Security, Abu Dhabi, United Arab Emirates, 2–6 April 2017; Association for Computing Machinery: New York, NY, USA, 2017; pp. 506–519. [CrossRef]
146. Cao, Y.; Xiao, C.; Cyr, B.; Zhou, Y.; Park, W.; Rampazzi, S.; Chen, Q.A.; Fu, K.; Mao, Z.M. Adversarial Sensor Attack on LiDAR-Based Perception in Autonomous Driving. In Proceedings of the CCS '19: 2019 ACM SIGSAC Conference on Computer and Communications Security, London, UK, 11–15 November 2019; Association for Computing Machinery: New York, NY, USA, 2019; pp. 2267–2281. [CrossRef]
147. Biggio, B.; Corona, I.; Maiorca, D.; Nelson, B.; Srndic, N.; Laskov, P.; Giacinto, G.; Roli, F. Evasion Attacks against Machine Learning at Test Time. In Proceedings of the Machine Learning and Knowledge Discovery in Databases—European Conference, ECML PKDD 2013, Prague, Czech Republic, 23–27 September 2013; Proceedings, Part III; Blockeel, H., Kersting, K., Nijssen, S., Zelezný, F., Eds.; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2013; Volume 8190, pp. 387–402._25. [CrossRef]
148. Talpur, A.; Gurusamy, M. Machine Learning for Security in Vehicular Networks: A Comprehensive Survey. *IEEE Commun. Surv. Tutor.* **2021**, *24*, 346–379. [CrossRef]
149. Wang, X.; Li, J.; Kuang, X.; Tan, Y.A.; Li, J. The security of machine learning in an adversarial setting: A survey. *J. Parallel Distrib. Comput.* **2019**, *130*, 12–23. [CrossRef]
150. *ISO/IEC 15408-1:2009*; Information Technology—Security Techniques—Evaluation Criteria for IT Security—Part 1: Introduction and General Model. International Organization for Standardization: Geneva, Switzerland, 2009.
151. Common Criteria. Arrangement on the Recognition of Common Criteria Certificates In the Field of Information Technology Security. 2014. Available online: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.687.974&rep=rep1&type=pdf (accessed on 7 March 2022).
152. Matheu, S.N.; Hernandez-Ramos, J.L.; Skarmeta, A.F.; Baldini, G. A survey of cybersecurity certification for the internet of things. *ACM Comput. Surv. (CSUR)* **2020**, *53*, 1–36. [CrossRef]
153. *ISO/SAE 21434:2021*; Road Vehicles—Cybersecurity Engineering. International Organization for Standardization: Geneva, Switzerland, 2021.
154. Omitola, T.; Wills, G. Towards Mapping the Security Challenges of the Internet of Things (IoT) Supply Chain. *Procedia Comput. Sci.* **2018**, *126*, 441–450. [CrossRef]
155. Neisse, R.; Hernández-Ramos, J.L.; Matheu-Garcia, S.N.; Baldini, G.; Skarmeta, A.; Siris, V.; Lagutin, D.; Nikander, P. An interledger blockchain platform for cross-border management of cybersecurity information. *IEEE Internet Comput.* **2020**, *24*, 19–29. [CrossRef]
156. Seri, B.; Vishnepolsky, G.; Zusman, D. Critical vulnerabilities to remotely compromise VxWorks, the most popular RTOS. In *White Paper, ARMIS, URGENT/11*; 2019. Available online: https://www.armis.com/research/urgent11/ (accessed on 3 March 2022).
157. Nasahl, P.; Timmers, N. Attacking AUTOSAR using Software and Hardware Attacks. *ESCAR USA*, 2019. Available online: ttps://www.riscure.com/publication/attacking-autosar-using-software-and-hardware-attacks (accessed on 3 March 2022).
158. Wollschlaeger, M.; Sauter, T.; Jasperneite, J. The Future of Industrial Communication: Automation Networks in the Era of the Internet of Things and Industry 4.0. *IEEE Ind. Electron. Mag.* **2017**, *11*, 17–27. [CrossRef]
159. Bello, L.L. The case for ethernet in automotive communications. *SIGBED Rev.* **2011**, *8*, 7–15. [CrossRef]
160. Zago, G.M.; de Freitas, E.P. A quantitative performance study on CAN and CAN FD vehicular networks. *IEEE Trans. Ind. Electron.* **2017**, *65*, 4413–4422. [CrossRef]
161. Pimentel, J.R.; Fonseca, J.A. FlexCAN: A Flexible Architecture for highly dependable embedded applications. In Proceedings of the 3rd International Workshop on Real-Time Networks, Catania, Italy, 2–5 July 2004; IEEE Press: New York, NY, USA, 2004.
162. *IEEE 802.3ch-2020*; IEEE Standard for Ethernet–Amendment 8:Physical Layer Specifications and Management Parameters for 2.5 Gb/s, 5 Gb/s, and 10 Gb/s Automotive Electrical Ethernet. IEEE Press: New York, NY, USA, 2020.
163. *IEEE 802.3-2018*; IEEE Standard for Ethernet. IEEE, New York, NY, USA, 2018.
164. *ISO 13400-2:2019*; Road Vehicles—Diagnostic Communication over Internet Protocol (DoIP)—Part 2: Transport Protocol and Network Layer Services. International Organization for Standardization: Geneva, Switzerland, 2019.
165. Baldini, G.; Hernández-Ramos, J.L.; Steri, G.; Neisse, R.; Fovino, I.N. A Review on the Application of Distributed Ledgers in the Evolution of Road Transport. *IEEE Internet Comput.* **2020**, *24*, 27–36. [CrossRef]

166. Liang, H.; Wu, J.; Mumtaz, S.; Li, J.; Lin, X.; Wen, M. MBID: Micro-Blockchain-Based Geographical Dynamic Intrusion Detection for V2X. *IEEE Commun. Mag.* **2019**, *57*, 77–83. [CrossRef]
167. Sharma, T.; Satija, S.; Bhushan, B. Unifying Blockchian and IoT:Security Requirements, Challenges, Applications and Future Trends. In Proceedings of the 2019 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS), Greater Noida, India, 18–19 October 2019; pp. 341–346. [CrossRef]
168. Forestiero, A. Self-organizing anomaly detection in data streams. *Inf. Sci.* **2016**, *373*, 321–336. [CrossRef]
169. Forestiero, A. Metaheuristic algorithm for anomaly detection in Internet of Things leveraging on a neural-driven multiagent system. *Knowl. Based Syst.* **2021**, *228*, 107241. [CrossRef]
170. Kiran, B.R.; Sobh, I.; Talpaert, V.; Mannion, P.; Al Sallab, A.A.; Yogamani, S.; Pérez, P. Deep reinforcement learning for autonomous driving: A survey. *IEEE Trans. Intell. Transp. Syst.* **2021**. [CrossRef]
171. Sedar, R.; Kalalas, C.; Vázquez-Gallego, F.; Alonso-Zarate, J. Reinforcement Learning-based Misbehaviour Detection in V2X Scenarios. In Proceedings of the 2021 IEEE International Mediterranean Conference on Communications and Networking (MeditCom), Athens, Greece, 5–8 July 2021; pp. 109–111.
172. McMahan,Brendan and Ramage,Daniel. Federated Learning: Collaborative Machine Learning Without Centralized Training Data. 2017. Available online: https://ai.googleblog.com/2017/04/federated-learning-collaborative.html (accessed on 3 March 2022).
173. Kolias, C.; Kolias, V.; Kambourakis, G. TermID: A distributed swarm intelligence-based approach for wireless intrusion detection. *Int. J. Inf. Sec.* **2017**, *16*, 401–416. [CrossRef]
174. Aledhari, M.; Razzak, R.; Parizi, R.M.; Saeed, F. Federated learning: A survey on enabling technologies, protocols, and applications. *IEEE Access* **2020**, *8*, 140699–140725. [CrossRef] [PubMed]
175. Du, Z.; Wu, C.; Yoshinaga, T.; Yau, K.L.A.; Ji, Y.; Li, J. Federated learning for vehicular internet of things: Recent advances and open issues. *IEEE Open J. Comput. Soc.* **2020**, *1*, 45–61. [CrossRef] [PubMed]
176. Elbir, A.M.; Soner, B.; Coleri, S. Federated learning in vehicular networks. *arXiv* **2020**, arXiv:2006.01412.
177. Campos, E.M.; Saura, P.F.; González-Vidal, A.; Hernández-Ramos, J.L.; Bernabe, J.B.; Baldini, G.; Skarmeta, A. Evaluating Federated Learning for Intrusion Detection in Internet of Things: Review and Challenges. *arXiv* **2021**, arXiv:2108.00974.
178. Ruzafa-Alcazar, P.; Fernandez-Saura, P.; Marmol-Campos, E.; Gonzalez-Vidal, A.; Ramos, J.L.H.; Bernal, J.; Skarmeta, A.F. Intrusion Detection based on Privacy-preserving Federated Learning for the Industrial IoT. *IEEE Trans. Ind. Inform.* **2021**. [CrossRef]
179. Posner, J.; Tseng, L.; Aloqaily, M.; Jararweh, Y. Federated learning in vehicular networks: Opportunities and solutions. *IEEE Netw.* **2021**, *35*, 152–159. [CrossRef]
180. Wahab, O.A.; Mourad, A.; Otrok, H.; Bentahar, J. CEAP: SVM-based intelligent detection model for clustered vehicular ad hoc networks. *Expert Syst. Appl.* **2016**, *50*, 40–54. [CrossRef]
181. Dzambic, M.; Kreuzberger, C.; Veledar, O.; Macher, G. A Rapid Prototyping System, Intelligent Watchdog and Gateway Tool for Automotive Applications. In Proceedings of the 2021 IEEE 18th International Conference on Software Architecture Companion (ICSA-C), Stuttgart, Germany, 22–26 March 2021; pp. 149–154. [CrossRef]
182. Panda, S.; Rass, S.; Moschoyiannis, S.; Liang, K.; Loukas, G.; Panaousis, E. HoneyCar: A Framework to Configure Honeypot Vulnerabilities on the Internet of Vehicles. 2021. Available online: http://xxx.lanl.gov/abs/2111.02364 (accessed on 3 March 2022).
183. Liu, L.; Chen, C.; Pei, Q.; Maharjan, S.; Zhang, Y. Vehicular edge computing and networking: A survey. *Mob. Networks Appl.* **2021**, *26*, 1145–1168. [CrossRef]
184. Grover, H.; Alladi, T.; Chamola, V.; Singh, D.; Choo, K.K.R. Edge Computing and Deep Learning Enabled Secure Multitier Network for Internet of Vehicles. *IEEE Internet Things J.* **2021**, *8*, 14787–14796. [CrossRef]
185. Liu, H.; Zhang, S.; Zhang, P.; Zhou, X.; Shao, X.; Pu, G.; Zhang, Y. Blockchain and Federated Learning for Collaborative Intrusion Detection in Vehicular Edge Computing. *IEEE Trans. Veh. Technol.* **2021**, *70*, 6073–6084. [CrossRef]
186. Tavallaee, M.; Bagheri, E.; Lu, W.; Ghorbani, A.A. A detailed analysis of the KDD CUP 99 data set. In Proceedings of the 2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications, Ottawa, OT, Canada, 8–10 July 2009; pp. 1–6.
187. Statista. Automotive Electronics Cost as a Percentage of Total Car Cost Worldwide from 1950 to 2030. 2022. Available online: ttps://www.statista.com/statistics/277931/automotive-electronics-cost-as-a-share-of-total-car-cost-worldwide/ (accessed on 3 March 2022).